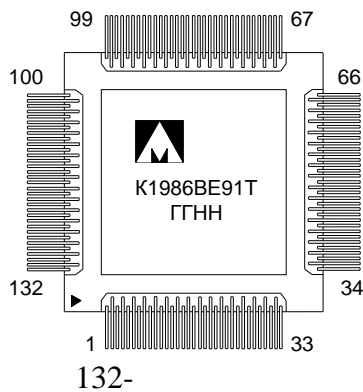


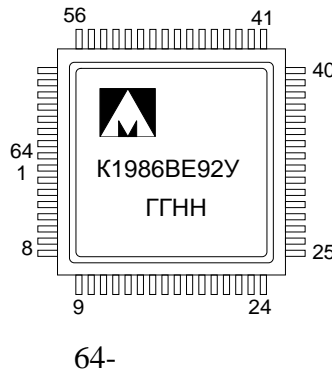


**Микросхема 32-разрядного однокристального микро-ЭВМ  
с памятью Flash-типа 1986BE9ху, К1986BE9ху, К1986BE9хуК  
К1986BE92QI, К1986BE92QC, 1986BE91Н4, К1986BE91Н4,  
1986BE94Н4, К1986BE94Н4**

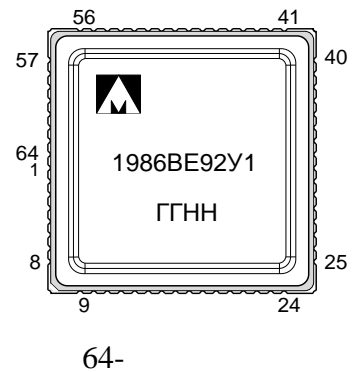
**1986BE91T, 1986BE94T**



**1986BE92Y**

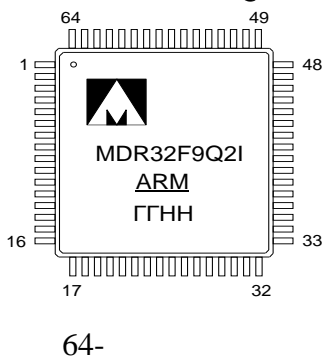


**1986BE92Y1**



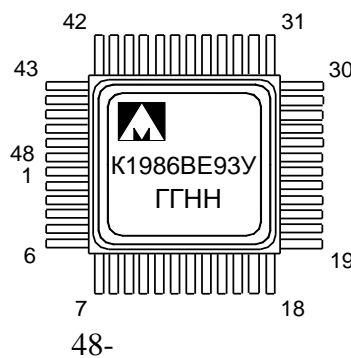
4229.132-3

**К1986BE92Q**



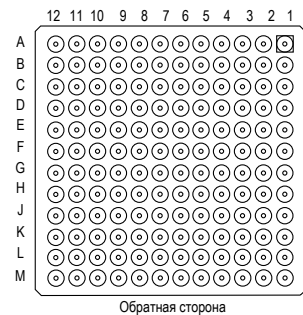
Н18.64-1

**1986BE93Y**



5153.64-2

**1986BE94Ф**



LQFP64

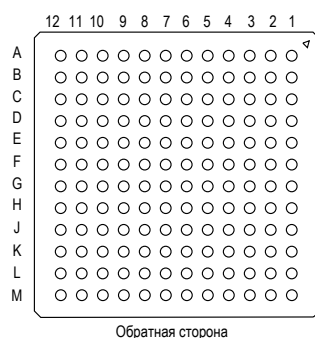
48-

144-

16.48-1

6109.144-

**1986BE94Я**



144-

8307.144-

Обозначение	Температурный диапазон
1986 9	60 ÷ 125 °
1986 9	60 ÷ 125 °
1986 9 К	0 ÷ 70 °
1986 9	0 ÷ 70 °
1986 9	0 ÷ 70 °
1986 92Q	0 ÷ 70 °
1986 92QI	40 ÷ 85 °

— = 1, 2, 3, 4  
— = : , , 1,

1986 9 у K1986BE9хуK

1986 9 уK, K1986BE9хГуK

немоноотонным АЦП

|-1...+2|

K1986BE9хДуK 1986 9 уK,

немоноотонным ЦАП

|-1...+2|

K1986BE92QC K1986BE92QI

1986 92

1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4

## Основные характеристики микроконтроллеров серии 1986BE9х

### Ядро:

– ARM 32- RISC- Cortex™-M3 2.0, 80 ;  
 1.25 DMIPS/ (Dhrystone 2.1)  
 – MPU;

### Память:

– Flash- 128 ;  
 – 32 ;  
 – , NAND Flash.

### Питание и тактовая частота:

– 2,2 ÷ 3,6 ;  
 – 1,8 ;  
 – ;  
 – RC 8 40 ;  
 – 2 ÷ 16 32 ;  
 – PLL ;  
 – PLL USB.

### Режим пониженного энергопотребления:

– Sleep, Deep Sleep Standby;

### Аналоговые модули:

– 12- ( 16 );  
 – ;  
 – 12- ;

### Периферия:

– DMA - , - ;  
 – CAN ;  
 – USB Device Host;  
 – UART, SPI, I2C;  
 – 16- ;  
 – 96

### Отладочные интерфейсы:

– SWD JTAG.

## Содержание

1	.....	7
2	.....	8
3	- .....	9
4	.....	10
4.1	.....	26
5	.....	29
6	.....	31
6.1	.....	33
7	.....	35
7.1	CODE .....	37
7.2	DATA .....	37
7.3	PERIPHERAL .....	37
7.4	EXTERNAL RAM .....	38
7.5	SYSTEM .....	38
7.6	BUS MATRIX .....	38
7.7	BOOT ROM .....	38
7.8	EEPROM .....	38
7.9	SRAM .....	39
7.10	.....	39
7.11	.....	40
7.12	.....	40
7.13	.....	41
7.14	Bit-band .....	43
7.15	.....	44
7.16	.....	46
7.17	.....	47
8	.....	49
8.1	UART .....	51
9	Flash- MDR_EEPROM .....	56
9.1	Flash- .....	56
9.2	Flash- .....	57
9.3	Flash- .....	61
10	ARM Cortex-M3 .....	65
10.1	.....	66
10.2	.....	67
10.3	.....	67
10.4	.....	75
11	.....	76
11.1	.....	80
11.2	.....	81
11.3	.....	89
11.4	.....	105
11.5	.....	119
11.6	.....	123
11.7	.....	125
11.8	.....	129
11.9	.....	137
12	SysTick .....	149

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

12.1		SysTick.....	149
12.2		.....	152
13	MPU .....		153
13.1	MPU .....		154
13.2	MPU.....		163
14	MDR_RST_CLK .....		165
14.1	.....		167
15	MDR_BKP.....		180
15.1	.....		180
15.2	.....		181
15.3	.....		181
16	MDR_PORTx.....		190
16.1	-	.....	193
17	MDR_POWER .....		197
18	MDR_EBC ( 1986 94 ).....		200
18.1	,	.....	200
18.2	NAND Flash- .....		203
18.3	.....		206
19	MDR_EBC ( 1986 94 ).....		209
19.1	,	.....	209
19.2	NAND Flash- .....		212
19.3	.....		215
20	MDR_USB.....		220
20.1	.....		220
20.2	USB / .....		220
20.3	.....		221
20.4	IN (USB Device).....		222
20.5	SETUP/OUT (USB Device).....		224
20.6	SETUP/OUT (USB Host) .....		225
20.7	IN (USB Host).....		227
20.8	SOF (USB Host) .....		228
20.9	USB .....		228
21	MDR_CAN .....		248
21.1	.....		249
21.2	.....		250
21.3	(Data Frame) .....		251
21.4	.....		255
21.5	.....		256
21.6	Remote Transmit Request (RTR) .....		256
21.7	.....		256
21.8	.....		257
21.9	.....		257
21.10	.....		257
21.11	.....		258
21.12	.....		259
21.13	.....		262
21.14	CAN .....		263
22	MDR_TIMERx.....		278
22.1	.....		278
22.2	.....		280
22.3	.....		282

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

22.4	.....	289
22.5	.....	291
22.6	.....	294
22.7	.....	298
23	MDR_ADC.....	314
23.1	.....	315
23.2	.....	316
23.3	.....	316
23.4	.....	316
23.5	.....	317
23.6	.....	317
23.7	.....	318
23.8	.....	320
24	MDR_DAC.....	328
24.1	.....	328
25	MDR_COMP .....	331
25.1	.....	332
25.2	c .....	332
25.3	c .....	332
25.4	.....	332
25.5	.....	334
26	MDR_I2C .....	336
26.1	.....	336
26.2	I2C .....	336
26.3	START.....	337
26.4	.....	337
26.5	.....	337
26.6	STOP .....	338
26.7	I2C.....	338
27	MDR_SSP.....	342
27.1	SSP .....	342
27.2	.....	342
27.3	SPI .....	343
27.4	Microwire.....	343
27.5	SSI .....	344
27.6	SSP.....	344
27.7	.....	362
27.8	.....	368
28	MDR_UART .....	370
28.1	.....	370
28.2	.....	371
28.3	UART .....	373
28.4	IrDA SIR .....	375
28.5	UART.....	377
28.6	.....	382
28.7	.....	383
28.8	.....	386
28.9	.....	388
29	MDR_DMA .....	406
29.1	DMA.....	406
29.2	.....	406

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

---

29.3	.....	408
29.4	DMA .....	411
29.5	.....	433
29.6	DMA .....	444
30	.....	464
30.1	.....	464
30.2	(IRQ) .....	465
30.3	.....	466
30.4	.....	466
30.5	.....	468
30.6	.....	468
30.7	.....	471
30.8	.....	473
31	NVIC.....	476
31.1	.....	477
31.2	.....	482
31.3	.....	482
31.4	.....	483
32	.....	484
32.1	.....	485
33	.....	501
33.1	.....	501
34	.....	506
35	.....	509
36	.....	511
37	.....	515
38	.....	520
39	.....	531
40	.....	532

## 1 Введение

1986 9х),  
Cortex-M3,

1986 9х,  
128

1986 9  
Flash-  
80

1986 92QI,  
RISC  
32

1986 92QC (  
ARM

UART, SPI I2C,  
24

USB  
12 / (Full Speed) 1,5 / (Low Speed),  
NAND Flash-  
16-  
« »  
16  
12- ; 12- ;  
( 0,5 / )

(2...16 ) RC LSE (32 ) HSI (8 ) LSI (40 ) PLL HSE USB

DMA

1,8  
2,2 3,6

## 2 Основные характеристики

**Таблица 1 – Основные характеристики микроконтроллеров серии 1986BE9х**

	1986BE91T 1986BE94T	1986BE94Ф 1986BE94Я	K1986BE91H4	1986BE92Y 1986BE92Y1 K1986BE92QI K1986BE92QC	1986BE93Y
<b>Корпус</b>	132	144		64	48
<b>Ядро</b>	ARM Cortex-M3				
<b>ПЗУ</b>	128 Flash				
<b>ОЗУ</b>	32				
<b>Питание</b>	2,2...3,6				
<b>Частота</b>	80				
<b>USER IO</b>	96	96	96	43	30
<b>USB</b>	Device Host FS ( 12 / )				PHY
<b>UART</b>	2	2	2	2	2
<b>CAN</b>	2	2	2	2	2
<b>SPI</b>	2	2	2	2	1
<b>I2C</b>	1	1	1	1	-
<b>2 x 12-разрядных АЦП</b>	16	16	16	8	4
<b>ЦАП 12 разрядов</b>	2	2	2	1	1
<b>Компаратор</b>	3	3	3	2	2
<b>Внешняя шина</b>	32	32	32	8	-



### 3 Структурная блок-схема микросхемы

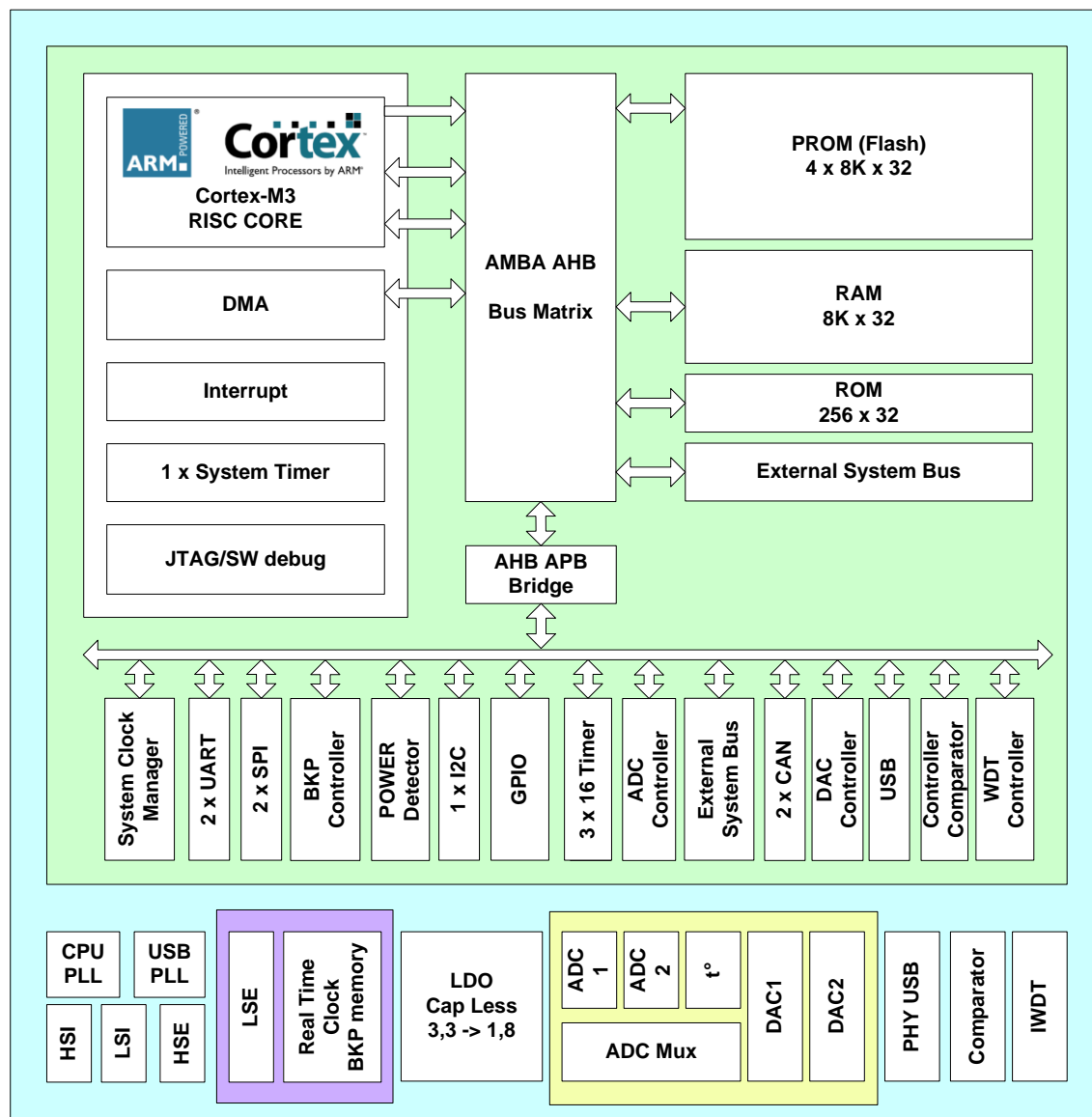


Рисунок 1. Структурная блок-схема микроконтроллера 1986BE9ху

## 4 Описание выводов

**Таблица 2 – Описание выводов микроконтроллеров серии 1986BE9х  
в корпусах 4229.132-3, Н18.64-1В, Н16.48-1В, МК 5153.64-2, LQFP64**

Вывод	Контактная площадка кристалла	Тип корпуса					Дополнительные функции вывода			
		4229.132-3	Н18.64-1В	Н16.48-1В	МК 5153.64-2	LQFP64	Аналог.	Основ.	Альтер.	Переопр.
<b>Порт А</b>										
PA0	137	130	55	41	55	63	-	DATA0	EXT_INT1	-
PA1	136	129	54	40	54	62	-	DATA1	TMR1_CH1	TMR2_CH1
PA2	135	128	53	39	53	61	-	DATA2	TMR1_CH1N	TMR2_CH1N
PA3	134	127	52	38	52	60	-	DATA3	TMR1_CH2	TMR2_CH2
PA4	133	126	51	37	51	59	-	DATA4	TMR1_CH2N	TMR2_CH2N
PA5	132	125	50	36	50	58	-	DATA5	TMR1_CH3	TMR2_CH3
PA6	131	124	49	35	49	57	-	DATA6	CAN1_TX	UART1_RXD
PA7	130	123	48	34	48	56	-	DATA7	CAN1_RX	UART1_TXD
PA8	129	122	-	-	-	-	-	DATA8	TMR1_CH3N	TMR2_CH3N
PA9	128	121	-	-	-	-	-	DATA9	TMR1_CH4	TMR2_CH4
PA10	125	119	-	-	-	-	-	DATA10	nUART1DTR	TMR2_CH4N
PA11	124	118	-	-	-	-	-	DATA11	nUART1RTS	TMR2_BLK
PA12	123	117	-	-	-	-	-	DATA12	nUART1RI	TMR2_ETR
PA13	122	115	-	-	-	-	-	DATA13	nUART1DCD	TMR1_CH4N
PA14	121	114	-	-	-	-	-	DATA14	nUART1DSR	TMR1_BLK
PA15	120	113	-	-	-	-	-	DATA15	nUART1CTS	TMR1_ETR
<b>Порт В</b>										
PB0 JA_TDO	97	92	35	25	35	43	-	DATA16	TMR3_CH1	UART1_TXD
PB1 JA_TMS	98	93	36	26	36	44	-	DATA17	TMR3_CH1N	UART2_RXD
PB2 JA_TCK	99	94	37	27	37	45	-	DATA18	TMR3_CH2	CAN1_TX
PB3 JA_TDI	100	95	38	28	38	46	-	DATA19	TMR3_CH2N	CAN1_RX
PB4 JA_TRST	101	96	39	29	39	47	-	DATA20	TMR3_BLK	TMR3_ETR
PB5	107	102	42	32	42	50	-	DATA21	UART1_TXD	TMR3_CH3
PB6	108	103	43	33	43	51	-	DATA22	UART1_RXD	TMR3_CH3N
PB7	109	104	44	-	44	52	-	DATA23	nSIROUT1	TMR3_CH4
PB8	110	105	45	-	45	53	-	DATA24	COMP_OUT	TMR3_CH4N
PB9	111	106	46	-	46	54	-	DATA25	nSIRIN1	EXT_INT4
PB10	112	107	47	-	47	55	-	DATA26	EXT_INT2	nSIROUT1
PB11	113	108	-	-	-	-	-	DATA27	EXT_INT1	COMP_OUT
PB12	114	109	-	-	-	-	-	DATA28	SSP1_FSS	SSP2_FSS
PB13	115	110	-	-	-	-	-	DATA29	SSP1_CLK	SSP2_CLK
PB14	116	111	-	-	-	-	-	DATA30	SSP1_RXD	SSP2_RXD
PB15	119	112	-	-	-	-	-	DATA31	SSP1_TXD	SSP2_TXD
<b>Порт С</b>										
PC0	96	91	34	23	34	42	-	READY*	SCL1	SSP2_FSS
PC1	95	90	33	-	33	41	-	OE	SDA1	SSP2_CLK
PC2	94	89	31	-	31	40	-	WE	TMR3_CH1	SSP2_RXD
PC3	93	88	-	-	-	-	-	BE0	TMR3_CH1N	SSP2_TXD
PC4	92	87	-	-	-	-	-	BE1	TMR3_CH2	TMR1_CH1
PC5	91	86	-	-	-	-	-	BE2	TMR3_CH2N	TMR1_CH1N
PC6	90	85	-	-	-	-	-	BE3	TMR3_CH3	TMR1_CH2
PC7	89	84	-	-	-	-	-	CLOCK	TMR3_CH3N	TMR1_CH2N
PC8	88	83	-	-	-	-	-	CAN1_TX	TMR3_CH4	TMR1_CH3
PC9	87	82	-	-	-	-	-	CAN1_RX	TMR3_CH4N	TMR1_CH3N

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Вывод	Контактная площадка кристалла	Тип корпуса					Дополнительные функции вывода			
		4229.132-3	H18.64-1B	H16.48-1B	МК 5153.64-2	LQFP64	Аналог.	Основ.	Альтер.	Переопр.
PC10	86	81	-	-	-	-	-	-	TMR3_ETR	TMR1_CH4
PC11	85	80	-	-	-	-	-	-	TMR3_BLK	TMR1_CH4N
PC12	84	79	-	-	-	-	-	-	EXT_INT2	TMR1_ETR
PC13	83	78	-	-	-	-	-	-	EXT_INT4	TMR1_BLK
PC14	82	77	-	-	-	-	-	-	SSP2_FSS	CAN2_RX
PC15	81	76	-	-	-	-	-	-	SSP2_RXD	CAN2_TX
<b>Порт D</b>										
PD0 JB_TMS	70	65	23	17	23	31	ADC0_R EF+	TMR1_CH1 N	UART2_RXD	TMR3_CH1
PD1 JB_TCK	71	66	24	18	24	32	ADC1_R EF-	TMR1_CH1	UART2_TXD	TMR3_CH1N
PD2 JB_TRST	72	67	25	19	25	33	ADC2	BUSY1	SSP2_RXD	TMR3_CH2
PD3 JB_TDI	73	68	26	20	26	34	ADC3	-	SSP2_FSS	TMR3_CH2N
PD4 JB_TDO	69	64	22	-	22	30	ADC4	TMR1_ETR	nSIROUT2	TMR3_BLK
PD5	74	69	27	-	27	35	ADC5	CLE	SSP2_CLK	TMR2_ETR
PD6	75	70	28	-	28	36	ADC6	ALE	SSP2_TXD	TMR2_BLK
PD7	68	63	21	-	21	29	ADC7	TMR1_BLK K	nSIRIN2	UART1_RXD
PD8	67	62	-	-	-	-	ADC8	TMR1_CH4 N	TMR2_CH1	UART1_TXD
PD9	76	71	-	-	-	-	ADC9	CAN2_TX	TMR2_CH1N	SSP1_FSS
PD10	66	61	-	-	-	-	ADC10	TMR1_CH2	TMR2_CH2	SSP1_CLK
PD11	65	60	-	-	-	-	ADC11	TMR1_CH2 N	TMR2_CH2N	SSP1_RXD
PD12	64	59	-	-	-	-	ADC12	TMR1_CH3	TMR2_CH3	SSP1_TXD
PD13	63	58	-	-	-	-	ADC13	TMR1_CH3 N	TMR2_CH3N	CAN1_TX
PD14	62	57	-	-	-	-	ADC14	TMR1_CH4	TMR2_CH4	CAN1_RX
PD15	61	56	-	-	-	-	ADC15	CAN2_RX	BUSY2	EXT_INT3
<b>Порт E</b>										
PE0	56	53	18	14	18	26	DAC2_O UT	ADDR16	TMR2_CH1	CAN1_RX
PE1	55	52	17	-	17	25	DAC2_R EF	ADDR17	TMR2_CH1N	CAN1_TX
PE2	48	45	14	11	14	22	COMP_I N1	ADDR18	TMR2_CH3	TMR3_CH1
PE3	47	44	13	10	13	21	COMP_I N2	ADDR19	TMR2_CH3N	TMR3_CH1N
PE4	45	42	-	-	-	-	COMP_R EF+	ADDR20	TMR2_CH4N	TMR3_CH2
PE5	44	41	-	-	-	-	COMP_R EF-	ADDR21	TMR2_BLK	TMR3_CH2N
PE6	36	33	8	6	8	16	OSC_IN3 2	ADDR22	CAN2_RX	TMR3_CH3
PE7	35	32	7	-	7	15	OSC_OU T32	ADDR23	CAN2_TX	TMR3_CH3N
PE8	46	43	-	-	-	-	COMP_I N3	ADDR24	TMR2_CH4	TMR3_CH4
PE9	54	51	-	-	-	-	DAC1_O UT	ADDR25	TMR2_CH2	TMR3_CH4N
PE10	53	50	-	-	-	-	DAC1_R EF	ADDR26	TMR2_CH2N	TMR3_ETR
PE11	26	23	-	-	-	-	-	ADDR27	nSIRIN1	TMR3_BLK
PE12	21	20	-	-	-	-	-	ADDR28	SSP1_RXD	UART1_RXD

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Вывод	Контактная площадка кристалла	Тип корпуса					Дополнительные функции вывода			
		4229.132-3	H18.64-1B	H16.48-1B	МК 5153.64-2	LQFP64	Аналог.	Основ.	Альтер.	Переопр.
PE13	20	19	-	-	-	-	-	ADDR29	SSP1_FSS	UART1_TXD
PE14	43	40	-	-	-	-	-	ADDR30	TMR2_ETR	SCL1
PE15	19	18	-	-	-	-	-	ADDR31	EXT_INT3	SDA1
<b>Порт F</b>										
PF0	3	2	58	44	58	2	-	ADDR0	SSP1_TXD	UART2_RXD
PF1	4	3	59	45	59	3	-	ADDR1	SSP1_CLK	UART2_TXD
PF2	5	4	60	46	60	4	-	ADDR2	SSP1_FSS	CAN2_RX
PF3	6	5	61	47	61	5	-	ADDR3	SSP1_RXD	CAN2_TX
PF4 MODE[0]	7	6	62	48	62	6	-	ADDR4	-	-
PF5 MODE[1]	8	7	63	1	63	7	-	ADDR5	-	-
PF6 MODE[2]	9	8	64	-	64	8	-	ADDR6	TMR1_CH1	-
PF7	10	9	-	-	-	-	-	ADDR7	TMR1_CH1N	TMR3_CH1
PF8	11	10	-	-	-	-	-	ADDR8	TMR1_CH2	TMR3_CH1N
PF9	12	11	-	-	-	-	-	ADDR9	TMR1_CH2N	TMR3_CH2
PF10	13	12	-	-	-	-	-	ADDR10	TMR1_CH3	TMR3_CH2N
PF11	14	13	-	-	-	-	-	ADDR11	TMR1_CH3N	TMR3_ETR
PF12	15	14	-	-	-	-	-	ADDR12	TMR1_CH4	SSP2_FSS
PF13	16	15	-	-	-	-	-	ADDR13	TMR1_CH4N	SSP2_CLK
PF14	17	16	-	-	-	-	-	ADDR14	TMR1_ETR	SSP2_RXD
PF15	18	17	-	-	-	-	-	ADDR15	TMR1_BLK	SSP2_TXD
<b>Системное управление</b>										
RESET	40	37	10	7	10	18	0 – 1 –			
WAKEUP	38	35	9	-	9	17	0 – 1 –	StandBy	StandBy	
STANDBY	34	31	6	-	6	14	0 – 1 –	StandBy	StandBy	StandBy
OSC_IN	41	38	11	8	11	19	HSE			
OSC_OUT	42	39	12	9	12	20	HSE			
<b>USB интерфейс</b>										
DP	22	21	1	2	1	9	USB D+			
DN	25	22	2	3	2	10	USB D-			
<b>Питание</b>										
U <sub>CC</sub>	1,2,31,32, 77,78,103, 104	1,28,29, 72,73, 98,99	4,29,40, 57	5,21,30,4 3	4, 29, 40, 57	1, 12, 38, 48	2,2...3,6			
AU <sub>CC</sub>	59,60	55	20	16	20	28	2,4...3,6	, Comparator		
AU <sub>CC1</sub>	51,52	48,49	16	13	16	24	PLL 2,2...3,6			
BU <sub>CC</sub>	33	30	5	-	5	13	1.8...3,6			
GND	29,30,79, 105,139	26,27, 74,100, 132	3,30,41, 56	4,22,31,4 2	3, 30, 41, 56	11, 39, 49, 64				
AGND	57,58	54	19	15	19	27				
AGND1	49,50	46,47	15	12	15	23				
DU <sub>CC</sub>	28,80,106, 138	25,75,10 1,131	-	-	-	-				
<b>Выводы для тестирования и исследования</b>										
BDU <sub>CC</sub>	37	34	-	-	-	-				

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Вывод	Контактная площадка кристалла	Тип корпуса					Дополнительные функции вывода			
		4229.132-3	H18.64-1B	H16.48-1B	МК 5153.64-2	LQFP64	Аналог.	Основ.	Альтер.	Переопр.
EXT_POR	39	36	-	-	-	-	«	»		
SHDN	27	24	-	-	-	-		LDO.	/	
JTAG_EN	102	97	-	-	-	-	«	»		
<b>Не используются</b>										
NC	23,24, 117,118, 126,127	116,120	32	24	32	37	«	»		

\* —

1986 94 .

**Таблица 3 – Описание выводов микроконтроллеров серии 1986BE9х в корпусах МК 6109.144-А, МК 8307.144-АН3**

Номер вывода	Обозначение вывода	Назначение и функции вывода			
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода	
<b>Порт А</b>					
B2	PA0		/	0	
			–	–	
			DATA0	0	DATA[31:0] EXT_BUS
			EXT_INT1		
A3	PA1		/	1	
			–	–	
			DATA1	1	DATA[31:0] EXT_BUS
			TMR1_CH1	/	1 1
B3	PA2		/	2	
			–	–	
			DATA2	2	DATA[31:0] EXT_BUS
			TMR1_CH1N		1 1
C3	PA3		/	3	
			–	–	
			DATA3	3	DATA[31:0] EXT_BUS
			TMR1_CH2	/	2 1
D4	PA4		/	4	
			–	–	
			DATA4	4	DATA[31:0] EXT_BUS
			TMR1_CH2N		2 1
				2 2	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
A4	PA5		/	5
			–	–
			DATA5	5 DATA[31:0] EXT_BUS
			TMR1_CH3	/ 3 1
		TMR2_CH3	/ 3 2	
B4	PA6		/	6
			–	–
			DATA6	6 DATA[31:0] EXT_BUS
			CAN1_TX	CAN1
		UART1_RXD	UART1	
C4	PA7		/	7
			–	–
			DATA7	7 DATA[31:0] EXT_BUS
			CAN1_RX	CAN1
		UART1_TXD	UART1	
C5	PA8		/	8
			–	–
			DATA8	8 DATA[31:0] EXT_BUS
			TMR1_CH3N	3 1
		TMR2_CH3N	3 2	
A5	PA9		/	9
			–	–
			DATA9	9 DATA[31:0] EXT_BUS
			TMR1_CH4	/ 4 1
		TMR2_CH4	/ 4 2	
B5	PA10		/	10
			–	–
			DATA10	10 DATA[31:0] EXT_BUS
			nUART1DTR	UART1
		TMR2_CH4N	4 2	
D5	PA11		/	11
			–	–
			DATA11	11 DATA[31:0] EXT_BUS
			nUART1RTS	UART1
		TMR2_BLK	2	
D6	PA12		/	12
			–	–
			DATA12	12 DATA[31:0] EXT_BUS
			nUART1RI	
		TMR2_ETR	2	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода			
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода	
A6	PA13		/	13	
			–	–	
			DATA13	13	DATA[31:0] EXT_BUS
			nUART1DCD TMR1_CH4N		4 1
B6	PA14		/	14	
			–	–	
			DATA14	14	DATA[31:0] EXT_BUS
			nUART1DSR TMR1_BLK		1
C6	PA15		/	15	
			–	–	
			DATA15	15	DATA[31:0] EXT_BUS
			nUART1CTS TMR1_ETR		1
<b>Порт B</b>					
D11	PB0/ JA_TDO		/	0 /	
				JTAG-	
			–	–	
			DATA16	16	DATA[31:0] EXT_BUS
		TMR3_CH1	/	1 3	
		UART1_TXD		UART1	
D12	PB1/ JA_TMS		/	1 /	
				JTAG	
			–	–	
			DATA17	17	DATA[31:0] EXT_BUS
		TMR3_CH1N		1 3	
		UART2_RXD		UART2	
D9	PB2/ JA_TCK		/	2 /	
				JTAG	
			–	–	
			DATA18	18	DATA[31:0] EXT_BUS
		TMR3_CH2	/	2 3	
		CAN1_TX		CAN1	
C10	PB3/ JA_TDI		/	3 /	
				JTAG	
			–	–	
			DATA19	19	DATA[31:0] EXT_BUS
		TMR3_CH2N		2 3	
		CAN1_RX		CAN1	
C11	PB4/ JA_TRST		/	4 /	
				JTAG	
			–	–	
			DATA20	20	DATA[31:0] EXT_BUS
		TMR3_BRK		3	
		TMR3_ETR		3	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода			
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода	
A10	PB5		/	5	
			–	–	
			DATA21	21	DATA[31:0] EXT_BUS
			UART1_TXD		UART1
B10	PB6		/	6	
			–	–	
			DATA22	22	DATA[31:0] EXT_BUS
			UART1_RXD		UART1
B9	PB7		/	7	
			–	–	
			DATA23	23	DATA[31:0] EXT_BUS
			nSIROUT1		IRDA SIR ART1
A9	PB8		/	8	
			–	–	
			DATA24	24	DATA[31:0] EXT_BUS
			COMP_OUT		
D8	PB9		/	9	
			–	–	
			DATA25	25	DATA[31:0] EXT_BUS
			nSIRIN1		IRDA SIR UART1
C8	PB10		/	10	
			–	–	
			DATA26	26	DATA[31:0] EXT_BUS
			EXT_INT2		
B8	PB11		/	11	
			–	–	
			DATA27	27	DATA[31:0] EXT_BUS
			EXT_INT1		
A8	PB12		/	12	
			–	–	
			DATA28	28	DATA[31:0] EXT_BUS
			SSP1_FSS	/	SPI1
		/	SPI2		



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода			
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода	
B7	PB13		/	13	
			–	–	
			DATA29	29	DATA[31:0] EXT_BUS
			SSP1_CLK	/	SPI1
		SSP2_CLK	/	SPI2	
A7	PB14		/	14	
			–	–	
			DATA30	30	DATA[31:0] EXT_BUS
			SSP1_RXD		SPI1
		SSP2_RXD		SPI2	
C7	PB15		/	15	
			–	–	
			DATA31	31	DATA[31:0] EXT_BUS
			SSP1_TXD		SPI1
		SSP2_TXD		SPI2	
<b>Попр С</b>					
D10	PC0		/	0	
			–	–	
			–	–	
			SCL1		I2C
		SSP2_FSS	/	SPI 2	
E10	PC1		/	1	
			–	–	
			OE		EXT_BUS
			SDA1	/	I2C
		SSP2_CLK	/	SPI2	
E12	PC2		/	2	
			–	–	
			WE		EXT_BUS
			TMR3_CH1	/	1 3
		SSP2_RXD		SPI2	
E11	PC3		/	3	
			–	–	
			BE0		0 32
					EXT_BUS
		TMR3_CH1N		1 3	
		SSP2_TXD		SPI2	
E9	PC4		/	4	
			–	–	
			BE1		1 32
					EXT_BUS
		TMR3_CH2	/	2 3	
		TMR1_CH1	/	1 1	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
F9	PC5		/	5
			–	
		BE2		2 32 EXT_BUS
		TMR3_CH2N		2 3
		TMR1_CH1N		1 1
F12	PC6		/	6
			–	
		BE3		3 32 EXT_BUS
		TMR3_CH3	/	3 3
		TMR1_CH2	/	2 1
F11	PC7		/	7
			–	
		CLOCK		EXT_BUS
		TMR3_CH3N		3 3
		TMR1_CH2N		2 1
F10	PC8		/	8
			–	
		CAN1_TX		CAN1
		TMR3_CH4	/	4 3
		TMR1_CH3	/	3 1
G10	PC9		/	9
			–	
		CAN1_RX		CAN1
		TMR3_CH4N		4 3
		TMR1_CH3N		3 1
G12	PC10		/	10
			–	
			–	
		TMR3_ETR		3
		TMR1_CH4	/	4 1
G11	PC11		/	11
			–	
			–	
		TMR3_BLK		3
		TMR1_CH4N		4 1
G9	PC12		/	12
			–	
			–	
		EXT_INT2		
		TMR1_ETR		1

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
H12	PC13		/	13
			–	–
			–	–
			EXT_INT4 TMR1_BLK	1
H11	PC14		/	14
			–	–
			–	–
			SSP2_FSS CAN2_RX	/ SPI2 CAN2
H10	PC15		/	15
			–	–
			–	–
			SSP2_RXD CAN2_TX	SPI2 CAN2
<b>Попр D</b>				
L11	PD0/ JB_TMS		/	0 D/ JTAG
			ADC0_REF+	0 /
			TMR1_CH1N	1 1
			UART2_RXD	UART2
			TMR3_CH1	/ 1 3
M11	PD1/ JB_TCK		/	1 D/ JTAG
			ADC1_REF-	1 /
			TMR1_CH1	/ 1 1
			UART2_TXD	UART2
			TMR3_CH1N	1 3
L12	PD2/ JB_TRST		/	2 D/ JTAG
			ADC2	2
			BUSY1	EXT_BUS
			SSP2_RXD	SPI2
			TMR3_CH2	/ 2 3
M12	PD3/ JB_TDI		/	3 D/ JTAG
			ADC3	3
			–	–
			SSP2_FSS	/ SPI2
			TMR3_CH2N	3 2
M10	PD4/ JB_TDO		/	4 D/ JTAG-
			ADC4	4
			TMR1_ETR	1
			nSIROUT2	IRDA SIR UART2
			TMR3_BLK	3

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
K12	PD5		/	5 D
			ADC5	5
			CLE	EXT_BUS
			SSP2_CLK	/ SPI2
			TMR2_ETR	2
K11	PD6		/	6 D
			ADC6	6
			ALE	EXT_BUS
			SSP2_TXD	SPI2
			TMR2_BLK	2
L10	PD7		/	7 D
			ADC7	7
			TMR1_BLK	1
			nSIRIN2	IRDA SIR UART2
			UART1_RXD	UART1
K10	PD8		/	8 D
			ADC8	8
			TMR1_CH4N	4 1
			TMR2_CH1	/ 1 2
			UART1_TXD	UART1
J10	PD9		/	9 D
			ADC9	9
			CAN2_TX	CAN2
			TMR2_CH1N	1 2
			SSP1_FSS	/ SPI1
J9	PD10		/	10 D
			ADC10	10
			TMR1_CH2	/ 2 1
			TMR2_CH2	/ 2 2
			SSP1_CLK	/ SPI1
M9	PD11		/	11 D
			ADC11	11
			TMR1_CH2N	2 1
			TMR2_CH2N	2 2
			SSP1_RXD	SPI1
L9	PD12		/	12 D
			ADC12	12
			TMR1_CH3	/ 3 1
			TMR2_CH3	/ 3 2
			SSP1_TXD	SPI1

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
K9	PD13		/	13 D
			ADC13	13
			TMR1_CH3N	3 1
			TMR2_CH3N	3 2
		CAN1_TX	CAN1	
K8	PD14		/	14 D
			ADC14	14
			TMR1_CH4	/ 4 1
			TMR2_CH4	/ 4 2
		CAN1_RX	CAN1	
M8	PD15		/	15 D
			ADC15	15
			CAN2_RX	CAN2
			BUSY2	EXT_BUS
		EXT_INT3		
<b>Порт E</b>				
L7	PE0		/	0 E
			DAC2_OUT	2
			ADDR16	16
			TMR2_CH1	/ 1 2
		CAN1_RX	CAN1	
K7	PE1		/	1 E
			DAC2_REF	2
			ADDR17	17 ADDR [31:0] EXT_BUS
			TMR2_CH1N	1 2
		CAN1_TX	CAN1	
L5	PE2		/	2 E
			COMP_IN1	1
			ADDR18	18 ADDR [31:0] EXT_BUS
			TMR2_CH3	/ 3 2
		TMR3_CH1	/ 1 3	
K5	PE3		/	3 E
			COMP_IN2	2
			ADDR19	19
			TMR2_CH3N	3 2
		TMR3_CH1N	1 3	
M4	PE4		/	4 E
			COMP_REF+	
			ADDR20	20 ADDR [31:0] EXT_BUS
			TMR2_CH4N	4 2
		TMR3_CH2	/ 2 3	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
L4	PE5		/	5 E
			COMP_REF-ADDR21	21 ADDR [31:0] EXT_BUS
			TMR2_BLK	2
			TMR3_CH2N	2 3
L1	PE6		/	6 E
			OSC_IN32	32
			ADDR22	22 ADDR [31:0] EXT_BUS
			CAN2_RX	CAN2
	TMR3_CH3	/ 3 3		
L2	PE7		/	7 E
			OSC_OUT32	32
			ADDR23	23 ADDR [31:0] EXT_BUS
			CAN2_TX	CAN2
	TMR3_CH3N	3 3		
J5	PE8		/	8 E
			COMP_IN3	3
			ADDR24	24 ADDR [31:0] EXT_BUS
			TMR2_CH4	/ 4 2
	TMR3_CH4	/ 4 3		
K6	PE9		/	9 E
			DAC1_OUT	1
			ADDR25	25 ADDR [31:0] EXT_BUS
			TMR2_CH2	/ 2 2
	TMR3_CH4N	3 4		
M6	PE10		/	10 E
			DAC1_REF	1
			ADDR26	26 ADDR [31:0] EXT_BUS
			TMR2_CH2N	2 2
	TMR3_ETR	3		
H3	PE11		/	11 E
			-	-
			ADDR27	27 ADDR [31:0] EXT_BUS
			nSIRIN1	IRDA SIR UART1
	TMR3_BRK	3		
H4	PE12		/	12 E
			-	-
			ADDR28	28 ADDR [31:0] EXT_BUS
			SSP1_RXD	SPI1
	UART1_RXD	UART1		

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
G4	PE13		/	13 E
			–	–
			ADDR29	29 ADDR [31:0] EXT_BUS
			SSP1_FSS	/ SPI1
		UART1_TXD	UART1	
K4	PE14		/	14 E
			–	–
			ADDR30	30 ADDR [31:0] EXT_BUS
			TMR2_ETR	2
		SCL1	I2C	
G1	PE15		/	15 E
			–	–
			ADDR31	31
			EXT_INT3	
		SDA1	/ I2C	
<b>Порт F</b>				
C1	PF0		/	0 F
			–	–
			ADDR0	0 ADDR [31:0] EXT_BUS
			SSP1_TXD	SPI1
		UART2_RXD	UART2	
C2	PF1		/	1 F
			–	–
			ADDR1	1 ADDR [31:0] EXT_BUS
			SSP1_CLK	/ SPI1
		UART2_TXD	UART2	
D3	PF2		/	2 F
			–	–
			ADDR2	2 ADDR [31:0] EXT_BUS
			SSP1_FSS	/ SPI1
		CAN2_RX	CAN2	
D2	PF3		/	3 F
			–	–
			ADDR3	3 ADDR [31:0] EXT_BUS
			SSP1_RXD	SPI1
		CAN2_TX	CAN2	
D1	PF4/ MODE[0]		/	4 F/
			–	–
			ADDR4	4 ADDR [31:0] EXT_BUS
			–	–
		–	–	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
E4	PF5/ MODE[1]		/	5 F/
		–	–	
		ADDR5		5 ADDR [31:0] EXT_BUS
		–	–	
E3	PF6/ MODE[2]		/	6 F/
		–	–	
		ADDR6		6 ADDR [31:0] EXT_BUS
		TMR1_CH1	/	1 1
E2	PF7		/	7 F
		–	–	
		ADDR7		7 ADDR [31:0] EXT_BUS
		TMR1_CH1N		1 1
E1	PF8		/	8 F
		–	–	
		ADDR8		8 ADDR [31:0] EXT_BUS
		TMR1_CH2	/	2 1
E5	PF9		/	9 F
		–	–	
		ADDR9		9 ADDR [31:0] EXT_BUS
		TMR1_CH2N		2 1
F4	PF10		/	10 F
		–	–	
		ADDR10		10 ADDR [31:0] EXT_BUS
		TMR1_CH3	/	3 1
F2	PF11		/	11 F
		–	–	
		ADDR11		11 ADDR [31:0] EXT_BUS
		TMR1_CH3N		3 1
F1	PF12		/	12 F
		–	–	
		ADDR12		12 ADDR [31:0] EXT_BUS
		TMR1_CH4	/	4 1
		SSP2_FSS	/	SPI2



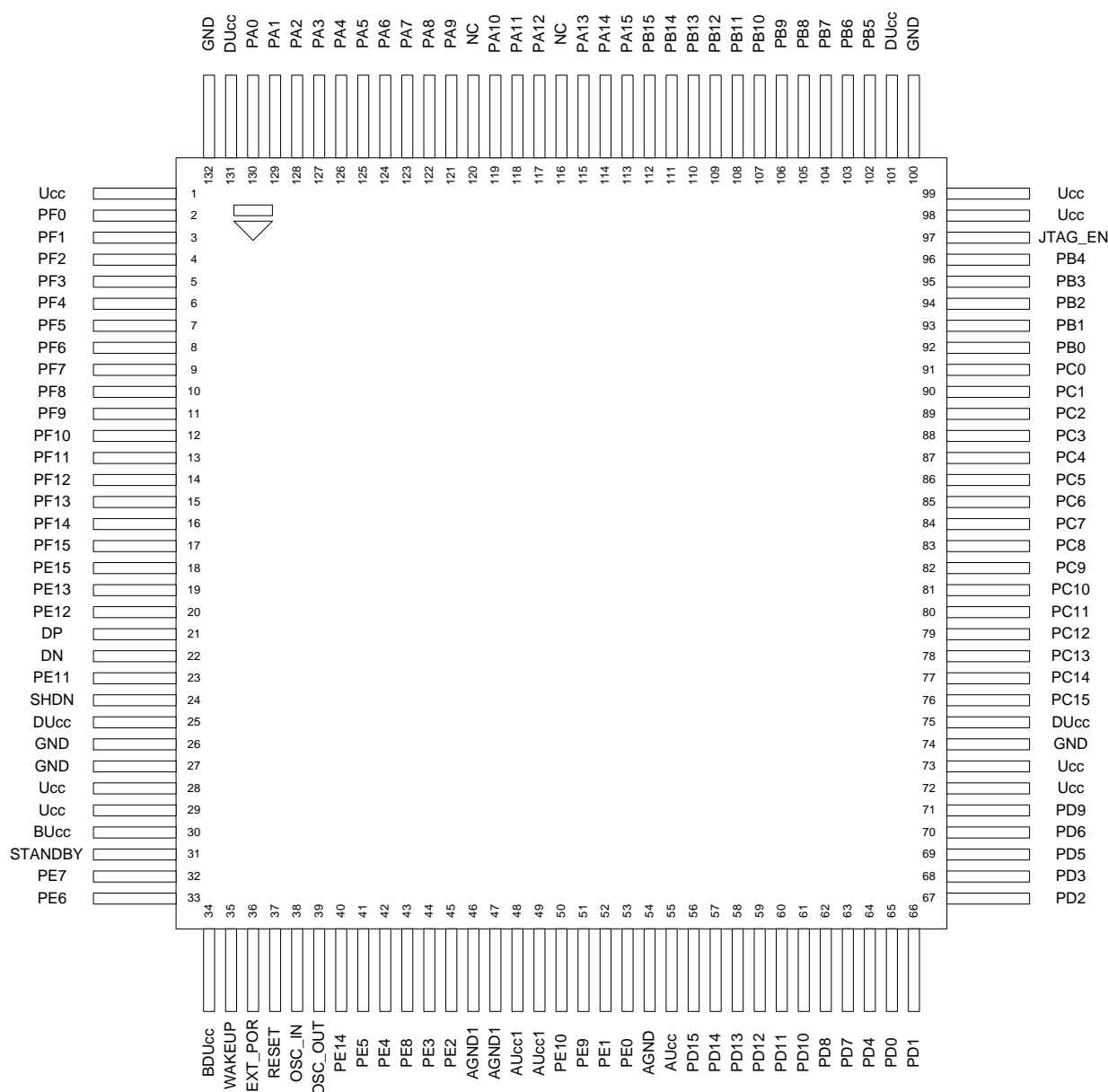
**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
F3	PF13		/	13 F
			–	–
			ADDR13	13 ADDR [31:0] EXT_BUS
			TMR1_CH4N	4 1
		SSP2_CLK	/ SPI2	
G3	PF14		/	14 F
			–	–
			ADDR14	14 ADDR [31:0] EXT_BUS
			TMR1_ETR	1
		SSP2_RXD	SPI2	
G2	PF15		/	15 F
			–	–
			ADDR15	15 ADDR [31:0] EXT_BUS
			TMR1_BLK	1
		SSP2_TXD	SPI2	
<b>Системное управление</b>				
M1	RESET			
M2	WAKEUP		Standby	
K1	STANDBY	Standby		
M3	OSC_IN	HSE		
L3	OSC_OUT	HSE		
<b>USB интерфейс</b>				
H2	DP	USB D+		
H1	DN	USB D-		
<b>Питание</b>				
A1, B1, B11, B12, J4, J11, J12, K3	Ucc	(2,2 – 3,6)		
J8, L8	AUcc	( U )		
J6, L6	AUcc1	( U )		
K2	BUcc	(1,8 – 3,6)		
J1	GND			
J7, M7	AGND			
H5, M5	AGND1			
E6, E7, F5-F8, G5-G8, H6, H7,	GND_EXP			
<b>Выводы для тестирования</b>				
A2, A11, H9, J2	DUcc		1,8 .	
J3	SHDN	/	LDO.	« »

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

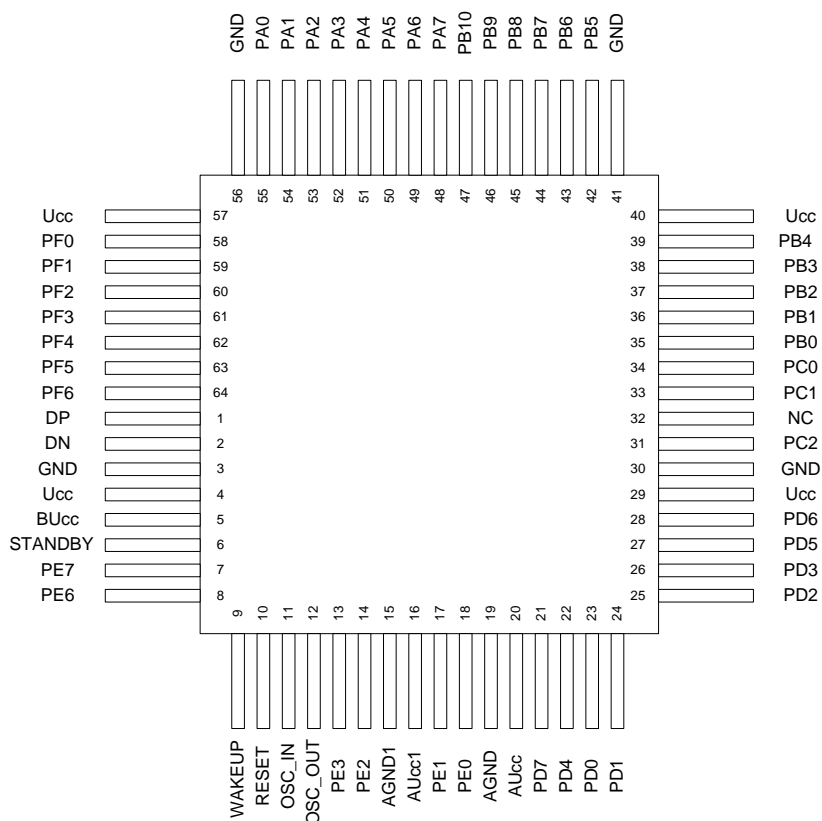
Номер вывода	Обозначение вывода	Назначение и функции вывода		
		Тип функции вывода	Обозначение функции вывода	Функциональное назначение вывода
C12	JTAG_EN	/	TAP-	« »
Не используются				
A12, C9, D7, E8, H8	NC	.		« »

### 4.1 Диаграммы расположения выводов в корпусах с планарным расположением выводов

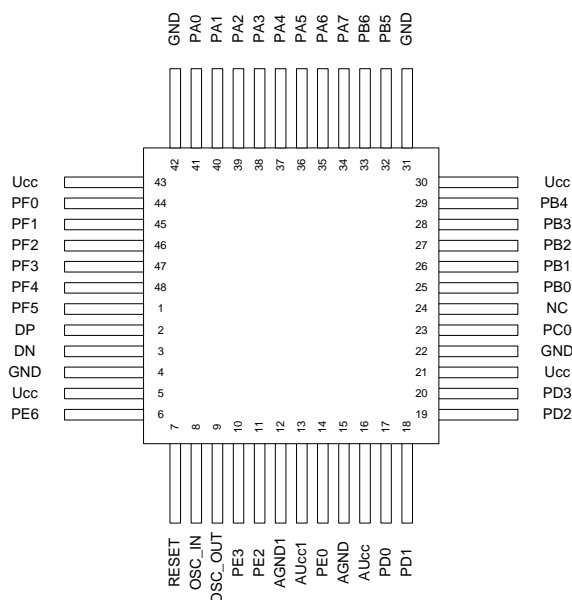


**Рисунок 2. Расположение выводов в 132-х выводном корпусе 4229.132-3**

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

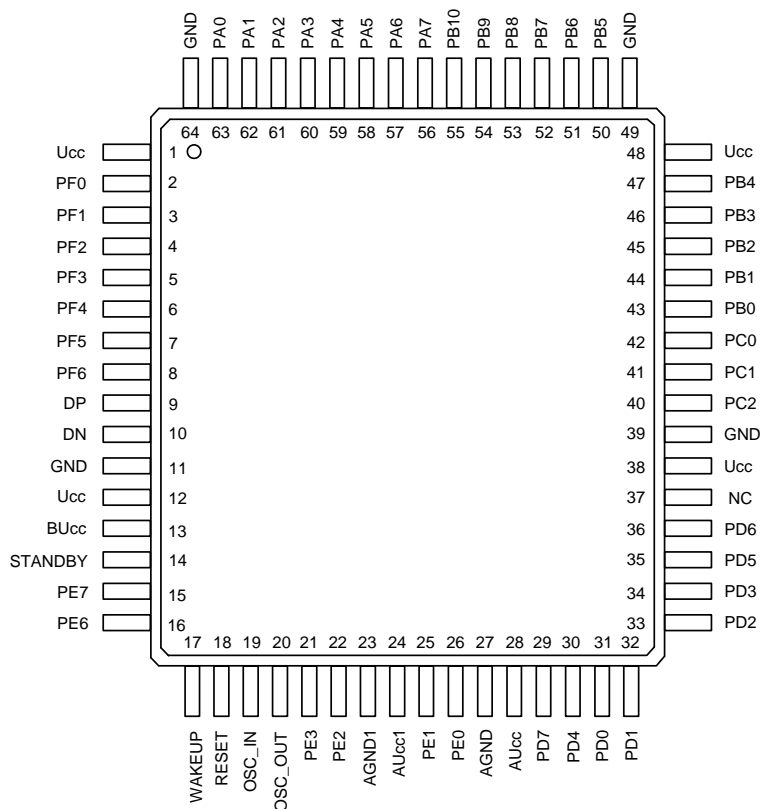


**Рисунок 3. Расположение выводов в 64-х выводном корпусе Н18.64-1В**



**Рисунок 4. Расположение выводов в 48-ми выводном корпусе Н16.48-1В**

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**



**Рисунок 5. Расположение выводов в 64-х выводном пластиковом корпусе LQFP64**

## 5 Указания по применению и эксплуатации

0,1 . « » « »

MODE

( 10 ) «0» ,

:

- 6, 7, 8 1986 91 , 1986 94 ;
- 62, 63, 64 1986 92 , 1986 92 ;
- 48,1 1986 93 ;
- D1, E3, E4 1986 94 , 1986 94 .

1986 93 :

- 6- MDR\_PORTF->OE, MDR\_PORTF->ANALOG, MDR\_PORTF->GFEN MDR\_PORTF->RXTX;
- 12- / 13- MDR\_PORTF->FUNC MDR\_PORTF->PWR;
- 6- / 22- MDR\_PORTF->PULL MDR\_PORTF->PD.

В D,

1986 94

4° / .

4.

**Таблица 4 – Параметры профиля пайки оловянно-свинцовой паяльной пастой**

Параметр	Оловянно-свинцовый припой
, °	183
, °	210–220
), °	205
, ° /	1-4
, ° /	2-4
, °	100-180
,	60-120
,	60-90
,	20

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

1986 94

: Sn63 / Pb37.

4.

:

– 116 1986 91 , 1986 94 ;

– 32 1986 92 ;

– 24 1986 93 ;

– E6, E7, F5, F6, F7, G5, G6, G7, G8, H6, H7 1986 94 ,  
1986 94 .

- , « »,

1986 92 1

« ».

- , « »,

:

– 36, 24, 97 1986 91 , 1986 94 ;

– J3, C12 1986 94 , 1986 94 .

PA – PF, nRESET, WAKEUP

(

)

(1 – 100) ,

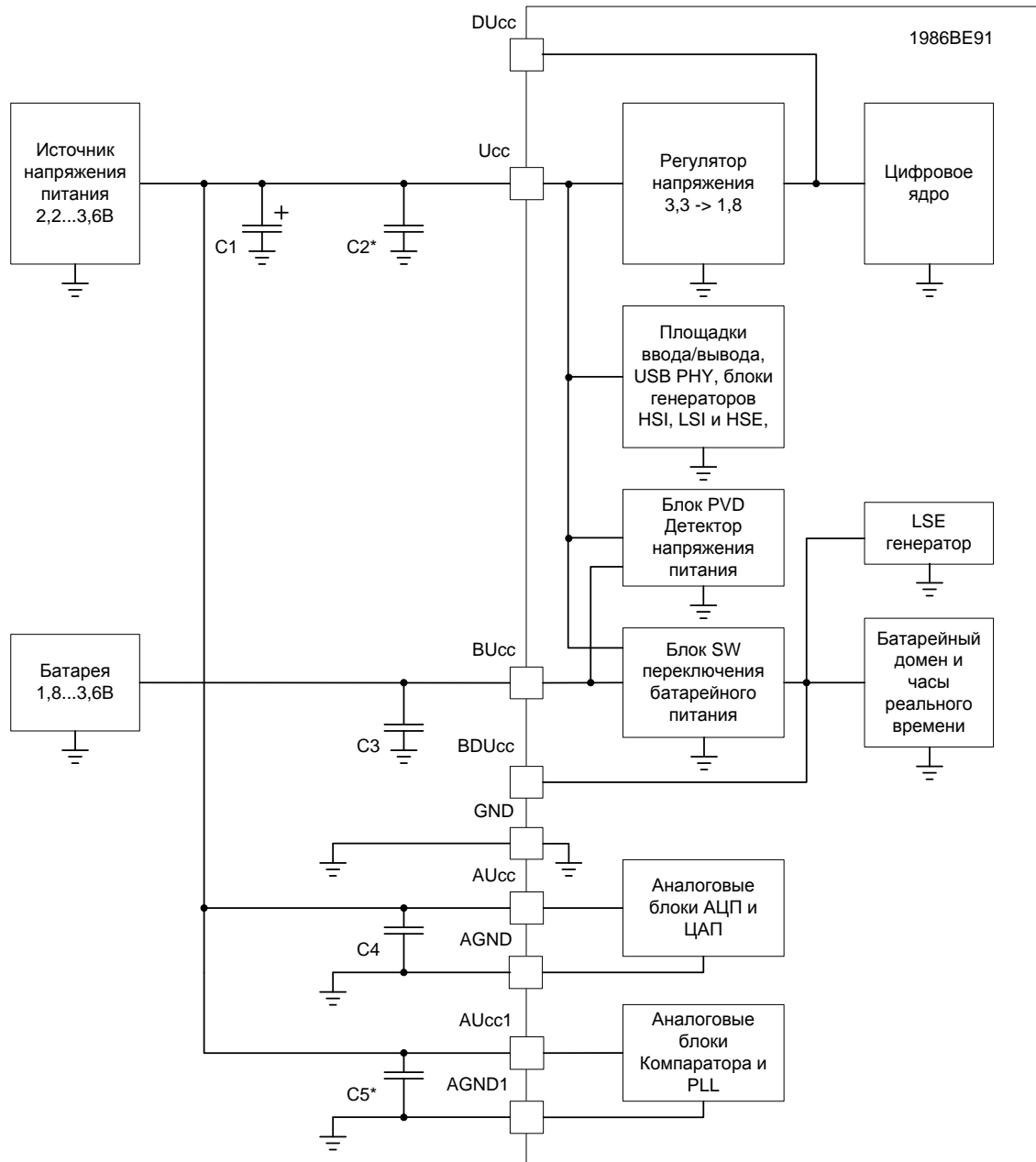
:

– 25, 34, 75, 101, 131 1986 91 , 1986 94 ;

– A2, A11, H9, J2 1986 94 , 1986 94 .

« » , « ») – (





**Рисунок 6. Структурная блок-схема подачи питания**

Примечания:

1. ;
2.  $1 = 22$  ,  $2 = 3 = 4 = 5 = 0,1$  ;
3. ,  $BU_{cc}$
4.  $U_{cc}$ ;  
USB,  $U_{cc}$
5.  $3,0$   $3,6$  ;  
 $U_{cc}$  ( $AU_{cc}$   $AU_{cc1}$ )  
 $2,4$   $3,6$  .

« » « » . « » . « » .



### 6.1 Схема сброса при включении и выключении основного питания

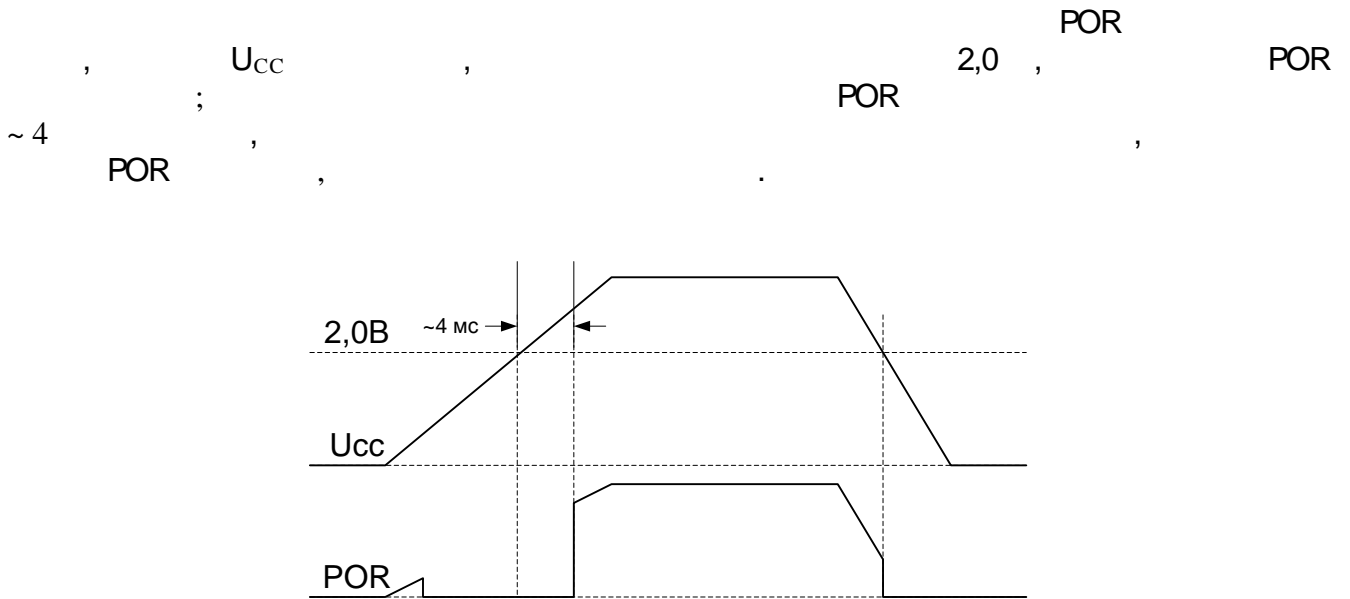


Рисунок 7. Сигнал сброса при включении и выключении основного напряжения питания

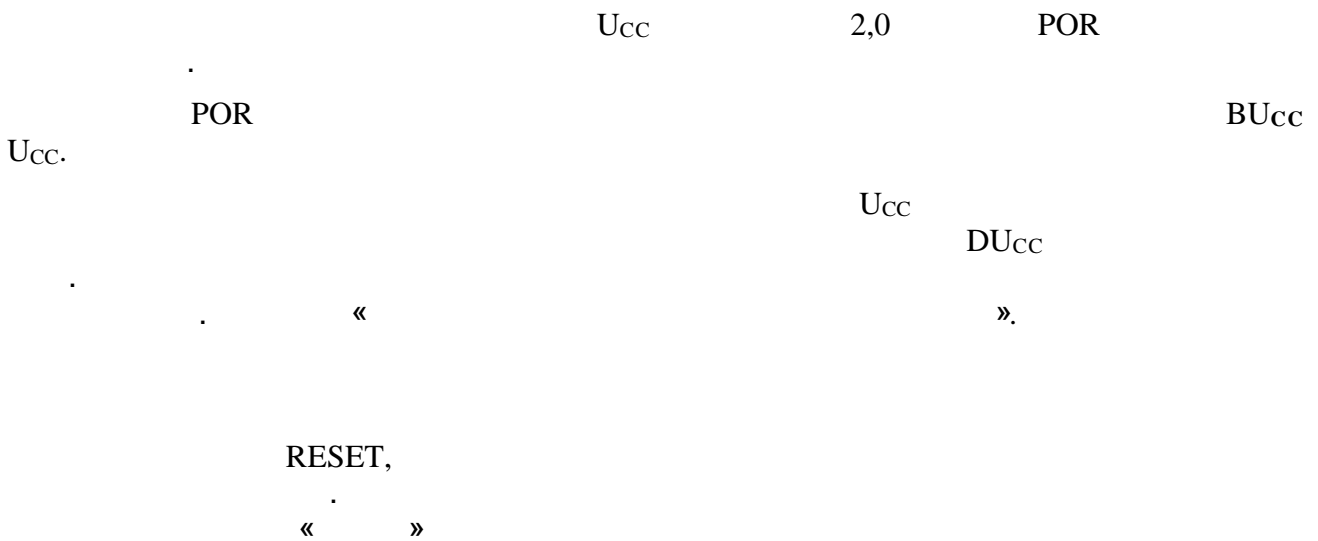
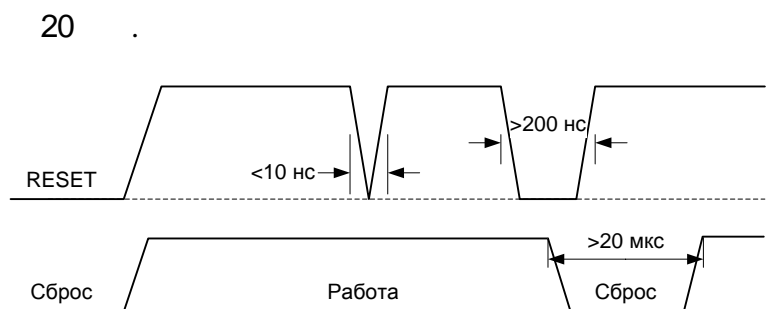


Рисунок 8. Структурная блок-схема сброса

200 ,



**Рисунок 9. Формирование сигнала сброса**

## 7 Организация памяти

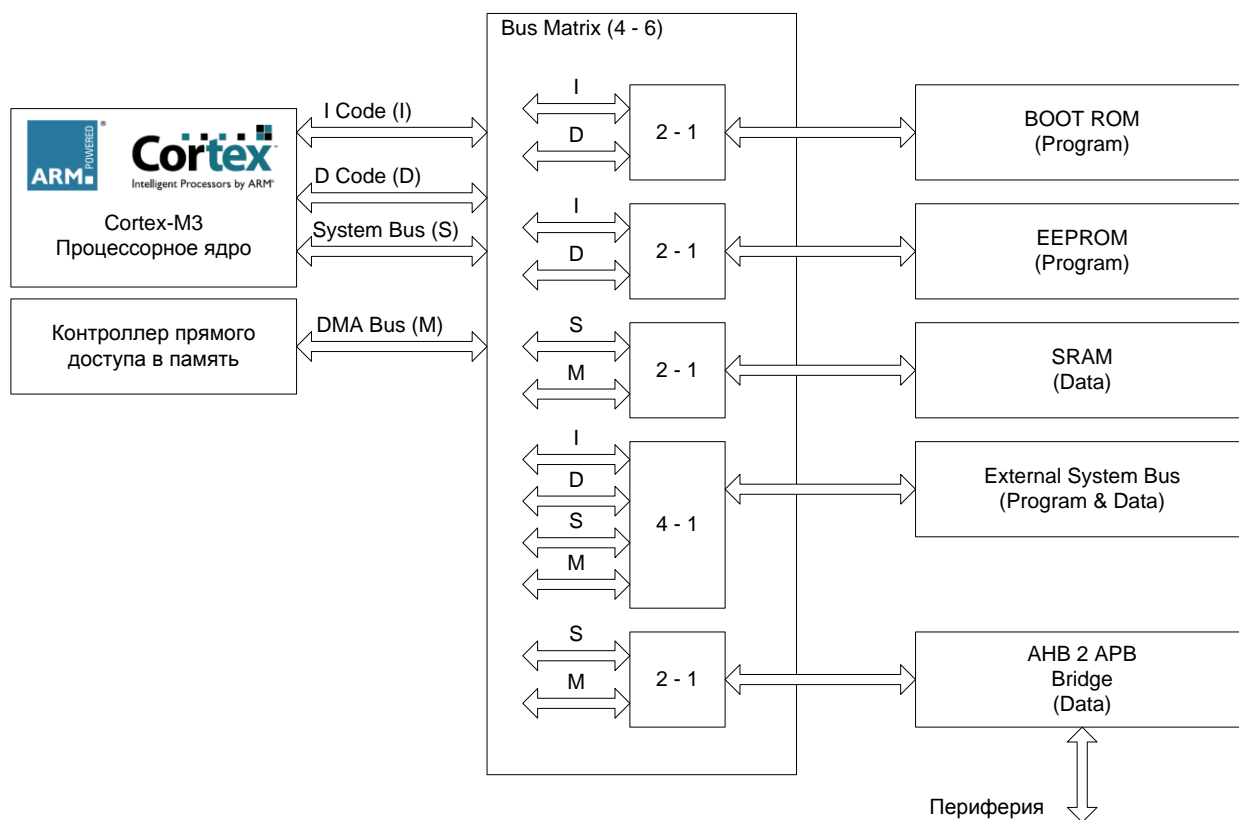
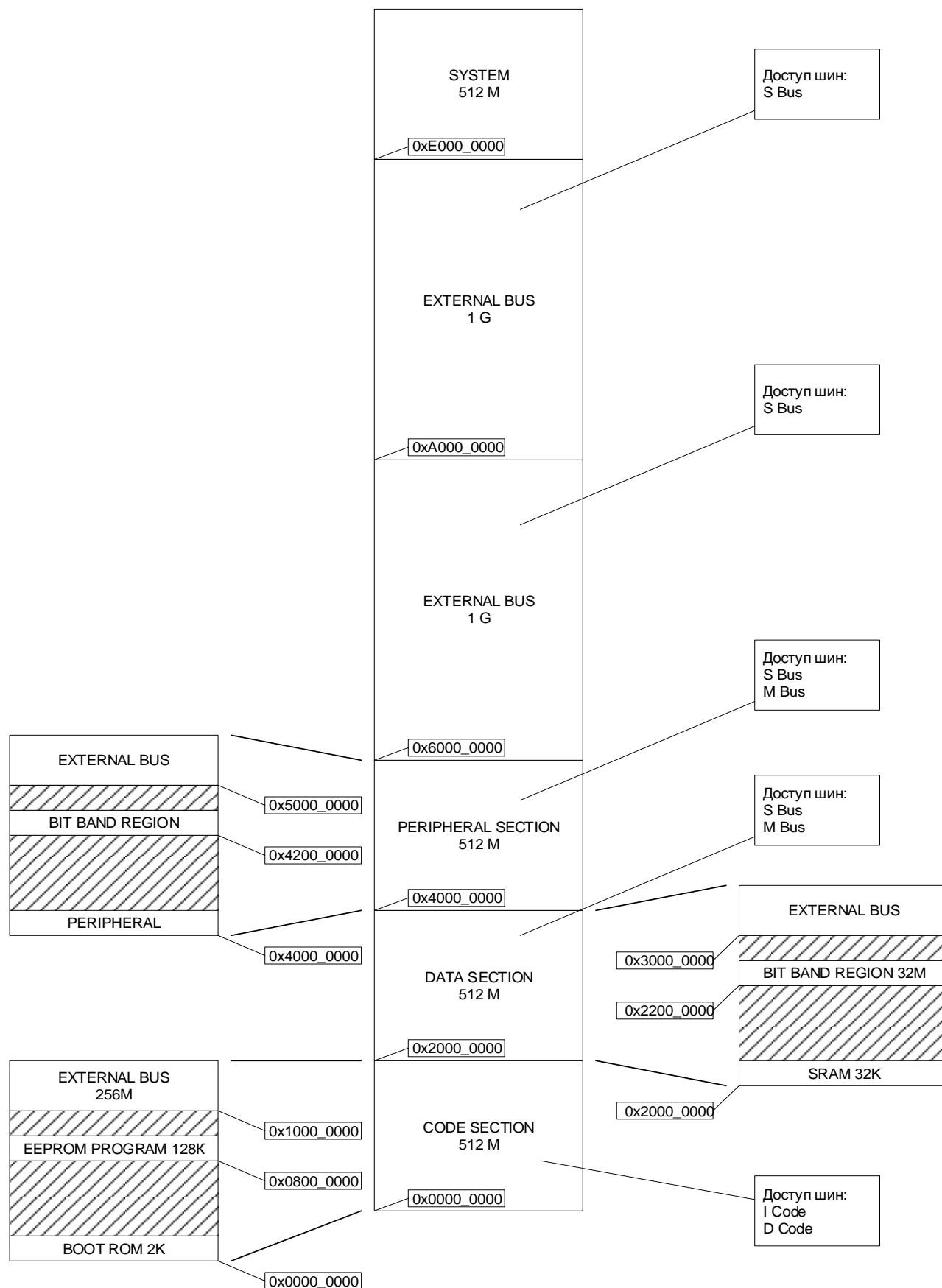


Рисунок 10. Структурная схема организации памяти

- I Code – ;
  - D Code – , ;
  - S Bus – , .
- (DMA),
- DMA Bus.



**Рисунок 11. Карта распределения основных областей памяти**

## 7.1 Секция CODE

**Область BOOT ROM:**

;

**Область EEPROM PROGRAM:**

,

**Область EXTERNAL BUS:**

,

## 7.2 Секция DATA

**Область Internal SRAM (Data):**

(stack) « » (heap)

« »

**Область BIT BAND REGION TO SRAM (Data):**

Internal SRAM. BIT BAND REGION  
 « » « » « - -  
 ».  
 alias-  
 bit-band [0]  
 alias-  
 bit-band [0] 0 1 bit-  
 bit-band [31:1] alias- 0xFF. 0x00  
 bit-band 0x01 0xFE. alias-  
 0x01 0x00. [31:1] 0x01  
 bit-band 0x00 bit-  
 band

**Область EXTERNAL BUS:**

,

## 7.3 Секция PERIPHERAL

**Область PERIPHERAL (Data):**

**Область BIT BAND REGION TO PERIPHERAL (Data):**

PERIPHERAL. BIT BAND REGION  
« » « » « - -  
».

**Область EXTERNAL BUS:**

## 7.4 Секция EXTERNAL RAM

**Область EXTERNAL BUS:**

## 7.5 Секция SYSTEM

## 7.6 Блок BUS MATRIX

BUS MATRIX I Code, D Code,  
System Bus DMA Bus  
D Code, I Code System Bus,  
DMA Bus.

## 7.7 Память BOOT ROM

BOOT ROM MASK ROM, с  
BOOT ROM – 1

## 7.8 Память EEPROM

EEPROM EEPROM – 40 5  
100  
1  
EEPROM « Flash- ».

## 7.9 Память SRAM

SRAM  
SRAM – 1

## 7.10 Регионы памяти, типы и атрибуты

(MPU)

« ».

:

- Normal
- Device
- Strongly-ordered (« »)

### **Normal**

### **Device**

Device Strongly-ordered.

### **Strongly-ordered**

Device Strongly-Ordered Device,

Strongly-ordered

:

- Shareable (« »)
- Execute Never XN (« »)

### **Shareable**

Shareable

DMA.

Strongly-ordered

Shareable.

Shareable,

### **Execute Never (XN)**

XN

( ).  
“Memory Management Fault”.

## 7.11 Последовательность обращений к памяти

instruction), « (memory barrier».

Device Strongly-ordered. 1 2, 1  
2  
5.

Таблица 5 – Последовательность обращений инструкций к памяти

1	2	Normal	Device		Strongly-ordered
			“non-shareable”	shareable	
Normal		-	-	-	-
Device, “non-shareable”		-	<	-	<
Device, shareable		-	-	<	<
Strongly-ordered		-	<	<	<

« - »  
« < » 1  
2.

Normal, Device, Strongly-ordered Shareable « Shareable.  
»; “non-shareable”

## 7.12 Поведение обращений к памяти

6.

Таблица 6 – Поведение обращений к памяти

Адресный диапазон	Секция памяти	Тип памяти	XN	Описание
0x00000000–0x1FFFFFFF	Code	Normal	-	
0x20000000–0x3FFFFFFF	SRAM	Normal	-	bit-band
0x40000000–0x5FFFFFFF	Peripheral	Device	XN	bit-band
0x60000000–0x9FFFFFFF	External RAM	Normal	-	
0xA0000000–0xDFFFFFFF	External Device	Device	XN	
0xE0000000–0xE00FFFFF	Private Peripheral	Strongly-ordered	XN	NVIC,



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

	Bus			
0xE0100000– 0xFFFFFFFF		Device	XN	

Code, SRAM External RAM Code,

MPU

« ».

### 7.12.1 Дополнительные условия доступа к совместно используемой памяти

7.

**Таблица 7 – Дополнительные условия совместного использования памяти**

Адресный диапазон	Секция памяти	Тип памяти	Возможность совместного использования	
0x00000000– 0x1FFFFFFF	Code	Normal	-	
0x20000000– 0x3FFFFFFF	SRAM	Normal	-	
0x40000000– 0x5FFFFFFF	Peripheral	Device	-	
0x60000000– 0x7FFFFFFF	External RAM	Normal	-	WBWA
0x80000000– 0x9FFFFFFF				WT
0xA0000000– 0xBFFFFFFF	External device	Device	Shareable "non-shareable"	
0xC0000000– 0xDFFFFFFF				
0xE0000000– 0xE00FFFFF	Private peripheral bus	Strongly-ordered	Shareable	
0xE0100000– 0xFFFFFFFF	Vendor-specific device	Device	-	

Normal, Device, Strongly-ordered Shareable « Shareable,  
»; "non-shareable" Shareable.

### 7.13 Программное упорядочение обращений к памяти

( )

;

, :



- DSB DSB
- DMB
- Strongly-ordered, DMB

## 7.14 Bit-band регионы

bit-band bit-band bit-band alias  
 . Bit-band 1 SRAM  
 (Peripheral). 32 bit-band alias,  
 8.

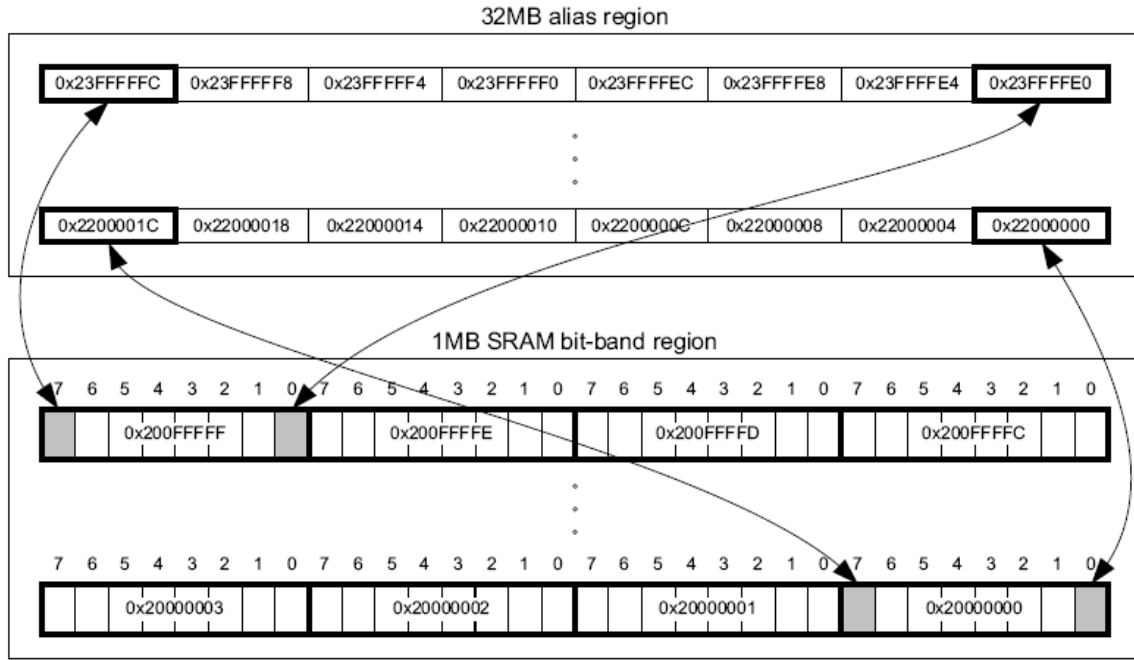
**Таблица 8 – Описание bit-band регионов**

Адресный диапазон	Регион памяти	Доступ к инструкциям и данным
0x2000_0000– 0x200F_FFFF	SRAM bit-band	
0x2200_0000– 0x23FF_FFFF	SRAM bit-band alias	SRAM bit-band SRAM bit-band alias
0x4000_0000– 0x400F_FFFF	Peripheral bit-band	
0x4200_0000– 0x43FF_FFFF	Peripheral bit-band alias	Peripheral bit-band Peripheral bit-band alias

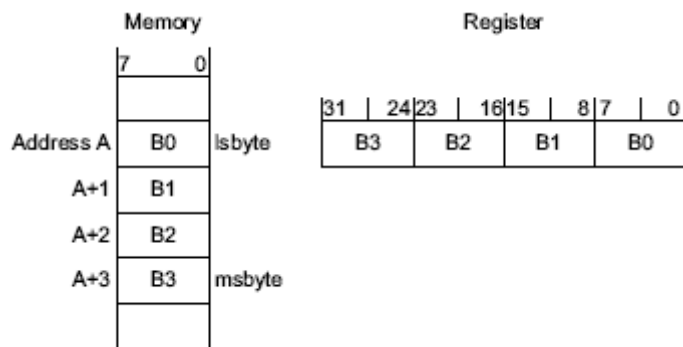
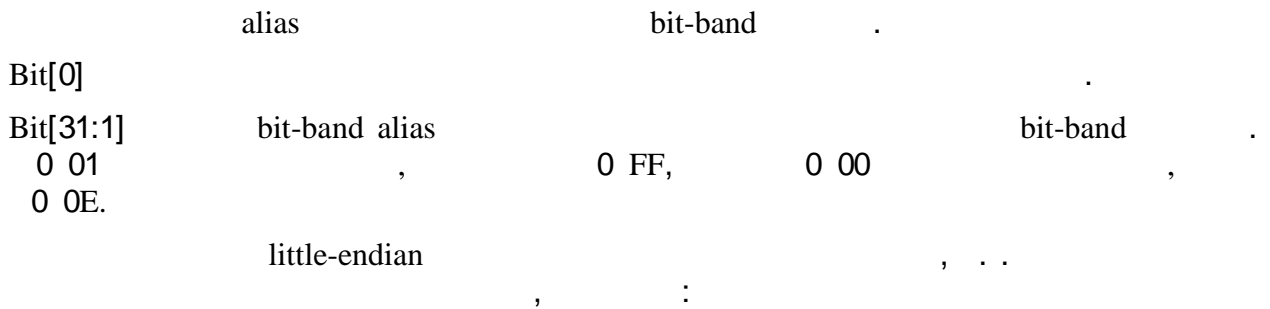
bit-band alias bit-band region:  
 $bit\_word\_offset = (byte\_offset * 32) + (bit\_number * 4)$   
 $bit\_word\_addr = bit\_band\_base + bit\_word\_offset$

```

:
bit_word_offset - bit-band ;
bit_word_addr - bit-band alias ,
bit-band ;
bit_band_base - bit-band alias ;
byte_offset - bit-band ;
bit_number - .
    
```



**Рисунок 12. Схема отображения региона bit-band alias в регионе bit-band region**



**Рисунок 13. Организация расположения байтов в 32-х битной памяти**

## 7.15 Примитивы синхронизации

Cortex-M3 примитивов синхронизации. write read-modify-

- Load-Exclusive;
- Store-Exclusive.

### 7.15.1 Команда Load-Exclusive

### 7.15.2 Команда Store-Exclusive

- 0 –
- 1 –

- LDREX    STREX –
- LDREXH   STREXH –
- LDREXB   STREXB –

Load Exclusive

Store-Exclusive.

- Load-Exclusive
- Store-Exclusive

- 0, « - »  
 - 1, , « -  
 - ».

- Load-Exclusive
- Store-Exclusive
- Store-Exclusive, Exclusive

Cortex-M3  
 Load-Exclusive.

- CLREX;

- Store-Exclusive, ;
- ;
- CLREX ;
- Store-Exclusive

LDREX, STREX CLREX.

## 7.16 Указания по программированию примитивов синхронизации

ANSI C

**Таблица 9 – Встроенные функции для создания инструкций эксклюзивного доступа**

Инструкции	Функции
LDREX, LDREXH, LDREXB	unsigned int __ldrex(volatile void *ptr)
STREX, STREXH, STREXB	int __strex(unsigned int val, volatile void *ptr)
CLREX	void clrex(void)

LDREXB:

```
__ldrex((volatile char *) 0xFF);
```

## 7.17 Базовые адреса процессора

Таблица 10 – Базовые адреса процессора

Адрес	Размер	Блок		Примечание
<b>Память программ</b>				
0x0000_0000	1	BOOT ROM		
0x0800_0000	128	EEPROM		Flash-
0x1000_0000	256	EXTERNAL BUS		
<b>Память данных</b>				
0x2000_0000	32	SYSTEM RAM		
0x2200_0000	16	SYSTEM RAM Bit Band Region		
0x3000_0000	256	EXTERNAL BUS		
<b>Периферия</b>				
0x4000_0000	1536	0	CAN1	CAN1
0x4000_8000	1536	1	CAN2	CAN2
0x4001_0000	904	2	USB	USB
0x4001_8000	20	3	EEPROM_CNTRL	Flash-
0x4002_0000	48	4	RST_CLK	
0x4002_8000	80	5	DMA	
0x4003_0000	72	6	UART1	UART1
0x4003_8000	72	7	UART2	UART2
0x4004_0000	36	8	SPI1	SSP1
0x4004_8000	-	9	-	
0x4005_0000	28	10	I2C1	I2C1
0x4005_8000	4	11	POWER	
0x4006_0000	12	12	WWDT	WWDT
0x4006_8000	16	13	IWDT	IWDT
0x4007_0000	128	14	TIMER1	1
0x4007_8000	128	15	TIMER2	2
0x4008_0000	128	16	TIMER3	3
0x4008_8000	48	17	ADC	
0x4009_0000	12	18	DAC	
0x4009_8000	12	19	COMP	
0x400A_0000	36	20	SPI2	SSP2
0x400A_8000	32	21	PORTA	
0x400B_0000	32	22	PORTB	B
0x400B_8000	32	23	PORTC	C
0x400C_0000	32	24	PORTD	D
0x400C_8000	32	25	PORTE	E
0x400D_0000	-	26	-	
0x400D_8000	84	27	BKP	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

0x400E_0000	-	28	-	
0x400E_8000	32	29	PORTF	F
0x400F_0000	88	30	EXT_BUS_CNTRL	
0x400F_8000	-	31	-	
0x4200_0000	16	PERIPHERAL Bit Band Region		
0x5000_0000	256	EXTERNAL BUS		
<b>Внешняя системная шина</b>				
0x6000_0000	1	EXTERNAL BUS		
0xA000_0000	1	EXTERNAL BUS		
<b>SYSTEM REGION</b>				
0xE000_0000	256			ARM Cortex-M3



## 8 Загрузочное ПЗУ и режимы работы микроконтроллера

BOOT ROM. (POR) (RESET)

MODE[2:0] (PF[6:4]),

« » (

« » ~50 ). FPOR Ucc.

BKP\_REG\_0E, MODE[2:0] FPOR.

PF[6:4]

**Таблица 11 – Режимы первоначального запуска микроконтроллера**

MODE[2:0]	Режим	Стартовый адрес / таблица векторов прерываний	Описание
000		0x0800_0000	Flash- JTAG_B
001		0x0800_0000	Flash- JTAG_
010		0x1000_0000	ROM Wait_States = 0xF JTAG_B
011		0x1000_0000	ROM Wait_States = 0xF JTAG/SW
100		-	-
101	UART		PD[1:0] UART2
110	UART		PF[1:0] UART2
111		-	-



## 8.1 UART загрузчик

UART, UART2,  
13.

**Таблица 13 – Используемые порты ввода/вывода UART загрузчиком**

Режим MODE[2:0]	TX	RX
101	PD[1]	PD[0]
110	PF[1]	PF[0]

(Flash-),  
(EEPROM, ROM, ).

UART2, RC-  
HSI, 8, HSI,  
UART2

### 8.1.1 Параметры связи по UART

- UART :
- – 9600 ;
  - – 8;
  - – ;
  - Stop – 1;
  - FIFO UART2;
  - Slave,
  - Master;
  - .

### 8.1.2 Протокол обмена по UART

(Master),  
:

**Таблица 14 – Команды UART загрузчика**

Команда	Код	ASCII Символ	Описание
CMD_SYNC	0x00		
CMD_CR	0x0D		Master-
CMD_BAUD	0x42	'B'	
CMD_LOAD	0x4C	'L'	
CMD_VFY	0x59	'Y'	
CMD_RUN	0x52	'R'	

### 8.1.3 Синхронизация с внешним устройством

Master (Master) Rx  
 - 0.  
 (3 0x0D  
 ), 0x0A ( ), 0x3E ('>'), Master-  
 Master

### 8.1.4 Команда CMD\_SYNC

(Slave)

**Таблица 15 – Команда CMD\_SYNC**

	CMD_SYNC = 0x00		
ASCII			
			0
Master	CMD_SYNC	Slave	ERR_CHN ERR_CMD

### 8.1.5 Команда CMD\_CR

Master-

**Таблица 16 – Команда CMD\_CR**

	CMD_CR = 0x0D		
ASCII			
			0
Master	CMD_CR	Slave	ERR_CHN ERR_CMD CMD_CR. 0x0A 0x3E (ASCII '>')

### 8.1.6 Команда CMD\_BAUD

**Таблица 17 – Команда CMD\_BAUD**

	CMD_BAUD = 0x42
ASCII ,	'B'
	1
	[ ]
:	
Master CMD_BAUD	Slave ERR_CHN ERR_CMD
Master	ERR_CHN ERR_BAUD CMD_BAUD.

### 8.1.7 Команда CMD\_LOAD

**Таблица 18 – Команда CMD\_LOAD**

	CMD_LOAD = 0x4C
ASCII ,	'L'
	2
1.	
2.	
:	
Master CMD_LOAD	Slave ERR_CHN ERR_CMD
Master 1.	Slave ERR_CHN
Master 2.	Slave ERR_CHN CMD_LOAD
Master	Slave ERR_CHN REPLY_OK = 0x4B ('K')

### 8.1.8 Команда CMD\_VFY

**Таблица 19 – Команда CMD\_VFY**

		CMD_VFY = 0x59	
ASCII	,	'Y'	
		2	
	1		
	2		
	:		
Master	CMD_VFY	Slave	ERR_CHN ERR_CMD
Master	1	Slave	ERR_CHN
Master	2	Slave	ERR_CHN CMD_VFY. REPLY_OK = 0x4B ( 'K' )

### 8.1.9 Команда CMD\_RUN

**Таблица 20 – Команда CMD\_RUN**

		CMD_RUN = 0x52	
ASCII	,	'R'	
		1	
	:		
Master	CMD_RUN.	Slave	ERR_CHN ERR_CMD
Master		ERR_CHN	CMD_RUN. MSP PC (NVIC ) , Slave

### 8.1.10 Прием параметров команды

– 4

0xFFFFFFFF.

(UART '1' -

),

### 8.1.11 Сообщения об ошибках

– 2-

0x45 ('E').

Master

Master

CMD\_CR

: ERR\_CHN, ERR\_CMD, ERR\_BAUD

#### **Ошибка ERR\_CHN**

UART.

0x69 ('i').

UART '1'

#### **Ошибка ERR\_CMD**

0x63 ('c').

#### **Ошибка ERR\_BAUD**

0x62 ('b').

Master-  
UART.

## 9 Контроллер Flash-памяти программ MDR\_EEPROM

Flash- 128

4

( CON = 0, EEPROM\_CMD)  
I Code D code

( CON = 1, EPROM\_CMD)

Flash-

### 9.1 Работа Flash-памяти программ в обычном режиме

Flash- 40 , 25

Flash- 8 128

(4 8 32 ).

40 Flash- 16 ,

4 8

Flash. 25 ;

25 50 Flash-

EEPROM\_CMD Delay[2:0].

Flash-

21

**Таблица 21 – Дополнительная пауза для работы Flash-памяти**

Delay[2:0]	Тактов паузы	Тактовая частота	Примечание
0x00	0	25	
0x01	1	50	
0x02	2	75	
0x03	3	100	80
0x04	4	125	
0x05	5	150	
0x06	6	175	
0x07	7	200	





3 - Sector\_C ;  
 4 - Sector\_D .

IFREN (1 -

0 - ),

EEPROM\_ADR[3:2] (00 - Sector\_A, 01 - Sector\_B, 10 - Sector\_C 11 - Sector\_D)

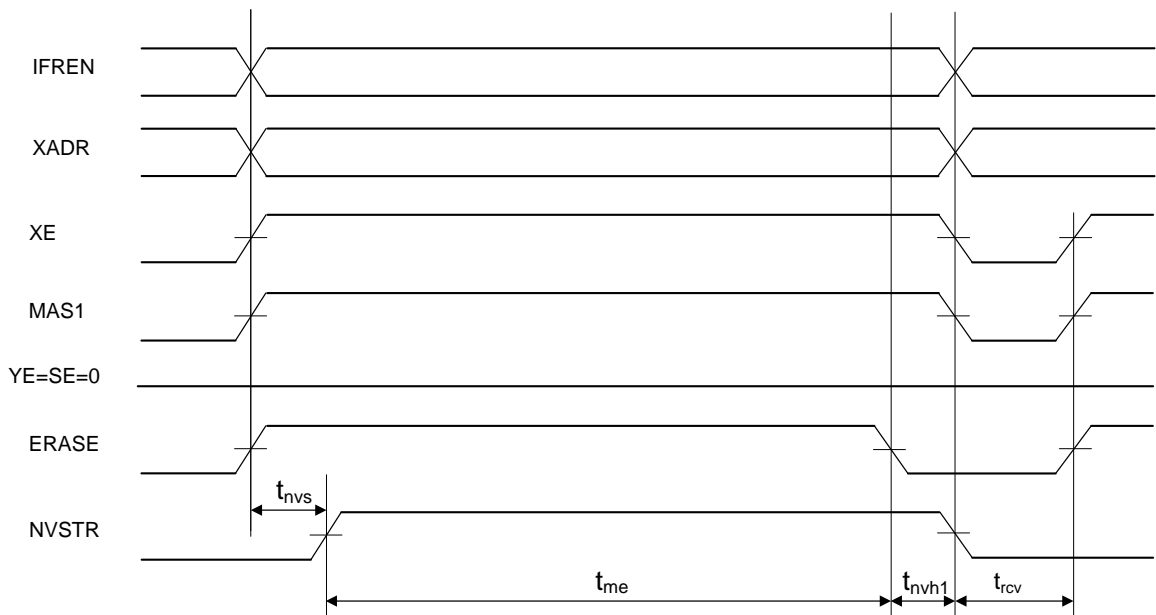
XE, MAS1 ERASE , t<sub>nvs</sub> = 5

NVSTR . t<sub>me</sub> = 40 .

ERASE, t<sub>nvh1</sub> = 100 XE, MAS1

NVSTR. t<sub>rcv</sub> = 1 .

( . 16).



**Рисунок 16. Временная диаграмма стирания памяти**

**9.2.2 Стирание банка памяти одной страницы размером 4 Кбайт**

4 :

1 - Sector\_A ;  
 2 - Sector\_B ;  
 3 - Sector\_C ;  
 4 - Sector\_D .

IFREN (1 -

0 - ),

EEPROM\_ADR[16:12] EEPROM\_ADR[3:2] (00 - Sector\_A,

01 - Sector\_B, 10 - Sector\_C 11 - Sector\_D) XE ERASE ,

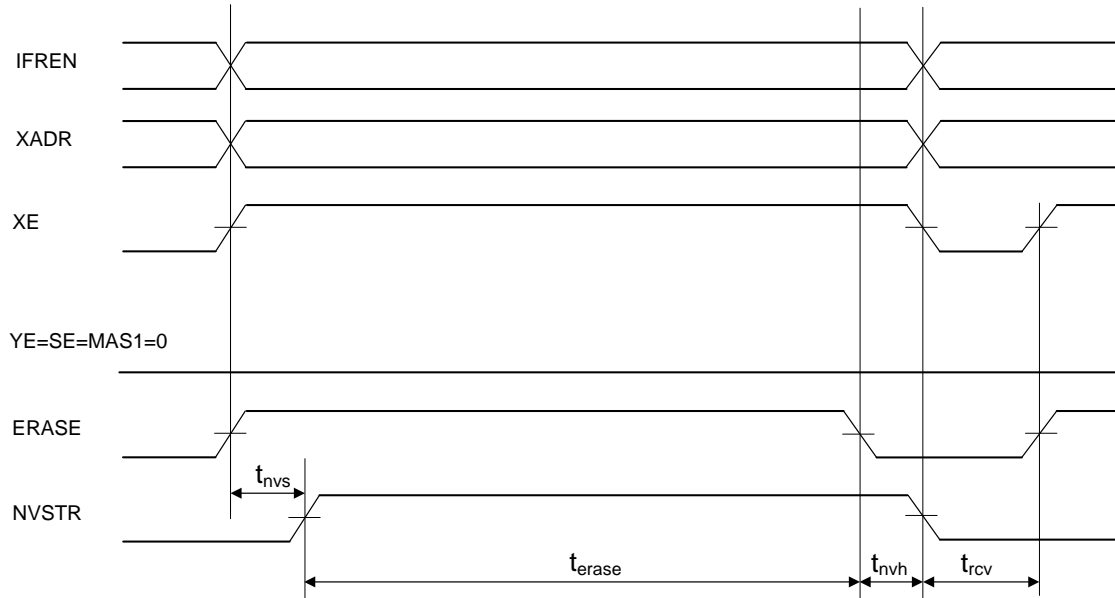
t<sub>nvs</sub> = 5 NVSTR .

terase = 40 . ERASE, t<sub>nvh</sub> =

5 XE NVSTR.

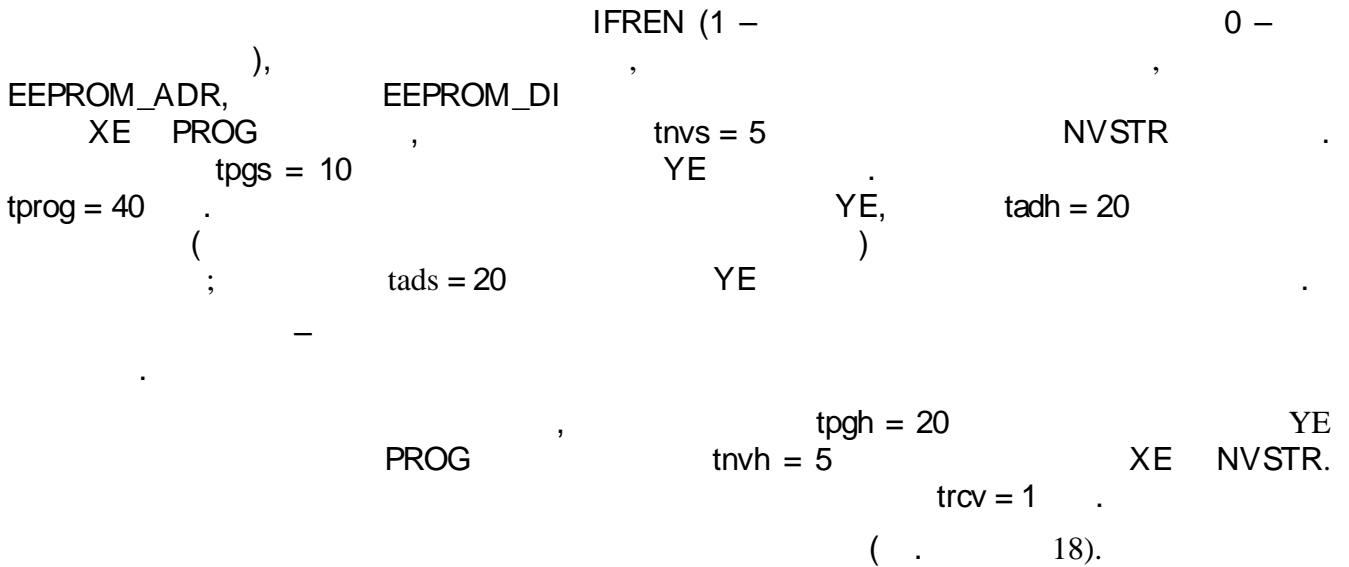
trcv = 1  
17).

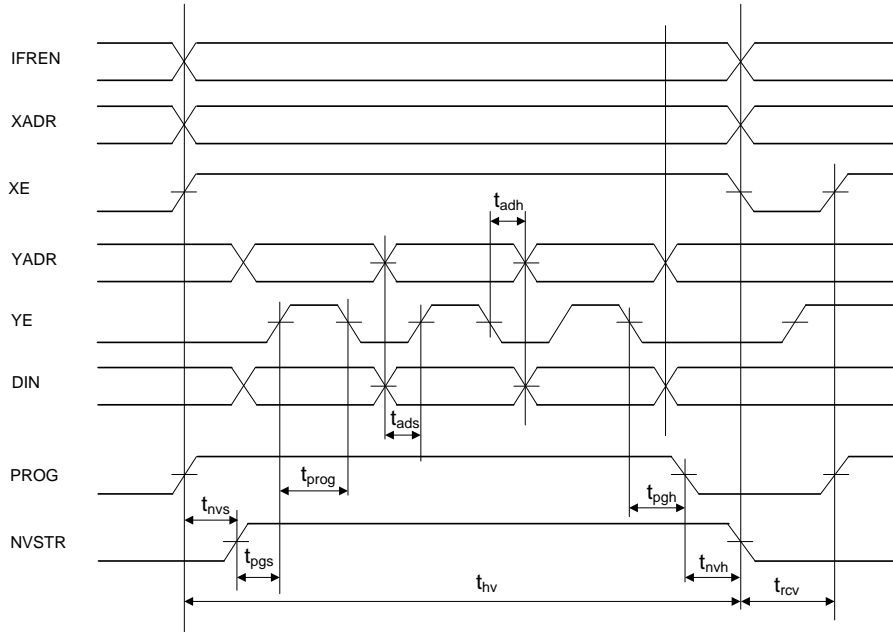
( .



**Рисунок 17. Временная диаграмма стирания банка памяти**

**9.2.3 Запись 32-х битного слова в память**

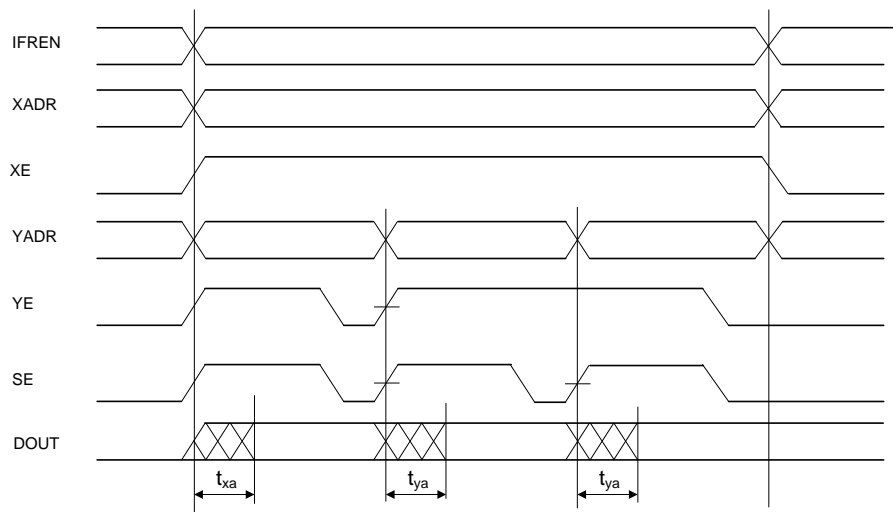




**Рисунок 18. Временная диаграмма записи памяти**

**9.2.4 Чтение 32-х битного слова из памяти**

IFREN (1 – 0 – ),  
 EEPROM\_ADR, SE  
 XE, YE  
 EEPROM\_DO  
 EEPROM\_ADR  
 EEPROM\_DO  
 txa = 30  
 txa = 30  
 ( 19).



**Рисунок 19. Временная диаграмма чтения памяти**

Flash-

10 000

4

### 9.3 Регистры управления контроллера Flash-памяти программ

22

Flash-

Таблица 22 – Регистры управления контроллера Flash-памяти программ

Базовый адрес	Название	Описание
0x4001_8000	MDR_EEPROM	Flash-
<b>Смещение</b>		
0x00	CMD	
0x04	ADR	
0x08	DI	
0x0C	DO	
0x10	KEY	

R/W - ;  
RO - ;  
U -

**9.3.1 MDR\_EEPROM->CMD**

**Таблица 23 – Регистр команды EEPROM\_CMD**

<b>Номер</b>	31...14	13	12	11	10
<b>Доступ</b>	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
		<b>NVSTR</b>	<b>PROG</b>	<b>MAS1</b>	<b>ERASE</b>

<b>Номер</b>	9	8	7	6	5...3	2, 1	0
<b>Доступ</b>	R/W	R/W	R/W	R/W	R/W	U	R/W
<b>Сброс</b>	0	0	0	0	100	0	0
	<b>IFREN</b>	<b>SE</b>	<b>YE</b>	<b>XE</b>	<b>Delay[2:0]</b>		<b>CON</b>

**Таблица 24 – Описание бит регистра EEPROM\_CMD**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...14	-	
13	NVSTR	: 0 – ; 1 –
12	PROG	ADR[16:2] EEPROM_DI: 0 – ; 1 –
11	MAS1	ERASE =1: 0 – ADR[16:12] ADR[3:2]; 1 – ADR[3:2]
10	ERASE	: 0 – ; 1 –
9	IFREN	: 0 – ; 1 –
8	SE	: 0 – ; 1 –
7	YE	ADR[8:2]: 0 – ; 1 –
6	XE	ADR[16:9]: 0 – ; 1 –
5...3	Delay[2:0]	( ): 000 – 0 001 – 1 111 – 7

2, 1	-	
0	CON	EEPROM
		EEPROM: 0 – EEPROM ; 1 –

### 9.3.2 MDR\_EEPROM->ADR

**Таблица 25 – Регистр адреса EEPROM\_ADR**

<b>Номер</b>	31...0
<b>Доступ</b>	R/W
<b>Сброс</b>	0
	<b>ADR [31:0]</b>

**Таблица 26 – Описание бит регистра адреса EEPROM\_ADR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	ADR[31:0]	: ADR[1:0] – 32

### 9.3.3 MDR\_EEPROM->DI

**Таблица 27 – Регистр записываемых данных EEPROM\_DI**

<b>Номер</b>	31...0
<b>Доступ</b>	R/W
<b>Сброс</b>	0
	<b>DATA [31:0]</b>

**Таблица 28 – Описание бит регистра записываемых данных EEPROM\_DI**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	DATA[31:0]	EEPROM

### 9.3.4 MDR\_EEPROM->DO

**Таблица 29 – Регистр считываемых данных EEPROM\_DO**

<b>Номер</b>	31...0
<b>Доступ</b>	R/W
<b>Сброс</b>	0
	<b>DATA [31:0]</b>

**Таблица 30 – Описание бит регистра считываемых данных EEPROM\_DO**

№	Функциональное	Расшифровка функционального имени бита, краткое
---	----------------	---

бита	имя бита	описание назначения и принимаемых значений
31...0	DATA[31:0]	, EEPROM

### 9.3.5 MDR\_EEPROM->KEY

**Таблица 31 – Регистр ключа EEPROM\_KEY**

	31...0
	R/W
	0
	KEY [31:0]

**Таблица 32 – Описание бит регистра ключа EEPROM\_KEY**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	KEY[31:0]	Flash-  EEPROM_KEY  0x8AAA5551



## 10 Процессорное ядро ARM Cortex-M3

- Cortex-M3 – 32-;
- ;
- ;
- ;
- Sleep;
- MPU.

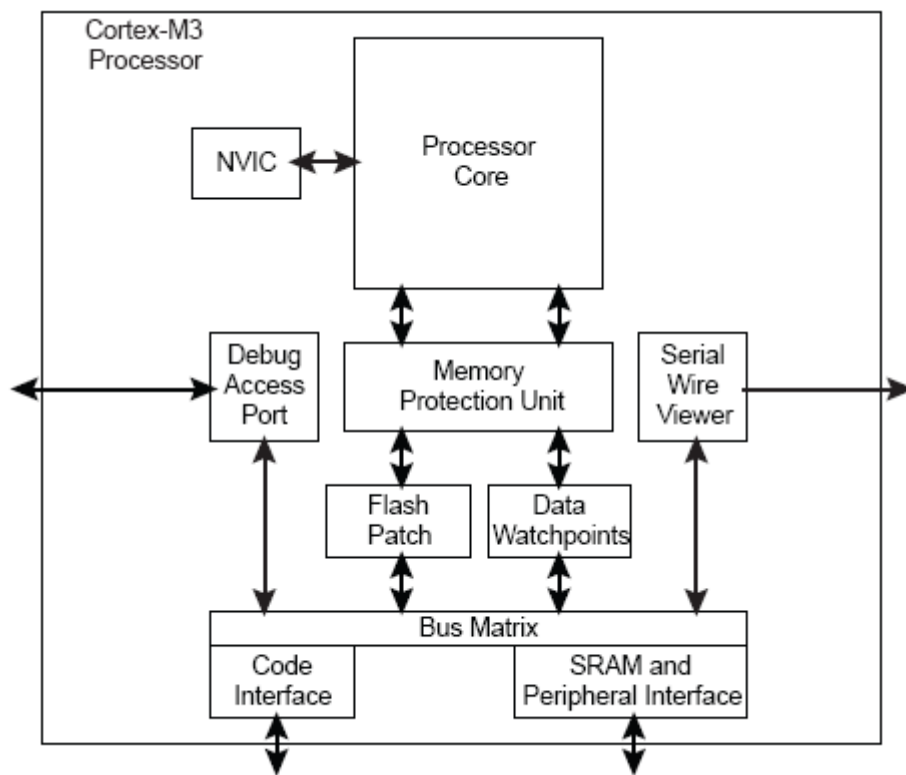


Рисунок 20. Структурная блок-схема процессорного ядра Cortex-M3

Cortex-M3

3-

32 32

Thumb2,

Cortex-M3

32-

8- 16-

NVIC,

. NVIC

8-

(interrupt service routines – ISR),

Sleep Deep Sleep,  
Cortex-M3

NVIC

Cortex-M3 (MPU)

Cortex-M3

JTAG

2-

SWD.

ITM,

- NVIC

- SBC

- SysTick

24-

- MPU

8-

## 10.1 Программная модель

- Thread

- Handler

Thread

- Unprivileged

MSR MRS ,  
 CPS , NVIC  
 unprivileged.

- Privileged

privileged.

Thread CONTROL handler  
 unprivileged privileged.  
 privileged CONTROL.  
 Thread SVC supervisor call

## 10.2 Стек

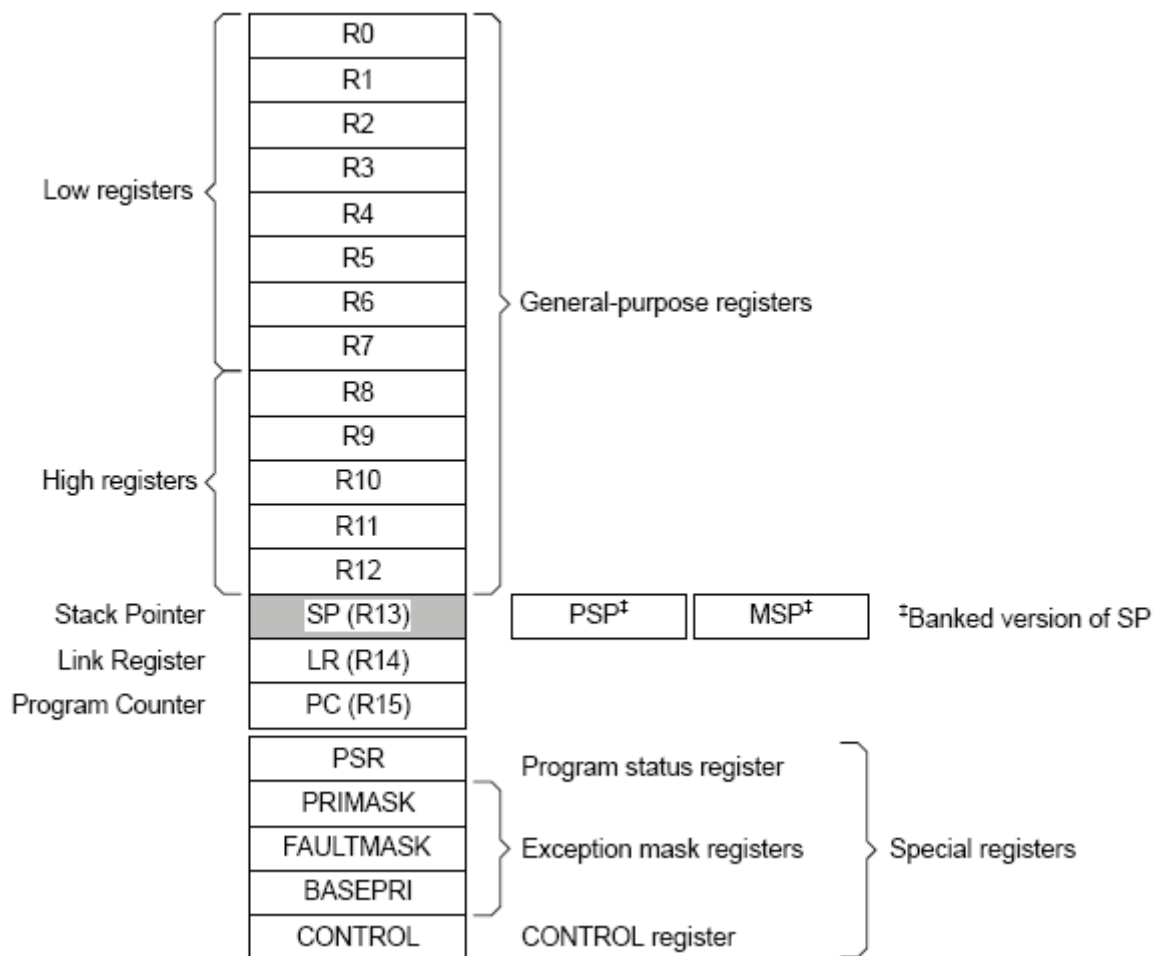
Thread CONTROL – main process  
 process, CONTROL – main  
 main Handler

**Таблица 33 – Режимы работы процессора при выполнении программы**

Режим процессора	Использование	Уровни привилегии для программного обеспечения	Используемый стек
Thread		Privileged Unprivileged <sup>(1)</sup>	Main Process <sup>(1)</sup>
Handler		Privileged	Main

1. CONTROL

## 10.3 Регистры ядра



**Рисунок 21. Регистры ядра**

**Таблица 34 – Сводная таблица регистров ядра**

Название	Тип <sup>(1)</sup>	Требуемый уровень привилегий	Значение после сброса	Описание
R0-R12	RW	<sup>(2)</sup>		
MSP	RW	Privileged	.	<i>main</i> Stack Pointer
PSP	RW	<sup>(2)</sup>		<i>process</i> Stack Pointer
LR	RW	<sup>(2)</sup>	0xFFFFFFFF	Link Register
PC	RW	<sup>(2)</sup>	.	Program Counter
PSR	RW	Privileged	0x01000000	Program Status Register
ASPR	RW	<sup>(2)</sup>	0x00000000	Application Program Status Register
IPSR	RO	Privileged	0x00000000	Interrupt Program Status Register

ESPR	RO	Privileged	0x01000000	Execution Program Status Register
PRIMASK	RW	Privileged	0x00000000	Priority Mask Register
FAULTMASK	RW	Privileged	0x00000000	Fault Mask Register
BASEPRI	RW	Privileged	0x00000000	Base Priority Mask Register
CONTROL	RW	Privileged	0x00000000	CONTROL Register

1. *thread handler*
- 2.

### 10.3.1 Регистры общего назначения R0-R12

R0-R12 - 32-

### 10.3.2 Указатель стека SP R13

Stack Pointer Register (SP) - R13. Thread 1 CONTROL

- 0 – Main Stack Pointer (MSP);
- 1 – Process Stack Pointer (PSP).

MSP 0x00000000.

### 10.3.3 Регистр связи LR R14

Link Register - R14.

0xFFFFFFFF.

### 10.3.4 Счетчик команд PC R15

Program Counter – R15. 0

0,

0x00000004.

### 10.3.5 Программный регистр состояния PSR

Program Status Register (PSR) :

- Application Program Status Register (APSR);
- Interrupt Program Status Register (IPSR);
- Execution Program Status Register (EPSR).

32- PSR.

MSR MRS.

– : PSR MRS ;  
 – APSR, APSR MSR .

**Таблица 35 – Комбинация PSR и их атрибуты**

Регистр	Тип	Комбинация
PSR	RW (1),(2)	APSR, EPSR IPSR
IEPSR	RO	EPSR IPSR
IAPSR	RW(1)	APSR IPSR
EAPSR	RW(2)	APSR EPSR

3. IPSR

4. EPSR

MRS MSR.

### 10.3.6 Программный регистр состояния приложения APSR

APSR

**Таблица 36 – Регистр APSR**

Номер	31	30	29	28	27	26...0
Доступ	R/W	R/W	R/W	R/W	R/W	
Сброс	0	0	0	0	0	
	N	Z	C	V	Q	-

**Таблица 37 – Описание бит регистра APSR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31	N	<b>Negative</b> 0 – « »; 1 – « ».
30	Z	<b>Zero:</b> 0 – ; 1 – .
29	C	<b>Carry:</b> 0 – , ; 1 – .
28	V	<b>Overflow:</b> 0 – ; 1 – .

27	Q	Saturation: 0 – , ; 1 – , SSAT USAT . MRS.
26...0	-	

### 10.3.7 Программный регистр состояния прерываний IPSR

IPSR

**Таблица 38 – Регистр IPSR**

<b>Номер</b>	31...9	8...0
<b>Доступ</b>	-	RO
<b>Сброс</b>	-	0
	-	<b>ISR_NUMBER</b>

**Таблица 39 – Описание бит регистра IPSR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...9	-	
8...0	ISR_NUMBER	<b>Номер текущего исключения</b> 0 – Thread ; 1 – ; 2 – NMI; 3 – Hard Fault; 4 – Memory Management Fault; 5 – Bus Fault; 6 – Usage Fault; 7...10 – ; 11 – SVCcall; 12 – ; 13 – PendSV; 15 – SysTick; 16 – IRQ0; ... 48 – IRQ31.







### 10.3.12 Регистр базового приоритета маски Base Priority Mask

BASEPRI

BASEPRI

BASEPRI.

**Таблица 46 – Регистр BASEPRI**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>BASEPRI</b>

**Таблица 47 – Описание бит регистра BASEPRI**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	BASEPRI	0 –

BASEPRI.

[7:5]

[4:0]

447 –

### 10.3.13 Регистр управления CONTROL

CONTROL

Thread

**Таблица 48 – Регистр CONTROL**

<b>Номер</b>	31...2	1	0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	<b>Active Stack Pointer</b>	<b>Thread Mode Privilege Level</b>

**Таблица 49 – Описание бит регистра CONTROL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	-	
1	Active Stack Pointer	0 – MSP 1 – PSP
0	Thread Mode Privilege Level	0 – 1 –

Handler Active Stack Pointer MSP, CONTROL handler .  
 CONTROL.  
 , , Thread  
 , PSP , MSP .  
 Thread , Thread MSP.  
 1, PSP , MSR Active Stack Pointer  
 MSR MSR  
 MSR , ISB  
 ISB, .  
 ISB.

## 10.4 Исключения и прерывания

ARM Cortex-M3  
 NVIC  
 handler  
 NVIC «  
 NVIC» .  
 :  
 • 32- words;  
 • 16- halfwords;  
 • 8- bytes.  
 64-  
 little-endian  
 Private Peripheral Bus (PPB) little-endian «  
 » .

## 11 Система команд

Thumb.

( 50).

:

- <>
- ;
- { }
- " "
- Op2 , ;
- .

Таблица 50 – Система команд процессора Cortex-M3

Мnemonic коды команд	Операнды	Краткое описание	Флаги	Прим.
ADC, ADCS	{Rd,} Rn, Op2		N,Z,C,V	
ADD, ADDS	{Rd,} Rn, Op2		N,Z,C,V	
ADD, ADDW	{Rd,} Rn, #imm12		N,Z,C,V	
ADR	Rd, label	,	-	
AND, ANDS	{Rd,} Rn, Op2		N,Z,C	
ASR, ASRS	Rd, Rm, <Rs #n>		N,Z,C	
B	label		-	
BFC	Rd, #lsb, #width		-	
BFI	Rd,Rn,#lsb,#width		-	
BIC, BICS	{Rd,} Rn, Op 2		N,Z,C	
BKPT	#imm		-	
BL	label		-	
BLX	Rm		-	
BX	Rm		-	
CBNZ	Rn, label		-	
CBZ	Rn, label		-	
CLREX	-		-	
CLZ	Rd, Rm		-	
CMN, CMNS	Rn, Op2		N,Z,C,V	
CMP, CMPS	Rn, Op2		N,Z,C,V	
CPSID	iflags	,	-	
CPSIE	iflags	,	-	

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

Мнемокод команды	Операнды	Краткое описание	Флаги	Прим.
DMB	-		-	
DSB	-		-	
EOR, EORS	{Rd,} Rn, Op2		N,Z,C	
ISB	-		-	
IT	-		-	
LDM	Rn{!}, reglist	,	-	
LDMDB, LDMEA	Rn{!}, reglist	,	-	
LDMFD, LDMIA	Rn{!}, reglist	,	-	
LDR	Rt, [Rn, #offset]		-	
LDRB, LDRBT	Rt, [Rn, #offset]		-	
LDRD	Rt, Rt2, [Rn,#offset]		-	
LDREX	Rt, [Rn, #offset]		-	
LDREXB	Rt, [Rn]	,	-	
LDREXH	Rt, [Rn]	,	-	
LDRH, LDRHT	Rt, [Rn, #offset]		-	
LDRSB, LDRSBT	Rt, [Rn, #offset]		-	
LDRSH, LDRSHT	Rt, [Rn, #offset]		-	
LDRT	Rt, [Rn, #offset]		-	
LSL, LSLs	Rd, Rm, <Rs #n>		N,Z,C	
LSR, LSRS	Rd, Rm, <Rs #n>		N,Z,C	
MLA	Rd, Rn, Rm, Ra	, 32-	-	
MLS	Rd, Rn, Rm, Ra	, 32-	-	
MOV, MOVS	Rd, Op2		N,Z,C	
MOVT	Rd, #imm16		-	
MOVW, MOV	Rd, #imm16	16-	N,Z,C	
MRS	Rd, spec_reg		-	
MSR	spec_reg, Rm		N,Z,C,V	
MUL, MULS	{Rd,} Rn, Rm	, 32-	N,Z	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Мнемокод команды	Операнды	Краткое описание	Флаги	Прим.
MVN, MVNS	Rd, Op2		N,Z,C	
NOP	-		-	
ORN, ORNS	{Rd,} Rn, Op2	-	N,Z,C	
ORR, ORRS	{Rd,} Rn, Op2		N,Z,C	
POP	reglist		-	
PUSH	reglist		-	
RBIT	Rd, Rn		-	
REV	Rd, Rn		-	
REV16	Rd, Rn		-	
REVSH	Rd, Rn		-	
ROR, RORS	Rd, Rm, <Rs #n>		N,Z,C	
RRX, RRXS	Rd, Rm		N,Z,C	
RSB, RSBS	{Rd,} Rn, Op2		N,Z,C,V	
SBC, SBCS	{Rd,} Rn, Op2		N,Z,C,V	
SBFX	Rd, Rn, #lsb, #width		-	
SDIV	{Rd,} Rn, Rm		-	
SEV	-		-	
SMLAL	RdLo, RdHi, Rn, Rm	, 64-	-	
SMULL	RdLo, RdHi, Rn, Rm	, 64-	-	
SSAT	Rd,#n,Rm{,shift#s}	32- n-	Q	
STM	Rn{!}, reglist		-	
STMDB, STMEA	Rn{!}, reglist		-	
STMFD, STMIA	Rn{!}, reglist		-	
STR	Rt, [Rn, #offset]		-	
STRB, STRBT	Rt, [Rn, #offset]		-	
STRD	Rt, Rt2, [Rn, #offset]		-	
STREX	Rd, Rt, [Rn, #offset]		-	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Мнемокод команды	Операнды	Краткое описание	Флаги	Прим.
STREXB	Rd, Rt, [Rn]	,	-	
STREXH	Rd, Rt, [Rn]	,	-	
STRH, STRHT	Rt, [Rn, #offset]	,	-	
STRT	Rt, [Rn, #offset]	,	-	
SUB, SUBS	{Rd,} Rn, Op2		N,Z,C,V	
SUB, SUBW	{Rd,} Rn, #imm12		N,Z,C,V	
SVC	#imm		-	
SXTB	{Rd,}Rm{,ROR#n}		-	
SXTH	{Rd,}Rm{,ROR#n}		-	
TBB	[Rn, Rm]	,	-	
TBH	[Rn, Rm, LSL #1]	,	-	
TEQ	Rn, Op2		N,Z,C	
TST	Rn, Op2		N,Z,C	
UBFX	Rd, Rn, #lsb, #width	,	-	
UDIV	{Rd,} Rn, Rm		-	
UMLAL	RdLo, RdHi, Rn, Rm	, 64-	-	
UMULL	RdLo, RdHi, Rn, Rm	, 64-	-	
USAT	Rd,#n,Rm{,shift#s}	32- n- ,	Q	
UXTB	{Rd,}Rm{,ROR#n}		-	
UXTH	{Rd,}Rm{,ROR#n}		-	
WFE	-		-	
WFI	-		-	

## 11.1 Встроенные функции

ANSI C  
Cortex-M3. (intrinsic)

CMSIS  
ANSI C.

**Таблица 51 – Встроенные функции CMSIS, позволяющие генерировать некоторые инструкции процессора Cortex-M3**

Мнемокод команды процессора	Описание встроенной функции
CPSIE I	void __enable_irq(void)
CPSID I	void __disable_irq(void)
CPSIE F	void __enable_fault_irq(void)
CPSID F	void __disable_fault_irq(void)
ISB	void __ISB(void)
DSB	void __DSB(void)
DMB	void __DMB(void)
REV	uint32_t __REV(uint32_t int value)
REV16	uint32_t __REV16(uint32_t int value)
REVSH	uint32_t __REVSH(uint32_t int value)
RBIT	uint32_t __RBIT(uint32_t int value)
SEV	void __SEV(void)
WFE	void __WFE(void)
WFI	void __WFI(void)

, CMSIS

MRS MSR.

**Таблица 52 – Встроенные функции CMSIS для доступа к специальным регистрам процессора**

Наименование специального регистра	Режим доступа	Описание встроенной функции
PRIMASK		uint32_t __get_PRIMASK (void)
		void __set_PRIMASK (uint32_t value)
FAULTMASK		uint32_t __get_FAULTMASK (void)
		void __set_FAULTMASK (uint32_t value)
BASEPRI		uint32_t __get_BASEPRI (void)
		void __set_BASEPRI (uint32_t value)
CONTROL		uint32_t __get_CONTROL (void)
		void __set_CONTROL (uint32_t value)
MSP		uint32_t __get_MSP (void)
		void __set_MSP (uint32_t TopOfMainStack)
PSP		uint32_t __get_PSP (void)
		void __set_PSP (uint32_t TopOfProcStack)





constant

ORRS, ORNS, EORS, BICS, TEQ TST Operand2 MOVNS, MVNS, ANDS, 255

Operand2 [31]

Operand2

« »,

CMP Rd, #0xFFFFFFFFE

CMN Rd, #0x2.

### 11.2.3.2 Регистр с необязательным параметром сдвига

Operand2 : Rm {, shift}

Rm – ;

shift – Rm.

ASR #n - n, 1 n 32;

LSL #n - n, 1 n 31;

LSR #n - n, 1 n 32;

ROR #n - n, 1 n 31;

RRX -

LSL #0.

Rm -

Rm 32- Rm,

Rm

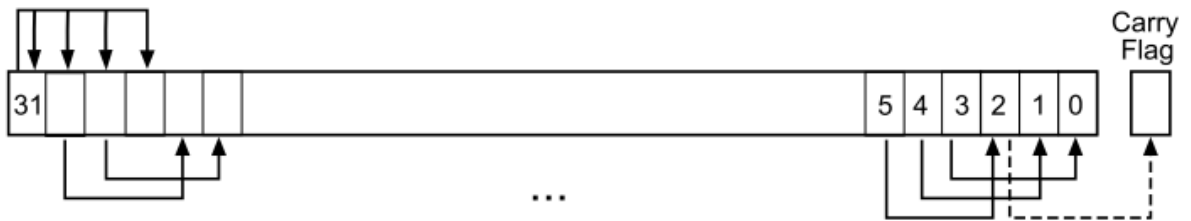
### 11.2.4 Операции сдвига

- ASR, LSR, LSL, ROR RRX,
  - Operand2
- ( . .95). 0,

0.  
(Rm – , n – ).

**11.2.4.1 ASR**

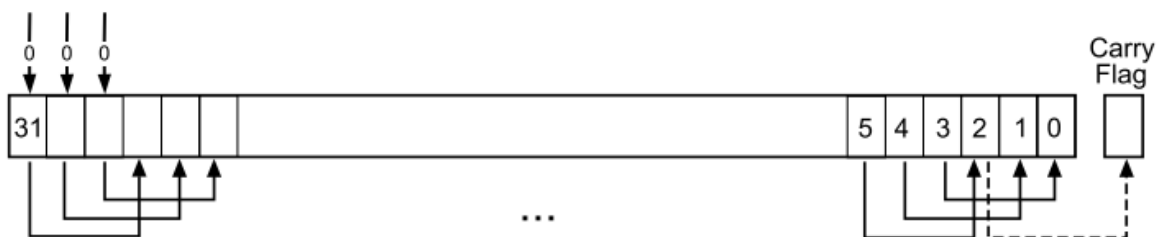
n , n 32-n. [31] Rm  
ASR # 3.  
ASR # n Rm 2<sup>n</sup>,  
( ).  
ASRS, ASR #n  
TEQ TST, [n-1] Rm.  
n 32, [31]  
Rm. [31] Rm.



**Рисунок 22. Инструкция ASR # 3**

**11.2.4.2 LSR**

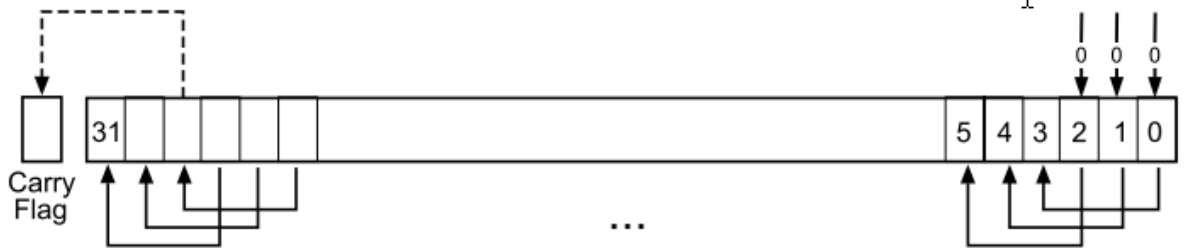
n , n 32-n. Rm  
n 0. 23 LSR # 3.  
LSR # n Rm 2<sup>n</sup>,  
( ).  
LSRS, LSR #n  
TEQ TST, [n-1] Rm.  
n 32, 0. n 33  
0.



**Рисунок 23. Инструкция LSR # 3**

**11.2.4.3 LSL**

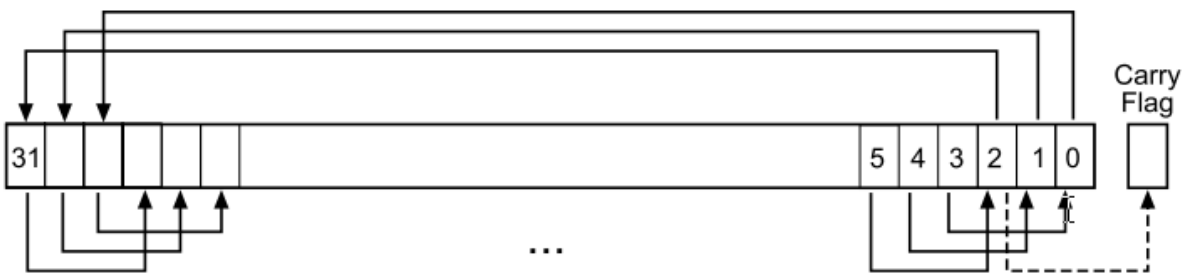
$n$ ,  $0$ ,  $24$ ,  $32-n$ ,  $LSL \# 3$ ,  $n$ ,  $32-n$ ,  $Rm$   
 $LSL \# n$ ,  $Rm$ ,  $2^n$   
 LSLs,  $LSL \# n$   
 MOVs, MVNS, ANDS, ORRS, ORNS, EORS, BICS,  
 TEQ, TST,  $[32-n]$ ,  $Rm$ ,  $LSL \# 0$   
 $n$ ,  $32$ ,  $0$ ,  $n$ ,  $33$



**Рисунок 24. Инструкция LSL # 3**

**11.2.4.4 ROR**

$n$ ,  $n$ ,  $32-n$ ,  $32-n$ ,  $n$ ,  $Rm$   
 $ROR \# 3$ ,  $25$   
 RORS,  $ROR \# n$   
 MOVs, MVNS, ANDS, ORRS, ORNS, EORS, BICS,  
 TEQ, TST,  $[n-1]$ ,  $Rm$ ,  $[n-$   
 $n = 32$ ,  $n = 32$   
 $[31]$   
 $Rm$ ,  $ROR$ ,  $32$ ,  
 $n-32$



**Рисунок 25. Инструкция ROR # 3**

### 11.2.4.5 RRX

Rm

RRX. [31]  
 RRXS, RRX #n  
 MOVs, MVNS, ANDS, ORRS, ORNS, EORS, BICS,  
 TEQ TST, [0] Rm.

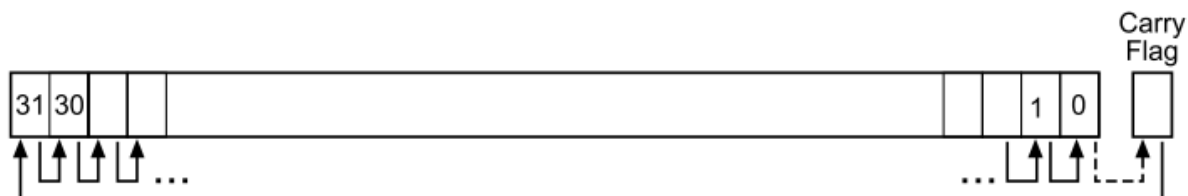


Рисунок 26. Инструкция RRX

### 11.2.5 Выравнивание адресов

Cortex-M3

- LDR, LDRT;
- LDRH, LDRHT;
- LDRSH, LDRSHT;
- STR, STRT;
- STRH, STRHT.

(usage fault).

ARM

1

UNALIGN\_TRP

CCR,  
 ("SCB->CCR").

### 11.2.6 Адресация относительно счетчика команд PC

Cortex-M3

PC /

- B, BL, CBNZ CBZ  
 4 ;

- 4 , [1] 0
- PC, " / " [PC, #number].

### 11.2.7 Условное исполнение

« (APSR) APSR»).

53

- ;
  - - ;
  - ;
  - .
- ( - IT- IT ). IT- .

CBZ CBNZ.

- ;
- .

#### 11.2.7.1 Флаги условий

- N=1 , APSR ,0 :
- Z=1 , ,0 .
- C=1 , ,0 .
- V=1 , ,0 .

APSR « PSR».

- : 232;
- ;
- 

$2^{31}$ ,  $-2^{31}$ .

S. IT.

### 11.2.7.2 Суффиксы условного исполнения

{ cond}.

IT.

APSR.

( 53).

**Таблица 53 – Суффиксы условного исполнения**

Суффикс	Флаги	Значение
EQ	Z = 1	
NE	Z = 0	
CS HS	C = 1	,
CC LO	C = 0	,
MI	N = 1	
PL	N = 0	
VS	V = 1	
VC	V = 0	
HI	C = 1 and Z = 0	,
LS	C = 0 or Z = 1	,
GE	N = V	,
LT	N != V	,
GT	Z = 0 and N = V	,
LE	Z = 1 and N != V	,
AL	1	

### 11.2.7.3 Пример. Вычисление абсолютного значения

: R0 = ABS(R1).

MOVS R0, R1 ; R0 = R1,

IT MI ; IT

RSBMI R0, R1, #0 ; R0 = -R1





### 11.3 Команды доступа к памяти

54.

Таблица 54 – Команды доступа к памяти

Мнемокод	Краткое описание	Прим.
ADR	,	
CLREX		
LDM{mode}		
LDR{type}	,	
LDR{type}	,	
LDR{type}T		
LDR		
LDREX{type}		
POP		
PUSH		
STM{mode}		
STR{type}	,	
STR{type}	,	
STR{type}T		
STREX{type}		

### 11.3.1 ADR

#### Синтаксис

ADR{cond} Rd, label

:

cond - , " " .

Rd - - .

label - , " " .

#### Описание

ADR

PC

-

.

,

,

.

ADR

[0]

BX

BLX

1.

PC

-4095...+4095.

,

,

.W ( . « »).

#### Ограничения

Rd

SP

PC.

#### Флаги

#### Примеры

ADR R1, TextMessage

;

,

;

TextMessage,

R1.

### 11.3.2 LDR и STR, непосредственно заданное смещение

#### 11.3.2.1 Синтаксис

```

op{type}{cond} Rt, [Rn {, #offset}] ;
op{type}{cond} Rt, [Rn, #offset]! ;
op{type}{cond} Rt, [Rn], #offset ;
opD{cond} Rt, Rt2, [Rn {, #offset}] ;
opD{cond} Rt, Rt2, [Rn, #offset]! ;
opD{cond} Rt, Rt2, [Rn], #offset ;
:
op - :
- LDR - ;
- STR - .
type - :
- B - , ;
- SB - ,
( LDR);
- H - , ;
- SH - ,
( LDR);
- -32- .
cond - , " ".
Rt - ,
Rn - ,
offset - Rn.
Rt2 - ,

```

#### 11.3.2.2 Описание

```

LDR - .
STR - .
:

```

#### 11.3.2.3 Адресация со смещением

```

Rn.
Rn
:
[Rn, #offset].

```

**11.3.2.4 Адресация с пре-индексированием**

Rn.

Rn.

:

[Rn, #offset]!

**11.3.2.5 Адресация с пост-индексированием**

Rn

Rn,

Rn.

:

[Rn], #offset .

— “ ”.

55

**Таблица 55 – Диапазон значений смещения**

Тип инструкции	Смещение	Преиндексирование	Пост-индексирование
, ,	-255 4095	-255 255	-255 255
		4,	-1020 1020

**11.3.2.6 Ограничения**

- Rt PC SP
- ;
- Rt Rt2
- Rt Rt2. Rn
- Rt
- PC: [0] 1;
- [0] 0;
- IT-
- Rt SP ;
- Rt Rn PC;
- Rt Rt2. Rn

**11.3.2.7 Флаги**

**11.3.2.8 Примеры**

```

LDR R8, [R10] ; R8
; R10.
LDRNE R2, [R5, #960]! ; R2
; 960
; R5, R5 960.
STR R2, [R9, #const-struct] ; const-struct -
; 0-4095.
STRH R3, [R4], #4 ; R3,
; R4,
; R4 4
LDRD R8, R9, [R3, #0x20] ; R8
; R3, R9
; 36 R3
STRD R0, R1, [R8], #-16 ; R0
; R1
; R8,
; R8 16.
    
```

### 11.3.3 LDR и STR, смещение задано в регистре

#### 11.3.3.1 Синтаксис

op{type}{cond} Rt, [Rn, Rm {, LSL #n}]

op – :  
 - LDR :  
 - STR :  
 type – :  
 - B – ,  
 - SB – ,  
 ( LDR).  
 - H – ,  
 - SH – ,  
 ( LDR).  
 - –32- .  
 cond – , “ ”.  
 Rt – ,  
 Rn – ,  
 Rm – ,  
 LSL #n – , 0 3.

#### 11.3.3.2 Описание

LDR –

STR –

Rm Rn

“ ”.

#### 11.3.3.3 Ограничения

- Rn PC;
- Rm SP PC;
- Rt SP
- ;
- Rt PC ;
- Rt
- PC:
- [0] 1, ;

11.3.3.4 Флаги

11.3.3.5 Примеры

```
STR R0, [R5, R1] ; R0 , R5 R1
LDRSB R0, [R5, R1, LSL #1]
; , R5 R1,
; ,
; R0
STR R0, [R1, R2, LSL #2]
; R0 , R1+4*R2.
```

### 11.3.4 LDR and STR, непривилегированный доступ

#### 11.3.4.1 Синтаксис

op{type}T{cond} Rt, [Rn {, #offset}]

:

op – :

- LDR .
- STR .
- type – :
- B – , .
- SB – , .
- H – ( LDR).
- SH – , ( LDR).
- -32- .

cond – , “ ”.

Rt – ,

Rn – ,

offset – Rn 0 255.

#### 11.3.4.2 Описание

“LDR” STR,

#### 11.3.4.3 Ограничения

- Rn PC
- Rt SP PC.

#### 11.3.4.4 Флаги

#### 11.3.4.5 Примеры

STRBTEQ R4, [R7] ; R4

; , R7, ;

LDRHT R2, [R2, #8] ; , -

; R2 8 R2, .



**LDR, адресация относительно счетчика команд PC**

**11.3.4.6 Синтаксис**

LDR{type}{cond} Rt, label

LDRD{cond} Rt, Rt2, label ; Load two words

:  
 type – :  
 - B – ,  
 - SB – ,  
 ( LDR).  
 - H – ,  
 - SH – ,  
 ( LDR).  
 - –32-  
 cond – , “ ”.  
 Rt – ,  
 Rt2 – ,  
 label – , “ ” PC”.

**11.3.4.7 Описание**

LDR –  
 PC,  
 ,  
 - “ ”.  
 56  
 .W ( ”).

**Таблица 56 – Диапазон значений смещения**

Тип инструкции	Диапазон значений смещения
, ,	-4095 4095
, ,	-1020 1020

**11.3.4.8 Ограничения**

- :  
 • Rt PC SP  
 ;  
 • Rt2 PC SP;  
 • Rt Rt2

- Rt
- PC: [0] 1, ;
  - , IT-

#### 11.3.4.9 Флаги

#### 11.3.4.10 Примеры

```
LDR R0, LookUpTable ; R0
; LookUpTable;
LDRSB R7, localdata ; local data,
;
; R7.
```

### 11.3.5 LDM и STM

#### 11.3.5.1 Синтаксис

op{addr\_mode}{cond} Rn{!}, reglist

:

op –

- LDM
- STM
- addr\_mode –
- IA –
- DB –

cond –

Rn –

! –

reglist –

“ ”

Rn.

“-” ( )

LDM LDMFD LDMIA. LDMFD

(Full Descending stack).

LDMEA LDMDB,

(Empty Ascending stack).

STM STMEA STMIA. STMEA

STMFD STMDB,

#### 11.3.5.2 Описание

LDM reglist,

Rn.

STM reglist,

Rn.

LDM, LDMIA, LDMFD, STM, STMIA STMEA

Rn Rn+4\*(n-1), n - reglist.

Rn. "!", Rn+4\*(n-1)  
 LDMDB, LDMEA, STMDB STMFD  
 Rn Rn-4\*(n-1), n - reglist.  
 Rn. "!", Rn-4\*(n-1)  
 PUSH POP LDM STM.  
 "PUSH POP".

### 11.3.5.3 Ограничения

- Rn PC;
- reglist SP;
- STM reglist PC;
- LDM reglist PC LR;
- reglist Rn, "!".
- LDM reglist PC:
- [0] 1, ;
- , IT-

### 11.3.5.4 Флаги

### 11.3.5.5 Примеры

LDM R8,{R0,R2,R9} ; LDMIA - LDM  
 STMDB R1!,{R3-R6,R11,R12}

### 11.3.5.6 Примеры неправильного использования

STM R5!,{R5,R4,R9} ; R5 ;  
 LDM R2, {} ;

### 11.3.6 PUSH и POP

(full-descending stack).

#### 11.3.6.1 Синтаксис

PUSH{cond} reglist

POP{cond} reglist

:

cond –

reglist –

PUSH POP

STMDB LDM (LDMIA)  
SP,

PUSH POP

#### 11.3.6.2 Описание

PUSH –

POP –

“LDM STM”.

#### 11.3.6.3 Ограничения

- reglist SP;
  - PUSH PC;
  - POP PC
- LR.
- POP reglist PC:
  - [0] 1;
  - IT-

#### 11.3.6.4 Флаги

#### 11.3.6.5 Примеры

PUSH {R0,R4-R7}

PUSH {R2,LR}

POP {R0,R10,PC}

### 11.3.7 LDREX и STREX

#### 11.3.7.1 Синтаксис

LDREX{cond} Rt, [Rn {, #offset}]  
STREX{cond} Rd, Rt, [Rn {, #offset}]  
LDREXB{cond} Rt, [Rn]  
STREXB{cond} Rd, Rt, [Rn]  
LDREXH{cond} Rt, [Rn]  
STREXH{cond} Rd, Rt, [Rn]

:  
cond – , “ ”.  
Rd – ,  
Rt – ( )  
Rn – ,  
offset – –

#### 11.3.7.2 Описание

LDREX, LDREXB LDREXH

STREX, STREXB STREXH

### 11.3.7.3 Ограничения

- PC;
- SP
- STREX Rd Rt; Rn;
- offset 4 0 1020.

### 11.3.7.4 Флаги

### 11.3.7.5 Примеры

```

MOV R1, #0x1 ; R1
;
try:
LDREX R0, [LockAddr] ;
CMP R0, #0 ; ?
ITT EQ ; IT STREXEQ CMPEQ
STREXEQ R0, R1, [LockAddr] ;
CMPEQ R0, #0 ; ?
BNE try ; -
.... ; -

```

## **11.3.8 CLREX**

### **11.3.8.1 Синтаксис**

CLREX{cond}

:

cond –

, “ ”.

### **11.3.8.2 Описание**

	CLREX		STREX,
STREXB	STREXH	1	

’ ” “ ”.

### **11.3.8.3 Флаги**

### **11.3.8.4 Примеры**

CLREX



## 11.4 Инструкции обработки данных

57

:

Таблица 57 – Команды обработки данных

Мнемокод	Краткое описание	Прим.
ADC		
ADD		
ADDW		
AND		
ASR		
BIC		
CLZ		
CMN		
CMP		
EOR		
LSL		
LSR		
MOV		
MOVT		
MOVW	16-	
MVN		
ORN	-	
ORR		
RBIT		
REV		
REV16		
REVSH	-	
ROR		
RRX		
RSB		
SBC		
SUB		
SUBW		
TEQ		
TST		



- SP Rd ADD SUB,
  - Rn SP;
  - Operand2 3 LSL;
- SP Rn ADD SUB;
- PC Rd :
- ADD{cond} PC, PC, Rm :
  - S;
  - Rm PC SP;
  - , IT-
- Rn PC
  - ADD SUB ( ADD{cond} PC, PC, Rm)
    - S;
    - 0 4095.
    - PC [1:0]
    - 0b00 ,
    - ;
    - ,
    - PC. ARM
    - ADR, ;
    - PC Rd
    - ADD{cond} PC, PC, Rm
    - [0] , PC, ,

#### 11.4.1.4 Флаги

S, N, Z, C V

#### 11.4.1.5 Примеры

```

ADD R2, R1, R3
SUBS R8, R6, #240 ;
RSB R4, R4, #1280 ; R4 1280
ADCHI R11, R0, R3 ;
;C , Z
  
```

#### 11.4.1.6 Арифметика с повышенной разрядностью

#### 11.4.1.7 64-разрядное сложение

```

R2 , R3, 64-
R0 R1, R4 R5.
ADDS R4, R0, R2 ;
ADC R5, R1, R3 ;
  
```

### 11.4.1.8 96-разрядное вычитание

```

R6, R2 R8.
SUBS R6, R6, R9 ;
SBCS R9, R2, R1 ;
SBC R2, R8, R11 ;
    
```

### 11.4.2 AND, ORR, EOR, BIC и ORN

#### 11.4.2.1 Синтаксис

op{S}{cond}{Rd,} Rn, Operand2

op -

- AND -
- ORR -
- EOR -
- BIC -
- ORN -

S -

cond -

Rd -

Rn -

Operand2 -

#### 11.4.2.2 Описание

AND, ORR EOR

Operand2.

BIC

Rn,

Operand2.

ORN

Operand2

#### 11.4.2.3 Ограничения

SP

PC.

#### 11.4.2.4 Флаги

- 
- 
- 

N Z  
C  
";

V.

11.4.2.5    Примеры

AND R9, R2, #0xFF00  
ORREQ R2, R0, R5  
ANDS R9, R8, #0x19  
EORS R7, R11, #0x18181818  
BIC R0, R1, #0xab  
ORN R7, R11, R14, ROR #4  
ORNS R7, R11, R14, ASR #32

### 11.4.3 ASR, LSL, LSR, ROR и RRX

#### 11.4.3.1 Синтаксис

op{S}{cond} Rd, Rm, Rs

op{S}{cond} Rd, Rm, #n

RRX{S}{cond} Rd, Rm

:

op -

- ASR -

- LSL -

- LSR -

- ROR -

S -

cond -

Rd -

Rm -

Rs -

0 255.

n -

- ASR - 1 32;

- LSL - 0 31;

- LSR - 1 32;

- ROR - 1 31.

LSL{S}{cond} Rd, Rm, #0

MOV{S}{cond} Rd, Rm.

#### 11.4.3.2 Описание

ASR, LSL, LSR ROR

Rm

n

Rs.

RRX

Rm

Rd,

Rm

“

”.

#### 11.4.3.3 Ограничения

SP

PC.

#### 11.4.3.4 Флаги

- S, : ;
- N Z ;
- C ;

#### 11.4.3.5 Примеры

ASR R7, R8, #9 ; 9

LSLS R1, R2, #3 ; 3

LSR R4, R5, #6; 6

ROR R4, R5, R6 ; R6

RRX R4, R5 ;

#### 11.4.4 CLZ

##### **Синтаксис**

CLZ{cond} Rd, Rm

:

cond -

Rd -

Rm -

##### **Описание**

CLZ

32, Rm, Rd, Rm, 0 -  
[31].

##### **Ограничения**

SP

PC.

##### **Флаги**

##### **Примеры**

CLZ R4,R9

CLZNE R2,R3.



#### 11.4.5 CMP и CMN

##### **Синтаксис**

CMP{cond} Rn, Operand2

CMN{cond} Rn, Operand2

:

cond - , “ ”

Rn - , “ ”

Operand2 - “ ”

##### **Описание**

CMP Rn, Operand2  
SUBS, Rn, Operand2  
CMN Rn, Operand2  
ADDS, Rn, Operand2

##### **Ограничения**

- PC;
- Operand2 SP.

##### **Флаги**

N, Z, C V

##### **Примеры**

CMP R2, R9

CMN R0, #6400

CMPGT SP, R7, LSL #2

## 11.4.6 MOV и MVN

### 11.4.6.1 Синтаксис

MOV{S}{cond} Rd, Operand2

MOV{cond} Rd, #imm16

MVN{S}{cond} Rd, Operand2

S - "S" (signed) or "." (unsigned).  
 cond - "S" (signed) or "." (unsigned).  
 Rd - register name.  
 Operand2 - register name or "#imm16".  
 imm16 - 0 65535.

### 11.4.6.2 Описание

MOV Operand2, Rd, LSL #0, #n

MOV{S}{cond} Rd, Rm, #n

- ASR{S}{cond} Rd, Rm, #n
- LSL{S}{cond} Rd, Rm, #n
- LSR{S}{cond} Rd, Rm, #n
- ROR{S}{cond} Rd, Rm, #n
- RRX{S}{cond} Rd, Rm

MOV{S}{cond} Rd, Rm, ASR #n;  
 MOV{S}{cond} Rd, Rm, LSL #n n != 0;  
 MOV{S}{cond} Rd, Rm, LSR #n;  
 MOV{S}{cond} Rd, Rm, ROR #n;  
 MOV{S}{cond} Rd, Rm, RRX.

MOV{S}{cond} Rd, Rm, ASR Rs;  
 MOV{S}{cond} Rd, Rm, LSL Rs  
 MOV{S}{cond} Rd, Rm, LSR Rs  
 MOV{S}{cond} Rd, Rm, ROR Rs

ASR, LSL, LSR, ROR RRX.

MVN Operand2, Rd, MOV, #imm16.

### 11.4.6.3 Ограничения

- SP PC MOV, #imm16;
- S
- Rd PC:
- [0], PC;

- [0], 0. MOV
- , ARM BX BLX,

#### 11.4.6.4 Флаги

- S, :
- N Z ;
- C , . “ ;
- ”; V.

#### 11.4.6.5 Примеры

```

MOVSR11, #0x000B ;          0x000B R11,
MOV R1, #0xFA05 ;          0xFA05 R1,
MOVSR10, R12 ;          R12 R10,
MOV R3, #23 ;          23 R3
MOV R8, SP ;          R8
MVNS R2, #0xF ;          0xFFFFFFFF ( R8 0x0F)
;          R2,

```

## 11.4.7 MOVТ

### Синтаксис

MOVТ{cond} Rd, #imm16

:  
cond - , " "  
Rd - -  
imm16 - 0 65535.

### Описание

MOVТ 16- imm16  
- Rd[31:16]. Rd[15:0]  
MOV MOVТ 32-

### Ограничения

Rd SP PC.

### Флаги

### Примеры

MOVТ R3, #0xF123 ; 0xF123 R3,  
; APSR

#### 11.4.8 REV, REV16, REVSH и RBIT

##### Синтаксис

op{cond} Rd, Rn

:

op -

- REV - ;

- REV16 - ;

- REVSH - ,

- RBIT - 32-

cond - , " "

Rd - -

Rn - ,

##### Описание

- REV - 32- (endianness) :  
endian big-endian little-
- REV16 - 32- big-endian little-  
endian
- REVSH - :  
- 16- big-endian 32-  
little-endian;  
- 16- little-endian 32-bit 32-  
big-endian.
- RBIT - 32-

##### Ограничения

SP

PC.

##### Флаги

##### Примеры

```

REV R3, R7 ; R7, R3
REV16 R0, R0 ; 16- R0
REVSH R0, R5 ;
REVHS R3, R7 ; " (HS)
RBIT R7, R8 ; R8, R7
    
```

### 11.4.9 TST и TEQ

#### **Синтаксис**

TST{cond} Rn, Operand2

TEQ{cond} Rn, Operand2

:

cond - , “ ”.

Rn - ,

Operand2 - . “ ”.

#### **Описание**

Operand2.

Operand2. TST ANDS, Rn  
 . , Rn 0 1,  
 TST Operand2 ,  
 1, - 0.

Operand2. TEQ EORS, Rn  
 . TEQ ,  
 V C. TEQ ,  
 N

#### **Ограничения**

SP PC.

#### **Флаги**

- S, :
- N Z ;
- C “ ”;
- V.

#### **Примеры**

TST R0, #0x3F8 ; R0 0x3F8,  
 ;  
 TEQEQ R10, R9 ; R10  
 ; R9, .

## 11.5 Инструкции умножения и деления

:

Таблица 58 – Инструкции умножения и деления

Мнемокод	Краткое описание
MLA	, 32-
MLS	, 32-
MUL	, 32-
SDIV	
SMLAL	(32 x 32 + 64), 64-
SMULL	, 64-
UDIV	
UMLAL	(32 x 32 + 64), 64-
UMULL	, 64-

### 11.5.1 MUL, MLA и MLS

32- ( , )  
32- .

#### Синтаксис

MUL{S}{cond} {Rd,} Rn, Rm ;  
MLA{cond} Rd, Rn, Rm, Ra ;  
MLS{cond} Rd, Rn, Rm, Ra ;

:

S - , " "

cond - , " "

Rd - Rd ,  
Rn.

Rn, Rm - ,

Ra - ,

#### Описание

MUL 32 , Rd. Rn Rm,  
MLA Rn Rm, Rd.  
Ra, 32  
MLS Rn Rm, Rd.  
Ra, 32

#### Ограничения

- SP PC.
- MUL S:
- Rd, Rn Rm R0 R7;
- Rd Rm;
- cond.

#### Флаги

- S, :
- N Z ;
- C V.

#### Примеры

MUL R10, R2, R5 ; R10 = R2 x R5  
MLA R10, R2, R1, R5 ; R10 = (R2 x R1) + R5  
MULS R0, R2, R2 ; R0 = R2 x R2,  
MULLT R2, R3, R2 ; R2 = R3 x R2  
MLS R4, R5, R6, R7 ; R4 = R7 - (R5 x R6)



## 11.5.2 UMULL, UMLAL, SMULL и SMLAL

, 64- , 32-

### Синтаксис

оп{cond} RdLo, RdHi, Rn, Rm

:

оп -

- UMULL -

- UMLAL -

- SMULL -

- SMLAL -

Cond -

RdLo, RdHi -

UMLAL SMLAL

Rn, Rm -

### Описание

UMULL

Rn Rm,

RdHi ( 32

) RdLo ( 32 ).

UMLAL

Rn Rm,

64-

RdHi RdLo,

RdHi RdLo.

SMULL

Rn Rm,

RdHi ( 32 ) RdLo ( 32 ).

SMLAL

Rn Rm,

64-

RdHi RdLo,

RdHi RdLo.

### Ограничения

SP

PC.

RdHi RdLo

### Флаги

### Примеры

UMULL R0, R4, R5, R6 ;

(R4,R0) = R5 x R6

SMLAL R4, R5, R3, R8 ;

(R5,R4) = (R5,R4) + R3 x R8

### 11.5.3 SDIV и UDIV

#### **Синтаксис**

SDIV{cond} {Rd,} Rn, Rm

UDIV{cond} {Rd,} Rn, Rm

:

cond - , " " .

Rd - - Rd , Rn.

Rn - , .

Rm - , .

#### **Описание**

SDIV  
Rn, , Rm. ,

UDIV  
Rn, , Rm. ,  
Rn Rm,

#### **Ограничения**

SP PC.

#### **Флаги**

#### **Примеры**

SDIV R0, R2, R4 ; , R0 = R2/R4

UDIV R8, R8, R1 ; , R8 = R8/R1.



Q 0,

MSR,

Q

MRS.

### 11.6.1.3 Ограничения

SP

PC.

### 11.6.1.4 Флаги

1

Q.

Q

### 11.6.1.5 Примеры

SSAT R7, #16, R7, LSL #4 ;

;

;

;

USATNE R0, #7, R5 ;

;

;

;

16-

R7

4

R7

R5

R0/

## 11.7 Команды работы с битовыми полями

59

**Таблица 59 – Инструкции упаковки и распаковки данных**

<b>Мнемокод команд</b>	<b>Краткое описание</b>
BFC	
BFI	
SBFX	,
SXTB	
SXTH	
UBFX	,
UXTB	
UXTH	

### 11.7.1 BFC и BFI

#### Синтаксис

BFC{cond} Rd, #lsb, #width

BFI{cond} Rd, Rn, #lsb, #width

:

cond - , " ".

Rd - - .

Rm - - .

lsb - . lsb

0 31.

width - , 1

32-lsb.

#### Описание

BFC , Rd, width

, lsb. Rd

BFI width ,

Rn, 0, width , Rd,

lsb. Rd .

#### Ограничения

SP PC.

#### Флаги

#### Примеры

BFC R4, #8, #12 ; 12- , 8- 19-  
R4.

BFI R9, R2, #8, #12 ; 12- , 8- 19-  
R9.

; 12- , 0- 11- .

; R2.

## 11.7.2 SBFX и UBFX

### Синтаксис

SBFX{cond} Rd, Rn, #lsb, #width

UBFX{cond} Rd, Rn, #lsb, #width

:  
 cond - , “ ”.  
 Rd - -  
 Rn - -  
 lsb - lsb  
 width - 0 31. 1  
 32-lsb.

### Описание

SBFX - 32-  
 UBFX - 32-

### Ограничения

SP PC.

### Флаги

### Примеры

SBFX R0, R1, #20, #4 ; 4 ( 20 23) R1,  
 ; R0  
 UBFX R8, R11, #9, #10 ; 10 ( 9 18) R11,  
 ; R8.





;

,

R3.

## 11.8 Инструкции передачи управления

Таблица 60 – Инструкции передачи управления

Мнемокод команды	Краткое описание
B	
BL	
BLX	
BX	
CBNZ	
CBZ	
IT	
TBB	, -
TBH	, -

### 11.8.1 В, BL, BX и BLX

#### 11.8.1.1 Синтаксис

```

B{cond} label
BL{cond} label
BX{cond} Rm
BLX{cond} Rm
:
B - ;
BL - ;
BX - , ;
BLX - .
cond - , " ".
label - , ".

LDR, PC".
Rm - [0] , , 1, [0].
    
```

#### 11.8.1.2 Описание

• BL BLX LR (R14);

• BX BLX (usage fault), bit[0] Rm 0.

B cond label – IT- , "IT". ( 61) .W

**Таблица 61 – Диапазон адресуемых переходов для команд ветвления**

Инструкция	Диапазон адресации	
B label	-16	+16
B cond label ( IT- )	-1	+1
B cond label ( IT- )	-16	+16
BL{cond} label	-16	+16
BX{cond} Rm	,	
BLX{cond} Rm	,	

### 11.8.1.3 Ограничения

- BLX PC;
- BX BLX, [0] Rm 1, [0];
- IT-
- B cond - IT- IT-

### 11.8.1.4 Флаги

### 11.8.1.5 Примеры

```

B loopA ; loopA
BLE ng ; ng
B.W target ; target, +/- 16
BEQ target ; target
BEQ.W target ; target +/- 1
BL funC ; ( ) funC,
; LR
BX LR ;
BXNE R0 ; R0
BLX R0 ; ( ) , R0.
    
```

## 11.8.2 CBZ и CBNZ

### Синтаксис

CBZ Rn, label  
 CBNZ Rn, label

:

Rn - ,

label - ,

### Описание

CBZ      CBNZ

CBZ Rn, label

CMP Rn, #0

BEQ label

CBNZ Rn, label

CMP Rn, #0

BNE label

### Ограничения

- Rn R0 R7;
- 4 130
- ;
- IT-

### Флаги

### Примеры

CBZ R5, target ;

R5 = 0

CBNZ R0, target ;

R0 != 0.

### 11.8.3 ИТ

#### 11.8.3.1 Синтаксис

ИТ{x{y{z}}} cond

:

x

y

z

cond

ИТ- ;

ИТ- ;

ИТ- ;

ИТ- .

ИТ-

:

- T - Then.

- E - Else.

, cond ;

, cond .

ИТ- cond AL (

).

x, y z

ИТ-  
T,

#### 11.8.3.2 Описание

ИТ

ИТ, ИТ- .  
ИТ- ,  
{cond}.

ИТ,

ВКРТ

ИТ-

ИТ-

ИТ

PSR

LR.

ИТ-

ИТ-

PC.

#### 11.8.3.3 Ограничения

ИТ- :

ИТ, CBZ CBNZ, CPSI D CPSI E.

ИТ- :

- , , ИТ- PC,

ИТ- ,

ИТ-

- ADD PC, PC, Rm;
- MOV PC, Rm;
- B, BL, BX, BLX;
- LDM, LDR POP, PC;
- TBB and TBH.

- IT-
- B cond, IT-
- IT-
- IT-
- IT-

#### 11.8.3.4 Флаги

#### 11.8.3.5 Примеры

```

ITTE NE ;
ANDNE R0, R0, R1 ; ANDNE
ADDSNE R2, R2, #1 ; ADDSNE
MOVEQ R2, R3 ;

CMP R0, #9 ; R0 ( 0 15) ASCII
('0'-'9', 'A'-'F')
ITE GT ;
ADDGT R1, R0, #55 ; [R0 > 9] 0xA -> 'A'
ADDLE R1, R0, #48 ; [R0 <= 9] 0x0 -> '0'

IT GT ; IT-
ADDGT R1, R1, #1 ; R1

ITTEE EQ ;
MOVEQ R0, R1 ;
ADDEQ R2, R2, #10 ;
ANDNE R3, R3, #1 ;
BNE.W dloop ;

IT NE ;
ADD R0, R0, R1 ; IT-
    
```

## 11.8.4 ТВВ и ТВН

### 11.8.4.1 Синтаксис

TBB [Rn, Rm]

ТВН [Rn, Rm, LSL #1]

:  
 Rn - , PC, TBB ТВН. Rn  
 Rm - , LSL #1,

### 11.8.4.2 Описание

( PC TBB) ( ТВН).  
 Rn , Rm -  
 TBB  
 ТВ  
 TBB ТВН.

### 11.8.4.3 Ограничения

- Rn SP;
- Rm SP PC;
- TBB ТВН IT-

### 11.8.4.4 Флаги

#### 11.8.4.5 Примеры

ADR.W R0, BranchTable\_Byte

TBB [R0, R1] ; R1 - , R0 -

Case1

; R1 = 0

Case2

; R1 = 1

Case3

; R1 = 2

BranchTable\_Byte

DCB 0 ; Case1

DCB ((Case2-Case1)/2) ; Case2

DCB ((Case3-Case1)/2) ; Case3

TBH [PC, R1, LSL #1] ; R1 - ,

;

TBH

BranchTable\_H

DCI ((CaseA - BranchTable\_H)/2) ; CaseA

DCI ((CaseB - BranchTable\_H)/2) ; CaseB

DCI ((CaseC - BranchTable\_H)/2) ; CaseC

CaseA

; CaseA

CaseB

; CaseB

CaseC

; CaseC



## 11.9 Прочие инструкции

Cortex-M3,

**Таблица 62 – Прочие инструкции**

<b>Мнемокод</b>	<b>Краткое описание</b>
BKPT	
CPSID	,
CPSIE	,
DMB	
DSB	
ISB	
MRS	
MSR	
NOP	
SEV	
SVC	
WFE	
WFI	

### 11.9.1 CPS

#### Синтаксис

CPSeffect iflags

:  
effect - :  
- IE - 0;  
- ID - 1.  
iflags - :  
- i - PRIMASK;  
- f - FAULTMASK.

#### Описание

CPS PRIMASK  
FAULTMASK. “ Exception mask”.

#### Ограничения

- CPS ;
- CPS ; IT-

#### Флаги

#### Примеры

CPSID i;  
CPSID f;  
CPSIE i;  
CPSIE f;

## **11.9.2 DMB**

### **Синтаксис**

DMB{cond}

:

cond -

, " "

### **Описание**

DMB

,

DMB,

,

DMB

,

### **Флаги**

### **Примеры**

DMB ;

### 11.9.3 DSB

#### **Синтаксис**

DSB{cond}

:

cond -

, " "

#### **Описание**

DSB

DSB,

DSB

#### **Флаги**

#### **Примеры**

DSB ; Data Synchronization Barrier.

#### **11.9.4 ISB**

##### **Синтаксис**

ISB{cond}

:

cond -

, " "

##### **Описание**

ISB

ISB,

##### **Флаги**

##### **Примеры**

ISB ;

## 11.9.5 MRS

### Синтаксис

MRS{cond} Rd, spec\_reg

:

cond - " " .

Rd -

spec\_reg - : APSR, IPSR, EPSR, IEPSR, IAPSR,  
EAPSR, PSR, MSP, PSP, PRIMASK, BASEPRI, BASEPRI\_MAX, FAULTMASK  
CONTROL.

### Описание

MRS MSR -  
PSR, , Q.  
, ,  
, ( )  
PSR.  
MRS, - MSR.  
MRS BASEPRI\_MAX  
BASEPRI.  
MSR.

### Ограничения

- Rd SP PC.

### Флаги

### Примеры

MRS R0, PRIMASK ; PRIMASK R0.

## 11.9.6 MSR

### Синтаксис

MSR{cond} spec\_reg, Rn

:

cond - , " " .

Rn -

spec\_reg - : APSR, IPSR, EPSR, IEPSR, IAPSR,  
EAPSR, PSR, MSP, PSP, PRIMASK, BASEPRI, BASEPRI\_MAX, FAULTMASK  
CONTROL.

### Описание

MSR

APSR ( " " APSR").

EPSR

BASEPRI\_MAX

BASEPRI

- Rn BASEPRI 0;
- Rn BASEPRI.

MRS.

### Ограничения

Rn SP PC.

### Флаги

### Примеры

MSR CONTROL, R1 ; R1 CONTROL

### 11.9.7 NOP

#### **Синтаксис**

NOP{cond}

:

cond -

, " "

#### **Описание**

NOP

NOP

64-

#### **Флаги**

#### **Примеры**

NOP ;



### **11.9.8 SEV**

#### **Синтаксис**

SEV{cond}

:

cond -

, " "

#### **Описание**

SEV

1.

"

"

#### **Флаги**

#### **Примеры**

SEV ;

### 11.9.9 SVC

#### Синтаксис

SVC{cond} #imm

:

cond -

imm -

).

, " 0 255 (8-

#### Описание

SVC

SVC.

imm

#### Флаги

#### Примеры

SVC 0x32 ;

;(

;

SVC

PC

### 11.9.10 WFE

#### **Синтаксис**

WFE{cond}

:

cond -

, " "

#### **Описание**

0,

WFE

:

•

,

;

•

SEVONPEND

SCR;

1

•

, ,

;

•

( SEV)

1,

WFE

0,

"

"

#### **Флаги**

#### **Примеры**

WFE

;

### 11.9.11 WFI

#### **Синтаксис**

WFI{cond}

:

cond -

, " "

#### **Описание**

WFI

:

•

;

•

,

,

#### **Флаги**

#### **Примеры**

WFI ;

## 12 Системный таймер SysTick

24 ; ( , SysTick, )  
LOAD

### 12.1 Описание регистров системного таймера SysTick

**Таблица 63 – Описание регистров системного таймера SysTick**

Адрес	Название	Тип	Доступ	Значение после сброса	Описание
0xE000E010	SysTick				SYSTICK
0xE000E010	CTRL	RW		0x00000004	SysTick->CTRL
0xE000E014	LOAD	RW		0x00000000	SysTick->LOAD
0xE000E018	VAL	RW		0x00000000	SysTick->VAL
0xE000E01C	CALIB	RO		0x00002904 <sup>1)</sup>	SysTick->CAL

1)

### 12.1.1 SysTick->CTRL

CTRL

:

**Таблица 64 – Регистр контроля и статуса CTRL**

Номер	31...17	16	15...3	2	1	0
Доступ						
Сброс						
	-	<b>COUNTFLAG</b>	-	<b>CLKSOURCE</b>	<b>TICKINT</b>	<b>ENABLE</b>

**COUNTFLAG**

1,

**CLKSOURCE**

:

0 - LSI

1 - HCLK

**TICKINT**

:

0 -

;

1 -

COUNTFLAG,

,

**ENABLE**

:

0 -

;

1 -

.

ENABLE

,

RELOAD

LOAD

0

COUNTFLAG

TICKINT

RELOAD

**12.1.2 SysTick->LOAD**

LOAD

VAL.

**Таблица 65 – Регистр перегружаемого значения LOAD**

<b>Номер</b>	31...24	23...0
<b>Доступ</b>		
<b>Сброс</b>		
	-	<b>RELOAD</b>

**RELOAD**

VAL,

RELOAD

RELOAD

0x00000001-0x00FFFFFF.

0

COUNTFLAG

1 0.

RELOAD

:

•

N

RELOAD,

N-1.

100

RELOAD,

99;

•

N

N.

400

RELOAD,

400.

### 12.1.3 SysTick->VAL

VAL

**Таблица 66 – Регистр текущего значения таймера VAL**

<b>Номер</b>	31...24	23...0
<b>Доступ</b>		
<b>Сброс</b>		
	-	<b>CURRENT</b>

**CURRENT**

CTRL.

COUNTFLAG

### 12.1.4 SysTick->CAL

CALIB

**Таблица 67 – Регистр калибровочного значения таймера CAL**

<b>Номер</b>	31	30	29...24	23...0
<b>Доступ</b>				
<b>Сброс</b>				
	<b>NOREF</b>	<b>SKEW</b>	-	<b>TENMS</b>

**NOREF**

**SKEW**

**TENMS**

0x0002904.

1

10.5

0x0002904 (10500),  
(84/8=10.5 ).

## 12.2 Советы и особенности при применении системного таймера



## 13 Модуль защиты памяти MPU

(MPU).

MPU

- 
- 
- 

( ) ;

. Cortex-M3 MPU :

, 0-7;

- 
- 

7

7.

, default ,

Cortex-M3

MPU,

OS.

OS

MPU  
OS

MPU

MPU

68

MPU

, *shareable* ,

Таблица 68 – Обзор атрибутов памяти

Тип памяти	Атрибут <i>shareable</i>	Другие атрибуты	Описание
	-	-	
		-	
		-	

### 13.1 Описание регистров MPU

MPU

Таблица 69 – Обзор регистров MPU

Адрес	Обозначение	Тип	Доступ	Значение после сброса	Описание
0xE000ED90	MPU				MPU
0x000	TYPE	RO		0x00000800	MPU->TYPE
0x004	CTRL	RW		0x00000000	MPU->CTRL
0x008	RNR	RW		0x00000000	MPU->RNR
0x00C	RBAR	RW		0x00000000	MPU->RBAR
0x010	RASR	RW		0x00000000	MPU->RASR
0x014	RBAR_A1	RW		0x00000000	RBAR
0x018	RASR_A1	RW		0x00000000	RASR
0x01C	RBAR_A2	RW		0x00000000	RBAR
0x020	RASR_A2	RW		0x00000000	RASR
0x24	RBAR_A3	RW		0x00000000	RBAR
0x28	RASR_A3	RW		0x00000000	RASR

#### 13.1.1 MPU->TYPE

TYPE, MPU,

Таблица 70 – Регистр TYPE

Номер	31...24	23...16	15...8	7...1	0
Доступ					
Сброс					
	-	IREGION	DREGION	-	SEPARATE

#### IREGION

MPU

0x00.

MPU

DREGION.

#### DREGION

MPU

0x08 -

MPU

#### SEPARATE

0 -

### 13.1.2 MPU->CTRL

CTRL:

- MPU;
- default ;
- MPU, ,  
(NMI), FAULTMASK .

**Таблица 71 – Регистр CTRL**

Номер	31...4	3	2	1	0
Доступ					
Сброс					
	-	<b>PRIVDEFENA</b>	<b>HFNMIENA</b>	<b>ENABLE</b>	

#### **PRIVDEFENA**

default :  
 0 - MPU , default .  
 , ;  
 1 - MPU , default .  
 , -1. ,  
 , default .  
 MPU , .

#### **HFNMIENA**

MPU , NMI,  
 FAULTMASK .  
 MPU :  
 0 - MPU , NMI, FAULTMASK  
 , ENABLE;  
 1 - MPU , NMI, FAULTMASK  
 .  
 MPU , .

#### **ENABLE**

MPU:  
 0 - MPU ;  
 1 - MPU .  
 ENABLE PRIVDEFENA :  
 , default “  
 ”.  
 , default .

XN

ENABLE.

ENABLE

PRIVDEFENA

PRIVDEFENA

ENABLE

MPU

default

default

MPU

PRIVDEFENA.

HFNMIENA

1, MPU

-1

-2.

NMI,

FAULTMASK

HFNMIENA

### 13.1.3 MPU->RNR

RNR

RBAR RASR.

**Таблица 72 – Регистр номера региона RNR**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>		
<b>Сброс</b>		
	-	<b>REGION</b>

#### **REGION**

MPU

RBAR RASR.

MPU

8

0

7.

RBAR RASR.

VALID.

RBAR

REGION.

#### **MPU->RBAR**

RBAR

MPU

RNR,

RNR.

RBAR

VALID

RNR.

**Таблица 73 – Регистр базового адреса региона RBAR**

<b>Номер</b>	31...N	N-1...6	5	4	3...0
<b>Доступ</b>					
<b>Сброс</b>					
	<b>ADDR</b>	-	<b>VALID</b>	<b>REGION</b>	

**ADDR**

N  
“ ADDR”.

**VALID**

MPU:

0 - RNR :  
- , : RNR;  
- REGION.  
1 - :  
- RNR REGION;  
- REGION.

**REGION**

MPU :  
- ( . VALID);  
- RNR.

**Поле ADDR**

ADDR [31:N] RBAR. SIZE  
RASR,  
N = Log2 ( ),  
4 , RASR, ADDR  
0x00000000.  
64 , , 0x00010000 0x00020000. , 64

**13.1.4 MPU->RASR**

RASR

MPU

RNR,

RASR

:

;

**Таблица 74 – Назначение бит регистра RASR.**

Номер	31	30	29	28	27	26	24	23	22	21	19	18	17	16	15	8	7	6	5	1	0	
Доступ																						
Сброс																						
	-	<b>XN</b>	-	<b>AP</b>	-	<b>TEX</b>	<b>S</b>	<b>C</b>	<b>B</b>	<b>SRD</b>	-	<b>SIZE</b>	<b>ENABLE</b>									

**XN**

0 -

1 -

**AP**

78 -

AP.

**TEX, C, B**

76 -

**S**

75 -

SIZE.

**SRD**

0 -

1 -

128

SRD

0x00.

**SIZE**

MPU

3(b00010),

SIZE”.

**ENABLE**

MPU”.

**Значения поля SIZE**

SIZE MPU , RNR  
:

$$(\text{Region size in bytes}) = 2^{(\text{SIZE}+1)}$$

4. 75 32 , SIZE,  
N RBAR.

**Таблица 75 – Пример значений поля SIZE**

<b>Значение SIZE</b>	<b>Размер региона</b>	<b>Значение N<sup>(1)</sup></b>	<b>Комментарий</b>
b00100 (4)	32	5	
b01001 (9)	1	10	-
b10011 (19)	1	20	-
b11101 (29)	1	30	-
b11111 (31)	4	b01100	

<sup>1)</sup>. RBAR, . “ MPU ”.

### 13.1.5 Атрибуты разрешения доступа MPU

TEX, C, B, S,

AP XN

RASR

, MPU

**Таблица 76 – Кодирование бит разрешения доступа TEX, C, B, S**

TEX	C	B	S	Тип памяти	Возможность общего доступа	Другие атрибуты	
b000	0	0	x <sup>(1)</sup>				
		1	x <sup>(1)</sup>				
	1	0	0	0			
			1	1			
		1	0	0			
			1	1			
b001	0	0	0				
			1	x <sup>(1)</sup>			-
		1	0	x <sup>(1)</sup>			-
	1	1	0	0			
			1	1			
		0	x <sup>(1)</sup>	x <sup>(1)</sup>			-
b010	0	0	x <sup>(1)</sup>				
		1	x <sup>(1)</sup>			-	
	1	x <sup>(1)</sup>	x <sup>(1)</sup>			-	
b1BB	A	A	0				
			1				

<sup>1)</sup> MPU

77  
4 7.

TEX

**Таблица 77 – Кодирование режима кэша атрибутом TEX**

Значение AA или BB при TEX=1xx	Соответствующий режим кэша
00	
01	,
10	,
11	,

78

AP,

( ).



**Таблица 78 – Кодирование привилегий доступа в поле AP**

AP[2:0]	Привилегированный доступ	Непривилегированный доступ	Описание
000			
001	RW		
010	RW	RO	
011	RW	RW	
100			
101	RO		
110	RO	RO	,
111	RO	RO	,

### 13.1.6 Несоответствие MP

MPU, “ MMFSR MMFSR”.

### 13.1.7 Обновление MPU региона

MPU RNR, RBAR RASR.

RBAR RASR, 4 STM.

#### 13.1.7.1 Обновление MPU региона через отдельные регистры

```

; R1 =
; R2 = /
; R3 =
; R4 =
LDR R0,=MPU_RNR ; 0xE000ED98, MPU
STR R1, [R0, #0x0] ;
STR R4, [R0, #0x4] ;
STRH R2, [R0, #0x8] ;
STRH R3, [R0, #0xA] ;

MPU,
:
; R1 =
; R2 = /
; R3 =
; R4 =
LDR R0,=MPU_RNR ; 0xE000ED98, MPU
    
```

```
STR R1, [R0, #0x0] ;
BIC R2, R2, #1 ;
STRH R2, [R0, #0x8] ;
STR R4, [R0, #0x4] ;
STRH R3, [R0, #0xA] ;
ORR R2, #1 ;
STRH R2, [R0, #0x8] ;
```

```

• barrier :
  MPU
  MPU;
• MPU,
  MPU.
  barrier , MPU
  barrier
MPU, barrier PPB,
  DSB MPU, DSR ISB.
  ISB MPU
  (branch) MPU
  (call).
  (return),
  ISB

```

### 13.1.7.2 Обновление MPU региона через множественную запись регистров

```

; R1 =
; R2 =
; R3 =
LDR R0, =MPU_RNR ; 0xE000ED98, MPU
STR R1, [R0, #0x0] ;
STR R2, [R0, #0x4] ;
STR R3, [R0, #0x8] ;
STM :
; R1 =
; R2 =
; R3 =
LDR R0, =MPU_RNR ; 0xE000ED98, MPU
STM R0, {R1-R3} ;

```

RBAR

VALID,

```

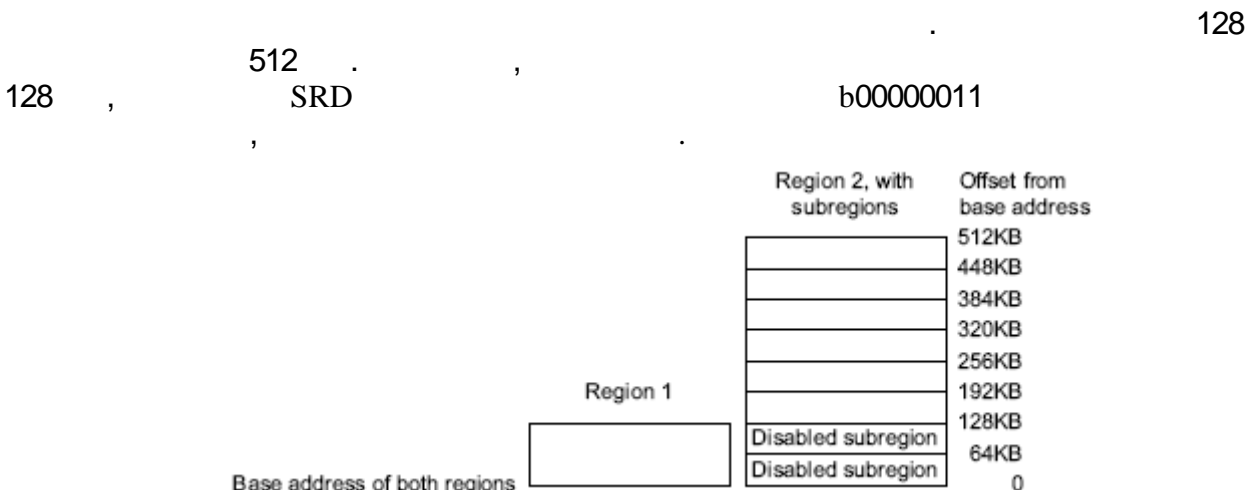
    "MPU->RBAR".
    ; R1 =
    ; R2 =
    LDR R0, =MPU_RBAR ; 0xE000ED9C,
    STR R1, [R0, #0x0] ;
    ;
    STR R2, [R0, #0x4] ;
    ;
    ; R1 =
    ; R2 =
    LDR R0, =MPU_RBAR ; 0xE000ED9C,
    STM R0, {R1-R2} ;
    ;
    
```

**13.1.7.3 Подрегионы**

```

    256
    SRD
    RASR
    "MPU->RASR".
    SRD
    MPU
    32, 64 128
    SRD 0x00,
    MPU
    
```

**13.1.7.4 Пример применения SRD**



**Рисунок 27. Применение SRD**

**13.2 Советы и особенности применения MPU**

- MPU: RASR, ;

- RASR

MPU.

MPU

### 13.2.1 Конфигурация MPU для микроконтроллера

MPU

**Таблица 79 – Атрибуты регионов памяти для микроконтроллера**

Регион памяти	TEX	C	B	S	Типа памяти и атрибут
-	b000	1	0	0	
SRAM	b000	1	0	1	
SRAM	b000	1	1	1	
	b000	0	1	1	

MPU

DMA

## 14 Сигналы тактовой частоты MDR\_RST\_CLK

HSI  
 RST\_CLK,  
 PER\_CLOCK. (UART, CAN, USB, )  
 (UART\_CLOCK, CAN\_CLOCK, USB\_CLOCK, TIM\_CLOCK)  
 CPU\_CLOCK USB\_CLOCK.

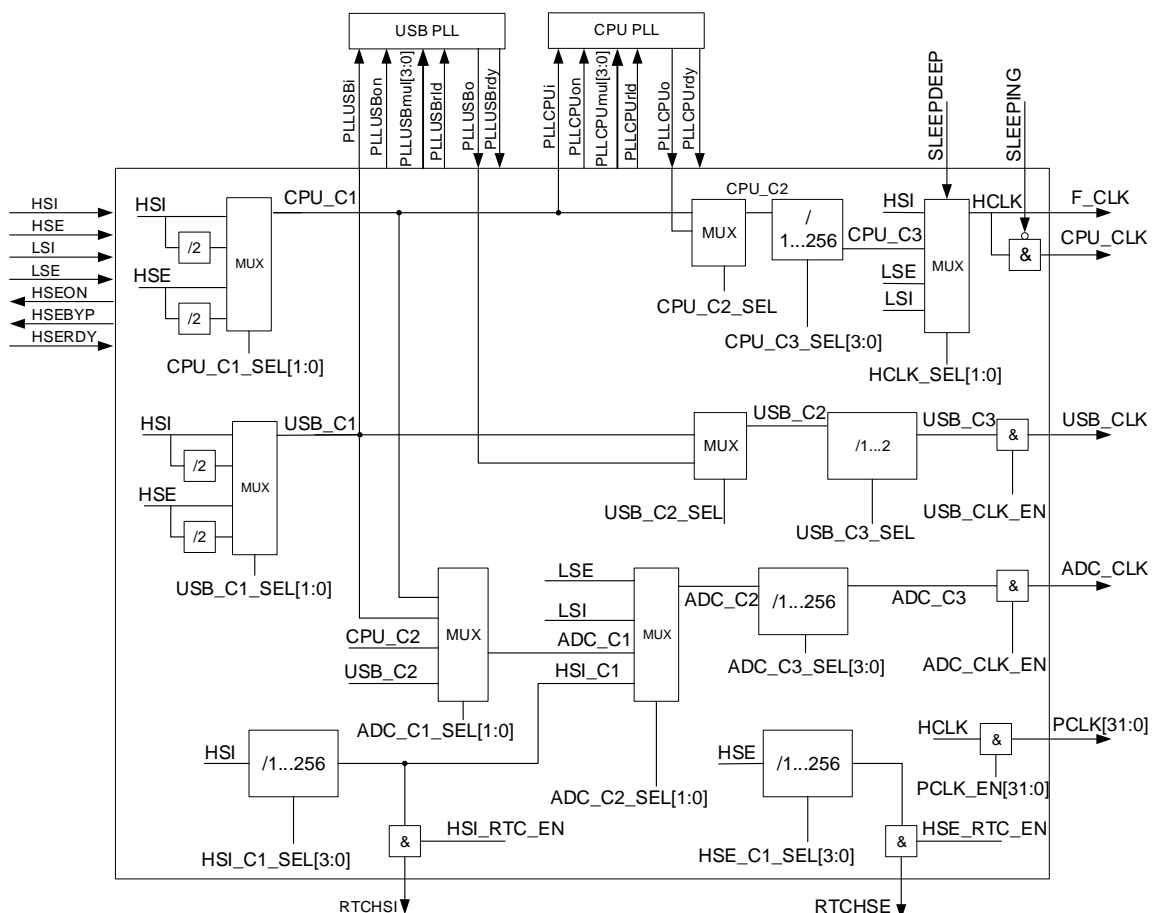


Рисунок 28. Структурная блок – схема формирования тактовой частоты

### Встроенный RC генератор HSI

HSI

$f_{O\_HSI}$

8

$U_{CC}$

HSIRDY

BKP\_REG\_0F.

HSI

HSI.  
HSION

BKP\_REG\_0F.

HSITRIM

BKP\_REG\_0F.

**Встроенный RC генератор LSI**

LSI

fo\_LSI 40

U<sub>cc</sub>

LSIRDY

BKP\_REG\_0F.

LSI

tpor.

LSI

LSION

BKP\_REG\_0F.

**Внешний генератор HSE**

HSE

2..16

U<sub>cc</sub>

HSEON

HS\_CONTROL.

HSERDY

CLOCK\_STATUS.

HSEBYP,

OSC\_IN

HSE.

OSC\_OUT

**Внешний генератор LSE**

LSE

32

BDU<sub>cc</sub>

LSEON

BKP\_REG\_0F.

LSERDY

BKP\_REG\_0F.

LSEBYP,

OSC\_IN32

LSE.

OSC\_OUT32

BKP\_REG\_0F

LSE

BDU<sub>cc</sub>

LSE

U<sub>cc</sub>.

**Встроенный блок умножения системной тактовой частоты**

2 16,

PLLCPUMUL[3:0]

2...16

PLL\_CONTROL.

100

PLLCPURDY

CLOCK\_STATUS.

PLLCPUON

PLL\_CONTROL.

**Встроенный блок умножения USB тактовой частоты**

2 16,

PLLUSBMUL[3:0]

2...16

PLL\_CONTROL.

48

PLLUSBRDY

CLOCK\_STATUS.

PLLUSBON

PLL\_CONTROL.

USB

## 14.1 Описание регистров блока контроллера тактовой частоты

**Таблица 80 – Описание регистров блока контроллера тактовой частоты**

<b>Базовый Адрес</b>	<b>Название</b>	<b>Описание</b>
0x4002_0000	MDR_RST_CLK	
<b>Смещение</b>		
0x00	CLOCK_STATUS	MDR_RST_CLK->CLOCK_STATUS
0x04	PLL_CONTROL	MDR_RST_CLK->PLL_CONTROL
0x08	HS_CONTROL	MDR_RST_CLK->HS_CONTROL
0x0C	CPU_CLOCK	MDR_RST_CLK->CPU_CLOCK
0x10	USB_CLOCK	MDR_RST_CLK->USB_CLOCK USB
0x14	ADC_MCO_CLOCK	MDR_RST_CLK->ADC_MCO_CLOCK
0x18	RTC_CLOCK	MDR_RST_CLK->RTC_CLOCK RTC
0x1C	PER_CLOCK	MDR_RST_CLK->PER_CLOCK
0x20	CAN_CLOCK	MDR_RST_CLK->CAN_CLOCK CAN
0x24	TIM_CLOCK	MDR_RST_CLK->TIM_CLOCK TIMER
0x28	UART_CLOCK	MDR_RST_CLK->UART_CLOCK UART
0x2C	SSP_CLOCK	MDR_RST_CLK->SSP_CLOCK SSP

**14.1.1 MDR\_RST\_CLK->CLOCK\_STATUS**

**Таблица 81 – Регистр CLOCK\_STATUS**

<b>Номер</b>	31...3	2	1	0
<b>Доступ</b>	U	RO	RO	RO
<b>Сброс</b>	0	0	0	0
	-	<b>HSE RDY</b>	<b>PLL CPU RDY</b>	<b>PLL USB RDY</b>

**Таблица 82 – Описание бит регистра CLOCK\_STATUS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...3	-	
2	HSE RDY	HSE: 0 – ; 1 –
1	PLL CPU RDY	CPU PLL: 0 – PLL ; 1 – PLL
0	PLL USB RDY	USB PLL: 0 – PLL ; 1 – PLL



**14.1.2 MDR\_RST\_CLK->PLL\_CONTROL**

**Таблица 83 – Регистр PLL\_CONTROL**

<b>Номер</b>	31...12	11...8	7...4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	-	<b>PLL CPU MUL[3:0]</b>	<b>PLL USB MUL[3:0]</b>	<b>PLL CPU RLD</b>	<b>PLL CPU ON</b>	<b>PLL USB RLD</b>	<b>PLL USB ON</b>

**Таблица 84 – Описание бит регистра PLL\_CONTROL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...12	-	
11...8	PLL CPU MUL[3:0]	CPU PLL: $PLL_{CPUo} = PLL_{CPUi} \times (PLL_{CPUMUL} + 1)$
7...4	PLL USB MUL[3:0]	USB PLL: $PLL_{USB0} = PLL_{USBi} \times (PLL_{USBMUL} + 1)$
3	PLL CPU RLD	PLL.  1
2	PLL CPU ON	PLL: 0 – PLL ; 1 – PLL
1	PLL USB RLD	PLL.  1
0	PLL USB ON	PLL: 0 – PLL ; 1 – PLL

**14.1.3 MDR\_RST\_CLK->HS\_CONTROL**

**Таблица 85 – Регистра HS\_CONTROL**

<b>Номер</b>	31...2	1	0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	<b>HSE BYP</b>	<b>HSE ON</b>

**Таблица 86 – Описание бит регистра HS\_CONTROL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	-	
1	HSE BYP	HSE : 0 – ; 1 –
0	HSE ON	HSE : 0 – ; 1 –

**14.1.4 MDR\_RST\_CLK->CPU\_CLOCK**

**Таблица 87 – Регистр CPU\_CLOCK**

<b>Номер</b>	31...10	9...8	7...4	3	2	1...0
<b>Доступ</b>	U	R/W	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>HCLK SEL[1:0]</b>	<b>CPU C3 SEL[3:0]</b>	-	<b>CPU C2 SEL</b>	<b>CPU C1 SEL[1:0]</b>

**Таблица 88 – Описание бит регистра CPU\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...10	-	
9...8	HCLK SEL[1:0]	HCLK: 00 – HSI 01 – CPU_C3 10 – LSE 11 – LSI
7...4	CPU C3 SEL[3:0]	CPU_C3: 0xxx – CPU_C3 = CPU_C2 1000 - CPU_C3 = CPU_C2 / 2 1001 - CPU_C3 = CPU_C2 / 4 1010 - CPU_C3 = CPU_C2 / 8 ... 1111 - CPU_C3 = CPU_C2 / 256
3	-	
2	CPU C2 SEL	CPU_C2: 0 – CPU_C1 1 – PLLCPUo
1...0	CPU C1 SEL[1:0]	CPU_C1: 00 – HSI 01 – HSI/2 10 – HSE 11 – HSE/2

**14.1.5 MDR\_RST\_CLK->USB\_CLOCK**

**Таблица 89 – Регистр USB\_CLOCK**

<b>Номер</b>	31...9	8	7...5	4	3	2	1...0
<b>Доступ</b>	U	R/W	U	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	-	<b>USB CLK EN</b>	-	<b>USB C3 SEL</b>	-	<b>USB C2 SEL</b>	<b>USB C1 SEL[1:0]</b>

**Таблица 90 – Описание бит регистра USB\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...9	-	
8	USB CLK EN	USB: 0 – ; 1 –
7...5	-	
4	USB C3 SEL	USB_C3: 0 – USB_C3 = USB_C2 1 – USB_C3 = USB_C2 / 2
3	-	
2	USB C2 SEL	USB_C2: 0 – USB_C1 1 – PLLUSB <sub>o</sub>
1...0	USB C1 SEL[1:0]	USB_C1: 00 – HSI 01 – HSI/2 10 – HSE 11 – HSE/2

**14.1.6 MDR\_RST\_CLK->ADC\_MCO\_CLOCK**

**Таблица 91 – Регистр ADC\_MCO\_CLOCK**

<b>Номер</b>	31...14	13	12	11...8	7...6	5...4	3...2	1...0
<b>Доступ</b>	U	R/W	U	R/W	U	R/W	U	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0
	-	ADC CLK EN	-	ADC C3 SEL[3:0]	-	ADC C2 SEL[1:0]	-	ADC C1 SEL[1:0]

**Таблица 92 – Описание бит регистра ADC\_MCO\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...14	-	
13	ADC CLK EN	ADC CLK: 0 – ; 1 –
12	-	
11...8	ADC C3 SEL[3:0]	ADC_C3: 0xxx – ADC_C3 = ADC_C2 1000 - ADC_C3 = ADC_C2 / 2 1001 - ADC_C3 = ADC_C2 / 4 1010 - ADC_C3 = ADC_C2 / 8 ... 1111 - ADC_C3 = ADC_C2 / 256
7...6	-	
5...4	ADC C2 SEL[1:0]	ADC_C1: 00 – LSE 01 – LSI 10 – ADC_C1 11 – HSI_C1
3...2	-	
1...0	ADC C1 SEL[1:0]	ADC_C1: 00 – CPU_C1 01 – USB_C1 10 – CPU_C2 11 – USB_C2

**14.1.7 MDR\_RST\_CLK->RTC\_CLOCK**

**Таблица 93 – Регистр RTC\_CLOCK**

<b>Номер</b>	31...10	9	8	7...4	3...0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	-	<b>HSI RTC EN</b>	<b>HSE RTC EN</b>	<b>HSI_C1 SEL[1:0]</b>	<b>HSE_C1 SEL[1:0]</b>

**Таблица 94 – Описание бит регистра RTC\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...10	-	
9	HSI RTC EN	HSI RTC: 0 – ; 1 –
8	HSE RTC EN	HSE RTC: 0 – ; 1 –
7...4	HSI_C1 SEL[3:0]	HSI_C1: 0xxx – RTCHSI = HSI_C2 1000 - RTCHSI = HSI_C2 / 2 1001 - RTCHSI = HSI_C2 / 4 1010 - RTCHSI = HSI_C2 / 8 ... 1111 - RTCHSI = HSI_C2 / 256
3...0	HSE_C1 SEL[3:0]	HSE_C1: 0xxx – RTCHSE = HSE_C2 1000 - RTCHSE = HSE_C2 / 2 1001 - RTCHSE = HSE_C2 / 4 1010 - RTCHSE = HSE_C2 / 8 ... 1111 - RTCHSE = HSE_C2 / 256

**14.1.8 MDR\_RST\_CLK->PER\_CLOCK**

**Таблица 95 – Регистр PER\_CLOCK**

<b>Номер</b>	31...5	4	3...0
<b>Доступ</b>	R/W	R/W	R/W
<b>Сброс</b>	0	1	0
	<b>PCLK_EN[31:5]</b>	<b>PCLK_EN[4]</b>	<b>PCLK_EN[3:0]</b>

**Таблица 96 – Описание бит регистра PER\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	PCLK EN[31:5]	: 0 – ; 1 – . PCLK[5] – DMA PCLK[6] – UART1 PCLK[7] – UART2 PCLK[8] – SPI1 PCLK[9] – PCLK[10] – I2C1 PCLK[11] – POWER PCLK[12] – WWDT PCLK[13] – IWDT PCLK[14] – TIMER1 PCLK[15] – TIMER2 PCLK[16] – TIMER3 PCLK[17] – ADC PCLK[18] – DAC PCLK[19] – COMP PCLK[20] – SPI2 PCLK[21] – PORTA PCLK[22] – PORTB PCLK[23] – PORTC PCLK[24] – PORTD PCLK[25] – PORTE PCLK[26] – PCLK[27] – BKP PCLK[28] – PCLK[29] – PORTF PCLK[30] – EXT_BUS_CNTRL PCLK[31] –
4	PCLK EN[4]	: 0 – ; 1 – . PCLK[4] – RST_CLK. 1
3...0	PCLK EN[3:0]	: 0 – ; 1 – . PCLK[0] – CAN1 PCLK[1] – CAN2 PCLK[2] – USB PCLK[3] – EEPROM_CNTRL

**14.1.9 MDR\_RST\_CLK->CAN\_CLOCK**

**Таблица 97 – Регистр CAN\_CLOCK**

<b>Номер</b>	31...26	25	24	23...16	15...8	7...0
<b>Доступ</b>	U	R/W	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>CAN2 CLK EN</b>	<b>CAN1 CLK EN</b>	-	<b>CAN2 BRG [7:0]</b>	<b>CAN1 BRG [7:0]</b>

**Таблица 98 – Описание бит регистра CAN\_CLOCK**

	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...26	-	
25	CAN2 CLK EN	CAN2: 0 – ; 1 –
24	CAN1 CLK EN	CAN2: 0 – ; 1 –
23..16	-	
15...8	CAN2 BRG [7:0]	CAN2  xxxxx000 – CAN2_CLK == HCLK xxxxx001 – CAN2_CLK == HCLK/2 xxxxx010 – CAN2_CLK == HCLK/4 ... xxxxx111 – CAN2_CLK == HCLK/128
7...0	CAN1 BRG [7:0]	CAN1  xxxxx000 – CAN1_CLK == HCLK xxxxx001 – CAN1_CLK == HCLK/2 xxxxx010 – CAN1_CLK == HCLK/4 ... xxxxx111 – CAN1_CLK == HCLK/128



**14.1.10 MDR\_RST\_CLK->TIM\_CLOCK**

**Таблица 99 – Регистр TIM\_CLOCK**

<b>Номер</b>	31...27	26	25	24	23...16	15...8	7...0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	-	<b>TIM3 CLK EN</b>	<b>TIM2 CLK EN</b>	<b>TIM1 CLK EN</b>	<b>TIM3 BRG [7:0]</b>	<b>TIM2 BRG [7:0]</b>	<b>TIM1 BRG [7:0]</b>

**Таблица 100 – Описание бит регистра TIM\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...27	-	
26	TIM3 CLK EN	TIM3: 0 – ; 1 –
25	TIM2 CLK EN	TIM2: 0 – ; 1 –
24	TIM1 CLK EN	TIM1: 0 – ; 1 –
23..16	TIM3 BRG [7:0]	TIM3: xxxxx000 – TIM3_CLK == HCLK xxxxx001 – TIM3_CLK == HCLK/2 xxxxx010 – TIM3_CLK == HCLK/4 ... xxxxx111 – TIM3_CLK == HCLK/128
15...8	TIM2 BRG [7:0]	TIM2: xxxxx000 – TIM2_CLK == HCLK xxxxx001 – TIM2_CLK == HCLK/2 xxxxx010 – TIM2_CLK == HCLK/4 ... xxxxx111 – TIM2_CLK == HCLK/128
7...0	TIM1 BRG [7:0]	TIM1: xxxxx000 – TIM1_CLK == HCLK xxxxx001 – TIM1_CLK == HCLK/2 xxxxx010 – TIM1_CLK == HCLK/4 ... xxxxx111 – TIM1_CLK == HCLK/128

**14.1.11 MDR\_RST\_CLK->UART\_CLOCK**

**Таблица 101 – Регистр UART\_CLOCK**

<b>Номер</b>	31...26	25	24	23...16	15...0	7...0
<b>Доступ</b>	U	R/W	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>UART2 CLK EN</b>	<b>UART 1 CLK EN</b>	-	<b>UART 2 BRG [7:0]</b>	<b>UART 1 BRG [7:0]</b>

**Таблица 102 – Описание бит регистра UART\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...27	-	
26	-	
25	UART2 CLK EN	UART2: 0 – ; 1 –
24	UART1 CLK EN	UART 1: 0 – ; 1 –
23..16	-	
15...8	UART2 BRG [7:0]	UART 2: xxxxx000 – UART 2_CLK == HCLK xxxxx001 – UART 2_CLK == HCLK/2 xxxxx010 – UART 2_CLK == HCLK/4 ... xxxxx111 – UART 2_CLK == HCLK/128
7...0	UART1 BRG [7:0]	UART1: xxxxx000 – UART 1_CLK == HCLK xxxxx001 – UART 1_CLK == HCLK/2 xxxxx010 – UART 1_CLK == HCLK/4 ... xxxxx111 – UART 1_CLK == HCLK/128

**14.1.12 MDR\_RST\_CLK->SSP\_CLOCK**

**Таблица 103 – Регистр SSP\_CLOCK**

<b>Номер</b>	31...26	25	24	23...16	15...8	7...0
<b>Доступ</b>	U	R/W	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>SSP2 CLK EN</b>	<b>SSP 1 CLK EN</b>	-	<b>SSP 2 BRG [7:0]</b>	<b>SSP 1 BRG [7:0]</b>

**Таблица 104 – Описание бит регистра SSP\_CLOCK**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...27	-	
26	-	
25	SSP2 CLK EN	SSP 2: 0 – ; 1 –
24	SSP1 CLK EN	SSP 1: 0 – ; 1 –
23...16	-	
15...8	SSP2 BRG [7:0]	SSP 2: xxxxx000 – SSP 2_CLK == HCLK xxxxx001 – SSP 2_CLK == HCLK/2 xxxxx010 – SSP 2_CLK == HCLK/4 ... xxxxx111 – SSP 2_CLK == HCLK/128
7...0	SSP1 BRG [7:0]	SSP1: xxxxx000 – SSP 1_CLK == HCLK xxxxx001 – SSP 1_CLK == HCLK/2 xxxxx010 – SSP 1_CLK == HCLK/4 ... xxxxx111 – SSP 1_CLK == HCLK/128

## 15 Батарейный домен и часы реального времени MDR\_VKP

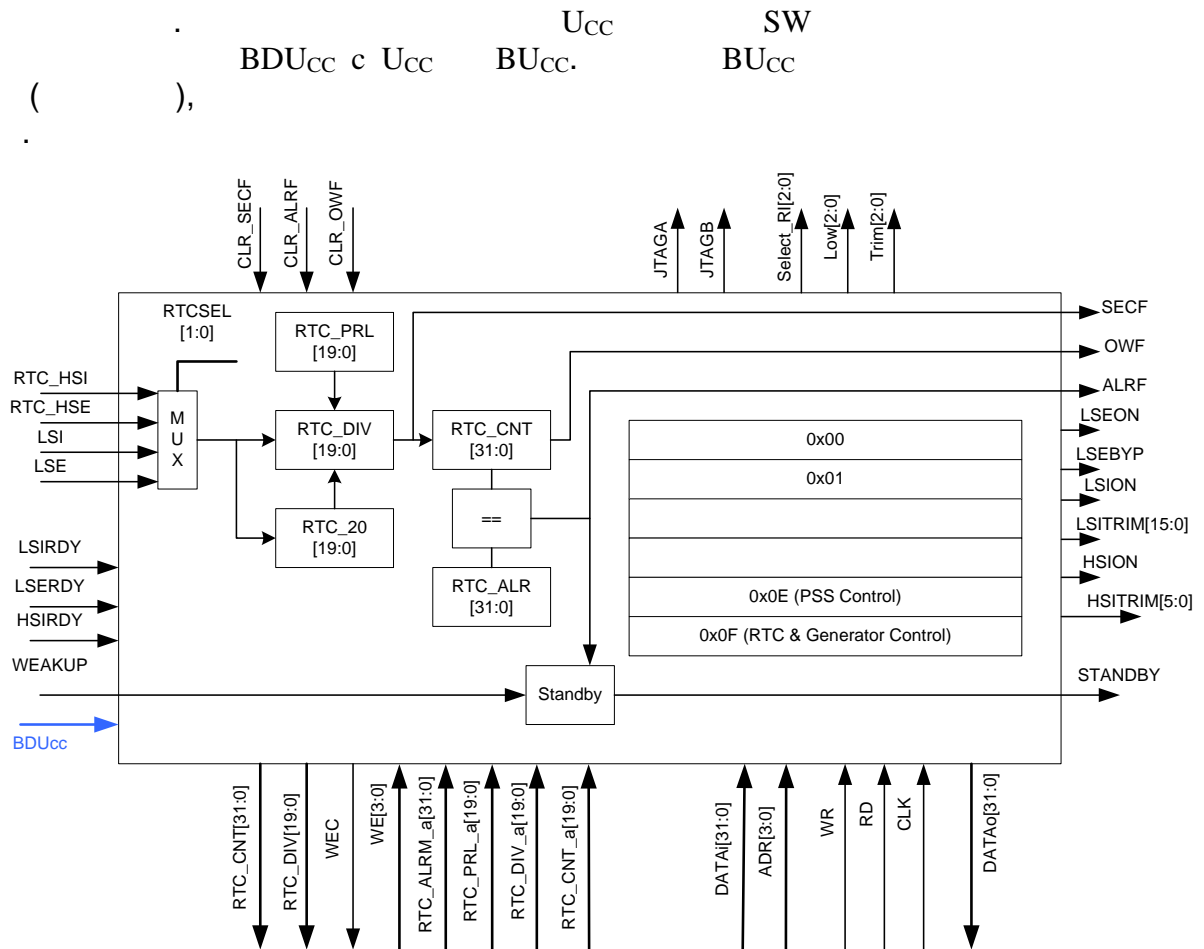
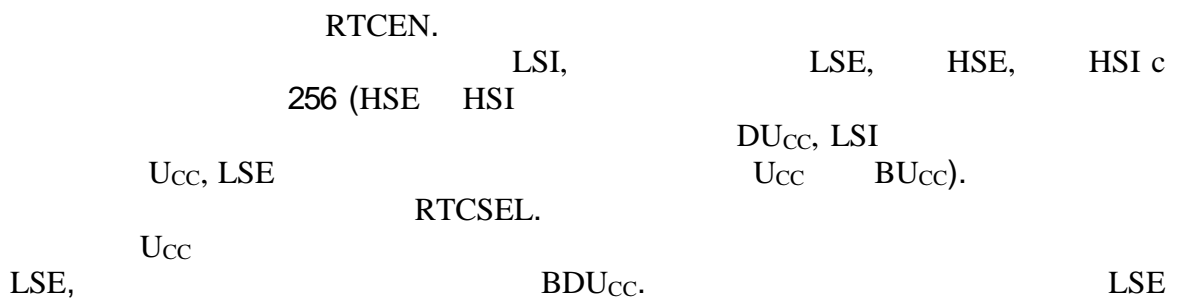


Рисунок 29. Структурная блок-схема батарейного домена и часов реального времени

### 15.1 Часы реального времени



CAL[6:0]. CAL , RTC\_20  
 CAL[6:0] 2<sup>20</sup>  
 CAL  
 RTC\_DIV 20-  
 RTC\_DIV 1  
 RTC\_PRL.  
 RTC\_CNT  
 RTC\_DIV. RTC\_ALR  
 STANDBY,  
 RTC\_CNT RTC\_ALR.  
 STANDBY WAKEUP.

## 15.2 Регистры аварийного сохранения

16 32-  
 16- 15-  
 14

## 15.3 Описание регистров блока батарейного домена

**Таблица 105 – Описание регистров блока батарейного домена**

Базовый Адрес	Название	Описание
0x400D_8000	MDR_BKP	
<b>Смещение</b>		
0x00	REG_00	MDR_BKP->REG_[0D...00] MDR_BKP->REG_00 0
...		
0x38	REG_0E	MDR_BKP->REG_0E 14
0x3C	REG_0F	MDR_BKP->REG_0F 15 RTC, LSE, LSI HSI
0x40	RTC_CNT	MDR_BKP->RTC_CNT
0x44	RTC_DIV	MDR_BKP->RTC_DIV
0x48	RTC_PRL	MDR_BKP->RTC_PRL
0x4C	RTC_ALRM	MDR_BKP->RTC_ALRM ALRF
0x50	RTC_CS	MDR_BKP->RTC_CS

### 15.3.1 MDR\_BKP->REG\_[0D...00]

MDR\_BKP->REG\_00  
 MDR\_BKP->REG\_01  
 MDR\_BKP->REG\_02  
 MDR\_BKP->REG\_03  
 MDR\_BKP->REG\_04  
 MDR\_BKP->REG\_05  
 MDR\_BKP->REG\_06  
 MDR\_BKP->REG\_07  
 MDR\_BKP->REG\_08  
 MDR\_BKP->REG\_09  
 MDR\_BKP->REG\_0A  
 MDR\_BKP->REG\_0B  
 MDR\_BKP->REG\_0C  
 MDR\_BKP->REG\_0D

**Таблица 106 – Регистры REG\_[0D...00]**

<b>Номер</b>	31...0
<b>Доступ</b>	R/W
<b>Сброс</b>	U
	<b>ВКР REG[31:0]</b>

**Таблица 107 – Описание бит регистров REG\_[0D...00]**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	ВКР REG[31:0]	

### 15.3.2 MDR\_BKP->REG\_0E

**Таблица 108 – Регистр REG\_0E**

<b>Номер</b>	31...15	14...12	11	10...8	7	6	5...3	2...0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	U	1	0	U	U	0	0

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

-	MODE [2:0]	FPOR	Trim [2:0]	JTAG_B	JTAG_A	SelectRI [2:0]	LOW [2:0]
---	---------------	------	---------------	--------	--------	-------------------	--------------

Таблица 109 – Описание бит регистра REG\_0E

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...15	-	
14...12	MODE[2..0]	<p align="center">MODE[2:0] (PF[6:4]):</p> <p>000 – JTAG_B</p> <p>001 – JTAG_A</p> <p>010 – JTAG_B</p> <p>011 –</p> <p>100 –</p> <p>101 – UART</p> <p>110 – UART</p> <p>111 – ( )</p> <p align="right">RESET, MODE[2:0]</p>
11	FPOR	<p align="center">POR.</p> <p align="center">1</p> <p align="center">U<sub>cc</sub>,</p> <p align="center">0.</p>
10...8	Trim[2:0]	<p align="center">DU<sub>cc</sub>.</p> <p align="center">DU<sub>cc</sub>:</p> <p>000 – DU<sub>cc</sub> + 0,10</p> <p>001 – DU<sub>cc</sub> + 0,06</p> <p>010 – DU<sub>cc</sub> + 0,04</p> <p>011 – DU<sub>cc</sub> + 0,01</p> <p>100 – DU<sub>cc</sub> – 0,01</p> <p>101 – DU<sub>cc</sub> – 0,04</p> <p>110 – DU<sub>cc</sub> – 0,06</p> <p>111 – DU<sub>cc</sub> – 0,10</p> <p align="right">Trim</p>
7	JTAG B	<p align="center">JTAG B:</p> <p>0 – ;</p> <p>1 – .</p> <p align="center">JTAG B JTAG A</p>
6	JTAG A	<p align="center">JTAG A:</p> <p>0 – ;</p> <p>1 –</p>

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

5...3	SelectRI[2:0]	<p align="right">DUcc:</p> <p>000 – ~ 6 ( 300 )</p> <p>001 – ~ 270 ( 6,6 )</p> <p>010 – ~ 90 ( 20 )</p> <p>011 – ~ 24 ( 80 )</p> <p>100 – ~ 900 ( 2 )</p> <p>101 – ~ 2 ( 900 )</p> <p>110 – ~ 400 ( 4,4 )</p> <p>111 – ~ 100 ( 19 )</p>
2...0	LOW[2:0]	<p align="right">DUcc.</p> <p align="center">LOW SelectRI :</p> <p>000 – 10</p> <p>001 – 200</p> <p>010 – 500</p> <p>011 – 1</p> <p>100 –</p> <p>101 – 40</p> <p>110 – 80</p> <p>111 – 80</p>

**15.3.3 MDR\_BKP->REG\_0F**

**Таблица 110 – Регистр REG\_0F**

<b>Номер</b>	15	14	13	12...5	4	3...2	1	0
<b>Доступ</b>	R/W	U	RO	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	1	0	0	0	0	0	0	0
	<b>LSI ON</b>	-	<b>LSE RDY</b>	<b>CAL [7:0]</b>	<b>RTC EN</b>	<b>RTC SEL[1:0]</b>	<b>LSE BYP</b>	<b>LSE ON</b>

<b>Номер</b>	31	30	29...24	23	22	21	20...16
<b>Доступ</b>	R/W	R/W	R/W	RO	R/W	RO	R/W
<b>Сброс</b>	0	0	0	1	1	1	0
	<b>RTC RESET</b>	<b>STANDBY</b>	<b>HSI TRIM [5:0]</b>	<b>HSI RDY</b>	<b>HSI ON</b>	<b>LSI RDY</b>	<b>LSI TRIM [4:0]</b>

**Таблица 111 – Описание бит регистра REG\_0F**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31	RTC RESET	: 0 – ; 1 –
30	STANDBY	DUcc: 0 – ; 1 –  ALRF WAKEUP
29...24	HSI TRIM[5:0]	HSI. ( 31)



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

23	HSI RDY	0 – 1 –	HSI : ;
22	HSI ON	0 – 1 –	HSI: ;
21	LSI RDY	0 – 1 –	LSI : ;
20...16	LSI TRIM[4:0]		LSI. ( 30)
15	LSI ON	0 – 1 –	LSI: ;
14	-		
13	LSE RDY	0 – 1 –	LSE : ;
12...5	CAL[7:0]		$2^{20}$ CAL : 00000000 – 0 00000001 – 1 .... 11111111 – 256 , CAL = 0 = 32768.00000 CAL = 1 = 32767,96875 ; ... CAL = 255 = 32760,03125
4	RTC EN	0 – 1 –	;
3...2	RTC SEL[1:0]		: 00 – LSI 01 – LSE 10 – HSIRTC ( CLKRST) 11 – HSERTC ( CLKRST)
1	LSE BYP	0 – 1 –	LSE: ; ( )
0	LSE ON	0 – 1 –	LSE: ;

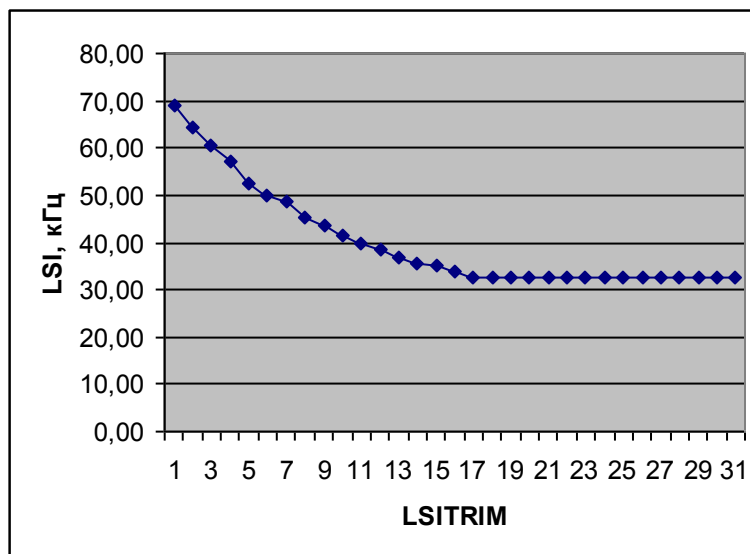


Рисунок 30. Зависимость частоты LSI от значения LSITRIM

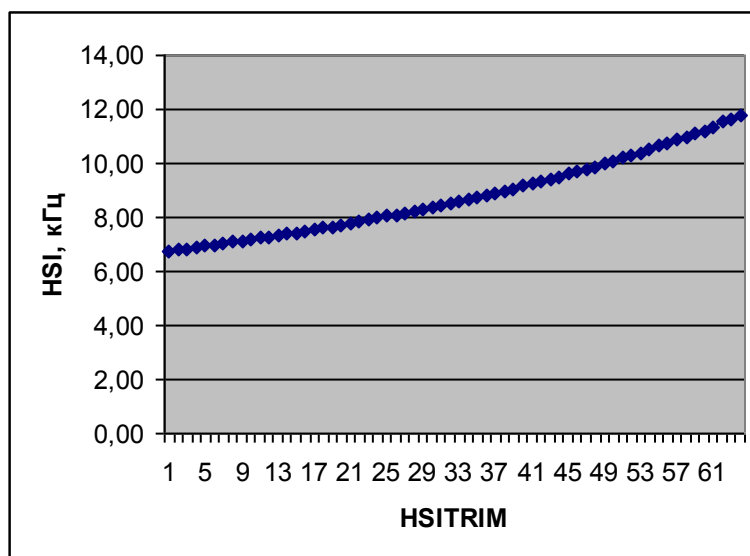


Рисунок 31. Зависимость частоты HSI от значения HSITRIM

### MDR\_BKP->RTC\_CNT

Таблица 112 – Регистр RTC\_CNT

Номер	31...0
Доступ	R/W
Сброс	0
	<b>RTC CNT[31:0]</b>

Таблица 113 – Описание бит регистра RTC\_CNT

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	RTC CNT[31:0]	

### 15.3.4 MDR\_BKP->RTC\_DIV

Таблица 114 – Регистр RTC\_DIV

Номер	31...20	19...0
Доступ	U	R/W
Сброс	0	0
	-	<b>RTC DIV[19:0]</b>

Таблица 115 – Описание бит регистра RTC\_DIV

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...20	-	
19...0	RTC DIV[19:0]	

### 15.3.5 MDR\_BKP->RTC\_PRL

Таблица 116 – Регистр RTC\_PRL

Номер	31...20	19...0
Доступ	U	R/W
Сброс	0	0
	-	<b>RTC PRL[19:0]</b>

Таблица 117 – Описание бит регистра RTC\_PRL

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...20	-	
19...0	RTC PRL[19:0]	

### 15.3.6 MDR\_BKP->RTC\_ALARM

Таблица 118 – Регистр RTC\_ALARM

Номер 31...0  
 Доступ R/W  
 Сброс 0

RTC ALRM[31:0]
-------------------

Таблица 119 – Описание бит регистра RTC\_ALARM

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	RTC ALRM[31:0]	ALRF

### 15.3.7 MDR\_BKP->RTC\_CS

Таблица 120 – Регистр RTC\_CS

Номер	31...7	6	5	4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0	0
	-	WEC	ALRF_IE	SECF_IE	OWF_IE	ALRF	SECF	OWF

Таблица 121 – Описание бит регистра RTC\_PRL

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
30...7	-	
6	WEC	: 0 – RTC; 1 – , RTC
5	ALRF_IE	RTC_ALARM: 0 – ; 1 –
4	SECF_IE	: 0 – ; 1 –
3	OWF_IE	RTC_CNT: 0 – 1 –
2	ALRF	RTC_ALARM: 0 – ; 1 – . 1
1	SECF	: 0 – ; 1 – . 1
0	OWF	RTC_CNT: 0 – ;

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

---

		1 –	.	1
--	--	-----	---	---

## 16 Порты ввода-вывода MDR\_PORTx

6 / 16-

RXTX OE

JTAG.

1

Таблица 122 – Порты ввода-вывода

Вывод	Аналоговая функция ANALOG_EN=0	Цифровая функция			
		Порт IO MODE[1:0]=00 ANALOG_EN=1	Основная MODE[1:0]=01 ANALOG_EN=1	Альтернативная MODE[1:0]=10 ANALOG_EN=1	Переопределенная MODE[1:0]=11 ANALOG_EN=1
<b>Порт А</b>					
PA0	-	PA0	DATA0	<sup>1)</sup> EXT_INT1	<sup>9)</sup> -
PA1	-	PA1	DATA1	TMR1_CH1	TMR2_CH1
PA2	-	PA2	DATA2	TMR1_CH1N	TMR2_CH1N
PA3	-	PA3	DATA3	TMR1_CH2	TMR2_CH2
PA4	-	PA4	DATA4	TMR1_CH2N	TMR2_CH2N
PA5	-	PA5	DATA5	TMR1_CH3	TMR2_CH3
PA6	-	PA6	DATA6	CAN1_TX	<sup>2)</sup> UART1_RXD
PA7	-	PA7	DATA7	CAN1_RX	UART1_TXD
PA8	-	PA8	DATA8	TMR1_CH3N	TMR2_CH3N
PA9	-	PA9	DATA9	TMR1_CH4	TMR2_CH4
PA10	-	PA10	DATA10	nUART1DTR	<sup>10)</sup> TMR2_CH4N
PA11	-	PA11	DATA11	nUART1RTS	TMR2_BLK
PA12	-	PA12	DATA12	nUART1RI	TMR2_ETR
PA13	-	PA13	DATA13	nUART1DCD	TMR1_CH4N
PA14	-	PA14	DATA14	nUART1DSR	TMR1_BLK
PA15	-	PA15	DATA15	nUART1CTS	TMR1_ETR
<b>Порт В</b>					
PB0	-	PB0 JA_TDO	DATA16	<sup>1)</sup> TMR3_CH1	UART1_TXD
PB1	-	PB1 JA_TMS	DATA17	TMR3_CH1N	UART2_RXD
PB2	-	PB2 JA_TCK	DATA18	TMR3_CH2	CAN1_TX
PB3	-	PB3 JA_TDI	DATA19	TMR3_CH2N	CAN1_RX
PB4	-	PB4 JA_TRST	DATA20	TMR3_BLK	TMR3_ETR
PB5	-	PB5	DATA21	UART1_TXD	<sup>10)</sup> TMR3_CH3
PB6	-	PB6	DATA22	UART1_RXD	TMR3_CH3N
PB7	-	PB7	DATA23	nSIROUT1	TMR3_CH4
PB8	-	PB8	DATA24	COMP_OUT	<sup>7)</sup> TMR3_CH4N
PB9	-	PB9	DATA25	nSIRIN1	<sup>10)</sup> EXT_INT4
PB10	-	PB10	DATA26	EXT_INT2	<sup>9)</sup> nSIROUT1
PB11	-	PB11	DATA27	EXT_INT1	COMP_OUT
PB12	-	PB12	DATA28	SSP1_FSS	SSP2_FSS
PB13	-	PB13	DATA29	SSP1_CLK	SSP2_CLK
PB14	-	PB14	DATA30	SSP1_RXD	SSP2_RXD
PB15	-	PB15	DATA31	SSP1_TXD	SSP2_TXD

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Попр С										
P 0	-		P 0	READY <sup>17)</sup>	1)	SCL1	11)	SSP2_FSS		
P 1	-		P 1	OE		SDA1		SSP2_CLK		
P 2	-		P 2	WE		TMR3_CH1	12)	SSP2_RXD		
P 3	-		P 3	BE0		TMR3_CH1N		SSP2_TXD		
P 4	-		P 4	BE1		TMR3_CH2		TMR1_CH1		
P 5	-		P 5	BE2		TMR3_CH2N		TMR1_CH1N		
P 6	-		P 6	BE3		TMR3_CH3		TMR1_CH2		
P 7	-		P 7	CLOCK		TMR3_CH3N		TMR1_CH2N		
P 8	-		P 8	CAN1_TX	2)	TMR3_CH4		TMR1_CH3		
P 9	-		P 9	CAN1_RX		TMR3_CH4N		TMR1_CH3N		
P 10	-		P 10	-		TMR3_ETR		TMR1_CH4		
P 11	-		P 11	-		TMR3_BLK		TMR1_CH4N		
P 12	-		P 12	-		EXT_INT2		TMR1_ETR		
P 13	-		P 13	-		EXT_INT4	9)	TMR1_BLK		
P 14	-		P 14	-		SSP2_FSS	13)	CAN2_RX		
P 15	-		P 15	-		SSP2_RXD		CAN2_TX		
Попр D										
PD0	ADC0_REF+	5	PD0	JB_TMS		TMR1_CH1N	3)	UART2_RXD	14)	TMR3_CH1
PD1	ADC1_REF-		PD1	JB_TCK		TMR1_CH1		UART2_TXD		TMR3_CH1N
PD2	ADC2		PD2	JB_TRST		BUSY1	1)	SSP2_RXD	13)	TMR3_CH2
PD3	ADC3		PD3	JB_TDI		-		SSP2_FSS		TMR3_CH2N
PD4	ADC4		PD4	JB_TDO		TMR1_ETR		nSIROUT2	14)	TMR3_BLK
PD5	ADC5		PD5			CLE	1)	SSP2_CLK	13)	TMR2_ETR
PD6	ADC6		PD6			ALE		SSP2_TXD	13)	TMR2_BLK
PD7	ADC7		PD7			TMR1_BLK	3)	nSIRIN2	14)	UART1_RXD
PD8	ADC8		PD8			TMR1_CH4N		TMR2_CH1		UART1_TXD
PD9	ADC9		PD9			CAN2_TX	4)	TMR2_CH1N		SSP1_FSS
PD10	ADC10		PD10			TMR1_CH2	3)	TMR2_CH2		SSP1_CLK
PD11	ADC11		PD11			TMR1_CH2N		TMR2_CH2N		SSP1_RXD
PD12	ADC12		PD12			TMR1_CH3		TMR2_CH3		SSP1_TXD
PD13	ADC13		PD13			TMR1_CH3N		TMR2_CH3N		CAN1_TX
PD14	ADC14		PD14			TMR1_CH4		TMR2_CH4		CAN1_RX
PD15	ADC15		PD15			CAN2_RX	4)	BUSY2	1)	EXT_INT3
Попр E										
PE0	DAC2_OUT	6	PE0			ADDR16	1)	TMR2_CH1	15)	CAN1_RX
PE1	DAC2_REF		PE1			ADDR17		TMR2_CH1N		CAN1_TX
PE2	COMP_IN1	7	PE2			ADDR18		TMR2_CH3		TMR3_CH1
PE3	COMP_IN2		PE3			ADDR19		TMR2_CH3N		TMR3_CH1N
PE4	COMP_REF+		PE4			ADDR20		TMR2_CH4N		TMR3_CH2
PE5	COMP_REF-		PE5			ADDR21		TMR2_BLK		TMR3_CH2N
PE6	OSC_IN32	8	PE6			ADDR22		CAN2_RX	4)	TMR3_CH3
PE7	OSC_OUT32		PE7			ADDR23		CAN2_TX		TMR3_CH3N
PE8	COMP_IN3	7	PE8			ADDR24		TMR2_CH4	15)	TMR3_CH4
PE9	DAC1_OUT	6	PE9			ADDR25		TMR2_CH2		TMR3_CH4N
PE10	DAC1_REF		PE10			ADDR26		TMR2_CH2N		TMR3_ETR
PE11	-		PE11			ADDR27		nSIRIN1		TMR3_BLK
PE12	-		PE12			ADDR28		SSP1_RXD	16)	UART1_RXD
PE13	-		PE13			ADDR29		SSP1_FSS		UART1_TXD
PE14	-		PE14			ADDR30		TMR2_ETR	15)	SCL1
PE15	-		PE15			ADDR31		EXT_INT3	9)	SDA1

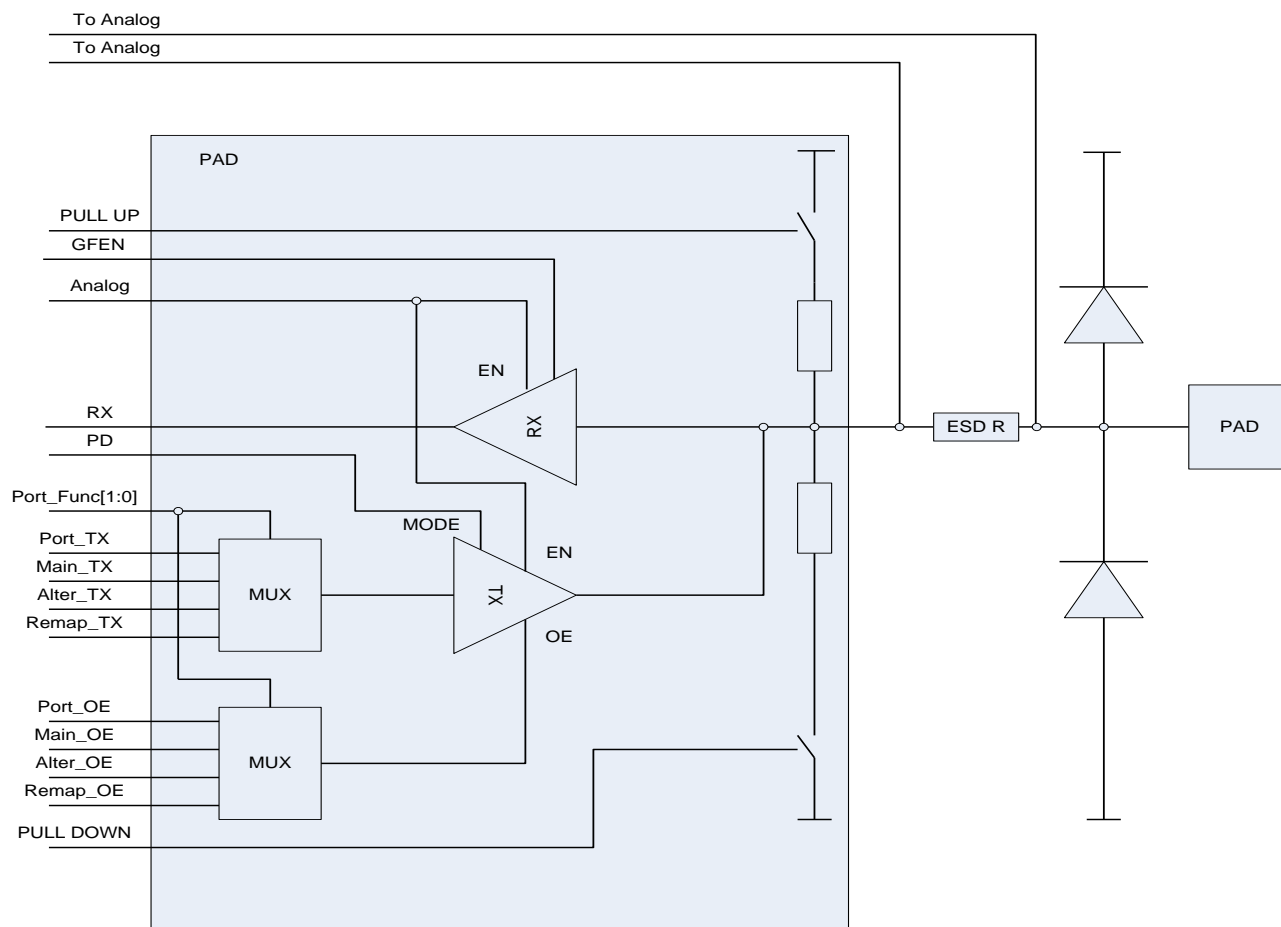
**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

Порт F									
PF0	-	PF0	ADDR0	1)	SSP1TXD	16)	UART2_RXD	14)	
PF1	-	PF1	ADDR1		SSP1CLK		UART2_TXD		
PF2	-	PF2	ADDR2		SSP1FSS		CAN2_RX		
PF3	-	PF3	ADDR3		SSP1RXD		CAN2_TX		
PF4	-	PF4 MODE[0]	ADDR4		-		-		
PF5	-	PF5 MODE[1]	ADDR5		-		-		
PF6	-	PF6 MODE[2]	ADDR6		TMR1_CH1	3)	-		
PF7	-	PF7	ADDR7		TMR1_CH1N		TMR3_CH1		
PF8	-	PF8	ADDR8		TMR1_CH2		TMR3_CH1N		
PF9	-	PF9	ADDR9		TMR1_CH2N		TMR3_CH2		
PF10	-	PF10	ADDR10		TMR1_CH3		TMR3_CH2N		
PF11	-	PF11	ADDR11		TMR1_CH3N		TMR3_ETR		
PF12	-	PF12	ADDR12		TMR1_CH4		SSP2_FSS		
PF13	-	PF13	ADDR13		TMR1_CH4N		SSP2_CLK		
PF14	-	PF14	ADDR14		TMR1_ETR		SSP2_RXD		
PF15	-	PF15	ADDR15		TMR1_BLK		SSP2_TXD		

Примечания:

- |     |         |          |
|-----|---------|----------|
| 1)  |         | EXT_BUS. |
| 2)  |         | CAN1.    |
| 3)  | 1.      |          |
| 4)  |         | CAN2.    |
| 5)  | .       |          |
| 6)  | .       |          |
| 7)  | .       |          |
| 8)  |         | LSE.     |
| 9)  |         | .        |
| 10) |         | UART1.   |
| 11) |         | I2C.     |
| 12) | 3.      |          |
| 13) |         | SSP2.    |
| 14) |         | UART2.   |
| 15) | 2.      |          |
| 16) |         | SSP1.    |
| 17) | 1986 94 |          |





**Рисунок 32. Порты ввода-вывода**

## 16.1 Описание регистров портов ввода-вывода

**Таблица 123 – Описание регистров портов ввода-вывода**

Базовый Адрес	Название	Описание
0x400A_8000	MDR_PORTA	
0x400B_0000	MDR_PORTB	B
0x400B_8000	MDR_PORTC	C
0x400C_0000	MDR_PORTD	D
0x400C_8000	MDR_PORTE	E
0x400E_8000	MDR_PORTF	F
<b>Смещение</b>		
0x00	RXTX[15:0]	MDR_PORTx->RXTX
0x04	OE[15:0]	MDR_PORTx->OE
0x08	FUNC[31:0]	MDR_PORTx->FUNC
0x0C	ANALOG[15:0]	MDR_PORTx->ANALOG
0x10	PULL[31:0]	MDR_PORTx->PULL
0x14	PD[31:0]	MDR_PORTx->PD
0x18	PWR[31:0]	MDR_PORTx->PWR
0x1C	GFEN[15:0]	MDR_PORTx->GFEN

**16.1.1 MDR\_PORTx->RXTX**

**Таблица 124 – Регистр RXTX**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>PORT RXTX[15:0]</b>

**Таблица 125 – Описание бит регистра RXTX**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...0	PORT RXTX[15:0]	.

**16.1.2 MDR\_PORTx->OE**

**Таблица 126 – Регистр OE**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>PORT OE[15:0]</b>

**Таблица 127 – Описание бит регистра OE**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...0	PORT OE[15:0]	.  :
		1 – ; 0 –

**16.1.3 MDR\_PORTx->FUNC**

**Таблица 128 – Регистр FUNC**

<b>Номер</b>	31	30	...	3	2	1	0
<b>Доступ</b>	R/W	R/W	...	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	...	0	0	0	0
	<b>MODE15[1:0]</b>		...	<b>MODE1[1:0]</b>		<b>MODE0[1:0]</b>	

**Таблица 129 – Описание бит регистра FUNC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	MODEx	MODE0
1...0	MODE0[1:0]	. 00 – ; 01 – ; 10 – ; 11 –

### 16.1.4 MDR\_PORTx->ANALOG

**Таблица 130 – Регистр ANALOG**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>ANALOG EN[15:0]</b>

**Таблица 131 – Описание бит регистра ANALOG**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...0	ANALOG EN[15:0]	: 0 – ; 1 –

### 16.1.5 MDR\_PORTx->PULL

**Таблица 132 – Регистр PULL**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	R/W	R/W
<b>Сброс</b>	0	0
	<b>PULL UP[15:0]</b>	<b>PULL DOWN[15:0]</b>

**Таблица 133 – Описание бит регистра PULL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	PULL UP[15:0]	: 0 – ; 1 – ( ~50 )
15...0	PULL DOWN[15:0]	: 0 – ; 1 – ( ~ 50 )

### 16.1.6 MDR\_PORTx->PD

**Таблица 134 – Регистр PD**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	R/W	R/W
<b>Сброс</b>	0	0
	<b>PORT SHM[15:0]</b>	<b>PORT PD[15:0]</b>

**Таблица 135 – Описание бит регистра PD**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	PORT SHM[15:0]	: 0 – 200 ; 1 – 400
15...0	PORT PD[15:0]	: 0 – ; 1 –

### 16.1.7 MDR\_PORTx->PWR

**Таблица 136 – Регистр PWR**

Номер	31	30	...	3	2	1	0
Доступ	R/W	R/W	...	R/W	R/W	R/W	R/W
Сброс	0	0	...	0	0	0	0
	<b>PWR15[1:0]</b>		...	<b>PWR1[1:0]</b>		<b>PWR0[1:0]</b>	

**Таблица 137 – Описание бит регистра PORTx\_PWR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2	PWRx	PWR0
1...0	PWR0[1:0]	: 00 – ( ) 01 – ( 100 ) 10 – ( 20 ) 11 – ( 10 )

### 16.1.8 MDR\_PORTx->GFEN

**Таблица 138 – Регистр GFEN**

Номер	31...16	15...0
Доступ	R/W	R/W
Сброс	0	0
	-	<b>GFEN[15:0]</b>

**Таблица 139 – Описание бит регистра GFEN**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...0	GFEN[15:0]	: 0 – ; 1 – ( 10 )

## 17 Детектор напряжения питания MDR\_POWER

PVD U<sub>CC</sub> BU<sub>CC</sub>

PVD

U<sub>CC</sub> PLS[2:0]  
PVDCS.

PVDCS, BU<sub>CC</sub> PLBS[1:0] PVD PBVD.

( 0 000003FC  
2011) BU<sub>CC</sub><U<sub>CC</sub>

0x83400FDF

BU<sub>CC</sub>

**Таблица 140 – Типовые уровни напряжений детектора питания**

Параметр	Не менее	Типовое	Не более
, U <sub>CC</sub> ,	2,0	-	3,6
, BU <sub>CC</sub> ,	1,8	-	3,6
PVD U <sub>CC</sub> , PLS = "000",		2,0	
PVD U <sub>CC</sub> , PLS = "001",		2,2	
PVD U <sub>CC</sub> , PLS = "010",		2,4	
PVD U <sub>CC</sub> , PLS = "011",		2,6	
PVD U <sub>CC</sub> , PLS = "100",		2,8	
PVD U <sub>CC</sub> , PLS = "101",		3,0	
PVD U <sub>CC</sub> , PLS = "110",		3,2	
PVD U <sub>CC</sub> , PLS = "111",		3,4	
PBVD BU <sub>CC</sub> , PLS = "00",		1,8	
PBVD BU <sub>CC</sub> , PLS = "01",		2,2	
PBVD BU <sub>CC</sub> , PLS = "10",		2,6	
PBVD BU <sub>CC</sub> , PLS = "11",		3,0	

**Таблица 141 – Описание регистров блока PVD**

Базовый Адрес	Название	Описание
0x4005_8000	MDR_POWER	
<b>Смещение</b>		
0x00	PVDCS [12:0]	MDR_POWER->PVDCS

**17.1.1 MDR\_POWER->PVDCS**

**Таблица 142 – Регистр PVDCS**

<b>Номер</b>	9	8	7	6	5...3	2...1	0
<b>Доступ</b>	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	000	00	0
	<b>IEPVVD</b>	<b>IEPVBD</b>	<b>PVD</b>	<b>PVBD</b>	<b>PLS [2:0]</b>	<b>PBLS [1:0]</b>	<b>PVD EN</b>

<b>Номер</b>	31...12	11	10
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	<b>INV</b>	<b>INVB</b>

**Таблица 143 – Описание бит регистра PVDCS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...12	-	
11	INV	PVD: 0 – ; 1 – . , ,
10	INVB	PVBD: 0 – ; 1 – . , ,
9	IEPVVD	PVD: 0 – ; 1 – . , , 0, ,
8	IEPVBD	PVBD: 0 – ; 1 – . , , 0, ,
7	PVD	, 0. : 0 – , PLS; 1 – , PLS <i>Примечание –</i> -

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
6	PVBD	<p align="right">0.</p> <p align="center">:</p> <p>0 – , PBLs;</p> <p>1 – , PBLs</p> <p><u>Примечание –</u></p>
5...3	PLS[2:0]	<p align="center">:</p> <p>000 – 2,0</p> <p>001 – 2,2</p> <p>010 – 2,4</p> <p>011 – 2,6</p> <p>100 – 2,8</p> <p>101 – 3,0</p> <p>110 – 3,2</p> <p>111 – 3,4</p>
2...1	PBLs[1:0]	<p align="center">:</p> <p>00 – 1,8</p> <p>01 – 2,2</p> <p>10 – 2,6</p> <p>11 – 3,0</p>
0	PVDEN	<p align="right">:</p> <p>0 – ;</p> <p>1 –</p>

## 18 Внешняя системная шина MDR\_EBC (кроме 1986BE94Т)

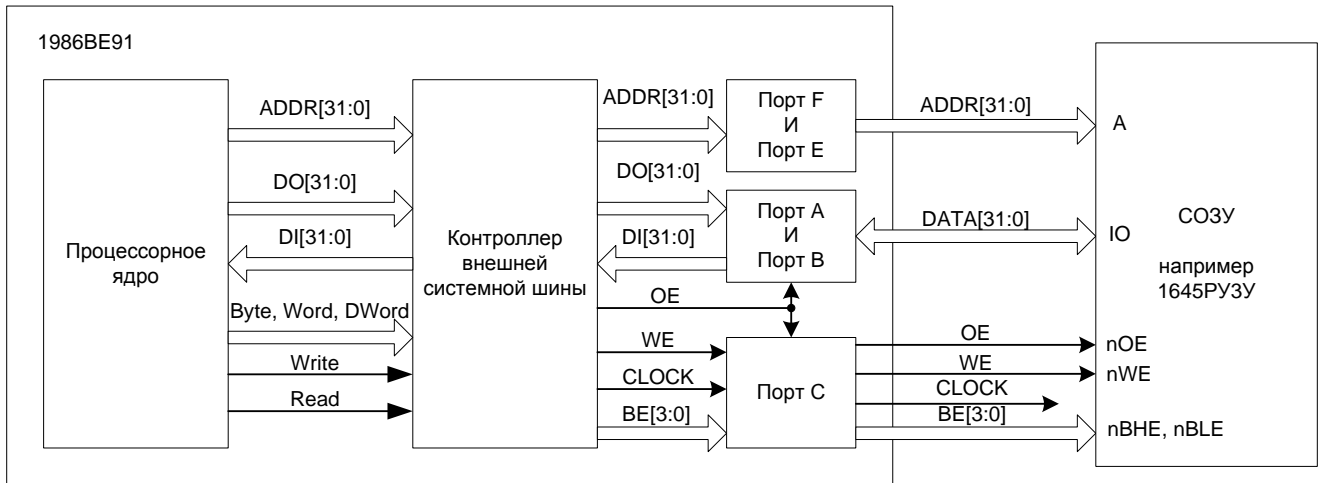
Таблица 144 – Адресные диапазоны внешней системной шины

Адресный диапазон	Размер	Описание
0x1000_0000 – 0x1FFF_FFFF	256	CODE I Code D code
0x3000_0000 – 0x3FFF_FFFF	256	DATA S Bus. DMA
0x5000_0000 – 0xDFFF_FFFF	2,256	PERIPHERAL EXTERNAL BUS S Bus. DMA

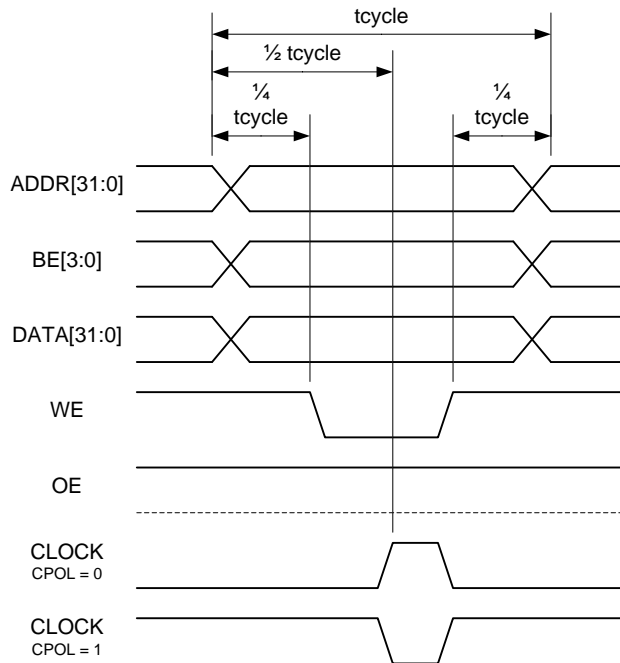
### 18.1 Работа с внешними статическими ОЗУ, ПЗУ и периферийными устройствами

ROM EXT\_BUS\_CONROL. RAM  
WAIT\_STATE[3:0].  
ADDR, DATA OE, WE, BE[3:0]  
CLOCK.



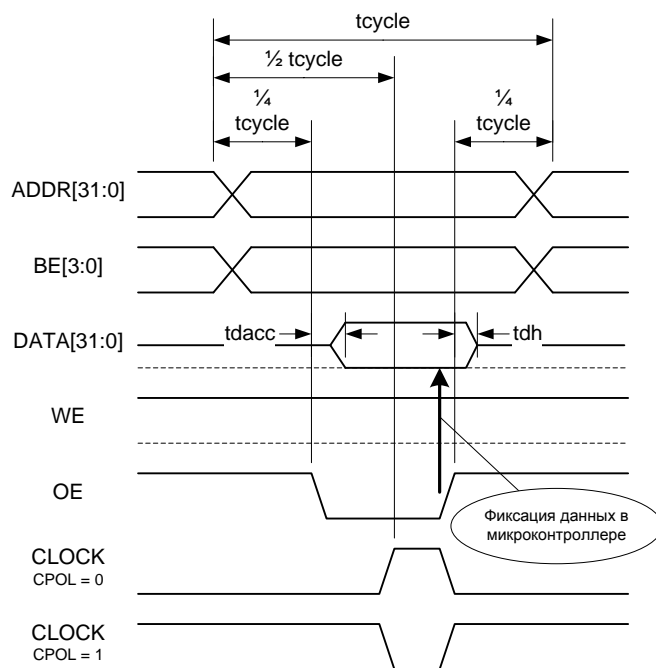


**Рисунок 33. Обмен по внешней системной шине**



**Рисунок 34. Диаграмма записи**

tcycle  
WE, OE, BE[3:0] . WAIT\_STATE[3:0].  
CLOCK



**Рисунок 35. Диаграмма чтения**

tcycle,

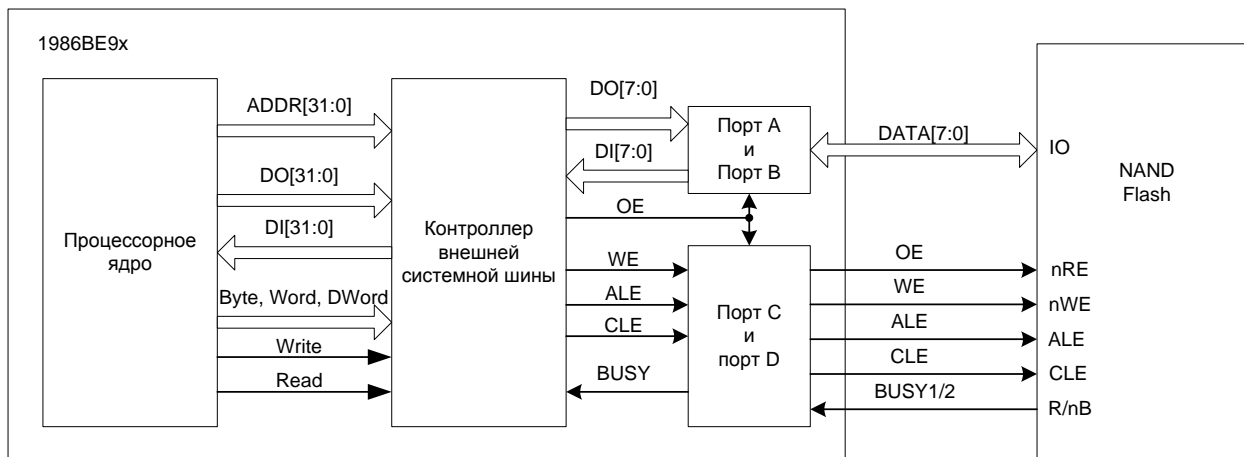
tdh

**Таблица 145. Длительность фаз обращения в тактах процессора**

WAIT_STATE	Предустановка адреса и данных перед сигналом WE или OE	Длительность WE или OE	Удержание адреса и данных после сигнала WE или OE
0	1	1	0
1	1	1	1
2	1	1	1
3	1	2	1
4	2	2	1
5	2	3	1
6	2	3	2
7	2	4	2
8	3	4	2
9	3	5	2
10	3	5	3
11	3	6	3
12	4	6	3
13	4	7	3
14	4	7	4
15	4	8	4

## 18.2 Работа с внешней NAND Flash-памятью

NAND Flash  
EXT\_BUS\_CONROL.  
NAND  
NAND Flash  
NAND Flash-  
NAND\_CYCLES.  
NAND Flash  
DATA[7:0], ALE, CLE, BUSY1 BUSY2.



**Рисунок 36. Подключение внешней NAND Flash**

BUSY1 BUSY2  
NAND Flash.  
BUSY.  
BUSY  
D (BUSY1 –  
PD2 ( ) BUSY2 – PD15 ( ) ).  
NAND Flash-

( 146).

**Таблица 146 – Формат кодирования адреса обращения**

Адрес обращения	Фаза команды	Фаза данных
ADDR[31:24]		: 0x10...0x1F 0x30...0x3F 0x50...0xCF
ADDR[23:21]	ADR_CYCLES[2:0] 000 – 0 001 – 1 ... 111 – 7	A[23:22]
		A[21]

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Адрес обращения	Фаза команды	Фаза данных
ADDR[20]	0 – ; 1 –	:
ADDR[19]	0	1
ADDR[18:11]	ECMD[7:0] 0x10/0x11 - Page Program 0xD0 - Block Erase	
ADDR[10:3]	SCMD[7:0] 0x00/0x01 – Read1 0x50 – Read2 0x90 – Read ID 0xFF – Reset 0x80 – Page Program 0x60 – Block Erase 0x70 – Read Status	
ADDR[2:0]		

NAND Flash-

NAND Flash-

```

// =====
// Инициализация контроллера внешней системной шины для работы с NAND Flash
// =====

NAND_CYCLES = 0x02A63466;
// время trr = 2 цикла HCLK или 20 нс при частоте HCLK 100 МГц
// время talea = 10 циклов
// время twhr = 6 циклов
// время twp = 3 цикла
// время trea = 4 цикла
// время tws = 6 циклов
// время trc = 6 циклов

EXT_BUS_CONTROL = 0x00000004;
//  $\overline{\text{NAND}}$  = 1;

// =====
// Чтение ID микросхемы
// =====

unsigned char IDH;
unsigned char IDL;

// Фаза команды
*((volatile unsigned char *) (0x77200480)) = 0x00;
// ADR CYCLE = 1
// SCMD = 0x90 (READ)
// Address 1 cycle = 0x00

// Фаза данных
IDL = *((volatile unsigned char *) (0x77080000));
IDH = *((volatile unsigned char *) (0x77080000));

```

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

```
// =====
// Стирание блока памяти
// =====

// Фаза команды
*((volatile unsigned char *) (0x70768300))=0x11;
*((volatile unsigned char *) (0x70768301))=0x22;
*((volatile unsigned char *) (0x70768302))=0x33;
// ADR CYCLE = 3
// выполнять завершающую команду
// ЕСМД= 0xD0
// SCMD = 0x60
// Address 1 cycle = 0x11
// Address 2 cycle = 0x22
// Address 1 cycle = 0x33
while (EXT_BUS_CONTROL!=0x080 ) {};
// Ждем R/nB

// Фаза команды
*((volatile unsigned char *) (0x70000380+addon))=0x00;
// ADR CYCLE = 0
// SCMD = 0x70
// Фаза данных
IDL = *((volatile unsigned char *) (0x77080000));
If (IDL & 0x01==0x01) Error ();
// Если бит IO0==1, то стирание не выполнено

// =====
// Запись страницы
// =====

// Фаза команды
*((volatile unsigned char *) (0x70800400))=0x11;
*((volatile unsigned char *) (0x70800400))=0x22;
*((volatile unsigned char *) (0x70800400))=0x33;
*((volatile unsigned char *) (0x70800400))=0x44;
// ADR CYCLE = 4
// SCMD = 0x80

// Фаза данных
*((volatile unsigned char *) (0x70088000+addon))=0xBB;
*((volatile unsigned char *) (0x70088000+addon))=0xCC;
*((volatile unsigned char *) (0x70088000+addon))=0xDD;
// не выполнять завершающую команду
// ЕСМД= 0x10
...
*((volatile unsigned char *) (0x70188000+addon))=0xEE;
// не выполнять завершающую команду
// ЕСМД= 0x10
// Данные 0 - 0xBB, 1 - 0xCC,... N - 0xEE
// N от 1 до 528
while (EXT_BUS_CONTROL!=0x080 ) {};
// Ждем R/nB

// Фаза команды
*((volatile unsigned char *) (0x70000380+addon))=0x00;
// ADR CYCLE = 0
// SCMD = 0x70
// Фаза данных
IDL = *((volatile unsigned char *) (0x77080000));
If (IDL & 0x01==0x01) Error ();
// Если бит IO0==1, то запись не выполнена

// =====
// Чтение страницы
// =====
```

```

// Фаза команды
*((volatile unsigned char *) (0x70800000))=0x11;
*((volatile unsigned char *) (0x70800000))=0x22;
*((volatile unsigned char *) (0x70800000))=0x33;
*((volatile unsigned char *) (0x70800000))=0x44;
// ADR_CYCLE = 4
// SCMD = 0x00
while (EXT_BUS_CONTROL!=0x080 ) {};
// Ждем R/nB

// Фаза данных
IDL=*((volatile unsigned char *) (0x70080000));
IDH=*((volatile unsigned char *) (0x70080000));
If (IDL != 0xBB || IDH != 0xCC) Error ();
// Если считали не то, что записали, то ошибка
    
```

### 18.3 Описание регистров блока контроллера внешней системной шины

**Таблица 147 – Описание регистров блока контроллера внешней системной шины**

Базовый Адрес	Название	Описание
0x400F_0000	MDR_EBC	
<b>Смещение</b>		
0x50	NAND_CYCLES	MDR_EBC->NAND_CYCLES NAND_Flash
0x54	CONTROL	MDR_EBC->CONTROL

### 18.3.1 MDR\_EBC->CONTROL

**Таблица 148 – Регистр CONTROL**

<b>Номер</b>	31...16	15	14	13	12	11...8	7	6...4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	U	RO	U	R/W	R/W	R/W	R/W
<b>Срок</b>	0	0	0	0	0	0	1	0	0	0	0	0
	-	WAIT_STATE[3:0]			-	BUSY	-	CPOL	NAND	RAM	ROM	

**Таблица 149 – Описание бит регистра CONTROL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...12	WAIT STATE[3:0]	<p align="center">AHB, OE/WE</p> <p align="center">¼ WAIT_STATE, ¾ WAIT_STATE:</p> <p>0000 – 3 HCLK 0001 – 4 HCLK ... 1111 – 17 HCLK</p>
11...8	-	
7	BUSY	<p align="center">NAND Flash- :</p> <p>1 – ; 0 –</p>
6..4	-	
3	CPOL	<p align="center">CLOCK:</p> <p>0 – ; 1 –</p>
2	NAND	<p align="center">NAND:</p> <p>1 – NAND; 0 – NAND</p> <p align="right">3.0 ,</p>
1	RAM	<p align="center">RAM:</p> <p>1 – RAM; 0 – RAM</p>
0	ROM	<p align="center">ROM:</p> <p>1 – ROM; 0 – ROM</p>

### 18.3.2 MDR\_EBC->NAND\_CYCLES

**Таблица 150 – Регистр NAND\_CYCLES**

<b>Номер</b>	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Срок</b>		0	0	0	0	0	0	0
	-	t_rr	t_alea	t_whr	t_wp	t_rea	t_wc	t_rc

**Таблица 151 – Описание бит регистра NAND\_CYCLES**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...28		
27...24	t_rr[3:0]	busy : 0000 – 0 HCLK 0001 – 1 HCLK .... 1111 – 15 HCLK NAND Flash 20
23...20	t_alea[3:0]	ID. t_rr. NAND Flash 100
19...16	t_whr[3:0]	t_rr. NAND Flash 60
15...12	t_wp[3:0]	t_rr. NAND Flash 25
11...8	t_rea[3:0]	t_rr. NAND Flash 35
7...4	t_wc[3:0]	t_rr. NAND Flash 60
3...0	t_rc[3:0]	t_rr. NAND Flash 60



## 19 Внешняя системная шина MDR\_EBC (для 1986BE94T)

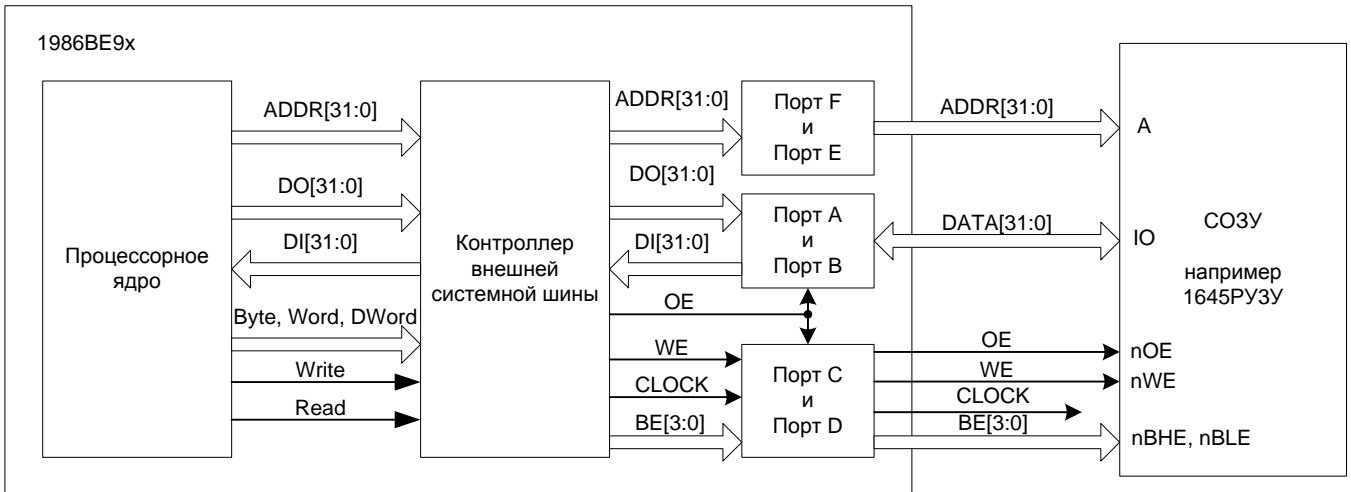
**Таблица 152 – Адресные диапазоны внешней системной шины**

Адресный диапазон	Размер	Описание
0x1000_0000 – 0x1FFF_FFFF	256	CODE I Code D code
0x3000_0000 – 0x3FFF_FFFF	256	DATA S Bus. DMA
0x5000_0000 – 0x5FFF_FFFF	256	PERIPHERAL EXTERNAL BUS S Bus. DMA
0x6000_0000 – 0xDFFF_FFFF	2	PERIPHERAL EXTERNAL BUS S Bus. DMA

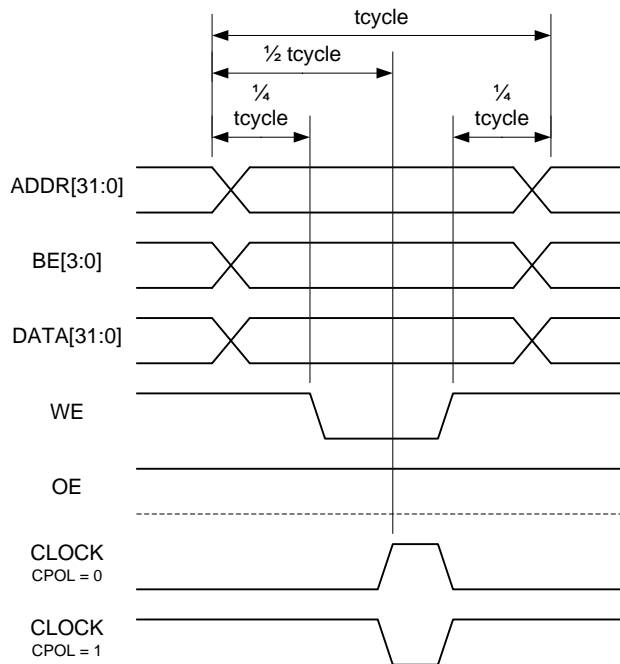
4

### 19.1 Работа с внешними статическими ОЗУ, ПЗУ и периферийными устройствами

	EXT_BUS_CONROL.	RAM
ROM		
ENABLE=0	WAIT_STATE[3:0]	EXT_BUS_CTRL_MODE=0
WS_HOLD[2:0]	EXT_BUS_CTRL_MODE=1	WS_ACTIVE[6:0], WS_SETUP[2:0]
		ENABLE=1
		ADDR, DATA
OE, WE, BE[3:0]		CLOCK. CLOCK
		EXT_BUS_CTRL_MODE=0
		EXT_BUS_CTRL_MODE=1
CLOCK		CLOCKOUT_EN=1. CPOL.

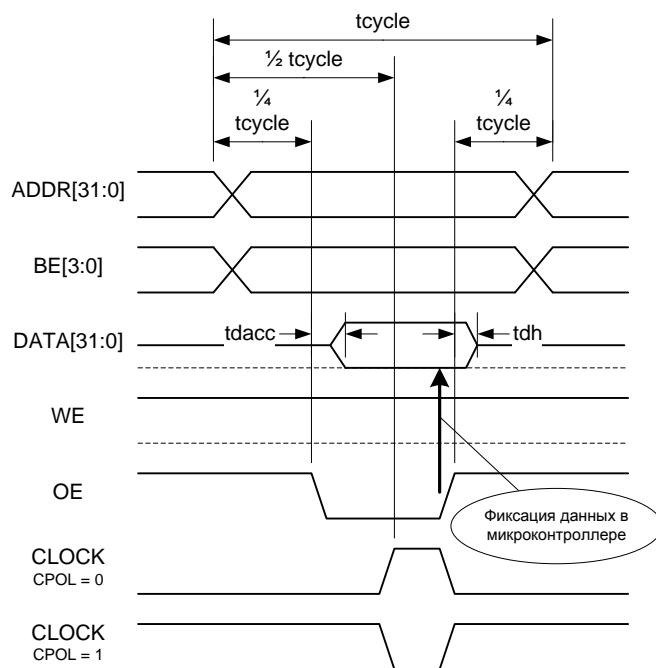


**Рисунок 37. Обмен по внешней системной шине при задании длительности через биты WAIT\_STATE**



**Рисунок 38. Диаграмма записи**

WE, OE, BE[3:0]      tcycle      WAIT\_STATE[3:0].      CLOCK



**Рисунок 39. Диаграмма чтения**

tcycle,

tdh

**Таблица 153 – Длительность фаз обращения в тактах процессора при EXT\_BUS\_CTRL\_MODE=0 или ENABLE=0 для соответствующего региона**

WAIT_STATE	Предустановка адреса и данных перед сигналом WE или OE	Длительность WE или OE	Удержание адреса и данных после сигнала WE или OE
0	1	1	0
1	1	1	1
2	1	1	1
3	1	2	1
4	2	2	1
5	2	3	1
6	2	3	2
7	2	4	2
8	3	4	2
9	3	5	2
10	3	5	3
11	3	6	3
12	4	6	3
13	4	7	3
14	4	7	4
15	4	8	4

**19.1.1 Обмен по внешней системной шине при задании длительности через WS\_ACTIVE, WS\_SETUP, WS\_HOLD**

WS_SETUP [2:0]	–	REGx_CNTRL,		WE/OE	:
		/	,		
		(WS_SETUP + 1)		1 8;	
WS_ACTIVE [6:0]	–			WE/OE	
		/	,		
		(WS_ACTIVE + 1)		1 128;	
WS_HOLD [2:0]	–			WE/OE	
		/	,		
		(WS_HOLD + 1)		1 8;	
ENABLE	–				
					:
	0	–			
				WAIT_STATE;	
	1	–		EXT_BUS_CNTRL_MODE = 1	
					WS_ACTIVE,
					WS_SETUP WS_HOLD.

**19.2 Работа с внешней NAND Flash-памятью**

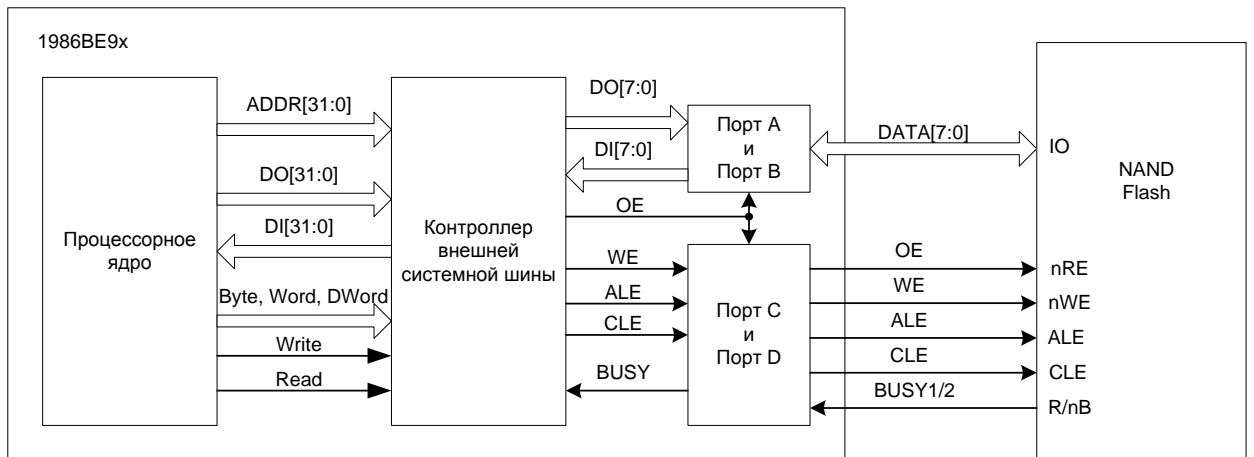
NAND Flash  
EXT\_BUS\_CNTRLOL.

NAND NAND Flash .

NAND Flash- NAND\_CYCLES.

, NAND Flash

DATA[7:0], ALE, CLE, BUSY1 BUSY2.



**Рисунок 40. Подключение внешней NAND Flash**



**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

**NAND Flash-**

```
// =====  
// Инициализация контроллера внешней системной шины для работы с NAND Flash  
// =====  
  
NAND_CYCLES = 0x02A63466;  
// время trr = 2 цикла HCLK или 20 нс при частоте HCLK 100 МГц  
// время talea = 10 циклов  
// время twhr = 6 циклов  
// время twr = 3 цикла  
// время trea = 4 цикла  
// время tws = 6 циклов  
// время trs = 6 циклов  
  
EXT_BUS_CONTROL = 0x00000004;  
// NAND = 1;  
  
// =====  
// Чтение ID микросхемы  
// =====  
  
unsigned char IDH;  
unsigned char IDL;  
  
// Фаза команды  
*((volatile unsigned char *) (0x77200480)) = 0x00;  
// ADR_CYCLE = 1  
// SCMD = 0x90 (READ)  
// Address 1 cycle = 0x00  
  
// Фаза данных  
IDL = *((volatile unsigned char *) (0x77080000));  
IDH = *((volatile unsigned char *) (0x77080000));  
  
// =====  
// Стирание блока памяти  
// =====  
  
// Фаза команды  
*((volatile unsigned char *) (0x70768300))=0x11;  
*((volatile unsigned char *) (0x70768301))=0x22;  
*((volatile unsigned char *) (0x70768302))=0x33;  
// ADR_CYCLE = 3  
// выполнять завершающую команду  
// ESCMD= 0xD0  
// SCMD = 0x60  
// Address 1 cycle = 0x11  
// Address 2 cycle = 0x22  
// Address 1 cycle = 0x33  
while (EXT_BUS_CONTROL!=0x080 ) {};  
// Ждем R/nB  
  
// Фаза команды  
*((volatile unsigned char *) (0x70000380+addon))=0x00;  
// ADR_CYCLE = 0  
// SCMD = 0x70  
// Фаза данных  
IDL = *((volatile unsigned char *) (0x77080000));  
If (IDL & 0x01==0x01) Error ();  
// Если бит IO0==1, то стирание не выполнено  
  
// =====  
// Запись страницы  
// =====  
  
// Фаза команды  
*((volatile unsigned char *) (0x70800400))=0x11;  
*((volatile unsigned char *) (0x70800400))=0x22;  
*((volatile unsigned char *) (0x70800400))=0x33;  
*((volatile unsigned char *) (0x70800400))=0x44;  
// ADR_CYCLE = 4  
// SCMD = 0x80
```

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

```

// Фаза данных
*((volatile unsigned char *) (0x70088000+addon))=0xBB;
*((volatile unsigned char *) (0x70088000+addon))=0xCC;
*((volatile unsigned char *) (0x70088000+addon))=0xDD;
// не выполнять завершающую команду
// ЕСМД= 0x10
...
*((volatile unsigned char *) (0x70188000+addon))=0xEE;
// не выполнять завершающую команду
// ЕСМД= 0x10
// Данные 0 - 0xBB, 1 - 0xCC,... N - 0xEE
// N от 1 до 528
while (EXT_BUS_CONTROL!=0x080 ) {};
// Ждем R/nB

// Фаза команды
*((volatile unsigned char *) (0x70000380+addon))=0x00;
// ADR CYCLE = 0
// SCMD = 0x70
// Фаза данных
IDL = *((volatile unsigned char *) (0x77080000));
If (IDL & 0x01==0x01) Error ();
// Если бит IO==1, то запись не выполнена

// =====
// Чтение страницы
// =====

// Фаза команды
*((volatile unsigned char *) (0x70800000))=0x11;
*((volatile unsigned char *) (0x70800000))=0x22;
*((volatile unsigned char *) (0x70800000))=0x33;
*((volatile unsigned char *) (0x70800000))=0x44;
// ADR CYCLE = 4
// SCMD = 0x00
while (EXT_BUS_CONTROL!=0x080 ) {};
// Ждем R/nB

// Фаза данных
IDL=*((volatile unsigned char *) (0x70080000));
IDH=*((volatile unsigned char *) (0x70080000));
If (IDL != 0xBB || IDH != 0xCC) Error ();
// Если считали не то, что записали, то ошибка

```

### 19.3 Описание регистров блока контроллера внешней системной шины

**Таблица 155 – Описание регистров блока контроллера внешней системной шины**

Базовый Адрес	Название	Описание
0x400F 0000	MDR_EBC	
<b>Смещение</b>		
0x50	NAND_CYCLES	MDR_EBC->NAND_CYCLES NAND_Flash
0x54	CONTROL	MDR_EBC->CONTROL
0x58	REG0_CNTRL	MDR_EBC->REG0_CNTRL 0
0x5C	REG1_CNTRL	MDR_EBC->REG1_CNTRL 1
0x60	REG2_CNTRL	MDR_EBC->REG2_CNTRL 2
0x64	REG3_CNTRL	MDR_EBC->REG3_CNTRL 3

**19.3.1 MDR\_EBC->CONTROL**

156 157 1986 91, 986 92 986 93: 1986 94,  
- - ;  
- - .

**Таблица 156 – Регистр CONTROL**

Номер	31...18	17	16	15...12	11	110	9	8	7	6	5	4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Срок	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
	-	FF_RDY	FS_RDY	WAIT_STATE [3:0]	EN_FF_RDY	EN_FS_RDY	EXT_BUS_CNTRL_MODE	CLOCKOUT_EN	BUSY	LOW16	LOW8	ENDIAN	CPOL	NAND	RAM	ROM

**Таблица 157 – Описание бит регистра CONTROL**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
17	FF_RDY	READY (1) 0 – 1 –
16	FS_RDY	READY (1) 0 – 1 –
15...12	WAIT STATE[3:0]	AHB, OE/WE ¼ WAIT_STATE, ¾ WAIT_STATE: 0000 – 3 HCLK 0001 – 4 HCLK ... 1111 – 17 HCLK
11	EN_FF_RDY	READY (USE_READY=1) 0 – BUS FAULT READY 1 – BUS FAULT READY



**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
10	EN_FS_RDY	<p align="center">READY ( USE_READY=1)</p> <p>0 – BUS FAULT</p> <p>1 – BUS FAULT</p>
9	EXT_BUS_MODE	<p>0 – WAIT_STATE</p> <p>1 – ENABLE=1, WS_ACTIVE, WS_SETUP, WS_HOLD</p>
8	CLOCKOUT_EN	<p align="center">CLOCK WE</p> <p>0 – OE.</p> <p>1 – EXT_BUS_MODE=1</p>
7	BUSY	<p align="center">NAND Flash- :</p> <p>1 – ;</p> <p>0 –</p>
6	LOW16	<p align="center">16 PA[15:0]</p> <p>0 –</p> <p>1 – 16 PA[15:0]</p> <p>/ 8- ,</p> <p>/ 16- ,</p> <p>/ 32- ,</p>
5	LOW8	<p align="center">8 PA[7:0]</p> <p>0 –</p> <p>1 – 8 PA[7:0]</p> <p>/ 16- ,</p> <p>/ 8- ,</p> <p>/ 32- ,</p>
4	ENDIAN	
3	CPOL	<p align="center">CLOCK:</p> <p>0 – ;</p> <p>1 –</p>
2	NAND	<p align="center">NAND:</p> <p>1 – NAND;</p> <p>0 – NAND .</p> <p align="right">3.0 ,</p>

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
1	RAM	RAM: 1 – RAM; 0 – RAM
0	ROM	ROM: 1 – ROM; 0 – ROM

### 19.3.2 MDR\_EBC->NAND\_CYCLES

Таблица 158 – Регистр NAND\_CYCLES

Номер	31-28	27-24	23-20	19-16	15-12	11-8	7-4	3-0
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Срок		0	0	0	0	0	0	0
	-	t_rr	t_alea	t_whr	t_wp	t_rea	t_wc	t_rc

Таблица 159 – Описание бит регистра NAND\_CYCLES

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28		
27...24	t_rr[3:0]	busy : 0000 – 0 HCLK 0001 – 1 HCLK .... 1111 – 15 HCLK NAND Flash 20
23...20	t_alea[3:0]	ID. t_rr. NAND Flash 100
19...16	t_whr[3:0]	t_rr. NAND Flash 60
15...12	t_wp[3:0]	t_rr. NAND Flash 25
11...8	t_rea[3:0]	t_rr. NAND Flash 35
7...4	t_wc[3:0]	t_rr. NAND Flash 60
3...0	t_rc[3:0]	t_rr. NAND Flash 60

**19.3.3 MDR\_EBC->REGx\_CNTRL**

1986 94,

1986 91, 986 92 986 93.

	31...15	14	13...11	10...8	7...1	0
	U	R/W	R/W	R/W	R/W	R/W
C		0	010	010	100000	0
		USE_READY	WS_HOLD	WS_SETUP	WS_ACTIVE	ENABLE

№	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31..15		
14	USE_READY	PC[0], READY 1 – 0 – READY ACTIVE, READY – HOLD , 256. После этого обмен завершается в любом случае
13..11	WS_HOLD[2:0]	nWE/nOE, 1 8
10..8	WS_SETUP[2:0]	nWE/nOE / , 1 8
7..1	WS_ACTIVE[6:0]	nWE/nOE / , 1 128
0	ENABLE	EXT_BUS_CNTRL_MODE = 1 1 – 0 –

## 20 Контроллер интерфейса MDR\_USB

USB (Host) : USB 2.0. (Device)  
 USB : Full Speed (12 / ) Low Speed (1.5 / ), (CRC), NRZI / , (Control), (Bulk), (Isochronous) (Interrupt), USB Device  
 1- 4 ; USB Device  
 FIFO 64 . USB Host 16 . USB  
 Host: FIFO 64 ; SOF ;

### 20.1 Инициализация контроллера при включении

USB. « ». HSE. USB  
 48 .  
 PLL\_USB.  
 2 16, PLLUSBMUL PLL\_CONTROL.  
 48 2...16 ,  
 PLLRDY. USB PLLUSBON.  
 ( 3 PER\_CLOCK).  
 USB\_CLOCK USBCLKEN,  
 USBC1SEL USBC2SEL. PLLUSBON  
 PLLUSBMUL PLL\_CONTROL, USBPLL.  
 USB  
 RESET\_CORE USB\_HSCR.  
 10 .  
 USB ( , , ).

### 20.2 Задание параметров шины USB и события подключения/отключения

USB USB Host USB Device.  
 HOST\_MODE HSCR (0 – Device, 1 – EN\_RX  
 Host). / USB  
 EN\_TX .  
 (EN\_TX=0).  
 EN\_RX=0.

Device SC.  
SCFSR (0 – 1,5 / , 1 – 12 / ), SCFSP (0 – Low speed, 1 – Full speed)

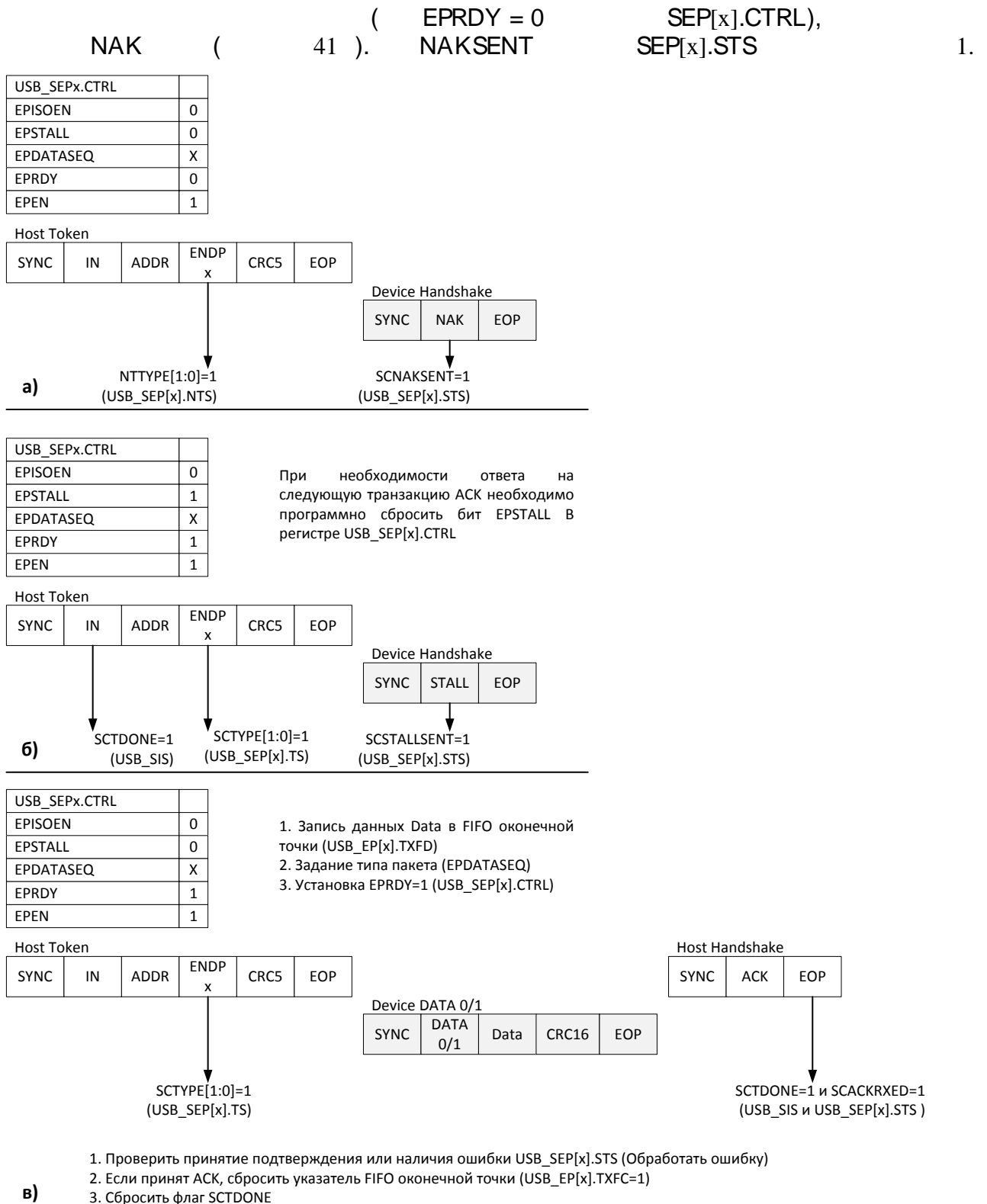
Host HTXLC.  
FSLR (0 – 1,5 / , 1 – 12 / ), FSPL (0 – Low speed, 1 – Full speed)

Host  
CONEV USB\_HSI 1

### 20.3 Задание адреса и инициализация оконечных точек

USB SA.  
(SCGEN = 1 SC). EPEN  
SEP[x].CTRL  
EPISEN SEP[x].CTRL.

## 20.4 Транзакция IN (USB Device)



**Рисунок 41. Транзакция IN (USB Device)**

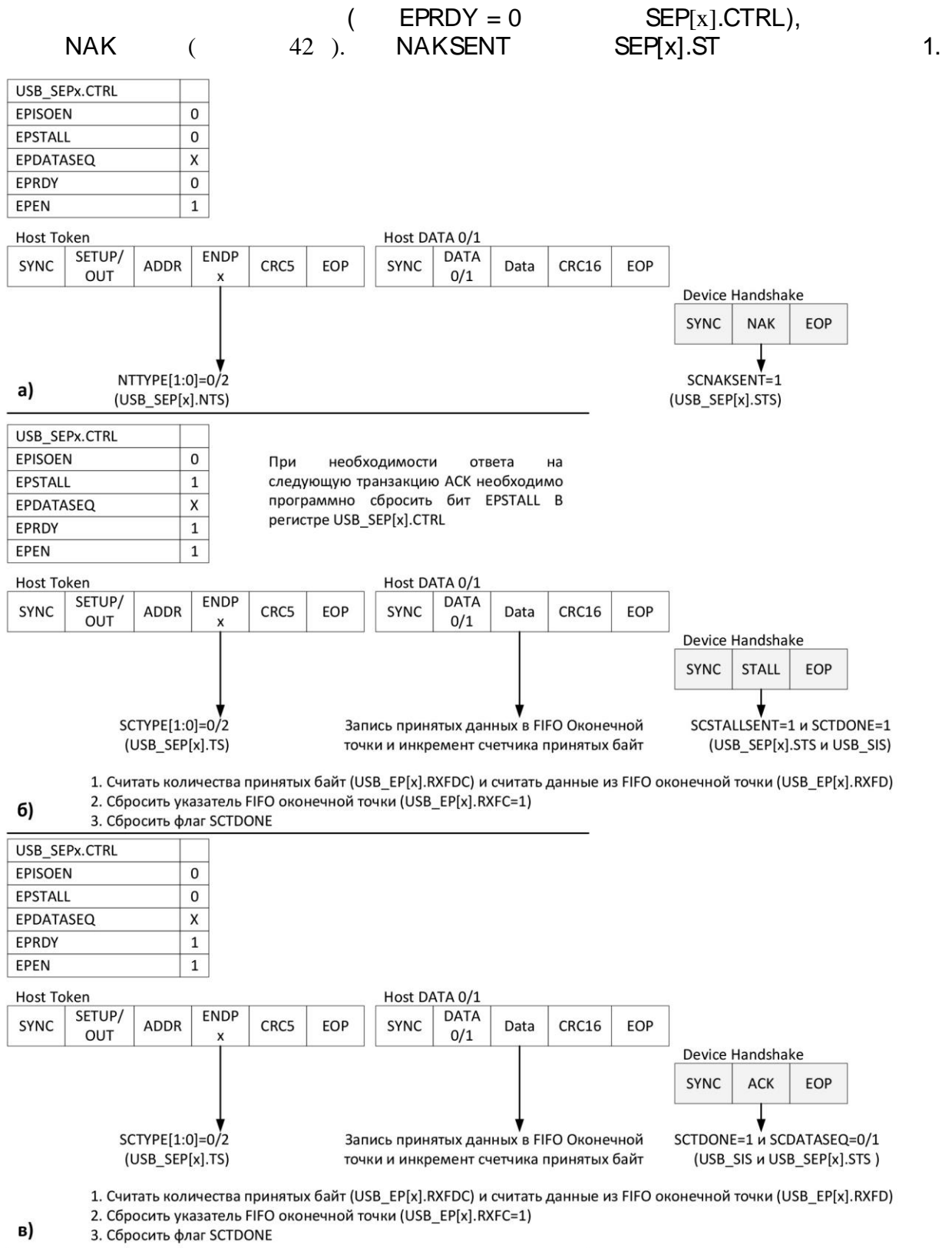
**а – оконечная точка не готова;  
б – установлен бит EPSTALL;  
в – оконечная точка готова**

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

---

		STALL	(	41	).	EPSSTALL SCSTALLSENT	SEP[x].CTRL, SEP[x].STS
	1.						
			(	41	),	SCTTYPE[1:0]	SEP[x].TS
		1				,	.
EP[x].TXFD FIFO	0.	FIFO	.	1	EP[x].TXFC		64
	64				FIFO. FIFO.		
					ACK	,	SCACKRXED
	SEP[x].STS		1.		SEP[x].CTRL,		
	EPDATA SEQ DATA0, DATA1.						

## 20.5 Транзакция SETUP/OUT (USB Device)



**Рисунок 42. Транзакция SETUP/OUT (USB Device)**

**а – оконечная точка не готова;**  
**б – установлен бит EPSTALL;**  
**в – оконечная точка готова**

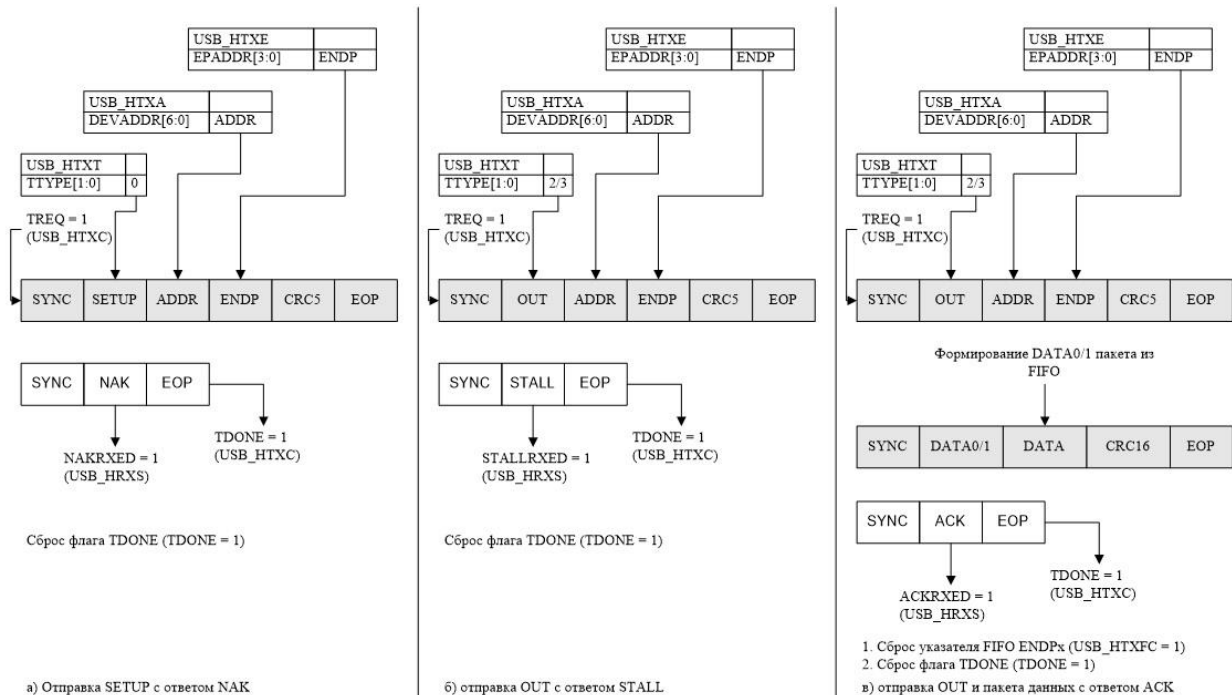


1. STALL ( 42 ). EPSSTALL SCSTALLSENT SEP[x].CTRL, SEP[x].STS

SCTTYPE[1:0] SEP[x].TS ( 42 ) SETUP, 00  
 , . OUT, SCTTYPE[1:0] = 2.  
 DATA0/DATA1 ,  
 FIFO .  
 EP[x].RXFD. EP[x].RXFDC.  
 EP[x].RXFC. FIFO 1

## 20.6 Транзакция SETUP/OUT (USB Host)

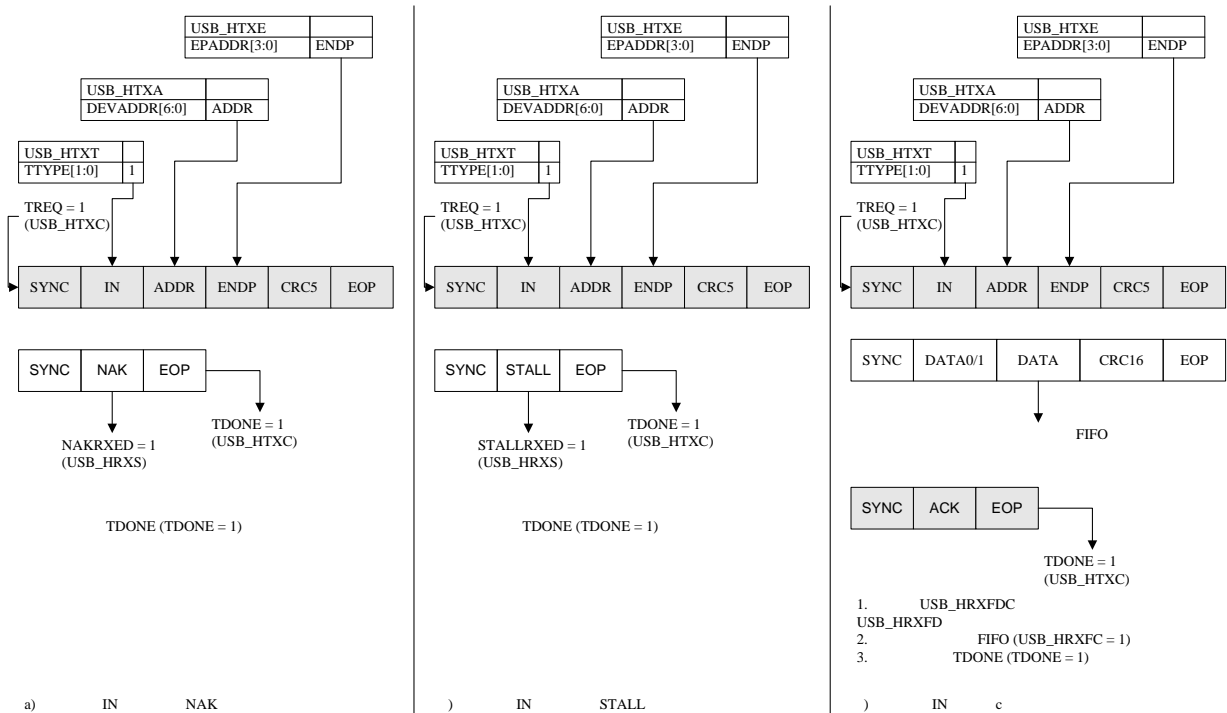
( HTXE) token ( HTXT). ( HTXA),  
 HTXFD. 64  
 64 FIFO. 1 HTXFC  
 FIFO. 0. TREQ  
 HTXC. Host Setup/Out .  
 TDONE = 1 ( HSI).  
 1. PID  
 HRXP.  
 NAK ( 43 ), NAKRXED = 1 ( HRXS).  
 STALL ( 43 ), STALLRXED = 1 ( HRXS).  
 HRXS).  
 ACK ( 43 ), ACKRXED = 1 ( HRXS).



**Рисунок 43 (а, б, в). Транзакция SETUP/OUT (USB Host)**

## 20.7 Транзакция IN (USB Host)

( HTXE) token ( HTXT). ( HTXA),  
 TREQ HTXC. Host IN .  
 TDONE = 1 ( HSI).  
 1. PID  
 HRXP.  
 NAK ( 44 ), NAKRXED = 1 ( HRXS).  
 STALL ( 44 ), STALLRXED = 1 ( HRXS).  
 HRXS).  
 DATA0/DATA1 ( 44 ),  
 FIFO .  
 HRXFD.  
 HRXFDC.  
 FIFO 1 HRXFC. DATASEQ  
 HRXS (0 – DATA0, 1 – DATA1).



**Рисунок 44 (а, б, в). Транзакция IN (USB Host)**

## 20.8 Отправка SOF пакетов и отсчет времени (USB Host)

SOF Full speed, SOF  
 HTXSE. FSPL = 1 ( TXLC),  
 1 . FSPL = 0,  
 EOP 1 .  
 SOF SOFS = 1 ( HSI).  
 1.  
 48 48000 1 .  
 HSTM. , .

## 20.9 Описание регистров управление контроллером USB интерфейса

**Таблица 160 – Описание регистров управление контроллером USB интерфейса**

Базовый Адрес	Название	Описание
0x4001_0000	MDR_USB	USB
<b>Смещение</b>		
0x380	MDR_USB->HSCR	USB
0x384	MDR_USB->HSVR	USB
	<b>Контроллер HOST</b>	
0x00	MDR_USB->HTXC	
0x04	MDR_USB->HTXT	
0x08	MDR_USB->HTXLC	USB
0x0C	MDR_USB->HTXSE	SOF
0x10	MDR_USB->HTXA	
0x14	MDR_USB->HTXE	
0x18	MDR_USB->HFN_L	
0x1C	MDR_USB->HFN_H	SOF
0x20	MDR_USB->HSI	
0x24	MDR_USB->HIM	
0x28	MDR_USB->HRXS	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Базовый Адрес	Название	Описание
0x2C	MDR_USB->HRXP	PID
0x30	MDR_USB->HRXA	,
0x34	MDR_USB->HRXE	,
0x38	MDR_USB->HRXCS	
0x3C	MDR_USB->HSTM	
0x80	MDR_USB->HRXFD	
0x88	MDR_USB->HRXFDC_L	
0x8C	MDR_USB->HRXFDC_H	
0x90	MDR_USB->HRXFC	
0xC0	MDR_USB->HTXFD	
0xD0	MDR_USB->HTXFC	
	<b>Контроллер SLAVE</b>	
0x100 0x110 0x120 0x130	MDR_USB->SEP[x].CTRL	
0x104 0x114 0x124 0x134	MDR_USB->SEP[x].STS	
0x108 0x118 0x128 0x138	MDR_USB->SEP[x].TS	
0x10C 0x11C 0x12C 0x13C	MDR_USB->SEP[x].NTS	NAK
0x140		SLAVE
	MDR_USB->SC	
0x144	MDR_USB->SLS	USB
0x148		SLAVE
	MDR_USB->SIS	
0x14C	MDR_USB->SIM	SLAVE
0x150	MDR_USB->SA	
0x154	MDR_USB->SFN_L	
0x158	MDR_USB->SFN_H	

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

<b>Базовый Адрес</b>	<b>Название</b>	<b>Описание</b>
0x180 0x200 0x280 0x300	MDR_USB->SEP[x].RXFD	
0x188 0x18C 0x208 0x20C 0x288 0x28C 0x308 0x30C	MDR_USB->SEP[x].RXFDC_L MDR_USB->SEP[x].RXFDC_H	
0x190 0x210 0x290 0x310	MDR_USB->SEP[x].RXFC	
0x1C0 0x240 0x2C0 0x340	MDR_USB->SEP[x].TXFD	
0x1D0 0x250 0x2D0 0x350	MDR_USB->SEP[x].TXFDC	

**20.9.1 MDR\_USB->HSCR**

**Таблица 161 – Регистр HSCR**

<b>Номер</b>	31...8	7	6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	-	<b>D- PULL DOWN</b>	<b>D- PULL UP</b>	<b>D+ PULL DOWN</b>	<b>D+ PULL UP</b>	<b>EN RX</b>	<b>EN TX</b>	<b>RESET CORE</b>	<b>HOST MODE</b>

**Таблица 162 – Описание бит регистра HSCR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7	D- PULLDOWN	D-: 0 – ; 1 –
6	D- PULLUP	D-: 0 – ; 1 –
5	D+ PULLDOWN	D+: 0 – ; 1 –
4	D+ PULLUP	D+: 0 – ; 1 –
3	EN_RX	USB: 0 – ; 1 – .
2	EN_TX	USB: 0 – ; 1 – .
1	RESET_CORE	: 1 – ( 10 USBCLK); 0 –
0	HOST_MODE	: 1 – HOST; 0 – Device

### 20.9.2 MDR\_USB->HSVR

Таблица 163 – Регистр HSVR

Номер	31...8	7...4	3...0
Доступ	U	RO	RO
Сброс	0	0	0
	-	REVISION	VERSION

Таблица 164 – Описание бит регистра HSVR

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7...4	REVISION	
3...0	VERSION	

### 20.9.3 Регистры HOST режима

#### 20.9.3.1 MDR\_USB->HTXC

Таблица 165 – Регистр HTXC

Номер	31...4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0
	-	ISOEN	PREEN	SOFS	TREQ

Таблица 166 – Описание бит регистра HTXC

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений.
31...4	-	
3	ISOEN	1 – : , ACK TRANS_TYPE_REG IN_TRANS , OUTDATA0_TRANS. ; 0 –
2	PREEN	1 – : host low speed full speed FULL_SPEED_LINE_RATE_BIT.
1	SOFS	1 – SOF: SOF. 0 – SOF;
0	TREQ	1 – : ; 0 –



**20.9.3.2 MDR\_USB->HTXT**

**Таблица 167 – Регистр HTXT**

<b>Номер</b>	31...2	1	0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	<b>TTYPE</b>	

**Таблица 168 – Описание бит регистра HTXT**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	-	
1...0	TTYPE	: 00 – setup_trans 01 – in_trans 10 – outdata0_trans 01 – outdata1_trans

**20.9.3.3 MDR\_USB->HTXLC**

**Таблица 169 – Регистр HTXLC**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>FSLR</b>	<b>FSLP</b>	<b>DC</b>	<b>TXLS[1:0]</b>	

**Таблица 170 – Описание бит регистра HTXLC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	-	
4	FSLR	1 – 12 / 0 – 1,5 /
3	FSPL	1 – FULL SPEED USB. 0 – LOW SPEED USB.  host full speed , full speed , low speed , low speed , full speed low speed ,
2	DC	USB: 1 – USB ; 0 –
1...0	TXLC[1:0]	DIRECT_CONTROL_BIT, USB: TXL [0] = D- TXLC[1] = D+

**20.9.3.4 MDR\_USB->HTXSE**

**Таблица 171 – Регистр HTXSE**

<b>Номер</b>	31...1	0
--------------	--------	---

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Доступ	U	R/W
Сброс	0	0
	-	<b>SOFEN</b>

**Таблица 172 – Описание бит регистра HTXSE**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...1	-	
0	SOFEN	1 – FSPL, SOF full speed 1 . SOF FSPL. FSPL, low EOP 1 . ( ). 0 – speed SOF/EOP suspend

**20.9.3.5 MDR\_USB->HTXA**

**Таблица 173 – Регистр HTXA**

Номер	31...7	6...0
Доступ	U	R/W
Сброс	0	0
	-	<b>DEVADDR[6:0]</b>

**Таблица 174 – Описание бит регистра HTXA**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...7	-	
6...0	DEVADDR[6:0]	USB Device address.

**20.9.3.6 MDR\_USB->HTXE**

**Таблица 175 – Регистр HTXE**

Номер	31...4	3...0
Доступ	U	R/W
Сброс	0	0
	-	<b>EPADDR[3:0]</b>

**Таблица 176 – Описание бит регистра HTXE**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...4	-	
3...0	EPADDR[3:0]	Endpoint address.

**20.9.3.7 MDR\_USB->HFN**

**Таблица 177 – Регистр HFN**

Номер	31...11	10...0
Доступ	U	R/W
Сброс	0	0
	-	<b>FNUM[10:0]</b>

**Таблица 178 – Описание бит регистра HFN**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...11	-	
10...0	FNUM[10:0]	

**20.9.3.8 MDR\_USB->HSI**

**Таблица 179 – Регистр HSI**

<b>Номер</b>	31...4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	-	<b>SOFS</b>	<b>CONEV</b>	<b>RESUME</b>	<b>TDONE</b>

**Таблица 180 – Описание бит регистра HSI**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений.</b>
31...4	-	
3	SOFS	1 – , SOF 0 – SOF
2	CONEV	1 – , 0 – . 1.
1	RESUME	1 – , 0 – . 1.
	TDONE	1 – , 0 – 1.

**20.9.3.9 MDR\_USB->HIM**

**Таблица 181 – Регистр HIM**

<b>Номер</b>	31...4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	-	<b>SOFSIE</b>	<b>CONEVIE</b>	<b>RESUMEIE</b>	<b>TDONEIE</b>

**Таблица 182 – Описание бит регистра HIM**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...4	-	
3	SOFIE	1 – SOF. 0 –
2	CONEVIE	1 – 0 –
1	RESUMEIE	1 – 0 –
0	TDONEIE	1 – 0 –

**20.9.3.10 MDR\_USB->HRXS**

**Таблица 183 – Регистр HRXS**

<b>Номер</b>	31...8	7	6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	-	DATA SEQ	ACK RXED	STALL RXED	NAK RXED	RX TO	RXOF	BSERR	CRCER

**Таблица 184 – Описание бит регистра HRXS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7	DATASEQ	IN_TRANS, DATA0 = 0, DATA1 = 1
6	ACK RXED	1 – ACK. 0 – ACK
5	STALL RXED	1 – STALL. 0 – STALL
4	NAK RXED	1 – NAK 0 – NAK
3	RXTO	1 – 0 –
2	RXOF	1 – FIFO 0 –
1	BSERR	1 – stuff 0 – stuff
0	CRCERR	1 – CRC 0 – CRC

**20.9.3.11 MDR\_USB->HRXP**

**Таблица 185 – Регистр HRXP**

<b>Номер</b>	31...4	3...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>RPID[3:0]</b>

**Таблица 186 – Описание бит регистра HRXP**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...4	-	
3...0	RPID[3:0]	Packet identifier

**20.9.3.12 MDR\_USB->HRXA**

**Таблица 187 – Регистр HRXA**

<b>Номер</b>	31...7	6...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>RADDR[6:0]</b>

**Таблица 188 – Описание бит регистра HRXA**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...7	-	
6...0	RADDR[6:0]	

**20.9.3.13 MDR\_USB->HRXE**

**Таблица 189 – Регистр HRXE**

<b>Номер</b>	31...4	3...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>RXENDP[3:0]</b>

**Таблица 190 – Описание бит регистра HRXE**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений.</b>
31...4	-	
3...0	RXENDP[3:0]	

**20.9.3.14 MDR\_USB->HRXCS**

**Таблица 191 – Регистр HRXCS**

Номер	31...2	1	0
Доступ	U	R/W	R/W
Сброс	0	0	0
	-	<b>RXLS[1:0]</b>	

**Таблица 192 – Описание бит регистра HRXCS**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2	-	
1...0	RXLS[1:0]	USB: DISCONNECT = 0 LOW_SPEED_CONNECT = 1 FULL_SPEED_CONNECT = 2

**20.9.3.15 MDR\_USB->HSTM**

**Таблица 193 – Регистр HSTM**

Номер	31...8	7...0
Доступ	U	R/W
Сброс	0	0
	-	<b>HSTM[7:0]</b>

**Таблица 194 – Описание бит регистра HSTM**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7...0	HSTM[7:0]	SOF , 48 , 48000 SOF. 1 .

**20.9.3.16 MDR\_USB->HRXFD**

**Таблица 195 – Регистр HRXFD**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>RX FIFO DATA[7:0]</b>

**Таблица 196 – Описание бит регистра HRXFD**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	RX FIFO DATA[7:0]	IN_TRANS,

**20.9.3.17 MDR\_USB->HRXFDC**

**Таблица 197 – Регистр HRXFDC**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FIFO DATA COUNT[15:0]</b>

**Таблица 198 – Описание бит регистра HRXFDC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...0	FIFO DATA COUNT[15:0]	

**20.9.3.18 MDR\_USB->HRXFC**

**Таблица 199 – Регистр HRXFC**

<b>Номер</b>	31...1	0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FIFO FORCE EMPTY</b>

**Таблица 200 – Описание бит регистра HRXFC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...1	-	
0	FIFO FORCE EMPTY	1

**20.9.3.19 MDR\_USB->HTXFD**

**Таблица 201 – Регистр HTXFD**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>TX FIFO DATA[7:0]</b>

**Таблица 202 – Описание бит регистра HTXFD**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	TX FIFO DATA[7:0]	OUTDATA0_TRANS, OUTDATA1_TRANS,

**20.9.3.20 MDR\_USB->HTXFC**

**Таблица 203 – Регистр HTXFC**

<b>Номер</b>	31...1	0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FIFO FORCE EMPTY</b>

**Таблица 204 – Описание бит регистра HTXFC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...1	-	
0	FIFO FORCE EMPTY	1



## 20.9.4 USB Slave (Device)

### 20.9.4.1 MDR\_USB->SEP[x].CTRL

**Таблица 205 – Регистр SEP[x].CTRL**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>EPISOEN</b>	<b>EPSSTALL</b>	<b>EPDATASEQ</b>	<b>EPRDY</b>	<b>EPEN</b>

**Таблица 206 – Описание бит регистра USB\_SEPx.CTRL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	-	
4	EPISOEN	0 – ; 1 – .
3	EPSSTALL	0 – STALL ; 1 – , , STALL ,
2	EPDATASEQ	0 – IN DATA0; 1 – IN DATA1.
1	EPRDY	0 – ; 1 – .
0	EPEN	0 – ; 1 – .  , NAK ,

### 20.9.4.2 MDR\_USB->SEP[x].STS

**Таблица 207 – Регистр SEP[x].STS**

<b>Номер</b>	31...8	7	6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	-	<b>SC DATA SEQ</b>	<b>SC ACK RXED</b>	<b>SC STALL SENT</b>	<b>NAK SENT</b>	<b>SC RXTO</b>	<b>SC RXOF</b>	<b>SC BS ERR</b>	<b>SC CRC ERR</b>

**Таблица 208 – Описание бит регистра USB\_SEPx.STS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7	SC DATA SEQ	OUT_TRANS, DATA0 = 0, DATA1 = 1
6	SC ACK RXED	0 – ; 1 – ACK
5	SC STALL SENT	0 – STALL; 1 – STALL
4	NAK SENT	1 – NAK 0 – NAK
3	SC RXTO	1 – 0 –
2	SC RXOF	0 – ; 1 –
1	SC BS ERR	0 – ; 1 – STUFF
0	SC CRC ERR	0 – ; 1 – CRC

#### 20.9.4.3 MDR\_USB->SEP[x].TS

**Таблица 209 – Регистр SEP[x].TS**

<b>Номер</b>	<b>31...2</b>	<b>1</b>	<b>0</b>
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	SCTTYPE[1:0]	

**Таблица 210 – Описание бит регистра SEP[x].TS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	-	
1...0	SCTTYPE[1:0]	ENDPOINT_READY_BIT      1 0.  SC_SETUP_TRANS = 0 SC_IN_TRANS = 1 SC_OUTDATA_TRANS = 2

**20.9.4.4 MDR\_USB->SEP[x].NTS**

**Таблица 211 – Регистр SEP[x].NTS**

<b>Номер</b>	31...2	1	0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	NTTYPE[1:0]	

**Таблица 212 – Описание бит регистра USB\_SEPx.NTS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	-	
1...0	NTTYPE[1:0]	NAK. SC_SETUP_TRANS = 0 SC_IN_TRANS = 1 SC_OUTDATA_TRANS = 2

**20.9.4.5 MDR\_USB->SC**

**Таблица 213 – Регистр SC**

<b>Номер</b>	31...6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	-	SCFSR	SCFSP	SCDC	SCTXLS[1:0]	SCGEN	

**Таблица 214 – Описание бит регистра USB\_SC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...6	-	
5	SCFSR	: 1 – 12 / ; 0 – 1,5 /
4	SCFSP	USB : 1 – FULL SPEED; 0 – LOW SPEED
3	SCDC	USB : 1 – 0 –
2...1	SCTXL[1:0]	SC_DIRECT_CONTROL_BIT, SC_TX_LINE_STATE USB : SC_TX_LINE_STATE [2] = D+ SC_TX_LINE_STATE [1] = D-
0	SCGEN	1 – 0 –

**20.9.4.6 MDR\_USB->SLS**

**Таблица 215 – Регистр SLS**

<b>Номер</b>	31...2	1	0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	SCRXLS[1:0]	

**Таблица 216 – Описание бит регистра SLS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2	-	
1...0	SCRXLS[1:0]	USB:  RESET = 0 LOW_SPEED_CONNECT = 1 FULL_SPEED_CONNECT = 2

**20.9.4.7 MDR\_USB->SIS**

**Таблица 217 – Регистр SIS**

<b>Номер</b>	31...6	5	4	3	2	1	0
<b>Доступ</b>	U	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	1	0	0	0	0	0
	-	-	SC NAK SENT	SC SOF REC	SC RESET EV	SC RESUME	SC TDONE

**Таблица 218 – Описание бит регистра USB\_SIS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...6	-	
5	-	
4	SC NAK SENT	NAK  1. 1
3	SC SOF REC	SOF  1. 1
2	SC RESET EV	USB.  1 1
1	SC RESUME	.  1 1
0	SC TDONE	.  1 1

**20.9.4.8 MDR\_USB->SIM**

**Таблица 219 – Регистр SIM**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>SC NAK SENT IE</b>	<b>SC SOF RECIE</b>	<b>SC RESET EVIE</b>	<b>SC RESUME IE</b>	<b>SC TDONE IE</b>

**Таблица 220 – Описание бит регистра B\_SIM**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...6	-	
4	SC NAK SENT IE	NAK: 1 – ; 0 –
3	SC SOF RECIE	SOF: 1 – ; 0 –
2	SC RESET EVIE	: 1 – ; 0 –
1	SC RESUME IE	: 1 – ; 0 –
0	SC TDONE IE	: 1 – ; 0 –

**20.9.4.9 MDR\_USB->SA**

**Таблица 221 – Регистр SA**

<b>Номер</b>	31...7	6...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>SDEVADDR[6:0]</b>

**Таблица 222 – Описание бит регистра SA**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...7	-	
6...0	SDEVADDR[6:0]	USB

**20.9.4.10 MDR\_USB->SFN**

**Таблица 223 – Регистр SFN**

<b>Номер</b>	31...11	10...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FRAME NUM [10:0]</b>

**Таблица 224 – Описание бит регистра SFN**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...11	-	
10...0	FRAME NUM [10:0]	, SOF

**20.9.4.11 MDR\_USB->SEP[x].RXFD**

**Таблица 225 – Регистр SEP[x].RXFD**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>RX FIFO DATA[7:0]</b>

**Таблица 226 – Описание бит регистра SEP[x].RXFD**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	RX FIFO DATA[7:0]	OUTDATA_TRANS    SETUP_TRANS ,    RX_FIFO_DATA

**20.9.4.12 MDR\_USB->SEP[x].RXFDC**

**Таблица 227 – Регистр SEP[x].RXFDC**

<b>Номер</b>	31...16	15...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FIFO DATA COUNT [15:0]</b>

**Таблица 228 – Описание бит регистра SEP[x].RXFDC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...0	FIFO DATA COUNT [15:0]	

**20.9.4.13 MDR\_USB->SEP[x].RXFC**

**Таблица 229 – Регистр SEP[x].RXFC**

<b>Номер</b>	31...1	0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FIFO FORCE EMPTY</b>

**Таблица 230 – Описание бит регистра SEP[x].RXFC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...1	-	
0	FIFO FORCE EMPTY	1

**20.9.4.14 MDR\_USB->SEP[x].TXFD**

**Таблица 231 – Регистр SEP[x].TXFD**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>TX FIFO DATA[7:0]</b>

**Таблица 232 – Описание бит регистра SEP[x].TXFD**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	TX FIFO DATA [7:0]	IN_TRANS

**20.9.4.15 MDR\_USB->SEP[x].TXFDC**

**Таблица 233 – Регистр SEP[x].TXFDC**

<b>Номер</b>	31...1	0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>FIFO FORCE EMPTY</b>

**Таблица 234 – Описание бит регистра SEP[x].TXFDC**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...1	-	
0	FIFO FORCE EMPTY	1

## 21 Контроллер интерфейса MDR\_CAN

CAN.

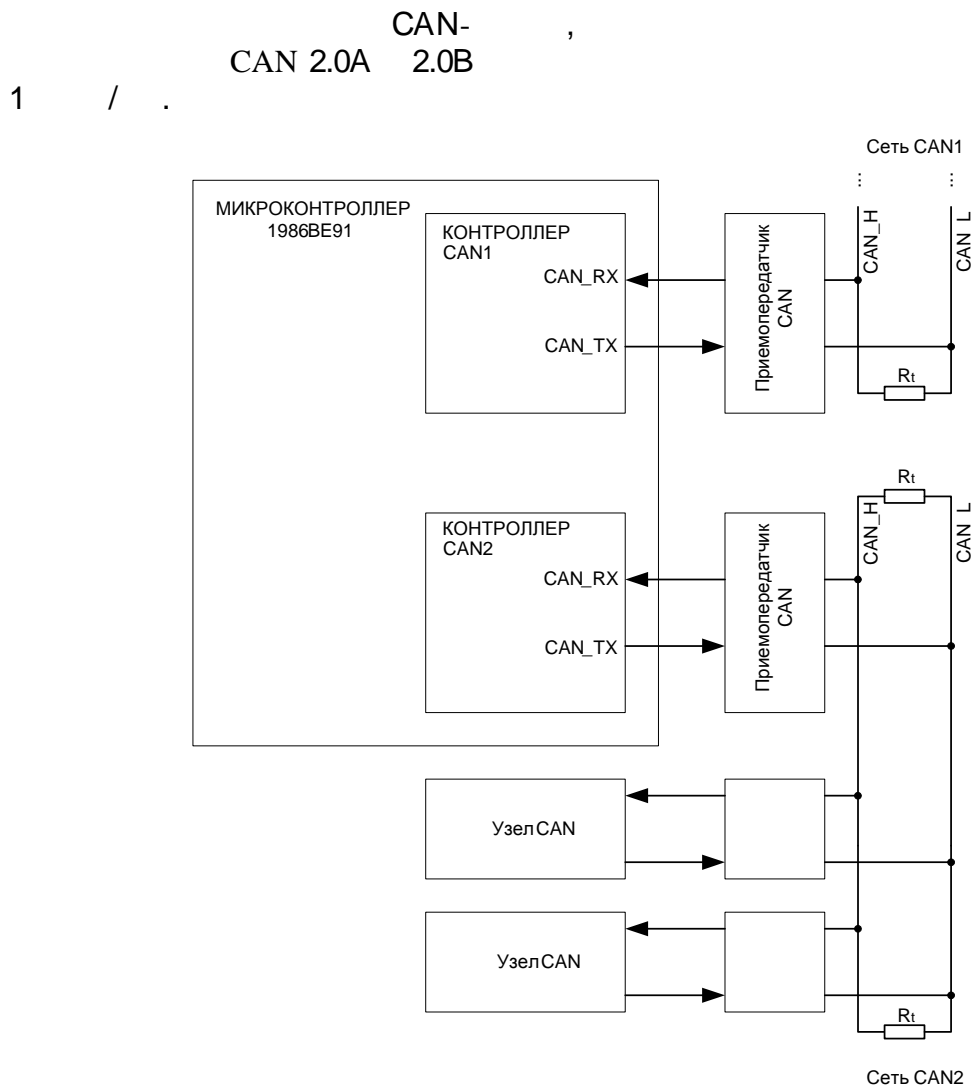


Рисунок 45. Структурная блок – схема организации сети CAN

CAN

CAN

CAN

- CAN CAN 2.0 A B;
- 1 / ;
- 32 / ;
- ;
- 32 ;
- .



## 21.1 Режимы работы

CAN-

- ( CAN\_STATUS : ROM = 0, STM = 0)  
CAN\_TX CAN\_RX

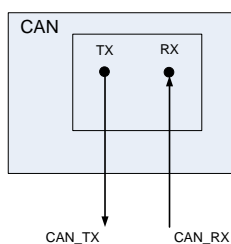


Рисунок 46. Режим нормальной передачи

ACK ( CAN\_CONTROL SAP ROP).

- - Receive Only Mode  
( CAN\_STATUS: ROM = 1, STM = 0)  
CAN  
«1»,

- - Self Test Mode  
( CAN\_STATUS : STM = 1, ROM = 0)

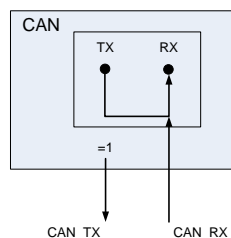


Рисунок 47. Режим работы только на прием - Receive Only Mode

CAN\_TX CAN\_RX

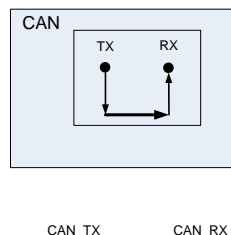
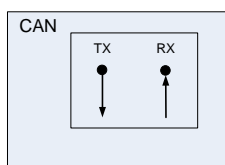


Рисунок 48. Режим самотестирования - Self Test Mode

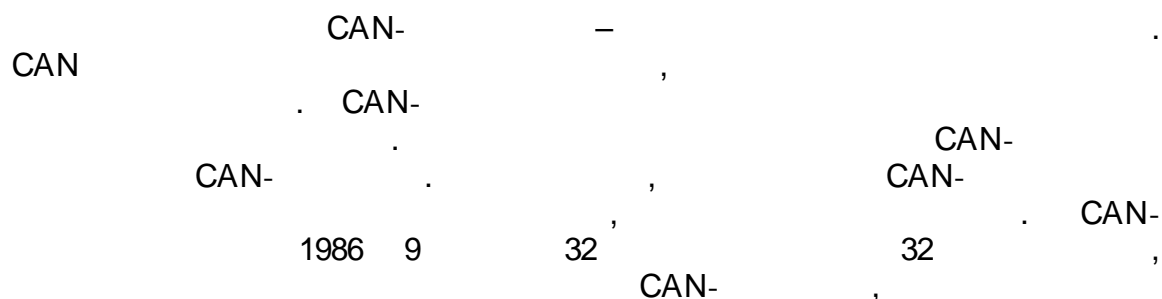
CAN\_CONTROL      SAP      ROP).

ACK (



CAN\_TX      CAN\_RX

**Рисунок 49. Режим инициализации для задания параметров связи**



## 21.2 Типы пакетов сообщений

- пакет удаленного запроса данных

- пакет ошибки

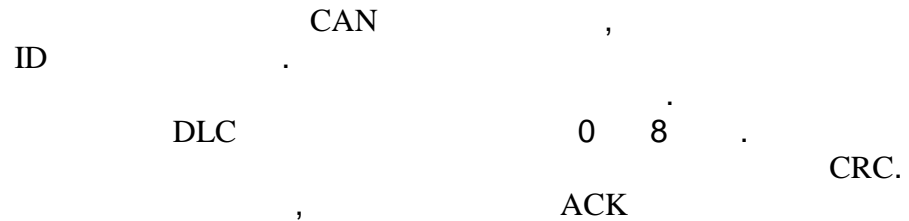
- пакет перегрузки

ISO 11898-1. CAN

- CAN пакеты данных.

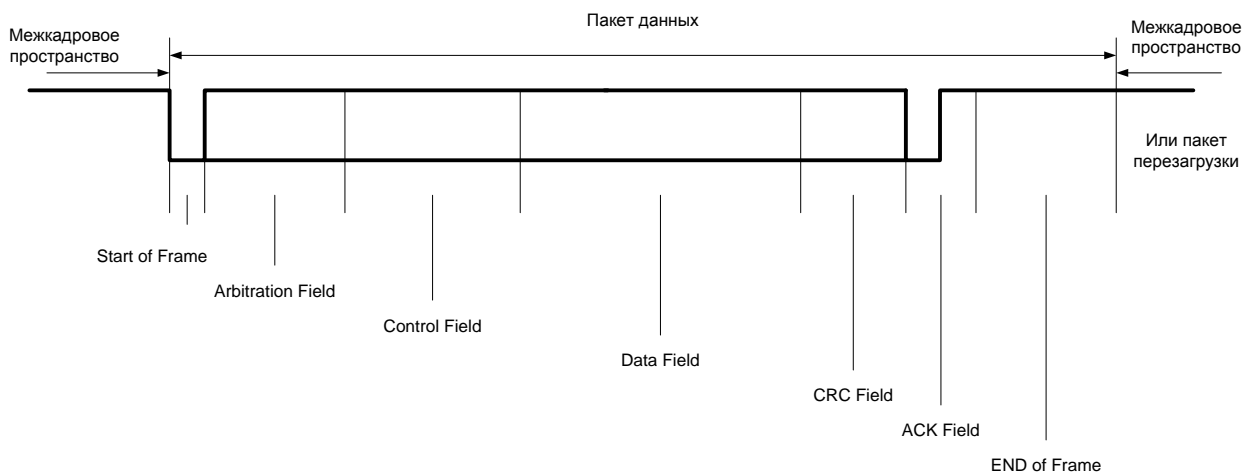
11

29



### 21.3 Структура пакета данных (Data Frame)

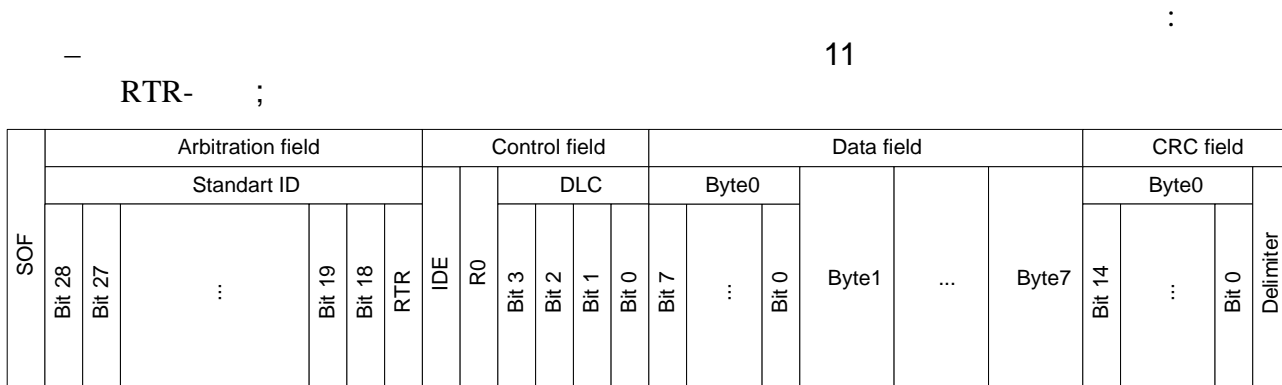
- " (SOF-start of frame);
- " (arbitration field);
- " (control field);
- " (data field);
- " CRC" (CRC field);
- " (ACK field);
- " (end of frame).



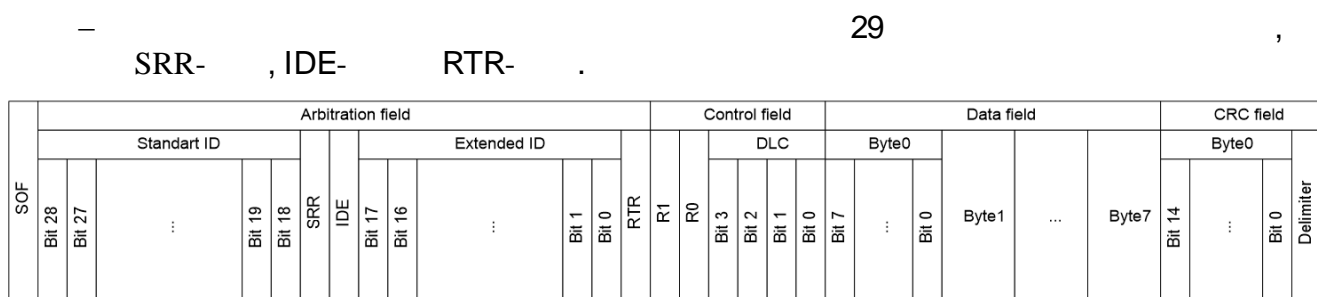
**Рисунок 50. Пакет сообщения CAN**

#### 21.3.1 Начало пакета (Start of frame)

### 21.3.2 Поле арбитража (Arbitration field)

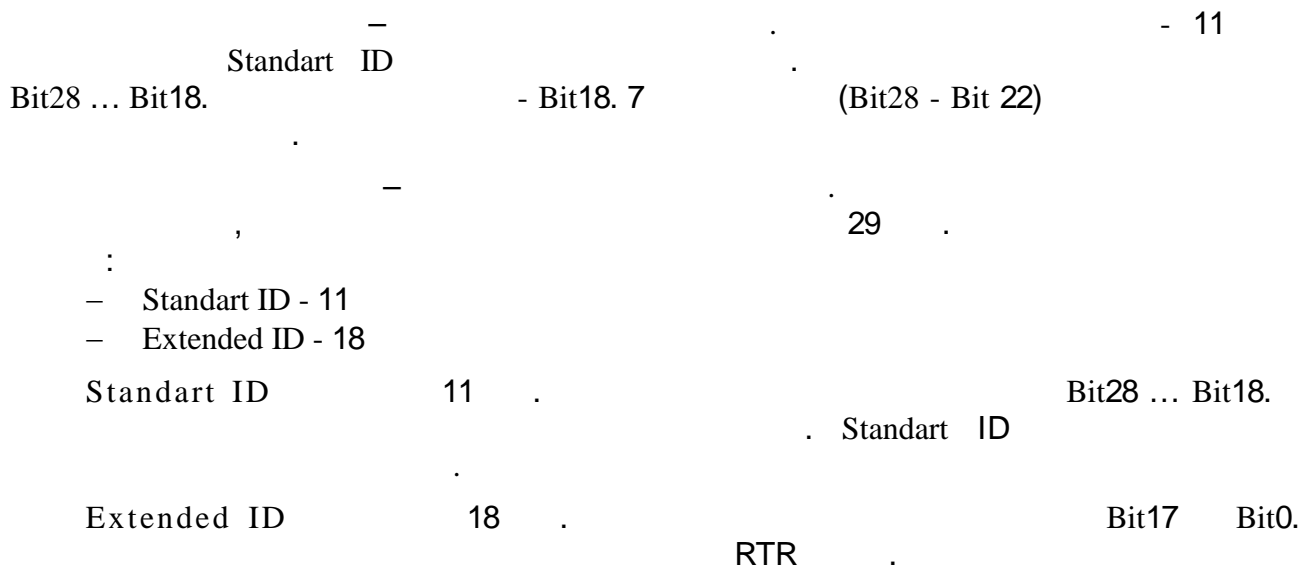


**Рисунок 51. Структура стандартного пакета данных**

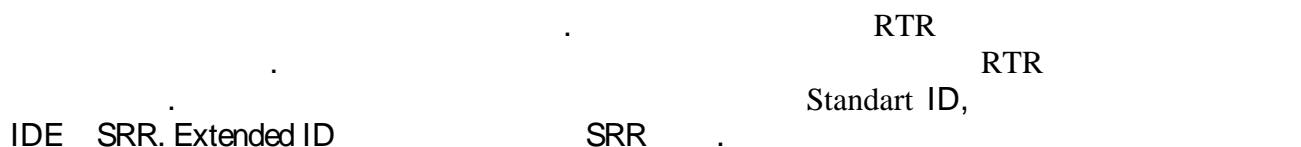


**Рисунок 52. Структура расширенного пакета данных**

#### 21.3.2.1 Идентификатор



#### 21.3.2.2 Бит RTR



### 21.3.2.3 Бит SRR (расширенный формат)

SRR - RTR -

Standart ID

### 21.3.2.4 Бит IDE (расширенный формат)

IDE :  
 - ;  
 - .  
 IDE , IDE  
 - .

### 21.3.3 Поле управления (Control field)

(DLC), IDE,  
 r0.  
 r1 r0.

### Код длины данных (Data length code)

4 :  
 {0,1, ..., 7,8}.

### 21.3.4 Поле данных (Data field)

0 8 , 8 ,

### 21.3.5 Поле CRC (CRC field)

CRC CRC - 15  
 CRC " " " " " ( ) " " CRC  
 CRC,

### 21.3.6 Поле подтверждения (ACK field)

: " " (CRC )

### 21.3.7 Конец пакета (End of frame)

### 21.3.8 Структура пакета удаленного запроса данных (Remote frame)

- " (Start of frame);
- " (Arbitration field);
- " (Control field);
- " CRC" (CRC - field);
- " (ACK field);
- " (End of frame).

, RTR

[0,8].

. RTR

### 21.3.9 Арбитраж на шине

( ).

CAN

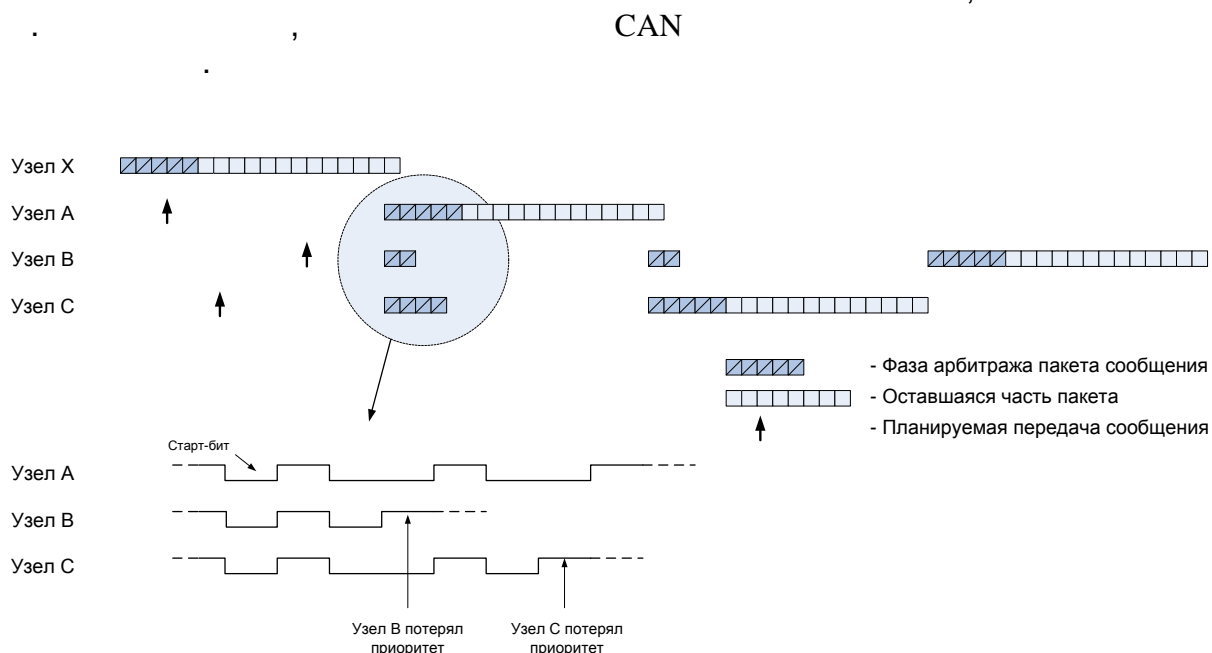
«

».

A C

B

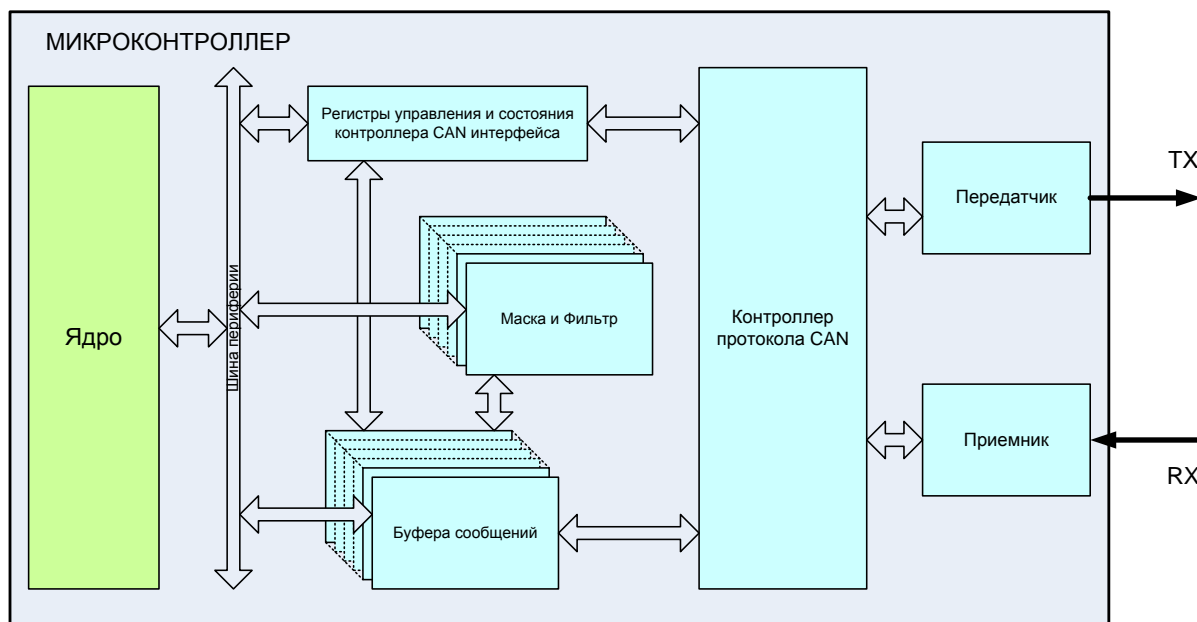
B



**Рисунок 53. Арбитраж на шине CAN**

« ID\_LOWER.

» CAN



**Рисунок 54. Структурная блок-схема контроллера CAN**

## 21.4 Инициализация

CAN

«

»

CAN\_CLOCK (0 CAN1, 1 CAN2 PER\_CLOCK).  
 CANyCLKEN,  
 CAN, HCLK  
 CAN  
 CAN  
 CAN.  
 SB, SJW,  
 SEG2, SEG1, PSEG BRP CAN\_BITTMNG.  
 EN ( ) RXTXn (1 – ,  
 0 – ) BUF\_xx\_CON.  
 CANEN CONTROL.  
 CAN

### 21.5 Передача сообщений

CAN\_BUF[x].DLC, CAN\_BUF[x].DATAL CAN\_BUF[x].DATAH), CAN\_BUF[x].ID,  
 TX\_REQ.  
 TX\_REQ  
 PRIOR\_0 , PRIOR\_0.  
 ID  
 CAN ( ) ID

### 21.6 Передача сообщений по Remote Transmit Request (RTR)

Remote Transmit Request

RTR , INT\_TX  
 (BUFF\_CON[x]) , TX\_REQ = 0,  
 PRIOR\_0, RTR  
 (RTR\_EN=1), RX\_TX = 0 EN = 1  
 SID EID,  
 BUF\_xx\_DLC ( )  
 CAN\_BUF[x].DATAL  
 CAN  
 RTR ,

### 21.7 Прием сообщений

CAN



## 21.8 Автоматическая фильтрация принимаемых сообщений

CAN

(CAN\_BUF\_FILTER[x].FILTER) (CAN\_BUF\_FILTER[x].MASK)

$$ID \& CAN\_BUF\_FILTER[x].MASK == CAN\_BUF\_FILTER[x].FILTER$$

CAN\_BUF\_FILTER[x].MASK CAN\_BUF\_FILTER[x].FILTER

## 21.9 Перезапись принятых сообщений

OVER\_EN.

OVER\_WR.

1, OVER\_WR ( OVER\_WR. RX\_FULL),

RX\_FULL. OVER\_WR

OVER\_WR,

OVER\_WR

## 21.10 Задание скорости передачи и момента семплирования

CAN (NRZ). CAN

DPLL.

CAN 1 / Nominal Bit

Time

$$T_{BIT} = 1/\text{Скорость передачи}$$

DPLL Time Quanta (TQ).

4 :

- Synchronization Segment (Sync\_Seg);
- Propagation Time Segment (PSEG);

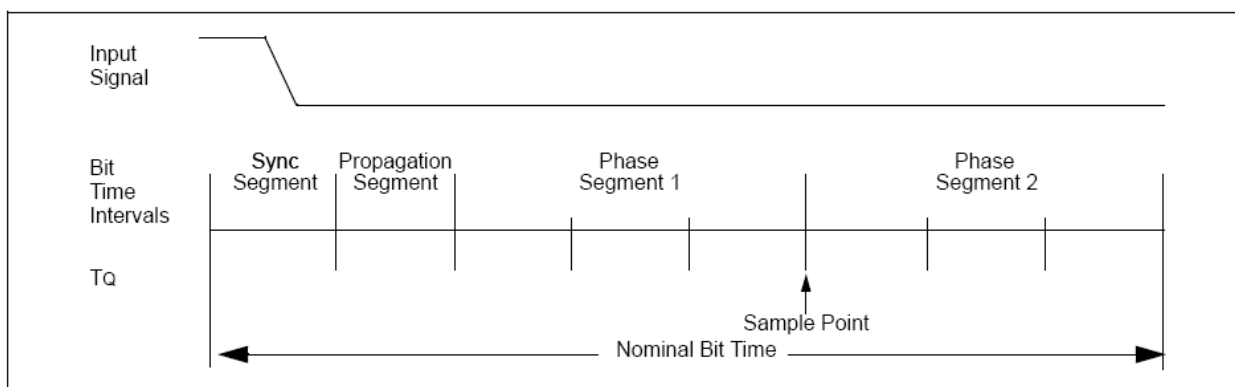
- Phase Buffer Segment 1 (SEG1);
- Phase Buffer Segment 2 (SEG2).

Nominal Bit Time 8 25 TQ.

$$\text{Nominal Bit Time} = TQ * (\text{Sync\_Seg} + \text{PSEG} + \text{SEG1} + \text{SEG2})$$

$$TQ = \frac{BRP + 1}{\text{CANx\_CLK (MHz)}}$$

$$TQ (\mu s) = (BRP + 1) * T_{clk} (\mu s)$$



**Рисунок 55. Структура битового интервала**

Synchronization Segment

$$1 TQ.$$

Propagation Time Segment

$$1 8 TQ$$

Phase Buffer Segments

$$1 8 TQ.$$

## 21.11 Синхронизация

DPLL

Hard Synchronization

DPLL

Sync\_Seg.

Resynchronization

Sync\_Seg.

Phase Segment 1

Phase Segment 2

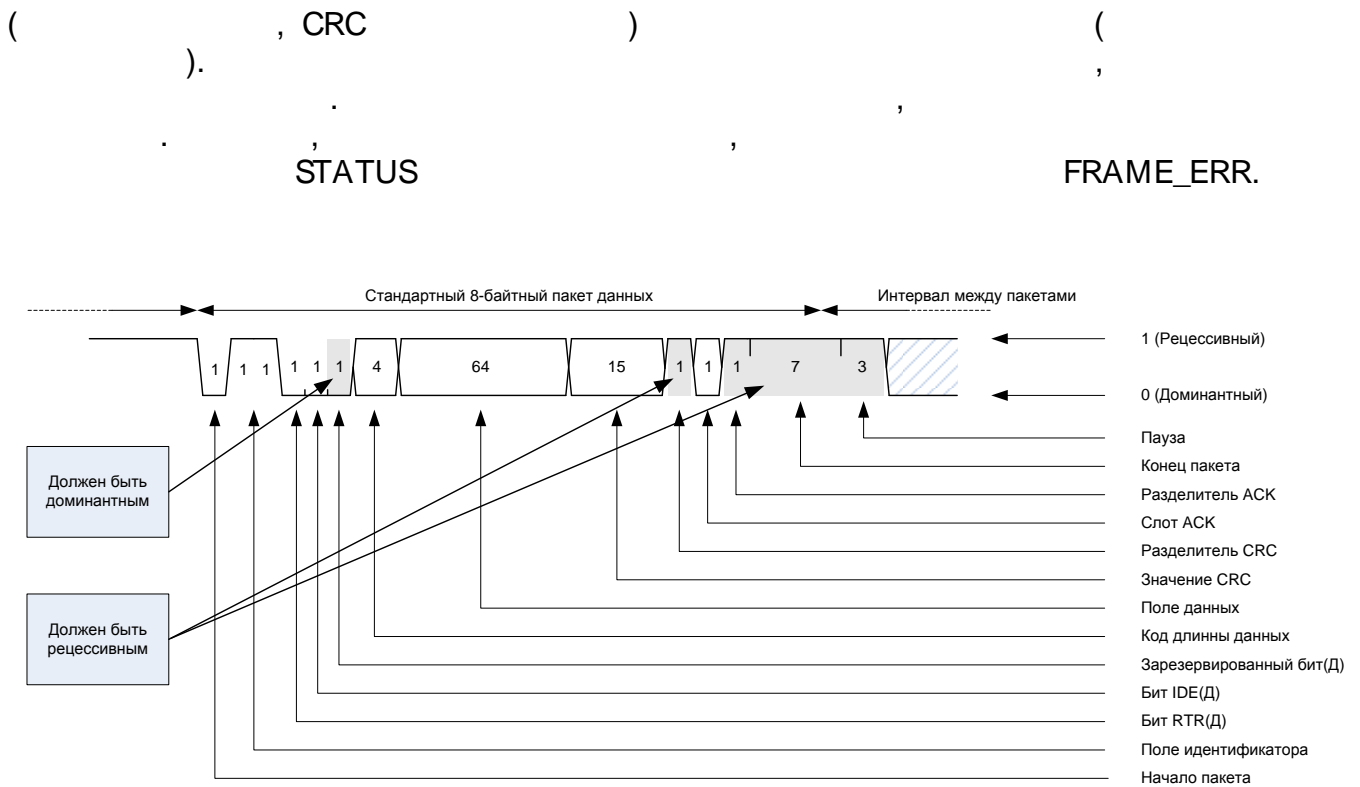
Phase Segment 1

Phase Segment 2

Synchronization Jump Width (SJW).

## 21.12 Обработка ошибок

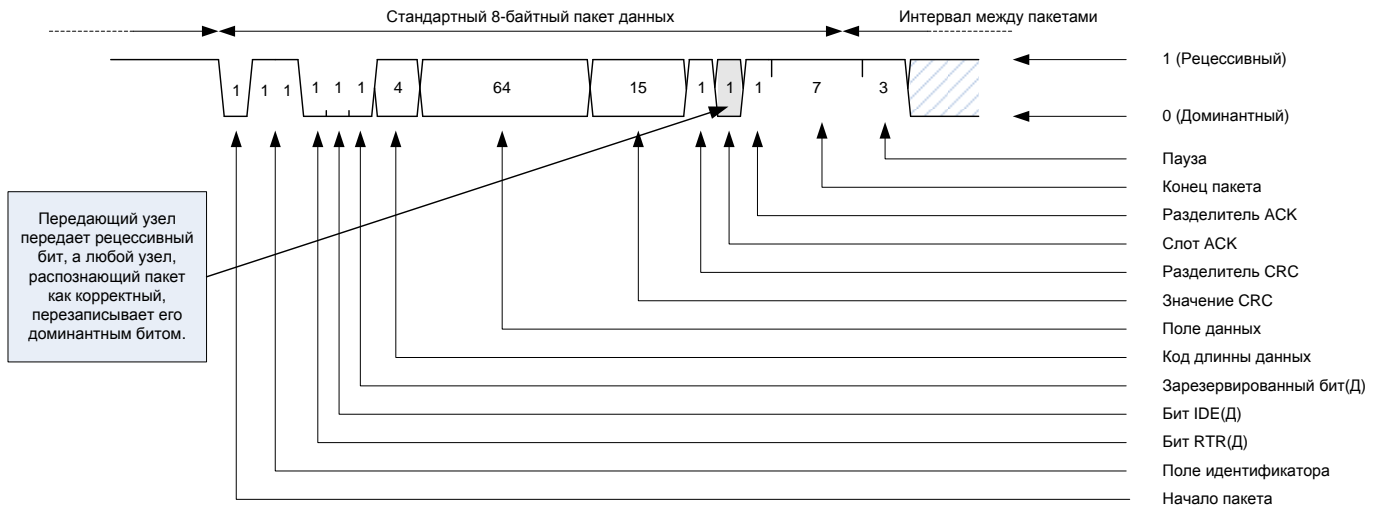
CAN



**Рисунок 56. Контроль формата пакета**

STATUS

ACK\_ERR.



**Рисунок 57. Контроль подтверждения**

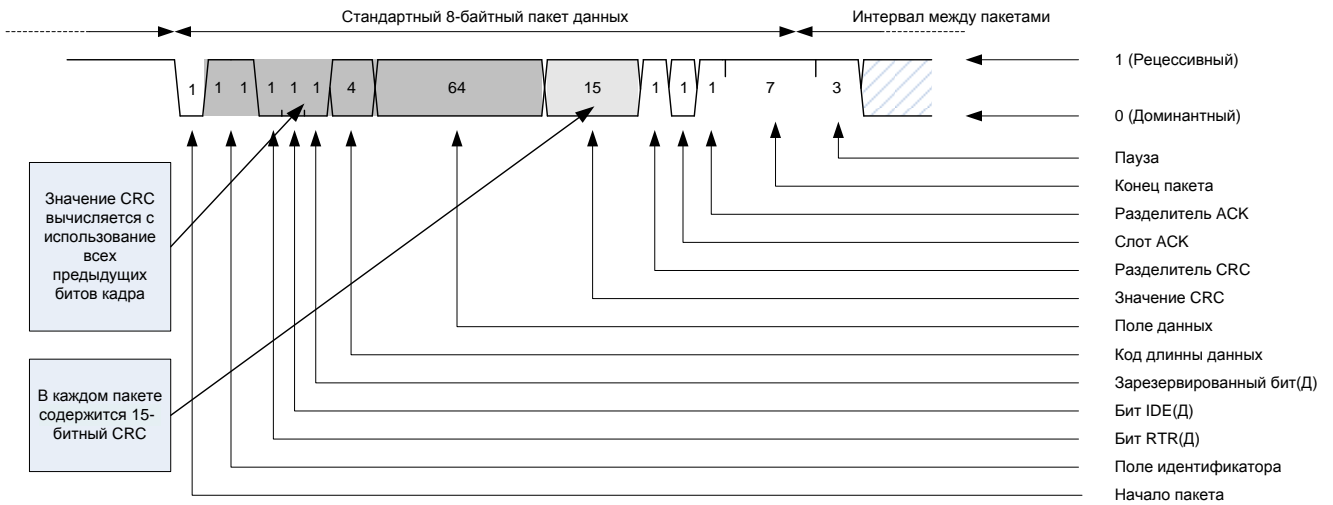
CAN                      15-                      CRC,

4-

CRC.

CRC  
STATUS

CRC\_ERR.

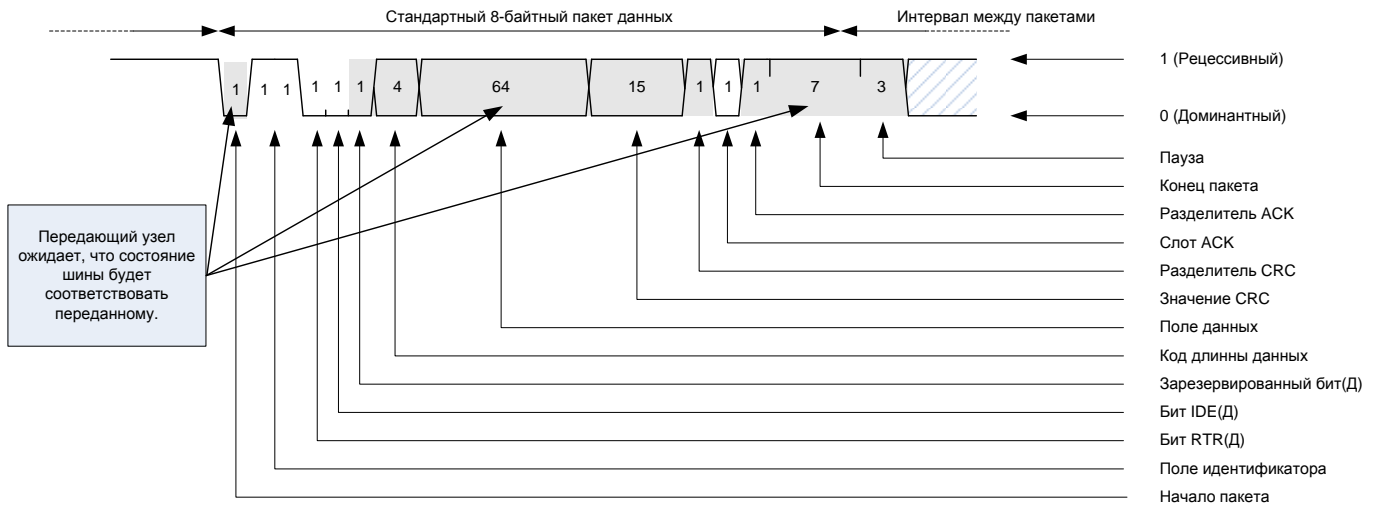


**Рисунок 58. Контроль CRC**

, CAN-

STATUS

BIT\_ERR



**Рисунок 59. Контроль передаваемых бит**

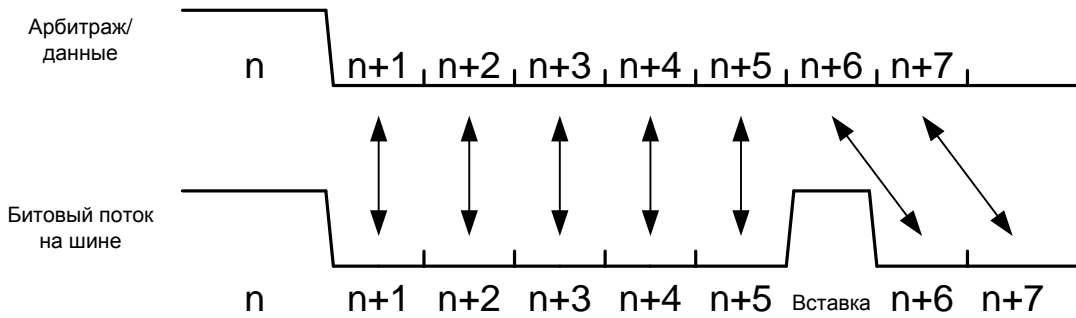
CAN

( )

BIT\_STUF\_ERR.

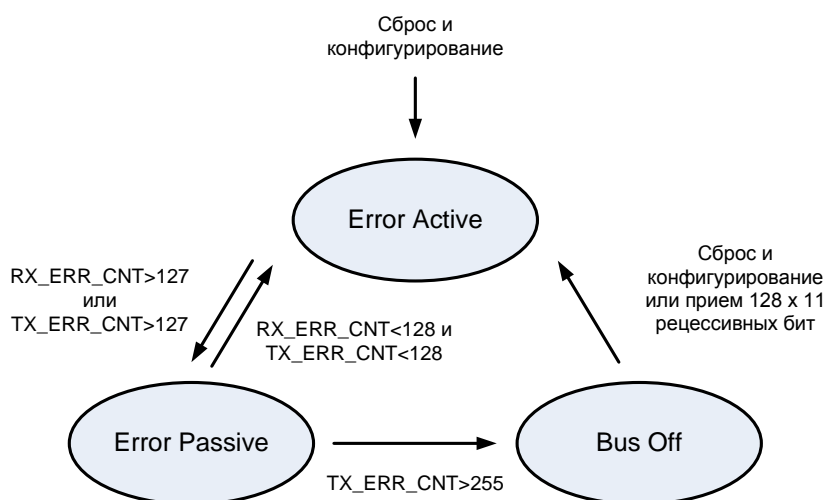
CAN

CAN



**Рисунок 60. Битстаффинг**

CAN  
( STATUS, RX\_ERR\_CNT )  
STATUS, TX\_ERR\_CNT).  
128, CAN  
«error passive».  
255, CAN  
«bus-off»  
( CAN- Error Active)  
CAN-  
CAN ( )  
128 11 = 1408  
STATUS.

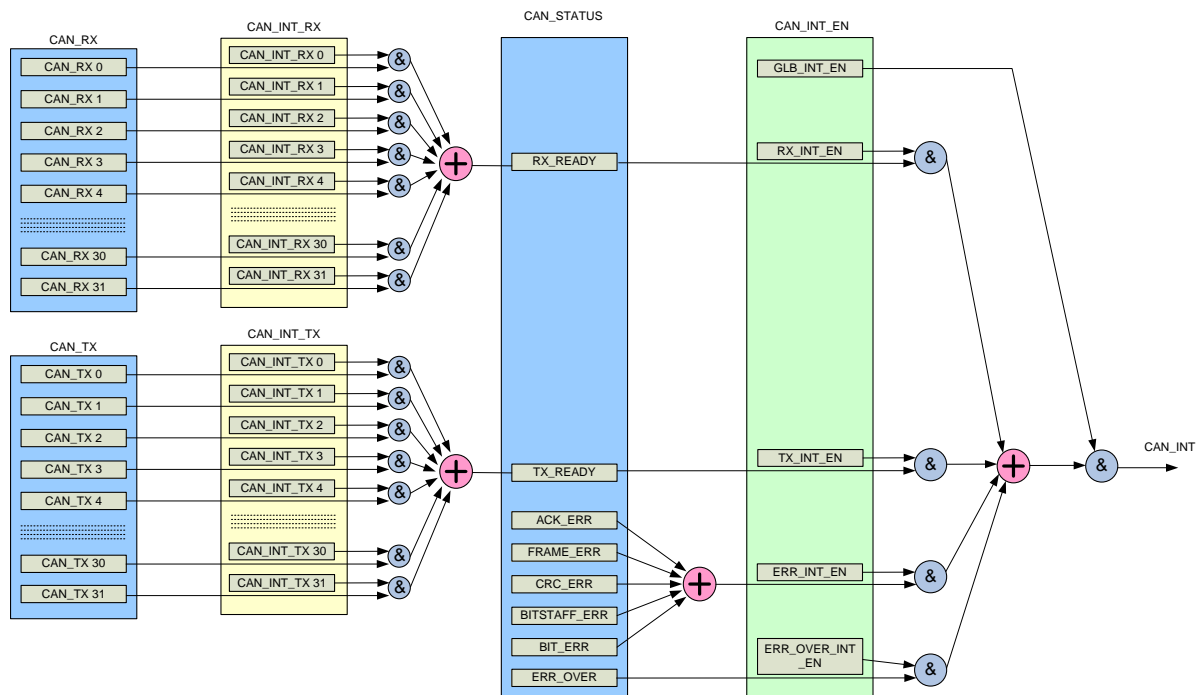


**Рисунок 61. Счетчики ошибок**

CAN CAN\_STATUS  
 ERROR\_OVER.  
 CAN\_OVER. CAN\_OVER

### 21.13 Прерывания

CAN  
 ( )  
 ( )  
 CAN  
 CAN\_INT\_TX/CAN\_INT\_RX.  
 ( CAN\_INT\_EN).



**Рисунок 62. Схема формирования прерывания блока CAN**

## 21.14 Описание регистров контроллера CAN

**Таблица 235 – Описание регистров контроллера CAN**

Базовый адрес	Название	Описание
0x4000 0000	MDR_CAN1	CAN1
0x4000 8000	MDR_CAN2	CAN2
0x00	CONTROL	MDR_CANx->CONTROL CAN
0x04	STATUS	MDR_CANx->STATUS CAN
0x08	BITTMNG	MDR_CANx->BITTMNG
0x10	INT_EN	MDR_CANx->INT_EN
0x1C	OVER	MDR_CANx->OVER
0x20	RXID	MDR_CANx->RXID ID
0x24	RXDLC	MDR_CANx->RXDLC DLC

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Базовый адрес	Название	Описание
0x28	RXDATA1	MDR_CANx->RXDATA1
0x2C	RXDATAH	MDR_CANx->RXDATAH
0x30	TXID	MDR_CANx->TXID ID
0x34	TXDLC	MDR_CANx->TXDLC DLC
0x38	DATAL	MDR_CANx->TXDATAL
0x3C	DATAH	MDR_CANx->TXDATAH
0x40	BUF_CON[0]	MDR_CANx->BUF_CON 01
	...	
0xBC	BUF_CON[31]	MDR_CANx->BUF_CON 32
0xC0	INT_RX	MDR_CANx->INT_RX
0xC4	RX	RX_FULL MDR_CANx->RX
0xC8	INT_TX	MDR_CANx->INT_TX
0xCC	TX	~TX_REQ MDR_CANx->TX
0x200	CAN_BUF[0].ID	MDR_CANx->CAN_BUF[x].ID ID 01
0x204	CAN_BUF[0].DLC	MDR_CANx->CAN_BUF[x].DLC DLC 01
0x208	CAN_BUF[0].DATAL	MDR_CANx->CAN_BUF[x].DATAL 01
0x20C	CAN_BUF[0].DATAH	MDR_CANx->CAN_BUF[x].DATAH 01



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Базовый адрес	Название	Описание
0x210	CAN_BUF[1].ID	MDR_CANx->CAN_BUF[x].ID ID 02
	...	
0x3FC	CAN_BUF[31].DATAH	MDR_CANx->CAN_BUF[x].DATAH 32
0x500	CAN_BUF_FILTER[0].MASK	MDR_CANx->CAN_BUF_FILTER[x].MASK 01
0x504	CAN_BUF_FILTER[0].FILTER	MDR_CANx->CAN_BUF_FILTER[x].FILTER 01
0x508	CAN_BUF_FILTER[1].MASK	MDR_CANx->CAN_BUF_FILTER[x].MASK 02
	...	
0x5FC	CAN_BUF_FILTER[31].FILTER	MDR_CANx->CAN_BUF_FILTER[x].FILTER 32

**21.14.1 MDR\_CANx->CONTROL**

**Таблица 236 – Регистр управления контроллером CONTROL**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>ROP</b>	<b>SAP</b>	<b>STM</b>	<b>ROM</b>	<b>CAN EN</b>

**Таблица 237 – Описание бит регистра CONTROL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	-	
4	ROP	(Receive own packets): 1 – ; 0 –
3	SAP	(Send ACK on own packets): 1 – ; 0 –
2	STM	(Self Test Mode): 1 – ; 0 –
1	ROM	« » (Read Only Mode): 1 – ; 0 –
0	CAN_EN	CAN: 1 – ; 0 –

**21.14.2 MDR\_CANx->STATUS**

**Таблица 238 – Регистр состояния контроллера STATUS**

<b>Номер</b>	7	6	5	4	3	2	1	0
<b>Доступ</b>	R/W	R/W	R/W	R/W	R/W	R/W	RO	RO
<b>Сброс</b>	0	0	0	0	0	0	0	0
	<b>ACK ERR</b>	<b>FRAME ERR</b>	<b>CRC ERR</b>	<b>BIT STUFF ERR</b>	<b>BIT ERR</b>	<b>ERROR OVER</b>	<b>TX READY</b>	<b>RX READY</b>

<b>Номер</b>	31...24	23...16	15...13	12	11	10...9	8
<b>Доступ</b>	RO	RO	U	RO	RO	RO	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	<b>TX ERR CNT [7:0]</b>	<b>RX ERR CNT [7:0]</b>	-	<b>TX ERR CNT8</b>	<b>RX ERR CNT8</b>	<b>ERR STATUS[1:0]</b>	<b>ID LOWER</b>

**Таблица 239 – Описание бит регистра STATUS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...24	TX ERR CNT [7:0]	TEC, [7:0]: TEC > 127, ERROR PASSIVE
23...16	RX ERR CNT [7:0]	REC, [7:0]: REC > 127, ERROR PASSIVE
15...13	-	
12	TX ERR CNT8	TEC, 8: 0 – TEC 255; 1 – TEC 255
11	RX ERR CNT8	REC, 8: 0 – REC 255; 1 – REC 255
10...9	ERR STATUS[1:0]	CAN: 00 – ERROR ACTIVE, ; 01 – ERROR PASSIVE, ; 1x – BUS OFF,
8	ID LOWER	«            » : 0 –            ; 1 –            ;
7	ACK ERR	: 0 –            ; 1 –            ;
6	FRAME ERR	: 0 –            ; 1 –            ;

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
5	CRC ERR	0 – ; 1 – :
4	BIT STUFF ERR	0 – ; 1 – :
3	BIT ERR	0 – ; 1 – :
2	ERROR OVER	TEC REC , ERROR_MAX: 0 – ERROR_MAX < TEC REC; 1 – ERROR_MAX TEC REC
1	TX READY	0 – ; 1 – :
0	RX READY	0 – ; 1 – :

**21.14.3 MDR\_CANx->BITTMNG**

**Таблица 240 – Регистр задания скорости работы BITTMNG**

<b>Номер</b>	31...28	27	26...25	24...22	21...19	18...16	15...0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	-	<b>SB</b>	<b>SJW</b> [1:0]	<b>SEG2</b> [2:0]	<b>SEG1</b> [2:0]	<b>PSEG</b> [2:0]	<b>BRP</b> [15:0]

**Таблица 241 – Описание бит регистра BITTMNG**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...28	-	
27	SB	: 0 – ; 1 –
26...25	SJW [1:0]	SJW: 11 = Synchronization jump width time = 4 x TQ 10 = Synchronization jump width time = 3 x TQ 01 = Synchronization jump width time = 2 x TQ 00 = Synchronization jump width time = 1 x TQ SJW – , CAN. SJW
24...22	SEG2 [2:0]	SEG2: 111 = Phase Segment 2 time = 8 x TQ 110 = Phase Segment 2 time = 7 x TQ 101 = Phase Segment 2 time = 6 x TQ 100 = Phase Segment 2 time = 5 x TQ 011 = Phase Segment 2 time = 4 x TQ 010 = Phase Segment 2 time = 3 x TQ 001 = Phase Segment 2 time = 2 x TQ 000 = Phase Segment 2 time = 1 x TQ SEG2 – ,
21...19	SEG1 [2:0]	SEG1: 111 = Phase Segment 1 time = 8 x TQ 110 = Phase Segment 1 time = 7 x TQ 101 = Phase Segment 1 time = 6 x TQ 100 = Phase Segment 1 time = 5 x TQ 011 = Phase Segment 1 time = 4 x TQ 010 = Phase Segment 1 time = 3 x TQ 001 = Phase Segment 1 time = 2 x TQ 000 = Phase Segment 1 time = 1 x TQ SEG1 – ,

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
18...16	PSEG[2:0]	<p align="center">PSEG</p> <p>111 = Propagation time = 8 x TQ                      110 = Propagation time = 7 x TQ                      101 = Propagation time = 6 x TQ                      100 = Propagation time = 5 x TQ                      011 = Propagation time = 4 x TQ                      010 = Propagation time = 3 x TQ                      001 = Propagation time = 2 x TQ                      000 = Propagation time = 1 x TQ</p> <p>PSEG -</p> <p align="center">CAN</p>
15...0	BRP [15:0]	<p align="center">:</p> <p><math>CLK = CANx\_CLK / (BRP + 1)</math>  <math>TQ(us) = 1 / CLK(MHz) = (BRP + 1) / CANx\_CLK(MHz)</math></p>

**21.14.4 MDR\_CANx->INT\_EN**

**Таблица 242 – Регистр разрешения прерываний INT\_EN**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>ERR OVER INT EN</b>	<b>ERR INT EN</b>	<b>TX INT EN</b>	<b>RX INT EN</b>	<b>GLB INT EN</b>

**Таблица 243 – Описание бит регистра INT\_EN**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	-	
4	ERR OVER INT EN	<p align="right">TEC    REC</p> <p align="center">ERROR_MAX:</p> <p>0 – ;</p> <p>1 –</p>
3	ERR INT EN	<p align="right">:</p> <p>0 – ;</p> <p>1 –</p>
2	TX INT EN	<p align="right">:</p> <p>0 – ;</p> <p>1 –</p>
1	RX INT EN	<p align="right">:</p> <p>0 – ;</p> <p>1 –</p>
0	GLB INT EN	<p align="right">CAN:</p> <p>0 – ;</p> <p>1 –</p>

### 21.14.5 MDR\_CANx->OVER

**Таблица 244 – Регистр границы счета ошибок OVER**

<b>Номер</b>	31...8	7...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>ERROR_MAX[7:0]</b>

**Таблица 245 – Описание бит регистра OVER**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	ERROR MAX [7:0]	TEC REC, ERROR_OVER

### 21.14.6 MDR\_CANx->BUF\_CON[x]

**Таблица 246 – Регистр управления буфером BUF\_CON[x]**

<b>Номер</b>	31...8	7	6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	-	<b>OVER WR</b>	<b>RX FULL</b>	<b>TX REQ</b>	<b>PRIO R 0</b>	<b>RTR EN</b>	<b>OVER EN</b>	<b>RX TXn</b>	<b>EN</b>

**Таблица 247 – Описание бит регистра BUF\_CON[x]**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7	OVER_WR	:
		0 – ; 1 –
6	RX_FULL	:
		0 – ; 1 –
5	TX_REQ	:
		0 – ; 1 –
4	PRIOR_0	:
		0 – ; 1 –
3	RTR_EN	RTR:
		0 – RTR; 1 – RTR
2	OVER_EN	:
		0 – ; 1 –



№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
1	RX_TXn	: 0 – ; 1 –
0	EN	: 0 – ; 1 –

#### 21.14.7 MDR\_CANx->INT\_RX

**Таблица 248 – Регистр разрешения прерываний от приемных буферов INT\_RX**

Номер	31...0
Доступ	R/W
Сброс	0
<b>CAN_INT_RX[31:0]</b>	

**Таблица 249 – Описание бит регистра INT\_RX**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	CAN INT RX[31:0]	: CAN_INT_RX[0] – CAN_INT_RX[1] –

#### 21.14.8 MDR\_CANx->RX

**Таблица 250 – Регистр RX флагов RX\_FULL от приемных буферов**

Номер	31...0
Доступ	RO
Сброс	0
<b>CAN_RX[31:0]</b>	

**Таблица 251 – Описание бит регистра RX**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	CAN RX[31:0]	: RX_FULL CAN_RX[0] – RX_FULL CAN_RX[1] – RX_FULL

### 21.14.9 MDR\_CANx->INT\_TX

**Таблица 252 – Регистр разрешения прерываний от передающих буферов INT\_TX**

<b>Номер</b>	31...0
<b>Доступ</b>	R/W
<b>Сброс</b>	0
	<b>CAN_INT_TX[31:0]</b>

**Таблица 253 – Описание бит регистра INT\_TX**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	CAN INT TX[31:0]	: CAN_INT_TX[0] – CAN_INT_TX[1] –

### 21.14.10 MDR\_CANx->TX

**Таблица 254 – Регистр TX флагов ~TX\_REQ от передающих буферов**

<b>Номер</b>	31...0
<b>Доступ</b>	RO
<b>Сброс</b>	0
	<b>CAN_TX[31:0]</b>

**Таблица 255 – Описание бит TX**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	CAN TX[31:0]	: ~TX_REQ CAN_TX[0] – ~TX_REQ CAN_TX[1] – ~TX_REQ

### 21.14.11 MDR\_CANx->RXID

**MDR\_CANx->TXID**

**MDR\_CANx->CAN\_BUF[x].ID**

**MDR\_CANx->CAN\_BUF\_FILTER[x].MASK**

**MDR\_CANx->CAN\_BUF\_FILTER[x].FILTER**

**Таблица 256 – Регистры RXID, TXID и CAN\_BUF[x].ID идентификаторов**

<b>Номер</b>	31...29	28...18	17...0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	<b>SID</b> [10:0]	<b>EID</b> [17:0]

**Таблица 257 – Описание бит регистров RXID, TXID и CAN\_BUF[x].ID**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...29	-	
28...18	SID [10:0]	SID.  CAN.
17...0	EID [17:0]	EID.  CAN.

**21.14.12 MDR\_CANx->RXDLC  
MDR\_CANx->TXDLC  
MDR\_CANx->CAN\_BUF[x].DLC**

**Таблица 258 – Регистры RXDLC, TXDLC и CAN\_BUF[x].DLC сообщения**

Номер	31...13	12	11	10	9	8	7...4	3...0
Доступ	U	R/W	R/W	R/W	R/W	R/W	U	R/W
Сброс	0	0	0	0	0	0	0	0
	-	IDE	SRR	R0	R1	RTR	-	DLC [3:0]

**Таблица 259 – Описание бит регистров RXDLC, TXDLC и CAN\_BUF[x].DLC**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...13	-	
12	IDE	IDE.  , 0 – ; 1 –
11	SRR	SRR,  "1"
10	R0	R0.  "0"
9	R1	R1,  "1"
8	RTR	RTR,  0 – ; 1 – .  ,
7...4	-	

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
3...0	DLC[3:0]	DLC, : 0000 – 0001 – 1 0010 – 2 0011 – 3 0100 – 4 0101 – 5 0110 – 6 0111 – 7 1000 – 8 1xxx – 8

**21.14.13 MDR\_CANx->RXDATAL  
MDR\_CANx->TXDATAL  
MDR\_CANx->CAN\_BUF[x].DATAL**

**Таблица 260 – Регистры RXDATAL, TXDATAL и CAN\_BUF[x].DATAL данных сообщения**

Номер	31...24	23...16	15...8	7...0
Доступ	R/W	R/W	R/W	R/W
Сброс	0	0	0	0
	<b>DB3[7:0]</b>	<b>DB2[7:0]</b>	<b>DB1[7:0]</b>	<b>DB0[7:0]</b>

**Таблица 261 – Описание бит регистров RXDATAL, TXDATAL и CAN\_BUF[x].DATAL**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...24	DB3[7:0]	DB3. ,
23...16	DB2[7:0]	DB2. ,
15...8	DB1[7:0]	DB1. ,
7...0	DB0[7:0]	DB0. ,

**21.14.14 MDR\_CANx->RXDATAH  
MDR\_CANx->TXDATAH  
MDR\_CANx->CAN\_BUF[x].DATAH**

**Таблица 262 – Регистры RXDATAH, TXDATAH и CAN\_BUF[x].DATAH данных сообщения**

Номер	31...24	23...16	15...8	7...0
Доступ	R/W	R/W	R/W	R/W
Сброс	0	0	0	0
	<b>DB7[7:0]</b>	<b>DB6[7:0]</b>	<b>DB5[7:0]</b>	<b>DB4[7:0]</b>

**Таблица 263 – Описание бит регистров RXDATAH, TXDATAH и CAN\_BUF[x].DATAH**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...24	DB7[7:0]	DB7. ,
23...16	DB6[7:0]	DB6. ,
15...8	DB5[7:0]	DB5. ,
7...0	DB4[7:0]	DB4. ,

## 22 Таймеры общего назначения MDR\_TIMERx

16-  
16-  
(  
16-  
4- / 16- , 16-  
/ 4  
« »  
DMA.  
:  
• 16- ;  
• 16-  
• 16- ( )  
/ DMA.  
• 16- ( ),  
/ DMA ;  
• :  
- ;  
- ;  
- ( ) ;  
- ;  
- .

### 22.1 Функционирование

Fdts  
DMA

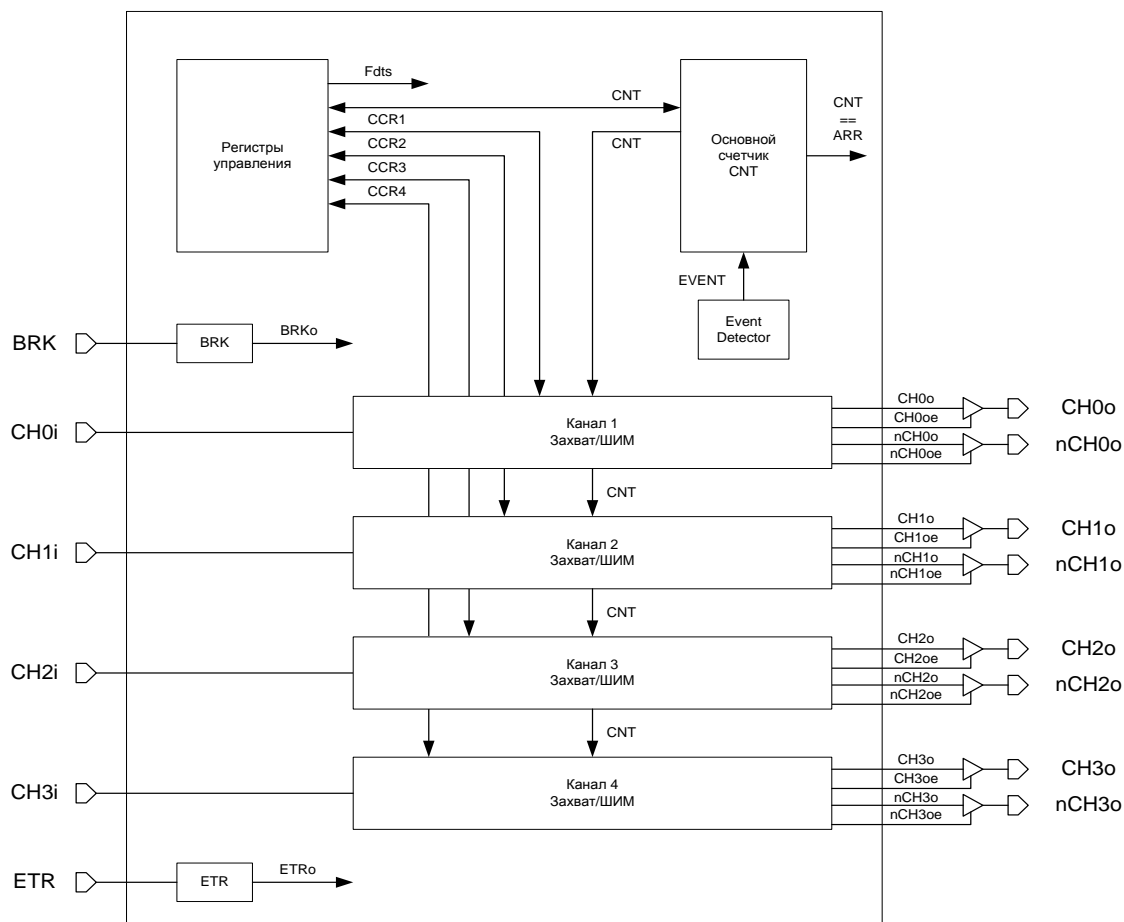
DMA.

#### 22.1.1 Структурная схема

63

16- CNT,

/



**Рисунок 63. Структурная схема таймера**

— ;  
 — ;  
 — ;  
 — ;

**22.1.2 Соответствие выводов таймера выводам микроконтроллера**

BRK	—	TMR <sub>x</sub> _BLK
ETR	—	TMR <sub>x</sub> _ETR
CH0i	—	TMR <sub>x</sub> _CH1
CH1i	—	TMR <sub>x</sub> _CH2
CH2i	—	TMR <sub>x</sub> _CH3
CH3i	—	TMR <sub>x</sub> _CH4
CH0o	—	TMR <sub>x</sub> _CH1
CH1o	—	TMR <sub>x</sub> _CH2
CH2o	—	TMR <sub>x</sub> _CH3
CH3o	—	TMR <sub>x</sub> _CH4
nCH0o	—	TMR <sub>x</sub> _CH1N
nCH1o	—	TMR <sub>x</sub> _CH2N
nCH2o	—	TMR <sub>x</sub> _CH3N
nCH3o	—	TMR <sub>x</sub> _CH4N

### 22.1.3 Инициализация таймера

```

        .
        «
        ».
        ( 14 1, 15 2, 16 3
        PER_CLOCK). TIM_CLOCK TIM_CLKEN,
        HCLK
    
```

### 22.1.4 Режим таймера

```

        16-
        ,
        16-
        .
        ,
        (
        ,
        ).
        :
        • (TIMx_CNT);
        • (TIMx_PSC);
        • (TIMx_ARR).
        CNT TIM_CLK,
        , TxCHi
        :
        - -TIMx_CNT;
        - -TIMx_PSG,
        - CLK= TIMx_CLK/(PSG+1);
        - TIMx_ARR;
        - TIMx_CNTRL:
        - EVENT_SEL;
        - CNT_MODE ( 00 01
        , 10 11 );
        - DIR;
        - CNT_EN.
        DMA,
    
```

### 22.2 Режимы счета

```

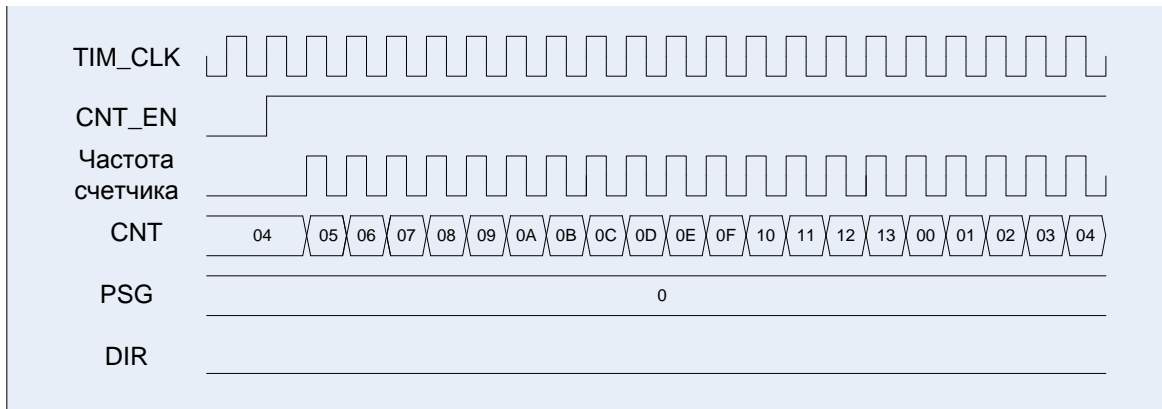
        : CNT_MODE = 00, DIR = 0 ( : 0 0 13,
        0 04)
    
```

```

MDR_TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
MDR_TIMERx->CNT = 0x00000004; //Начальное значение счетчика
MDR_TIMERx->PSG = 0x00000000; //Предделитель частоты
MDR_TIMERx->ARR = 0x00000013; //Основание счета
    
```



```
//Разрешение работы таймера.
MDR_TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK.
```



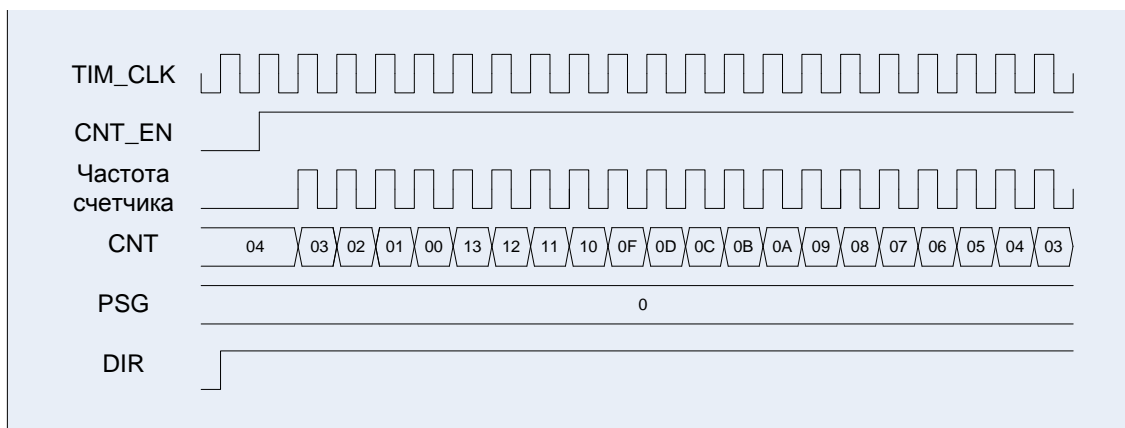
**Рисунок 64. Диаграммы работы таймера, счет вверх**

: CNT\_MODE = 00, DIR = 1 ( : 0 13 0,

0 04)

```
MDR_TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
MDR_TIMERx->CNT = 0x00000004; //Начальное значение счетчика
MDR_TIMERx->PSG = 0x00000000; //Предделитель частоты
MDR_TIMERx->ARR = 0x00000013; //Основание счета
```

```
//Разрешение работы таймера.
MDR_TIMERx->CNTRL = 0x00000009; //Счет вниз по TIM_CLK.
```



**Рисунок 65. Диаграммы работы таймера, счет вниз**

/ : CNT\_MODE = 01, DIR = 0

```
MDR_TIMERx ->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
MDR_TIMERx ->CNT = 0x00000004; //Начальное значение счетчика
MDR_TIMERx ->PSG = 0x00000000; //Предделитель частоты
MDR_TIMERx ->ARR = 0x00000013; //Основание счета
```

```
//Разрешение работы таймера.
MDR_TIMERx ->CNTRL = 0x00000041; //Счет вверх/вниз по TIM_CLK.
```

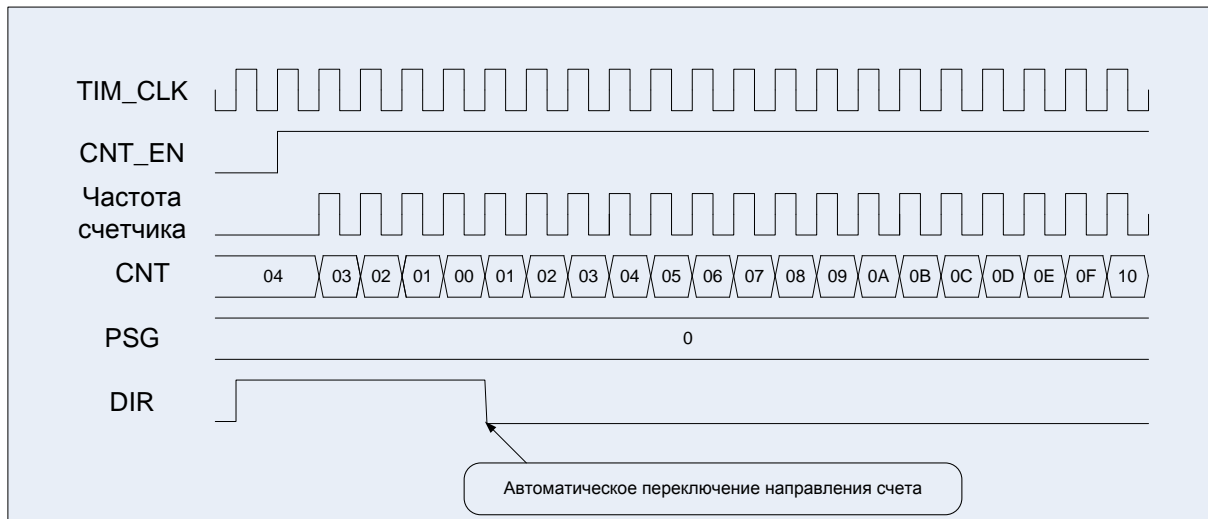


**Рисунок 66. Диаграммы работы таймера, счет вверх/вниз, сначала вверх**

/ : CNT\_MODE = 01, DIR = 1

```
MDR_TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
MDR_TIMERx->CNT = 0x00000004; //Начальное значение счетчика
MDR_TIMERx->PSG = 0x00000000; //Предделитель частоты
MDR_TIMERx->ARR = 0x00000013; //Основание счета

//Разрешение работы таймера.
MDR_TIMERx->CNTRL = 0x00000049; //Счет вверх/вниз по TIM_CLK.
```



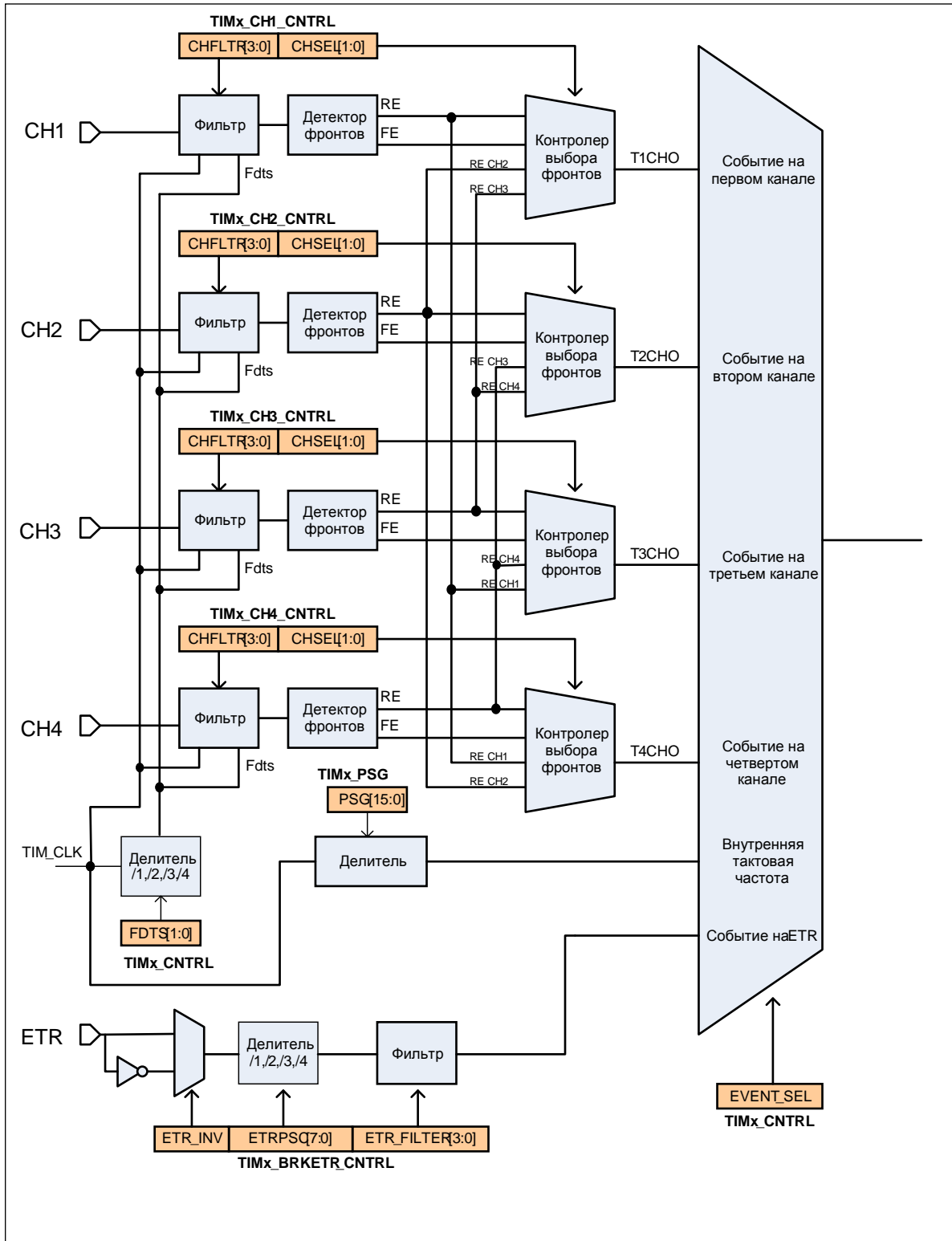
**Рисунок 67. Диаграммы работы таймера, счет вверх/вниз, сначала вниз**

### 22.3 Источник событий для счета

```

:
• (TIM_CLK);
• (CNT==ARR);
• 1: TxCHO;
• 2: ETR;

```



**Рисунок 68. Структурная схема формирования события для счета**

### 22.3.1 Внутренний тактовый сигнал (TIM\_CLK)

CNT\_MODE = 0x, EVENT\_SEL = 0000.

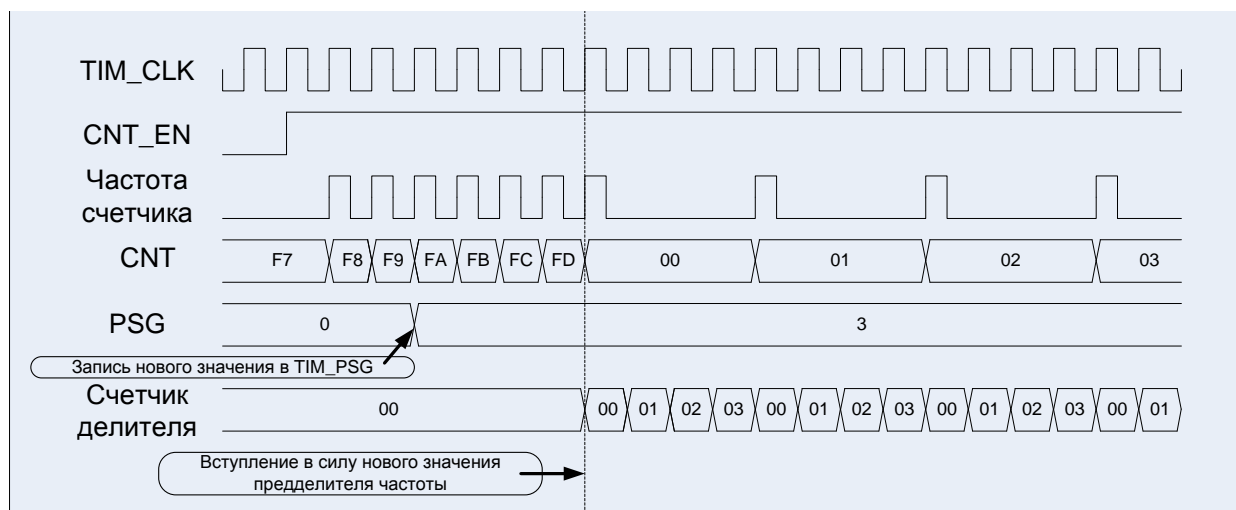
CNTRL.

CNT, PSG ARR

CNT = ARR

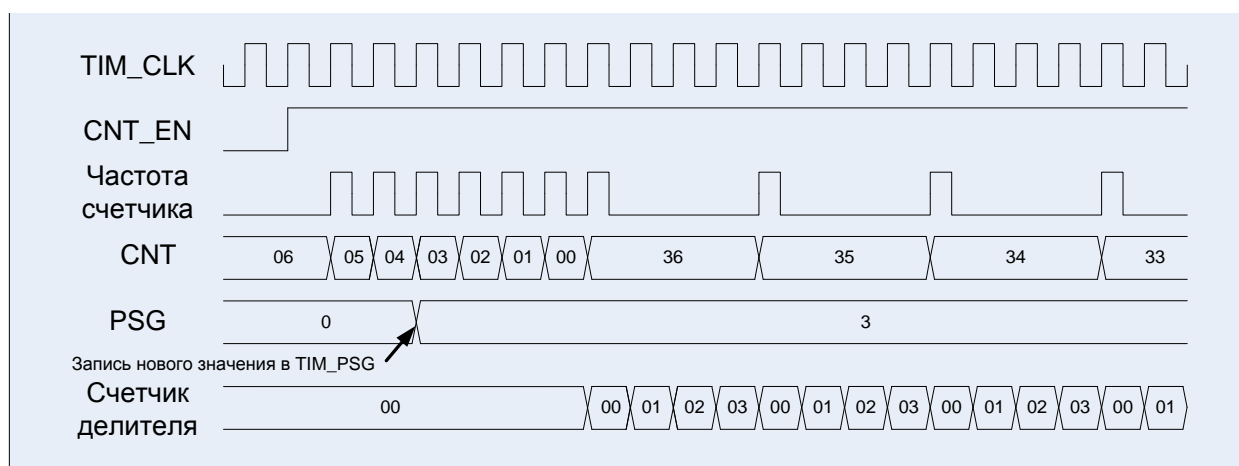
CNT = 0,

(ARR) ARRB\_EN = 1 ( CNTRL).  
 TIM\_CLK  
 1 : DIR  
 = 0 - ( . 69. ) : ), DIR = 1  
 - ( . 70. : ).  
 CNT\_MODE = 00, EVENT\_SEL = 0000, DIR = 0



**Рисунок 69. Диаграммы работы счетчика: счет вверх**

CNT\_MODE = 00, EVENT\_SEL = 0000, DIR = 1



**Рисунок 70. Диаграммы работы счетчика: счет вниз**

CNT\_MODE = 00, DIR, CNT\_MODE  
 = 01, / DIR ( . 71).  
 CNT\_MODE = 01, EVENT\_SEL = 0000, DIR = 1

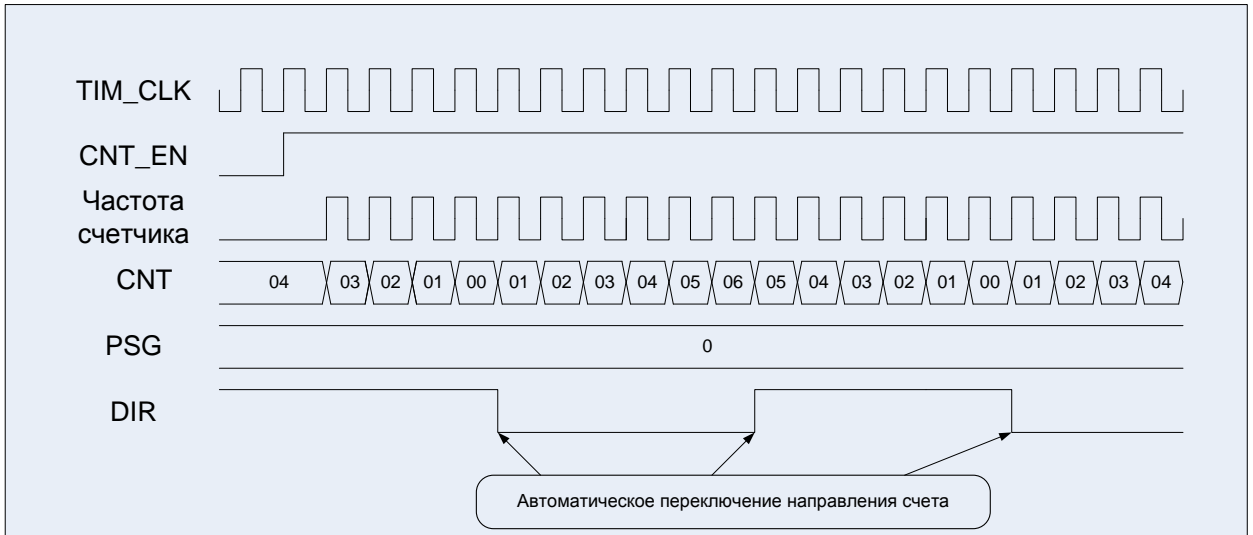


Рисунок 71. Диаграммы работы счетчика: счет вниз/вверх

### 22.3.2 События в других счетчиках (CNT==ARR в таймере X)

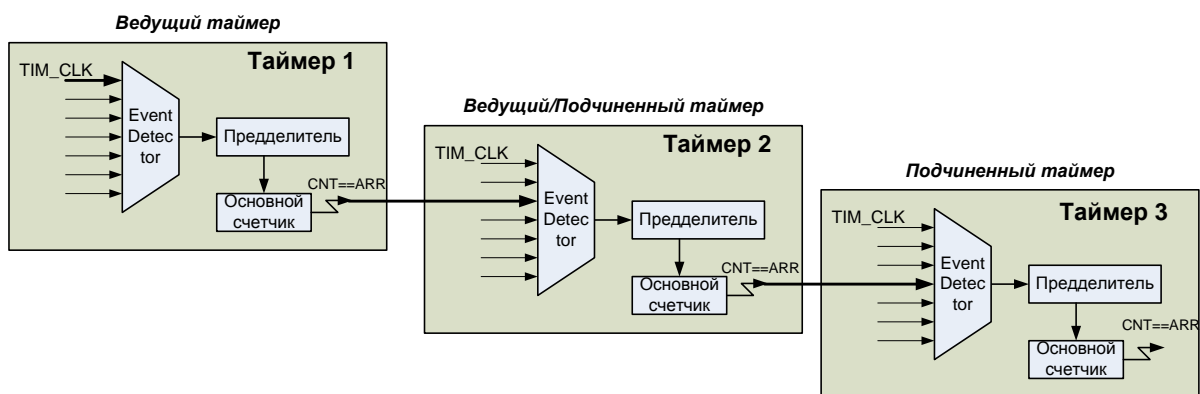
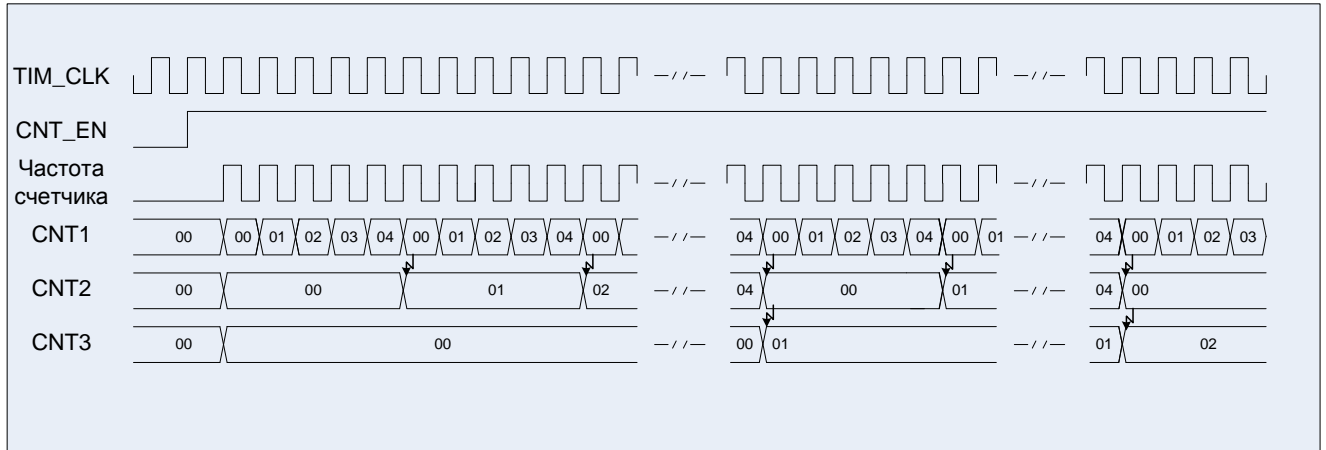


Рисунок 72. Пример каскадного соединения таймеров

DIR\_1, DIR\_2, DIR\_3 = 0;

EVENT\_SEL\_1 = 0000, EVENT\_SEL\_2 = 0001, EVENT\_SEL\_3 = 0010;

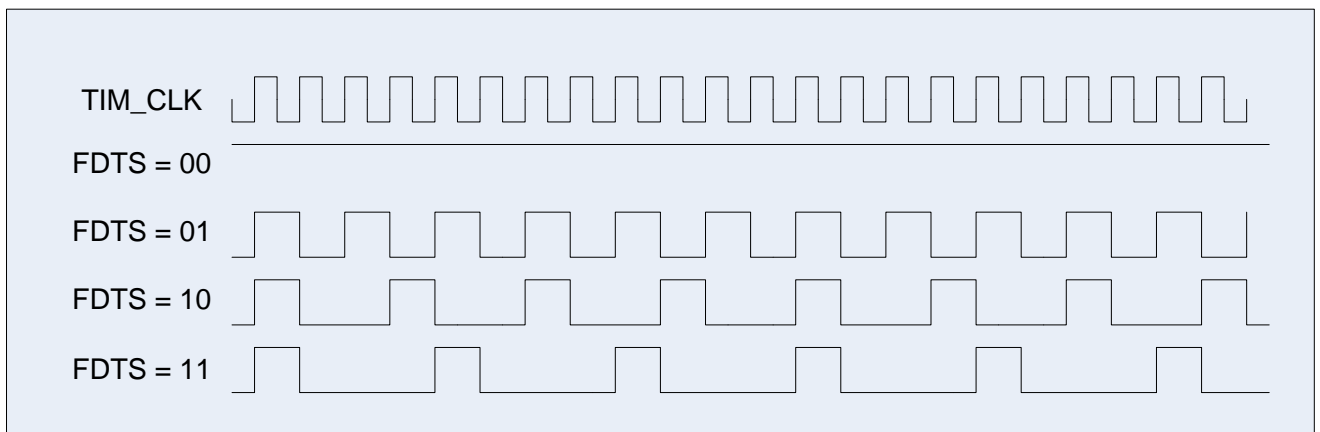
CNT\_MODE\_1, CNT\_MODE\_2, CNT\_MODE\_3 = 00;



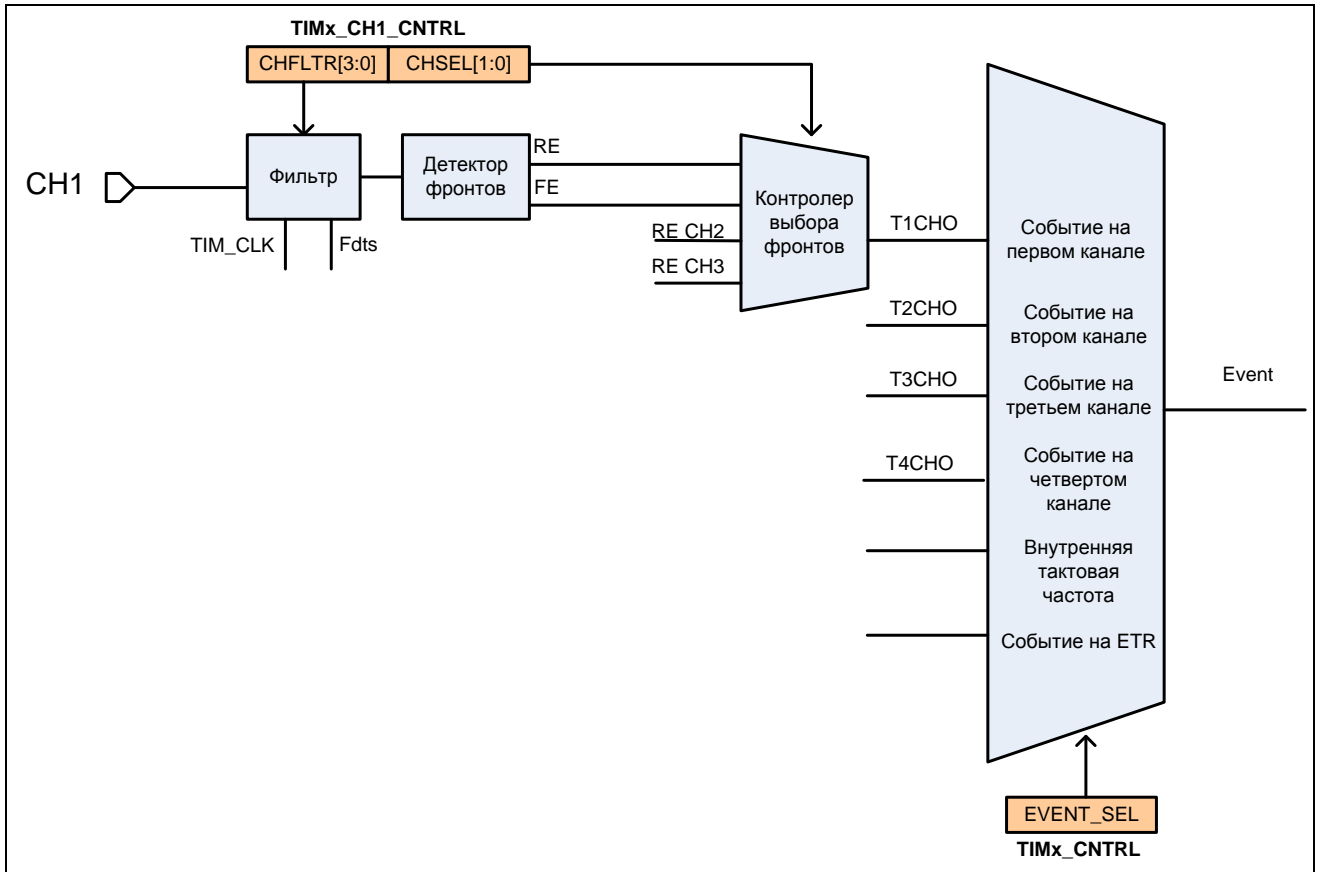
**Рисунок 73. Диаграммы работы трех таймеров в каскаде**

**22.3.3 Внешний тактовый сигнал «Режим 1». События на линиях TxCHO данного счетчика**

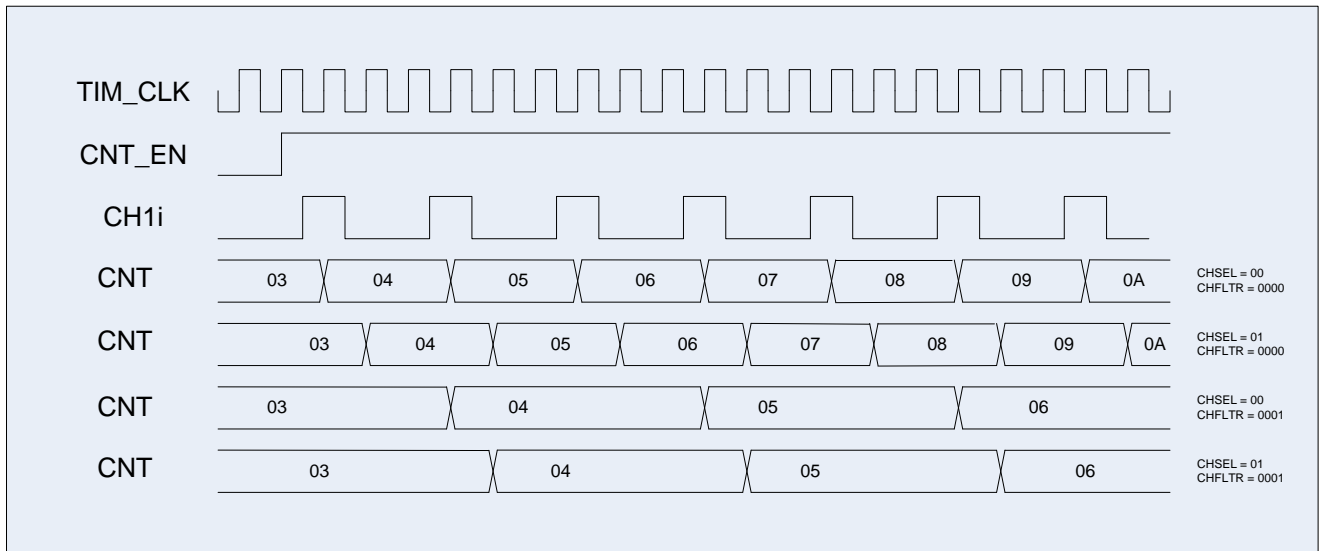
, EVENT\_SEL = 01xx CNTRL.  
 ( . 74). CHSEL[1:0] MDR\_TIMERx-  
 >CHy\_CNTRL . CHSEL[1:0]  
 , .  
 , TIM\_CLK,  
 1, 2, 4, 8 TIM\_CLK,  
 CNTRL TIM\_CLK FDS.  
 FDS.



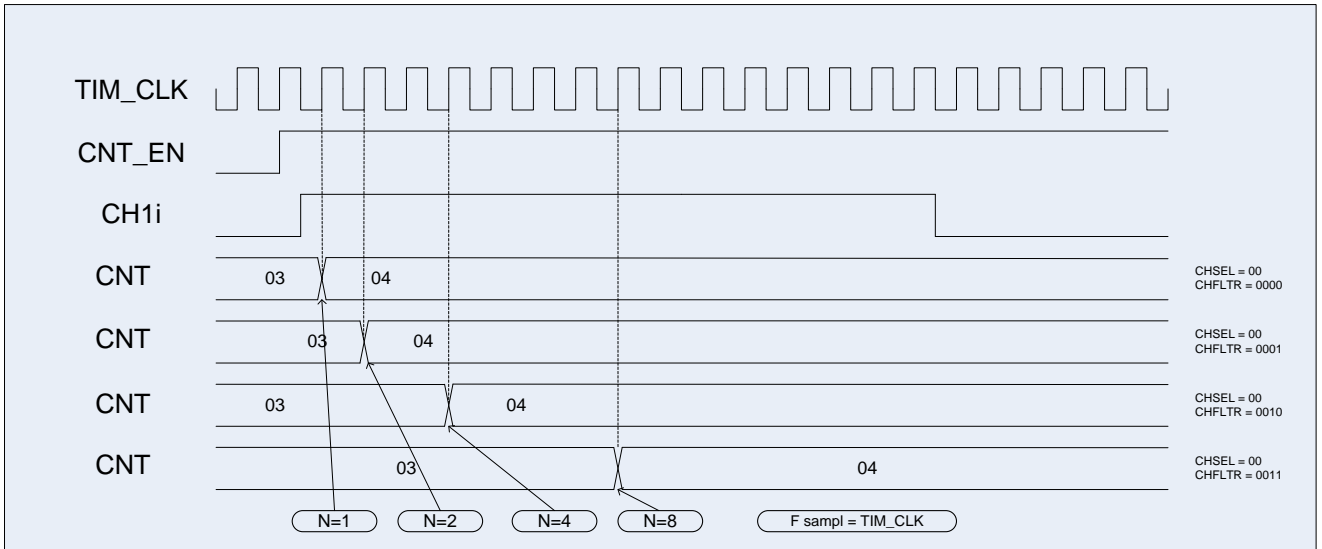
**Рисунок 74. Диаграммы возможных частот семплирования данных (FDS)**



**Рисунок 75. Тактирование с входа первого канала**



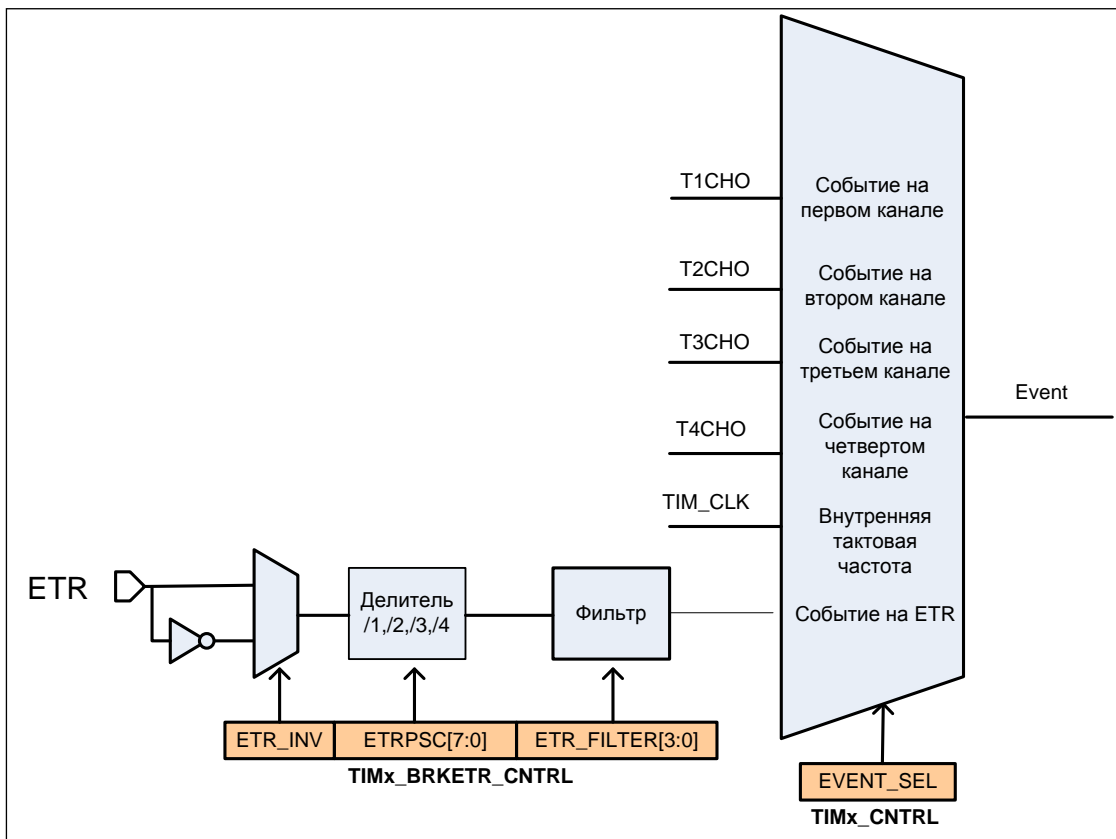
**Рисунок 76. Диаграмма внешнего тактирования с разными вариантами фильтра**



**Рисунок 77. Диаграмма внешнего тактирования с разными вариантами фильтра**

**22.3.4 Внешний тактовый сигнал «Режим 2». События на входе ETR данного счетчика**

BRKETR\_CNTRL, EVENT\_SEL = 1000, CNTRL. 2, 4, 8 (ETRPSC)



**Рисунок 78. Схема тактирования сигналом со входа ETR**



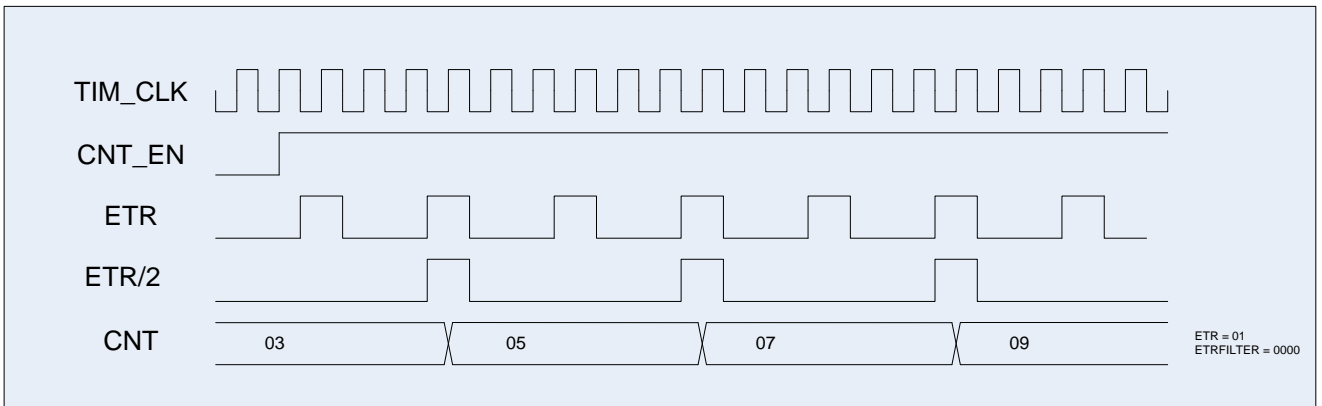


Рисунок 79. Диаграмма тактирования сигналом со входа ETR

## 22.4 Режим захвата

80.

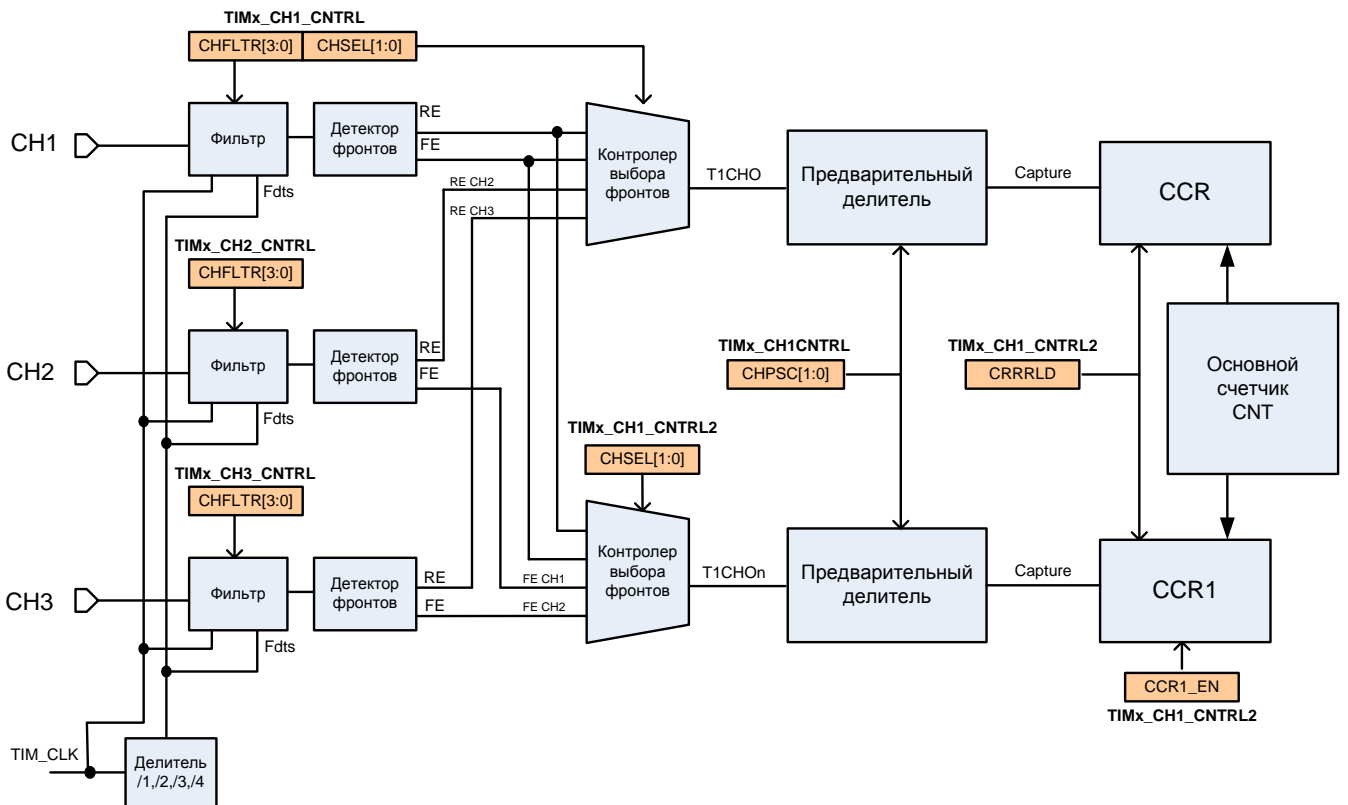
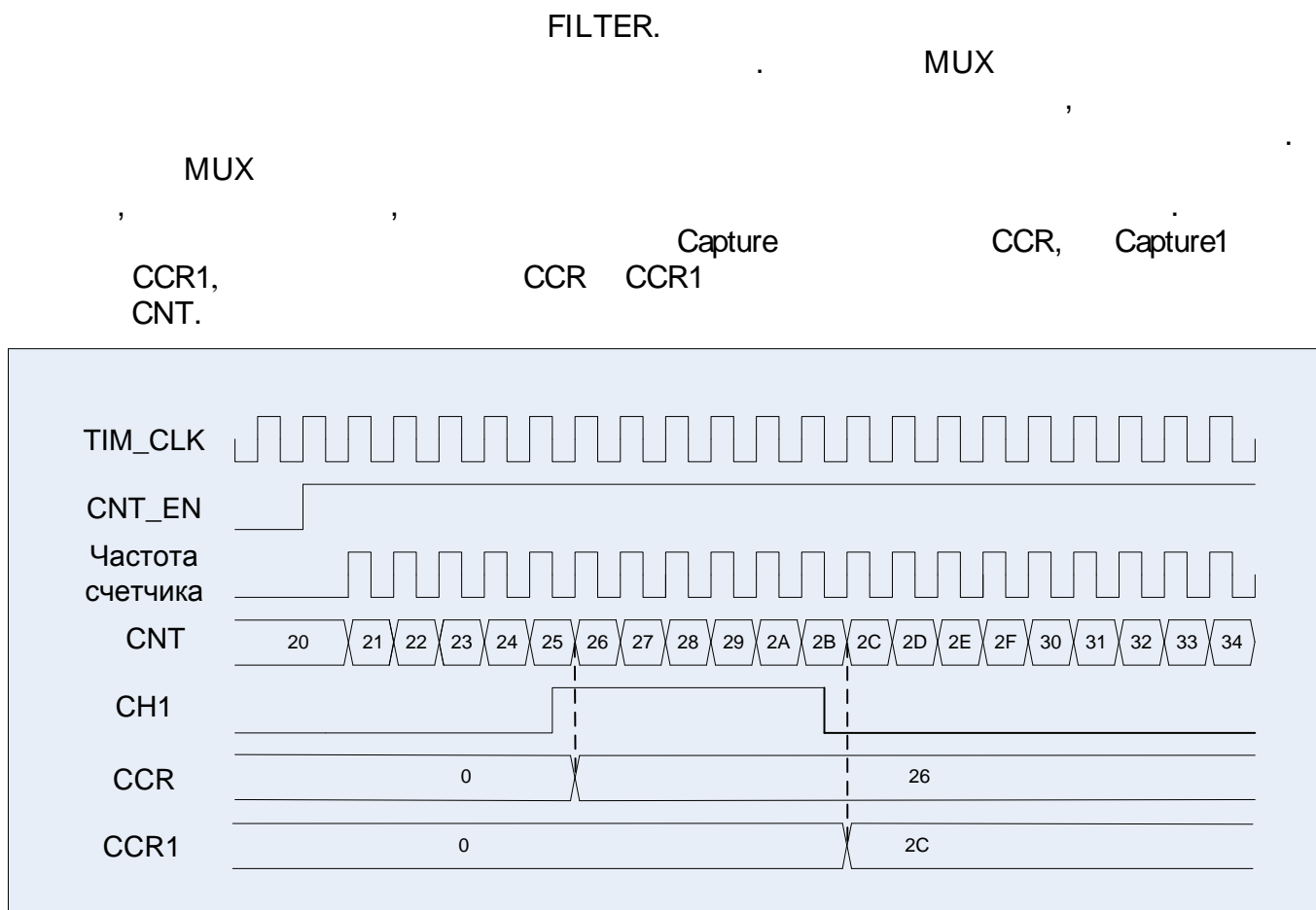


Рисунок 80. Структурная схема блока захвата на примере канала 1

CHxi                    CHy\_CNTRL                    “1”                    CAPnPWM.  
Fdts,                    TIM\_CLK.



**Рисунок 81. Диаграмма захвата события со входа первого канала**



## 22.5 Режим ШИМ

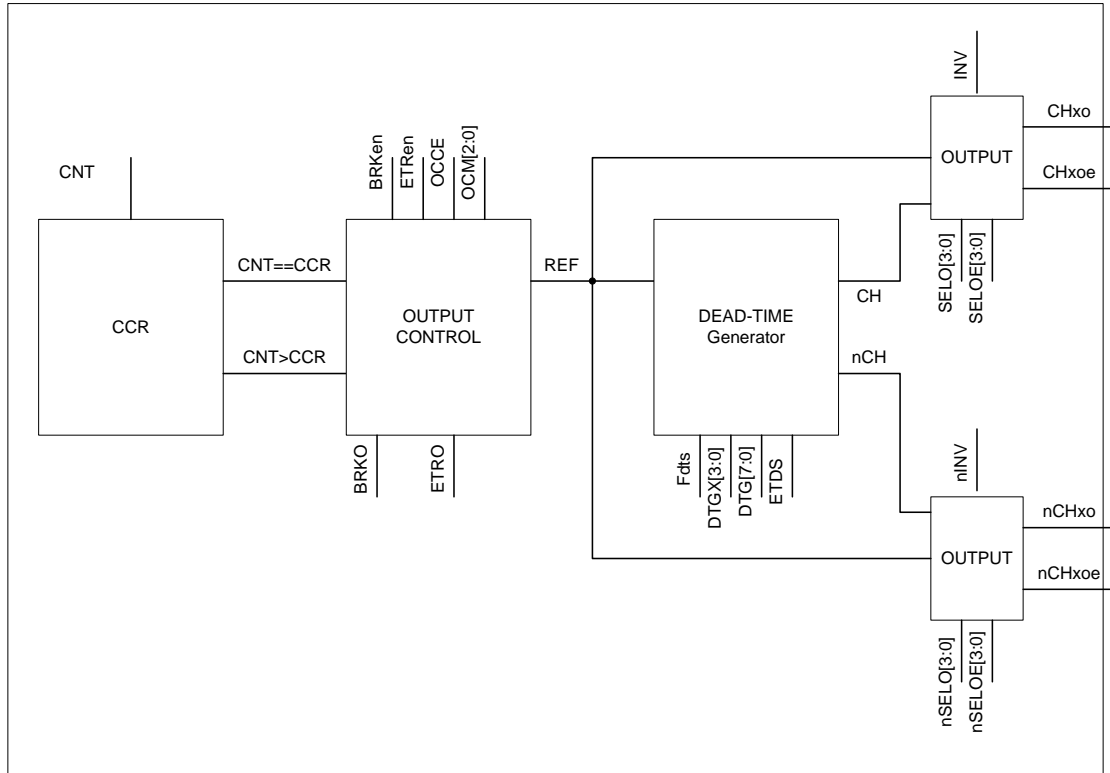


Рисунок 82. Структурная схема блока формирования ШИМ

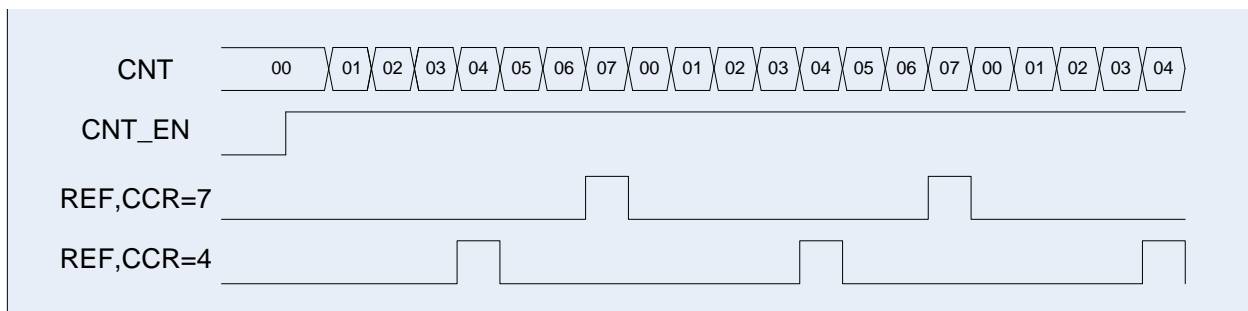
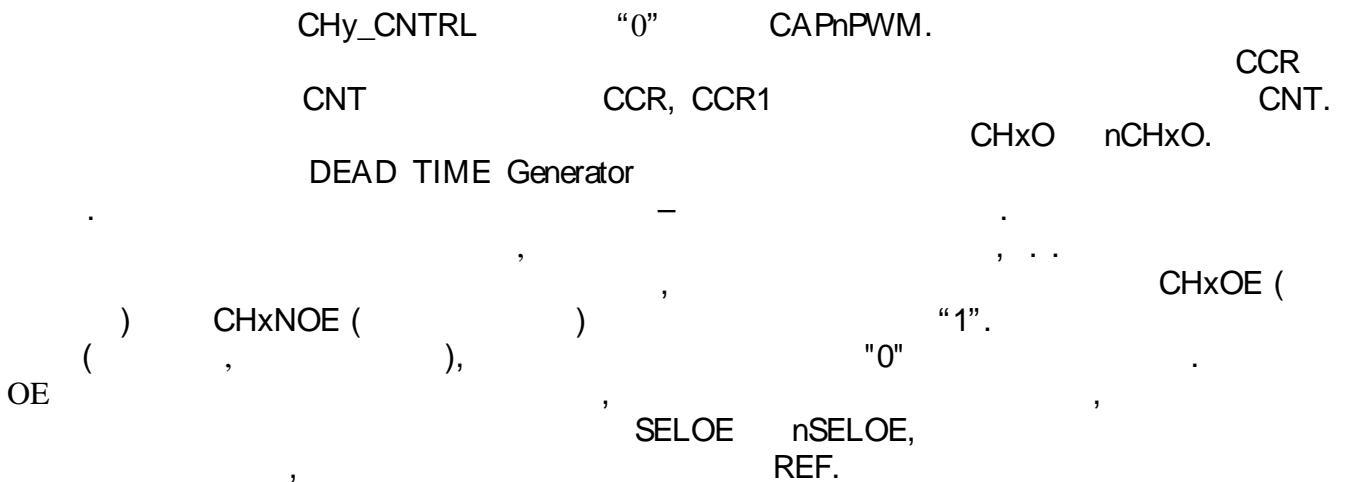
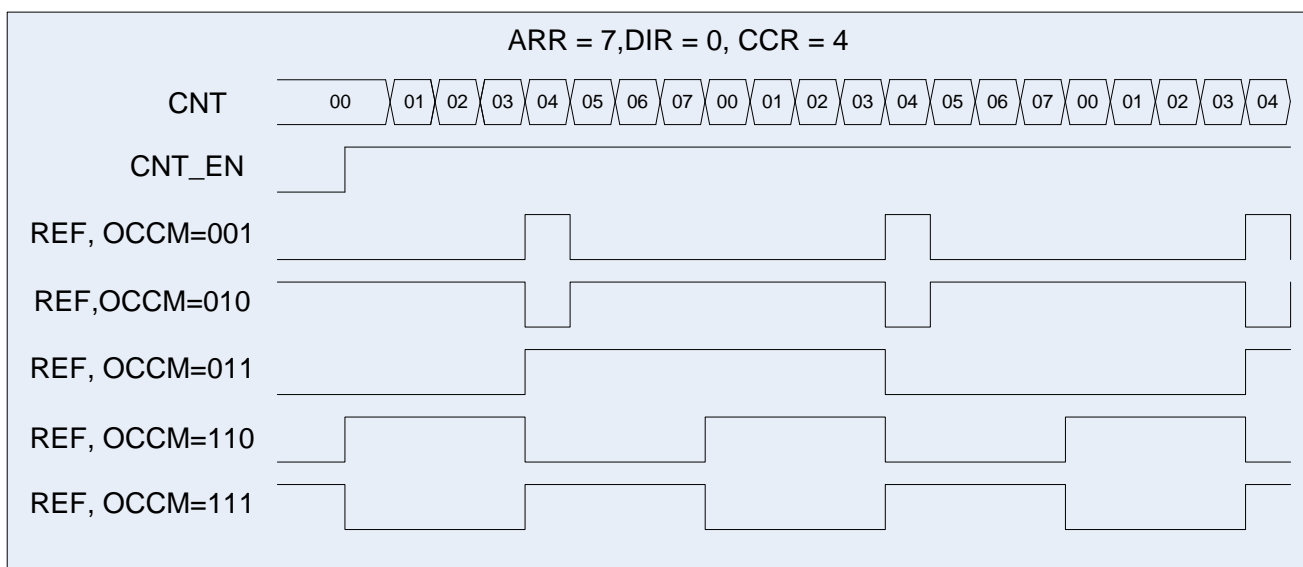
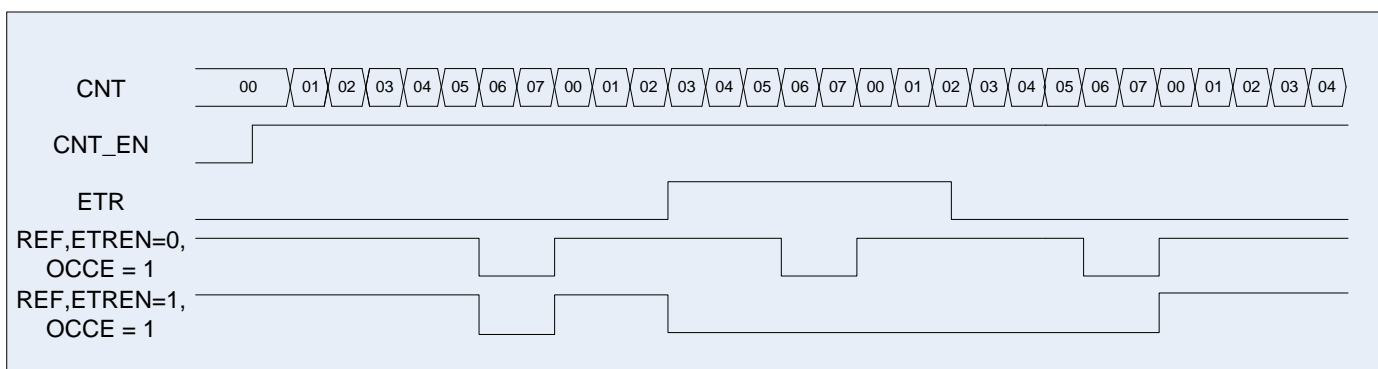


Рисунок 83. Диаграмма работы схемы в режиме ШИМ, CCR1\_EN=0

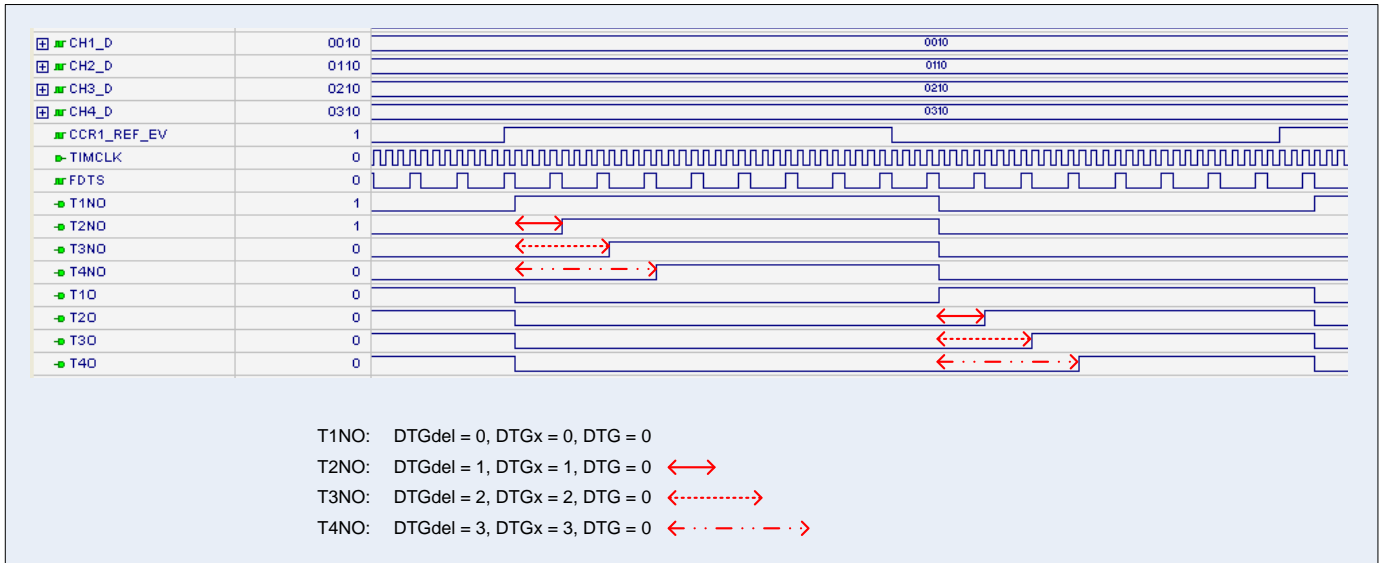


**Рисунок 84. Диаграмма работы схемы в режиме ШИМ, CCR1\_EN = 0**

REF, PCLK, BRK., ETR,

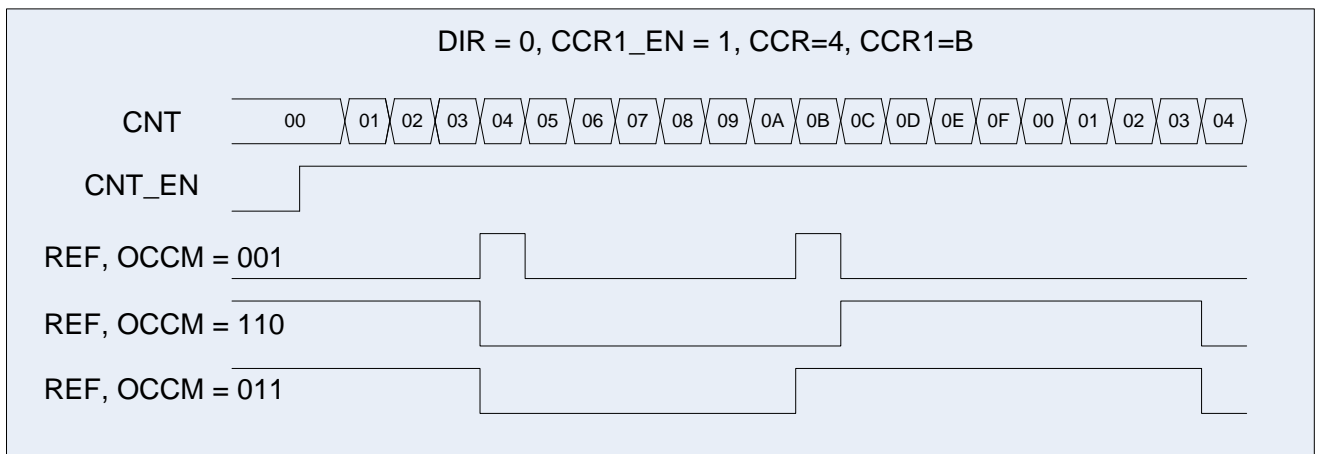


**Рисунок 85. Диаграмма работы схемы в режиме ШИМ, CCR1\_EN = 0**



**Рисунок 86. Диаграмма работы схемы DTG**

CCR1\_EN = 1, CNT  
 CCR CCR1, CHy\_CNTRL OCCM)  
 REF (



**Рисунок 87. Диаграмма работы схемы в режиме ШИМ, CCR1\_EN = 1**

CCR CCR1, RRRLD,  
 CCR1 CCR CNT = 0, WR\_CMPL.

## 22.6 Примеры

### 22.6.1 Обычный счетчик

```
MDR_RST_CLK->PER_CLOCK = 0xFFFFFFFF;  
MDR_RST_CLK->TIM_CLOCK = 0x07000000;  
MDR_TIMERx->CNTRL = 0x00000000;  
//Настраиваем работу основного счетчика  
MDR_TIMERx->CNT = 0x00000000; //Начальное значение счетчика  
MDR_TIMERx->PSG = 0x00000000; //Предделитель частоты  
MDR_TIMERx->ARR = 0x0000000F; //Основание счета  
  
MDR_TIMERx->IE = 0x00000002; //Разрешение генерировать  
прерывание при CNT=ARR  
  
MDR_TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK.  
Разрешение работы таймера.
```

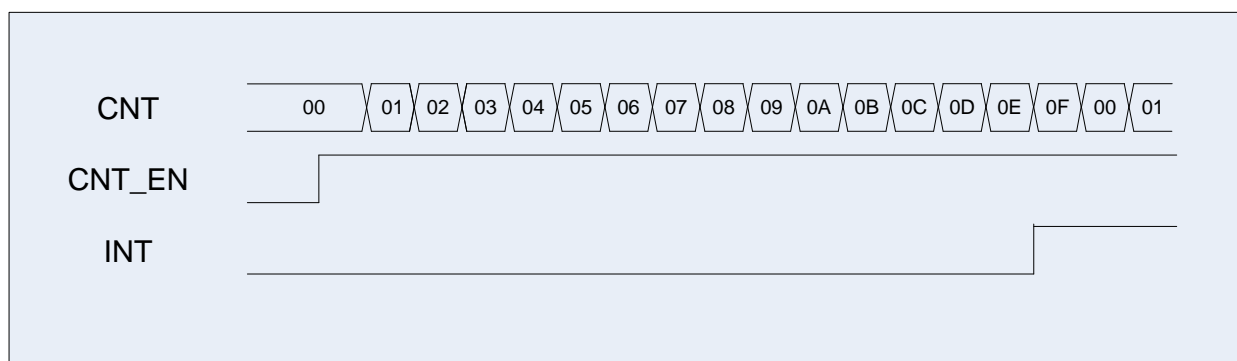


Рисунок 88. Режим обычного счетчика

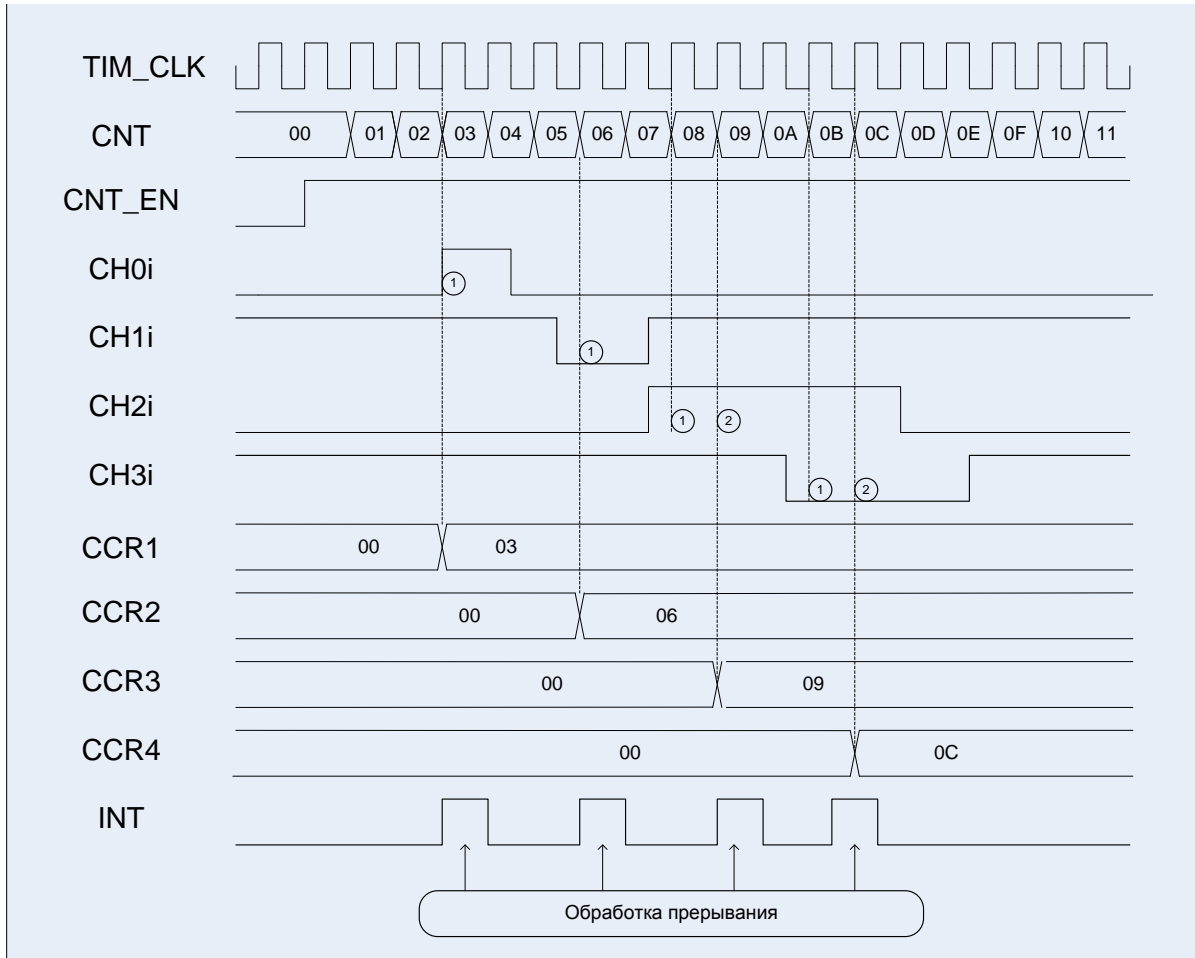
## 22.6.2 Режим захвата

```
MDR_RST_CLK->PER_CLOCK = 0xFFFFFFFF; //Разрешение тактовой
частоты таймеров
MDR_RST_CLK->TIM_CLOCK = 0x07000000; //Включение тактовой
частоты таймеров
MDR_TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
MDR_TIMERx->CNT = 0x00000000; //Начальное значение счетчика
MDR_TIMERx->PSG = 0x00000000; //Предделитель частоты
MDR_TIMERx->ARR = 0x000000FF; //Основание счета

MDR_TIMERx->IE = 0x00001E00; //Разрешение генерировать
прерывание
//по переднему фронту на выходе CAP по
всем каналам
//Режим работы каналов - захват
MDR_TIMERx->CHy_CNTRL[0] = 0x00008000;
MDR_TIMERx->CHy_CNTRL[1] = 0x00008002;
MDR_TIMERx->CHy_CNTRL[2] = 0x00008001;
MDR_TIMERx->CHy_CNTRL[3] = 0x00008003;

//Режим работы выхода канала - канал на выход не работает
MDR_TIMERx->CHy_CNTRL1[0]= 0x00000000;
MDR_TIMERx->CHy_CNTRL1[1]= 0x00000000;
MDR_TIMERx->CHy_CNTRL1[2]= 0x00000000;
MDR_TIMERx->CHy_CNTRL1[3]= 0x00000000;

MDR_TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK.
Разрешение работы таймера
```



**Рисунок 89. Диаграммы примера работы в режиме захвата**

### 22.6.3 Режим ШИМ

```

MDR_RST_CLK->PER_CLOCK = 0xFFFFFFFF; //Разрешение тактовой
частоты таймеров
MDR_RST_CLK->TIM_CLOCK = 0x07000000; //Включение тактовой
частоты таймеров
MDR_TIMERx->CNTRL = 0x00000000; //Режим инициализации таймера
//Настраиваем работу основного счетчика
MDR_TIMERx->CNT = 0x00000000; //Начальное значение счетчика
MDR_TIMERx->PSG = 0x00000000; //Предделитель частоты
MDR_TIMERx->ARR = 0x00000010; //Основание счета

MDR_TIMERx->IE = 0x000001E0; //Разрешение генерировать
прерывание
//по переднему фронту на выходе REF по
всем каналам
//Режим работы каналов - ШИМ
MDR_TIMERx->CHy_CNTRL[0] = 0x00000200;
MDR_TIMERx->CHy_CNTRL[1] = 0x00000200;
MDR_TIMERx->CHy_CNTRL[2] = 0x00000400;
MDR_TIMERx->CHy_CNTRL[3] = 0x00000600;

//Режим работы выхода канала - канал на выход не работает
    
```

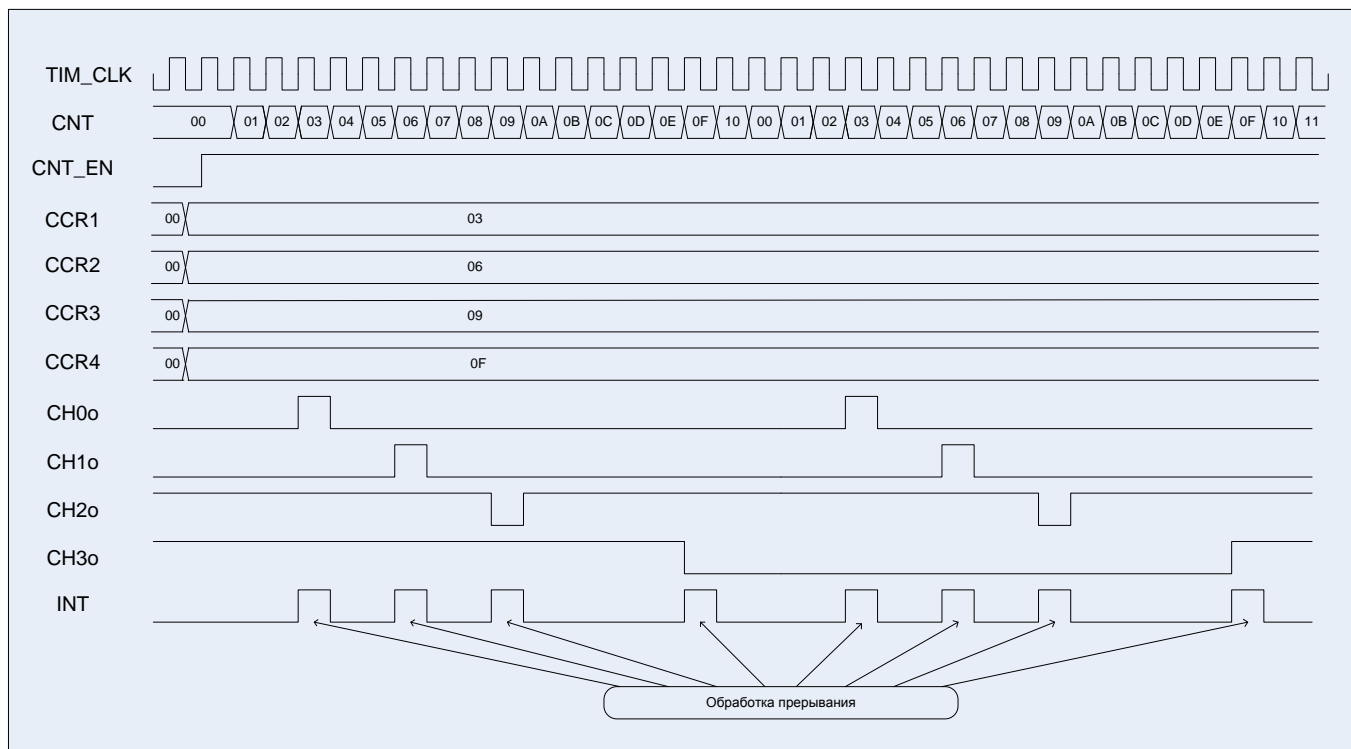


**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

```
MDR_TIMERx->CHy_CNTRL1[0]= 0x00000099;  
MDR_TIMERx->CHy_CNTRL1[1]= 0x00000099;  
MDR_TIMERx->CHy_CNTRL1[2]= 0x00000099;  
MDR_TIMERx->CHy_CNTRL1[3]= 0x00000099;
```

*//Разрешение работы таймера.*

```
MDR_TIMERx->CNTRL = 0x00000001; //Счет вверх по TIM_CLK.
```



**Рисунок 90. Диаграммы примера работы в режиме ШИМ**



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Адрес	Название	Описание
0x50	BRKETR_CNTRL[15:0]	MDR_TIMERx->BRKETR_CNTRL BRK ETR
0x54	STATUS[15:0]	MDR_TIMERx->STATUS
0x58	IE[15:0]	MDR_TIMERx->IE
0x5C	DMA_RE[15:0]	MDR_TIMERx->DMA_RE DMA
0x60	CH1_CNTRL2[15:0]	MDR_TIMERx->CHy_CNTRL2 2 1
0x64	CH2_CNTRL2[15:0]	MDR_TIMERx->CHy_CNTRL2 2 2
0x68	CH3_CNTRL2[15:0]	MDR_TIMERx->CHy_CNTRL2 2 3
0x6C	CH4_CNTRL2[15:0]	MDR_TIMERx->CHy_CNTRL2 2 4
0x70	CCR11[15:0]	MDR_TIMERx->CCRy1 1, 1
0x74	CCR21[15:0]	MDR_TIMERx->CCRy1 1, 2
0x78	CCR31[15:0]	MDR_TIMERx->CCRy1 1, 3
0x7C	CCR41[15:0]	MDR_TIMERx->CCRy1 1, 4

### 22.7.1 MDR\_TIMERx->CNT

Таблица 265 – Основной счетчик таймера CNT

Номер	31...16	15... 0
Доступ	U	R/W
Сброс	0	0
	-	CNT[15:0]

Таблица 266 – Описание бит регистра CNT

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...0	CNT[15:0]	

### 22.7.2 MDR\_TIMERx->PSG

Таблица 267 – Делитель частоты при счете основного счетчика PSG

Номер	31...16	15... 0
Доступ	U	R/W
Сброс	0	0
	-	PSG[15:0]

Таблица 268 – Описание бит регистра PSG

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...0	PSG[15:0]	:  CLK = TIM_CLK/(PSG+1)

### 22.7.3 MDR\_TIMERx->ARR

Таблица 269 – Основание счета основного счетчика ARR

Номер	31...16	15... 0
Доступ	U	R/W
Сброс	0	0
	-	ARR[15:0]

Таблица 270 – Описание бит регистра ARR

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...0	ARR[15:0]	:  CNT = [0...ARR]

### 22.7.4 MDR\_TIMERx->CNTRL

**Таблица 271 – Регистр управления основного счетчика CNTRL**

<b>Номер</b>	31..12	11..8	7...6	5...4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0
	-	EVENT SEL[3:0]	CNT MODE[1:0]	FDTS [1:0]	DIR	WR CMPL	ARRB EN	CNT EN

**Таблица 272 – Описание бит регистра CNTRL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...11	-	
11...8	EVENT_SEL [3:0]	: 0000 – TIM_CLK; 0001 – CNT == ARR 1; 0010 – CNT == ARR 2; 0011 – CNT == ARR 3; 0100 – « 1»; 0101 – « 1»;; 0110 – « 1»; 0111 – « 1»; 1000 – ETR « 2»
7...6	CNT_MODE [1:0]	: 00 – DIR=0 ( PSG = 0) DIR=1 ( PSG = 0); 01 – / DIR PSG = 0; 10 – DIR=0 DIR=1; 11 – / DIR CNT_MODE[1:0] = 00 CNT_MODE[1:0] = 01 , EVENT_SEL = 0000; CNT_MODE[1:0] = 10 CNT_MODE[1:0] = 11 , EVENT_SEL != 0000
5...4	FDTS[1:0]	FDTS: 00 – TIM_CLK; 01 – TIM_CLK; 10 – TIM_CLK; 11 – TIM_CLK
3	DIR	: 0 – , 0 ARR; 1 – , ARR 0
2	WR_CMPL	, CNT, PSG ARR: 0 – ; 1 –
1	ARRB_EN	ARR 0 – ARR ARR; 1 – ARR CNT

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
0	CNT_EN	: 0 – ; 1 –

### 22.7.5 MDR\_TIMERx->CCRy

**Таблица 273 – Регистр сравнения/захвата для ‘у’ канала таймера CCRy**

Номер	31...16	15... 0
Доступ	U	R/W
Сброс	0	0
	-	CCR[15:0]

**Таблица 274 – Описание бит регистра CCRy**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...0	CCR[15:0]	CCR, с CNT, ,

### 22.7.6 MDR\_TIMERx->CCRy1

**Таблица 275 – Регистр сравнения/захвата для ‘у’ канала таймера CCRy1**

Номер	31...16	15...0
Доступ	U	R/W
Сброс	0	0
	-	CCR1[15:0]

**Таблица 276 – Описание бит регистра CCRy1**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...0	CCR1[15:0]	CCR1, с CNT, ,

### 22.7.7 MDR\_TIMERx->CHy\_CNTRL

**Таблица 277 – Регистр управления для ‘у’ канала таймера CHy\_CNTRL**

<b>Номер</b>	31...16	15	14	13	12	11...9	8	7...6	5...4	3...0
<b>Доступ</b>	U	R/W	RO	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0	0
	-	CAP nPWM	WR CMPL	ETREN	BRKEN	OCCM [2:0]	OCCE	CHPSC [1:0]	CHSEL [1:0]	CHFLTR [3:0]

**Таблица 278 – Описание бит регистра CHy\_CNTRL**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15	CAP nPWM	: 1 – ; 0 –
14	WR CMPL	, CCR6 1 – ; 0 –
13	ETREN	ETR: 0 – ; 1 –
12	BRKEN	BRK: 0 – ; 1 –
11...9	OCCM[2:0]	REF : CCR1_EN = 0: 000 – 0 001 – 1, CNT = CCR; 010 – 0, CNT = CCR; 011 – REF, CNT = CCR; 100 – 0; 101 – 1; 110 – 1, DIR= 0 ( ), CNT<CCR, 0; 0, DIR= 1 ( ), CNT>CCR, 1; 111 – 0, DIR= 0 ( ), CNT<CCR, 1; 1, DIR= 1 ( ), CNT>CCR, 0. CCR1_EN = 1: 000 – 0; 001 – 1, CNT = CCR CNT = CCR1 010 – 0, CNT = CCR CNT = CCR1; 011 – REF, CNT =CCR CNT =CCR1; 100 – 0; 101 – 1; 110 – 1, DIR = 0 ( ), CCR< CNT< CCR1, 0; 0, DIR = 1 ( ), CCR < CNT < CCR1, 1; 111 – 0, DIR = 0 ( ), CCR< CNT < CCR1, 1; 1, DIR = 1 ( ), CCR< CNT< CCR1, 0 , CCR < CCR1

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
8	OCCE	<p align="center">ETR:</p> 0 – ETR; 1 – ETR
7...6	CHPSC[1:0]	: 00 – ; 01 – /2; 10 – /4; 11 – /8
5...4	CHSEL[1:0]	<p align="center">CHxi ( MDR_TIMERx-&gt;CNT) CCR:</p> 00 – CHxi; 01 – CHxi; 10 – ;  2 ; 3 ; 4 ; 1 ; 11 – ;  3 ; 4 ; 1 ; 2
3...0	CHFLTR[3:0]	: 0000 – 1 TIM_CLK; 0001 – 2 TIM_CLK; 0010 – 4 TIM_CLK; 0011 – 8 TIM_CLK; 0100 – 6 FDTS/2; 0101 – 8 FDTS/2; 0110 – 6 FDTS/4; 0111 – 8 FDTS/4; 1000 – 6 FDTS/8; 1001 – 8 FDTS/8; 1010 – 5 FDTS/16; 1011 – 6 FDTS/16; 1100 – 8 FDTS/16; 1101 – 5 FDTS/32; 1110 – 6 FDTS/32; 1111 – 8 FDTS/32



**22.7.8 MDR\_TIMERx->CHy\_CNTRL1**

**Таблица 279 – Регистр управления 1 для ‘у’ канала таймера CHy\_CNTRL1**

<b>Номер</b>	31...13	12	11...10	9...8	7...5	4	3...2	1...0
<b>Доступ</b>	U	R/W	R/W	R/W	U	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0
	-	NINV	NSELO [1:0]	NSELOE [1:0]	-	INV	SELO [1:0]	SELOE [1:0]

**Таблица 280 – Описание бит регистра CHy\_CNTRL1**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...13	-	
12	NINV	nCHy: 0 – ; 1 –
11..10	NSELO[1:0]	nCHy: 00 – 0, ; 01 – 1, ; 10 – REF; 11 – DTG
9...8	NSELOE[1:0]	nCHy 00 – nCHyOE 0, ; 01 – nCHyOE 1, ; 10 – nCHyOE REF, REF = 0 11 – nCHyOE REF = 1 ; DTG, nCHyOE = 0 , nCHyOE = 1
7...5	-	
4	INV	CH : 0 – ; 1 –
3...2	SELO[1:0]	CH : 00 – 0, ; 01 – 1, ; 10 – REF; 11 – DTG
1...0	SELOE[1:0]	CH : 00 – CHyOE 0, ; 01 – CHyOE 1, ; 10 – CHyOE REF, REF = 0 11 – CHyOE REF = 1 ; DTG, CHyOE = 0 , CHyOE = 1

**22.7.9 MDR\_TIMERx->CHy\_CNTRL2**

**Таблица 281 – Регистр управления 2 для ‘у’ канала таймера CHy\_CNTRL2**

<b>Номер</b>	31... 5	4	3	2	1...0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	00
	-	<b>EV_DELAY</b>	<b>CCRRLD</b>	<b>CCR1_EN</b>	<b>CHSEL [1:0]</b>

**Таблица 282 – Описание бит регистра CHy\_CNTRL2**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...4	-	
4	EV_DELAY	0 – CCR CCR1; 1 – CCR CCR1;
3	CCRRLD	CCR CCR1: 0 – ; 1 – CNT = 0
2	CCR1_EN	CCR1: 0 – CCR1 ; 1 – CCR1
1...0	CHSEL1[1:0]	CHxi ( MDR_TIMERx->CNT) CCR1: 00 – CHxi; 01 – CHxi; 10 – : - 2 ; - 3 ; - 4 ; - 1 . : 11 – : - 3 ; - 4 ; - 1 ; - 2

**22.7.10 MDR\_TIMERx->CHy\_DTG**

**Таблица 283 – Регистр CHy\_DTG управления DTG**

<b>Номер</b>	31...16	15...8	7...5	4	3...0
<b>Доступ</b>	U	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	-	<b>DTG[7:0]</b>	-	<b>EDTS</b>	<b>DTGx[3:0]</b>

**Таблица 284 – Описание бит регистра CHy\_DTG**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16	-	
15...8	DTG[7:0]	$DTG_{del} = DTG * (DTG_{x+1})$
7...5	-	
4	EDTS	DTG: 0 – TIM_CLK; 1 – FDTS
3...0	DTGx [3:0]	DTGx

**22.7.11 MDR\_TIMERx->BRKETR\_CNTRL**

**Таблица 285 – Регистр BRKETR\_CNTRL управления входом BRK и ETR**

<b>Номер</b>	31...8	7...4	3...2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	-	<b>ETR FILTER [3:0]</b>	<b>ETR PSC [1:0]</b>	<b>ETR INV</b>	<b>BRK INV</b>

**Таблица 286 – Описание бит регистра BRKETR\_CNTRL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...4	ETR FILTER[3:0]	ETR. : 0000 – 1 TIM_CLK; 0001 – 2 TIM_CLK; 0010 – 4 TIM_CLK; 0011 – 8 TIM_CLK; 0100 – 6 FDTS/2; 0101 – 8 FDTS/2; 0110 – 6 FDTS/4; 0111 – 8 FDTS/4; 1000 – 6 FDTS/8; 1001 – 8 FDTS/8; 1010 – 5 FDTS/16; 1011 – 6 FDTS/165; 1100 – 8 FDTS/16; 1101 – 5 FDTS/32; 1110 – 6 FDTS/32; 1111 – 8 FDTS/32
3...2	ETRPSC[1:0]	: 00 – ; 01 - /2; 10 - /4; 11 - /8
1	ETR INV	ETR: 0 – ; 1 –
0	BRK INV	BRK: 0 – ; 1 –

### 22.7.12 MDR\_TIMERx->STATUS

**Таблица 287 – Регистр статуса таймера STATUS**

<b>Номер</b>	31...17	16...13	12...9	8...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	-	<b>CCR CAP1 EVENT [3:0]</b>	<b>CCR REF EVENT [3:0]</b>	<b>CCR CAP EVENT [3:0]</b>	<b>BRK EVENT</b>	<b>ETR FE EVENT</b>	<b>ETR RE EVENT</b>	<b>CNT ARR EVENT</b>	<b>CNT ZERO EVENT</b>

**Таблица 288 – Описание бит регистра STATUS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...17	-	
16...13	CCR CAP1 EVENT[3:0]	CHxi : 0 – ; 1 – . 0. , 0 – . 3 –
12...9	CCR REF EVENT[3:0]	REF 0 – ; 1 – . 0. , 0 – . 3 –
8...5	CCR CAP EVENT[3:0]	CHxi : 0 – ; 1 – . 0. , 0 – . 3 –
4	BRK EVENT	BRK, PCLK: 0 – BRK = 0; 1 – BRK = 1. 0, 0 BRK
3	ETR FE EVENT	ETR: 0 – ; 1 – . 0. , .
2	ETR RE EVENT	ETR: 0 – ; 1 – . 0. , .

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91Н4, К1986BE91Н4, 1986BE94Н4, К1986BE94Н4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
1	CNT ARR EVENT	CNT ARR: 0 – ; 1 – . 0. , . CNT ARR ,
0	CNT ZERO EVENT	CNT : 0 – ; 1 – . 0. , . CNT ,

### 22.7.13 MDR\_TIMERх->IE

**Таблица 289 – Регистр разрешения прерывания таймера IE**

<b>Номер</b>	31...17	16...13	12...9	8...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	-	CCR CAP1 EVENT IE [3:0]	CCR REF EVENT IE [3:0]	CCR CAP EVENT IE [3:0]	BRK EVENT IE	ETR FE EVENT IE	ETR RE EVENT IE	CNT ARR EVENT IE	CNT ZERO EVENT IE

**Таблица 290 – Описание бит регистра IE**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...17	-	
16...13	CCR CAP1 EVENT IE [3:0]	CHxi (CCR1): 0 – ; 1 – . 0 – . 3 –
12...9	CCR REF EVENT IE[3:0]	REF : 0 – ; 1 – . 0 – . 3 –
8...5	CCR CAP EVENT IE [3:0]	CHxi (CCR): 0 – ; 1 – . 0 – . 3 –
4	BRK EVENT IE	BRK, PCLK: 0 – ; 1 –
3	ETR FE EVENT IE	ETR: 0 – ; 1 –
2	ETR RE EVENT IE	ETR: 0 – ; 1 –

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
1	CNT ARR EVENT IE	ARR: 0 – ; 1 –
0	CNT ZERO EVENT IE	: 0 – ; 1 –

### 22.7.14 MDR\_TIMERx->DMA\_RE

**Таблица 291 – Регистр DMA\_RE разрешения запросов DMA от прерываний таймера**

Номер	31...17	16...13	12...9	8...5	4	3	2	1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0	0	0
	-	CCR CAP1 EVENT RE [3:0]	CCR REF EVENT RE [3:0]	CCR CAP EVENT RE [3:0]	BRK EVENT RE	ETR FE EVENT RE	ETR RE EVENT RE	CNT ARR EVENT RE	CNT ZERO EVENT RE

**Таблица 292 – Описание бит регистра DMA\_RE**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...17	-	
16...13	CCR CAP1 EVENT RE [3:0]	DMA ( CHxi CCR1): 0 – DMA; 1 – DMA . 0 – . 3 –
12...9	CCR REF EVENT RE[3:0]	DMA : REF 0 – DMA; 1 – DMA . 0 – . 3 –
8...5	CCR CAP EVENT RE [3:0]	DMA ( CHxi CCR): 0 – DMA; 1 – DMA . 0 – . 3 –

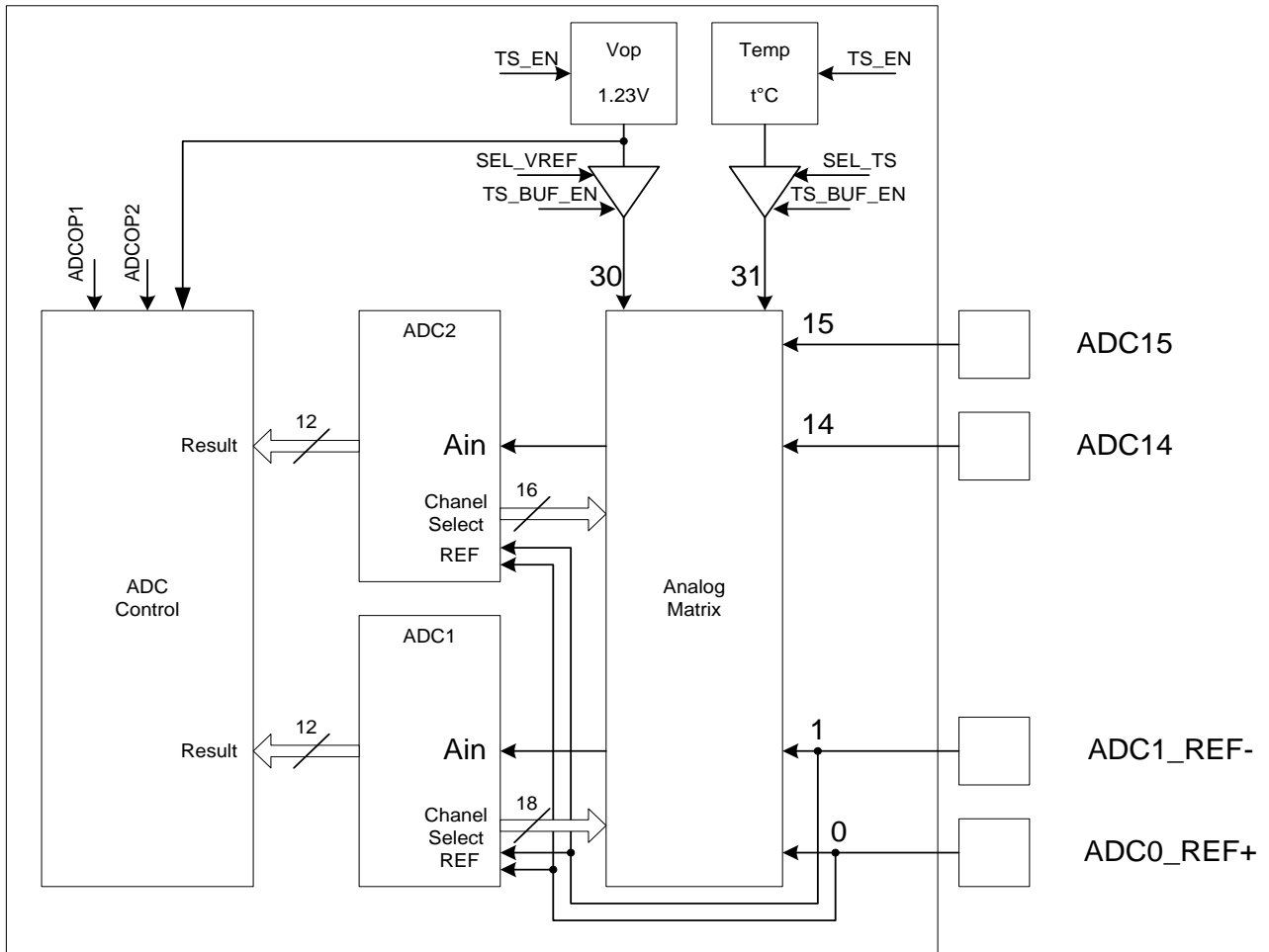


**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
4	BRK EVENT RE	BRK, PCLK: 0 – DMA; 1 – DMA
3	ETR FE EVENT RE	DMA ETR: 0 – DMA; 1 – DMA
2	ETR RE EVENT RE	DMA ETR: 0 – DMA; 1 – DMA
1	CNT ARR EVENT RE	DMA CNT ARR: 0 – DMA; 1 – DMA
0	CNT ZERO EVENT RE	DMA CNT : 0 – DMA; 1 – DMA

## 23 Контроллер MDR\_ADC

16 12- D ,  
512 :  
– AUCC AGND;  
– ADC0\_REF+ ADC\_REF-.  
:  
• 16 ;  
• ;  
• ;  
• ;  
• ;  
• ;  
• ;  
28 CLK.  
CPU\_CLK ADC\_CLK,  
« CPU\_CLK  
Cfг\_REG\_CLKS.  
Cfг\_REG\_DIVCLK[3:0].



**Рисунок 91. Структурная схема контроллера АЦП Cfg\_REG\_ADON.**

1. TS\_EN  
ADCx\_OP  
D,

### 23.1 Преобразование внешнего канала

ADCx\_CFG Cfg\_REG\_CHS[4:0]

Cfg\_M\_REF = 0 Cfg\_M\_REF = 1,  
ADC0\_REF+ ADC1\_REF- Cfg\_REG\_CHCH, Cfg\_REG\_RNGC, Cfg\_REG\_SAMPLE,  
TS\_BUF\_EN, SEL\_VREF, SEL\_TS Cfg\_Sync\_Conver

1 Cfg\_REG\_GO.  
Flg\_REG\_EOCIF

ADCx\_STATUS, ADCx\_RESULT  
Flg\_REG\_EOCIF

ADCx\_RESULT  
 Flg\_REG\_EOCIF  
 Flg\_REG\_OVERWRITE.  
 ADCx\_STATUS.

### 23.2 Последовательное преобразование нескольких каналов

ADCx\_CHSEL  
 Cfg\_M\_REF = 0      Cfg\_M\_REF = 1.  
 ADC0\_REF+    ADC1\_REF-    Cfg\_REG\_RNGC, TS\_BUF\_EN, SEL\_VREF,  
 SEL\_TS    Cfg\_Sync\_Conver    ,    Cfg\_REG\_SAMPLE    Cfg\_REG\_CHCH  
 Delay\_GO  
 CPU\_CLK,  
 ADC\_CLK    CPU\_CLK  
 1    Cfg\_REG\_GO.  
 ADCx\_STATUS,    ADCx\_RESULT    Flg\_REG\_EOCIF  
 Flg\_REG\_EOCIF  
 ADCx\_RESULT  
 Flg\_REG\_EOCIF    Flg\_REG\_OVERWRITE.  
 Flg\_REG\_OVERWRITE    ADCx\_STATUS.  
 ADCx\_CHSEL    Cfg\_REG\_CHCH    1,  
 Cfg\_REG\_CHS[4:0]    Cfg\_REG\_CHCH    0.

### 23.3 Преобразование с контролем границ

ADCx\_H\_LEVEL.    ADCx\_L\_LEVEL  
 Cfg\_REG\_RNGC,  
 Flg\_REG\_AWOIFEN,

### 23.4 Внутренний источник опорного напряжения

( V<sub>OP</sub>    91).  
 TS\_EN    1.  
 ADCx\_OP  
 Cfg\_REG\_CHS    30

```

TS_BUF_EN  SEL_VREF,
                                1      Cfg_REG_GO.
                                Flg_REG_EOCIF
ADC1_STATUS,      ADC1_RESULT
                                Flg_REG_EOCIF
                                ,
                                ADC1_RESULT
                                Flg_REG_EOCIF      Flg_REG_OVERWRITE.
                                ADC1_STATUS.
Flg_REG_OVERWRITE

ADC1_CHSEL      30      Cfg_REG_CHCH  1,
30-      Cfg_REG_CHS[4:0]      Cfg_REG_CHCH  0.

                                TS_BUF_EN  SEL_VREF.

```

### 23.5 Датчик температуры

```

                                TS_EN  1.
                                ADCx_OP
Cfg_REG_CHS      31      ,      TS_BUF_EN  SEL_TS,
1      Cfg_REG_GO.
                                Flg_REG_EOCIF
ADC1_STATUS,      ADC1_RESULT
                                Flg_REG_EOCIF
                                ,
                                ADC1_RESULT
                                Flg_REG_EOCIF      Flg_REG_OVERWRITE.
                                ADC1_STATUS.
Flg_REG_OVERWRITE

ADC1_CHSEL      31      Cfg_REG_CHCH  1,
31-      Cfg_REG_CHS[4:0]      Cfg_REG_CHCH  0.

                                TS_BUF_EN  SEL_TS.

```

### 23.6 Синхронный запуск двух АЦП

```

                                Delay_ADC.
                                Delay_ADC

```

CPU\_CLK, ADC\_CLK  
 CPU\_CLK  
 Cfg\_Sync\_Conver  
 Cfg\_REG\_GO.

### 23.7 Время заряда внутренней емкости

RAIN :  

$$R_{AIN} < (T_s / (f_{C_{ADC}} * C_{ADC} * \ln(2^N))) - R_{ADC}$$

- T<sub>s</sub> - ;
- f<sub>C<sub>ADC</sub></sub> - ;
- C<sub>ADC</sub> - (~ 15 – 20 );
- N - ;
- R<sub>ADC</sub> - (~500 ).

N = 14. DelayGo[2:0] 12 ± 1/4 LSB,  
 N=10. DelayGo[2:0] 10 ± 1 LSB,  
 CPU\_CLK, DelayGo[2:0]  
 CPU\_CLK, ADC\_CLK

**Таблица 293 – Время заряда внутренней емкости АЦП и время преобразования**

<b>DelayGo[2:0]</b>	<b>Дополнительная задержка перед началом преобразования</b>	<b>Общее время Ts заряда емкости АЦП перед началом преобразования</b>	<b>Общее время преобразования АЦП</b>
000	1 x CPU_CLK	4 x CLK + 1 x CPU_CLK	28 x CLK + 1 x CPU_CLK
001	2 x CPU_CLK	4 x CLK + 2 x CPU_CLK	28 x CLK + 2 x CPU_CLK
010	3 x CPU_CLK	4 x CLK + 3 x CPU_CLK	28 x CLK + 3 x CPU_CLK
011	4 x CPU_CLK	4 x CLK + 4 x CPU_CLK	28 x CLK + 4 x CPU_CLK
100	5 x CPU_CLK	4 x CLK + 5 x CPU_CLK	28 x CLK + 5 x CPU_CLK
101	6 x CPU_CLK	4 x CLK + 6 x CPU_CLK	28 x CLK + 6 x CPU_CLK
110	7 x CPU_CLK	4 x CLK + 7 x CPU_CLK	28 x CLK + 7 x CPU_CLK
111	8 x CPU_CLK	4 x CLK + 8 x CPU_CLK	28 x CLK + 8 x CPU_CLK

E<sub>DLADC</sub>, E<sub>ILADC</sub> E<sub>OFFADC</sub>.

ADCx\_CFG

Cfg\_REG\_GO,

## 23.8 Описание регистров блока контроллера АЦП

**Таблица 294 – Описание регистров блока контроллера АЦП**

<b>Базовый Адрес</b>	<b>Название</b>	<b>Описание</b>
0x4008 8000	MDR_ADC	ADC
<b>Смещение</b>		
0x00	MDR_ADC->ADC1_CFG	ADC1
0x04	MDR_ADC->ADC2_CFG	ADC2
0x08	ADC1_H_LEVEL	MDR_ADC->ADCx_H_LEVEL ADC1
0x0C	ADC2_H_LEVEL	MDR_ADC->ADCx_H_LEVEL ADC2
0x10	ADC1_L_LEVEL	MDR_ADC->ADCx_L_LEVEL ADC1
0x14	ADC2_L_LEVEL	MDR_ADC->ADCx_L_LEVEL ADC2
0x18	ADC1_RESULT	MDR_ADC->ADCx_RESULT ADC1
0x1C	ADC2_RESULT	MDR_ADC->ADCx_RESULT ADC2
0x20	ADC1_STATUS	MDR_ADC->ADCx_STATUS ADC1
0x24	ADC2_STATUS	MDR_ADC->ADCx_STATUS ADC2
0x28	ADC1_CHSEL	MDR_ADC->ADCx_CHSEL ADC1
0x2C	ADC2_CHSEL	MDR_ADC->ADCx_CHSEL ADC2



**23.8.1 MDR\_ADC->ADC1\_CFG**

**Таблица 295 – Регистр ADC1\_CFG**

<b>Номер</b>	11	10	9	8...4	3	2	1	0
<b>Доступ</b>	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0
	<b>Cfg M_REF</b>	<b>Cfg REG RNGC</b>	<b>Cfg REG CHCH</b>	<b>Cfg REG CHS[4:0]</b>	<b>Cfg REG SAMPLE</b>	<b>Cfg REG CLKS</b>	<b>Cfg REG GO</b>	<b>Cfg REG ADON</b>

<b>Номер</b>	31...28	27...25	24...21	20	19	18	17	16	15...12
<b>Доступ</b>	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0
	<b>Delay ADC [3:0]</b>	<b>Delay Go [2:0]</b>	<b>TR[3:0]</b>	<b>SEL VREF</b>	<b>SEL TS</b>	<b>TS_BUF EN</b>	<b>TS_EN</b>	<b>Cfg Sync Conver</b>	<b>Cfg REG DIVCLK [3:0]</b>

**Таблица 296 – Описание бит регистра ADC1\_CFG**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...28	Delay ADC [3:0]	ADC1 ADC2 : 0000 – 1 CPU_CLK; 0001 – 2 CPU_CLK; ... 1111 – 16 CPU_CLK
27...25	Delay Go [2:0]	: 000 – 1 CPU_CLK; 001 – 2 CPU_CLK; ... 111 – 8 CPU_CLK
24...21	TR[3:0]	92
20	SEL VREF	1,23 : 0 – ; 1 – . Cfg_REG_CHS = 30
19	SEL TS	: 0 – ; 1 – . Cfg_REG_CHS = 31
18	TS BUF EN	: 0 – ; 1 – . TS_EN = 1

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91Н4, К1986BE91Н4, 1986BE94Н4, К1986BE94Н4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
17	TS EN	0 – ; 1 – .  1
16	Cfg Sync Conver	Cfg_REG_ADON, Cfg_M_REF Cfg_REG_DIVCLK, Cfg_REG_CHS 0 – ; 1 –
15...12	Cfg REG DIVCLK [3:0]	0000 – CPU_CLK; 0001 – CPU_CLK/2; 0010 – CPU_CLK/4; 0011 – CPU_CLK/8; ... 1011 – CPU_CLK/2048 – CPU_CLK
11	Cfg M_REF	0 – ( AU AGND); 1 – ( ADC0_REF+ ADC1_REF-)
10	Cfg REG RNGC	0 – ; 1 –
9	Cfg REG CHCH	0 – ; 1 – ( , )
8...4	Cfg REG CHS [4:0]	00000 – 0 ; 00001 – 1 ; ... 11111 – 31
3	Cfg REG SAMPLE	0 – ; 1 – .
2	Cfg REG CLKS	0 – CPU_CLK; 1 – ADC_CLK
1	Cfg REG GO	“1” ,
0	Cfg REG ADON	0 – ; 1 –

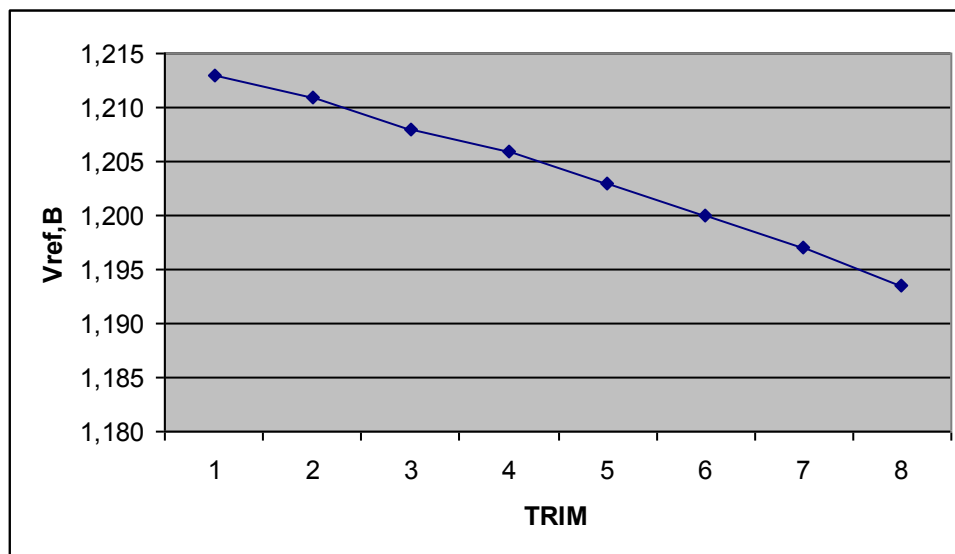


Рисунок 92. Зависимость источника опорного напряжения от подстройки

### 23.8.2 MDR\_ADC->ADC2\_CFG

Таблица 297 – Регистр ADC2\_CFG

Номер	11	10	9	8...4	3	2	1	0
Доступ	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0	0
	Cfg M_REF	Cfg REG RNGC	Cfg REG CHCH	Cfg REG CHS[4:0]	Cfg REG SAMPLE	Cfg REG CLKS	Cfg REG GO	Cfg REG ADON

Номер	31...28	27...25	24...19	18	17	16	15...12
Доступ	U	R/W	U	R/W	R/W	U	R/W
Сброс	0	0	0	0	0	0	0
	-	Delay Go [2:0]	-	ADC2 OP	ADC1 OP	-	Cfg REG DIVCLK [3:0]

Таблица 298 – Описание бит регистра ADC2\_CFG

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	-	
27...25	Delay Go [2:0]	: 000 – 1 CPU_CLK 001 – 2 CPU_CLK ... 111 – 8 CPU_CLK
24...19	-	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
18	ADC2 OP	0 – ( ); 1 – ( ). EN 1 TS
17	ADC1 OP	0 – ( ); 1 – ( ). EN 1 TS
16	-	
15...12	Cfg REG DIVCLK [3:0]	0000 – CPU_CLK = HCLK; 0001 – CPU_CLK = HCLK/2; 0010 – CPU_CLK = HCLK/4; 0011 – CPU_CLK = HCLK/8; ... 1011 – CPU_CLK = HCLK/2048 CPU_CLK = HCLK;
11	Cfg M_REF	0 – ( AU AGND); 1 – ( ADC0_REF+ ADC1_REF-)
10	Cfg REG RNGC	1 – , 0 – ;
9	Cfg REG CHCH	0 – ; 1 – ( , )
8...4	Cfg REG CHS [4:0]	00000 – 0 ; 00001 – 1 ; ... 11111 – 31
3	Cfg REG SAMPLE	0 – ; 1 – .
2	Cfg REG CLKS	0 – CPU_CLK; 1 – ADC_CLK
1	Cfg REG GO	“1” .
0	Cfg REG ADON	0 – ; 1 –

### 23.8.3 MDR\_ADC->ADCx\_H\_LEVEL

**Таблица 299 – Регистр ADCx\_H\_LEVEL**

<b>Номер</b>	31...12	11...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>REG H LEVEL [11:0]</b>

**Таблица 300 – Описание бит регистра ADCx\_H\_LEVEL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...12	-	
11...0	REG H LEVEL [11:0]	

### 23.8.4 MDR\_ADC->ADCx\_L\_LEVEL

**Таблица 301 – Регистр ADCx\_L\_LEVEL**

<b>Номер</b>	31...12	11...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>REG L LEVEL [11:0]</b>

**Таблица 302 – Описание бит регистра ADCx\_L\_LEVEL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...12	-	
11...0	REG L LEVEL [11:0]	

### 23.8.5 MDR\_ADC->ADCx\_RESULT

**Таблица 303 – Регистр ADCx\_RESULT**

<b>Номер</b>	31...21	20...16	15...12	11...0
<b>Доступ</b>	U	RO	U	RO
<b>Сброс</b>	0	0	0	0
	-	<b>CHANNEL [4:0]</b>	-	<b>RESULT [11:0]</b>

**Таблица 304 – Описание бит регистра ADCx\_RESULT**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...21	-	
20...16	CHANNEL [11:0]	
15...12	-	
11...0	RESULT [11:0]	

### 23.8.6 MDR\_ADC->ADCx\_STATUS

**Таблица 305 – Регистр ADCx\_STATUS**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>EOCIF IE</b>	<b>AWOIF IE</b>	<b>Flg REG EOCIF</b>	<b>Flg REG AWOIFEN</b>	<b>Flg REG OVERWRITE</b>

**Таблица 306 – Описание бит регистра ADCx\_STATUS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	-	
4	EOCIF_IE	Flg_REG_EOCIF: 0 – ; 1 –
3	AWOIF_IE	Flg_REG_AWOIFEN: 0 – ; 1 –
2	Flg REG EOCIF	ADCx_RESULT: 1 – ; 0 –
1	Flg REG AWOIFEN	: 0 – ; 1 –
0	Flg REG OVERWRITE	: 0 – ; 1 – ,

### 23.8.7 MDR\_ADC->ADCx\_CHSEL

**Таблица 307 – Регистр ADCx\_CHSEL**

<b>Номер</b>	31... 0
<b>Доступ</b>	R/W
<b>Сброс</b>	0
	<b>SI_Ch_Ch_REF[31:0]</b>

**Таблица 308 – Описание бит регистра ADCx\_CHSEL**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	SI_Ch_Ch_REF[31:0]	: ; 0 – ; 1 – ;

## 24 Контроллер MDR\_DAC

Cfg\_ON\_DACx 1,

DACx\_DATA

( Cfg\_SYNC\_A=0)  
DACx\_OUT

Cfg\_SYNC\_A=1)  
DACx\_DATA.

Cfg\_M\_REFx=0,  
AU<sub>CC</sub>.

Cfg\_M\_REFx=1  
DACx\_REF.

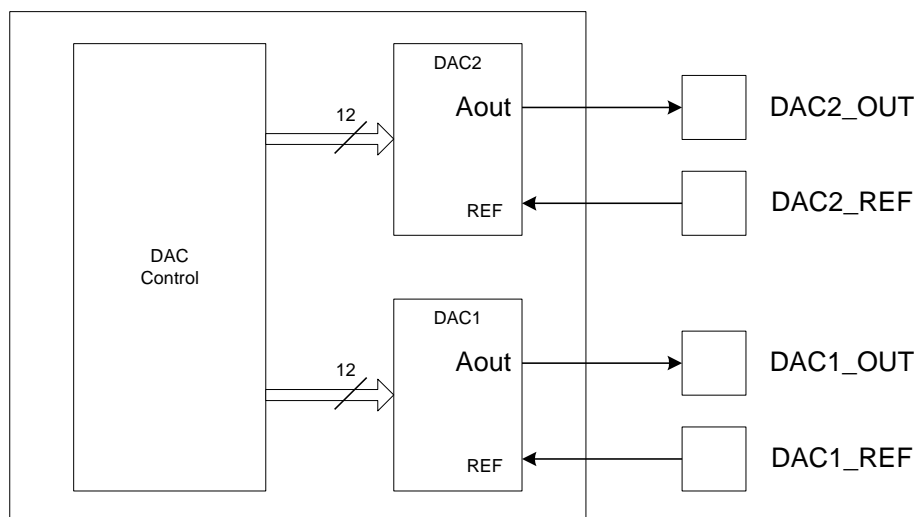


Рисунок 93. Структурная схема контроллера ЦАП

### 24.1 Описание регистров блока контроллера ЦАП

Таблица 309 – Описание регистров блока контроллера ЦАП

Базовый Адрес	Название	Описание
0x4009 0000	MDR_DAC	DAC
<b>Смещение</b>		
0x00	MDR_DAC->CFG	DAC
0x04	MDR_DAC->DAC1_DATA	DAC1
0x08	MDR_DAC->DAC2_DATA	DAC2



**24.1.1 MDR\_DAC->CFG**

**Таблица 310 – Регистр CFG**

<b>Номер</b>	31...5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0
	-	<b>Cfg SYNC_A</b>	<b>Cfg ON_DAC1</b>	<b>Cfg ON_DAC0</b>	<b>Cfg M_REF1</b>	<b>Cfg M_REF0</b>

**Таблица 311 – Описание бит регистра CFG**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...5	-	
4	Cfg_SYNC_A	DAC1 DAC2: 0 – ; 1 –
3	Cfg_ON_DAC1	DAC2: 1 – ; 0 –
2	Cfg_ON_DAC0	DAC1: 1 – ; 0 –
1	Cfg_M_REF1	DAC2: 0 – 1 – AU <sub>CC</sub> ; DAC2_REF
0	Cfg_M_REF0	DAC1: 0 – 1 – AU <sub>CC</sub> ; DAC1_REF

### 24.1.2 MDR\_DAC->DAC1\_DATA

**Таблица 312 – Регистр DAC1\_DATA**

<b>Номер</b>	31...28	27...16	15...12	11...0
<b>Доступ</b>	U	R/W	U	R/W
<b>Сброс</b>	0	0	0	0
	-	<b>DAC1_DATA[11:0]</b>	-	<b>DAC0_DATA[11:0]</b>

**Таблица 313 – Описание бит регистра DAC1\_DATA**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...28	-	
27...16	DAC1 DATA[11:0]	DAC1 Cfg_SYNC_A=1. . DAC2_DATA
15...12	-	
11...0	DAC0 DATA[11:0]	DAC0

### 24.1.3 MDR\_DAC->DAC2\_DATA

**Таблица 314 – Регистр DAC2\_DATA**

<b>Номер</b>	31...28	27...16	15...12	11...0
<b>Доступ</b>	U	R/W	U	R/W
<b>Сброс</b>	0	0	0	0
	-	<b>DAC0_DATA[11:0]</b>	-	<b>DAC1_DATA[11:0]</b>

**Таблица 315 – Описание бит регистра DAC21\_DATA**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...28	-	
27...16	DAC0 DATA[11:0]	DAC0 Cfg_SYNC_A=1. . DAC1_DATA
15...12	-	
11...0	DAC1 DATA[11:0]	DAC1

*Примечание* – DAC2 Cfg\_SYNC\_A, DAC1  
DACx\_DATA.

## 25 Контроллер схемы компаратора MDR\_COMP

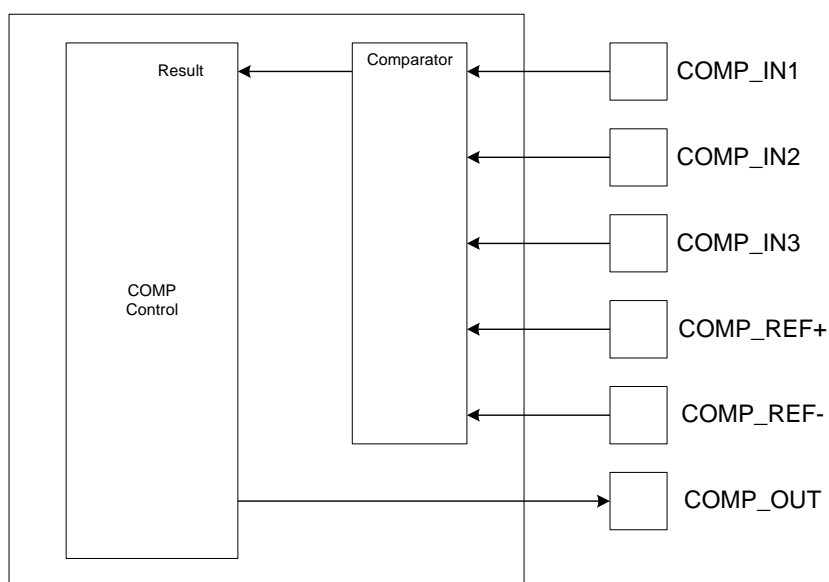
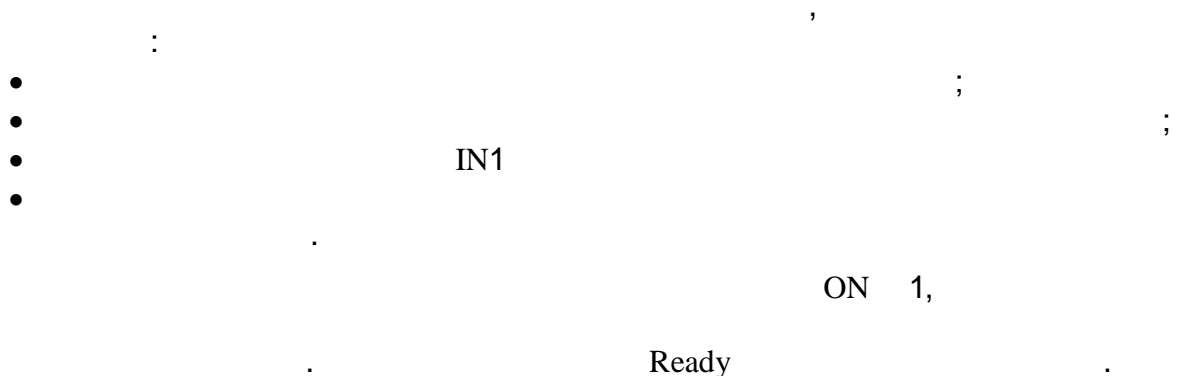
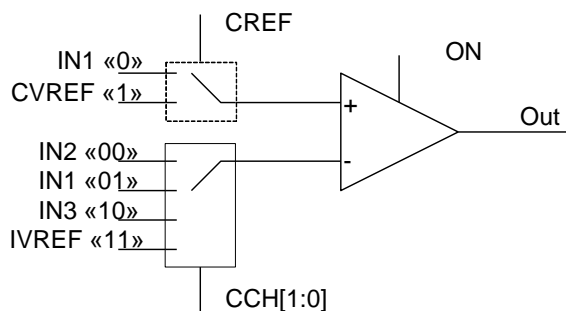


Рисунок 94. Структура блока компаратора

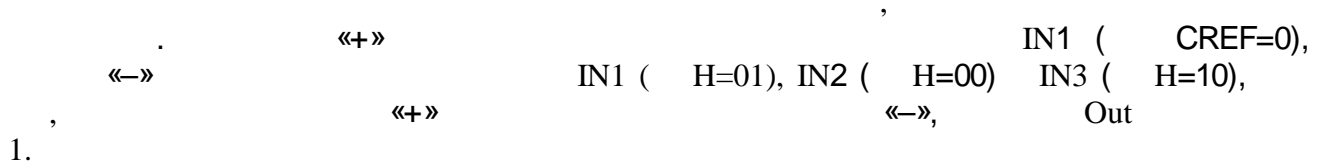


\*IVREF –

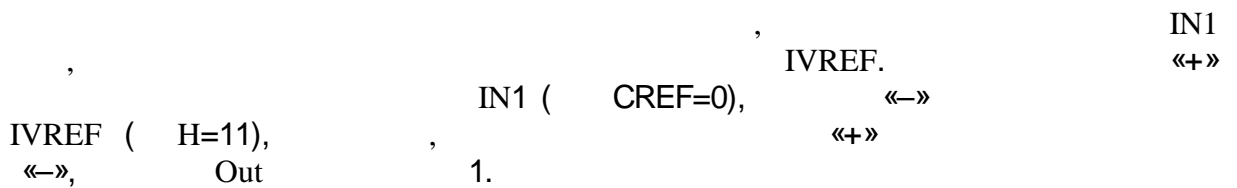
1,2 .

Рисунок 95. Структура мультиплексирования входов компаратора

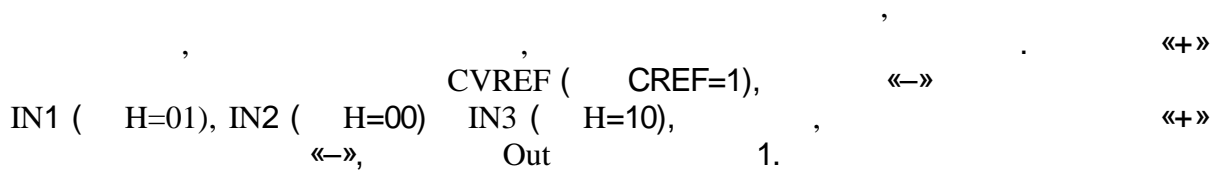
## 25.1 Сравнение внешних сигналов



## 25.2 Сравнение сигнала с внутренним источником опорного напряжения



## 25.3 Сравнение внешних сигналов с внутренней шкалой напряжений



## 25.4 Формирование внутренней шкалы напряжений

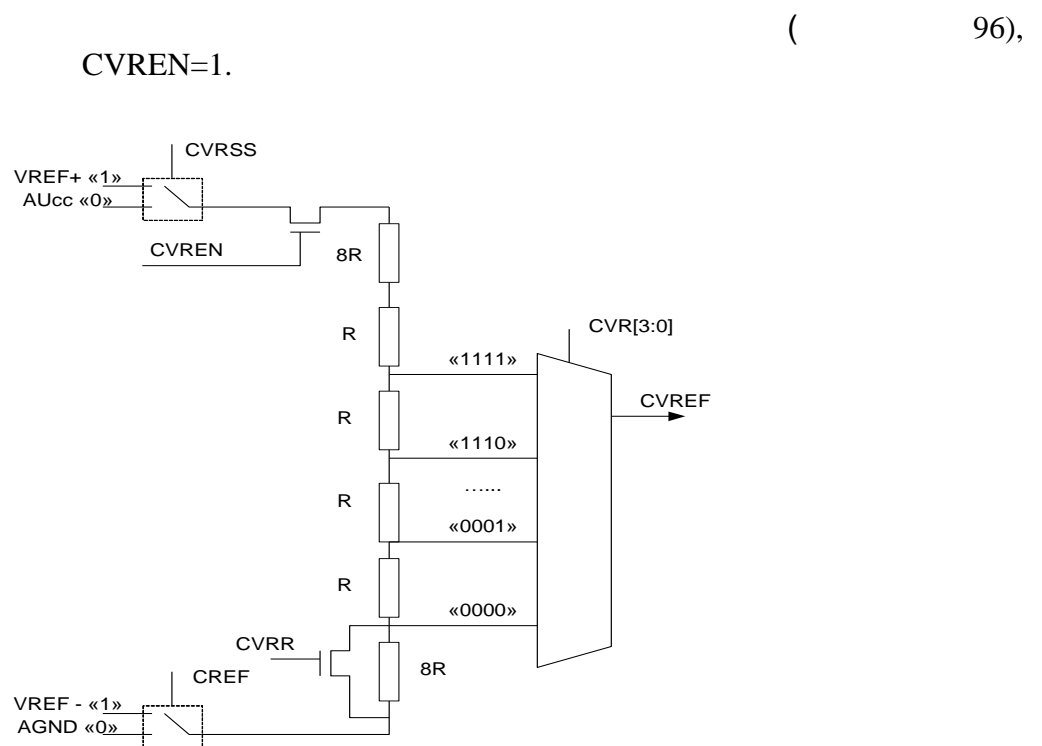


Рисунок 96. Структура блока формирования CVREF

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

AU<sub>CC</sub> (CVRSS = 0),

COMP\_VREF+ (CVRSS = 1).

CVREF  
316,

COMP\_VREF-  
CVRR CVR

**Таблица 316 – Формирование внутренней шкалы напряжений CVREF**

CVRR	CVR[3:0]	Отношение резисторов	Напряжение CREF при U <sub>CC</sub> =3.3 В, В	Входной импеданс VREF+, Ом	Примечание
0	0000	8/32	0.83	12К	
	0001	9/32	0.93	13К	
	0010	10/32	1.03	13.8К	
	0011	11/32	1.13	14.4К	
	0100	12/32	1.24	15К	
	0101	13/32	1.34	15.4К	
	0110	14/32	1.44	15.8К	
	0111	15/32	1.55	15.9К	
	1000	16/32	1.65	16К	
	1001	17/32	1.75	15.9К	
	1010	18/32	1.86	15.8К	
	1011	19/32	1.96	15.4К	
	1100	20/32	2.06	15К	
	1101	21/32	2.17	14.4К	
	1110	22/32	2.27	13.8К	
	1111	23/32	2.37	12.9К	
1	0000	0/24	0.00	0.5К	
	0001	1/24	0.14	1.9К	
	0010	2/24	0.28	3.7К	
	0011	3/24	0.41	5.3К	
	0100	4/24	0.55	6.7К	
	0101	5/24	0.69	7.9К	
	0110	6/24	0.83	9К	
	0111	7/24	0.96	9.9К	
	1000	8/24	1.10	10.7К	
	1001	9/24	1.24	11.3К	
	1010	10/24	1.38	11.7К	
	1011	11/24	1.51	11.9К	
	1100	12/24	1.65	12К	
	1101	13/24	1.79	11.9К	
	1110	14/24	1.93	11.7К	
	1111	15/24	2.06	11.3К	

INV

Out  
OUT\_COMP.

Rst\_As (

1,

).

HCLK

Rst\_Sy.

Rst\_Ich

1

COMP\_RESULT\_LATCH.

## 25.5 Описание регистров блока контроллера компаратора

**Таблица 317- Описание регистров блока контроллера компаратора**

Базовый Адрес	Название	Описание
0x4009_8000	MDR_COMP	
<b>Смещение</b>		
0x00	MDR_COMP->CFG	
0x04	MDR_COMP->RESULT	
0x08	MDR_COMP->RESULT_LATCH	-

### 25.5.1 MDR\_COMP->CFG

**Таблица 318 – Регистр CFG**

Номер	31...14	13	12	11	10...9	8	7...4	3	2	1	0
Доступ	U	R/W	RO	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0	0	0	0	0
	-	CMP IE	Ready	INV	CCH [1:0]	CREF	CVR [3:0]	CVR EN	CVRSS	CVRR	ON

**Таблица 319 – Описание бит регистра CFG**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...14	-	
13	CMP IE	Rst_lch: 0 – ; 1 –
12	Ready	: 0 – ; 1 –
11	INV	: 0 – ; 1 –
10...9	CCH [1:0]	: 00 – «-» IN2; 01 – «-» IN1; 10 – «-» IN3; 11 – «-» 1.2 (IVREF).
8	CREF	: 0 – «+» IN1; 1 – «+» CREF
7...4	CVR [3:0]	CVREF. . 316
3	CVREN	CVREF: 0 – ; 1 –

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
2	CVRSS	CVREF: 0 – CVREF AVdd AGND; 1 – CVREF Vref+ Vref-
1	CVRR	CVREF: 0 – CVREF ; 1 – CVREF
0	ON	: 0 – ; 1 –

### 25.5.2 MDR\_COMP->RESULT

Таблица 320 – Регистр RESULT

Номер	31...3	2	1	0
Доступ	U	R/W	R/W	R/W
Сброс	0	0	0	0
	-	Rst_lch	Rslt_As	Rslt_Sy

Таблица 321 – Описание бит регистра RESULT

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...3	-	
2	Rst_lch	COMP_RESULT_LATCH, .
1	Rslt_As	
0	Rslt_Sy	HCLK, ,

### 25.5.3 MDR\_COMP->RESULT\_LATCH

Таблица 322 – Регистр RESULT\_LATCH

Номер	31...1	0
Доступ	U	R/W
Сброс	0	0
	-	Rst_lch

Таблица 323 – Описание бит регистра RESULT\_LATCH

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...1	-	
0	Rst_lch	COMP_RESULT_LATCH, .

## 26 Контроллер интерфейса MDR\_I2C

I2C

1986 9

Master.

3

- : 100 Kbps (DIV=150 HCLK=80 );
- : 400 Kbps (DIV=25 HCLK=80 );
- : 1 Mbps (DIV=1 HCLK=80 ).

I2C

Fscl = HCLK/(5\*(DIV+1)).

### 26.1 Конфигурация системы

I2C SCL. SDA

« »

SDA

SCL.

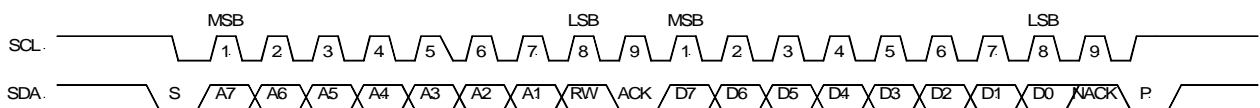
SCL SDA

( . « START » « STOP »).

### 26.2 Протокол I2C

I2C 4 :

- START;
- ;
- ;
- STOP.



**Рисунок 97. Передача по I2C**



### 26.3 Сигнал START

( SCL SDA ), START S SCL.  
 SDA START .  
 START START,  
 STOP.  
 ( , , , ) .  
 I2C\_CMD START RD WR. START  
 SCL START, START.

### 26.4 Передача адреса

7- START, RW RW  
 ACK SDA 9-  
 SCL 10-  
 I2C\_TXD WR I2C\_CMD.

### 26.5 Передача данных

ACK 9- SCL RW  
 NACK ( , ) START  
 STOP .  
 SDA NACK, STOP  
 START.  
 WR. I2C\_TXD RD.  
 TR\_PROG I2C\_STA.  
 INT. INT.  
 I2C\_RXD INT\_EN, TR\_PROG

## 26.6 Сигнал STOP

SDA STOP. STOP P  
, SCL

## 26.7 Описание регистров контроллера I2C

Таблица 324 – Описание регистров контроллера I2C

Базовый Адрес	Название	Описание
0x4005 0000	MDR_I2C	I2C
<b>Смещение</b>		
0x00	MDR_I2C->PRL	
0x04	MDR_I2C->PRH	
0x08	MDR_I2C->CTR	I2C
0x0C	MDR_I2C->RXD	I2
0x10	MDR_I2C->STA	I2C
0x14	MDR_I2C->TXD	I2C
0x18	MDR_I2C->CMD	I2

### 26.7.1 MDR\_I2C->PRL

Таблица 325 – Регистр PRL

Номер	31...8	7... 0
Доступ	U	R/W
Сброс	0	0
	-	PR[7:0]

Таблица 326 – Описание бит регистра PRL

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7...0	PR[7:0]	

### 26.7.2 MDR\_I2C->PRH

Таблица 327 – Регистр PRH

Номер	31...8	7... 0
Доступ	U	R/W
Сброс	0	0
	-	PR[15:8]

Таблица 328 – Описание бит регистра PRH

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7...0	PR[15:8]	

**26.7.3 MDR\_I2C->CTR**

**Таблица 329 – Регистр CTR**

<b>Номер</b>	31...8	7	6	5	4...0
<b>Доступ</b>	U	R/W	R/W	R/W	U
<b>Сброс</b>	0	0	0	0	0
	-	<b>EN_I2C</b>	<b>EN_INT</b>	<b>S_I2C</b>	-

**Таблица 330 – Описание бит регистра CTR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7	EN_I2C	I2C: 0 – ; 1 –
6	EN_INT	I2C: 0 – ; 1 –
5	S_I2C	I2 : 0 – 400 ; 1 – 1
4...0	-	

**26.7.4 MDR\_I2C->RXD**

**Таблица 331 – Регистр RXD**

<b>Номер</b>	31...8	7... 0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>RXD[7:0]</b>

**Таблица 332 – Описание бит регистра RXD**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7...0	RXD[7:0]	I2C

**26.7.5 MDR\_I2C->STA**

**Таблица 333 – Регистр STA**

<b>Номер</b>	31...8	7	6	5	4...2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0	0	0
	-	<b>Rx ACK</b>	<b>BUSY</b>	<b>LOST ARB</b>	-	<b>TR PROG</b>	<b>INT</b>

**Таблица 334 – Описание бит регистра STA**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	
7	Rx ACK	ACK: 0 – ACK ; 1 – NACK
6	BUSY	I2C: 0 – Stop bit; 1 – Start bit
5	LOST ARB	: 0 – ; 1 – . : ; - Stop bit, ; ; - SDA , SDA
4...2	-	
1	TR PROG	: 0 – ; 1 –
0	INT	, EN_INT: 0 – ; 1 – . : ; - ; -

### 26.7.6 MDR\_I2C->TXD

Таблица 335 – Регистр TXD

Номер	31...8	7...0
Доступ	U	R/W
Сброс	0	0
	-	TXD[7:0]

Таблица 336 – Описание бит регистра TXD

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7...0	TXD[7:0]	I2C. : 0 – ; 1 –

### 26.7.7 MDR\_I2C->CMD

Таблица 337 – Регистр CMD

Номер	31...8	7	6	5	4	3	2...1	0
Доступ	U	R/W	R/W	R/W	R/W	R/W	U	R/W
Сброс	0	0	0	0	0	0	0	0
	-	START	STOP	RD	WR	ACK	-	CLR INT

Таблица 338 – Описание бит регистра CMD

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7	START	START bit. 1. ,
6	STOP	STOP bit. 1. ,
5	RD	: 0 – ; 1 –
4	WR	; 0 – ; 1 –
3	ACK	ACK : 0 – ACK; 1 – NACK
2...1	-	
0	CLR INT	1 INT.

## 27 Контроллер MDR\_SSP

(SSP – Synchronous Serial Port)

– SPI Motorola;

– SSI Texas Instruments;

– Microwire National Semiconductor.

– SSP FIFO ;

– ( 16- ) ;

– FIFO ( 16- ).

– FIFO ;

– FIFO ;

– FIFO .

– SPI;

– Microwire;

– SSI.

### 27.1 Основные характеристики модуля SSP

– ;

– ;

– (8 16 )

– FIFO (First In First Out – );

– : SPI, Microwire, SSI;

– 4 16 ;

– FIFO , FIFO

– ;

– (DMA). ;

– . 98.

### 27.2 Программируемые параметры

– ;

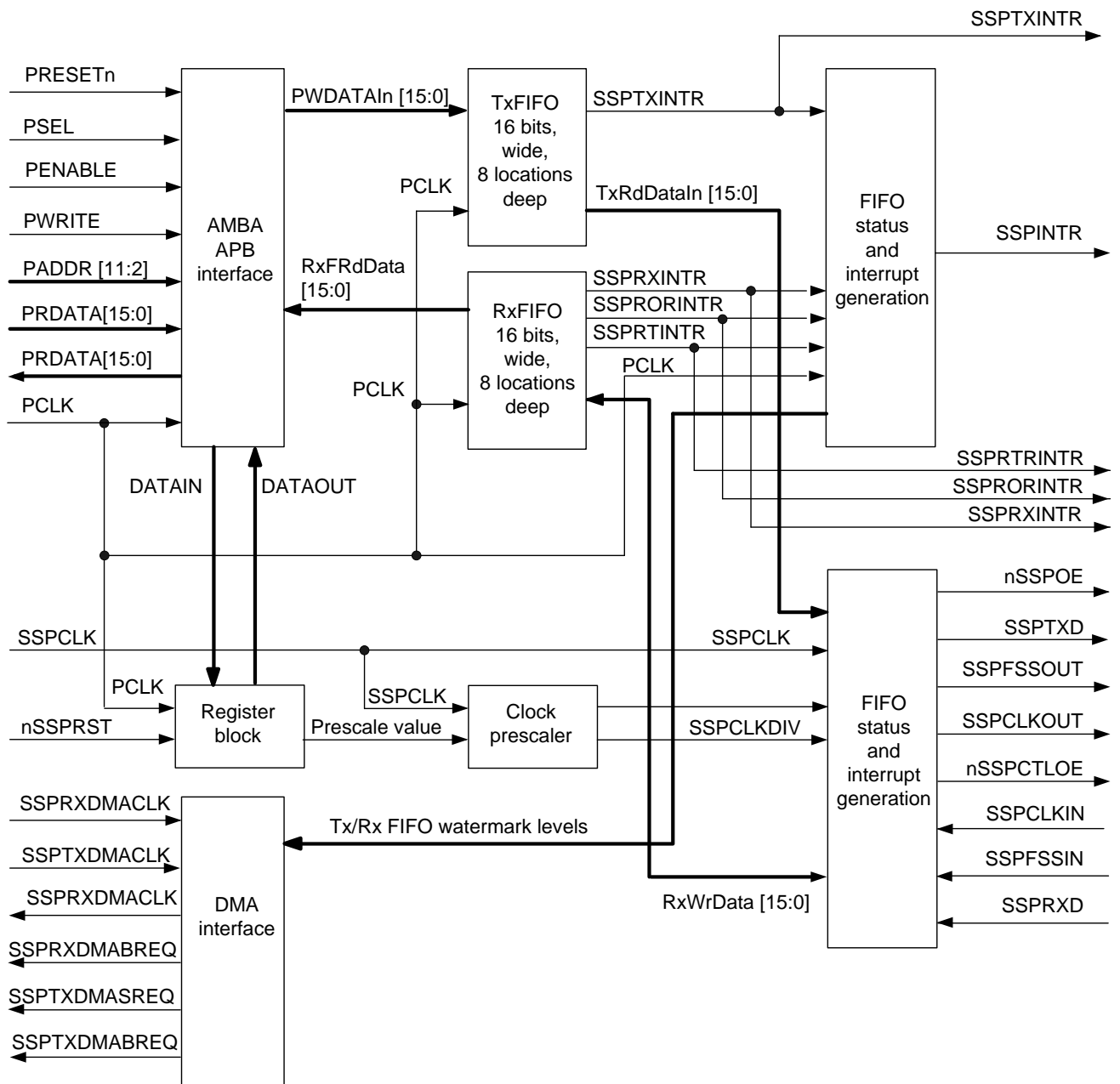
– ;

– ;

– ;

– 4 16 ;

– .



**Рисунок 98. Структурная схема модуля SSP**

### 27.3 Характеристики интерфейса SPI

SPI Motorola :

—  
—

### 27.4 Характеристики интерфейса Microwire

Microwire National Semiconductor :

—

8-

## 27.5 Характеристики интерфейса SSI

SSI Texas Instruments ;  
 – ;  
 – ( )

## 27.6 Общий обзор модуля SSP

SSP  
 ,  
 Semiconductor, SSI Texas Instruments. SPI Motorola, Microwire National  
 :  
 – , ;  
 – ;  
 – ;  
 – ;  
 – 16 FIFO,  
 SSP\_RXD. SSP\_TXD  
 SSP SSP\_CLK 2 , SSPCLK. SSPCLK  
 , CR0 CR1.  
 SSPTXINTR – ;  
 SSPRXINTR – ;  
 SSPRORINTR – FIFO;  
 SSPRTINTR – FIFO.  
 , SSPINTR,  
 NVIC. ,  
 (DMA)  
 DMA. SSPFSSOUT  
 ( SSI, – ),  
 ( SPI Microwire, – ).



### 27.6.1 Блок формирования тактового сигнала

SSP\_CLK  
 ,  
 .  
 SSPCPSR  
 2 254 2.  
 ,  
 ( , , ,  
 ).  
 ,  
 SSP\_CLK.  
 1 256,  
 SSPCR0.

### 27.6.2 Буфер FIFO передатчика

FIFO – « 16 , 8 , ».  
 , .

### 27.6.3 Буфер FIFO приемника

FIFO – « 16 , 8 , ».  
 , .

### 27.6.4 Блок приема и передачи данных

SSP\_CLK  
 SSPCLK.  
 FIFO  
 ,  
 SSP\_CLK, SSP\_TXD .  
 SSP\_RXD, , FIFO  
 ,  
 SSP\_CLK.  
 FIFO, , SSP\_TXD.  
 SSP\_RXD , SSP\_CLK, ,  
 FIFO , .

**27.6.5 Блок формирования прерываний**

SSP

NVIC,

**27.6.6 Интерфейс прямого доступа к памяти**

DMA

DMA.

**27.6.7 Конфигурирование приемопередатчика**

Microwave National Semiconductor), Motorola, SSI Texas Instruments,  
(SPI CR0 CR1.

SSPCLK

PSR.

**27.6.8 Разрешение работы приемопередатчика**

SSE

CR1.

16-

FIFO

SSP\_TXD SSP\_RXD.

**27.6.9 Соотношения между тактовыми сигналами**

CPU\_CLK SSPCLK. SSPCLK CPU\_CLK.

SSPCLK

CPU\_CLK

FSSPCLK <= FPCLK.

SSP\_CLK

SSP\_CLK,

SSP\_CLK.

SSP\_TXD

SSP\_CLK,

SSP\_RXD

SSP\_CLK

SSPCLK 12 ,  
 SSP\_CLK.  
 SSPCLK  
 SSP\_CLK 12,  
 1,8432 /  
 CPCR 0. 2, SCR[7:0] 3,6864 CR0  
 CPCR 0. SSPCLK 12, SCR[7:0] 22.12 CR0  
 SSPCLKOUT 254 \* 256. SSPCLK  
 SSPCLK  
 :  
 FSSPCLK(min) => 2 x FSSPCLKOUT(max) [for master mode]  
 FSSPCLK(min) => 12 x FSSPCLKIN(max) [for slave mode].  
 SSPCLK  
 :  
 FSSPCLK(max) <= 254 x 256 x FSSPCLKOUT(min) [for master mode]  
 FSSPCLK(max) <= 254 x 256 x FSSPCLKIN(min) [for slave mode].

**27.6.10 Программирование регистра управления CR0**

CR0 :  
 ;  
 ;  
 .  
 SSPCLK  
 SSPCR0 SCR (Serial Clock Rate – )  
 CPSDVSR (clock prescale divisor value – )  
 SSPCPSR.  
 FRF,  
 DSS SSPCR0.  
 SPI Motorola ( SPH  
 SPO).

**27.6.11 Программирование регистра управления CR1**

SSPCR1 :  
 ;  
 ;  
 .

0 MS SSPCR1 ( )  
 1 MS SSP\_TXD  
 SSP\_TXD ) SOD (slave mode SSP\_TXD output disable – CR1.  
 1 SSE (Synchronous Serial Port Enable – )

**27.6.12 Формирование тактового сигнала обмена данными**

SSPCLK.  
 CPSDVSr, 2 254,  
 CPSR. (1 +  
 SCR) 1 256, SCR CR0.  
 SSP\_CLK  
 :  

$$FSSPCLKOUT = FSSPCLK / (CPSDVR * (1+SCR))$$
  
 CPSDVSr = 2, SSP\_CLK SSPCLK 7.2 3.6864 1.8432

**27.6.13 Формат информационного кадра**

4 16 :  
 – SSI Texas Instruments;  
 – SPI Motorola;  
 – Microwire National Semiconductor.  
 SSP\_CLK SSP\_CLK  
 ,  
 SPI Microwire  
 SSP\_FSS  
 SSI Texas Instruments  
 SSP\_FSS ,  
 SSP,  
 SSP\_CLK,  
 SSI SPI, Microwire  
 National Semiconductor

8-

4 16  
13 25

### 27.6.14 Формат синхронного обмена SSI фирмы Texas Instruments

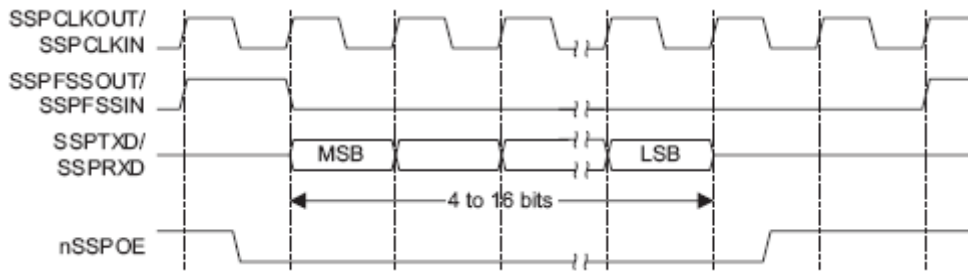
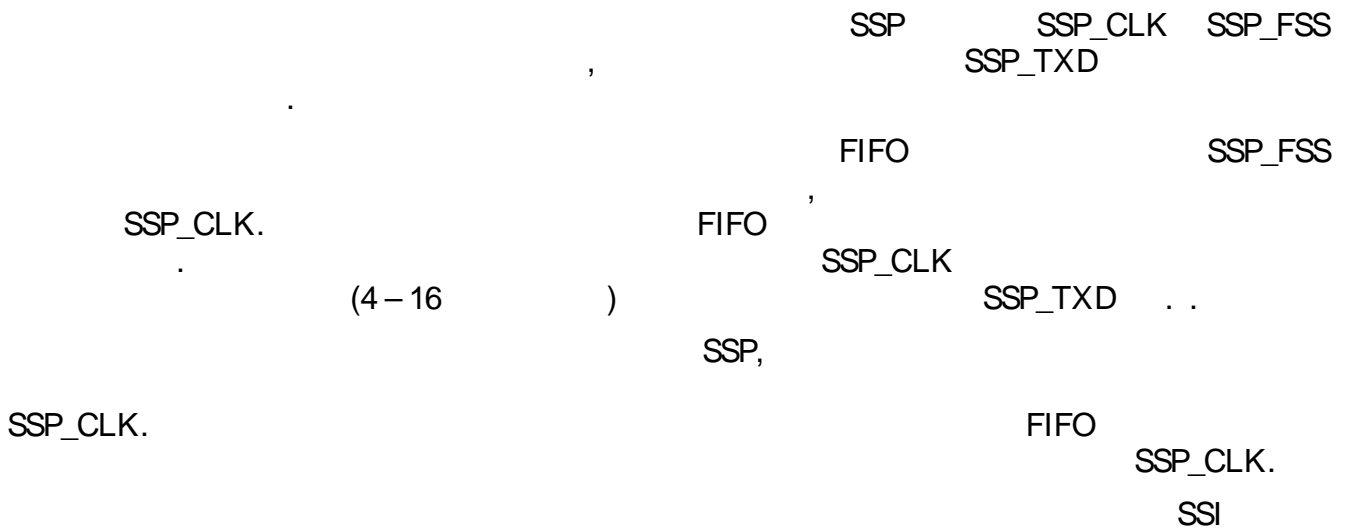


Рисунок 99. Формат синхронного обмена протокола SSI (единичный обмен)



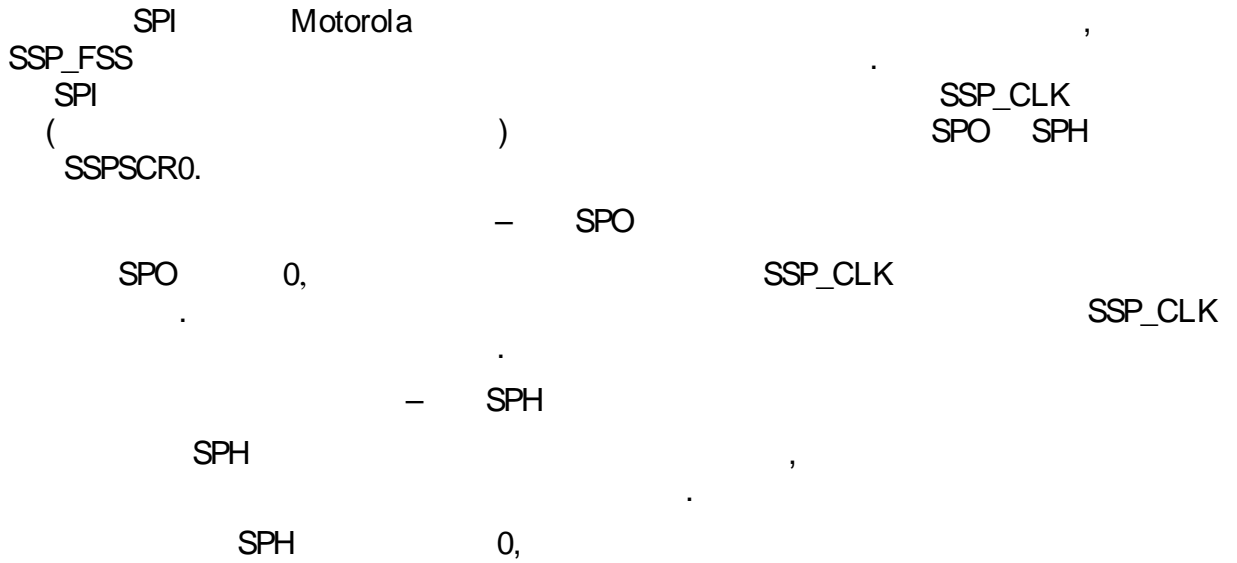
Texas Instruments

100 –

99 –

Рисунок 100. Формат синхронного обмена протокола SSI (непрерывный обмен)

27.6.15 Формат синхронного обмена SPI фирмы Motorola



27.6.16 Формат синхронного обмена SPI фирмы Motorola, SPO=0, SPH=0

SPI SPO=0, SPH=0 : 101 - 102 -

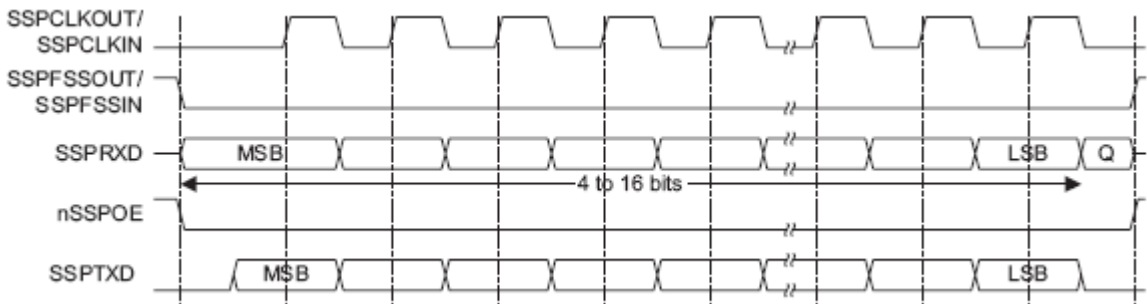


Рисунок 101. Формат синхронного обмена протокола SPI, SPO=0, SPH=0 (одиночный обмен)

Примечание – Q

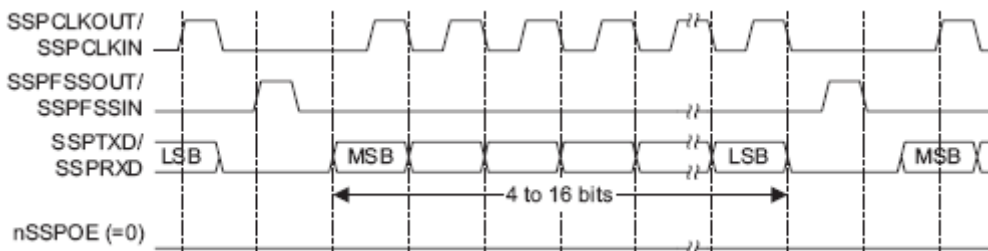
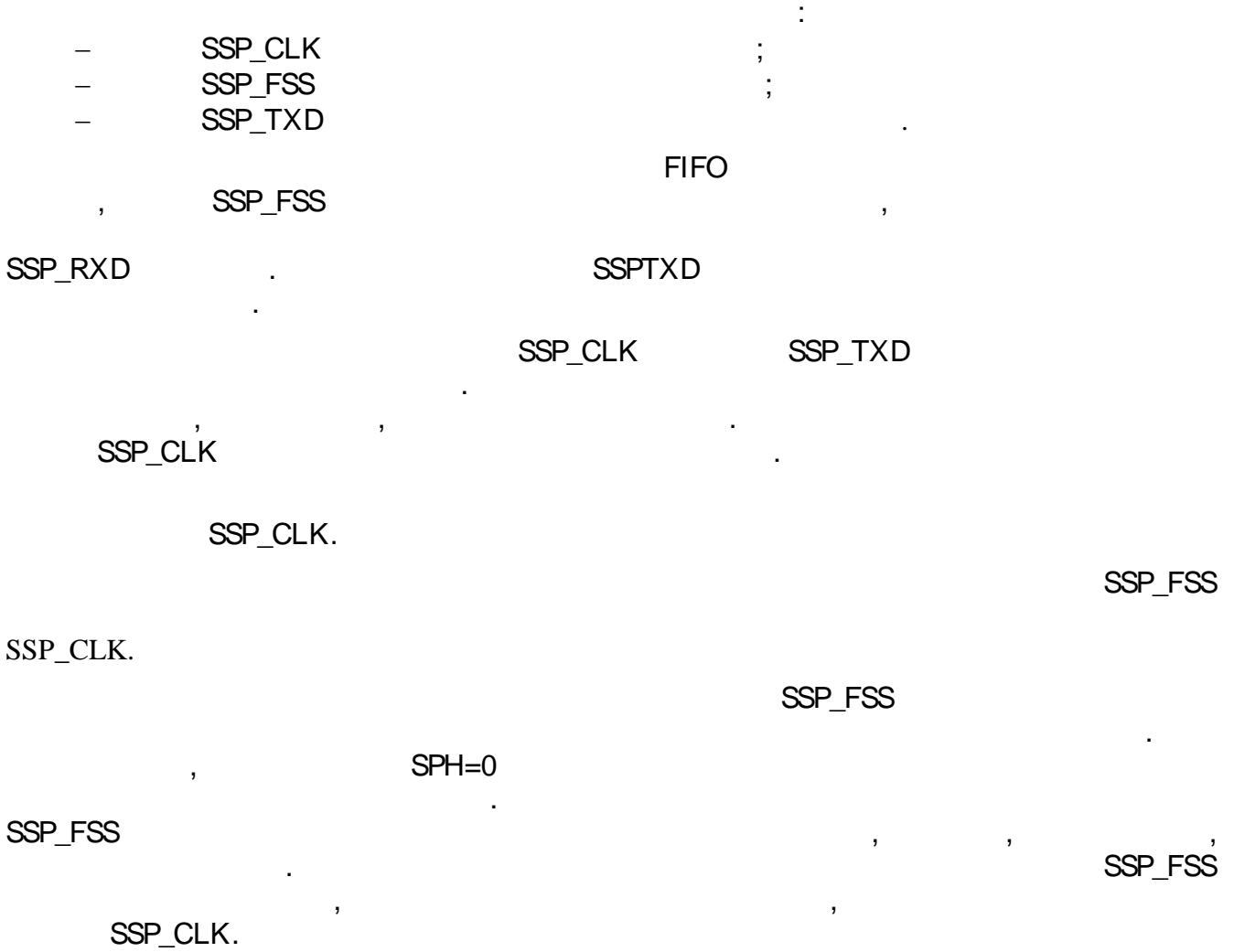
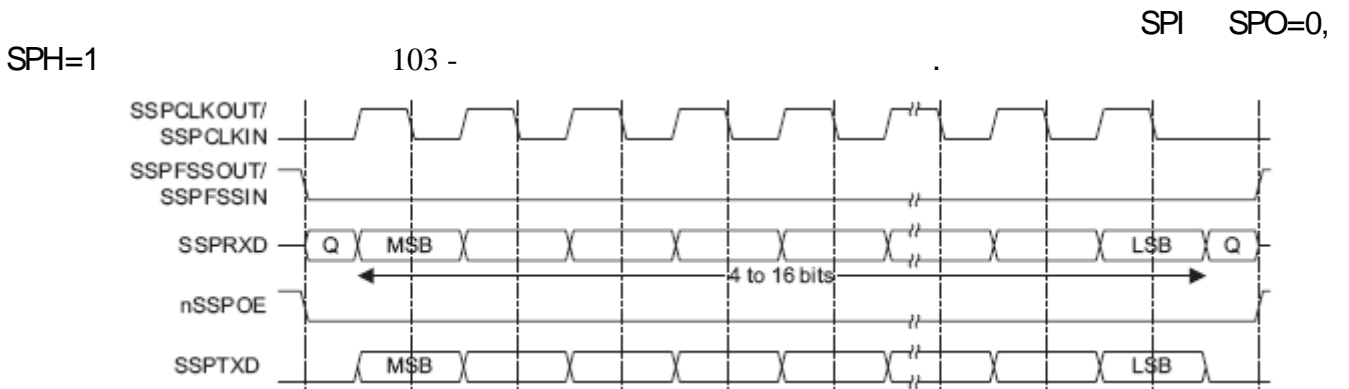


Рисунок 102. Формат синхронного обмена протокола SPI, SPO=0, SPH=0 (непрерывный обмен)



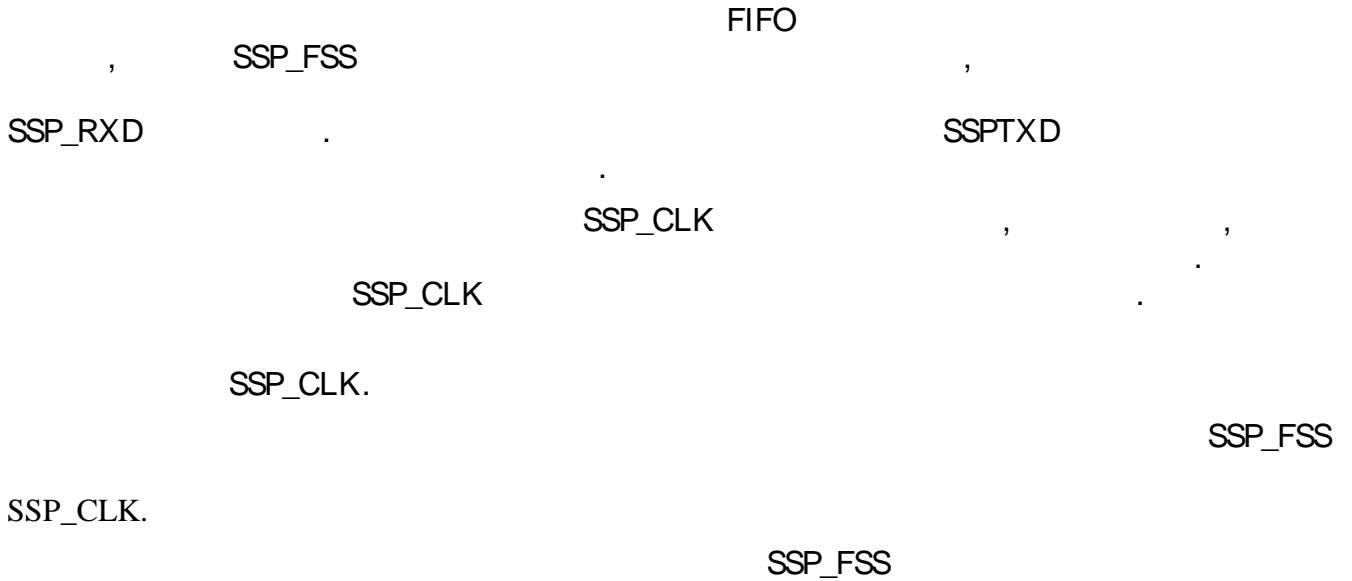
**27.6.17 Формат синхронного обмена SPI фирмы Motorola, SPO=0, SPH=1**



**Рисунок 103. Формат синхронного обмена протокола SPI, SPO=0, SPH=1**

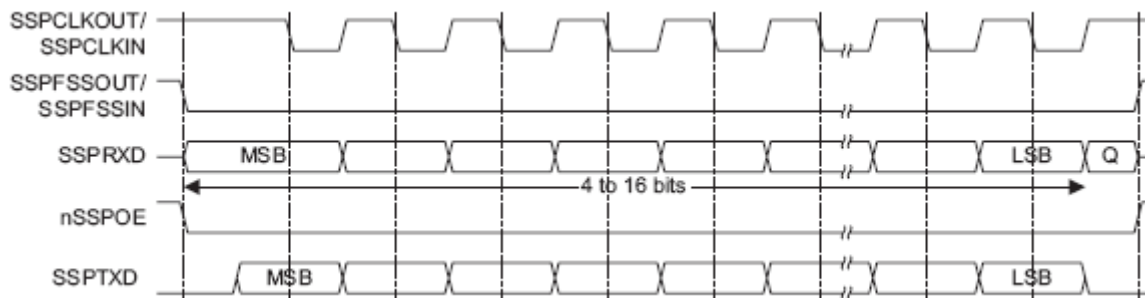
*Примечание* –

- SSP\_CLK
- SSP\_FSS
- SSP\_TXD



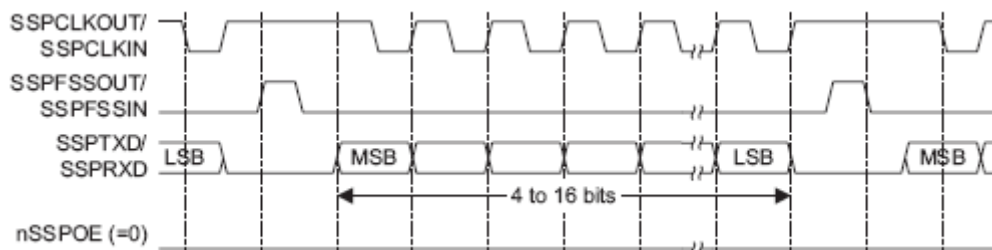
**27.6.18 Формат синхронного обмена SPI фирмы Motorola, SPO=1, SPH=0**

SPH=0 : 104 - 105 - SPI SPO=1,



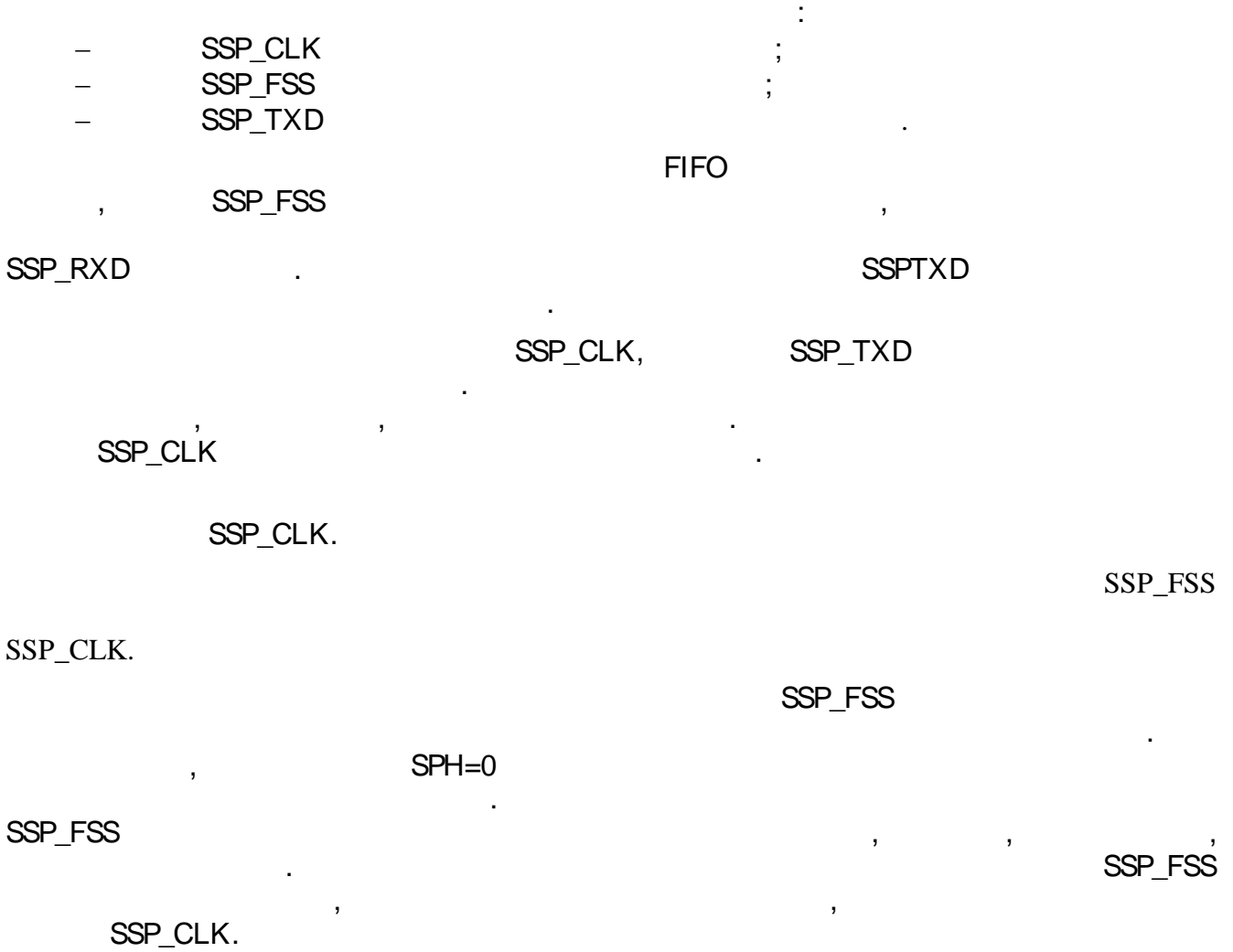
**Рисунок 104. Формат синхронного обмена протокола SPI, SPO=1, SPH=0 (одиночный обмен)**

Примечание – Q



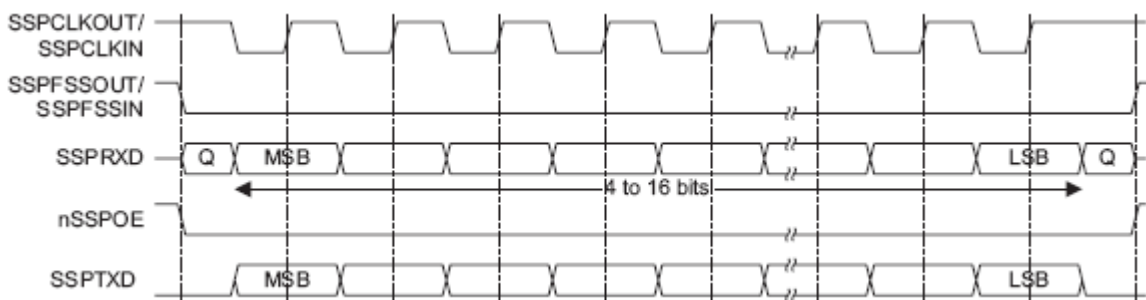
**Рисунок 105. Формат синхронного обмена протокола SPI, SPO=1, SPH=0 (непрерывный обмен)**





**27.6.19 Формат синхронного обмена SPI фирмы Motorola, SPO=1, SPH=1**

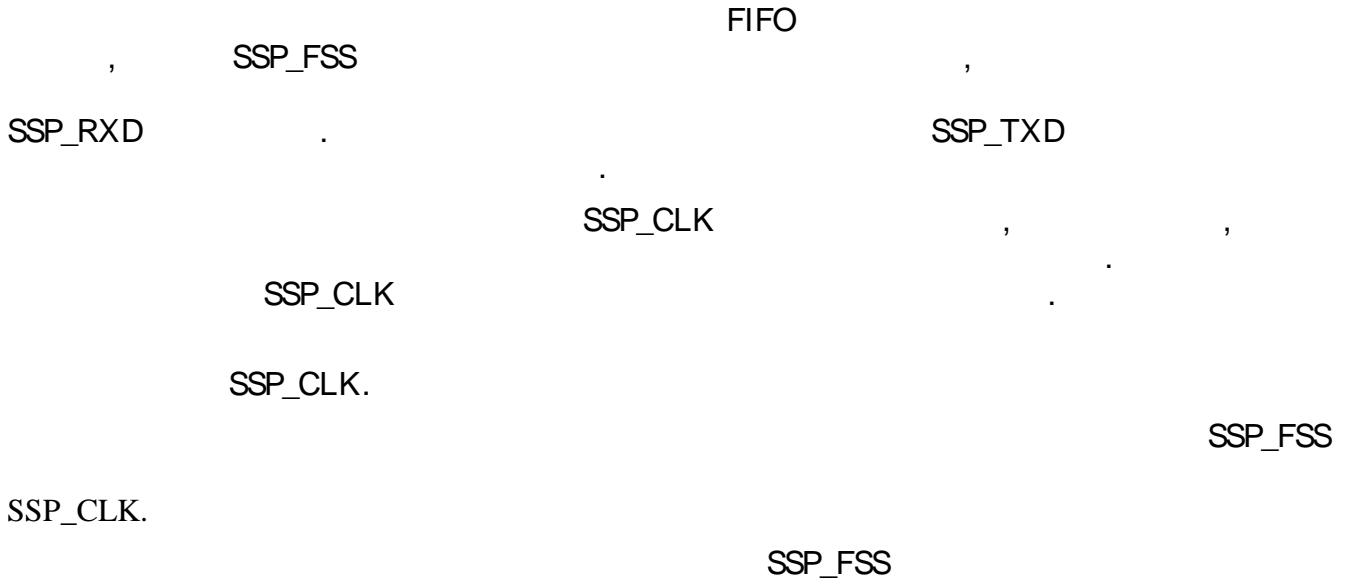
SPH=1 106 - SPI SPO=1,



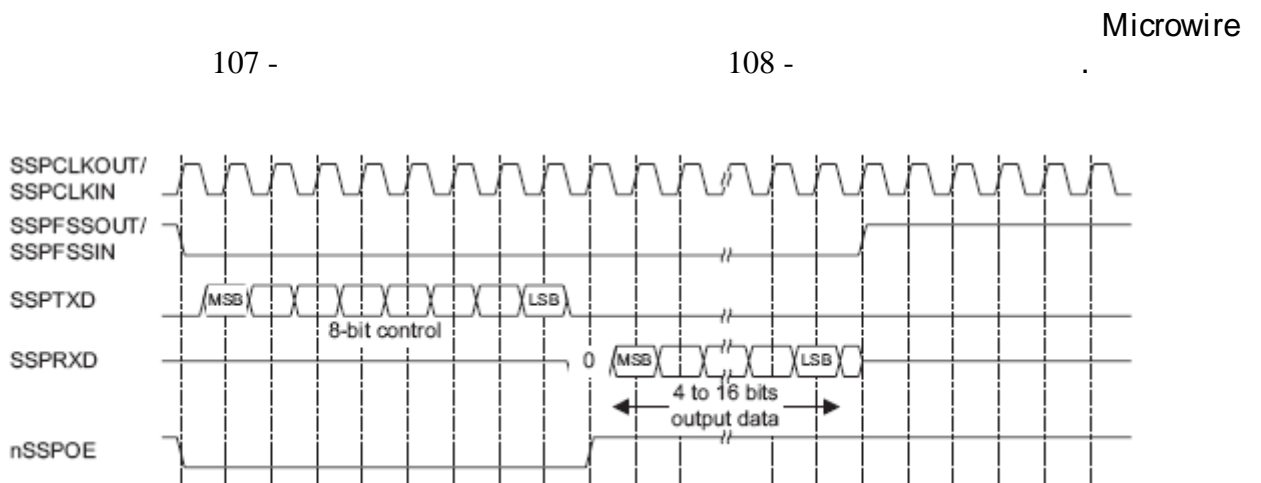
**Рисунок 106. Формат синхронного обмена протокола SPI, SPO=1, SPH=1**

Примечание – Q

- SSP\_CLK ;
- SSP\_FSS ;
- SSP\_TXD



**27.6.20 Формат синхронного обмена Microwire фирмы National Semiconductor**



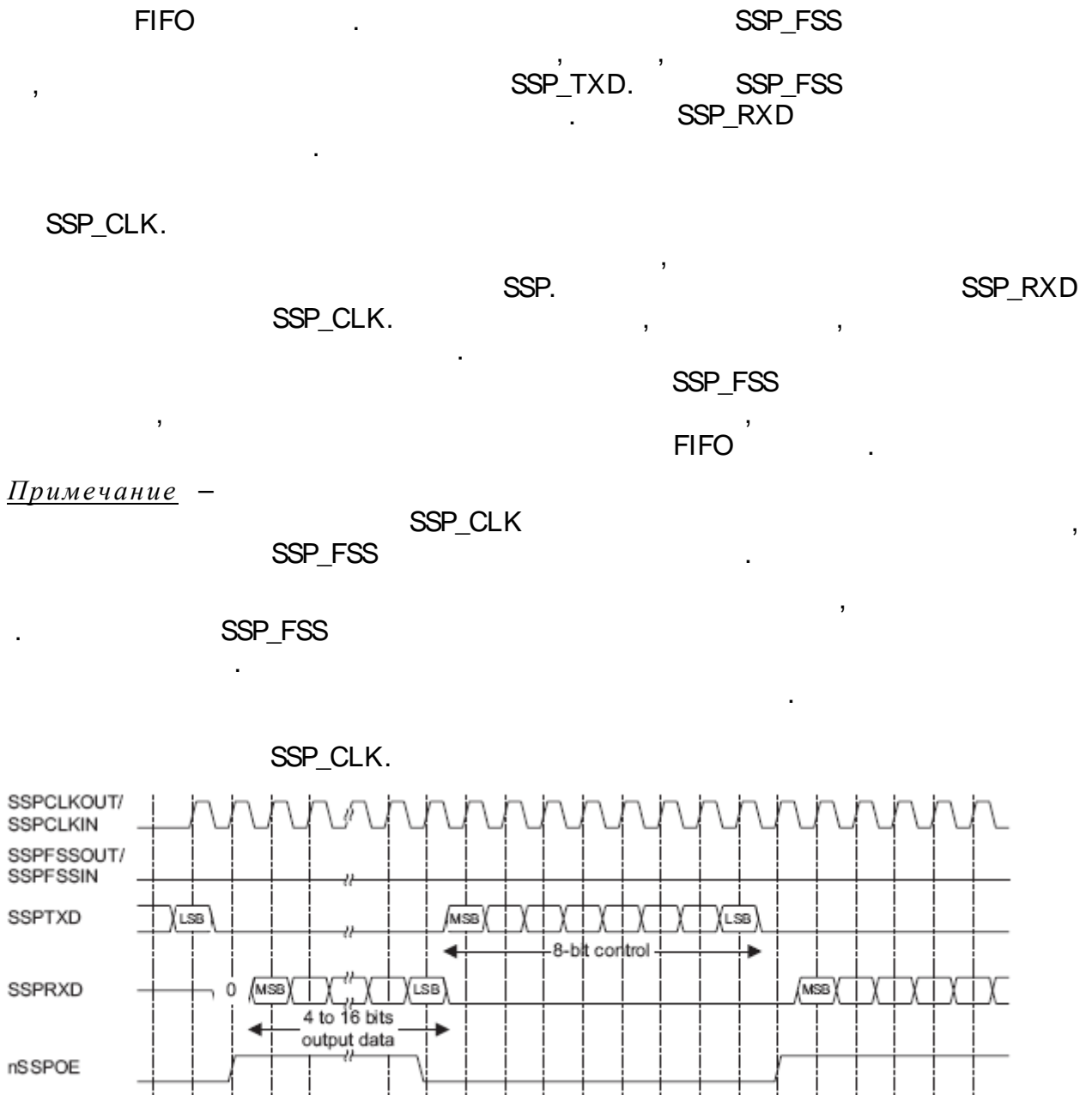
**Рисунок 107. Формат синхронного обмена протокола Microwire (одиночный обмен)**

Microwire SPI,

8-

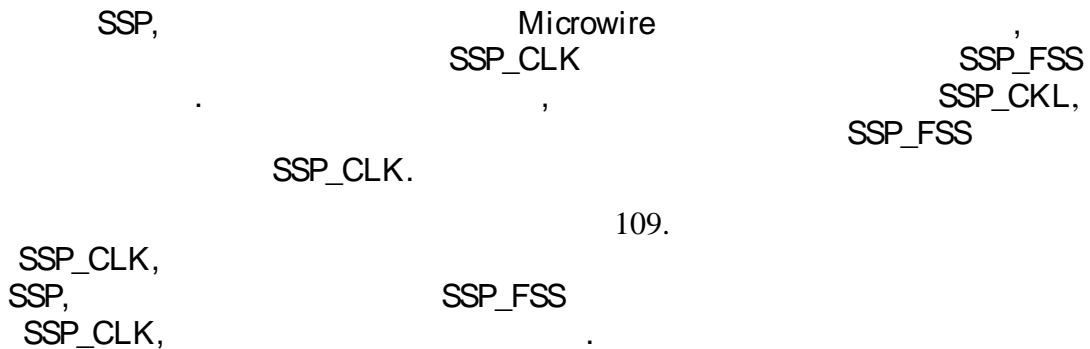
4 16  
13 25

- SSP\_CLK
- SSP\_FSS
- SSP\_TXD



**Рисунок 108. Формат синхронного обмена протокола Microwire (непрерывный обмен)**

**Требования к временным параметрам сигнала SSP\_FSS относительно тактового сигнала SSP\_CLK в режиме Microwire**



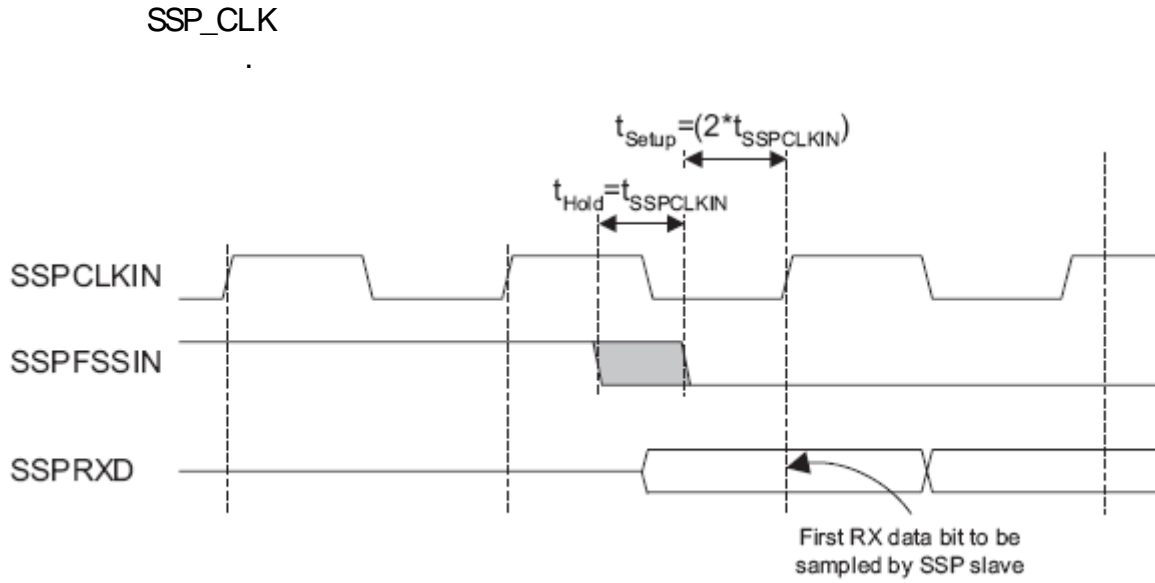
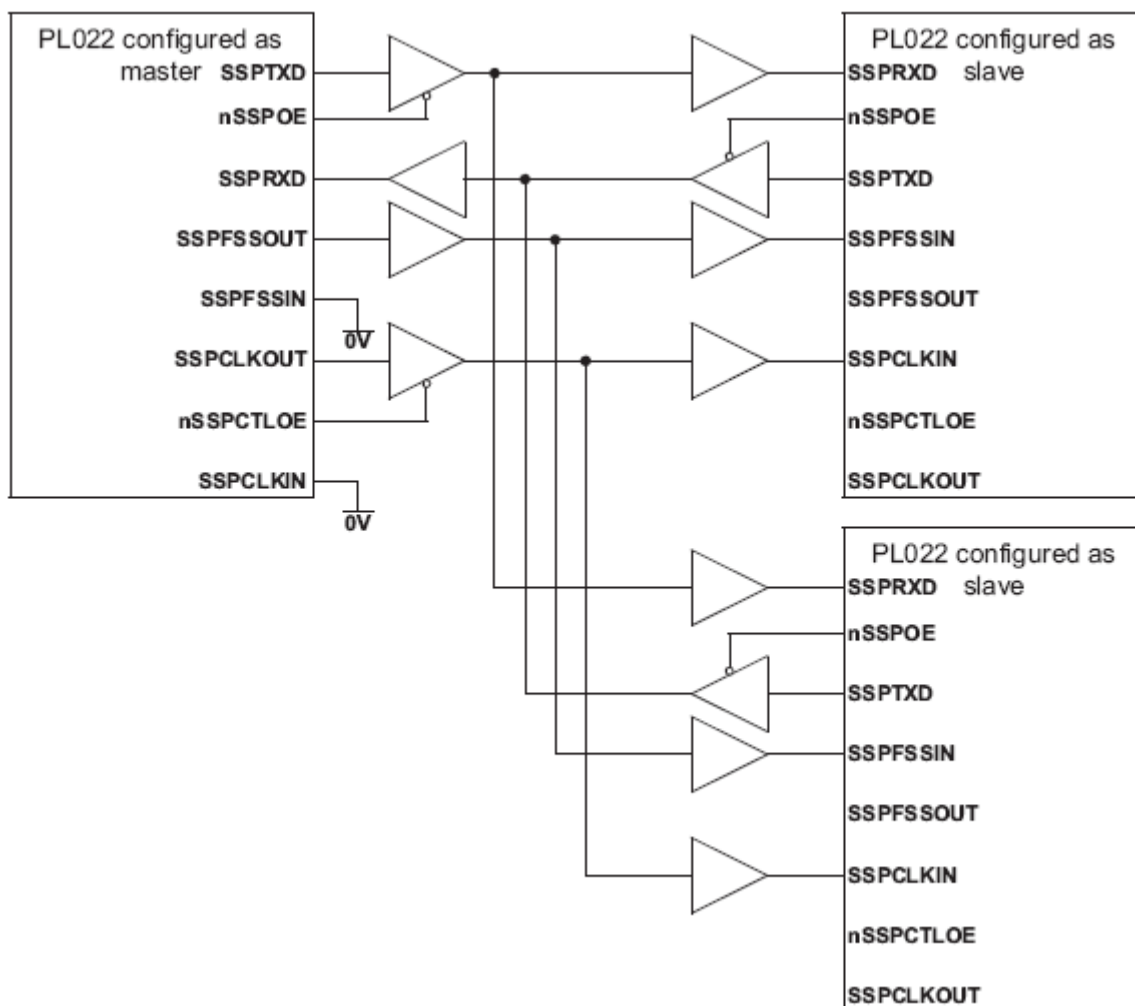


Рисунок 109. Формат Microwire, требования к времени установки и удержания сигнала

### 27.6.21 Примеры конфигурации модуля в ведущем и ведомом режимах

	110,	111	112
SSP	,		
<u>Примечание</u> –		SSP	
« – ».			



**Рисунок 110. Ведущее устройство SSP подключено к двум ведомым**

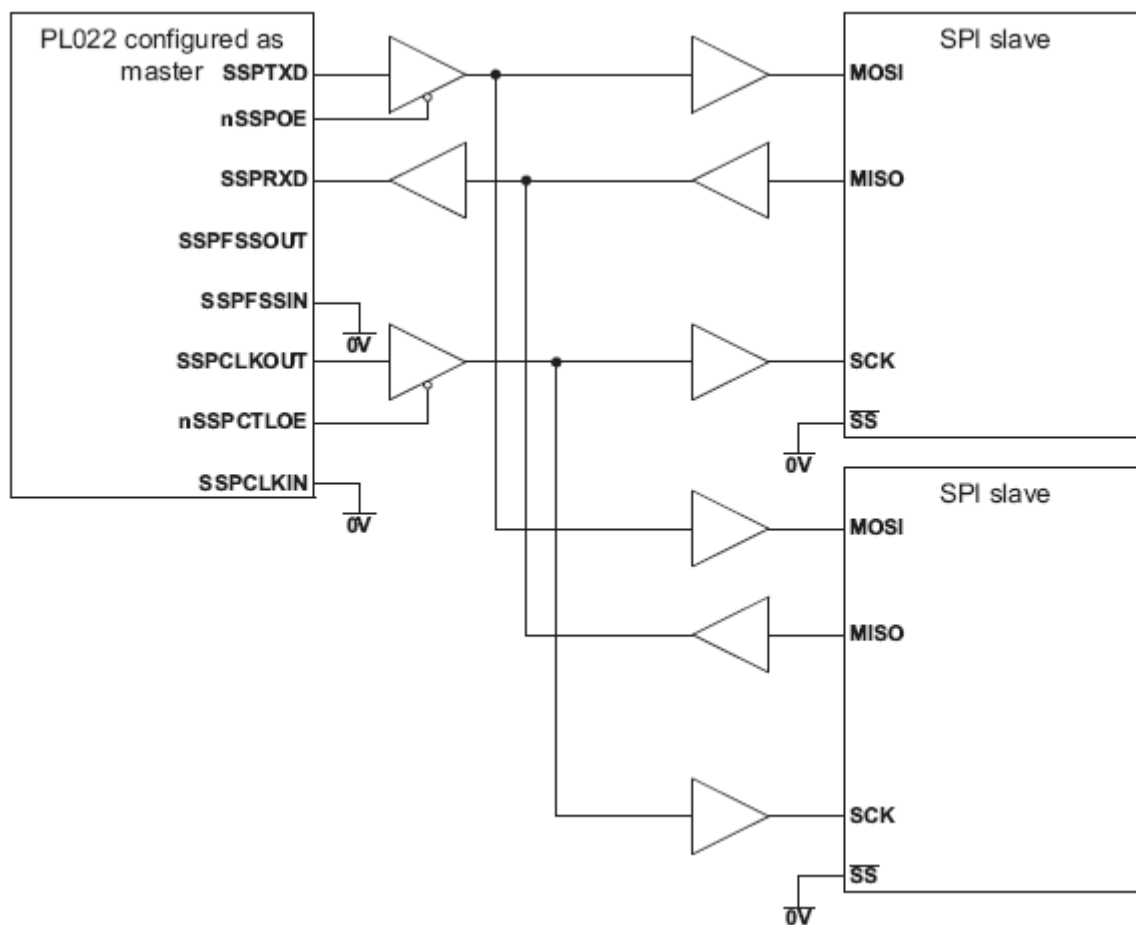
110

SSP,

SSP\_TXD.

SSP\_TXD

SSP\_RXD



**Рисунок 111. Ведущее устройство SSP подключено к двум ведомым, поддерживающим SPI**

111

SSP,

SPI

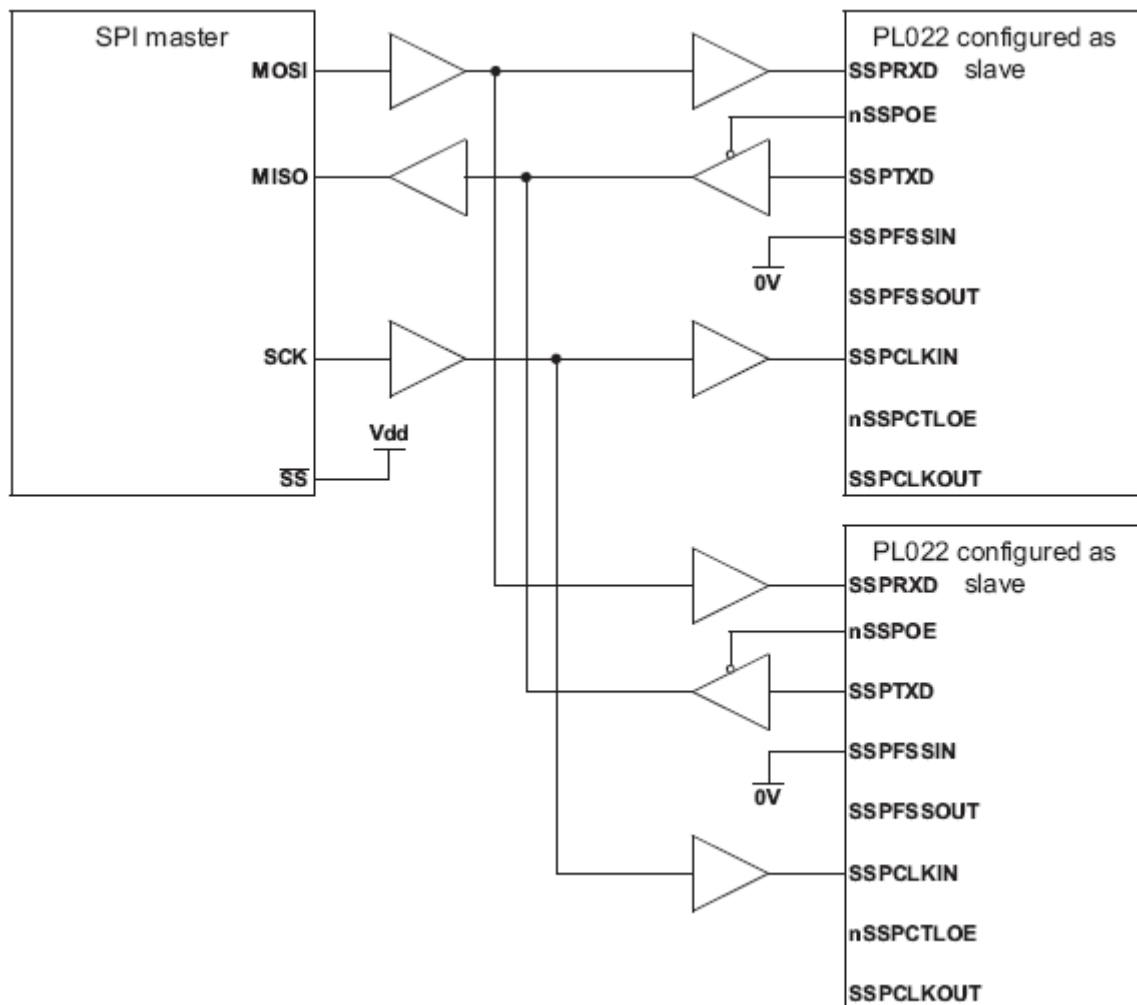
Motorola.

Slave Select (SS).

SSP\_TXD.

SSP\_RXD

MISO.



**Рисунок 112. Ведущее устройство, протокол SPI, подключено к двум ведомым модулям SSP**

Motorola, 112, SSP, SPI, Slave Select (SS), MOSI, SSP\_RXD, SSP\_TXD, SSP\_TXD

**27.6.22 Интерфейс прямого доступа к памяти**

SSP

DMA SSPDMACR.

DMA

:

– SSPRXDMAREQ –

FIFO

– SSPRXDMABREQ –

FIFO

– SSPRXDMACLR –

DMA,

DMA

– SSPTXDMAREQ –

FIFO

– SSPTXDMABREQ –

FIFO

– SSPTXDMACLR –

DMA,

DMA

19

DMA

Примечание –

SSP

DMA

DMACLR.

DMA

DMA

DMA.

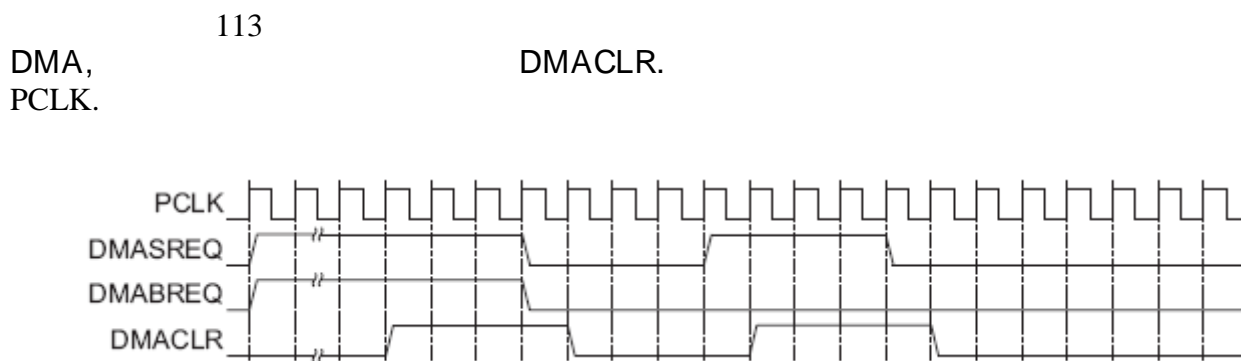
339

DMABREQ.



**Таблица 339 – Параметры срабатывания запросов блочного обмена данными в режиме DMA**

Пороговый уровень	Длина блока обмена данными	
	Буфер передатчика (количество незаполненных ячеек)	Буфер приемника (количество заполненных ячеек)
1/2	4	4



**Рисунок 113. Временные диаграммы обмена в режиме DMA**

## 27.7 Программное управление модулем

### 27.7.1 Общая информация

SSP,  
340.

– +0x028 ... +0x07C +0xFD0 ... +0xFDC  
;  
– +0x080 ... +0x088

### 27.7.2 Описание регистров контроллера SSP

SSP 340.

**Таблица 340 – Обобщенные данные о регистрах модуля SSP**

Базовый адрес	Наименование	Тип	Значение после сброса	Размер, бит	Описание
0x4004_0000	MDR_SSP1				SSP1
0x400A_0000	MDR_SSP2				SSP2
<b>Смещение</b>					
0x000	CR0	RW	0x0000	16	MDR_SSPx->CR0 0
0x004	CR1	RW	0x0	4	MDR_SSPx->CR1 1
0x008	DR	RW	0x-----	16	FIFO ( ) FIFO ( ) MDR_SSPx->DR
0x00C	SR	RO	0x03	3	MDR_SSPx->SR
0x010	CPSR	RW	0x00	8	MDR_SSPx->CPSR
0x014	IMSC	RW	0x0	4	MDR_SSPx->IMSC
0x018	RIS	RO	0x8	4	MDR_SSPx->RIS
0x01C	MIS	RO	0x0	4	MDR_SSPx->MIS
0x020	ICR	WO	0x0	4	MDR_SSPx->ICR
0x024	DMACR	RW	0x0	2	MDR_SSPx->DMACR

*Примечание* – « » : RW – , RO – , WO –

**27.7.3 MDR\_SSPx->CR0**

0

CR0  
SSP.

341.

**Таблица 341 – Формат регистра CR0**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...16	-	
15...8	SCR	SCR : $F\_SSPCLK / (CPSDVR * (1 + SCR))$ , CPSDVR – SSPCPSR), SCR – 0 255
7	SPH	SSPCLKOUT (SPI Motorola). « SPI Motorola»
6	SPO	SSPCLKOUT (SPI Motorola). « SPI Motorola»
5...4	FRF	00 – SPI Motorola; 01 – SSI Texas Instruments; 10 – Microwire National Semiconductor; 11 –
3...0	DSS	: 0000 – 0001 – 0010 – 0011 – 4 0100 – 5 0101 – 6 0110 – 7 0111 – 8 1000 – 9 1001 – 10 1010 – 11 1011 – 12 1100 – 13 1101 – 14 1110 – 15 1111 – 16

**27.7.4 MDR\_SSPx->CR1**

1

CR1  
SSP. 342.

**Таблица 342 – Регистр CR1**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15...4		0
3	SOD	(MS=1). SOD SSP SSP_TXD. 0 – SSP_TXD 1 – SSP_TXD
2	MS	0 – ; 1 – ( );
1	SSE	0 – ; 1 – ;
0	LBM	0 – ; 1 – ;

**27.7.5 MDR\_SSPx->DR**

SSPDR 16

FIFO

FIFO

FIFO

SSP\_TXD

SSPDR

16

Microwire National Semiconductor SSP

FIFO

SSE 0.

SSPDR 343.

**Таблица 343 – Формат регистра DR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15...0	DATA	( ) SSPDR 16

**27.7.6 MDR\_SSPx->SR**

FIFO SSP.

SSPCPSR 344.

**Таблица 344 – Регистр SR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15...5		
4	BSY	0 – SSP ; 1 – SSP /
3	RFF	FIFO ;
2	RNE	FIFO ;
1	TNF	FIFO ;
0	TFE	FIFO ;

### 27.7.7 MDR\_SSPx->CPSR

SSPCPSR

2 254.

SSPCPSR

345

SSPSR.

**Таблица 345 – Регистр CPSR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...8	-	
7... 0	CPSDVSR	2 254.

### 27.7.8 MDR\_SSPx->IMSC

1

0-

IMSC

**Таблица 346 – Регистр IMSC**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...4		
3	TXIM	50 % FIFO
		1 – 0 –
2	RXIM	50 % FIFO
		1 – 0 –
1	RTIM	( FIFO
		). 1 – 0 –
0	RORIM	1 – 0 –

**27.7.9 MDR\_SSPx->RIS**

347

RIS.

**Таблица 347 – Регистр RIS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31... 4		
3	TXRIS	SSPTXINTR
2	RXRIS	SSPRXINTR
1	RTRIS	SSPRTINTR
0	RORRIS	SSPRORINTR

**27.7.10 MDR\_SSPx->MIS**

SSPMIS

**Таблица 348 – Регистр MIS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...4		
3	TXMIS	SSPTXINTR
2	RXMIS	SSPRXINTR
1	RTMIS	SSPRTINTR
0	RORMIS	SSPRORINTR

**27.7.11 MDR\_SSPx->ICR**

0

1

SSPICR

**Таблица 349 – Регистр ICR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31... 2		
1	RTIC	SSPRTINTR
0	RORIC	SSPRORINTR

## 27.7.12 MDR\_SSPx->DMACR

### UARTDMACR

Таблица 350 – Регистр DMACR

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...2		
1	TXDMAE	DMA DMA FIFO
0	RXDMAE	DMA DMA FIFO

## 27.8 Прерывания

- SSPRXINTR – FIFO ;
- SSPTXINTR – FIFO ;
- SSPRORINTR – FIFO ;
- SSPRTINTR – ;
- SSPINTR – SSPRXINTR, SSPTXINTR, SSPRTINTR

SSPIMSC. 1

0–

SSPRXINTR SSPTXINTR

FIFO

SSPRIS,

SSPMIS.

**SSPRXINTR**

FIFO



**SSPTXINTR**

FIFO

SSP.

**SSPRORINTR**

FIFO

**SSPRTINTR**

FIFO

32

SSP\_RXD.

1

RTIC

SSPTICR.

**SSPINTR**

SSPRXINTR, SSPTXINTR, SSPRTINTR SSPRORINTR

## 28 Контроллер MDR\_UART

Asynchronous Receiver-Transmitter) (UART – Universal  
(ENDEC – ENcoder/DEcoder)  
Infra Red) ( ) SIR (SIR – Serial  
Infrared Data Association (IrDA).

### 28.1 Основные сведения

– ;  
– ;  
– 16C650.

#### 28.1.1 Основные характеристики модуля UART

Out – , (16x12) (16x8) (SIR).  
FIFO (First In First  
FIFO

(1x16 – 65535x16).

3.6864

FIFO

FIFO

( CTS, DCD, DSR, RTS, DTR RI).

– : 5, 6, 7 8 ;

– ( , );

– 1 2 ;  
 – 0 UARTCLK/16  
 IrDA SIR :  
 IrDA SIR;  
 (3/16)  
 (1.41 – 2.23 );  
 UARTCLK

### 28.1.2 Программируемые параметры

– ;  
 – ;  
 – ;  
 – ;  
 – FIFO ( –  
 16 , 16 );  
 – FIFO (1/8, 1/4, 1/2, 3/4 7/8);  
 – ( – 1.8432 )  
 1.42 – 2.12 ;

### 28.1.3 Отличия от контроллера UART 16C650

16C650 :  
 – FIFO – 1/8, 1/4,  
 1/2, 3/4 7/8;  
 – FIFO – 1/8, 1/4,  
 1/2, 3/4 7/8;  
 – ;  
 – ;  
 – 16C650 :  
 – ( 1 2  
 );

## 28.2 Функциональные возможности

– ;  
 – ;  
 – ;

FIFO, 16

Модуль приемопередатчика:

— UARTCLK;  
 — UART 16C650.  
 (UARTIBRD) UARTLCR\_H (UARTFBRD).  
 : ( ),  
 ;  
 ; (DMA)  
 — DMA.  
 , FIFO.  
 , FIFO  
 , FIFO  
 : « » (Clear To  
 Send, CTS), « » (Data Carrier Detected, DCD), «  
 » (Data Set Ready, DSR) « » (Ring Indicator, RI),  
 : « » (Request to Send, RTS) « »  
 (Data Terminal Ready, DTR).

nUARTCTS  
 nUARTRTS.

Блок последовательного интерфейса инфракрасной передачи данных

IrDA SIR ENDEC.  
 UARTRXD, nSIROUT SIRIN. UARTRXD  
 (,  
 UARTRXD. SIR ENDEC,  
 IrDA SIR,

## 28.3 Описание функционирования блока UART

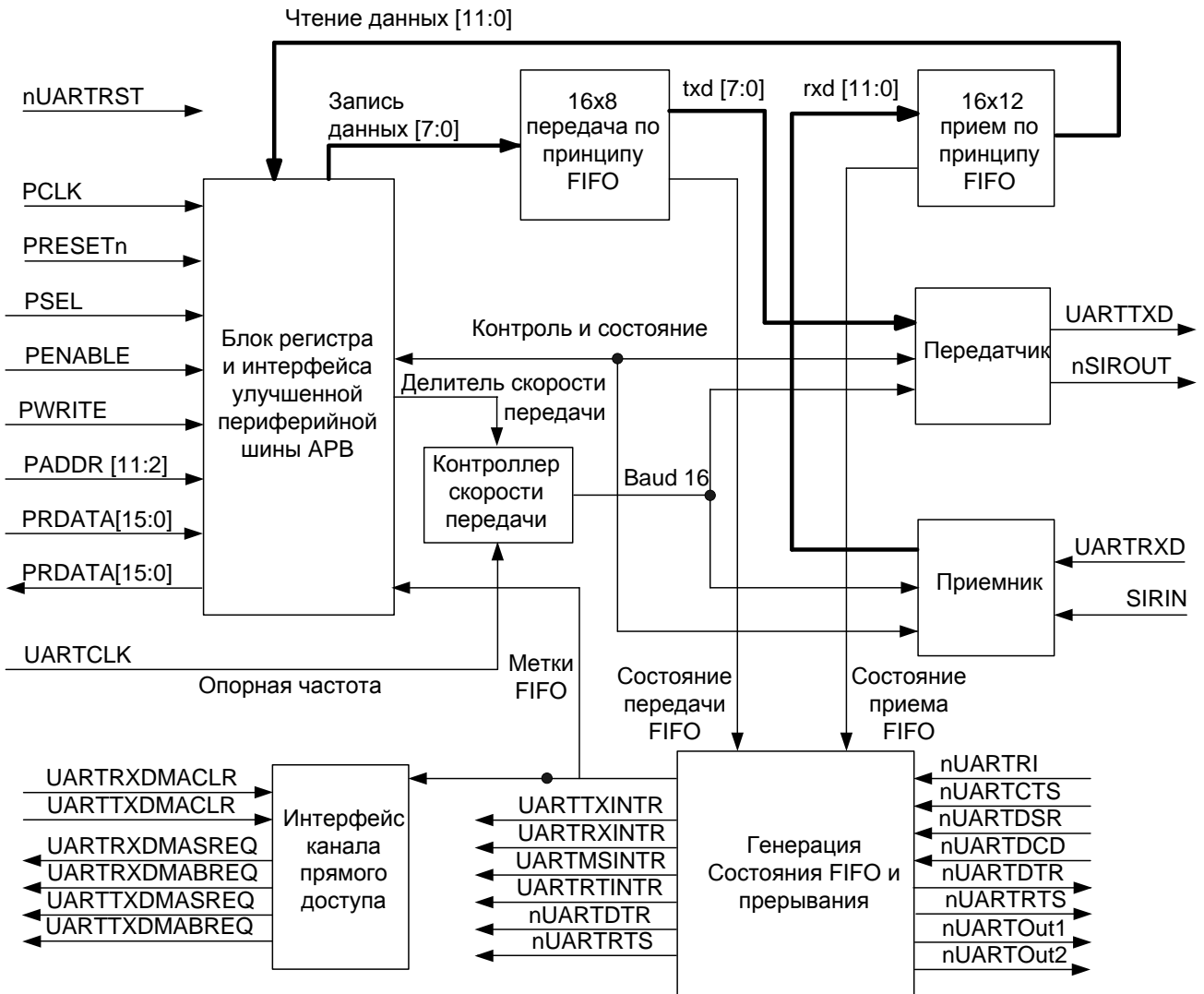


Рисунок 114. Блок-схема универсального асинхронного приёмопередатчика (UART)

### 28.3.1 Генератор тактового сигнала приёмопередатчика

Baud16 IrLPBaud16.

Baud16

UARTCLK

16

IrLPBaud16

**28.3.2 Буфер FIFO передатчика**

«  
АРВ,  
8 , 16 ,  
».  
FIFO

**28.3.3 Буфер FIFO приемника**

«  
12 , 16 ,  
».  
АРВ. FIFO

**28.3.4 Блок передатчика**

**28.3.5 Блок приемника**

**28.3.6 Блок формирования прерываний**

### 28.3.7 Интерфейс прямого доступа к памяти

DMA

DMA.

### 28.3.8 Блок и регистры синхронизации

CPU\_CLK UARTCLK.

CPU\_CLK

UARTCLK,

## 28.4 Описание функционирования ИК кодека IrDA SIR

( 115).

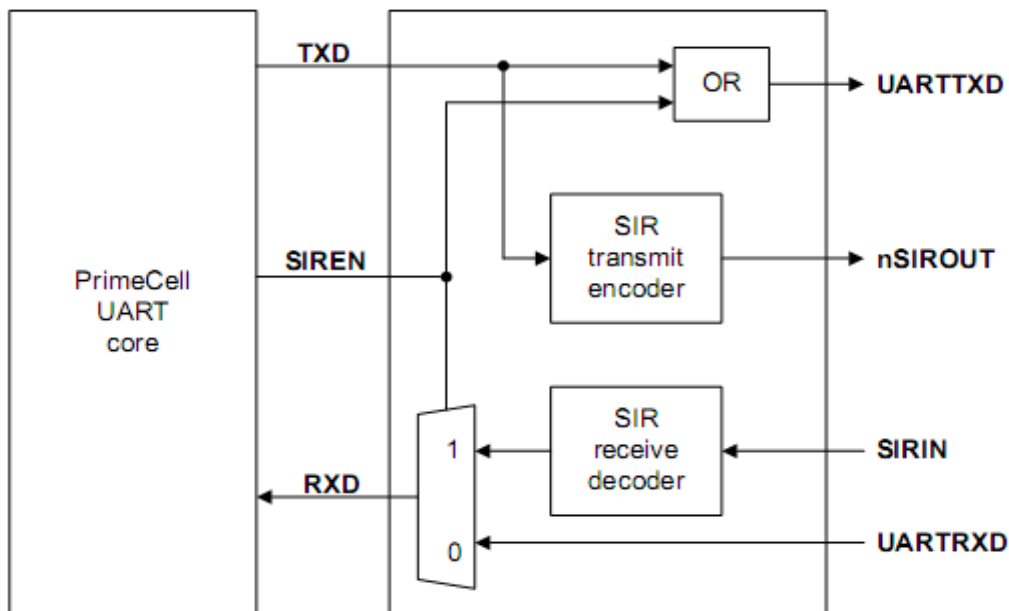


Рисунок 115. Структурная схема кодека IrDA

### 28.4.1 Кодер ИК передатчика

(NRZ).

IrDA SIR

(RZI),

IrDA

Baud16,

3/16

, 3

IrDA

3/16

115200 /

IrLPBaud16

UARTCLK. 1.8432 , ,  
IrLPBaud16 ,  
UARTILPR. -

« »  
Baud16 - Baud16  
UARTCLK. ,

UARTCLK. IrDA SIR,  
13%. UARTCLK 3.6834  
115200 / ,  
9% ,

#### 28.4.2 Декодер ИК приемника

UART.

Примечание –  
SIRIN

- 3/16
- 3/16

Baud16 IrDA;  
IrLPBaud16 IrDA

UART  
:



## 28.5 Описание работы UART

### 28.5.1 Сброс модуля

« ».

### 28.5.2 Тактовые сигналы

UARTCLK

:

$F_{\text{UARTCLK}}(\text{min}) \geq 16 * \text{baud\_rate\_max};$   
 $F_{\text{UARTCLK}}(\text{max}) \leq 16 * 65535 * \text{baud\_rate\_min}.$

UARTCLK,	7.3728	115.34	110	460800
UARTCLK,				

UARTCLK.	UARTCLK	5/3	CPU_CLK	CPU_CLK.
----------	---------	-----	---------	----------

$F_{\text{UARTCLK}} \leq 5/3 * F_{\text{CPU\_CLK}}.$

921600	UART	UARTCLK 14.7456	CPU_CLK
8.85276		UART	
		FIFO.	

### 28.5.3 Работа универсального асинхронного приемопередатчика

UARTLCR.  
APB

30  
:

- UARTLCR\_H - ;
- ;
- ;
- ;
- ;
- ;
- ;
- UARTIBRD - ;
- UARTFBRD - ;

### 28.5.4 Коэффициент деления частоты

16		6	-	22
----	--	---	---	----



Baud16, ( ) Baud16.

SIR IrDA. , UART\_RXD Baud16

16 Baud16 ( ).

( ).

UART\_RXD).

FIFO ( . 351).

**28.5.6 Биты ошибки**

[10...8] FIFO 11 FIFO-

351

**Таблица 351 – Назначение бит слова данных в FIFO-буфере приемника**

Бит буфера FIFO	Назначение
11	
10	– « »
9	
8	
7...0	

**28.5.7 Бит переполнения буфера**

**28.5.8 Запрет буфера FIFO**

FIFO

FIFO

FIFO UART

FIFO

FIFO

**Проверка по шлейфу**

1 LBE

UARTCR.

**28.5.9 Работа кодека ИК обмена данными IrDA SIR**

IrDA SIR.

В режиме IrDA

3/16

nSIROUT

SIRIN.

В режиме IrDA с пониженным энергопотреблением

1.8432 ).  
UARTCR.

IrLPBaud16 ( 1.63

SIRLP

IrDA SIR,

10

IrLPBaud16

UARTILPR.

UARTCLK

$F\_UARTCLK / F\_IrLPBaud16,$

$IrLPBaud16 \quad 1.8432 \quad .$

$1.42 \text{ МГц} < F\_IrLPBaud16 < 2.12 \text{ МГц}.$

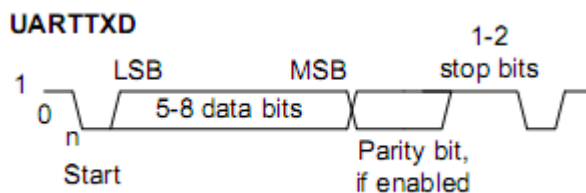
**Проверка по шлейфу**

UARTCR 1    LBE  
 UARTTCR. 1    SIRTEST

nSROUT,

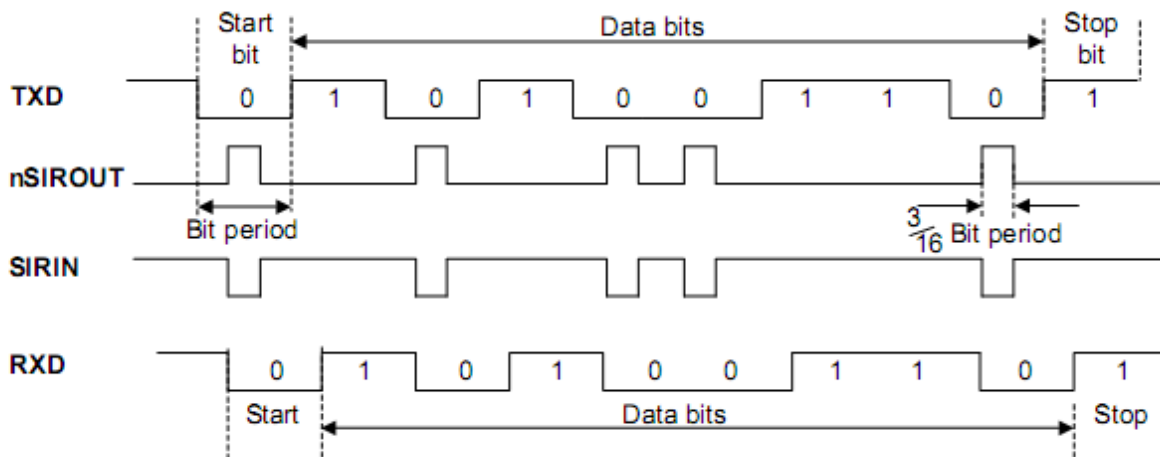
SIRIN.

Примечание –



**Рисунок 117. Кадр передачи данных**

**28.5.10 Модуляция данных IrDA**



**Рисунок 118. Модуляция данных IrDA**

## 28.6 Линии управления модемом

(DTE), (DCE).  
 (114).  
 DTE DCE

Таблица 352 – Назначение управления модемом в режимах DTE и DCE

Сигнал	Назначение	
	Режим оконечного оборудования	Режим оборудования передачи данных
nUARTCTS		
nUARTDSR		
nUARTDCD		-
nUARTRI		-
nUARTRTS		
nUARTDTR		
nUARTOUT1	-	
nUARTOUT2	-	

### 28.6.1 Аппаратное управление потоком данных

(nUARTRTS, nUARTCTS).  
 (119).

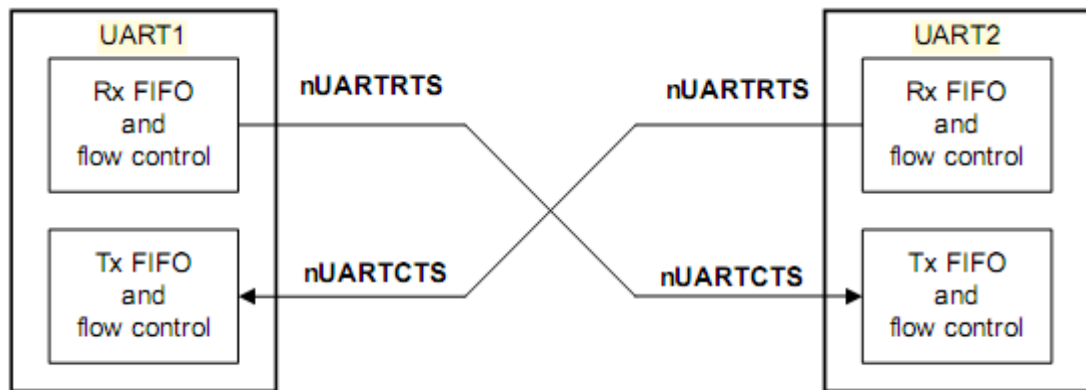


Рисунок 119. Взаимодействие двух устройств последовательной связи с аппаратным управлением потоком данных

RTS, nUARTRTS  
 FIFO  
 CTS,  
 nUARTCTS

RTSEn CTSEn

UARTCR.

353

**Таблица 353 – Режимы управления потоком данных**

CTSEn	RTSEn	Описание
1	1	CTS RTS
1	0	CTS
0	1	RTS
0	0	

*Примечание* –

RTS.

RTSEn

UARTCR

RTS,

### 28.6.2 Управление потоком данных по линии RTS

RTS  
 FIFO  
 nUARTRTS  
 ),  
 nUARTRTS  
 FIFO  
 UART  
 FIFO,

### 28.6.3 Управление потоком данных по линии CTS

nUARTCTS  
 CTS  
 nUARTCTS  
 UART  
 CTS  
 FIFO

## 28.7 Интерфейс прямого доступа к памяти

DMA UARTDMACR.

DMA :

**Для приема:**

UARTRXDMASREQ – , – 12 .  
 UART. , , , .  
 FIFO , , , .  
 UARTRXDMABREQ – , , , .  
 FIFO , , , .  
 FIFO , , , .  
 UARTIFLS. , , , .  
 UARTRXDMACLR – DMA, , , , .

**Для передачи:**

UARTTXDMASREQ – , – .  
 FIFO , , , .  
 UARTTXDMABREQ – , , , .  
 FIFO , , , .  
 FIFO , , , .  
 UARTIFLS. , , , .  
 UARTTXDMACLR – DMA, DMA , , , .

FIFO 19 DMA , , , .

Примечание – UART , , , .

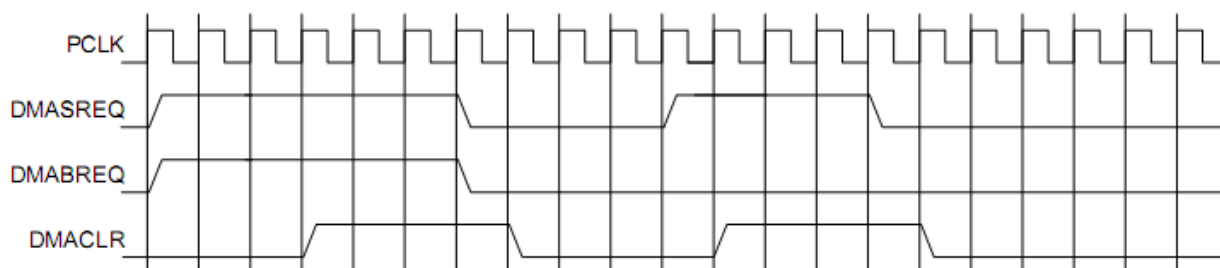


DMA DMACLR. DMA  
 DMA DMA TXDMAE RXDMAE DMA UARTDMACR.  
 FIFO  
 DMA UARTRXDMASREQ UARTTXDMASREQ. FIFO  
 UARTLCLR\_H.  
 FIFO  
 354  
 UARTRXDMABREQ  
 UARTTXDMABREQ

**Таблица 354 – Параметры срабатывания запросов блочного обмена данными в режиме DMA**

Пороговый уровень	Длина блока обмена данными	
	Буфер передатчика (количество незаполненных ячеек)	Буфер приемника (количество заполненных ячеек)
1/8	14	2
1/4	12	4
1/2	8	8
3/4	4	12
7/8	2	14

DMA UARTDMACR DMAONERR,  
 DMA UARTEINTR. DMA –  
 UARTRXDMASREQ UARTRXDMABREQ  
 ( ) UARTEINTR. DMA,  
 UARTEINTR  
 120  
 DMA, DMACLR.  
 CPU\_CLK.  
 DMA DMA



**Рисунок 120. Временные диаграммы одноэлементного и блочного запросов DMA**

## 28.8 Прерывания

11

UARTRXINTR –  
 UARTTXINTR –  
 UARTRTINTR –  
 UARTMSINTR –  
     UARTRIINTR, nUARTRI;  
     UARTCTSINTR, nUARTCTS;  
     UARTDCDINTR, nUARTDCD;  
     UARTDSRINTR, nUARTDSR.  
 UARTEINTR – :  
     UARTOEINTR, ;  
     UARTBEINTR, – ;  
     UARTPEINTR, ;  
     UARTFEINTR, .  
 UARTINTR – UARTRXINTR, UARTTXINTR,  
             UARTRTINTR, UARTMSINTR UARTEINTR.

UARTIMSC. 1

0–

UARTRXINTR UARTTXINTR

UARTRXINTR UARTTXINTR

FIFO

UARTEINTR

UARTRIS,

UARTMIS.

### 28.8.1 UARTMSINTR

(nUARTCTS, nUARTDCD, nUARTDSR, nUARTRI).

1 ( UARTRIS.  
 ) UARTICR.

**28.8.2 UARTRXINTR**

– FIFO :

– FIFO ( , ), ;

**28.8.3 UARTTXINTR**

– FIFO :

– FIFO ( , ), ;

---

FIFO

*Примечание* –

FIFO , FIFO.

**28.8.4 UARTRTINTR**

, FIFO

32 . ( , )

FIFO ), 1

UARTICR.

**28.8.5 UARTEINTR**

– ;

– ;

– ;

– ;

UARTRIS,

UARTMIS.

UARTICR.

7 10.

### 28.8.6 UARTINTR

UARTMSINTR UARTEINTR UARTRXINTR, UARTTXINTR, UARTRTINTR,

## 28.9 Программное управление модулем

### 28.9.1 Общая информация

- 
- 
- 
- ;
- ;
- ) ( 0 ;
- « » 355 ;
  - RW – ;
  - RO – ;
  - WO – .

### 28.9.2 Обобщенные данные о регистрах устройства

**Таблица 355 – Обобщенные данные о регистрах устройства**

Смещение	Наименование	Тип	Значение после сброса	Размер, бит	Описание
0x40030000	MDR_UART1				UART1
0x40038000	MDR_UART2				UART2
0x000	DR	RW	0x----	12/8	MDR_UARTx->DR
0x004	RSR_ECR	RW	0x0	4/0	MDR_UARTx->RSR_ECR /
0x008– 0x014					
0x018	FR	RO	0b-10010----	9	MDR_UARTx->FR
0x01C					
0x020	ILPR	RW	0x00	8	MDR_UARTx->ILPR
0x024	IBRD	RW	0x0000	16	MDR_UARTx->IBRD
0x028	FBRD	RW	0x00	6	MDR_UARTx->FBRD
0x02C	LCR_H	RW	0x00	8	MDR_UARTx->LCR_H
0x030	CR	RW	0x0300	16	MDR_UARTx->CR
0x034	IFLS	RW	0x12	6	MDR_UARTx->IFLS FIFO
0x038	IMSC	RW	0x000	11	MDR_UARTx->IMSC
0x03C	RIS	RO	0x00–	11	MDR_UARTx->RIS
0x040	MIS	RO	0x00–	11	MDR_UARTx->MIS
0x044	ICR	WO	–	11	MDR_UARTx->ICR
0x048	DMACR	RW	0x00	3	MDR_UARTx->DMACR DMA

### 28.9.3 MDR\_UARTx->DR

**В ходе передачи данных:**

FIFO  
, FIFO  
( FIFO).

**В ходе приема данных:**

FIFO  
, 12-  
( FIFO).

UARTDR

UARTRSR/UARTECR

( 356).

**Таблица 356 – Формат регистра UARTDR**

№ бита	Сигнал	Назначение
15...12		
11	OE	1
10	BE	1 FIFO
9	PE	1 UARTLCR_H. EPS SPS FIFO
8	FE	1 FIFO
7...0	DATA	{ }

Примечание –

**28.9.4 MDR\_UARTx->RSR\_ECR**

/

UARTRSR.

UARTDR.

UARTDR

0

357 UARTRSR/UARTECR.

**Таблица 357 – Регистр UARTRSR/UARTECR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
7...4		
3	OE	1 0 UARTECR. FIFO
2	BE	1 UARTECR. (FIFO 0) 1
1	PE	1 EPS SPS 0 FIFO UARTLCR_H (.3-12). UARTECR.
0	FE	1 (UARTECR. 1). FIFO 0

Примечания:

1. UARTRSR, UARTDR, UARTRSR, UARTDR.
2. UARTRSR/UARTECR

**28.9.5 MDR\_UARTx->FR**

TXFF, RXFF BUSY 0,  
TXFE RXFE – 1.  
UARTFR.

**Таблица 358 – Регистр UARTFR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15...9		
8	RI	nUARTRI
7	TXFE	FIFO FEN, UARTLCR_H. FIFO
6	RXFF	FIFO FEN, UARTLCR_H. FIFO
5	TXFF	FIFO FEN, UARTLCR_H. FIFO
4	RXFE	FIFO FEN, UARTLCR_H. FIFO
3	BUSY	UART, FIFO
2	DCD	nUARTDCD
1	DSR	nUARTDSR
0	CTS	nUARTCT



### 28.9.6 MDR\_UARTx->ILPR

$$ILPDVSR = \frac{F\_UARTCLK}{F\_IrLPBaud16} \cdot 1.8432$$

$$1.42 \text{ МГц} < F\_IrLPBaud16 < 2.12 \text{ МГц},$$

$$1,41 - 2,11 \text{ (IrLPBaud16)}.$$

**Таблица 359 – Регистр UARTILPR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
7...0	ILPDVSR	UARTCLK, IrLPBaud16. 0. <i>Примечание</i> – 0 – IrLPBaud16

*Примечание* –

IrLPBaud16. SIRIN IrDA

### 28.9.7 MDR\_UARTx->IBRD

UARTBIRD

**Таблица 360 – Регистр UARTBIRD**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15...0	BAUDDIV_INT	0

### 28.9.8 MDR\_UARTx->FBRD

361

**Таблица 361 – Регистр UARTBFRD**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
5...0	BAUDDIV_FRAC	0

$$BAUDDIV = FUARTCLK / (16 * Baud\_rate),$$

$$FUARTCLK - \text{UART, Baud\_rate} -$$

$$BAUDDIV - BAUDDIV\_INT$$

BAUDDIV\_FRAC,

Примечания:

- UARTIBRD, UARTFBRD ;
- , UARTIBRD, 0, -1, 65535 (2<sup>16</sup> - 1).
- , UARTFBRD ;
- , UARTIBRD 65535 (0xFFFF), UARTFBRD

**Пример. Вычисление коэффициента деления.**

$$FUARTCLK = 4 \cdot 10^6 = 4000000 \text{ / } 230400$$

$$= (4 \cdot 10^6) / (16 \cdot 230400) = 1.085.$$

$$BRDI = 1, BRDF = 0.085.$$

UARTBFRD,

$$m = \text{integer} ((0.085 \cdot 64) + 0.5) = 5.$$

$$= 1 + 5/64 = 1.078.$$

$$= (4 \cdot 10^6) / (16 \cdot 1.078) = 231911 \text{ / } .$$

$$= (231911 - 230400) / 230400 * 100\% = 0.656\%.$$

$$UARTBFRD = 1/64 * 100\% = 1.56\%.$$

m = 1,

64

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

( 362)

UARTCLK = 7.3728

UARTFBRD

**Таблица 362 – Коэффициенты деления при частоте UARTCLK = 7.3728 МГц**

Коэффициент деления	Скорость передачи данных
0x0001	460800
0x0002	230400
0x0004	115200
0x0006	76800
0x0008	57600
0x000C	38400
0x0018	19200
0x0020	14400
0x0030	9600
0x00C0	2400
0x0180	1200
0x105D	110

UARTCLK = 4

**Таблица 363 – Коэффициенты деления при частоте UARTCLK = 4 МГц**

Целая часть	Дробная часть	Требуемая скорость	Реальная скорость	Ошибка, %
0x001	0x05	230400	231911	0.656
0x002	0x0B	115200	115101	0.086
0x003	0x10	76800	76923	0.160
0x006	0x21	38400	38369	0.081
0x011	0x17	14400	14401	0.007
0x068	0x0B	2400	2400	~ 0
0x8E0	0x2F	110	110	~ 0

**28.9.9 MDR\_UARTx->LCR\_H**

364                      UARTLCR\_H                      29    22                      UARTLCR.  
 UARTLCR\_H                      UARTLCR\_H.

**Таблица 364 – Регистр UARTLCR\_H**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15...8		
7	SPS	0 – ; 1 – EPS, SPS                      1,                      EPS=1 –                      (                      EPS=0                      0). PEN
6...5	WLEN	– : 0b11 – 8 0b10 – 7 0b01 – 6 0b00 – 5
4	FEN	FIFO 0 – ; 1 –
3	STP2	0 – ; 1 –
2	EPS	/ 0 – 1 –                      EPS                      ; PEN
1	PEN	0 – ; 1 –
0	BRK	1, UARTTXD 0

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

30- UARTLCR\_H, UARTIBRD    UARTFBRD  
 UARTLCR, , ,  
 UARTLCR\_H.

UARTIBRD /    UARTFBRD  
 UARTLCR\_H.

Примечания:

- UARTIBRD,        UARTFBRD,        UARTLCR\_H;
- UARTFBRD,        UARTIBRD,        UARTLCR\_H.

UARTFBRD) ,        (UARTIBRD  
 -        UARTIBRD (    UARTFBRD),        UARTLCR\_H.

365  
 PEN, EPS    SPS        UARTLCR\_H.

**Таблица 365 – Управление режимом контроля четности**

PEN	EPS	SPS	Бит контроля четности
0	X	X	,
1	1	0	
1	0	0	
1	0	1	1
1	1	1	0

Примечания:

- UARTLCR\_H, UARTIBRD    UARTFBRD        :
- ;
- )        (
- FIFO        :
- BRK;
- FIFO

**28.9.10 MDR\_UARTx->CR**

9 8

9 8

**Таблица 366 – Регистр управления UARTCR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
15	CTSEn	CTS. 1 – , nUARTCTS.
14	RTSEn	RTS. 1 – . - FIFO
13	Out2	nUARTOut2. (DTE) « » (RI)
12	Out1	nUARTOut1. (DTE) « » (DCD)
11	RTS	nUARTRTS
10	DTR	nUARTDTR
9	RXE	1 , SIR, SIREN.
8	TXE	1 , SIR, SIREN. :

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
7	LBE	0 – ; 1 – . : SIREN=1 UARTTCR SIRTEST=1, nSROUT SIRIN. SIRTEST 1 , SIR. , SIRTEST 0. SIRTEST=0, UARTTXD SIR, UARTRXD. UART, 0
6...3		
2	SIRLP	0 – 3/16 ; 1 – ; IrLPBaud16 ,
1	SIREN	0 – IrDA SIR: nSROUT , SIRIN . 1 – nSROUT SIRIN. UARTTXD UARTRXD , UARTEN=0
0	UARTEN	0 – / 1 – . SIR, SIREN

Примечание – ,  
1 TXE UARTEN.  
1 RXE UARTEN.

Примечание –  
:  
;  
/  
; FEN UARTRXD\_H 0;  
UARTCR;

**28.9.11 MDR\_UARTx->IFLS**

FIFO

UARTTXINTR      UARTRXINTR,

UARTIFLS

**Таблица 367 – Регистр UARTIFLS**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...6		
5...3	RXIFLSEL	: b000 = 1/8 b001 = 1/4 b010 = 1/2 b011 = 3/4 b100 = 7/8 b101-b111 =
2...0	TXIFLSEL	: b000 = 1/8 b001 = 1/4 b010 = 1/2 b011 = 3/4 b100 = 7/8 b101-b111 =

FIFO,

FIFO

UART



**28.9.12 MDR\_UARTx->IMSC**

UARTIMSC

**Таблица 368 – Регистр UARTIMSC**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...11		
10	OEIM	UARTOEINTR: 1 – ; 0 –
9	BEIM	UARTBEINTR: 1 – ; 0 –
8	PEIM	UARTPEINTR: 1 – ; 0 –
7	FEIM	UARTFEINTR: 1 – ; 0 –
6	RTIM	UARTRTINTR: 1 – ; 0 –
5	TXIM	UARTTXINTR. 1 – ; 0 –
4	RXIM	UARTRXINTR. 1 – ; 0 –
3	DSRMIM	UARTDSRINTR nUARTDSR: 1 – ; 0 –
2	DCDMIM	UARTDCDINTR nUARTDCD: 1 – ; 0 –
1	CTSMIM	UARTCTSINTR nUARTCTS: 1 – ; 0 –
0	RIMIM	UARTRIINTR nUARTRI: 1 – ; 0 –

**28.9.13 MDR\_UARTх->RIS**

( 3 0), 0.

UARTRIS

**Таблица 369 – Регистр UARTRIS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...11		
10	OERIS	UARTOEINTR
9	BERIS	UARTBEINTR
8	PERIS	UARTPEINTR
7	FERIS	UARTFEINTR
6	RTRIS	UARTRTINTR UARTRTINTR UARTIMSC. UARTRIS UARTMIS
5	TXRIS	UARTTXINTR
4	RXRIS	UARTRXINTR
3	DSRRMIS	UARTDSRINTR nUARTDSR
2	DCDRMIS	UARTDCDINTR nUARTDCD
1	CTSRMIS	UARTCTSINTR nUARTCTS
0	RIRMIS	UARTRIINTR nUARTRI

**28.9.14 MDR\_UARTx->MIS**

( 3 0), 0.

UARTMIS

**Таблица 370 – Регистр UARTMIS**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...11		
10	OEMIS	UARTOEINTR
9	BEMIS	UARTBEINTR
8	PEMIS	UARTPEINTR
7	FEMIS	UARTFEINTR
6	RTMIS	UARTRTINTR
5	TXMIS	UARTTXINTR
4	RXMIS	UARTRXINTR
3	DSRMMIS	nUARTDSR UARTDSRINTR
2	DCDMMIS	nUARTDCD UARTDCDINTR
1	CTSMMIS	nUARTCTS UARTCTSINTR
0	RIMMIS	nUARTRI UARTRIINTR

**28.9.15 MDR\_UARTx->ICR**

1

UARTICR

**Таблица 371 – Регистр UARTICR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...11		
10	OEIC	UARTOEINTR
9	BEIC	UARTBEINTR
8	PEIC	UARTPEINTR
7	FEIC	UARTFEINTR
6	RTIC	UARTRTINTR
5	TXIC	UARTTXINTR
4	RXIC	UARTRXINTR
3	DSRMIC	UARTDSRINTR nUARTDSR
2	DCDMIC	UARTDCDINTR nUARTDCD
1	CTSMIC	UARTCTSINTR nUARTCTS
0	RIMIC	UARTRIINTR nUARTRI

**28.9.16 MDR\_UARTx->DMACR**

UARTDMACR

**Таблица 372 – Регистр UARTDMACR**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...3		
2	DMAONERR	1, DMA UARTRXDMAASREQ UARTRXDMABREQ
1	TXDMAE	DMA 1, DMA FIFO
0	RXDMAE	DMA 1, DMA FIFO

## 29 Контроллер прямого доступа в память MDR\_DMA

### 29.1 Основные свойства контроллера DMA

- 32 DMA;
- DMA ;
- DMA ;
- DMA;
- ;
- - ;
- - ;
- - ;
- DMA ;
- ;
- DMA ;
- ;
- ;
- DMA 1 1024;
- .

### 29.2 Термины и определения

Таблица 373 – Термины и определения

Альтернативная	
С	=1 - DMA 1 =23 - DMA 23
Канал	32. DMA ,
Управляющие данные канала	DMA , <i>Примечание</i> -

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

<b>Цикл DMA</b>	N DMA,
<b>Передача DMA</b>	, DMA,
<b>Пинг-понг</b>	DMA, DMA DMA, DMA, « »
<b>Первичная</b>	chnl_pri_alt_set 0.
<b>R</b>	2, DMA, DMA 2 0 2 10 1 1024
<b>Исполнение с изменением конфигурации</b>	4 DMA, DMA, 4 DMA, DMA, dma_done,

## 29.3 Функциональное описание

( 121)

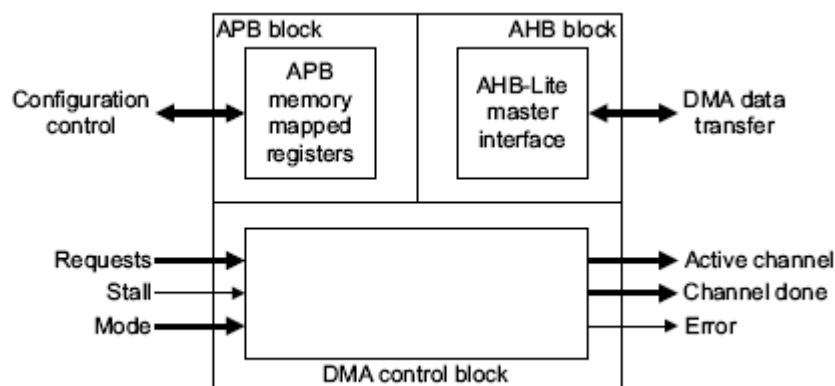


Рисунок 121. Структурная схема контроллера

- , APB;
- , AHB;
- DMA.

### 29.3.1 Распределение каналов DMA

Таблица 374 – Распределение каналов DMA

Номер канала	Источник sreg	Источник reg	Тип	Описание
0	UART1 TX	UART1 TX		DMA UART1
1	UART1 RX	UART1 RX		DMA UART1
2	UART2 TX	UART2 TX		DMA UART2
3	UART2 RX	UART2 RX		DMA UART2
4	SSP1 TX	SSP1 TX		DMA SSP1
5	SSP1 RX	SSP1 RX		DMA SSP1
6	SSP2 TX	SSP2 TX		DMA SSP2
7	SSP2 RX	SSP2 RX		DMA SSP2
8	ADC0_EC	-		DMA 1
9	ADC1_EC	-		DMA 2
10	TIMER1	-		DMA Timer1
11	TIMER2	-		DMA Timer2
12	TIMER3	-		DMA Timer3
13	-	-		
14	-	-		
15	-	-		
16	-	-		
17	-	-		
18	-	-		
19	-	-		
20	-	-		
21	-	-		
22	-	-		



Номер канала	Источник sreg	Источник reg	Тип	Описание
23	-	-		
24	-	-		
25	-	-		
26	-	-		
27	-	-		
28	-	-		
29	-	-		
30	-	-		
31	-	-		

### 29.3.2 Блок, подключенный к шине APB

APB . . . . . 4 . . . . .

### 29.3.3 Блок, подключенный к шине AHB

32- . . . . . « . . . . . » DMA Bus, . . . . .  
 АHB.

### 29.3.4 Управляющий блок DMA

. . . . . :  
 — ;  
 — ;  
 — ;  
 — DMA Bus;  
 — DMA;  
 — DMA;  
 — DMA  
 — DMA :  
 — — ;  
 — — ;  
 — — .

### 29.3.5 Типы передач

АHB, « . . . . . »  
 DMA  
 АHB, . . . . .

### 29.3.6 Разрядность передач данных

8, 16 32  
 HSIZE.

Таблица 375 – Комбинации шины HSIZE

HSIZE[2]*)	HSIZE[1]	HSIZE[0]	Разрядность данных (бит)
0	0	0	8
0	0	1	16
	1	0	32
	1	1	**)

\*) -

\*\*) -

32-

### 29.3.7 Управление защитой данных

AHB-Lite,

HPROT[3:1].

- ;  
- ;  
-

376

HPROT.

Таблица 376 – Режимы защиты данных

HPROT[3] Кэширование	HPROT[2] буферизация	HPROT[1] Привилегированный	HPROT[0] Данные/команда	Описание
-	-	-	1*)	
-	-	0	-	
-	-	1	-	
-	0	-	-	
-	1	-	-	
0	-	-	-	
1	-	-	-	

\*) -

HPROT[0]

DMA

«

».

DMA

DMA.

### 29.3.8 Инкремент адреса

**Таблица 377 – Инкремент адреса**

Разрядность данных	Величина инкремента
8	, ,
16	, ,
32	, ,

Примечание –

, FIFO, «»).

## 29.4 Управление DMA

### 29.4.1 Правила обмена данными

378,

– DMA

– chnl\_enable\_set[C] master\_enable;  
dma\_req[C] dma\_sreq[C]

– chnl\_req\_mask\_set [C];

– int\_test\_en bit[C].

**Таблица 378 – Правила, при которых передача данных по каналам разрешена, и запросы не маскируются**

Правило	Описание
1	dma_active[C] 0, 1 dma_req[C] dma_sreq[C] hclk,
2	1 dma_active[C]
3	1 dma_active[C]
4	DMA, «» », dma_active[C] 1 2 <sup>R</sup> , n_minus_1. «» 1 DMA, 2 <sup>R</sup> , , 2 <sup>R</sup> ( n_minus_1), dma_active[C] 0

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Правило	Описание
5	dma_active[C] 0 hclk dma_active[C] dma_active[ ] 1
6	1 dma_done[ ]
7	dma_req[C] 1 dma_active[C] dma_stall 1,
8	cycle_ctrl 3'b100, 3'b101, 3'b110, 3'b111, dma_done[C] 1
9	dma_done[C], dma_active[ ] cycle_ctrl - dma_stall 0, dma_done[ ] 1 hclk - dma_stall 1, dma_done[ ] 1 hclk dma_stall 0, dma_done[ ] 1 hclk
10	dma_waitonreq[C]
11	dma_waitonreq[C] 1, dma_active[C] 0 : - 2 <sup>R</sup> ( , n_minus_1); - dma_req[C] 0; - dma_sreq[C] 0
12	hclk dma_active[C] 0 dma_stall 1, - dma_active[C] 0 hclk; - 0 dma_stall
13	dma_sreq[C], dma_waitonreq[C] 0
14	dma_sreq[C], chnl_useburst_set[C] 1*)
15	DMA, « », 2 <sup>R</sup> chnl_useburst_set[C] 0, , 2 <sup>R</sup> . « » chnl_useburst_set[C] 0 , , 2 <sup>R</sup> .
16	DMA, « », hclk dma_active[C] 1 dma_sreq[C] dma_waitonreq[C] 1 dma_req[C] 0, DMA . « » hclk dma_active[C] 1 dma_sreq[C] dma_waitonreq[C] 1 dma_req[C] 0, 2 <sup>R</sup>

Правило	Описание
17	<p>DMA, « », hclk dma_active[C] 1, dma_sreq[C] dma_req[C] 1, dma_req[c], 2<sup>R</sup> ( n_minus_1) DMA , . « hclk dma_active[C] 1 dma_sreq[C] dma_req[C] », 1, dma_req[c], 2<sup>R</sup> , , , n_minus_1),</p>
18	<p>chnl_req_mask_set[C] 1, dma_sreq[C] dma_req[C]</p>

\*) \_  
n\_minus\_1 , 2<sup>R</sup>,  
chnl\_useburst\_set dma\_sreq[C]  
dma\_req[C] 1,  
DMA

379.

**Таблица 379 – Правила осуществления DMA передач при «запрещенных» каналах**

Правило	Описание
19	<p>dma_req[C] 1, dma_done[C] 1. ( ) ,</p>
20	<p>dma_sreq[C] 1, dma_done[C] 1 dma_waitonreq[C] 1 chnl_useburst_set[C] 0. ( ) ,</p>
21	<p>dma_active[C] 0</p>

#### 29.4.2 Диаграммы работы контроллера DMA

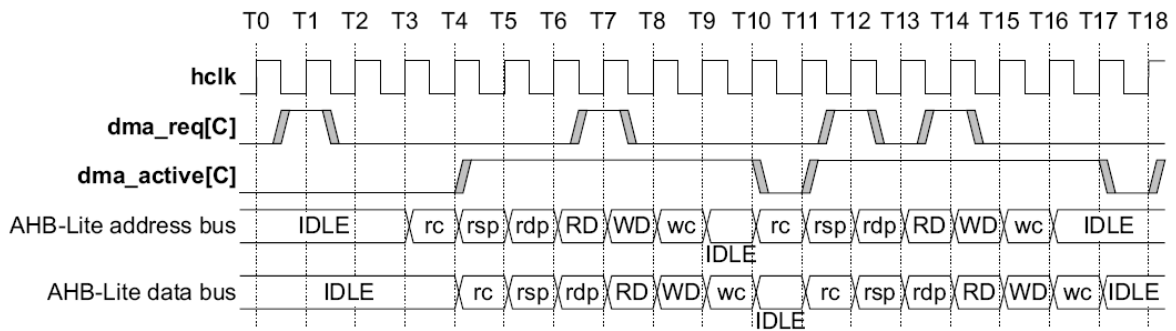
, ( 378):  
– ;  
– ;  
– ;  
– .

*Примечание* – ,  
( 122 – 126), :  
– hready 1;  
– АНВ « » «ОКAY».

**29.4.2.1 Импульсный запрос на обработку**

122

DMA



**Рисунок 122. Диаграмма работы при получении импульсного запроса**

( 122)

**Таблица 380 – Пояснения к диаграмме работы при получении импульсного запроса**

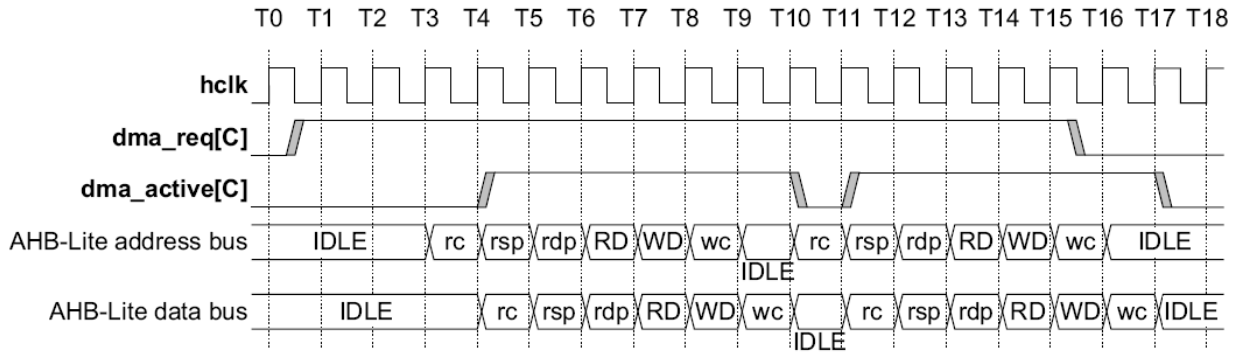
<b>T1</b>	chnl_req_mask_set[C] ( . 18) 1
<b>T4</b>	dma_active[C] ( . 2 3) DMA
<b>T4-T7</b>	rc – , channel_cfg; rsp – , src_data_end_ptr; rdp – , dst_data_end_ptr
<b>T7</b>	dma_active[C] 1 , chnl_req_mask_set[C] ( . 0, 7).
<b>T7-T9</b>	DMA , : RD – ; WD –
<b>T9-T10</b>	, channel_cfg,
<b>T10</b>	wc – , channel_cfg dma_active[C], DMA ( . 4)
<b>T10-T11</b>	dma_active[C] hclk ( . 5)
<b>T11</b>	dma_active[C], 7 ( . 2 3)
<b>T12</b>	dma_active[C] 1 , chnl_req_mask_set[C] ( . 0, 7).
<b>T14</b>	12
<b>T17</b>	dma_active[C], DMA ( . 4)
<b>T17-T18</b>	dma_active[C], hclk ( . 5)

<b>T18</b>	dma_active[C], 2 3)	12	,	( .
------------	------------------------	----	---	-----

**29.4.2.2 Запрос на обработку по уровню**

123

DMA



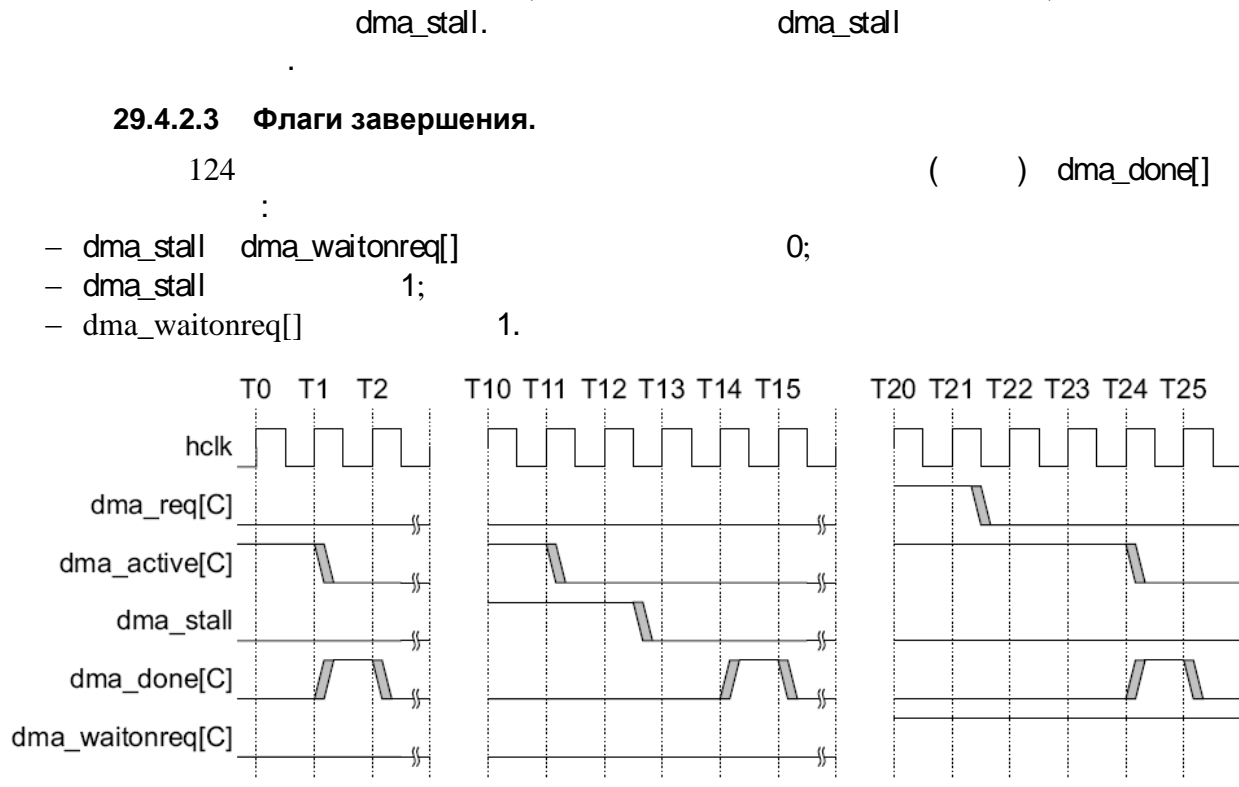
**Рисунок 123. Диаграмма работы при получении запроса на обработку по уровню.**

123

381.

**Таблица 381 – Пояснения к диаграмме работы при получении запроса на обработку по уровню**

<b>T1</b>				( 378, 1)
<b>T4</b>	chnl_req_mask_set[C]	0 ( . 18)	2 3)	DMA
<b>T4-T7</b>	rc – , channel_cfg;			
	rsp – , src_data_end_ptr;			
	rdp – , dst_data_end_ptr			
<b>T7-T9</b>	DMA			
	RD –			
	WD –			
<b>T9-T10</b>				
	wc – , channel_cfg			
<b>T10</b>				
	dma_active[C],			
	DMA ( . 4).			
<b>T10-T11</b>	chnl_req_mask_set[C]	0 ( . 18).	( . 1)	
<b>T11</b>	dma_active[C]	hclk ( . 5)		
<b>T11-T14</b>	dma_active[C]	DMA		
<b>T14-T16</b>	DMA			
<b>T15-T16</b>	DMA			
	dma_req[C]			
<b>T16-T17</b>	channel_cfg			
<b>T17</b>	dma_active[C],			
	DMA ( . 4)			



**Рисунок 124. Диаграммы функционирования dma\_done**

**Таблица 382 – Пояснения функционирования dma\_done, такты от T0 до T2**

<b>T1</b>	dma_active[C], DMA ( . 378, 4)
<b>T1-T2</b>	DMA cycle_ctrl[2] 0, 1 dma_done[C] hclk ( . 8 9). dma_done[C] 0 ( . 6)

**Таблица 383 – Пояснения функционирования dma\_done, такты от T10 до T15**

<b>T11</b>	dma_active[C], DMA ( . 4)
<b>T12-T13</b>	dma_stall
<b>T14-T15</b>	DMA cycle_ctrl[2] 0, 1 dma_done[C] hclk ( . 8 9). dma_done[C] 0 ( . 6)

Примечание к T11:

dma\_done[C], dma\_stall 1  
hclk ( . 9 12).

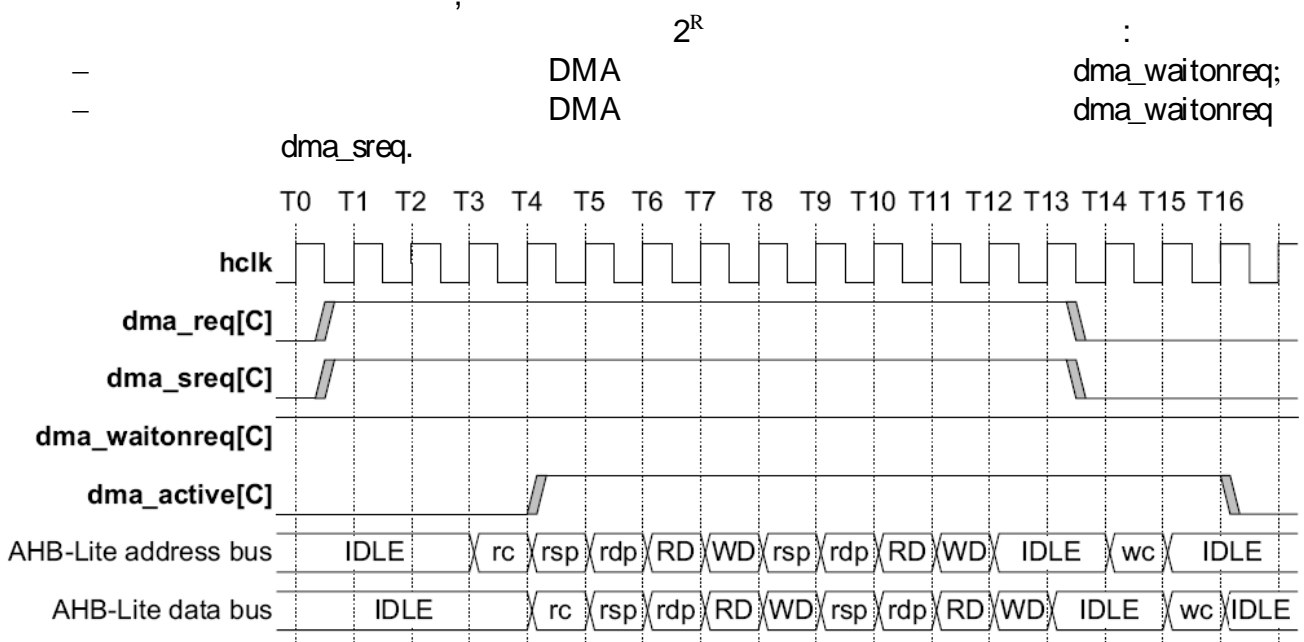


124, 20 25,

**Таблица 384 – Пояснения функционирования dma\_done, такты от T20 до T25**

<b>T20</b>	DMA, dma_waitonreq[C] (0), dma_req[C] (11), dma_active[C] (9)	1
<b>T21-T25</b>	dma_req[C]	
<b>T24</b>	dma_active[C], DMA (4)	
<b>T24-T25</b>	DMA, cycle_ctrl[2] (0), dma_done[C] (8), dma_done[C] (9), dma_done[C] (6)	

**29.4.2.4 Флаги ожидания запроса на обработку**



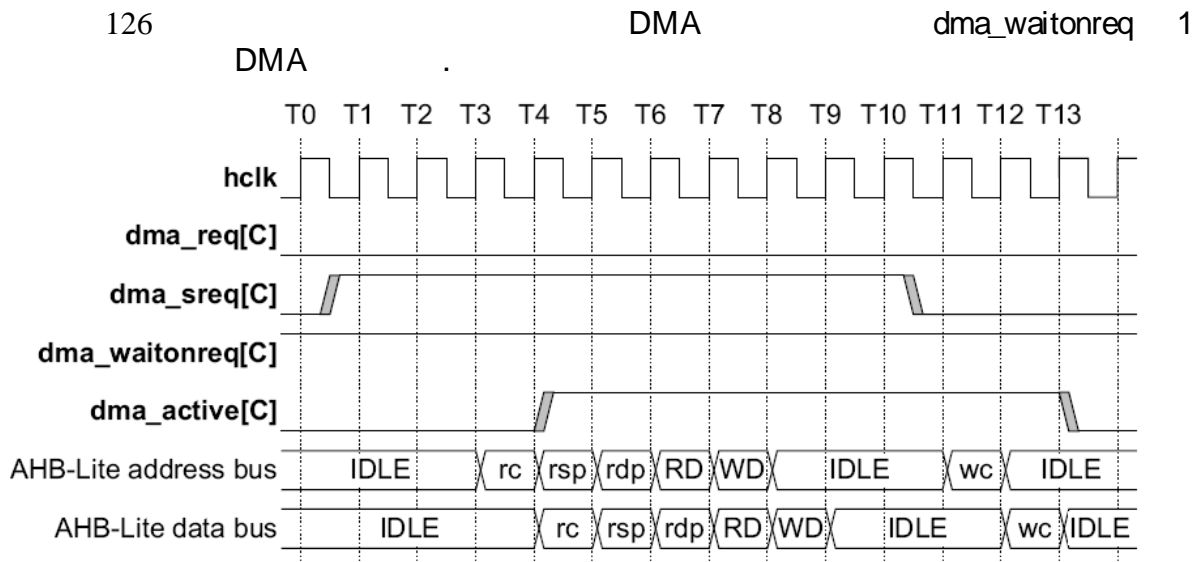
**Рисунок 125. Диаграмма работы контроллера DMA при использовании dma\_waitonreq**

( 125) 385.

**Таблица 385 – Пояснения работы контроллера DMA при использовании dma\_waitonreq**

<b>T0-T16</b>	dma_waitonreq[C] (10)	
<b>T0-T1</b>	chnl_req_mask_set[C] (18)	1
<b>T3-T4</b>	dma_req[C] (16), dma_sreq[C] (17), dma_req[C] (3)	
<b>T4</b>	dma_active[C] (2)	DMA

<b>T4-T7</b>	rc – , channel_cfg; rsp – , src_data_end_ptr; rdp - , dst_data_end_ptr
<b>T7-T9</b>	DMA , :
<b>T9-T11</b> <b>T11-T13</b> <b>T15-T16</b>	RD – ; WD – ; 2 dma_req[C] dma_sreq[C] rsp rdp , channel_cfg,
<b>T16</b>	wc – , channel_cfg dma_active[C], DMA ( . 11). chnl_useburst_set[C] 0, 2 <sup>R</sup> ( . 15)



**Рисунок 126. Работа DMA при использовании dma\_waitonreq совместно с dma\_sreq**

126

386.

**Таблица 386 – Пояснения работы DMA при использовании dma\_waitonreq совместно с dma\_sreq**

<b>T0-T13</b>	( . 10) dma_waitonreq[C]
<b>T0-T1</b>	( . 1) chnl_useburst_set[C] 0 ( . 13 14)
<b>T3-T4</b> <b>T4</b>	dma_sreq[C] ( . 16) dma_active[C] ( . 2 3) DMA
<b>T4-T7</b>	rc – , channel_cfg; rsp – , src_data_end_ptr; rdp - , dst_data_end_ptr

T7-T9	RD – ; WD – .	DMA , :
T10-T11 T12_T13	1 DMA	dma_sreq[], , R=0 ,
T13	wc – , channel_cfg	dma_sreq[C] , channel_cfg, dma_active[C],
	DMA ( . 11)	

### 29.4.3 Правила арбитража DMA

DMA.

4 , R 2; R AHB  
2 R 4,  
16 DMA.

387

**Таблица 387 – Периодичность арбитража в единицах передач по шине AHB**

Значение R	Периодичность арбитража каждые x передач DMA
b0000	1
b0001	2
b0010	4
b0011	8
b0100	16
b0101	32
b0110	64
b0111	128
b1000	256
b1001	512
b1010-b1111	1024

Примечание –

R

$N > 2^R$  ( $N -$  ) ,  $2^R$  N ,  
 $N < 2^R$ . N  $2^R$  ,  
 R 2 « ».

#### 29.4.4 Приоритет

DMA.

—  
—

( )

chnl\_priority\_set.

0

. Таблица 388

DMA

**Таблица 388 – Уровень приоритета каналов DMA**

Уровень приоритета в порядке его уменьшения	Номер канала	Уровень приоритета установленный битом chnl_priority_set
	0	
-	1	
-	2	
.....	.....	.....
-	30	
-	31	
-	0	( )
-	1	( )
-	2	( )
.....	.....	.....
-	30	( )
	31	( )

DMA  
DMA.

127

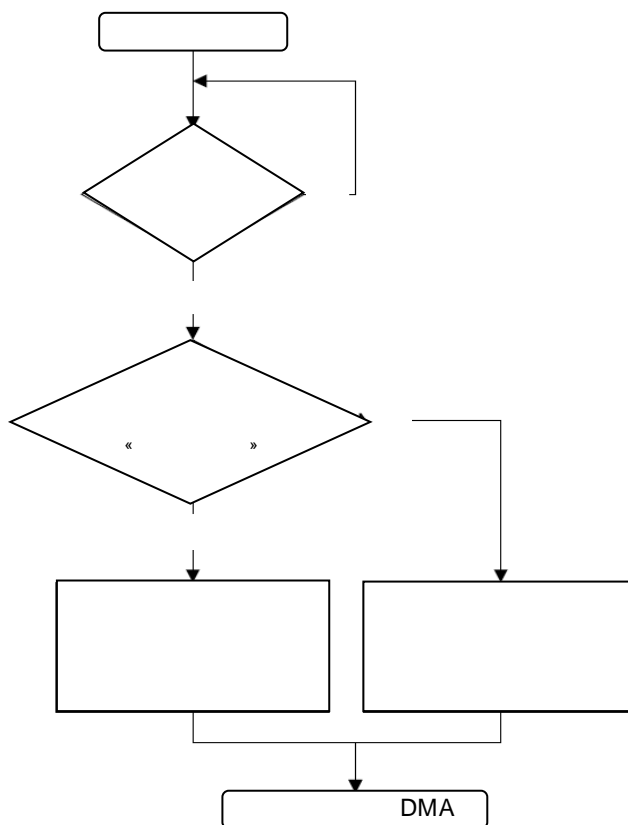


Рисунок 127. Алгоритм выбора следующего канала для обслуживания

#### 29.4.5 Типы циклов DMA

cycle\_ctrl , DMA.

Таблица 389 – Типы циклов DMA

cycle_ctrl	Описание
b000	
b001	DMA
b010	-
b011	« - »
b100	« »
b101	« »
b110	« »
b111	« »

Примечание – cycle\_ctrl ,  
channel\_cfg – « ».

DMA DMA. 2<sup>R</sup> DMA. 2<sup>R</sup> DMA R, 2<sup>R</sup>

- DMA:
- ;
  - ;
  - - ;
  - « - »;
  - « »;
  - « ».

**Недействительный**

« DMA DMA.

**Основной**

- 1. DMA : 0,
- 3. 2<sup>R</sup> .
- 2. :
- ;
- ( ), 1. hclk.
- 3. dma\_done[C] 1 DMA.

**Авто-запрос**

- DMA.
- 1. DMA : 0,
- 3. 2<sup>R</sup> .
- 2. :
- ;



128:

<b>Шаг А</b>	<p align="center">4 DMA.</p> <p align="center">2 DMA.</p> <p align="center">dma_done[C] 1</p> <p align="center">hclk</p>
--------------	--

<b>Шаг В</b>	<p align="center">4 DMA.</p> <p align="center">4 DMA.</p> <p align="center">4 DMA.</p> <p align="center">dma_done[C] 1</p> <p align="center">hclk</p>
--------------	---

D.

<b>Шаг С</b>	<p align="center">2 DMA.</p> <p align="center">dma_done[C] 1</p> <p align="center">hclk</p>
--------------	---

E.

D:

<b>Шаг D</b>	<p align="center">4 DMA.</p> <p align="center">DMA.</p> <p align="center">dma_done[C] 1</p> <p align="center">hclk</p>
--------------	--









**Первичная, копирование А**

4

DMA.

**Шаг А**

**Первичная, копирование В**

4

DMA.

В.

**Шаг В**

В.

**Первичная, копирование С**

4

DMA.

С.

**Шаг С**

С.

**Первичная, копирование D**

4

DMA.

D.

cycle\_ctrl

b000

«                    ».

**Шаг D**

D,

DMA.

dma\_done[C]

1

hclk

**29.4.5.2 Режим работы с периферией «исполнение с изменением конфигурации»**

DMA,

4

DMA, dma\_active[C] 0.

*Примечание* – , DMA, 4

DMA, DMA, dma\_active[C] 0.

: « »

; « »

*Примечание* – N « »

cycle\_ctrl 3'b000. dma\_done[C]

DMA

391 channel\_cfg,

**Таблица 391 – Channel\_cfg для первичной структуры управляющих данных в режиме работы с периферией «Исполнение с изменением конфигурации»**

№ бита	Обозначение	Значение	Описание
<b>Области с константными значениями</b>			
31...30	dst_inc	b'10	
29...28	dst_size	b'10	
27...26	src_inc	b'10	
25...24	src_size	b'10	
17...14	R_power	b'0010	4 DMA
2...0	cycle_ctrl	b'110	« »
<b>Области со значениями, определяемыми пользователем</b>			
23...21	dst_prot_ctrl	-	HPROT
20...18	src_prot_ctrl	-	HPROT
13...4	n_minus_1	N*)	N DMA,
3	next_useburst	-	1 chnl_useburst_set[C]
		1	

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

---

\*) – R\_power 4, N, 4.  
, N/4, ,

( 130)  
« ».

*Инициализация:*

1. A, B, C D:  
cycle\_ctrl=b110,  $2^R=4$ , N=16.  
2. ,  
.



dma\_req[].

**Первичная, копирование из области А памяти**

DMA. 4

**Шаг А**

4 B DMA. B.

**Шаг В**

3 B.

4 C DMA. C.

**Шаг С**

C.

4 D DMA. D.

cycle\_ctrl b000

« ».

**Шаг D**

D, DMA.

dma\_done[C] 1

hclk



#### 29.4.6 Индикация ошибок

```

                                АНВ
:
-                               ;
-                               dma_err      1.
                                dma_err
                                :
-                               chnl_enable_set ;
-                               dma_done[],
                                dma_done[];
-                               dma_done[], 1,
                                dma_done[],
                                dma_done[],

```

#### 29.5 Структура управляющих данных канала

```

:
-                               ;
-                               ,
                                131
                                ,
                                32

```

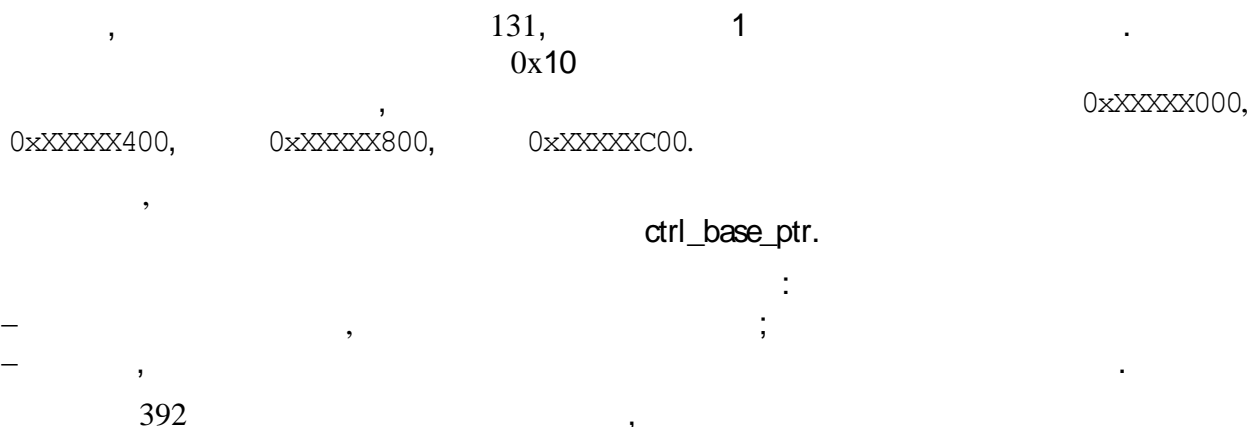
**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

Alternate data structure		Primary data structure	
Alternate_Ch_31	0x3F0	Primary_Ch_31	0x1F0
Alternate_Ch_30	0x3E0	Primary_Ch_30	0x1E0
Alternate_Ch_29	0x3D0	Primary_Ch_29	0x1D0
Alternate_Ch_28	0x3C0	Primary_Ch_28	0x1C0
Alternate_Ch_27	0x3B0	Primary_Ch_27	0x1B0
Alternate_Ch_26	0x3A0	Primary_Ch_26	0x1A0
Alternate_Ch_25	0x390	Primary_Ch_25	0x190
Alternate_Ch_24	0x380	Primary_Ch_24	0x180
Alternate_Ch_23	0x370	Primary_Ch_23	0x170
Alternate_Ch_22	0x360	Primary_Ch_22	0x160
Alternate_Ch_21	0x350	Primary_Ch_21	0x150
Alternate_Ch_20	0x340	Primary_Ch_20	0x140
Alternate_Ch_19	0x330	Primary_Ch_19	0x130
Alternate_Ch_18	0x320	Primary_Ch_18	0x120
Alternate_Ch_17	0x310	Primary_Ch_17	0x110
Alternate_Ch_16	0x300	Primary_Ch_16	0x100
Alternate_Ch_15	0x2F0	Primary_Ch_15	0x0F0
Alternate_Ch_14	0x2E0	Primary_Ch_14	0x0E0
Alternate_Ch_13	0x2D0	Primary_Ch_13	0x0D0
Alternate_Ch_12	0x2C0	Primary_Ch_12	0x0C0
Alternate_Ch_11	0x2B0	Primary_Ch_11	0x0B0
Alternate_Ch_10	0x2A0	Primary_Ch_10	0x0A0
Alternate_Ch_9	0x290	Primary_Ch_9	0x090
Alternate_Ch_8	0x280	Primary_Ch_8	0x080
Alternate_Ch_7	0x270	Primary_Ch_7	0x070
Alternate_Ch_6	0x260	Primary_Ch_6	0x060
Alternate_Ch_5	0x250	Primary_Ch_5	0x050
Alternate_Ch_4	0x240	Primary_Ch_4	0x040
Alternate_Ch_3	0x230	Primary_Ch_3	0x030
Alternate_Ch_2	0x220	Primary_Ch_2	0x020
Alternate_Ch_1	0x210	Primary_Ch_1	0x010
Alternate_Ch_0	0x200	Primary_Ch_0	0x000

Unused	0x00C
Control	0x008
Destination End Pointer	0x004
Source End Pointer	0x000

**Рисунок 131. Карта памяти для 32-х каналов, включая альтернативную структуру**



**Таблица 392 – Разряды адреса, соответствующие элементам структуры управляющих данных**

Количество каналов, используемых в контроллере	Разряды адреса						
	[9]	[8]	[7]	[6]	[5]	[4]	[3:0]
1						A	0x0 0x4 0 8
2					A	[0]	
3-4				A	[1]	[0]	
5-8			A	[2]	[1]	[0]	
9-16		A	[3]	[2]	[1]	[0]	
17-32	A	[4]	[3]	[2]	[1]	[0]	

= 0 ;  
= 1 ;

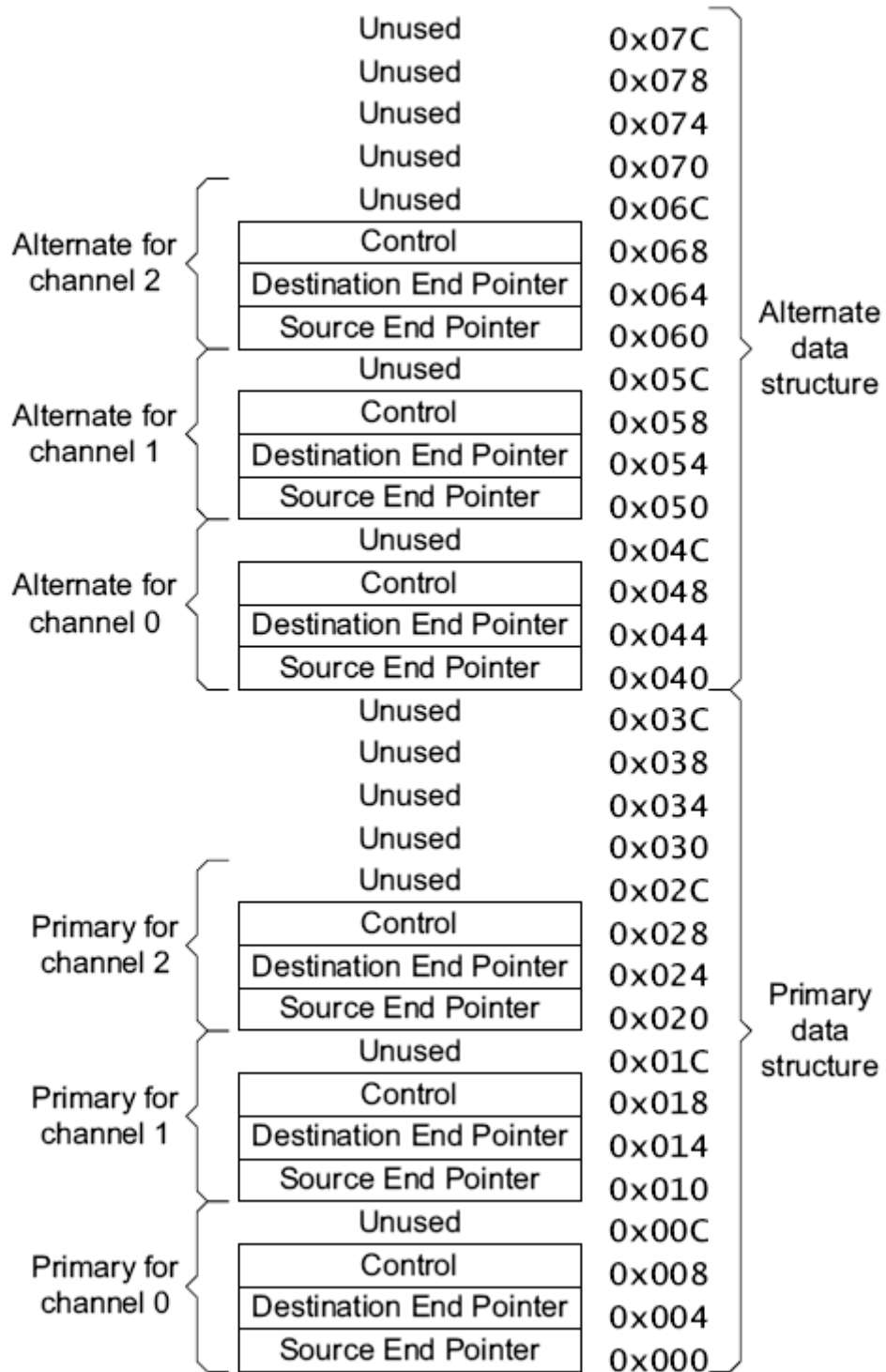
[x:0] DMA.

Address[3:0] :  
0 0 ;  
0 4 ;  
0 8 ;  
0 . ,

Примечание – , alt\_ctrl\_base\_ptr

DMA 132 3

Destination end pointer – ;  
Source end pointer – ;  
Control – .



**Рисунок 132. Карта памяти для трех каналов DMA, включая альтернативную структуру**

128

0x06

0xFFFFFFFF00,

0xFFFFFFFF80.

393

DMA,

**Таблица 393 – Разрешенные базовые адреса**

Количество каналов DMA	Разрешенные значения базового адреса для первичной структуры управляющих данных
17-32	0xXXXXXX000, 0xXXXXXX400, 0xXXXXXX800, 0xXXXXXXC00

32-  
DMA

**Указатель конца данных источника**

src\_data\_end\_ptr

**Таблица 394 – Значения разрядов src\_data\_end\_ptr**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	src_data_end_ptr	

DMA,  
2<sup>R</sup> DMA.

*Примечание –*

**Указатель конца данных приемника**

dst\_data\_end\_ptr

**Таблица 395 – Значения разрядов dst\_data\_end\_ptr**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	dst_data_end_ptr	

DMA,  
2<sup>R</sup> DMA.

*Примечание –*

**Разряды управления**

channel\_cfg

DMA.

**Таблица 396 – Название разрядов области памяти channel\_cfg**

<b>Номер</b>	31	30	29	28	27	26	25	24	23...21	20...18	17...14	13...4	3	2...0
<b>Доступ</b>														
<b>Сброс</b>														
	<b>dst_inc</b>	<b>dst_size</b>	<b>src_inc</b>	<b>src_size</b>	<b>dst_prot_ctrl</b>	<b>Src_prot_ctrl</b>	<b>R_power</b>	<b>n_minus_1</b>	<b>next_useburst</b>	<b>cycle_ctrl</b>				

397

**Таблица 397 – Назначение разрядов channel\_cfg**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31, 30	dst_inc	<p>b00 = ;</p> <p>b01 = (16 );</p> <p>b10 = (32 );</p> <p>b11 = dst_data_end_ptr.</p> <p>= :</p> <p>b00 = ;</p> <p>b01 = ;</p> <p>b10 = ;</p> <p>b11 = dst_data_end_ptr.</p> <p>= :</p> <p>b00 = ;</p> <p>b01 = ;</p> <p>b10 = (32 );</p> <p>b11 = dst_data_end_ptr</p>
29...28	dst_size	<i>Примечание</i> – src_size.
27...26	src_inc	<p>b00 = ;</p> <p>b01 = ;</p> <p>b10 = (32 );</p> <p>b11 = src_data_end_ptr.</p> <p>= :</p> <p>b00 =</p>

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
		b01 = b10 = b11 = src_data_end_ptr. = : b00 = ; b01 = ; b10 = ; b11 = src_data_end_ptr
25...24	src_size	b00 = ; b01 = ( ); b10 = ( ); b11 =
23...21	dst_prot_ctrl	HPROT[3:1], 23 HPROT[3]: 0 = HPROT[3] 0 ; 1 = HPROT[3] 1 ; 22 HPROT[2]: 0 = HPROT[2] 0 ; 1 = HPROT[2] 1 ; 21 HPROT[1]: 0 = HPROT[1] 0 ; 1 = HPROT[1] 1 ;
20...18	src_prot_ctrl	HPROT[3:1], 20 HPROT[3]: 0 = HPROT[3] 0 ; 1 = HPROT[3] 1 ; 19 HPROT[2]: 0 = HPROT[2] 0 ; 1 = HPROT[2] 1 ; 18 HPROT[1]: 0 = HPROT[1] 0 ; 1 = HPROT[1] 1 ;
17...14	R_power	DMA : b0000 - DMA; b0001 - 2 DMA; b0010 - 4 DMA; b0011 - 8 DMA; b0100 - 16 DMA; b0101 - 32 DMA; b0110 - 64 DMA; b0111 - 128 DMA;

**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК, К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
		b1000 - 256 DMA; b1001 - 512 DMA; b1010 - b1111 -- 1024 DMA. , DMA , 1024
13...4	n_minus_1	DMA DMA, DMA. , DMA. 10- 1 DMA. : b0000000000 = 1 DMA; b0000000001 = 2 DMA; b0000000010 = 3 DMA; b0000000011 = 4 DMA; b0000000100 = 5 DMA; b0000000101 = 6 DMA; .... b1111111111 = 1024 DMA. , DMA DMA
3	next_useburst	, chnl_useburst_set[C] 1, « » , DMA, . <i>Примечание</i> – DMA, , chnl_useburst_set[C] 0, DMA , 2 <sup>R</sup> . next_useburst , chnl_useburst_set[C]. DMA « », , , next_useburst: 0 – chnl_useburst_set[C]. chnl_useburst_set[C] 0, DMA « », dma_req[] dma_sreq[], DMA . 1 – chnl_useburst_set[C] 1. DMA





	DMA	2 <sup>R</sup>	DMA	channel_cfg
			2R N	
channel_cfg			dst_size,	src_size.
			,	src_size
n_minus_1,			dst_size,	src_size.
cycle_ctrl	b000,	N	channel_cfg	« ».
			DMA.	

**Вычисление адреса**

n\_minus\_1 DMA, src\_inc,

n\_minus\_1 DMA, dst\_inc,

src\_inc dst\_inc  
:

src\_inc=b00 and dst\_inc=b00

- адрес источника = src\_data\_end\_ptr - n\_minus\_1
- адрес приемника = dst\_data\_end\_ptr - n\_minus\_1.

src\_inc=b01 and dst\_inc=b01

- адрес источника = src\_data\_end\_ptr - (n\_minus\_1<<1)
- адрес приемника = dst\_data\_end\_ptr - (n\_minus\_1<<1).

src\_inc=b01 and dst\_inc=b10

- адрес источника = src\_data\_end\_ptr - (n\_minus\_1<<2)
- адрес приемника = dst\_data\_end\_ptr - (n\_minus\_1<<2).

src\_inc=b11 and dst\_inc=b11

- - адрес источника = src\_data\_end\_ptr
- - адрес приемника = dst\_data\_end\_ptr.

**Таблица 398 – Цикла DMA для 6 слов с пословным инкрементом**

<b>Начальные значения channel_cfg перед циклом DMA</b>				
src_size=b10, dst_inc=b10, n_minus_1=b101, cycle_ctrl=1				
<b>DMA передачи</b>	<b>Указатель конца данных</b>	<b>Счетчик</b>	<b>Отличие<sup>*)</sup></b>	<b>Адрес</b>
	0x2AC	5	0 14	0 298
	0x2AC	4	0 10	0 29
	0x2AC	3	0	0 2A0
	0x2AC	2	0 8	0 2A4
	0x2AC	1	0 4	0 2A8
0x2AC	0	0 0	0 2A	
<b>Конечные значения channel_cfg после цикла DMA</b>				
src_size=b10, dst_inc=b10, n_minus_1=0, cycle_ctrl=0				

\*

dst\_inc.

**Таблица 399 – Цикла DMA для 12 байт с «полусловным» инкрементом**

<b>Начальные значения channel_cfg перед циклом DMA</b>				
src_size=b00, dst_inc=b01, n_minus_1=b1011, cycle_ctrl=1, R_power=b11				
<b>DMA передачи</b>	<b>Указатель конца данных</b>	<b>Счетчик</b>	<b>Отличие<sup>*)</sup></b>	<b>Адрес</b>
	0x5 7	11	0 16	0 5D1
	0x5 7	10	0 14	0 5D3
	0x5 7	9	0 12	0 5D5
	0x5 7	8	0 10	0 5D7
	0x5 7	7	0 E	0 5D9
	0x5 7	6	0 C	0 5DB
	0x5 7	5	0 A	0 5DD
0x5 7	4	0 8	0 5DF	
<b>Значения channel_cfg после 2<sup>R</sup> передач DMA</b>				
src_size=b00, dst_inc=b01, n_minus_1=b011, cycle_ctrl=1, R_power=b11				
<b>DMA передачи</b>	0x5 7	3	0 6	0 5E1
	0x5 7	2	0 4	0 5E3
	0x5 7	1	0 2	0 5E5
	0x5 7	0	0 0	0 5E7
<b>Конечные значения channel_cfg после цикла DMA</b>				
src_size=b00, dst_inc=b01, n_minus_1=0, cycle_ctrl=0 <sup>**</sup> , R_power=b11				

\*

dst\_inc.

\*\*

0 cycle\_ctrl.

DMA

channel\_cfg «

»,

## 29.6 Описание регистров контроллера DMA

**Таблица 400 – Перечень регистров контроллера**

Смещение отн. базового адреса	Наименование	Тип	Значение по сбросу	Описание
0x40028000	MDR_DMA			DMA
0x000	STATUS	RO	0x-0nn0000*)	MDR_DMA->STATUS DMA
0x004	CFG	WO	-	MDR_DMA->CFG DMA
0x008	CTRL_BASE_PTR	R/W	0x00000000	MDR_DMA->CTRL_BASE_PTR
0x00C	ALT_CTRL_BASE_PTR	RO	0x000000nn**)	MDR_DMA->ALT_CTRL_BASE_PTR
0x010	WAITONREQ_STATUS	RO	0x00000000	MDR_DMA->WAITONREQ_STATUS
0x014	CHNL_SW_REQUEST	WO	-	MDR_DMA->CHNL_SW_REQUEST
0x018	CHNL_USEBURST_SET	R/W	0x00000000	MDR_DMA->CHNL_USEBURST_SET
0x01C	CHNL_USEBURST_CLR	WO	-	MDR_DMA->CHNL_USEBURST_CLR
0x020	CHNL_REQ_MASK_SET	R/W	0x00000000	MDR_DMA->CHNL_REQ_MASK_SET

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

0x024	CHNL_REQ_MASK_CLR	WO	-	MDR_DMA->CHNL_REQ_MASK_CLR
0x028	CHNL_ENABLE_SET	R/W	0x00000000	MDR_DMA->CHNL_ENABLE_SET
0x02C	CHNL_ENABLE_CLR	WO	-	MDR_DMA->CHNL_ENABLE_CLR
0x030	CHNL_PRI_ALT_SET	R/W	0x00000000	MDR_DMA->CHNL_PRI_ALT_SET /
0x034	CHNL_PRI_ALT_CLR	WO	-	MDR_DMA->CHNL_PRI_ALT_CLR /
0x038	CHNL_PRIORITY_SET	R/W	0x00000000	MDR_DMA->CHNL_PRIORITY_SET
0x03C	CHNL_PRIORITY_CLR	WO	-	MDR_DMA->CHNL_PRIORITY_CLR
0x040-0x048	-		-	
0x04C	ERR_CLR	R/W	0x00000000	MDR_DMA->ERR_CLR
0x050-0x0DFC	-	-		

\* - DMA,  
\*\* - DMA,

### 29.6.1 MDR\_DMA->STATUS

DMA

402

**Таблица 401 – Статусный регистр DMA**

<b>Номер</b>	31...28	27...21	20...16	15...8	7...4	3...1	0
<b>Доступ</b>	RO	U	RO	U	RO	U	RO
<b>Сброс</b>	0	0	0	0	0	0	0
	<b>test_status</b>	-	<b>chnls_minus1</b>	-	<b>State</b>	-	<b>master_enable</b>

**Таблица 402 – Назначение разрядов регистра dma\_status**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...28	test_status	: 0 0 = ; 0 1 = ; 0 2 – 0 F =
27...21	-	
20...16	chnls_minus1	DMA 1. : b00000 = 1 DMA; b00001 = 2 DMA; b00010 = 3 DMA; ... b11111 = 32 DMA
15...8	-	
7...4	state	: b0000 = ; b0001 = ; b0010 = ; b0011 = ; b0100 = ; b0101 = ; b0110 = DMA; b0111 = ; b1000 = ; b1001 = ; b1010 = « »; b1011-b1111 =
3...1	-	
0	master_enable	: 0 = ; 1 =

**29.6.2 MDR\_DMA->CFG**

DMA

404

**Таблица 403 – Регистр конфигурации DMA**

<b>Номер</b>	31...8	7...5	4...1	0
<b>Доступ</b>	U	WO	U	WO
<b>Сброс</b>	0	0	0	0
	-	<b>chnl_prot_ctrl</b>	-	<b>master_enable</b>

**Таблица 404 – Назначение разрядов регистра dma\_cfg**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8	-	0.
7...5	chnl_prot_ctrl	<p align="right">HPROT[3:1]</p> <p>AHB-Lite:</p> <p>7                    HPROT[3],</p> <p>                         ;</p> <p>6                    HPROT[2],</p> <p>                         ;</p> <p>5                    HPROT[1],</p> <p>                         ;</p> <p><u>Примечания:</u></p> <p>                         [n] = 1,                    HPROT</p> <p>                         1;</p> <p>                         [n] = 0,                    HPROT</p> <p>                         0</p>
4...1	-	0.
0	master_enable	<p align="center">:</p> <p>0 –                    ;</p> <p>1 –</p>

### 29.6.3 MDR\_DMA->CTRL\_BASE\_PTR

*Примечание* –

DMA,

406

ctrl\_base\_ptr.

**Таблица 405 – Регистр базового адреса управляющих данных каналов**

<b>Номер</b>	31...10	9...0
<b>Доступ</b>	R/W	U
<b>Сброс</b>	0	0
	ctrl_base_ptr	-

**Таблица 406 – Назначение разрядов регистра ctrl\_base\_ptr**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...10	ctrl_base_ptr	
9...0	-	0



#### 29.6.4 MDR\_DMA->ALT\_CTRL\_BASE\_PTR

408

**Таблица 407 – Регистр базового адреса альтернативных управляющих данных каналов**

<b>Номер</b>	31... 0
<b>Доступ</b>	RO
<b>Сброс</b>	0
	<b>Alt_ctrl_base_ptr</b>

**Таблица 408 – Назначение разрядов регистра alt\_ctrl\_base\_ptr**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	alt_ctrl_base_ptr	

**29.6.5 MDR\_DMA->WAITONREQ\_STATUS**

dma\_waitonreq[.]

410

**Таблица 409 – Регистр статуса ожидания запроса на обработку каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Номер</b>	RO	.....	RO	RO	RO
<b>Доступ</b>	0	.....	0	0	0
	dma_waitonreg_status for dma_waitnreg [31]	.....	dma_waitonreg_status for dma_waitnreg [2]	dma_waitonreg_status for dma_waitnreg [1]	dma_waitonreg_status for dma_waitnreg [0]

**Таблица 410 – Назначение разрядов регистра dma\_waitonreq\_status**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	dma_waitonreq_status	DMA. <b>При чтении:</b> [C] =0           ,    dma_waitonreq[C]            0 [C] =1           ,    dma_waitonreq[C]            1

**29.6.6 MDR\_DMA->CHNL\_SW\_REQUEST**

DMA.

412

**Таблица 411 – Регистр программного запроса на обработку каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	WO	.....	WO	WO	WO
<b>Сброс</b>	0	.....	0	0	0
	<b>chnl_sw_request for channel [31]</b>	.....	<b>chnl_sw_request for channel [2]</b>	<b>chnl_sw_request for channel [1]</b>	<b>chnl_sw_request for channel [0]</b>

**Таблица 412 – Назначение разрядов регистра chnl\_sw\_request**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_sw_request	<p align="center">DMA.</p> <p align="center">DMA</p> <p><b>При записи:</b></p> <p>[C] = 0 , DMA</p> <p>[C] = 1 , DMA</p> <p align="right">DMA</p>

**29.6.7 MDR\_DMA->CHNL\_USEBURST\_SET**

dma\_req[]  
dma\_sreq[]

414

**Таблица 413 – Регистр установки пакетного обмена каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	R/W	.....	R/W	R/W	R/W
<b>Сброс</b>	0	.....	0	0	0
	<b>chnl_useburst_set for channel [31]</b>	.....	<b>chnl_useburst_set for channel [2]</b>	<b>chnl_useburst_set for channel [1]</b>	<b>chnl_useburst_set for channel [0]</b>

**Таблица 414 – Назначение разрядов регистра chnl\_useburst\_set**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_useburst_set	<p align="right">DMA</p> <p>dma_sreq[]</p> <p><b>При чтении:</b></p> <p>[C] = 0 , DMA DMA dma_req[], dma_sreq[]</p> <p align="right">2<sup>R</sup></p> <p>[C] = 1 , DMA DMA dma_req[], dma_req[]</p> <p align="right">2<sup>R</sup></p> <p><b>При записи:</b></p> <p>[C] = 0 chnl_useburst_clr 0;</p> <p>[C] = 1 DMA, 2<sup>R</sup> dma_sreq[].</p>

(N)  $2^R$ ,  $2^R$ , chnl\_useburst\_set 0.  
dma\_sreq[] dma\_req[].

Примечание – channel\_cfg N,  $2^R$ ,  
chnl\_useburst\_set,  
dma\_req[].

next\_useburst « channel\_cfg, chnl\_useburst\_set [C] 1  
DMA, ».

### 29.6.8 MDR\_DMA->CHNL\_USEBURST\_CLR

chnl\_useburst\_clr. dma\_sreq[] 416

**Таблица 415 – Регистр сброса пакетного обмена каналов**

Номер	31	.....	2	1	0
Доступ	WO	.....	WO	WO	WO
Сброс	0	.....	0	0	0
	chnl_useburst_clr for channel [31]	.....	chnl_useburst_clr for channel [2]	chnl_useburst_clr for channel [1]	chnl_useburst_clr for channel [0]

**Таблица 416 – Назначение разрядов регистра chnl\_useburst\_clr**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_useburst_clr	<p align="center">DMA dma_sreq[].</p> <p><b>При записи:</b></p> <p>[C] = 0 chnl_useburst_set</p> <p>[C] = 1 dma_sreq[];</p> <p align="center">DMA, dma_sreq[].</p>

**29.6.9 MDR\_DMA->CHNL\_REQ\_MASK\_SET**

DMA dma\_sreq[] dma\_req[]  
 dma\_sreq[] dma\_req[]  
 418 chnl\_req\_mask\_set.

**Таблица 417 – Регистр маскирования запросов на обслуживание каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	R/W	.....	R/W	R/W	R/W
<b>Сброс</b>	0	.....	0	0	0
	chnl_reg_mask_set for dma_reg [31] and dma_sreg [31]	.....	chnl_reg_mask_set for dma_reg [2] and dma_sreg [2]	chnl_reg_mask_set for dma_reg [1] and dma_sreg [1]	chnl_reg_mask_set for dma_reg [0] and dma_sreg [0]

**Таблица 418 – Назначение разрядов регистра chnl\_req\_mask\_set**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_req_mask_set	<p align="center">DMA dma_sreq[] dma_req[]</p> <p><b>При чтении:</b>                      [C] = 0 DMA, DMA ;                      [C] = 1 DMA, DMA ;</p> <p><b>При записи:</b>                      [C] = 0                      chnl_req_mask_clr ;                      [C] = 1                      DMA, dma_sreq[] dma_req[].</p>



**29.6.11 MDR\_DMA->CHNL\_ENABLE\_SET**

DMA.

DMA.

422

chnl\_enable\_set.

**Таблица 421 – Регистр установки разрешения каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	WO	.....	WO	WO	WO
<b>Сброс</b>	0	.....	0	0	0
	chnl_enable_set for channel [31]	.....	chnl_enable_set for channel [2]	chnl_enable_set for channel [1]	chnl_enable_set for channel [0]

**Таблица 422 – Назначение разрядов регистра chnl\_enable\_set**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_enable_set	<p align="center">DMA</p> <p><b>При чтении:</b>                      [C] = 0 , DMA ;                      [C] = 1 , DMA .</p> <p><b>При записи:</b>                      [C] = 0                      chnl_enable_clr ;                      [C] = 1 DMA .</p>



**29.6.12 MDR\_DMA->CHNL\_ENABLE\_CLR**

DMA.

424

chnl\_enable\_clr.

**Таблица 423 – Регистр сброса разрешения каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	WO	.....	WO	WO	WO
<b>Сброс</b>	0	.....	0	0	0
	chnl_enable_clr for channel 31	.....	chnl_enable_clr for channel 2	chnl_enable_clr for channel 1	chnl_enable_clr for channel 0

**Таблица 424 – Назначение разрядов регистра chnl\_enable\_clr**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_enable_clr	<p align="center">DMA.</p> <p><b>При записи:</b>                      [C] = 0                      chnl_enable_set                      ;                      [C] = 1                      DMA .</p> <p><i>Примечание</i> – DMA,</p> <p>- DMA;                      - channel_cfg cycle_ctrl                      b000;                      - AHB-Lite</p>

**29.6.13 MDR\_DMA->CHNL\_PRI\_ALT\_SET**

/

DMA

DMA (DMA).

426

chnl\_pri\_alt\_set.

**Таблица 425 – Регистр установки первичной/альтернативной структуры управляющих данных каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	R/W	.....	R/W	R/W	R/W
<b>Сброс</b>	0	.....	0	0	0
	chnl_pri_alt_set for channel [31]	.....	chnl_pri_alt_set for channel [2]	chnl_pri_alt_set for channel [1]	chnl_pri_alt_set for channel [0]

**Таблица 426 – Назначение разрядов регистра chnl\_pri\_alt\_set**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_pri_alt_set	<p align="center">DMA,</p> <p><b>При чтении:</b></p> <p>[C] = 0 , DMA ;</p> <p>[C] = 1 , DMA ;</p> <p><b>При записи:</b></p> <p>[C] = 0 chnl_pri_alt_clr [C] 0;</p> <p>[C] = 1 DMA ;</p> <p><i>Примечание –</i> chnl_pri_alt_set[C] : DMA DMA « »;</p>

		-	DMA	DMA
		« - »;		
		-	DMA	
		DMA :		
		- « - »;		
		-	«	
		»;		
		-	«	
		;		

#### 29.6.14 MDR\_DMA->CHNL\_PRI\_ALT\_CLR

/

DMA

428

chnl\_pri\_alt\_clr.

**Таблица 427 – Регистр сброса  
первичной/альтернативной структуры  
управляющих данных каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	WO	.....	WO	WO	WO
<b>Сброс</b>	0	.....	0	0	0
	<b>chnl_pri_alt_clr for channel [31]</b>	.....	<b>chnl_pri_alt_clr for channel [2]</b>	<b>chnl_pri_alt_clr for channel [1]</b>	<b>chnl_pri_alt_clr for channel [0]</b>

**Таблица 428 – Назначение разрядов регистра chnl\_pri\_alt\_clr**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...0	chnl_pri_alt_clr	<p>DMA.</p> <p>При записи:</p> <p>[C] = 0 chnl_pri_alt_set ;</p> <p>[C] = 1 DMA .</p>

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

		<p><u>Примечание</u> – chnl_pri_alt_clr[C]</p> <p>- 4- DMA : DMA « »; DMA DMA « - »; DMA DMA : - « - » - « » - « »</p>
--	--	--

**29.6.15 MDR\_DMA->CHNL\_PRIORITY\_SET**

DMA. DMA.  
430 chnl\_priority\_set.

**Таблица 429 – Регистр установки приоритета каналов**

<b>Номер</b>	31	.....	2	1	0
<b>Доступ</b>	R/W	.....	R/W	R/W	R/W
<b>Сброс</b>	0	.....	0	0	0
	chnl_priorit_set for channel [31]	.....	chnl_priority_set for channel [2]	chnl_priority_set for channel [1]	chnl_priority_set for channel [0]

**Таблица 430 – Назначение разрядов регистра chnl\_priority\_set**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...0	chnl_priority_set	<p align="center">DMA, DMA.</p> <p><b>При чтении:</b> [C] = 0 , DMA ; [C] = 1 , DMA</p> <p><b>При записи:</b> [C] = 0 chnl_priority_dr ; [C] = 1 C DMA ;</p>



**29.6.17 MDR\_DMA->ERR\_CLR**

dma\_err 0. dma\_err.  
434 err\_clr.

**Таблица 433 – Регистр сброса флага ошибки**

	31...1	0
	U	R/W
C	0	0
	-	err_clr

**Таблица 434 – Назначение разрядов регистра err\_clr**

№ бита	Функциональное имя бита	Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений
31...1	-	0
0	err_clr	<p align="center">0,</p> <p align="center">( ) dma_err.</p> <p><b>При чтении:</b></p> <p align="center">[C] = 0 , dma_err 0;</p> <p align="center">[C] = 1 , dma_err 1.</p> <p><b>При записи:</b></p> <p align="center">[C] =0 . dma_err</p> <p align="center">[C] =1 ; ( ) dma_err 0.</p> <p><u>Примечание</u> – dma_err AHB-Lite, ( dma_err)</p>

## 30 Прерывания и исключения

Inactive – Active Pending

Pending –

Pending

Active –

Active

Active Pending –

pending

### 30.1 Типы исключений

#### 30.1.1 RESET

RESET

RESET

RESET,  
RESET

privileged thread

#### 30.1.2 NON MASKABLE INTERRUPT (NMI)

(NMI)

- 2.

Примечание –

1986 9

NMI

–

–

RESET.

#### 30.1.3 Hard Fault

Hard Fault

-1,

Hard fault

#### 30.1.4 Memory Management fault

Memory Management fault

MPU



EXECUTE NEVER (XN), MPU

**30.1.5 Bus Fault**

**30.1.6 Usage Fault**

USAGE FAULT

:

—

;

—

;

—

;

—

:

halfword word;

—

**30.1.7 SVCcall**

Supervisor Call (SVCALL)

SVC

SVC.

**30.1.8 PendSV**

PendSV

PendSV

**30.1.9 SysTick**

SysTick

SysTick.

**30.2 Прерывания (IRQ)**

IRQ –

**Таблица 435 – Различные типы исключений**

Номер исключения	Номер IRQ	Тип	Приоритет	Адрес вектора обработчика (смещение)	Активация
1	-	RESET	-3,	0x0000_0004	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Номер исключения	Номер IRQ	Тип	Приоритет	Адрес вектора обработчика (смещение)	Активация
2	-14	NMI	-2	0x0000_0008	
3	-13	Hard Fault	-1	0x0000_000C	-
4	-12	Memory Management Fault		0x0000_0010	
5	-11	Bus Fault		0x0000_0014	/
6	-10	Usage Fault		0x0000_0018	
7-10	-	-	-		-
11	-5	SVCall		0x0000_002C	
12-13	-	-	-		-
14	-2	PendSV		0x0000_0038	
15	-1	SysTick		0x0000_003C	
16	0	IRQ		0x0000_0040	

, RESET,

Privileged ,

### 30.3 Обработчики исключений

:

Процедуры обработки прерываний (Interrupt Service Routines – ISRs)

IRQ0 IRQ31 ISR.

Обработчики ошибок (Fault Handlers)

Hard fault, memory management fault, usage fault bus fault.

Системные обработчики (System handlers)

NMI, PendSV, SVCall SysTick.

### 30.4 Таблица векторов

RESET

133

1,

Thumb



## **30.5 Приоритеты исключений**

- 
- , RESET Hard Fault.
- , 0.
- 0 7.
- RESET, Hard Fault NMI, ,
- ,
- .
- ,
- ,

### **30.5.1 Группировка приоритетов прерываний**

NVIC

– ;

– .

,

.

,

.

## **30.6 Вход в обработчик и выход из обработчика**

:

### **30.6.1 Приоритетное прерывание**

– ), (

« , ».

, « ,

».

### **30.6.2 Возврат**

– , - ;

– ;

— exception). (late-arriving

### 30.6.3 Передача управления без восстановления контекста (tail-chaining)

### 30.6.4 Запоздавшее исключение (late-arriving exception)

### 30.6.5 Вход в процедуру обработки исключения

— (thread mode);

— R0-R3, R12;  
— ;  
— PSR;  
— LR.

1, STKALIGN (CCR)

LR

EXC\_RETURN,

« (late-arriving exception)».

### 30.6.6 Возврат из обработчика исключения

(handler mode)  
 EXC\_RETURN PC:  
 – POP PC;  
 – BX ;  
 – LDR LDM PC  
 EXC\_RETURN LR  
 EXC\_RETURN  
 EXC\_RETURN[3:0]  
 436.  
 EXC\_RETURN [31:4] 0xFFFFFFFF.  
 PC

**Таблица 436 – Возврат из обработчика исключения**

EXC_RETURN[3:0]	Описание
bXXX0	
b0001	MSP. MSP
b0011	
b01X1	
b1001	MSP. MSP
b1101	PSP. PSP
b1X11	

### 30.7 Обработка отказов

— :  
— :  
— ;  
— ;  
— BX;  
— (Non-Executable – XN);  
— MPU

#### 30.7.1 Типы отказов

437

Таблица 437 – Отказы

Отказ	Обработчик	Наименование бита регистра	Регистр отказа
		VECTTBL	« »
		FORCED	
:		-	« », »
-		IACCVIOL	
-		DACCVIOL	
-		MSTKERR	
-		MUNSKERR	
:		-	« », »
-		STKERR	
-		UNSTKERR	
-		IBUSERR	
		PRECISERR	
		IMPRECISERR	
		NOCP	« », »
		UNDEFINSTR	
*)		INVSTATE	
EXC_RETURN		INVPC	
		UNALIGNED	
0		DIVBYZERO	

\* –

### 30.7.2 Эскалация отказов и тяжелые отказы

fault) (hard

“SCB->SHP[x]”.

“SCB->SHCSR”.

(escalation).

Reset NMI.

Reset,

NMI,

### 30.7.3 Регистры состояния и адреса отказа

**Таблица 438 – Регистры состояния и адреса отказа**

Обработчик	Регистр состояния	Регистр адреса	Описание регистров
	HFSR	-	“ ”
	MMFSR	MMFAR	“ ”
	BFSR	BFAR	“ ”
	UFSR	-	“ ”



### 30.7.4 Блокировка

## 30.8 Управление электропитанием

Cortex-M3 (

- Deep Sleep;
- Sleep;
- Standby.

SLEEPDEEP SCR ( .“ ”).

### 30.8.1 Переход в режим пониженного энергопотребления

### 30.8.2 Ожидание прерывания

WFI (wait for interrupt)

### 30.8.3 Ожидание события

WFE (wait for event)

### 30.8.4 Переход в режим ожидания по выходу из обработчика исключения (режим Sleep)

SLEEPONEXIT SCR 1,

### 30.8.5 Выход из состояния ожидания

#### 30.8.5.1 Выход из ожидания по команде WFI и в режиме Sleep

PRIMASK 1, FAULTMASK – 0.

PRIMASK 0.

#### 30.8.5.2 Выход из ожидания по команде WFE

SEVONPEND SCR 1,

SCR “

### 30.8.6 Рекомендации по программированию режима энергопотребления

ANSI C  
WFI WFE. CMSIS  
:

```
void __WFE(void) // Wait for Event
void __WFI(void) // Wait for Interrupt
```

IRQ0 IRQ31

**Таблица 439 – Формирование прерывания с IRQ0 до IRQ31**

Прерывания	Блок	Принцип формирования
IRQ0	CAN1	CAN. GLB_INT_EN RX_INT_EN[31:0] RX_INT[31:0] EX_INT_EN[31:0] EX_INT[31:0] ERR_INT_EN (ACKERR FRAMEERR CRCERR BSERR BITERR) ERR_OVER_INT_EN REC > CAN_ERR_MAX TEC > CAN_ERR_MAX
IRQ1	CAN2	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Прерывания	Блок	Принцип формирования
IRQ2	USB	USB Host HostSOFSent, HostConnEvent, HostResume, HostTransDone. USB Slave SlaveNAKSent SlaveSOFRXed SlaveResetEvent SlaveResume SlaveTransDone
IRQ3...IRQ4		
IRQ5	DMA	DMA DMA_ERR DMA_DONE. DMA Error signaling DMA
IRQ6	UART1	UARTINTR
IRQ7	UART2	UARTINTR
IRQ8	SSP1	SSPINTR
IRQ9		
IRQ10	I2C	INT EN_INT
IRQ11	POWER	POWER Detecor
IRQ12	WWDG	WWDG
IRQ13		
IRQ14	Timer 1	TIM_STATUS TIM_IE
IRQ15	Timer 2	
IRQ16	Timer 3	
IRQ17	ADC	EOCIF_1 AWOIF_1 EOCIF_2 AWOIF_2
IRQ18		
IRQ19	COMP	Rst_Sy1
IRQ20	SSP2	SSPINTR
IRQ21 ... IRQ26		
IRQ27	BACKUP	BKP
IRQ28	1	EXT_INT1. 0 – 1 –
IRQ29	2	EXT_INT2. 0 – 1 –
IRQ30	3	EXT_INT3. 0 – 1 –
IRQ31	4	EXT_INT4. 0 – 1 –

## 31 Контроллер прерываний NVIC

(NVIC –

Nested Vectored Interrupt Controller)

– : 0 7 ;

– , 0 ;

– ;

– ;

– ;

– ;

( )

440.

**Таблица 440 – Обобщенная информация о регистрах контроллера NVIC**

Адрес	Название	Тип	Доступ	Значение после сброса	Описание
0xE000E100	NVIC				NVIC
0x000	ISER[0]	RW		0x00000000	ISER
...					
0x01C	ISER[7]				
...					
0x080	ICER[0]	RW		0x00000000	I ER
...					
0x09C	ICER[7]				
...					
0x100	ISPR[0]	RW		0x00000000	ISPR
...					
0x11C	ISPR[7]				
...					
0x180	ICPR[0]	RW		0x00000000	ICPR
...					
0x19C	ICPR[7]				
...					
0x200	IABR[0]	RO		0x00000000	IABR
...					
0x21C	IABR[7]				
...					



### 31.1.1 NVIC->ISER[x]

ISER0 ( ).

Таблица 442 – Регистр разрешения прерываний

Номер	31...0
Доступ	R/W
Сброс	0
<b>SETENA bits</b>	

SETENA:

запись: 0 – , 1 – ;

чтение: 0 – , 1 – .

NVIC

NVIC

### 31.1.2 NVIC->ICER[x]

ICER0 ( ).

Таблица 443 – Регистр запрета прерываний

Номер	31... 0
Доступ	R/W
Сброс	0
<b>CLRENA</b>	

CLRENA:

запись: 0 – , 1 – ;

чтение: 0 – , 1 – .

### 31.1.3 NVIC->ISPR[x]

ISPR0 ( ).

Таблица 444 – Регистр установки состояния ожидания для прерывания

Номер	31...0
Доступ	R/W
Сброс	0
<b>SETPEND</b>	

SETPEND:

запись: 0 – , 1 – ;

чтение: 0 – , 1 – .

1 ISPR, : ;

### 31.1.4 NVIC->ICPR[x]

ICPR0  
( ) ,  
( ).

**Таблица 445 – Регистр сброса состояния ожидания для прерывания**

<b>Номер</b>	31...0
<b>Доступ</b>	R/W
<b>Сброс</b>	0
	<b>CLRPEND</b>

CLRPEND:  
запись: 0 – , 1 – ;  
чтение: 0 – , 1 –

1 ICPR,

### 31.1.5 NVIC->IABR[x]

ICPR0 ,  
( 446).

**Таблица 446 – Регистр активных прерываний**

<b>Номер</b>	31...0
<b>Доступ</b>	RO
<b>Сброс</b>	0
	<b>ACTIVE</b>

ACTIVE:  
чтение: 0 – ;  
1 – ,

**31.1.6 NVIC->IP[x]**

IPR0-IPR7

3-

IP[0] .. IP[29] CMSIS,

**Таблица 447 – Регистры приоритета прерываний**

IP

<b>Номер</b>	31...16	15...8	7...0
<b>Доступ</b>	U	R/W	R/W
<b>Сброс</b>	0	0	0
	-	<b>IP[29]</b>	<b>IP[28]</b>

IP

<b>Номер</b>	31...24	23...16	15...8	7...0
<b>Доступ</b>	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0
	<b>IP[4m+3]</b>	<b>IP[4m+2]</b>	<b>IP[4m+1]</b>	<b>IP[4m]</b>

IP

<b>Номер</b>	31...24	23...16	15...8	7...0
<b>Доступ</b>	R/W	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0
	<b>IP[3]</b>	<b>IP[2]</b>	<b>IP[1]</b>	<b>IP[0]</b>

0 7,

[7:5] , [4:0]

IPR

N

- M

M = N DIV 4;

N MOD 4

- 0 – [7:0];
- 1 – [15:8];
- 2 – [23:16];
- 3 – [31:24].



### 31.1.7 NVIC->STIR

STIR  
 (SGI – Software Generated Interrupt).  
 USERSETMPEND SCR 1,  
 STIR ( . “ ”).

**Таблица 448 – Регистр программного формирования прерывания**

<b>Номер</b>	31...9	8...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	<b>INTID</b>

INTID – 0 – 239.  
*Например:* b000000011 IRQ3.

## 31.2 Прерывания, срабатывающие по уровню сигнала

## 31.3 Аппаратное и программное управление прерываниями

Cortex-M3

— NVIC  
— NVIC  
; ISPR0 ( . .NVIC->ISPR[x]) STIR ( .  
. NVIC->STIR).

NVIC

### 31.4 Рекомендации по работе с контроллером прерываний

SCB->VTOR.

CPSIE | CPSID | CMSIS

```
void __disable_irq(void) // Disable Interrupts
void __enable_irq(void) // Enable Interrupts
```

, CMSIS  
NVIC:

**Таблица 449 – Функции CMSIS для управления контроллером прерываний**

Функция	Описание
void NVIC_SetPriorityGrouping (uint32_t priority_grouping)	
void NVIC_EnableIRQ (IRQn_t IRQn)	IRQn
void NVIC_DisableIRQ (IRQn_t IRQn)	IRQn
uint32_t NVIC_GetPendingIRQ (IRQn_t IRQn)	TRUE, IRQn, FALSE
void NVIC_SetPendingIRQ (IRQn_t IRQn)	IRQn
void NVIC_ClearPendingIRQ (IRQn_t IRQn)	IRQn
uint32_t NVIC_GetActive (IRQn_t IRQn)	IRQ
void NVIC_SetPriority (IRQn_t IRQn, uint32_t priority)	IRQn
uint32_t NVIC_GetPriority (IRQn_t IRQn)	IRQn
void NVIC_SystemReset (void)	

CMSIS.

## 32 Блок управления системой

### SCB

450.

**Таблица 450 – Обобщенная информация о регистрах блока управления системой**

Адрес	Имя	Тип	Доступ	Значение после сброса	Описание
0xE000E000	InterruptType				
0x008	ACTLR	RW	-	0x00000000	
0xE00ED00	SCB				
0x000	CPUID	RO	-	0x412FC230	
0x004	ICSR	RW	-	0x00000000	
0x008	VTOR	RW	-	0x00000000	
0x00C	AIRCR	RW	-	0xFA050000	
0x010	SCR	RW	-	0x00000000	
0x014	CCR	RW	-	0x00000200	
0x018	SHPR1	RW	-	0x00000000	1
0x01C	SHPR2	RW	-	0x00000000	2
0x020	SHPR3	RW	-	0x00000000	3
0x024	SHCRS	RW	-	0x00000000	
0x028	CFSR	RW	-	0x00000000	
0x028	MMSR	RW	-	0x00	
0x029	BFSR	RW	-	0x00	
0x02A	UFSR	RW	-	0x0000	
0x02C	HFSR	RW	-	0x00000000	
0x034	MMAR	RW	-		
0x038	BFAR	RW	-		

### 32.1 Упрощенный доступ к регистрам блока управления системой

SCB CMSIS SHPR3 SHPR1- SHP[0]...SHP[12].

#### 32.1.1 InterruptType->ACTLR

ACTLR (IT folding); (default memory map);

ACTLR

**Таблица 451 – Дополнительный регистр управления**

Номер	31...3	2	1	0
Доступ	U	R/W	R/W	R/W
Сброс	0	0	0	0
	-	<b>DISFOLD</b>	<b>DISDEFWBUF</b>	<b>DISMCYCINT</b>

DISFOLD – 1 (IT folding)

DISDEFWBUF – 1 (default memory map).

Cortex-M3.

DISMCYCINT – 1 (LDM STM). LDM STM

*О вложении условных инструкций*

IT. IT- (« »). DISFOLD 1.

### 32.1.2 SCB->CPUID

CPUID

452.

**Таблица 452 – Регистр идентификации процессора**

<b>Номер</b>	31...24	23...20	19...16	15...4	3...0
<b>Доступ</b>	RO	RO	RO	RO	RO
<b>Сброс</b>	0x41	0x2	0xF	0xC23	0x0
	<b>Implementer</b>	<b>Variant</b>	<b>Constant</b>	<b>PartNo</b>	<b>Revision</b>

Implementer – 0x41 = ARM.  
 Variant – r rnpn : 0x2 = r2p0;  
 Constant – 0xF;  
 PartNo – : 0xC23 = Cortex-M3;  
 Revision – p rnpn : 0x0 = r2p0.

### 32.1.3 SCB->ICSR

ICSR

PendSV SysTick, :

– ;  
 – ;  
 – ;  
 – ;

**Таблица 453 – Регистр управления прерываниями**

<b>Номер</b>	31...29	28	27	26	25	24	23	22	21...12	11	10	9	8...0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	U	R/W	R/W	R/W	R/W	U	U	R/W
<b>Сброс</b>	0	0	0	0	0	0	0	0	0	0	0	0	0

	PENDSVSET	PENDSVCLR	PENDSTSET	PENDSTCLR		Reserved for Debug	ISR_PENDING	VECT_PENDING	RETTOBASE		VECTACTIVE
--	-----------	-----------	-----------	-----------	--	--------------------	-------------	--------------	-----------	--	------------

PENDSVSET (RW) –

PendSV.

Запись 0 – , 1 – PendSV

Чтение 0 – PendSV , 1 –

Запись 1 – это единственно возможный способ перевода исключения PendSV в состояние ожидания обслуживания.

PENDSVCLR (WO) –

PendSV.

Запись 0 –

Запись 1 –

PendSV.

PENDSTSET (RW) –

SysTick.

Запись 0 –

Запись 1 –

Чтение 0 –

SysTick SysTick

. 1 –

PENDSTCLR (WO) –

SysTick.

Запись 0 –

Запись 1 –

SysTick.

Reserved for Debug use (RO) –

0.

ISR\_PENDING (RO) –

. 0 –

( , 1 – )

VECT\_PENDING (RO) –

,

. 0 –

PRIMASK.

BASEPRI FAULTMASK,

RETTOBASE (RO) –

. 0 –

, 1 –

VECT\_ACTIVE (RO) –

. 0 –

(IRQ)

VECT\_ACTIVE

16.

ICSR

-

1

PENDSVSET

PENDSVCLR;

-

1

PENDSTSET

PENDSTCLR.

### 32.1.4 SCB->VTOR

VTOR

0x00000000.

**Таблица 454 – Регистр смещения таблицы векторов прерываний**

<b>Номер</b>	31	30	29	7	6...0
<b>Доступ</b>	U	U	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	-		<b>TBLOFF</b>		<b>Reserved</b>

TBLOFF –

[28:7].

[29] SRAM: 0 = TBLBASE, 1 = SRAM. [29]

TBLOFF

16

32

21

64

37

37

64

[6...0]

### 32.1.5 SCB->AIRC

#### 32.1.5.1 Регистр управления прерываниями и программного сброса AIRCR

(endian)

VECTKEY

0x05FA,

**Таблица 455 – Регистр управления прерываниями и программного сброса**

Номер	31...16	15	14...11	10...8	7...3	2	1	0
Доступ	R/W	R/W	U	R/W	U	R/W	R/W	R/W
Сброс	0	0	0	0	0	0	0	0
	On Read: VECTKEYSTAT, On Write: VECTKEY	ENDIANESS	-	PRIGROUP	-	SYSRESEREQ	VECTCLRACTIVE	VECTRESET

VECTKEYSTAT –

0xFA05.

VECTKEY –

0x05FA,

ENDIANESS (RO) –

(little-endian), 1 –

0 – (big-endian).  
BIGEND

PRIGROUP (RW) –



PRIGROUP –

PRI\_n

PRIGROUP

**Таблица 456 – Группировка приоритетов прерываний**

PRIGROUP	Значение приоритета в поле PRI_N[7:0]			Общее количество	
	Положение двоичной точки	Биты номера группы	Биты номера подгруппы	Групп	подгрупп
0b100	bxxx.00000	[7:5]	None	8	1
0b101	bxx.y00000	[7:6]	5	4	2
0b110	bх.уу00000	[7]	[6:5]	2	4
0b111	b.yyy00000	None	[7:5]	1	8

SYSRESETREQ (WO) –

. 0 – , 1 –  
0.

VECTCLRACTIVE (WO) –

. 0.  
0,

VECTRESET (WO) –

. 0.  
0,

### 32.1.6 SCB->SCR

SCR

**Таблица 457 – Регистр управления системой**

Номер	31...5	4	3	2	1	0
Доступ	U	R/W	U	R/W	R/W	U
Сброс	0	0	0	0	0	0
	-	<b>SEVONPEND</b>	-	<b>SLEEPDEEP</b>	<b>SLEEONEXIT</b>	-

SEVONPEND –

. 0 –

; 1 –

WFE,

SEV.

SLEEPDEEP – :  
 0 – (Sleep);  
 1 – (Deep Sleep).  
 SLEEPONEXIT –

: 0 – , 1 – .

### 32.1.7 SCB->CCR

CCR

FAULTMASK;

STIR

( . NVIC->STIR).

**Таблица 458 – Регистр конфигурации и управления**

Номер	31...10	9	8	7...5	4	3	2	1	0
Доступ	U	R/W	R/W	U	R/W	R/W	U	R/W	R/W
Сброс	0	0	0	0	0	0	0	0	0
		STKALIGN	BFHFNMIGN		DIV_0_TRP	UNALIGN_TRP		USERSETMPEND	NONBASETHRDENA

STKALIGN : 0 -  
 4 ; 1 - 8 [9]

PSR

BFHFNMIGN -1 -2  
 FAULTMASK. 0 =

; 1 = -1 -2

DIV\_0\_TRP

SDIV UDIV. 0 = 0. 1 =  
 0,

0.

UNALIGN\_TRP

. 0 =

1 =

1,

(usage fault).

LDM, STM, LDRD STRD  
UNALIGN\_TRP.

USERSETMPEND

. 0 =

STIR ( . NVIC->STIR)  
, 1 =

NONEBASETHRDENA

(Thread mode): 0 =

, 1 =

EXC\_RETURN

( . “ , ”).

### 32.1.8 SCB->SHP[x]

SHPR1-SHPR3

PRI\_N

8

[7...4],

[3...0]

**Таблица 459 – Поля приоритета обработчиков системных отказов**

Обработчик отказа	Поле	Описание регистра
	SHP[4]	1
	SHP[5]	
(usage fault)	SHP[6]	
SVCall	SHP[11]	2
PendSV	SHP[14]	3
SysTick	SHP[15]	

1

**Таблица 460 – Регистр №1 приоритета системных обработчиков**

Номер	31...24	23...16	15...8	7...0
Доступ	R/W	R/W	R/W	R/W
Сброс	0	0	0	0
	<b>PRI_7: Резерв</b>	<b>PRI_6</b>	<b>PRI_5</b>	<b>PRI_4</b>

PRI\_7

PRI\_6

6,

PRI\_5

5,

PRI\_4

4,



**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

MEMFAULTENA			, 1 –		, 0 –
SVCALLPENDE		SVC,	1,		
BUSFAULTPENDE					1,
MEMFAULTPENDE					1,
USGFAULTPENDE					
			1,		
SYSTICKACT		SysTick,			1,
PENDSVACT		PendSV,			1,
MONITORACT				1,	
SVCALLACT		SVC,	1,		
USGFAULTACT					
			1,		
BUSFAULTACT					1,
MEMFAULTACT					1,

Примечания:

- 0 – ; 1

- 0 – 1 ;

- 0 – ;

- 1 ;

1

( ) ,

SHCSR.

### 32.1.10 SCB->CFSR

CFSR

(usage fault).

**Таблица 464 – Регистр состояния отказов с конфигурируемым уровнем приоритета**

<b>Номер</b>	31...16	15...8	7...0
<b>Доступ</b>	RO	RO	RO
<b>Сброс</b>	0	0	0
	<b>Usage Fault Status Register: UFSR</b>	<b>Bus Fault Status Register: BFSR</b>	<b>Memory Management Fault Status Register: MMFSR</b>

CFSR

CFSR

- 0xE000ED28 — CFSR;
- 0xE000ED28 — MMFSR;
- 0xE000ED28 — MMFSR BFSR;
- 0xE000ED29 — BFSR;
- 0xE000ED2A — UFSR.

CFSR:

#### 32.1.10.1 Поле MMFSR

MMFSR

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

**Таблица 465 – Регистр состояния отказов доступа к памяти**

Номер	7	6	5	4	3	2	1	0
Доступ	RO	U	U	RO	RO	U	RO	RO
Сброс	0	0	0	0	0	0	0	0
	<b>MMARVALID</b>	<b>-</b>	<b>MSTKERR</b>	<b>MUNSTKERR</b>	<b>-</b>	<b>DACCVIOL</b>	<b>IACCVIOL</b>	

MMARVALID

(MMAR): 0 =

MMAR

, 1 =

0.

MMAR

MSTKERR

: 0 =

, 1 =

1,

SP

MMAR

MUNSTKERR

: 0 =

, 1 =

1,

MMAR

DACCVIOL

: 0 =

, 1 =

1,

PC,

MMAR

IACCVIOL

: 0 =

, 1 =

(disabled)

(XN),

1,

MPU

PC,

MMAR

**32.1.10.2 Поле BFSR**

BFSR

:

**Таблица 466 – Регистр состояния отказов доступа к шине**

<b>Номер</b>	7	6	5	4	3	2	1	0
<b>Доступ</b>	RO	U	U	RO	RO	RO	RO	RO
<b>Сброс</b>	0	0	0	0	0	0	0	0
	<b>BFRVALID</b>			<b>STKERR</b>	<b>UNSTKERR</b>	<b>IMPRECISERR</b>	<b>PRECISERR</b>	<b>IBUSERR</b>

BFRVALID

0 =

BFR

, 1 =

(BFR):

1

0.

0.

MMAR

STKERR

: 0 =

, 1 =

1,

SP

BFR

UNSTKERR

: 0 =

, 1 =

1,

BFR

IMPRECISERR

. 0 =

, 1 =

1,

BFR

PRECISERR

. 0 =

, 1 =

1,

BFR.



IBUSERR . 0 = , 1 =  
 , IBUSERR  
 1, BFAR

**32.1.10.3 Поле UFSR**

UFSR

**Таблица 467 – Регистр состояния отказов, вызванных ошибками программирования**

Номер	15...10	9	8	7...4	3	2	1	0
Доступ	U	RO	RO	U	RO	RO	RO	RO
Сброс	0	0	0	0	0	0	0	0
	-	<b>DIVBYZERO</b>	<b>UNALIGNED</b>	-	<b>NOCP</b>	<b>INVPC</b>	<b>INVSTATE</b>	<b>UNDEFINSTR</b>

DIVBYZERO : 0 =  
 , 1 = SDIV, UDIV  
 0. 1, PC,  
 1 DIV\_0\_TRP CCR ( . . “SCB->CCR”).

UNALIGNED : 0 =  
 , 1 =  
 1 UNALIGN\_TRP CCR ( . . “SCB->CCR”).  
 LDM, STM, LDRD, STRD,  
 UNALIGN\_TRP.

NOCP . 0 = , 1 =

INVPC PC. 0 = , 1 =  
 PC EXC\_RETURN,  
 EXC\_RETURN.  
 1, PC,  
 PC.

INVSTATE : 0 = , 1 =  
 EPSR. 1,  
 PC,  
 EPSR.

UNDEFINSTR . 0 = , 1 =  
 1,  
 PC,  
 1 UFSR  
 1,

**32.1.11 SCB->HFSR**

HFSR

1.

0

**Таблица 468 – Регистр состояния тяжелого отказа**

<b>Номер</b>	31	30	29...2	1	0
<b>Доступ</b>	R/W	R/W	U	R/W	R/W
<b>Сброс</b>	0	0	0	0	0
	<b>DEBUGEVT</b>	<b>FORCED</b>	-	<b>VECTTBL</b>	<b>Reserved</b>

DEBUGEVT

0,

FORCED

1,

HFSR.

VECTTBL

: 0 =

, 1 =

1,

PC,

1

HFSR

1,

### 32.1.12 SCB->MMFAR

MMFAR

**Таблица 469 – Регистр адреса отказа доступа к памяти**

<b>Номер</b>	31...0
<b>Доступ</b>	RO
<b>Сброс</b>	0
	<b>ADDRESS</b>

ADDRESS

MMARVALID

MMFSR

1,

MMFSR  
MMFAR

### 32.1.13 SCB->BFAR

BFAR

**Таблица 470 – Регистр адреса отказа доступа к шине**

<b>Номер</b>	31...0
<b>Доступ</b>	RO
<b>Сброс</b>	0
	<b>ADDRESS</b>

ADDRESS

BFARVALID

BFSR

1,

BFSR  
BFAR

### 32.1.14 Рекомендации по программированию блока управления системой

,  
CFSR SHPR1-SHPR3,  
, CFSR SHPR1-SHPR3 ,  
:  
— MMFAR BFAR;  
— MMARVALID MMFSR, BFARVALID  
BFSR. MMFAR BFAR ,  
1.  
, MMFAR  
BFAR, ,

## 33 Сторожевые таймеры

### 33.1 Описание регистров блока сторожевых таймеров

Таблица 471 – Описание регистров блока сторожевых таймеров

Базовый Адрес	Название	Описание
0x4006_8000	MDR_IWDG	IWDG
<b>Смещение</b>		
0x00	MDR_IWDG->KR[15:0]	
0 04	MDR_IWDG->PR[2:0]	
0x08	MDR_IWDG->RLR[11:0]	
0x0C	MDR_IWDG->SR[1:0]	

Таблица 472 – Оконный сторожевой таймер

Базовый Адрес	Название	Описание
0x4006_0000	MDR_WWDG	WWDG
<b>Смещение</b>		
0x00	MDR_WWDG->CR[7:0]	
0 04	MDR_WWDG->CFR[9:0]	
0x08	MDR_WWDG->SR[0]	

### 33.1.1 MDR\_IWDG->KR

**Таблица 473 – Регистр KR**

<b>Номер</b>	15...0
<b>Доступ</b>	WO
<b>Сброс</b>	0
	<b>KEY[15:0]</b>

**Таблица 474 – Описание бит регистра KR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...16		
15...0	KEY[15:0]	Значение ключа (только запись, читается 0000h).  AAAAh, , . 5555h PR RLR. CCCCh . IWDT .

### 33.1.2 MDR\_IWDG->PR

**Таблица 475 – Регистр PR**

<b>Номер</b>	31...3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W
<b>Сброс</b>	0	0	0	0
	-	<b>PR2</b>	<b>PR1</b>	<b>PR0</b>

**Таблица 476 – Описание регистра PR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...3		
2...0	PR[2:0]	Делитель частоты сторожевого таймера: 000 – 4 001 – 8 010 – 16 011 – 32 100 – 64 101 – 128 110 – 256 111 – 256  PVU = 0 SR IWDT LSI

### 33.1.3 MDR\_IWDG->RLR

**Таблица 477 – Регистр RLR**

<b>Номер</b>	31...12	11...0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	1
	-	<b>RLR[11:0]</b>

**Таблица 478 – Описание регистра RLR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...12		
11...0	RLR[11:0]	<p><b>Значение перезагрузки сторожевого таймера.</b></p> <p>KR.</p> <p align="center">AAAAh                      KR.</p> <p align="center">RVU=0                      SR</p>

### 33.1.4 MDR\_IWDG->SR

**Таблица 479 – Регистр SR**

<b>Номер</b>	31...2	1	0
<b>Доступ</b>	U	RO	RO
<b>Сброс</b>	0	0	0
	-	<b>RVU</b>	<b>PVU</b>

**Таблица 480 – Описание регистра SR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...2		
1	RVU	<p><b>Флаг обновления значения сторожевого таймера.</b></p>
0	PVU	<p><b>Флаг обновления делителя частоты сторожевого таймера.</b></p>

**33.1.5 MDR\_WWDG->CR**

**Таблица 481 – Регистр CR**

<b>Номер</b>	31...8	7	6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>		0	1	1	1	1	1	1	1
	-	WDGA	T6	T5	T4	T3	T2	T1	T0

**Таблица 482 – Описание бит регистра CR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...8		
7	WDGA	<p><b>Бит активации</b></p> <p>WDGA=1,</p> <p>:</p> <p>0 – ;</p> <p>1 –</p>
6...0	T[6:0]	<p><b>Значение семиразрядного счётчика (от старших разрядов к младшим)</b></p> <p>4096x2<sup>WDGTB</sup> , PCLK</p> <p>APB</p>



### 33.1.6 MDR\_WWDG->CFR

**Таблица 483 – Регистр CFR**

<b>Номер</b>	31...10	9	8	7	6	5	4	3	2	1	0
<b>Доступ</b>	U	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
<b>Сброс</b>		0	0	0	1	1	1	1	1	1	1
	-	EWI	WDGTB1	WDGTB0	W6	W5	W4	W3	W2	W1	W0

**Таблица 484 – Описание бит регистра CFR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...10		
9	EWI	Раннее предупреждающее прерывание.  40h.
8...7	WGTB[1:0]	Делитель частоты сторожевого таймера. 00 – (PCLK / 4096) /1 01 – (PCLK / 4096) /2 10 – (PCLK / 4096) /4 11 – (PCLK / 4096) /8
6...0	W[6:0]	Значение окна.  T[6:0] , 40h-7Fh. T>W, RESET. T=3Fh,

### 33.1.7 MDR\_WWDG->SR

**Таблица 485 – Регистр SR**

<b>Номер</b>	31...1	0
<b>Доступ</b>	U	R/W
<b>Сброс</b>	0	0
	-	EWIF

**Таблица 486 – Описание бит регистра SR**

<b>№ бита</b>	<b>Функциональное имя бита</b>	<b>Расшифровка функционального имени бита, краткое описание назначения и принимаемых значений</b>
31...1		
0	EWIF	Флаг раннего предупреждающего прерывания.  40h.  EWI=0

### 34 Предельно допустимые характеристики микросхемы

Таблица 487 – Предельно допустимые и предельные режимы эксплуатации микросхем

Наименование параметра, единица измерения	Буквенное обозначение параметра	Норма параметра			
		Предельно допустимый режим		Предельный режим	
		не менее	не более	не менее	не более
, USB,	U <sub>CC</sub>	2,2	3,6	–	4,0
		3,0	3,6	–	4,0
, *	U <sub>CCA</sub>	2,4	3,6	–	4,0
,	U <sub>CC</sub>	1,8	3,6	–	4,0
( : PA, PB, PC, PD, PE, PF, RESET, WAKEUP, SHDN, JTAG_EN : DN, DP : OSC_IN HSE BYPASS = 1	U <sub>IL</sub>	0	0,8	–0,3	–
		0	0,8	–0,3	–
		0	0,8	–0,3	–
( : PD, PE (0-10) : PA, PB, PC, PE (11-15), PF, RESET, WAKEUP, SHDN, JTAG_EN : DN, DP : OSC_IN HSE BYPASS = 1	U <sub>IH</sub>	2,0	3,6	–	4,0
		2,0	5,25	–	5,3
		2,0	3,6	–	4,0
		2,0	3,6	–	4,0
( : PA, PB, PC, PD, PE (0-5, 8-15), PF PE (6, 7), STANDBY : DN, DP	I <sub>OH</sub>	–6	–	–10	–
		–3	–	–10	–
		–6	–	–40	–
( : PA, PB, PC, PD, PE, PF, STANDBY : DN, DP	I <sub>OL</sub>	–	6	–	10
		–	6	–	40

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Наименование параметра, единица измерения	Буквенное обозначение параметра	Норма параметра			
		Предельно допустимый режим		Предельный режим	
		не менее	не более	не менее	не более
	f <sub>C</sub>	–	80	–	–
HSE, : BYPASS = 0 : BYPASS = 1	f <sub>C_HSE</sub>	2	16	–	–
		0	80	–	–
LSE, : BYPASS = 0 : BYPASS = 1	f <sub>C_LSE</sub>	32	33	–	–
		0	1 000	–	–
PLL,	f <sub>_PLL</sub>	6	16	–	–
<b>Параметры ЦАП</b>					
: REFD0, REFD1 : Cfg_M_REF = 1	U <sub>REF(DAC)</sub>	2,4	U <sub>CCA</sub>	–	–
	R <sub>LOAD</sub>	10	–	–	–
	C <sub>LOAD</sub>	–	100	–	–
<b>Параметры АЦП</b>					
: ADC1_Cfg_M_REF = 1 ADC2_Cfg_M_REF = 1	U <sub>ADC1_REF-</sub>	0	U <sub>CCA-2,4</sub>	–0,3	–
: ADC1_Cfg_M_REF = 1 ADC2_Cfg_M_REF = 1	U <sub>ADC0_REF+</sub>	2,4	U <sub>CCA</sub>	–	4,0
U <sub>REF(ADC)</sub> = U <sub>ADC0_REF+</sub> – U <sub>ADC1_REF-</sub>	U <sub>REF(ADC)</sub>	2,4	U <sub>CCA</sub>	–	–
, **	U <sub>AIN</sub>	U <sub>ADC1_REF-</sub>	U <sub>ADC0_REF+</sub>	–0,3	4,0
	f <sub>_ADC</sub>	–	14	–	–

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Наименование параметра, единица измерения	Буквенное обозначение параметра	Норма параметра			
		Предельно допустимый режим		Предельный режим	
		не менее	не более	не менее	не более
: PA, PB, PC, PD, PE, PF, Standby	C <sub>L</sub>	–	30	–	–
: = 125 °	N <sub>PR</sub>	10 000	–	–	–
: = 25 °	t <sub>GS</sub>	25	–	–	–
: = 85 °		10	–	–	–
: = 125 °		1	–	–	–
<p>* <math>U_{CC} \pm 0,2 \%</math></p> <p>** <math>U_{ADC1\_REF-} = GND_A</math> <math>U_{ADC0\_REF+} = U_{A}</math>, <math>U_{REF(DAC)} = U_{A}</math>.</p>					

## 35 Электрические параметры микросхемы

Таблица 488 – Электрические параметры микросхемы

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра		Температура среды, °С
		не менее	не более	
: PA, PB, PC, PD, PE, PF, STANDBY, DN, DP	$U_{OL}$	–	0,4	25, 125, – 60
: PA, PB, PC, PD, PE, PF, STANDBY, DN, DP : $U_{CC} = 3,0$	$U_{OH}$	2,4	–	
: PA, PB, PC, PD, PE, PF, STANDBY : $U_{CC} = 2,2$		1,6	–	
	$U_{BOR}$	1,8	2,1	
: PA, PB, PC, PD, PE, PF, RESET, WAKEUP, DN, DP	$I_{ILH}$	– 1,0	1,0	
: OSC_IN : BYPASS = 1		– 40,0	40,0	
: PA, PB, PC, PD, PE, PF, RESET, WAKEUP, DN, DP	$I_{ILL}$	– 1,0	1,0	
: OSC_IN : BYPASS = 1		– 1,0	1,0	
( ) ,	$I_{CCS}$	–	10	
		–	20	
	$I_{OCC1}$	–	120	
LSI RC-	$f_{O\_LSI}$	10	60	
HSI RC-	$f_{O\_HSI}$	6	10	
PLL,	$f_{O\_PLL}$	100	–	
		–	6	
<b>Параметры АЦП</b>				
	$E_{NADC}$	12	–	25, 125, – 60
	$E_{DLADC}$	– 1	2	
	$E_{ILADC}$	– 3	3	
	$E_{OFFADC}$	– 6	6	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра		Температура среды, °С
		не менее	не более	
, %	EGAINADC	- 1	1	25, 125, - 60
<b>Параметры ЦАП</b>				
	ENDAC	12	-	25, 125, - 60
,	EDLDAC	- 1	2	
,	EILDAC	- 6	6	
,	EOFFDAC	- 40	40	
, %	EGAINDAC	- 2	2	
,	UO_DAC min	-	0,08	
,	UO_DAC max	UREF(DAC)-0,08	-	
<b>Компаратор</b>				
,	tON_C*	-	100	25, 125, - 60
,	td_*	-	400	
<p>* tON_C, td_</p> <p>.431290.711</p>				

( 488).

### 36 Справочные данные

134 – 141.

:

- 1986 91 , 1986 94 - 3 700 ;
- 1986 92 - 4 300 ;
- 1986 92 1 - 5 800 ;
- 1986 93 - 4 500 ;
- 1986 94 - 8 300 ;
- 1986 94 - 8 600 .

-

:

- 1986 91 , 1986 94 - 39° / ;
- 1986 92 - 36° / ;
- 1986 92 1 - 6° / ;
- 1986 93 - 29° / ;
- 1986 94 - 10° / ;
- 1986 94 - 12° / .

-

1

:

- 1986 91 , 1986 94 - 12,5 ;
- 1986 92 - 11 ;
- 1986 93 - 10 .

489.

**Таблица 489 – Предельно допустимые значения ОИН**

Тип вывода	Длительность ОИН, мкс	
	Предельно допустимое напряжение ОИН, В	
	1,0	10,0
	1 000	300
	1 750	500
	1 750	1 000

490.

**Таблица 490 – Справочные данные**

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра		Температура среды, °С
		не менее	не более	
$U_{CC} = 3,6$	$I_{CCS1}$	–	1,5	25, 125, – 60
$U_{CC} = 0$	$I_{CC\_B}$	–	5	
SLEEPDEEP, $U_{CC} = 3,6$ , $f_C = f_{O\_LSI}$	$I_{OCC2}$	–	2	
PBD PBVD,	$t_{SU(PBD)}$ $t_{SU(PBVO)}$	–	2	
$U = 3,6$ , $I_{OL} = 80$ LDO,	$U_{O\_LDO}$	1,62	1,98	
: PA-PF : - ModeRX = 0 - ModeRX = 1	$U_{TH(PA-PF)}$	100 200	400 500	
: STANDBY, $C_1 = 30$	$t_{W(STANDBY)}$	–	10	
: PA – PF $U_{CC} = 2,2$ , - PowerTX = 00, $C_1 = 50$ - PowerTX = 01, $C_1 = 50$ - PowerTX = 10, $C_1 = 50$ - PowerTX = 11, $C_1 = 50$ - PowerTX = 11, $C_1 = 30$	$t_{W(PA-PF)}$	– – – – –	10 100 20 10 5	
: DN, DP $U_{CC} = 3,0$ , - Full Speed, $C_1 = 50$ - Low Speed, $C_1 = 600$	$t_{W(DN, DP)}$	– –	15 300	
	$R_{PULL}$	20	100	
<b>Компаратор</b>				
$U_{CC} = 3,6$	$U_{IO\_C}$	–	$ \pm 0,5 $	25, 125, – 60
$U = 3,6$	$U_{TH\_}$	8	12	
: $U = 3,6$	$U_{REF\_C}$	1,17	1,23	



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра		Температура среды, °С
		не менее	не более	
<b>Тактовые частоты и генераторы</b>				
HSIRDY HSION, , U = 2,2	t <sub>SU(HSD)</sub>	–	1	25, 125, – 60
LSIRDY LSION, с, U = 2,2 В	t <sub>SU(LSI)</sub>	–	80	
HSERDY HSEON, с, U = 2,2 В	t <sub>SU(HSE)</sub>	–	2048/f <sub>C_HSE</sub>	
LSERDY LSEON, с, U = 2,2 В	t <sub>SU(LSE)</sub>	–	4096/f <sub>C_LSE</sub>	
PLLRDY PLLON, , U = 2,2 В	t <sub>SU(PLL)</sub>	–	100	
U = 2,2 В	t <sub>w( )</sub>	20	–	
POR,	t <sub>POR</sub>	–	6	
nRESET « »,	t <sub>w(nRESET)min</sub>	200	–	
nRESET, « »,	t <sub>w(nRESET)max</sub>	–	10	
<b>АЦП</b>				
U <sub>CC</sub> = 3,6 В	t <sub>A_ADC</sub>	–	4/f <sub>C_ADC</sub>	25, 125, – 60
U = 3,6 В	t <sub>AO_ADC</sub>	–	28/f <sub>C_ADC</sub>	
- ADC1_Cfg_M_REF = 1 - ADC2_Cfg_M_REF = 1	I <sub>ADC0_VREF+</sub>	–	50	
- ADC1_Cfg_M_REF = 1 - ADC2_Cfg_M_REF = 1	I <sub>ADC0_VREF-</sub>	– 50	–	
: U <sub>CCA</sub> = 3,6 , f <sub>C_ADC</sub> = 14	I <sub>OCCADC</sub>	–	3	
,	f <sub>C_ADCMIN</sub>	10	–	

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

Наименование параметра, единица измерения, режим измерения	Буквенное обозначение параметра	Норма параметра		Температура среды, °С
		не менее	не более	
<b>ЦАП</b>				
$U_{CC} = 3,6 \text{ В}, C_1 = 50 \text{ нФ}, R_1 = 10 \text{ кОм}$	$t_{SU(DAC)}$	–	5,2	25, 125, – 60
$U_{REF} = 2,4 \text{ В}$	$t_{ON\_DA}$	–	10	
$Cfg\_M\_REF0 = 1$	$I_{DAC1\_VREF}$	–	500	
$Cfg\_M\_REF1 = 1$	$I_{DAC2\_VREF}$	–	500	
	$I_{OCCDAC}$	–	2	
<b>USB</b>				
DN, DP - D-PULLUP = 1 - D+PULLUP = 1	$R_{DN-UCC}$	1	2	25, 125, – 60
	$R_{DP-UCC}$	1	2	
DN, DP « », - D-PULLDOWN = 1 - D+PULLDOWN = 1	$R_{DN-GND}$	10	20	
	$R_{DP-GND}$	10	20	
DN, DP,	$R_{DN}$ $R_{DP}$	14	34	

### 37 Типовые зависимости

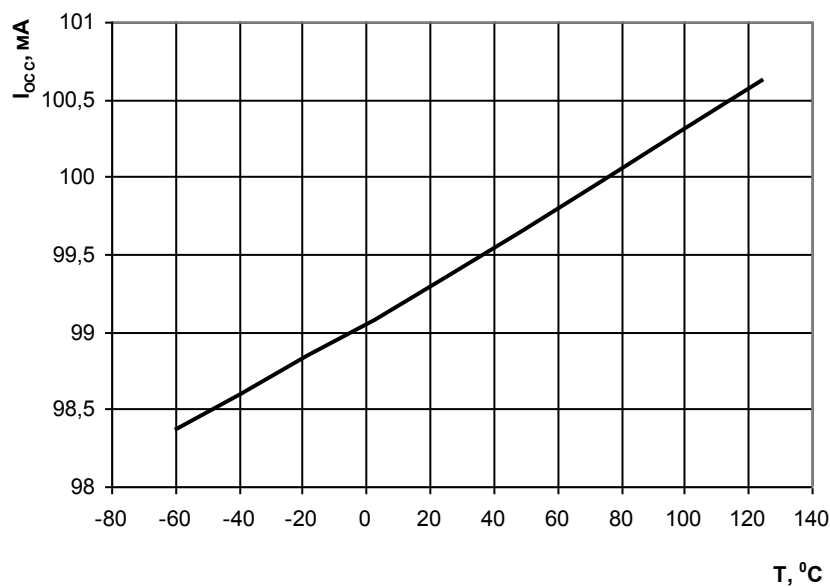


Рисунок 134. Зависимость динамического тока потребления от температуры при:  $f_c = 80$  МГц

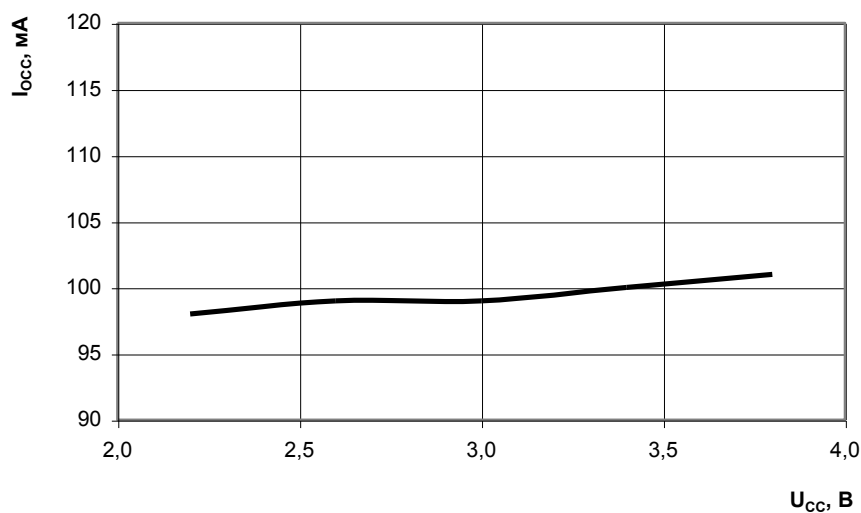


Рисунок 135. Зависимость динамического тока потребления от напряжения питания

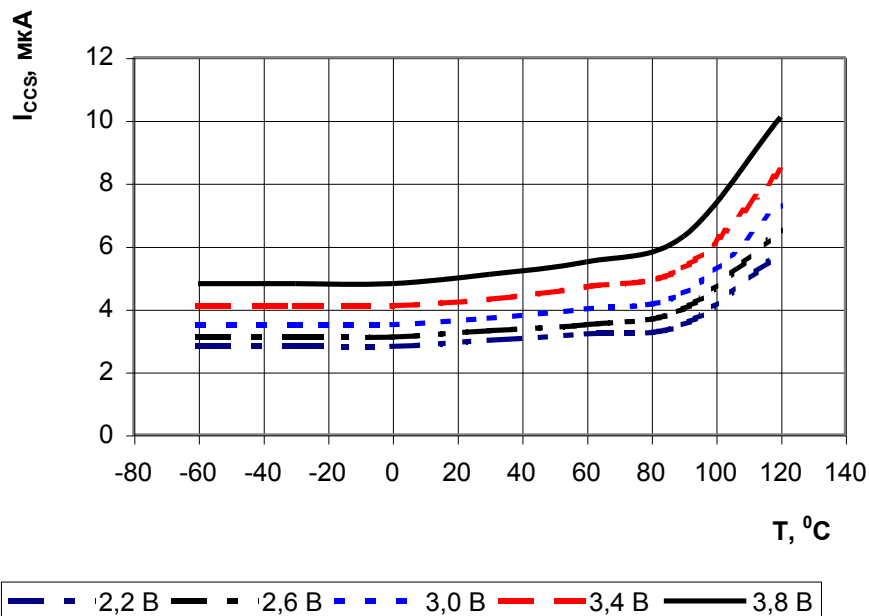


Рисунок 136. Зависимость статического тока потребления в режиме покоя (регулятор напряжения выключен) от температуры при разном напряжении

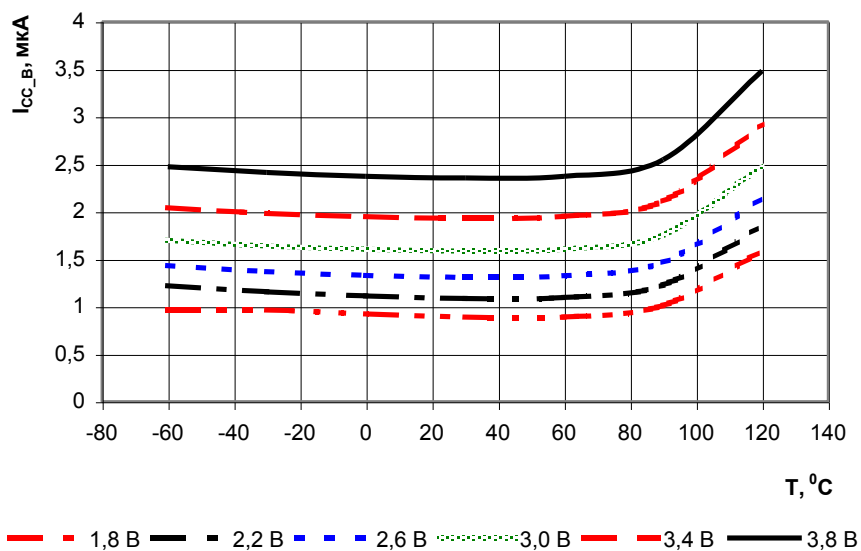


Рисунок 137. Зависимость тока потребления батарейного домена от температуры при разном напряжении

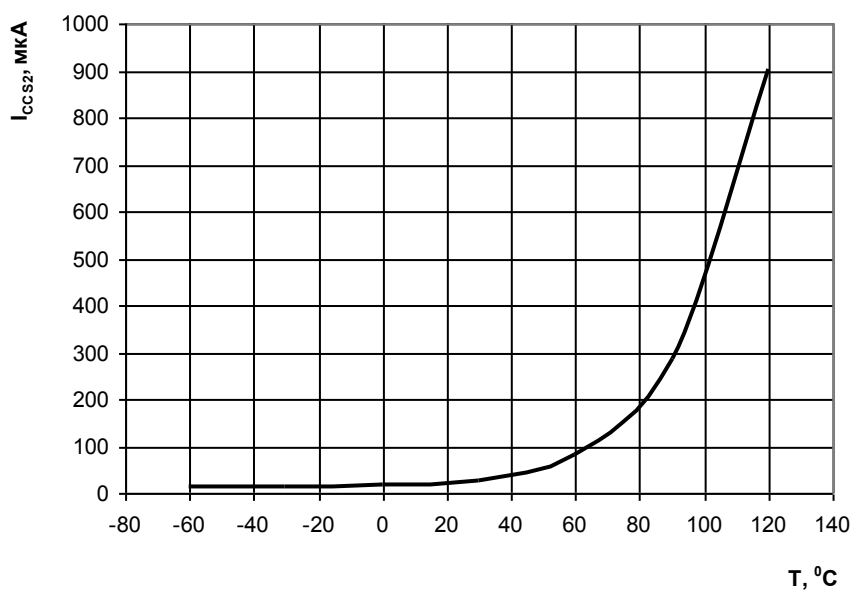


Рисунок 138. Зависимость статического тока потребления в режиме покоя (регулятор напряжения включен) от температуры

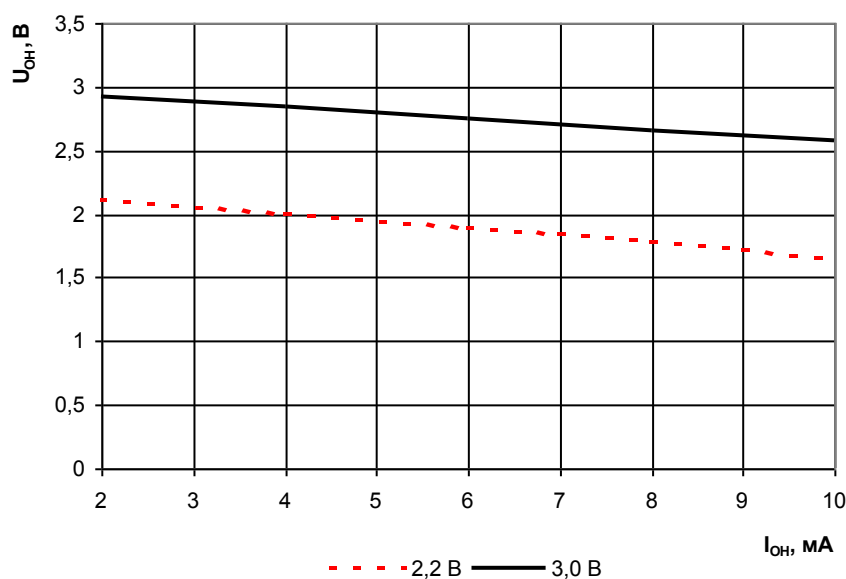


Рисунок 139. Зависимость выходного напряжения высокого уровня от выходного тока высокого уровня при напряжении питания 2,2 и 3,0 В

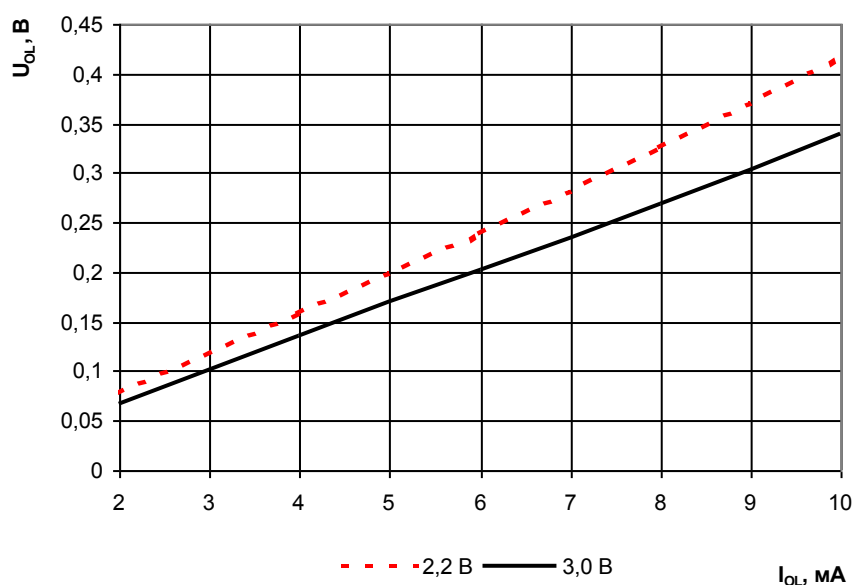
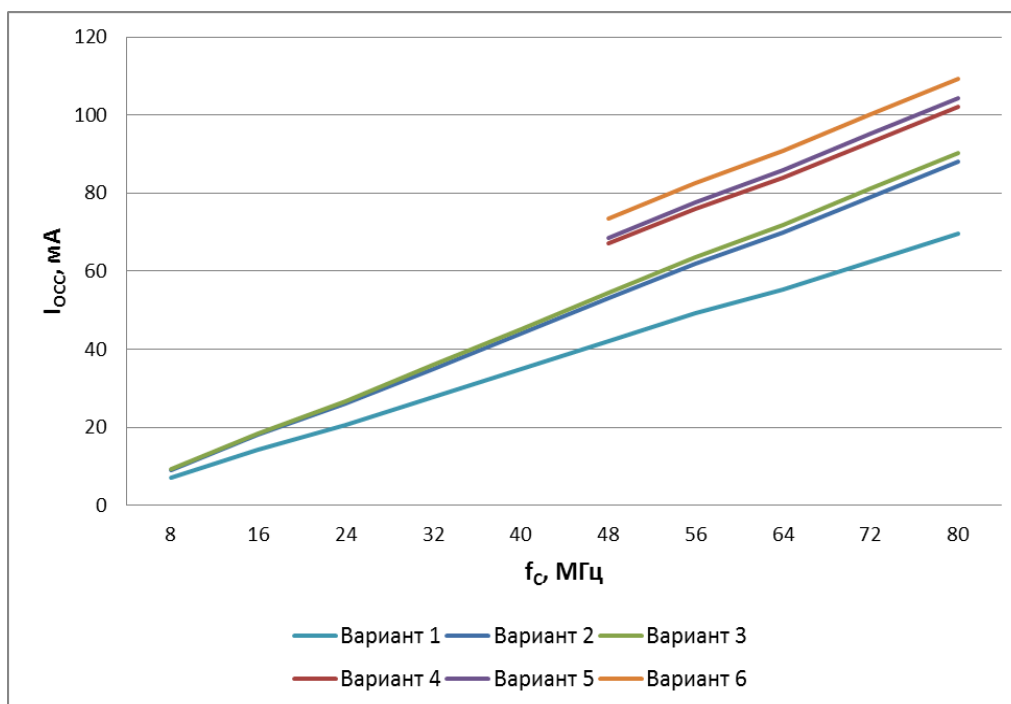


Рисунок 140. Зависимость выходного напряжения низкого уровня от выходного тока низкого уровня при напряжении питания 2,2 и 3,0 В



- 1 – ;
- 2 – + 1xUART+2xSPI+ 3xTIMER;
- 3 – + 1xUART+2xSPI+ 3xTIMER + 1xCAN;
- 4 – + 1xUART+2xSPI+ 3xTIMER + USB;
- 5 – + 1xUART+2xSPI+ 3xTIMER + USB + 1xCAN;
- 6 – + 1xUART+2xSPI+ 3xTIMER + USB + 1xCAN + 1xADC + 1xDAC.

Рисунок 141. Типовая зависимость тока потребления микросхемы от тактовой частоты, в зависимости от набора используемых периферийных блоков

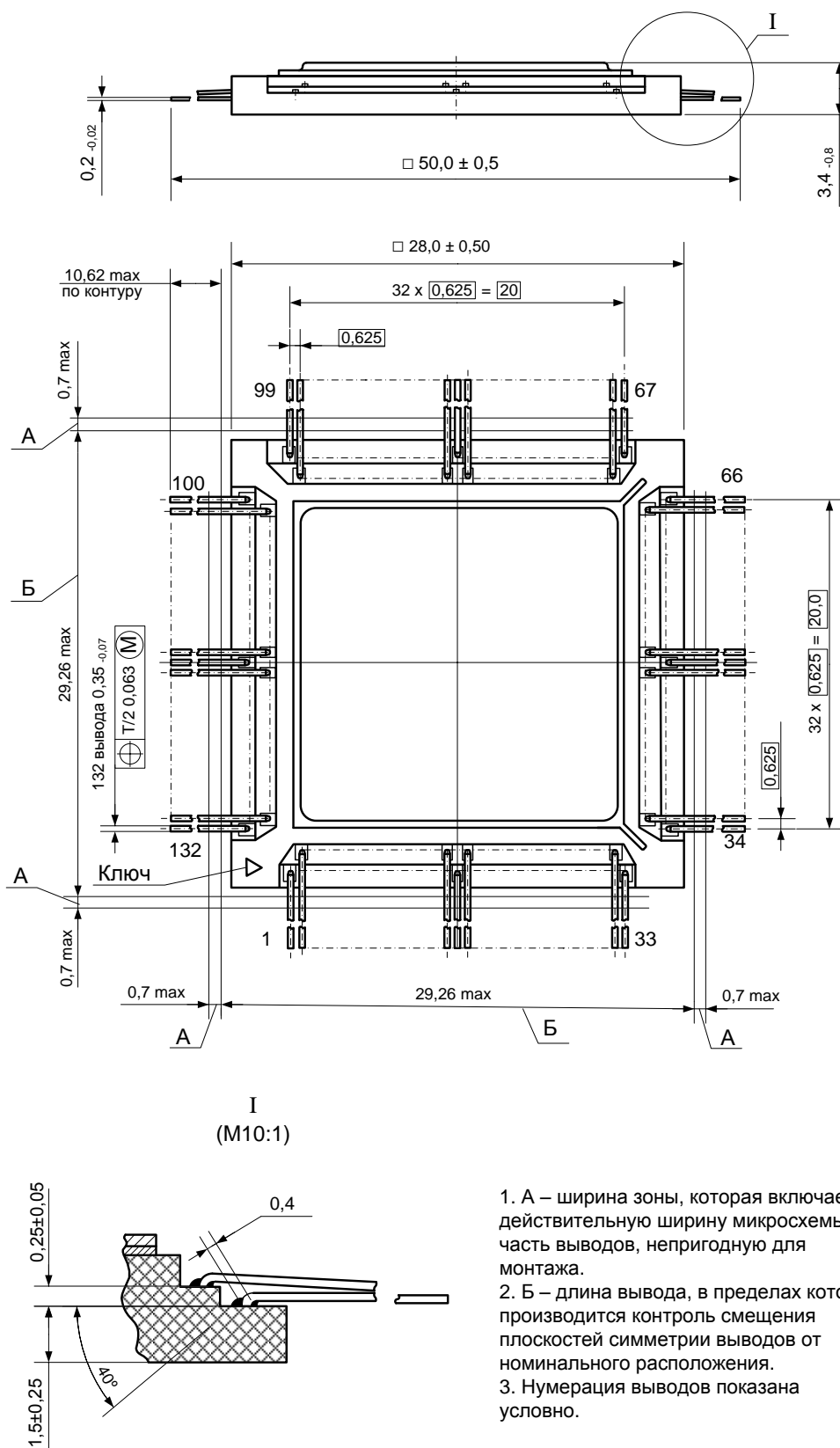
**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

---

**Таблица 491 – Типовые значения потребления основных блоков микросхемы при нормальных условиях: T = 25 °C, Ucc = 3,3 В**

Название блока	Потребление блока	Примечание
	0,5	/
USB	15	( 48 )
TIMER	0,04	/
SPI	0,03	/
UART	0,05	/
CAN	0,03	/
I2C	0,02	/

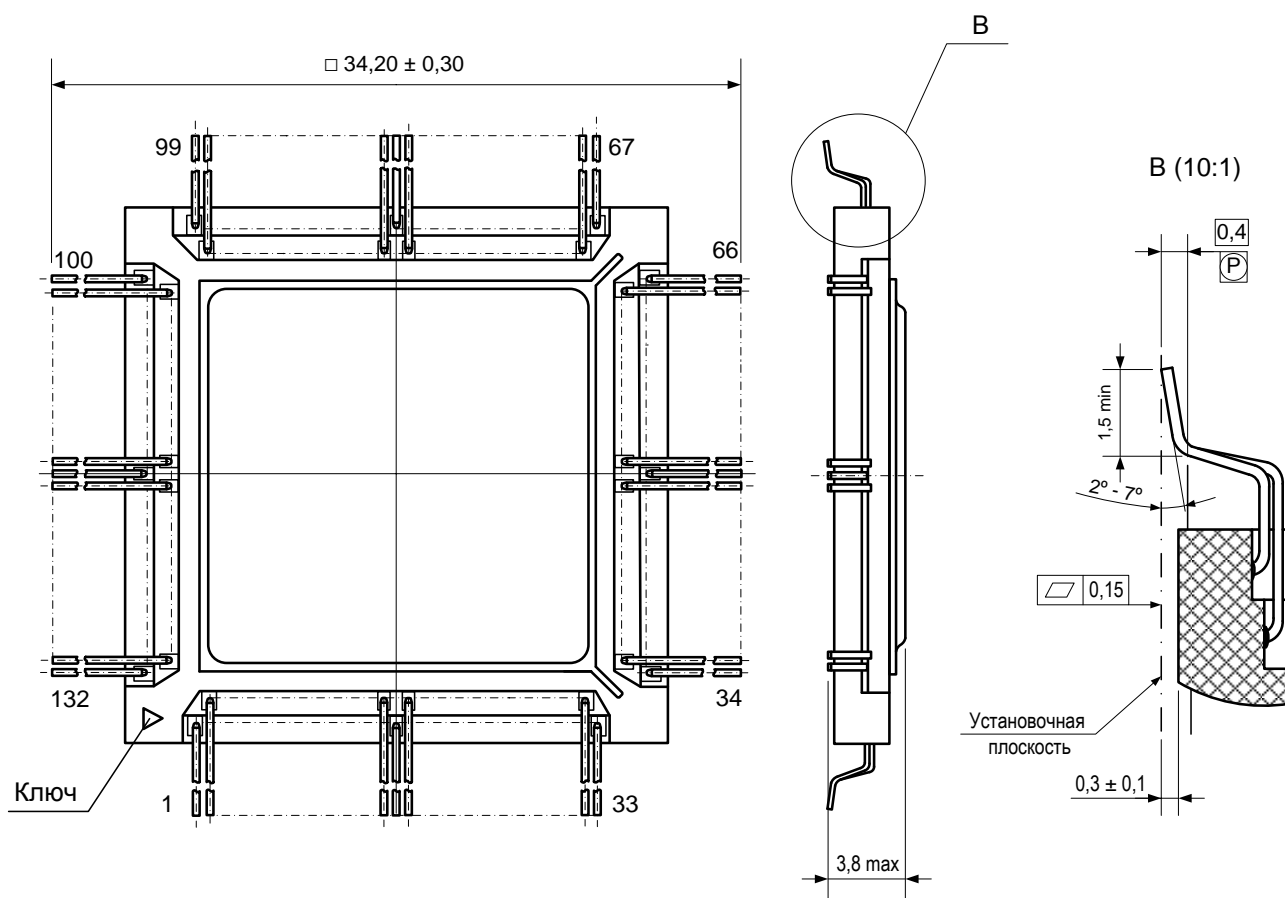
### 38 Габаритный чертеж микросхемы



1. А – ширина зоны, которая включает действительную ширину микросхемы и часть выводов, непригодную для монтажа.
2. Б – длина вывода, в пределах которой производится контроль смещения плоскостей симметрии выводов от номинального расположения.
3. Нумерация выводов показана условно.

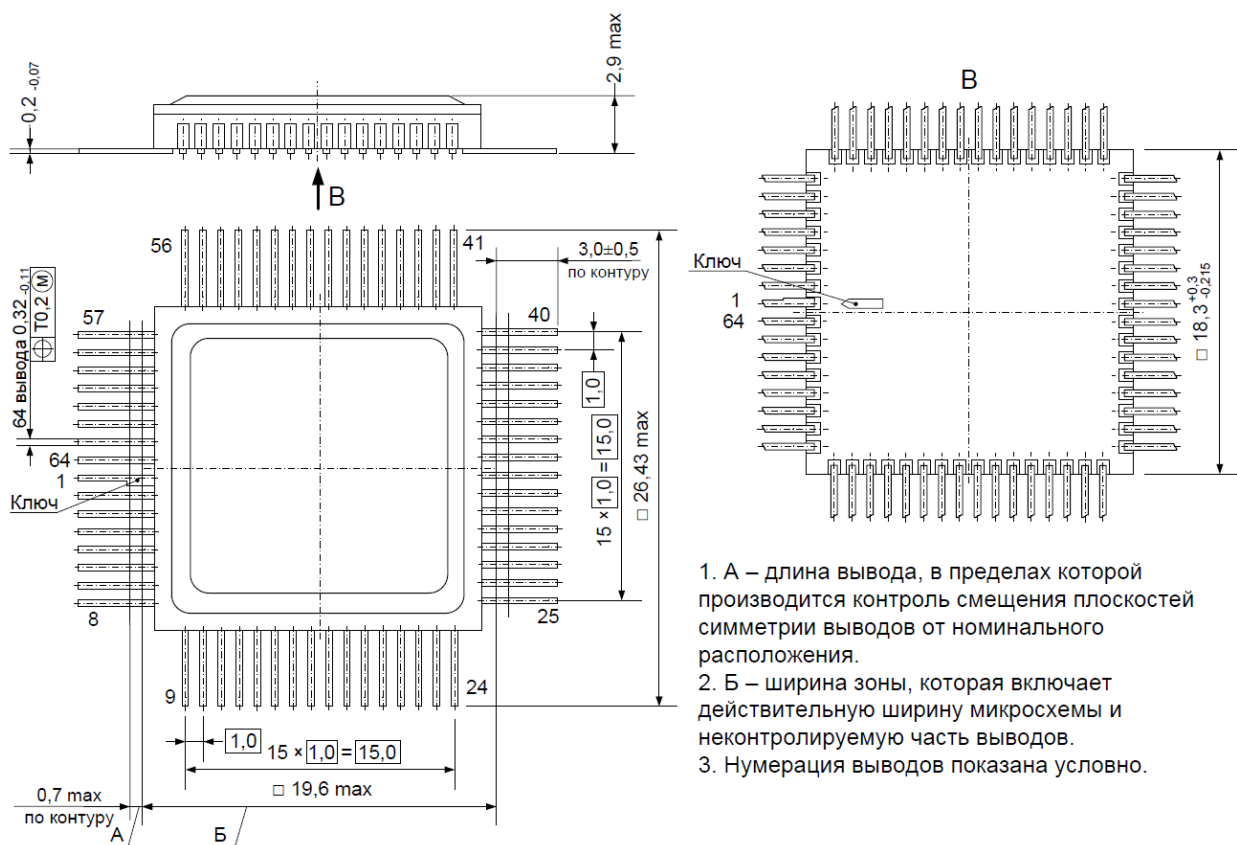
Рисунок 142. Корпус 4229.132-3



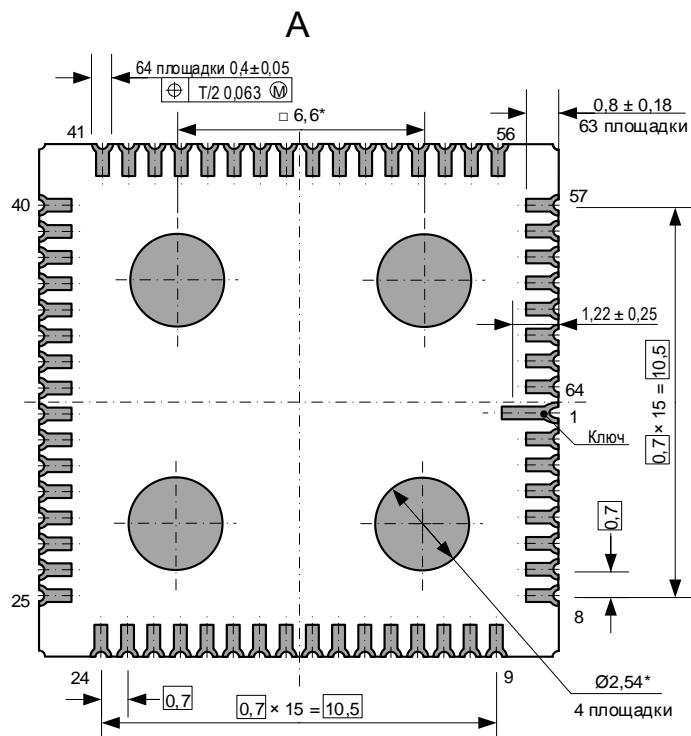
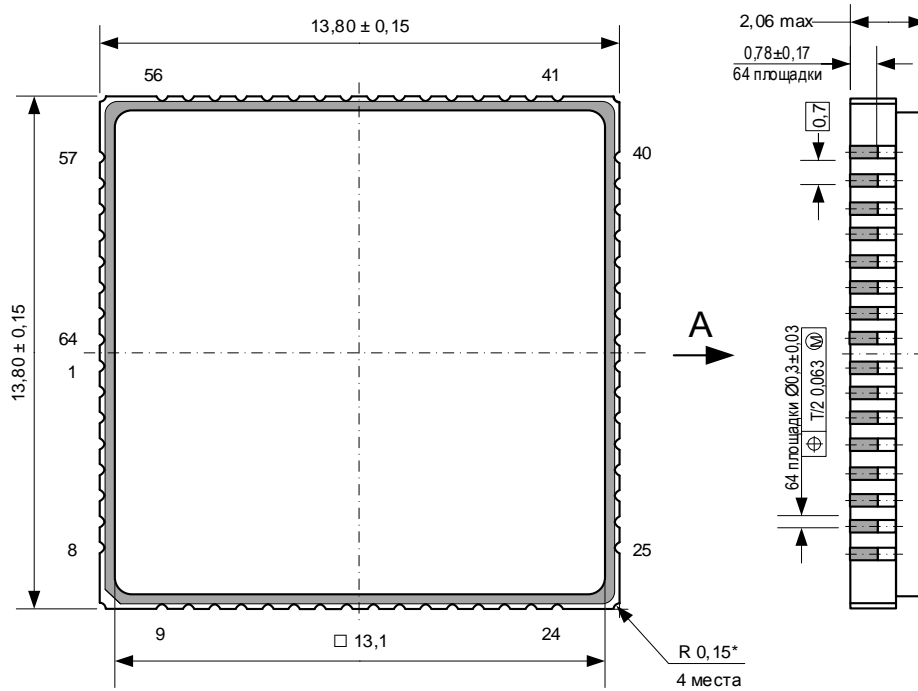


Остальное см. рисунок 142

**Рисунок 143. Корпус 4229.132-3 с формованными выводами**



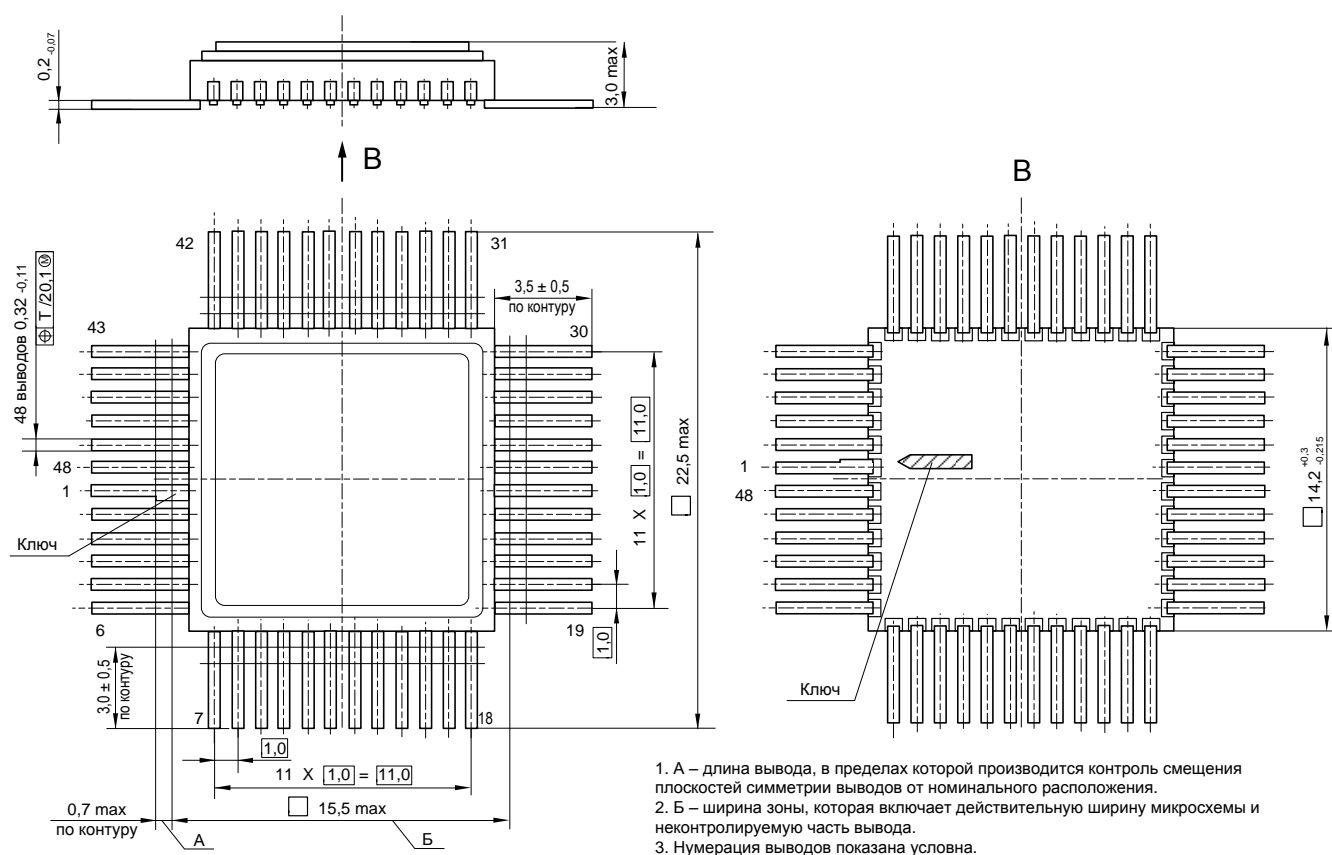
**Рисунок 144. Корпус N18.64-1B**



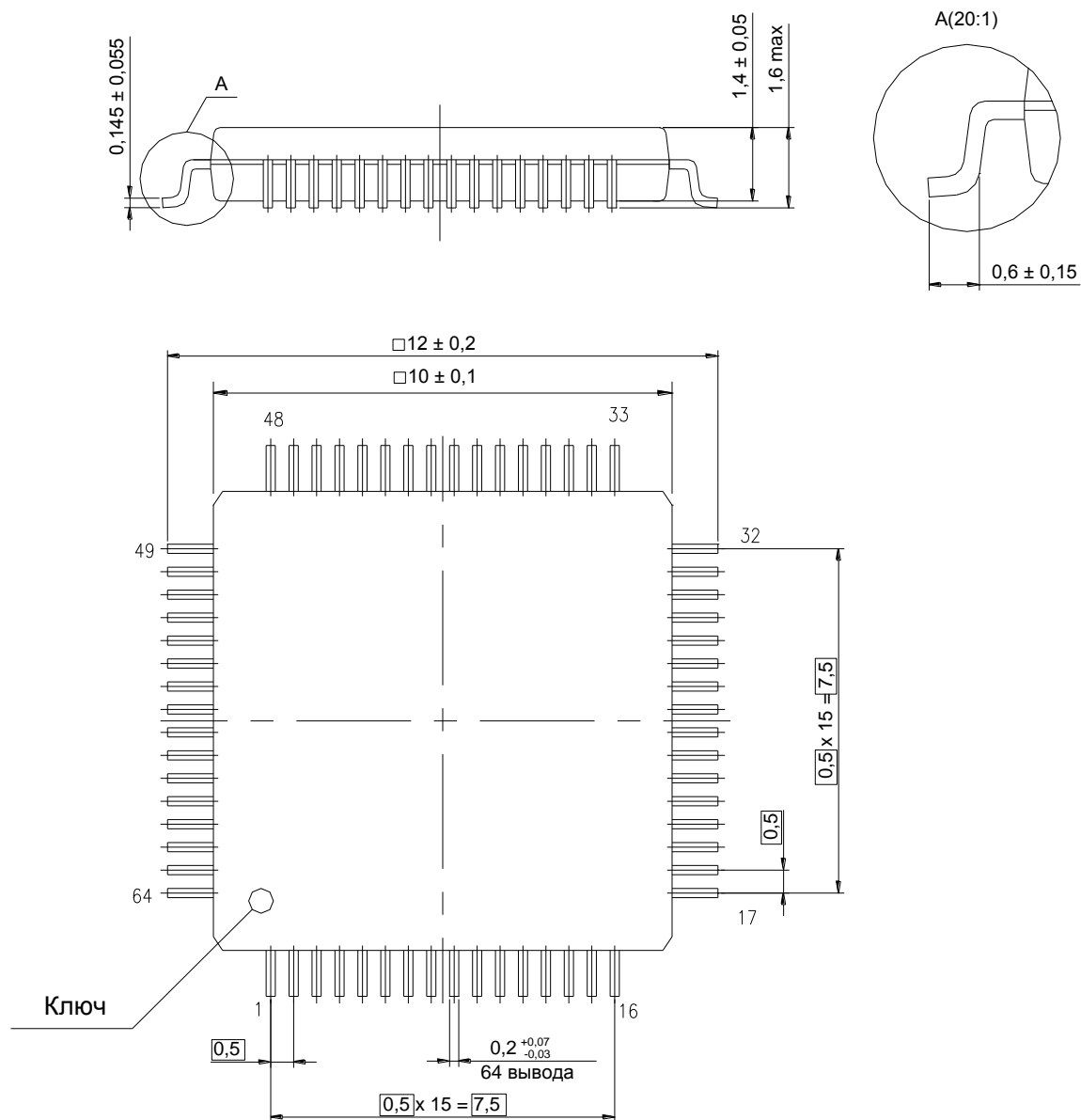
1. \* Размеры для справок.
2. Нумерация выводных площадок показана условно.
3. Заливкой обозначены металлизированные поверхности.

**Рисунок 145. Корпус МК 5153.64-2**

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

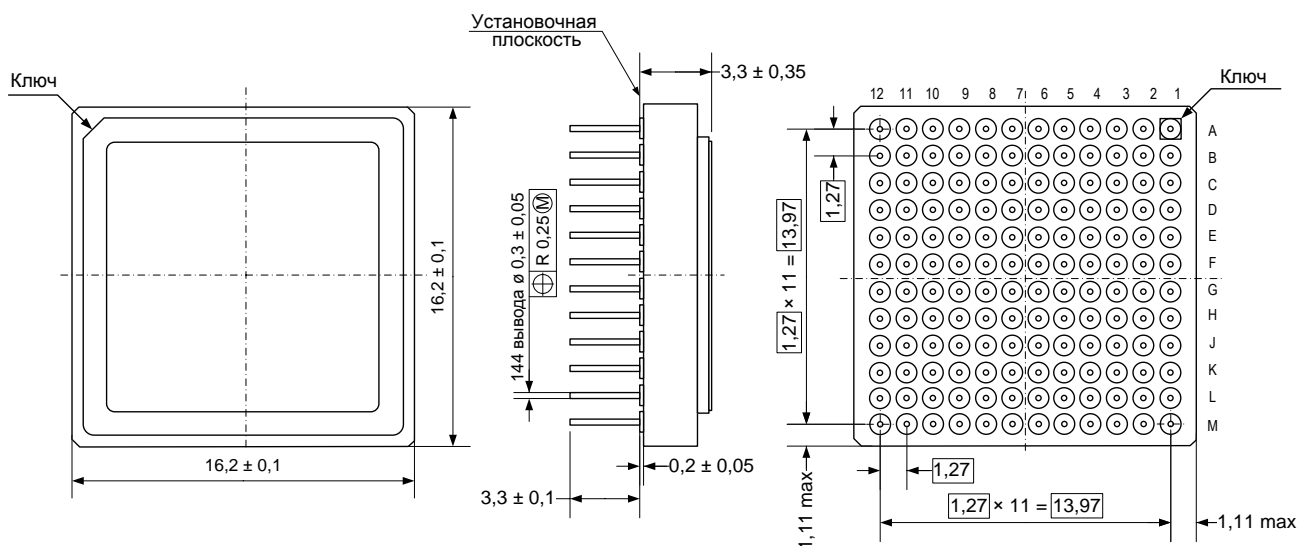


**Рисунок 146. Корпус N16.48-1B**

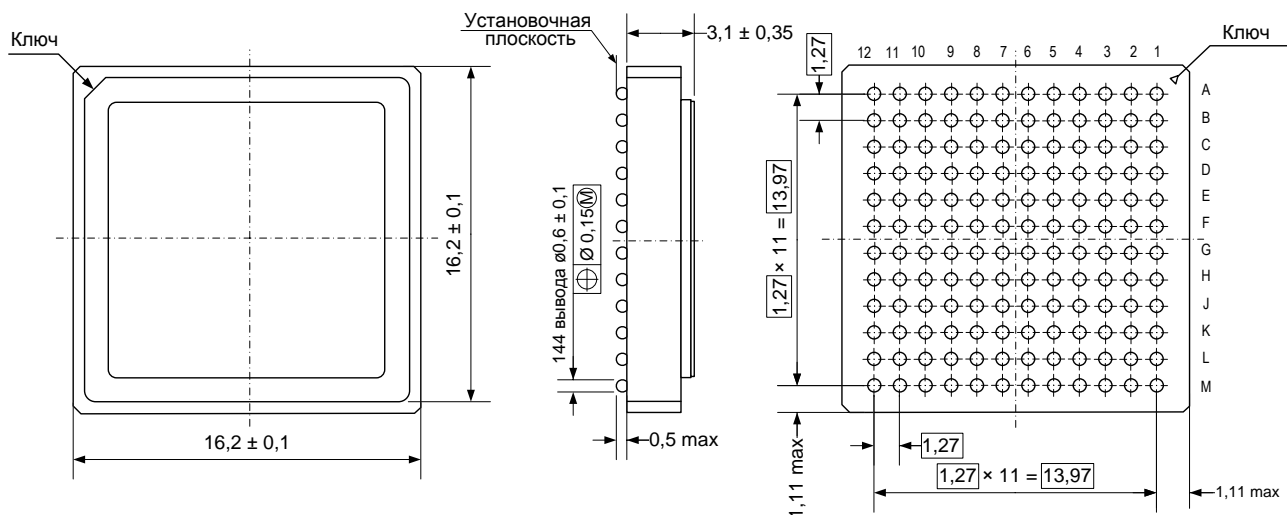


**Рисунок 147. Корпус LQFP64**

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QS, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

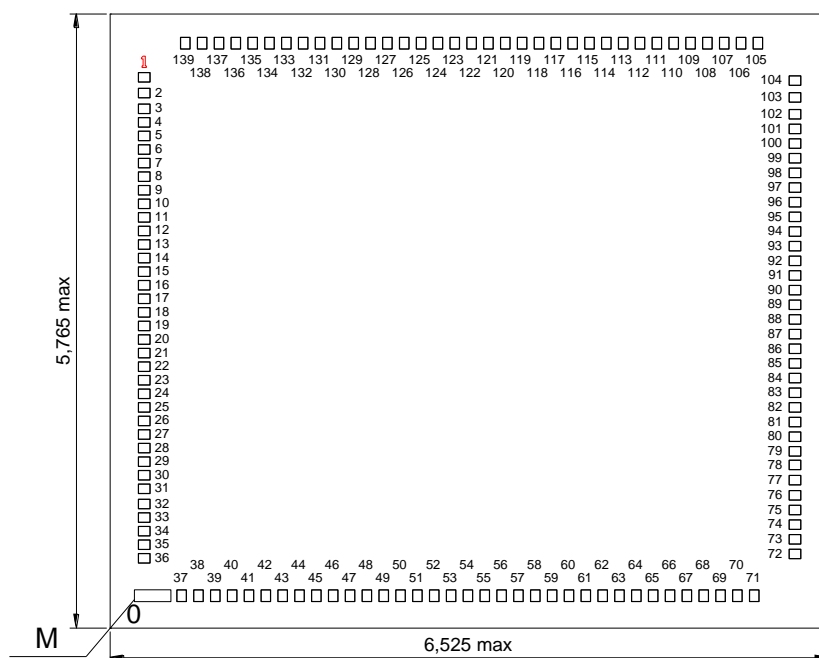


**Рисунок 148. Корпус МК 6109.144-А**



**Рисунок 149. Корпус МК 8307.144-АНЗ**

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**



Размеры КП 1 - 36, 72 - 104 (105 x 58) мкм  
 Размеры КП 37 - 71, 105 - 139 (85 x 105) мкм

- 1 Номера контактных площадок (КП), кроме первой КП, присвоены условно, их расположение соответствует топологическому чертежу.
- 2 М - маркировка кристалла, MLDR44.
- 3 Координаты КП смотри в таблице.
- 4 Толщина кристалла (0,480±0,015) мм.
- 5 Материал КП - сплав AlCu толщиной 3 мкм с содержанием Cu 0,5%.

**Рисунок 150. Кристалл (бескорпусное исполнение 1986BE91H4, K1986BE91H4)**

**Таблица 492 – Координаты КП**

№ КП	Обозначение КП	Координаты КП		№ КП	Обозначение КП	Координаты КП	
		X	Y			X	Y
1	Ucc (VD33!)	0,000	4774,63	17	PF14	0,000	2734,63
2	Ucc (VD33!)	0,000	4629,63	18	PF15	0,000	2609,63
3	PF0	0,000	4484,63	19	PE15	0,000	2484,63
4	PF1	0,000	4359,63	20	PE13	0,000	2359,63
5	PF2	0,000	4234,63	21	PE12	0,000	2234,63
6	PF3	0,000	4109,63	22	USBDR+	0,000	2109,63
7	PF4	0,000	3984,63	23	USBDR+	0,000	1984,63
8	PF5	0,000	3859,63	24	USBDR-	0,000	1859,63
9	PF6	0,000	3734,63	25	USBDR-	0,000	1734,63
10	PF7	0,000	3609,63	26	PE11	0,000	1609,63
11	PF8	0,000	3484,63	27	SHDN	0,000	1484,63
12	PF9	0,000	3359,63	28	Ducc (VDD!)	0,000	1359,63
13	PF10	0,000	3234,63	29	GND (VSS!)	0,000	1234,63
14	PF11	0,000	3109,63	30	GND* (VSS!)	0,000	1109,63
15	PF12	0,000	2984,63	31	Ucc (VD33!)	0,000	984,63
16	PF13	0,000	2859,63	32	Ucc (VD33!)	0,000	844,63

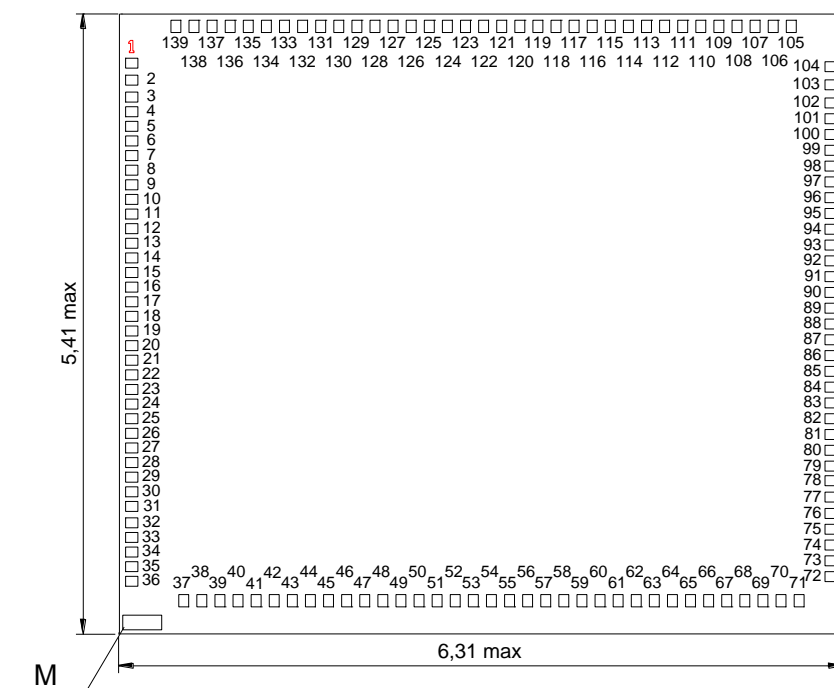
**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

№ КП	Обозначение КП	Координаты КП		№ КП	Обозначение КП	Координаты КП	
		X	Y			X	Y
33	BUcc	0,000	719,63	87	PC9	5989,000	2409,50
34	Standby	0,000	594,63	88	PC8	5989,000	2544,50
35	PE7	0,000	469,63	89	PC7	5989,000	2679,50
36	PE6	0,000	344,63	90	PC6	5989,000	2814,50
37	BDUcc	344,130	0,00	91	PC5	5989,000	2949,50
38	WAKEUP	499,130	0,00	92	PC4	5989,000	3084,50
39	EXT_POR	654,130	0,00	93	PC3	5989,000	3219,50
40	RESET	809,130	0,00	94	PC2	5989,000	3354,50
41	OSC_IN	964,130	0,00	95	PC1	5989,000	3489,50
42	OSC_OUT	1119,130	0,00	96	PC0	5989,000	3624,50
43	PE14	1274,130	0,00	97	PB0	5989,000	3759,50
44	PE5	1429,130	0,00	98	PB1	5989,000	3894,50
45	PE4	1584,130	0,00	99	PB2	5989,000	4029,50
46	PE8	1739,130	0,00	100	PB3	5989,000	4164,50
47	PE3	1894,130	0,00	101	PB4	5989,000	4299,50
48	PE2	2049,130	0,00	102	JTAG_EN	5989,000	4434,50
49	AGND (VSS!)	2204,130	0,00	103	Ucc (VD33!)	5989,000	4589,50
50	AGND (VSS!)	2359,130	0,00	104	Ucc (VD33!)	5989,000	4744,50
51	AUcc1	2514,130	0,00	105	GND (VSS!)	5647,945	5089,00
52	AUcc1	2669,130	0,00	106	Ducc (VDD!)	5492,945	5089,00
53	PE10	2824,130	0,00	107	PB5	5337,945	5089,00
54	PE9	2979,130	0,00	108	PB6	5182,945	5089,00
55	PE1	3134,130	0,00	109	PB7	5027,945	5089,00
56	PE0	3289,130	0,00	110	PB8	4872,945	5089,00
57	AGND (VSS!)	3444,130	0,00	111	PB9	4717,945	5089,00
58	AGND (VSS!)	3599,130	0,00	112	PB10	4562,945	5089,00
59	AUcc	3754,130	0,00	113	PB11	4407,945	5089,00
60	AUcc	3909,130	0,00	114	PB12	4252,945	5089,00
61	PD15	4064,130	0,00	115	PB13	4097,945	5089,00
62	PD14	4219,130	0,00	116	PB14	3942,945	5089,00
63	PD13	4374,130	0,00	117	TM[0]	3787,945	5089,00
64	PD12	4529,130	0,00	118	TM[1]	3632,945	5089,00
65	PD11	4684,130	0,00	119	PB15	3477,945	5089,00
66	PD10	4839,130	0,00	120	PA15	3322,945	5089,00
67	PD8	4994,130	0,00	121	PA14	3167,945	5089,00
68	PD7	5149,130	0,00	122	PA13	3012,945	5089,00
69	PD4	5304,130	0,00	123	PA12	2857,945	5089,00
70	PD0	5459,130	0,00	124	PA11	2702,945	5089,00
71	PD1	5614,130	0,00	125	PA10	2547,945	5089,00
72	PD2	5989,000	384,50	126	TM[2]	2392,945	5089,00
73	PD3	5989,000	519,50	127	Flash VPP	2237,945	5089,00
74	PD5	5989,000	654,50	128	PA9	2082,945	5089,00
75	PD6	5989,000	789,50	129	PA8	1927,945	5089,00
76	PD9	5989,000	924,50	130	PA7	1772,945	5089,00
77	Ucc (VD33!)	5989,000	1064,50	131	PA6	1617,945	5089,00
78	Ucc (VD33!)	5989,000	1204,50	132	PA5	1462,945	5089,00
79	GND (VSS!)	5989,000	1329,50	133	PA4	1307,945	5089,00
80	Ducc (VDD!)	5989,000	1464,50	134	PA3	1152,945	5089,00
81	PC15	5989,000	1599,50	135	PA2	997,945	5089,00
82	PC14	5989,000	1734,50	136	PA1	842,945	5089,00
83	PC13	5989,000	1869,50	137	PA0	687,945	5089,00
84	PC12	5989,000	2004,50	138	Ducc (VDD!)	532,945	5089,00
85	PC11	5989,000	2139,50	139	GND (VSS!)	377,945	5089,00
86	PC10	5989,000	2274,50				

—



**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK, K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**



Размеры КП 85 x 105 мкм<sup>2</sup>

- 1 Номера контактных площадок (КП) присвоены условно, кроме первой КП и их расположение соответствует топологическому чертежу.  
 2 М - маркировка кристалла, MLDR75.

**Рисунок 151. Кристалл (бескорпусное исполнение 1986BE94H4, K1986BE94H4)**

**Таблица 493 – Координаты КП**

№ КП	Обозначение КП	Координаты КП		№ КП	Обозначение КП	Координаты КП	
		X	Y			X	Y
1	Ucc (VD33!)	0,000	4774,63	24	USBDR-	0,000	1859,63
2	Ucc (VD33!)	0,000	4629,63	25	USBDR-	0,000	1734,63
3	PF0	0,000	4484,63	26	PE11	0,000	1609,63
4	PF1	0,000	4359,63	27	SHDN	0,000	1484,63
5	PF2	0,000	4234,63	28	Ducc (VDD!)	0,000	1359,63
6	PF3	0,000	4109,63	29	GND (VSS!)	0,000	1234,63
7	PF4	0,000	3984,63	30	GND* (VSS!)	0,000	1109,63
8	PF5	0,000	3859,63	31	Ucc (VD33!)	0,000	984,63
9	PF6	0,000	3734,63	32	Ucc (VD33!)	0,000	844,63
10	PF7	0,000	3609,63	33	BUcc	0,000	719,63
11	PF8	0,000	3484,63	34	Standby	0,000	594,63
12	PF9	0,000	3359,63	35	PE7	0,000	469,63
13	PF10	0,000	3234,63	36	PE6	0,000	344,63
14	PF11	0,000	3109,63	37	BDUcc	344,130	0,00
15	PF12	0,000	2984,63	38	WAKEUP	499,130	0,00
16	PF13	0,000	2859,63	39	EXT_POR	654,130	0,00
17	PF14	0,000	2734,63	40	RESET	809,130	0,00
18	PF15	0,000	2609,63	41	OSC_IN	964,130	0,00
19	PE15	0,000	2484,63	42	OSC_OUT	1119,130	0,00
20	PE13	0,000	2359,63	43	PE14	1274,130	0,00
21	PE12	0,000	2234,63	44	PE5	1429,130	0,00
22	USBDR+	0,000	2109,63	45	PE4	1584,130	0,00
23	USBDR+	0,000	1984,63	46	PE8	1739,130	0,00

**Спецификация микросхем серии 1986BE9ху, K1986BE9ху, K1986BE9хуK,  
K1986BE92QI, K1986BE92QC, 1986BE91H4, K1986BE91H4, 1986BE94H4, K1986BE94H4**

№ КП	Обозначение КП	Координаты КП		№ КП	Обозначение КП	Координаты КП	
		X	Y			X	Y
47	PE3	1894,130	0,00	94	PC2	5989,000	3354,50
48	PE2	2049,130	0,00	95	PC1	5989,000	3489,50
49	AGND (VSS!)	2204,130	0,00	96	PC0	5989,000	3624,50
50	AGND (VSS!)	2359,130	0,00	97	PB0	5989,000	3759,50
51	AUcc1	2514,130	0,00	98	PB1	5989,000	3894,50
52	AUcc1	2669,130	0,00	99	PB2	5989,000	4029,50
53	PE10	2824,130	0,00	100	PB3	5989,000	4164,50
54	PE9	2979,130	0,00	101	PB4	5989,000	4299,50
55	PE1	3134,130	0,00	102	JTAG_EN	5989,000	4434,50
56	PE0	3289,130	0,00	103	Ucc (VD33!)	5989,000	4589,50
57	AGND (VSS!)	3444,130	0,00	104	Ucc (VD33!)	5989,000	4744,50
58	AGND (VSS!)	3599,130	0,00	105	GND (VSS!)	5647,945	5089,00
59	AUcc	3754,130	0,00	106	Ducc (VDD!)	5492,945	5089,00
60	AUcc	3909,130	0,00	107	PB5	5337,945	5089,00
61	PD15	4064,130	0,00	108	PB6	5182,945	5089,00
62	PD14	4219,130	0,00	109	PB7	5027,945	5089,00
63	PD13	4374,130	0,00	110	PB8	4872,945	5089,00
64	PD12	4529,130	0,00	111	PB9	4717,945	5089,00
65	PD11	4684,130	0,00	112	PB10	4562,945	5089,00
66	PD10	4839,130	0,00	113	PB11	4407,945	5089,00
67	PD8	4994,130	0,00	114	PB12	4252,945	5089,00
68	PD7	5149,130	0,00	115	PB13	4097,945	5089,00
69	PD4	5304,130	0,00	116	PB14	3942,945	5089,00
70	PD0	5459,130	0,00	117	TM[0]	3787,945	5089,00
71	PD1	5614,130	0,00	118	TM[1]	3632,945	5089,00
72	PD2	5989,000	384,50	119	PB15	3477,945	5089,00
73	PD3	5989,000	519,50	120	PA15	3322,945	5089,00
74	PD5	5989,000	654,50	121	PA14	3167,945	5089,00
75	PD6	5989,000	789,50	122	PA13	3012,945	5089,00
76	PD9	5989,000	924,50	123	PA12	2857,945	5089,00
77	Ucc (VD33!)	5989,000	1064,50	124	PA11	2702,945	5089,00
78	Ucc (VD33!)	5989,000	1204,50	125	PA10	2547,945	5089,00
79	GND (VSS!)	5989,000	1329,50	126	TM[2]	2392,945	5089,00
80	Ducc (VDD!)	5989,000	1464,50	127	Flash VPP	2237,945	5089,00
81	PC15	5989,000	1599,50	128	PA9	2082,945	5089,00
82	PC14	5989,000	1734,50	129	PA8	1927,945	5089,00
83	PC13	5989,000	1869,50	130	PA7	1772,945	5089,00
84	PC12	5989,000	2004,50	131	PA6	1617,945	5089,00
85	PC11	5989,000	2139,50	132	PA5	1462,945	5089,00
86	PC10	5989,000	2274,50	133	PA4	1307,945	5089,00
87	PC9	5989,000	2409,50	134	PA3	1152,945	5089,00
88	PC8	5989,000	2544,50	135	PA2	997,945	5089,00
89	PC7	5989,000	2679,50	136	PA1	842,945	5089,00
90	PC6	5989,000	2814,50	137	PA0	687,945	5089,00
91	PC5	5989,000	2949,50	138	Ducc (VDD!)	532,945	5089,00
92	PC4	5989,000	3084,50	139	GND (VSS!)	377,945	5089,00
93	PC3	5989,000	3219,50				

—

### 39 Информация для заказа

Обозначение	Маркировка	Тип корпуса	Температурный диапазон
1986 91	1986 91	4229.132-3	60 – 125 °
1986 91	1986 91	4229.132-3	60 – 125 °
1986 91	1986 91 •	4229.132-3	0 – 70 °
1986 91	1986 91 •-1	4229.132-3	0 – 70 °
1986 91	1986 91 •-2	4229.132-3	0 – 70 °
1986 92	1986 92	.18.64-1	60 – 125 °
1986 92	1986 92	.18.64-1	60 – 125 °
1986 92	1986 92 •	.18.64-1	0 – 70 °
1986 92 1	1986 92 1	5153.64-2	60 – 125 °
1986 92 1	1986 92 1	5153.64-2	60 – 125 °
1986 92 1	1986 92 1•	5153.64-2	0 – 70 °
1986 92QC	MDR32F9Q2C <u>ARM</u>	LQFP64	0 – 70 °
1986 92QI	MDR32F9Q2I <u>ARM</u>	LQFP64	40 – 85 °
1986 92	1986 92 •-1	.18.64-1	0 – 70 °
1986 92	1986 92 •-2	.18.64-1	0 – 70 °
1986 93	1986 93	.16.48-1	60 – 125 °
1986 93	1986 93	.16.48-1	60 – 125 °
1986 93	1986 93 •	.16.48-1	0 – 70 °
1986 93	1986 93 •-1	.16.48-1	0 – 70 °
1986 93	1986 93 •-2	.16.48-1	0 – 70 °
1986 94	1986 94	4229.132-3	60 – 125 °
1986 94	1986 94	4229.132-3	60 – 125 °
1986 94	1986 94 •	4229.132-3	0 – 70 °
1986 94	1986 94	6109.144-	60 – 125 °
1986 94	1986 94	6109.144-	60 – 125 °
1986 94	1986 94 •	6109.144-	0 – 70 °
1986 94	1986 94	8307.144-	60 – 125 °
1986 94	1986 94	8307.144-	60 – 125 °
1986 94	1986 94 •	8307.144-	0 – 70 °

1.

1986 94 4, 1986 94 4

2.

« »  
« »

« ».



**Спецификация микросхем серии 1986BE9ху, К1986BE9ху, К1986BE9хуК,  
К1986BE92QI, К1986BE92QC, 1986BE91H4, К1986BE91H4, 1986BE94H4, К1986BE94H4**

			109	172, 173
			-	177
			MDR_PORTx	
24	04.10.2016	3.10.0		
25	24.07.2018	3.11.0		519, 527
26	30.08.2018	3.12.0		17, 23, 214, 228, 270, 279, 291, 306, 307, 314, 485, 521
27	21.09.2018	3.13.0	1986 92 1, 1986 94 , 1986 94	
28	12.10.2018	3.14.0	6109.144- , 8307.144-	25, 26
29	01.03.2019	3.15.0	145 23	523 252. 275 62
30	06.03.2019	3.16.0	PE4, PE5, PE6, PE7 5153.64-2	11
31	16.04.2019	3.17.0	281, 282 EV_DELAY CHy_CNTRL2. 18.64-1	306 522
32	10.09.2019	3.18.0	, 6 369 28 30, 31 397	319 402 165 438
33	16.09.2019	3.19.0	16.48-1	524