

.431295.008

1	4
1.1	5
1.2	5
1.3	5
2	13
2.1	-	13
2.2	(CPU).....	13
2.3	14
2.4	CPU.....	14
2.5	().....	15
2.6	15
2.7	16
2.7.1	17
2.7.2	17
2.7.3	22
2.7.4	24
2.7.5	24
3	24
3.1	25
3.2	26
3.3	28
3.4	30
3.5	32
4	40
4.1	- (PWM).....	40
4.2	40
4.3	(HSI).....	40
4.4	(HSO).....	41
4.5	42
4.6	43
4.7	-	43
4.8	43
5	44
5.1	46
5.2	47
5.3	48
5.4	49
5.5	50
6	- ().....	52
7	53
7.1	_1	53
7.2	_2	53
7.3	_2	54
7.4	54

.431295.008

1874 36, 1874 36,
1874 36 , 1874 36

2 120

8	(HSI)	55
8.1	55
8.2	(HSI)	56
8.3	HSI.....	57
8.4	HSI	58
8.5	58
9	(HSO)	59
9.1	HSO	59
9.2	(HSO).....	60
9.3	HSO	61
9.4	HSO	62
9.5	HSO	63
9.6 PWM	HSO	63
9.7	HSO.....	64
10	65
10.1	65
10.2	67
10.3	68
10.4	70
11	().....	71
11.1	71
11.2	72
11.3	73
11.4	73
11.5	74
12	-	76
12.1	76
12.2	-	77
12.3	77
13	79
14	83
15	85
15.1	(CCR).....	86
15.2	(READY).....	86
15.3 HOLD#/HLDA#	89
15.4	90
15.5	91
15.6	HOLD#.....	91
16	92
	119
	120

.431295.008

1874: 1874 36, 1874 36, 1874 36 , 1874 36 ,
16-
8 .

1874 36, 1874 36, 1874 36 , 1874 36
.431280.169 : 1874 36, 1874 36 -
6108.68-1; 1874 36, 1874 36 - 4235.88-1 .
1874 36, 1874 36, 1874 36 , 1874 36

83C196KB12 Intel.

- PICE-196 (POD
1874 36, 1874 36) ;
- PDS-96;
- -96 -96.

1874 36 , 1874 36 . 1874 36, 1874 36,

- :
- : (473) 226-20-35
- : (473) 226-98-95
- E-mail: niiet@niiet.ru
- http://www.niiet.ru

1

8 16- -

1.1

- 8- - -5;
 - - -8, 16;
 - - -232;
 - - -16;
 - - -64 ;
 - - -8 ;
 - - -105;
 - - -1, ..., 8;
 - 16- -6 (4);
 - / -28/16;

-8- 10- ;
 - (HOLD#/HLDA#);
 -16- ;

1.2

.757644.231

(7,84 × 6,74) ².

1) 68- (PGA) 6108.68-1 (" -68") - 1874 36, 1874 36 ;

2) 88- (LCC) 4235.88-1 (" -88") - 1874 36, 1874 36 .

1.3

- - 5,0;
 - , - ±0,5;
 - - 500;
 - - (3,5 - 20);
 - 60 85 .

.431280.169

1.3.1.

.431295.008

5

1.3.1 -

()

1	U_{OL}	-	0,45	$I_{OL} = 3,2$	1
2	U_{I1}	3,8	-	$I_{OH} = -0,03$	2
3	U_{OH2}	3,8	-	$I_{OH} = -3,2$	3
4	I_{IH}	-650	-	$U_{IH} = 2,0$	8
5	I_{IL}	-50	-	$U_{IL} = 0,45$	9
6	I_{ILH1} I_{ILL1}	-10	10	$U_{IH} = 5,2$ $U_{IH} = 5,5$	4 5
7	I_{ILH2} I_{ILL2}	-3	3	$U_{IH} = 5,5$	6 7
8	I_{CC1}	-	100	$U_{CC1} = 5,5$ $U_{CC2} = 5,5$	10
9	I_{CC2}	-	10	$U_{CC1} = 5,5$ $U_{CC2} = 5,5$	11
10	$t_{(C,LH-CO,H)}$	-	85	$U_{CC1} = 5,0$ $U_{CC2} = 5,0$ $U_{IH} = 3,0$	
11	1 2				12 13

1 AD.0 – AD.15, RD#, WR#, BHE#, INST, HSO.0 – HSO.5, PWM, CL OUT, P2.0/T D, R D (), 1.0 – 1.7, 2.6, 2.7.

2 1.0 – 1.7, 2.6, 2.7.

3 AD.0 – AD.15, RD#, WR#, BHE#, INST, HSO.0 – HSO.5, PWM, CL OUT, P2.0/T D, R D (), ALE.

4 HSI.0 – HSI.3, EA#, READY, BW, NMI, P2.1/R D, P2.2/E TINT, P2.3/T2CLK, P2.4/T2RST. U_{IH} , U_{IL} .

5 HSI.0 – HSI.3, EA#, READY, BW, NMI, P2.1/R D, P2.2/E TINT, P2.3/T2CLK, P2.4/T2RST. U_{IL} , U_{IH} .

6 0.0 – 0.7. U_{IH} , U_{IL} .

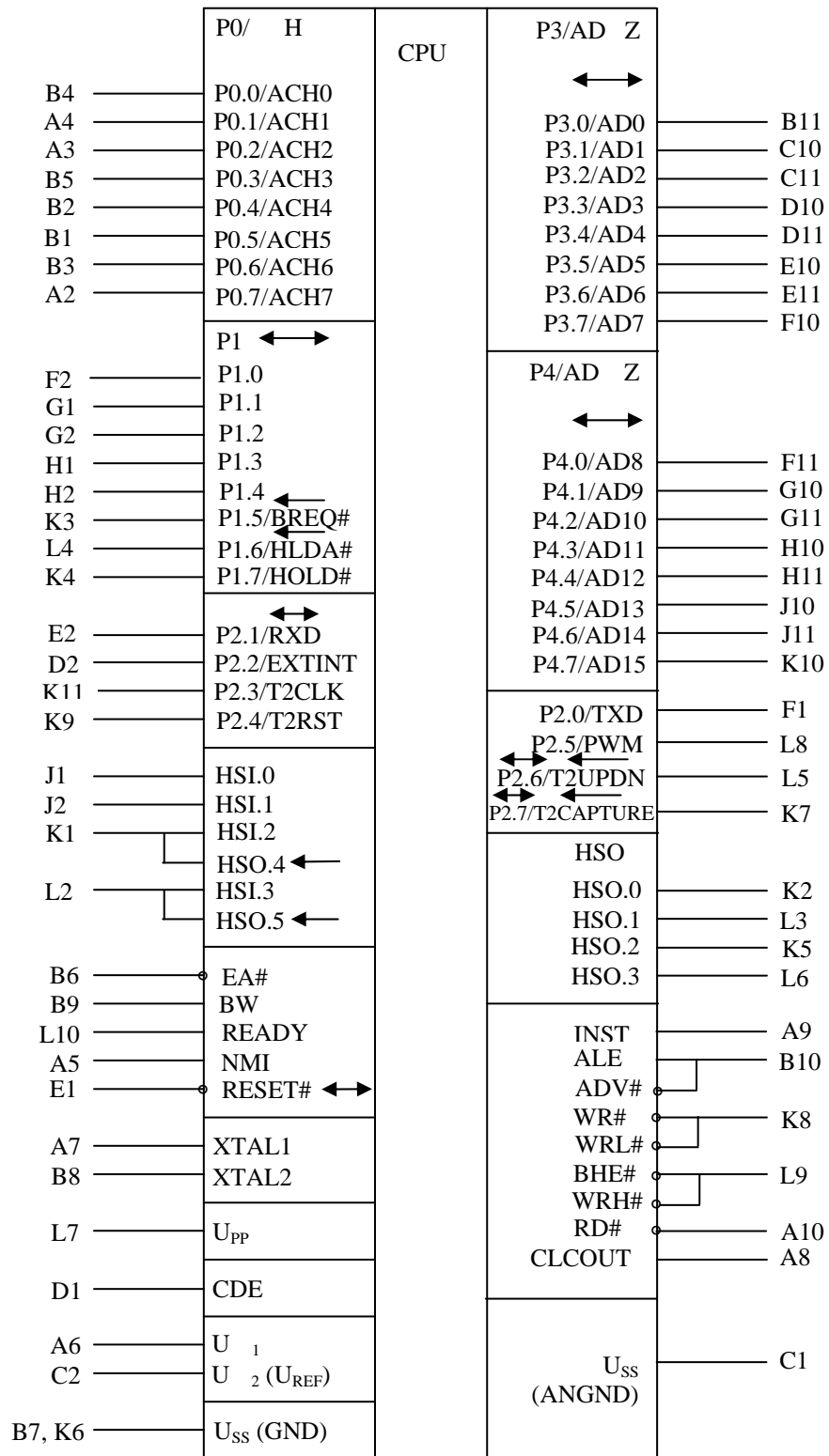
7 0.0 – 0.7. U_{IL} , U_{IH} .

Окончание таблицы 1.3.1

8	/	1.0 – 1.7, 2.6, 2.7.	U_{IH} ,
9	U_{IL} .	/	1.0 – 1.7, 2.6, 2.7.
10	U_{IH} .		$U_{CC1} = 5,5$,
$f_C = 12/20$	*		
11		-	
$f_C = 12/20$	*		
12		1	:
- $f_C = (3,5 - 20)$	*		
- U_{IL}	:	0.0 – 0,2 , 0.1 – 0,16 , 0.2 – 0,32 , 0.3 – 0,64 ;	
- U_{IH}	:	0.4 – 2,8 , 0.5 – 3,2 , 0.6 – 3,8 , 0.7 – 4,2 ;	
-	:	$U_{IH} = 3$, $U_{IL} = 0$.	
13		2	$f_C = 3,5$
* 20	-	1874 36, 1874 36; 12	- 1874 36 , 1874 36 .

1.3.2 –

1	1-	U_{CC1}	4,5	5,5
				7,0
2	2-	U_{CC2}	4,0	5,5
				7,0
3	, M	f_C	3,5	12/20*
4		U_{IL}	0	0,8
				- 0,5
5		U_{IH}	$0,2U_{CC1} + 1,0$	$U_{CC1} + 0,5$
				7,0 $U_{CC1} = 7$
6		U_{IHCI}	$0,7U_{CC1}$	$U_{CC1} + 0,5$
				7,0 $U_{CC1} = 7$
7	" "	$U_{IHRESET}$	2,6	$U_{CC1} + 0,5$
				7,0 $U_{CC1} = 7$
8	RD#, WR#, ALE, BHE#, INST, HSO, P2.5/PWM, CL OUT, 3.0 – 3.7, 4.0 – 4.7, P2.0/T D, P2.1/R D,	I_{OH}		-3,2
				-10
9		I_{OL}		3,2
				10
10		L		80
				200
11		t_{LH}, t_{HL}	-	7
				25
10	-			1, 2, 4 – 7
*		1874 36	1874 36	,
		1874 36	1874 36	.



1874 36

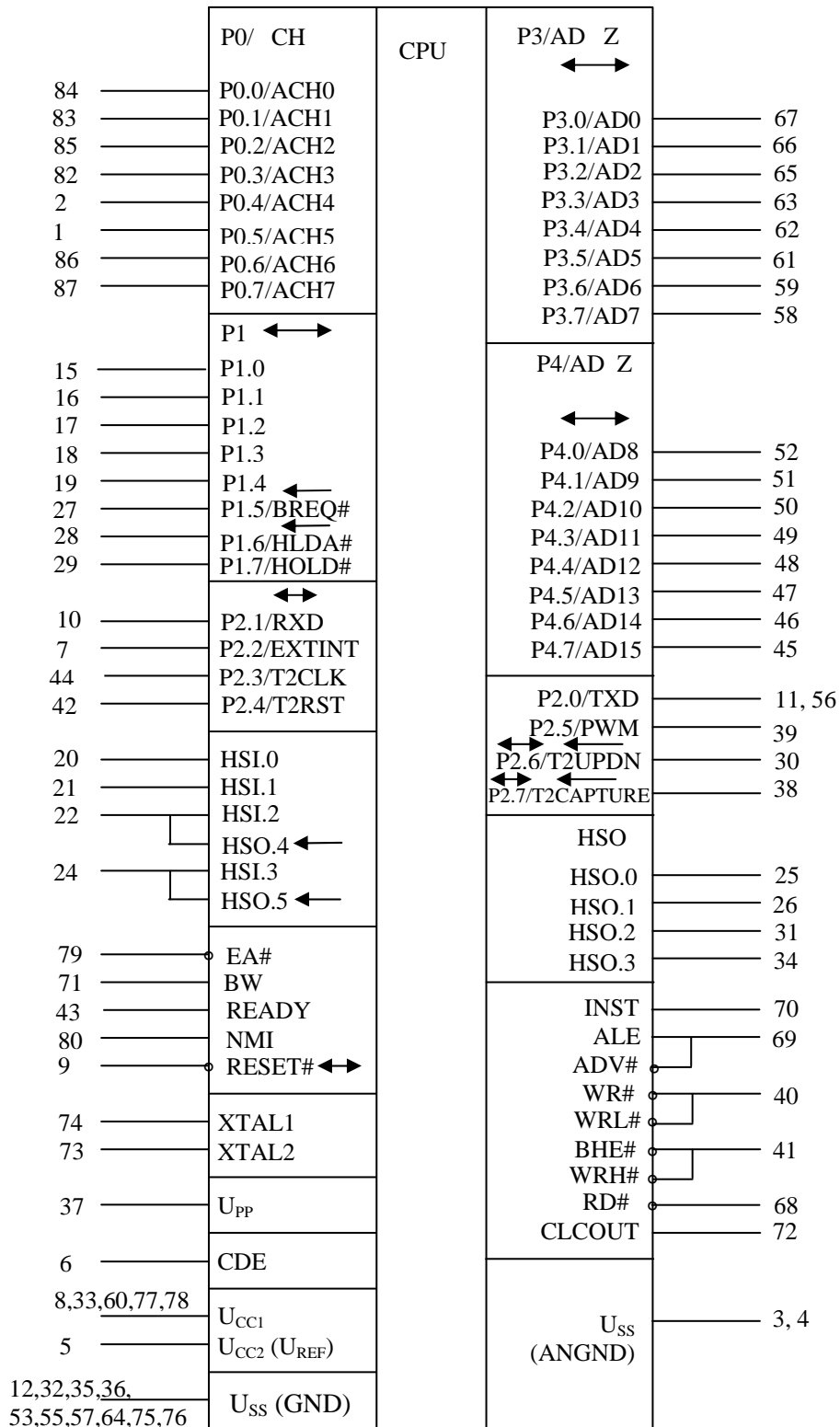
1.3.1 -

6108.68-1 (" -68")

1874 36

.431295.008

8



1874 36 1.3.2 - 4235.88-1 (" -88") 1874 36,

1.3.3 –

		6108.68-14235.88-1					
1	2	3	4		5	6	
P0.0	B4	84	"	0, 0	"	I	
				,	0	I	.0
P0.1	A4	83	"	0, 1	"	I	
				,	1	I	ACH.1
P0.2	A3	85	"	0, 2	"	I	
				,	2	I	ACH.2
P0.3	B5	82	"	0, 3	"	I	
				,	3	I	CH.3
P0.4	B2	2	"	0, 4	"	I	
				,	4	I	CH.4
P0.5	B1	1	"	0, 5	"	I	
				,	5	I	ACH.5
P0.6	B3	86	"	0, 6	"	I	
				,	6	I	CH.6
P0.7	A2	87	"	0, 7	"	I	
				,	7	I	.7
P1.0	F2	15	/	"	1, 0	"	I/O
P1.1	G1	16	/	"	1, 1	"	I/O
P1.2	G2	17	/	"	1, 2	"	I/O
P1.3	H1	18	/	"	1, 3	"	I/O
P1.4	H2	19	/	"	1, 4	"	I/O
P1.5	K3	27	/	"	1, 5	"	I/O
			"	"	"	"	O
P1.6	L4	28	/	"	1, 6	"	I/O
			"	"	"	"	O
P1.7	K4	29	/	"	1, 7	"	I/O
			"	"	"	"	I
P2.0	F1	11, 56	"	2, 0	"	"	O
			"	"	"	"	O
P2.1	E2	10	"	2, 1	"	"	I
			/	"	"	"	I/O
P2.2	D2	7	"	2, 2	"	"	I
			"	"	"	"	I
P2.3	K11	44	"	2, 3	"	"	I
			"	"	"	2"	I
P2.4	K9	42	"	2, 4	"	"	I
			"	"	"	2"	I
P2.5	L8	39	"	2, 5	"	"	O
			"	"	"	"	O
P2.6	L5	30	/	"	2, 6	"	I/O
			"	"	"	2"	I

Продолжение таблицы 1.3.3

1	2	3	4				5	6
P2.7	K7	38	/	"	2,7	"	I/O	T2CAPTURE
			"			2"	I	
P3.0	B11	67	/	"	3,0	"	I/O	D0
			/	"	-	,0	I/O/Z	
P3.1	C10	66	/	"	3,1	"	I/O	AD1
			/	"	-	,1	I/O/Z	
P3.2	C11	65	/	"	3,2	"	I/O	AD2
			/	"	-	,2	I/O/Z	
P3.3	D10	63	/	"	3,3	"	I/O	AD3
			/	"	-	,3	I/O/Z	
P3.4	D11	62	/	"	3,4	"	I/O	AD4
			/	"	-	,4	I/O/Z	
P3.5	E10	61	/	"	3,5	"	I/O	AD5
			/	"	-	,5	I/O/Z	
P3.6	E11	59	/	"	3,6	"	I/O	AD6
			/	"	-	,6	I/O/Z	
P3.7	F10	58	/	"	3,7	"	I/O	AD7
			/	"	-	,7	I/O/Z	
P4.0	F11	52	/	"	4,0	"	I/O	AD8
			/	"	-	,8	I/O/Z	
P4.1	G10	51	/	"	4,1	"	I/O	AD9
			/	"	-	,9	I/O/Z	
P4.2	G11	50	/	"	4,2	"	I/O	AD10
			/	"	-	,10	I/O/Z	
P4.3	H10	49	/	"	4,3	"	I/O	AD11
			/	"	-	,11	I/O/Z	
P4.4	H11	48	/	"	4,4	"	I/O	AD12
			/	"	-	,12	I/O/Z	
P4.5	J10	47	/	"	4,5	"	I/O	AD13
			/	"	-	,13	I/O/Z	
P4.6	J11	46	/	"	4,6	"	I/O	AD14
			/	"	-	,14	I/O/Z	
P4.7	K10	45	/	"	4,7	"	I/O	AD15
			/	"	-	,15	I/O/Z	
HSI.0	J1	20	"			0"	I	
HSI.1	J2	21	"			1"	I	
HSI.2	K1	22	"			2"	I	HSO.4
			"			4"	O	
HSI.3	L2	24	"			3"	I	HSO.5
			"			5"	O	
HSO.0	K2	25	"			0"	O	
HSO.1	L3	26	"			1"	O	
HSO.2	K5	31	"			2"	O	
HSO.3	L6	34	"			3"	O	
EA#	B6	79	"			"	I	
BW	B9	71	"			"	I	

Окончание таблицы 1.3.3

1	2	3	4	5	6
READY	L10	43	" "	I	
NMI	A5	80	" "	I	
RESET#	E1	9	/ " "	I/O	
INST	A9	70	" "	O	
ALE	B10	69	" "	O	ADV#
WR#	K8	40	" "	O	WRL#
BHE#	L9	41	" "	O	WRH#
RD#	A10	68	" "	O	
CLCOUT	A8	72	" "	O	
XTAL1	A7	74		- I	
XTAL2	B8	73		-	
U _{PP}	L7	37	" "	I	
CDE	D1	6		-	
U _{CC1} (U _{CC})	A6	8, 33, 60, 77, 78	5	-	
U _{CC2} (U _{REF})	C2	5	5	-	
U _{SS} (GND)	B7, K6	12, 32, 35, 36, 53, 55, 57, 64, 75, 76		-	
U _{SS} (ANGND)	C1	3, 4		-	
NC		13, 14, 23, 54, 81, 88		-	
- I - , « »:					
- O - ,					
- I/O - / ,					
- Z - .					

16- CPU

232

, 8- FIFO

8-

16-

2.1.1.

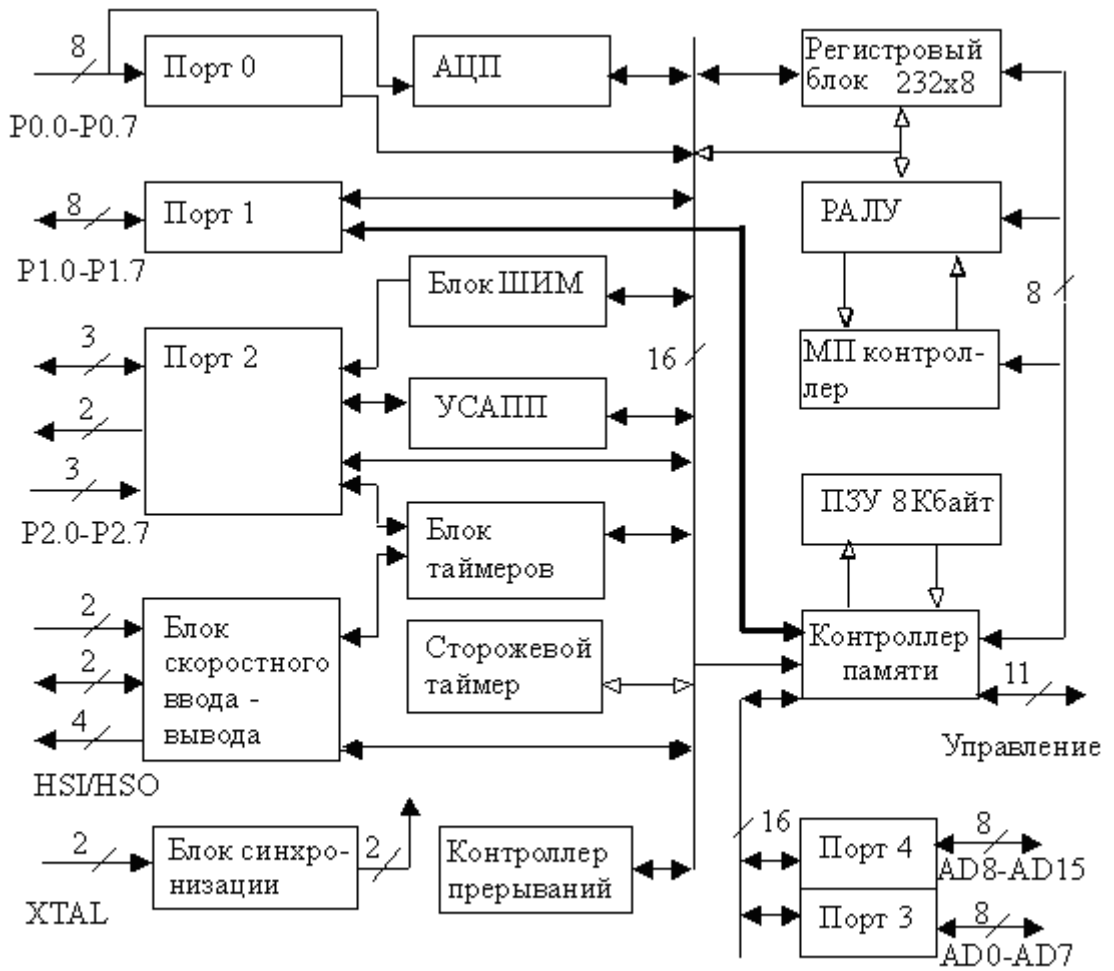


Рисунок 2.1.1 – Блок-схема микроконтроллеров 2.2 (CPU)

U

()

()

256-

()

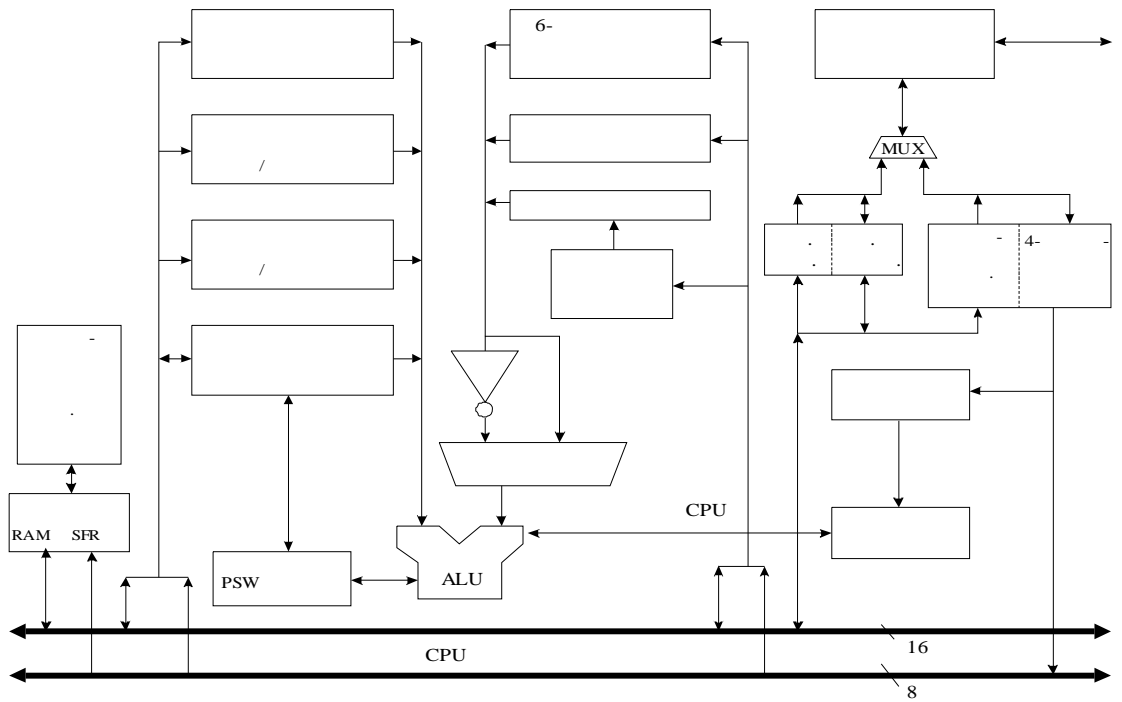
- : ;
 - " " - ;
 - .
 16- U 8- , -
 CPU PU, 16- -
2.3

().
 (Slave PC).
 (,)
 ; ,
 JUMP (, CALL ())
 ,
 .

16- .
 4 CLKOUT.
 JUMP () .
 , , ,

2.4 CPU
 CPU
 CPU 256-
 (,),
 2.4.1 , ,

.
.
.
.
.
.



2.4.1 – -

2.5 ()

2.4.1, 17- , -
 (PSW), 16- 17- (16- +). -
 , " " " " " -
 32- 6- " " -
 , 0, 1 2, " " -
 ().

2.6 _____

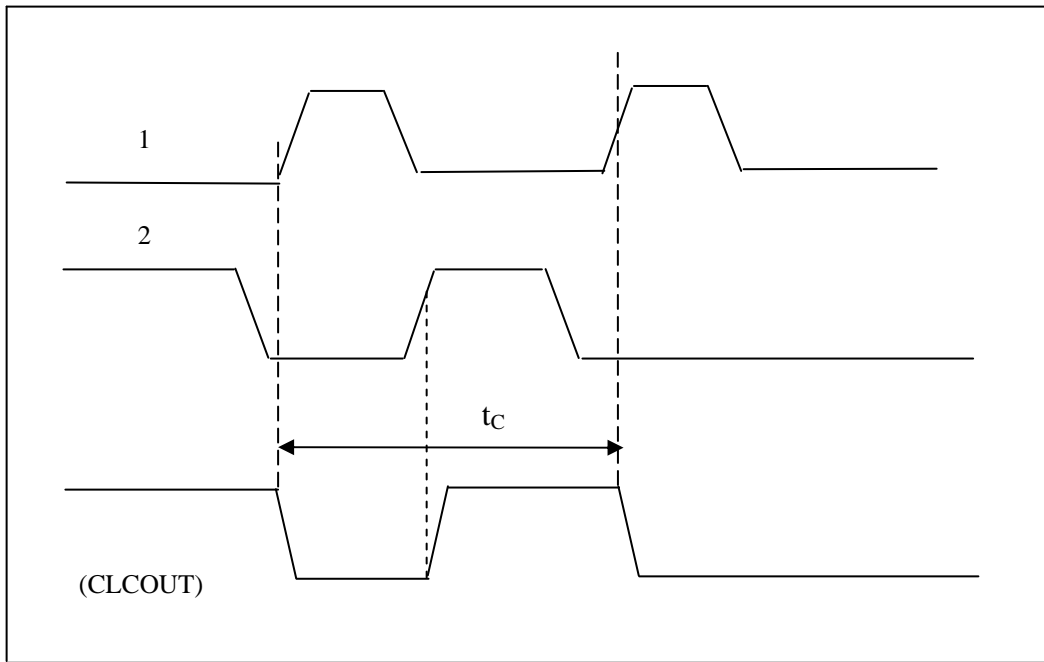
XTAL1. XTAL1 XTAL2, , -
 , 2. "State time", -
 time" () 100 . 20 "State

•
•
•
•
•
•
•
•

					.431295.008	15
--	--	--	--	--	-------------	----

time",

2.6.1,
 $f_c = 20$, $t_c = 100$.



2.6.1 -

2.7

64

0000

00FFH

1FFEH

2080

2.7.1.

	0FFFFH
	4000H
	2080H
8	2040H
()	2030H
	2020H
	2019H
	2018H
8 + 2	2014H
3 4	2000H
	1FFEH
()	0100H
	0000H

2.7.1 -

2.7.1

0000 00FFH 256
 ()
 000 0FFH,
 18 0FFH
 232 (8), (16)
 (32).
 232 " ".
 (Power down mode).
 18 19
 64-

2.7.2

00 17
 (3 4)
 2.7.2.1, ()
 2.7.2.2,
 15 - 2.7.2.3.

.
.
.
.
.
.

.431295.008	17
---	---	---	---	---	-------------	----

	16H		16H
WSR		WSR	
INT MASK1/PEND1	14H	INT MASK1/PEND1	14H
	10H		10H
	0EH		0EH
TIMER2		T2 CAPTURE	
	0CH		0CH
	ACH		ACH
INT MASK/PEND		INT MASK/PEND	
	08H		08H
	06H		06H
	04H		04H
	02H		02H
ZERO REG		ZERO REG	
	00H		00H

/
WSR=0

WSR=14

/
WSR=15

2.7.2.1 -

14

PUSHA POPA

(WSR)
WSR, -

WSR - 0, 14 15.

WSR = 0.

2.7.2.2,

() 24 ,

0

,

14. -

15 (WSR = 15)

0

15

0,

15

2,

2,

0

15,

2.7.2.3.

.431295.008

18

2.7.2.1 -

(SP)	19H	(SP)
IOS2	18H	PWM_CONTROL
IOS1	17H	IOC1
IOS0	16H	IOC0
WSR	15H	WSR
INT_MASK1	14H	INT_MASK1
INT_PEND1	13H	INT_PEND1
SP_STAT	12H	SP_CON
PORT2	11H	PORT2
PORT1	10H	PORT1
PORT0	0FH	BAUD_RATE
TIMER2 (HI)	0EH	TIMER2 (HI)
TIMER2 (LO)	0DH	TIMER2 (LO)
TIMER1 (HI)	0CH	IOC2
TIMER1 (LO)	0BH	WATCHDOG
INT_PEND	0AH	INT_PEND
INT_MASK	09H	INT_MASK
SBUF(RX)	08H	SBUF (TX)
HSI_STATUS	07H	HSEO_COMMAND
HSI_TIME (HI)	06H	HS _TIME (HI)
HSI_TIME (LO)	05H	HS _TIME (LO)
AD_RESULT (HI)	04H	HSI_MODE
AD_RESULT (LO)	03H	AD_COMMAND
ZER0 REG (HI)	02H	ZER0 REG (HI)
ZER0 REG (LO)	01H	ZER0 REG (LO)
	00H	

WSR=0

*
*
*
T2CAPTURE (HI)
T2CAPTURE (LO)
WSR = 15

SFR WSR = 15
,
WSR = 0
WSR = 0 ,

PPW
WSR = 14

.	.
.	.
.	.
.	.
.	.
.	.
.	.
.	.
.	.
.	.

.	.	.	.
.	.	.	.
.	.	.	.

" 0 , -
 . , , (,) -
 - 0, -
 , -
 -
 , , , -
 , -
 , -

2.7.3

1FFEH 1FFFH 3 4, , 3 4
 ,
 2000 2080 . 18

2.7.3.1

,
 , 5, 16.
 " , 2000 207FH, " -
 ,
 . 2019 FFH 20 -
) (-
) -
 " " 2080 . -
 8 , -

.
.
.
.
.
.

.431295.008	22
---	---	---	---	---	-------------	----

/	FFFFH
	4000H
	2080H
	2074H-207FH
	2072H-2073H
	2070H-2071H
	2040H-206FH
	2030H-203FH
	2020H-202FH
	2019H-201FH
	2018H
	2015H-2017H
PPW (2014H
EPROM)	2000H-2013H

2.7.3.1 –

•
•
•
•
•
•

•	•	•	•	•	.431295.008
---	---	---	---	---	-------------

2.7.4

2080 3FFFH,
(R),
2000' 207F . 2014 -
(PPW). -
() #
2000 3FFFH.

RESET#.

2.7.5

16- / , 8- : -
8- 16- . , -
HOLD#/HLDA# READY
RD# WR#,

3

8- WORD INTEGER. BYTE SHORT-INTEGERS 16-
DOUBLE WORD LONG-INTEGERS
16x16, 32 16,
32-
16-
AX, BX, CX, DX - 16- ;
AL - ;
- ;
BL - ;
CL - ;
DL - DX.

.	
.	
.	
.	
.	
.	

.							.431295.008	24
.								

3.1 _____

PL/M-96.

: "BYTE" -

8-

; "byte" -

(BYTES)

BYTES -

8-

0

255.

BYTES,

256.

BYTES

BYTE-

0

7, 0 -

BYTE ;

(WORDS)

WORDS -

16-

0...65535.

WORD,

65536.

0 15,

0 -

()

(SHORT-INTEGERS)

SHORT-INTEGERS - 8-

-128...+127.

(PSW).

BYTE.

SHORT-INTEGERS

(INTEGERS)

INTEGERS - 16-

-32768...+32767.

PSW.

WORD.

INTEGERS

WORDS.

(BITS)

BITS -

" " WORD,

BYTE

BYTES

WORDS,

1830BE51/31.

(DOUBLE-WORDS)

DOUBLE-WORDS – 32- ,
 0...4294967295. -
 16- , 16- ,
 DOUBLE-WORD
 , 4. DOUBLE-WORD
 DOUBLE-WORD,
 WORD.
 DOUBLE-WORDS.

(LONG-INTEGERS)

LONG-INTEGERS – 32- -2147483648...+2147483647.
 16- , 16- ,
 LONG-INTEGERS
 LONG-INTEGERS
 LONG-INTEGER , 4.
 LONG-INTEGERS,
 INTEGER.

LONG

3.2

ZERO [0]

[SP].

256-

8-

:
 ADD AX, BX, CX ; AX:=BX+CX
 MUL AX, BX ; AX:=AX*BX
 INCB CL ; CL:=CL+1

WORD

8-

```

LD      AX, [AX] ;      AX:=MEM_WORD (AX)
ADDB   AL, BL, [CX] ;    AL:=BL+MEM_BYTE (CX)
POP    [AX] ;      MEM_WORD (AX):=MEM_WORD(SP); SP:=SP+2

```

WORD,

BYTES SHORT-INTEGERS, WORDS INTEGERS 1;

2.

```

LD      AX, [BX]+ ;      AX:=MEM_WORD (BX); BX:=BX+2
ADDB   AL, BL, [CX]+ ;    AL:=BL+MEM_BYTE (CX); CX:=CX+1
PUSH   [AX]+ ;          SP:=SP-2
;      MEM_WORD (SP):=MEM_WORD (AX)
;      AX:=AX+2

```

BYTE SHORT-INTEGER 8 ;

WORD INTEGER 16 .

```

ADD    AX, #340 ;      AX:=AX+340
PUSH   #1234H ;      SP:=SP-2; MEM_WORD (SP)=1234H
DIVB   AX, #10 ;      AL:=AX/10; AH:=AX MOD 10

```

8-

WORD , 8-

WORD

WORD. -128...+127

```

LD      AX, 12[BX] ;      AX:=MEM_WORD (BX+12)
MULB   AX, BL, 3[CX];    AX:=BL*MEM_BYTE (CX+3)

```

16-

```

AND  AX, BX, TABLE[CX] ; AX:=BX.AND.MEM_WORD(TABLE+CX)
ST   AX, TABLE[BX]    ; MEM_WORD(TABLE+BX)=:AX
ADDB AL, BL, LOOKUP[CX] ; AL:=BL+MEM_BYTE(LOOKUP+CX)

```

ZERO

0.

WORD

```

ADD  AX, 1234[0] ; AX:=AX+MEM_WORD(1234)
POP  5678[0]    ; MEM_WORD(5678):=MEM_WORD(SP)
      ; SP:=SP+2

```

(SP)

18

SP,

SP

WORD

```

PUSH [SP] ; DUPLICATE TOP_OF STACK
LD   AX, 2[SP] ; AX:=NEXT_TO_TOP

```

3.3

(PSW) -

PSW

3.3.1

PSW. PSW

PSW

PUSHF.

7	6	5	4	3	2	1	0
Z	N	V	VT	C	X	I	ST

3.3.1 - PSW

Z: Z (ZERO)

ADDC SUBC Z

ADD SUB

Z

N: N (Negative)

N

SHL, SHR, SHRA, N-

V: V (oVerflow)
()

SHL, SHLB SHLL V

V:

	:	V	
UNSIGNED BYTE DIVIDE	>	255 (0FFH)	
UNSIGNED WORD DIVIDE	>	65535 (0FFFFH)	
SIGNED BYTE DIVIDE	<	- 127 (81H)	> 127 (7F)
SIGNED WORD DIVIDE	<	- 32767	> 32767 (7FFFH)

VT: VT (oVerflow Trap)

V,

CLRVT JVT JNVT.

VT

V

C: C (Carry)

" "

-

C (. .

, C 0).

X:

PSW

I:

NMI, TRAP

ST: ST (STicky bit)

C,

(" ").

ST

ST

C

8x8

12 :

MULUB

AX,CL,DL ;

AX:=CL*DL

SHR

AX,#4 ;

4

C

"

"

1/2

(LSB).

C

1/2 LSB.

ST

1,

C

).

ST

ST

	ST	
0	0	Value =0
0	1	0 < Value < 1/2 LSB
1	0	Value = 1/2 LSB
1	1	Value > 1/2 LSB

256-

/ ZERO.

230

32-

16-

(,)

4

SHORT-INTEGER)

(LONG-INTEGER, DOUBLE-WORD, REAL)

(BYTE

. 32-

16-

;

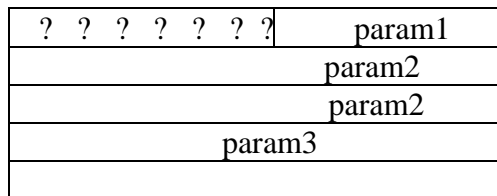
PL/M-96:

example _ procedure: PROCEDURE

(param1, param2, param3);

DECLARE param1 BYTE,
param2 DWORD,
param3 WORD

:



stack_pointer

8-, 16- 32-

PLMREG. PLMREG

PL/M-96,

1
PLMREG

8

2

8

3

PLMREG,

(Z, N, V, VT, C, ST)

.431295.008

31

4
PL/M-96

, PLMREG.
INTERRUPT,

– interrupt),

(
PSW PLMREG

3.5

() RST

(RST) OFFH,

(RESET).

NOP ()

JAMP,

RST.

(WDT) –
WDT

()

WDT.

WDT,

WDT

10

3.5.1
3.5.2

(),

.
.
.
.
.
.

					.431295.008	
						32

3.5.1 –

		(I)	(II)						
			Z	N	C	V	VT	ST	
1	2	3	4	5	6	7	8	9	10
ADD/ADDB	2	$D \leftarrow D + A$					↑	-	
ADD/ADDB	3	$D \leftarrow B + A$					↑	-	
ADDC/ADDCB	2	$D \leftarrow D + A + C$	↓				↑	-	
SUB/SUBB	2	$D \leftarrow D - A$					↑	-	
SUB/SUBB	3	$D \leftarrow B - A$					↑	-	
SUBC/SUBCB	2	$D \leftarrow D - A + C - 1$	↓				↑	-	
CMP/CMPB	2	$D - A$					↑	-	
MUL/MULU	2	$D, D + 2 \leftarrow D \ A$	-	-	-	-	-	-	2
MUL/MULU	3	$D, D + 2 \leftarrow B \ A$	-	-	-	-	-	-	2
MULB/MULUB	2	$D, D + 1 \leftarrow D \ A$	-	-	-	-	-	-	3
MULB/MULUB	3	$D, D + 1 \leftarrow B \ A$	-	-	-	-	-	-	3
DIVU	2	$D \leftarrow (D, D + 2) / A, D + 2 \leftarrow$	-	-	-		↑	-	2
DIVUB	2	$D \leftarrow (D, D + 1) / A, D + 1 \leftarrow$	-	-	-		↑	-	3
DIV	2	$D \leftarrow (D, D + 2) / A, D + 2 \leftarrow$	-	-	-		↑	-	
DIVB	2	$D \leftarrow (D, D + 1) / A, D + 1 \leftarrow$	-	-	-		↑	-	
AND/ANDB	2	$D \leftarrow D \text{ AND } A$			0	0	-	-	
AND/ANDB	3	$D \leftarrow B \text{ AND } A$			0	0	-	-	
OR/ORB	2	$D \leftarrow D \text{ OR } A$			0	0	-	-	
XOR/XORB	2	$D \leftarrow D \text{ (excl. or) } A$			0	0	-	-	
LD/LDB	2	$D \leftarrow A$	-	-	-	-	-	-	
ST/STB	2	$A \leftarrow D$	-	-	-	-	-	-	
LDBSE	2	$D \leftarrow A; D + 1 \leftarrow \text{SIGN}(A)$	-	-	-	-	-	-	3,4
LDBZE	2	$D \leftarrow A; D + 1 \leftarrow 0$	-	-	-	-	-	-	3,4
PUSH	1	$SP \leftarrow SP - 2; (SP) \leftarrow A$	-	-	-	-	-	-	
POP	1	$A \leftarrow (SP); SP + 2$	-	-	-	-	-	-	
PUSHF	0	$SP \leftarrow SP - 2; (SP) \leftarrow \text{PSW};$ $\text{PSW} \leftarrow 0000\text{H}; I \leftarrow 0$	0	0	0	0	0	0	
POPF	0	$\text{PSW} \leftarrow (SP); SP \leftarrow SP + 2; I \leftarrow$							
SJMP	1	$PC \leftarrow PC + 11-$	-	-	-	-	-	-	5
LJMP	1	$PC \leftarrow PC + 16-$	-	-	-	-	-	-	5
BR ()	1	$PC \leftarrow (A)$	-	-	-	-	-	-	
SCALL	1	$SP \leftarrow SP - 2;$ $(SP) \leftarrow PC; PC \leftarrow PC + 11-$	-	-	-	-	-	-	5
LCALL	1	$SP \leftarrow SP - 2; (SP) \leftarrow PC;$ $PC \leftarrow PC + 16-$	-	-	-	-	-	-	5
RET	0	$PC \leftarrow (SP); SP \leftarrow SP + 2$	-	-	-	-	-	-	
J ()	1	$PC \leftarrow PC + 8- (-)$	-	-	-	-	-	-	5
JC	1	$C = 1$	-	-	-	-	-	-	5
JNC	1	$C = 0$	-	-	-	-	-	-	5
JE	1	$Z = 1$	-	-	-	-	-	-	5
JNE	1	$Z = 0$	-	-	-	-	-	-	5
JGE	1	$N = 0$	-	-	-	-	-	-	5
JLT	1	$N = 1$	-	-	-	-	-	-	5
JGT	1	$N = 0 \ Z = 0$	-	-	-	-	-	-	5
JLE	1	$N = 1 \ Z = 1$	-	-	-	-	-	-	5

3.5.1

1	2	3	4	5	6	7	8	9	10
JH	1	=1 Z=0	-	-	-	-	-	-	5
JNH	1	=0 Z=1	-	-	-	-	-	-	5
JV	1	V=1	-	-	-	-	-	-	5
JNV	1	V=0	-	-	-	-	-	-	5
JVT	1	VT=1; VT	-	-	-	-	0	-	5
JNVT	1	VT=0; VT	-	-	-	-	0	-	5
JST	1	ST=1	-	-	-	-	-	-	5
JNST	1	ST=0	-	-	-	-	-	-	5
JBS	3	. = 1	-	-	-	-	-	-	5,6
JB	3	. = 0	-	-	-	-	-	-	5,6
DJNZ/ DJNZW	1	D ← D - 1 D 0 PC←PC+8	-	-	-	-	-	-	5 10
DEC/DECB	1	D ← D - 1					↑	-	
NEG/NEGB	1	D ← 0 - D					↑	-	
INC/INCB	1	D ← D + 1					↑	-	
EXT	1	D ← D; D + 2 ← Sign(D)			0	0	-	-	2
EXTB	1	D ← D; D + 1 ← Sign(D)			0	0	-	-	3
NOT/NOTB	1	D ← Logical Not (D)			0	0	-	-	
CLR/CLRB	1	D ← 0	1	0	0	0	-	-	
SHL/SHLB/ SHLL	2	C ← ← 0 — —					↑	-	7
SHR/SHRB/ SHRL	2	0 → → C — —				0	-	-	7
SHRA/SHRAB/ SHRAL	2	. → → — —				0	-	-	7
SETC	0	C ← 1	-	-	1	-	-	-	
CLRC	0	C ← 0	-	-	0	-	-	-	
CLRVT	0	VT ← 0	-	-	-	-	0	-	
RST	0	PC ← 2080H	0	0	0	0	0	0	8
DI	0	(I←0)	-	-	-	-	-	-	
EI	0	(I←1)	-	-	-	-	-	-	
NOP	0	PC ← PC + 1	-	-	-	-	-	-	
SKIP	0	PC ← PC + 2	-	-	-	-	-	-	
NORML	2	. - =1; D ←			0	-	-	-	7
TRAP	0	SP ← SP-2; (SP) ← PC; PC ← (2010H)	-	-	-	-	-	-	9
PUSHA	1	SP ← SP-2; (SP) ← PSW; PSW ← 0000H; SP ← SP-2; (SP)←IMASK1/WSR; IMASK1←00H	0	0	0	0	0	0	
POPA	1	IMASK1/WSR←(SP); SP←SP+2 PSW←(SP); SP←SP + 2							
IDLPD	1	IDLE = 1; = 2; (Power down) ;	-	-	-	-	-	-	
CMPL	2	D-A					↑	-	
BMOV	2	[PTR_HI]+ ← [PTR_LOW]+; = 0.	-	-	-	-	-	-	

I 3.5.1

1 " " _ , -
 . D, B A D B -
 .
 2 D, D + 2 - ; D -
 3 D, D + 1 - ; D -
 4 .
 5 .
 6 1 2048 .
 7 "L" () .
 8 RESET#. -
 2080 .
 9 .
 10 DJNZW - .

II 3.5.1 (,

)
 1 " " ,
 .
 2 "_ " ,
 3 "1" "0" ,
 4 "↑" , (-
),
 "↓".
 5 "?" , () .

.
.
.
.
.
.
.

.
.
.

		-			(1)	(1)
			(1)	(1)		
ADD(3-OP)	4/44	5/45	4/46	4/46	5/47	6/47
SUB(3-OP)	4/48	5/49	4/4A	4/4A	5/4B	6/4B
ADD(2-OP)	3/64	4/65	3/66	3/66	4/67	5/67
SUB(2-OP)	3/68	4/69	3/6A	3/6A	4/6B	5/6B
ADDC	3/A4	4/A5	3/A6	3/A6	4/A7	5/A7
SUBC	3/A8	4/A9	3/AA	3/AA	4/AB	5/AB
CMP	3/88	4/89	3/AB	3/AB	4/8B	5/8B
ADDB(3-OP)	4/54	4/55	4/56	4/56	5/57	6/57
SUBB(3-OP)	4/58	4/59	4/5A	4/5A	5/5B	6/5B
ADDB(2-OP)	3/74	3/75	3/76	3/76	4/77	5/77
SUBB(2-OP)	3/78	3/79	3/7A	3/7A	4/7B	5/7B
ADDCB	3/B4	3/B5	3/B6	3/B6	4/B7	5/B7
SUBCB	3/B8	3/B9	3/BA	3/BA	4/BB	5/BB
CMPB	3/98	3/99	3/9A	3/9A	4/9B	5/9B
MUL(3-OP)	5/ ⁽²⁾	6/ ⁽²⁾	5/ ⁽²⁾	5/ ⁽²⁾	6/ ⁽²⁾	7/ ⁽²⁾
MULU(3-OP)	4/4C	5/4D	4/4E	4/4E	5/4F	6/4F
MUL(2-OP)	4/ ⁽²⁾	5/ ⁽²⁾	4/ ⁽²⁾	4/ ⁽²⁾	5/ ⁽²⁾	6/ ⁽²⁾
MULU(2-OP)	3/6C	4/6D	3/6E	3/6E	4/6F	5/6F
DIV	4/ ⁽²⁾	5/ ⁽²⁾	4/ ⁽²⁾	4/ ⁽²⁾	5/ ⁽²⁾	6/ ⁽²⁾
DIVU	3/8C	4/8D	3/8E	3/8E	4/8F	5/8F
MULB(3-OP)	5/ ⁽²⁾	5/ ⁽²⁾	5/ ⁽²⁾	5/ ⁽²⁾	6/ ⁽²⁾	7/ ⁽²⁾
MULUB(3-OP)	4/5C	4/5D	4/5E	4/5E	5/5F	6/5F
MULB(2-OP)	4/ ⁽²⁾	4/ ⁽²⁾	4/ ⁽²⁾	4/ ⁽²⁾	5/ ⁽²⁾	6/ ⁽²⁾
MULUB(2-OP)	3/7C	3/7D	3/7E	3/7E	4/7F	5/7F
DIV	4/ ⁽²⁾	4/ ⁽²⁾	4/ ⁽²⁾	4/ ⁽²⁾	5/ ⁽²⁾	6/ ⁽²⁾
DIVUB	3/9C	3/9D	3/9E	3/9E	4/9F	5/9F
AND(3-OP)	4/40	5/41	4/42	4/42	5/43	6/43
AND(2-OP)	3/60	4/61	3/62	3/62	4/63	5/63
OR(2-OP)	3/80	4/81	3/82	3/82	4/83	5/83
XOR	3/84	4/85	3/86	3/86	4/87	5/87
ANDB(3-OP)	4/50	4/51	4/52	4/52	5/53	5/53
ANDB(2-OP)	3/70	3/71	3/72	3/72	4/73	4/73
ORB(2-OP)	3/90	3/91	3/92	3/92	4/93	5/93
XORB	3/94	3/95	3/96	3/96	4/97	5/97
PUSH	2/C8	3/C9	2/CA	2/CA	3/CB	4/CB
POP	2/CC	-	2/CE	2/CE	3/CF	4/CF
LD	3/A0	4/A1	3/A2	3/A2	4/A3	5/A3
LDB	3/B0	3/B1	3/B2	3/B2	4/B3	5/B3
ST	3/C0	-	3/C2	3/C2	4/C3	5/C3
STB	3/C4	-	3/C6	3/C6	4/C7	5/C7
LDBSE	3/BC	3/BD	3/BE	3/BE	4/BF	5/BF
LBSZE	3/AC	3/AD	3/AE	3/AE	4/AF	5/AF

•
•
•
•
•
•
•

3.5.2 -

()/

	/
PUSHF	1/F2
POPF	1/F3
PUSHA	1/F4
POPA	1/F5
TRAP	1/F7
LCALL	3/EF
SCALL	2/28-2F ⁽³⁾
RET	1/F0
LJMP	3/E7
SJMP	2/20-27 ⁽³⁾
BR[]	2/E3
JNST	1/D0
JST	1/D8
JNH	1/D1
JH	1/D9
JGT	1/D2
JLE	1/DA
JNC	1/B3
JC	1/D8
JNVT	1/D4
JVT	1/DC
JNV	1/D5
JV	1/DD
JGE	1/D6
JLT	1/DE
JNE	1/D7

	/
JE	1/DF
JBC	3/30-37
JBS	3/38-3F
DJNZ	3/EO
DJNZW	3/E1 ⁽⁴⁾
NORML	3/OF
SHRL	3/OC
SHLL	3/OD
SHRAL	3/OE
SHR	3/O8
SHRB	3/18
SHL	3/O9
SHLB	3/19
SHRA	3/OA
SHRAB	3/1A
CLRC	1/F8
S TC	1/F9
DI	1/FA
EI	1/FB
CLRVT	1/FC
NOP	1/FD
RST	1/FF
SKIP	2/00
IDLPD	1/F6
BMOV	3/C1

3.5.2.

1

2

3 3

4

DJNZW

;

"FE".

8 11-

		-	*	- *	* -	- *
ADD(3-OP)	5	6	7/10	8/11	7/10	8/11
SUB(3-OP)	5	6	7/10	8/11	7/10	8/11
ADD(2-OP)	4	5	6/8	7/9	6/8	7/9
SUB(2-OP)	4	5	6/8	7/9	6/8	7/9
ADDC	4	5	6/8	7/9	6/8	7/9
SUBC	4	5	6/8	7/9	6/8	7/9
CMP	4	5	6/8	7/9	6/8	7/9
ADDB(3-OP)	5	5	7/10	8/11	7/10	8/11
SUBB(3-OP)	5	5	7/10	8/11	7/10	8/11
ADDB(2-OP)	4	4	6/8	7/9	6/8	7/9
SUBB(2-OP)	4	4	6/8	7/9	6/8	7/9
ADDCB	4	4	6/8	7/9	6/8	7/9
SUBCB	4	4	6/8	7/9	6/8	7/9
CMPB	4	4	6/8	7/9	6/8	7/9
MUL(3-OP)	16	17	18/21	19/22	19/22	20/23
MULU(3-OP)	14	15	16/19	17/19	17/20	18/21
MUL(2-OP)	16	17	18/21	19/22	19/22	20/23
MULU(2-OP)	14	15	16/19	17/19	17/20	18/21
DIV	26	27	28/31	29/32	29/32	30/33
DIVU	24	25	26/29	27/30	27/30	28/31
MULB(3-OP)	12	12	14/17	13/15	15/18	16/19
MULUB(3-OP)	10	10	12/15	12/16	12/16	14/17
MULB(2-OP)	12	12	14/17	15/18	15/18	16/19
MULUB(2-OP)	10	10	12/15	13/15	12/16	14/17
DIV	18	18	20/23	21/24	21/24	22/25
DIVUB	16	16	18/21	19/22	19/22	20/23
AND(3-OP)	5	6	7/10	8/11	7/10	8/11
AND(2-OP)	4	5	6/8	7/9	6/8	7/9
OR(2-OP)	4	5	6/8	7/9	6/8	7/9
XOR	4	5	6/8	7/9	6/8	7/9
ANDB(3-OP)	5	5	7/10	8/11	7/10	8/11
ANDB(2-OP)	4	4	6/8	7/9	6/8	7/9
ORB(2-OP)	4	4	6/8	7/9	6/8	7/9
XORB	4	4	6/8	7/9	6/8	7/9
PUSH (.)	6	7	9/12	10/13	10/13	11/14
POP (.)	8	-	10/12	11/13	11/13	12/14
PUSH (.)	8	9	11/14	12/15	12/15	13/16
POP (.)	11	-	13/15	14/16	14/16	15/17
LD,LDB	4,4	5,4	5/8	6/8	6/9	7/10
ST,STB	4,4	-	5/8	6/8	6/9	7/10
LDBSE	4	4	5/8	6/8	6/9	7/10
LBSZE	4	4	5/8	6/8	6/9	7/10
BMOV		/ :6+8				
		/ :6+11				
		/ :6+14				

* : (0-1FFH) / (200H-0FFFFH).

200 -1FFH

200H-0FFFFH.

3.5.4 -

PUSHF(.)	6	PUSHF(.)	8
POPF (.)	7	POPF (.)	10
PUSHA(.)	12	PUSHA(.)	18
POPA (.)	12	POPA (.)	18
TRAP (.)	16	TRAP (.)	18
LCALL(.)	11	LCALL(.)	13
SCALL(.)	11	SCALL(.)	13
RET (.)	11	RET (.)	14
CMPL	7	DEC/DECB	3
CLR/CLRB	3	EXT/EXTB	4
NOT/NOTB	3	NC/INCB	3
NEG/NEGB	3		
LJMP	7		
SJMP	7		
BR ()	7		
JNST,JST	7		
JNH,JH	4/8	/	
JGT,JLE	4/8	/	
JNC,JC	4/8	/	
JNVT,JVT	4/8	/	
JNV,JV	4/8	/	
JGE,HLT	4/8	/	
JNE,JE	4/8	/	
JBC,JBS	5/9	/	
DJNZ	5/9	/	
DJNZW ⁽¹⁾	5/9	/	
NORML	8+1	(9 0)	
SHRL	7+1	(8 0)	
SHLL	7+1	(8 0)	
SHRAL	7+1	(8 0)	
SHR/SHRB	6+1	(7 0)	
SHL/SHLB	6+1	(7 0)	
SHRA/SHRAB	6+1	(7 0)	
CLRC	2		
SETC	2		
DI	2		
EI	2		
CLRVT	2		
NOP	2		
RST	15 ()		
SKIP	3		
IDLPD	8/25(/)		

- DJNZW .

4

(PWM), 1 2, : -
 (HSIO), -
 HSIO, -
 HSIO 16-
 (HSI) 1), 16- (2), (HSO).

CPU HSIO

HSIO 18 /

4.1 (PWM)

(D/A)
 (PWM). -
 256 512
 2. PWM

4.2

16-
 1, - 2. 1 2
 (HSI) (HSO)

1 -
 T2CLK HSI.1. 2
 HSO.
 (2.6)
 2, - 2 0FFFFH/0000H 7FFFH/8000H
 ().

4.3 (HSI)

(HSI) (HSI.0 – HSI.3).
 1, :

HSI

4.3.1.

HSI_TIME

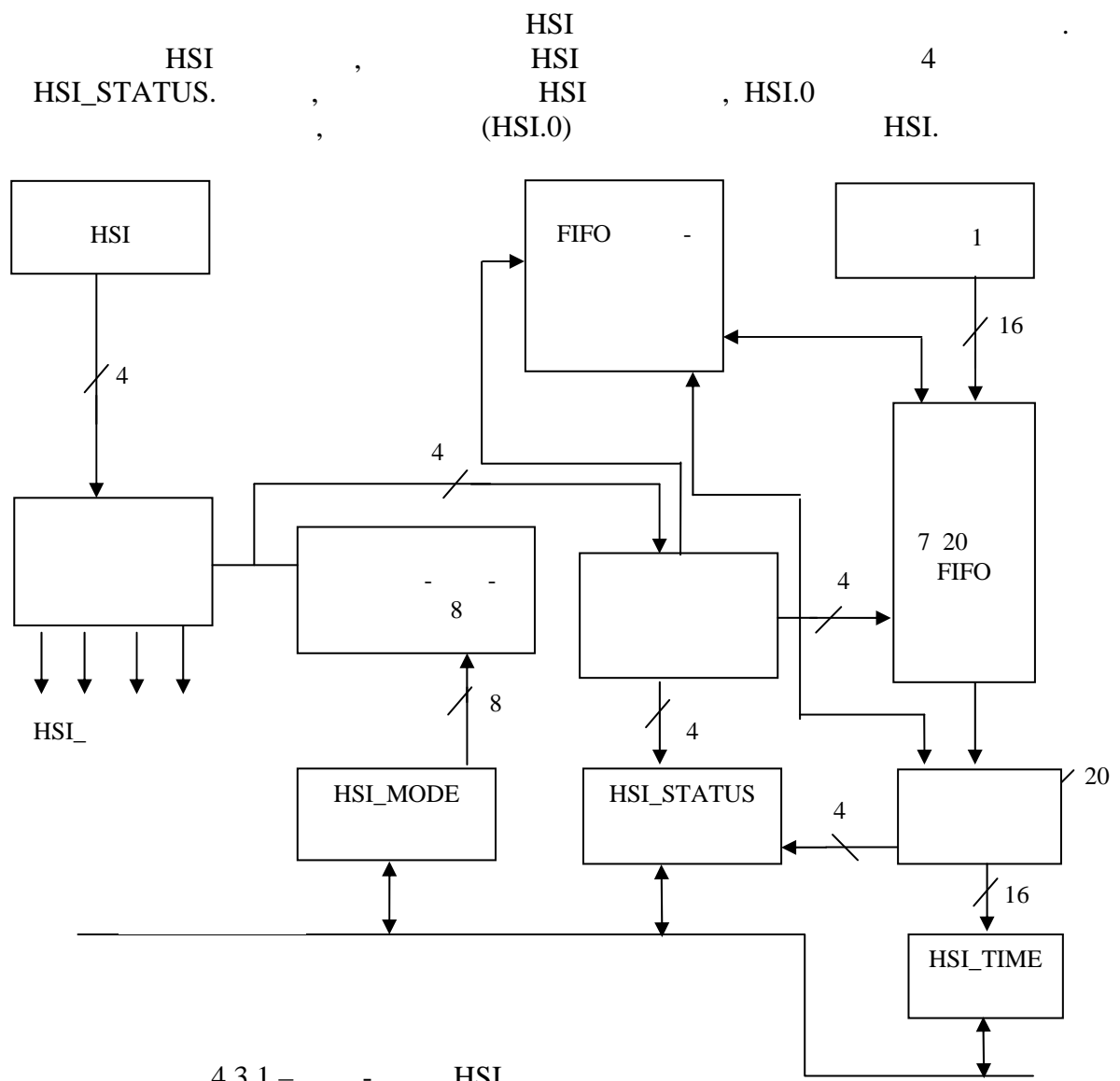
FIFO

HSI-

FIFO

FIFO (

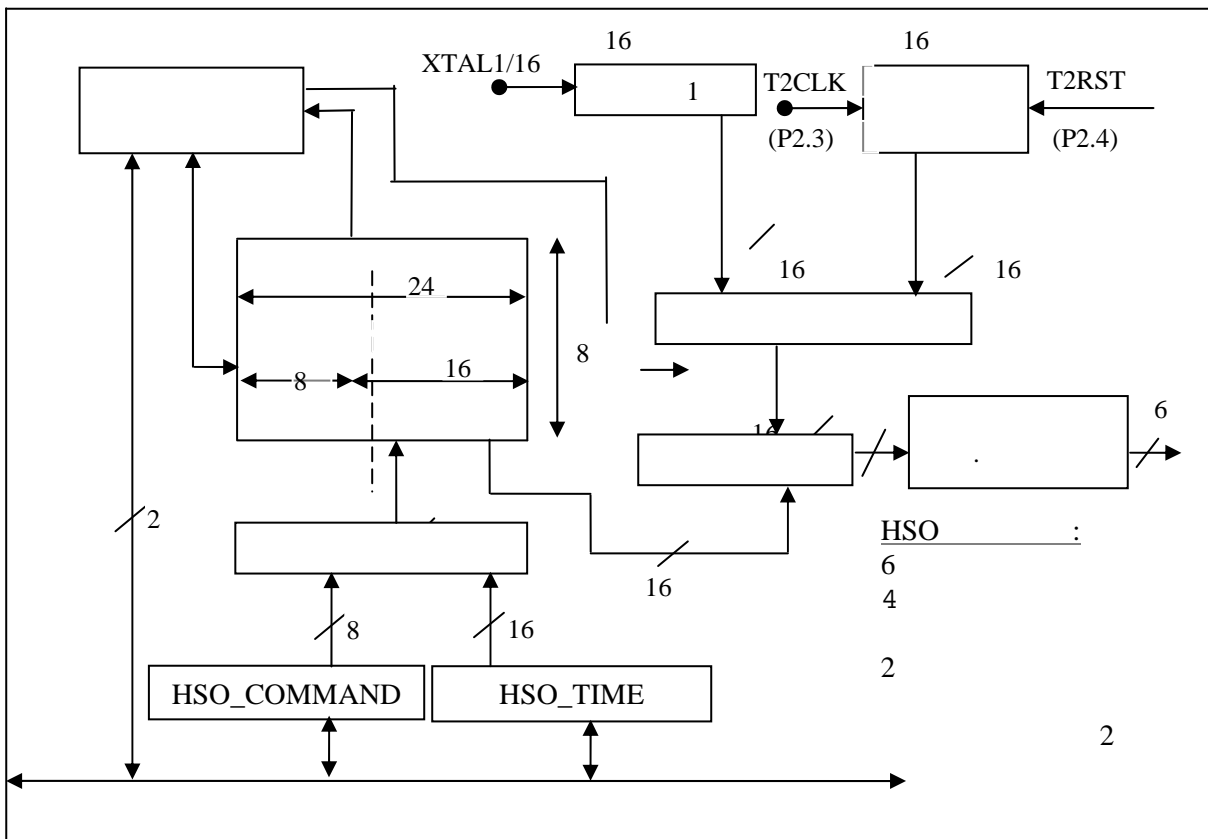
)



4.3.1 - HSI

4.4 (HSO)

HSO, HSO_TIME 14
 HSO CAM HSO HSO_COMMAND
 HSO: 8 6
 HSO: -
 HSO.0 - HSO.5.
 HSO



4.4.1 - - HSO

4.5

1830.

(SP_STAT)

1 0 (Mode 0)
0 -

RXD

TXD

8
RXD
2 1 (Mode 1)

(0), 8 (SPCON.2), (-), (1).

3 2 (Mode 2)
2 -

3

9-

(1) (0), 9 (1, -), 8
SBUF(). 8
SBUF() , -
(RI) 4 3 (Mode 3) 9-
2. (PEN = 1), (PEN = 0). 8
RB8. PEN = 1, PEN = 0, RB8
(RPE).
4.6
8- 10- 8 (ACH0 – ACH7),
0. - ;
A/D_Command 1.
HSO.

4.7 - 8-
(WDT)
WDT 64
WDT
RESET#
RESET#.

.
.
.
.
.
.