Welcome to E-XFL.COM

What is "Embedded - Microcontrollers"?

"Embedded - Microcontrollers" refer to small, integrated circuits designed to perform specific tasks within larger systems. These microcontrollers are essentially compact computers on a single chip, containing a processor core, memory, and programmable input/output peripherals. They are called "embedded" because they are embedded within electronic devices to control various functions, rather than serving as standalone computers. Microcontrollers are crucial in modern electronics, providing the intelligence and control needed for a wide range of applications.

Applications of "Embedded - Microcontrollers"

Details	
Product Status	Obsolete
Core Processor	AVR
Core Size	8-Bit
Speed	20MHz
Connectivity	I²C, SPI, UART/USART
Peripherals	Brown-out Detect/Reset, POR, PWM, WDT
Number of I/O	23
Program Memory Size	4KB (2K x 16)
Program Memory Type	FLASH
EEPROM Size	256 x 8
RAM Size	512 x 8
Voltage - Supply (Vcc/Vdd)	2.7V ~ 5.5V
Data Converters	A/D 8x10b
Oscillator Type	Internal
Operating Temperature	-40°C ~ 85°C (TA)
Mounting Type	Surface Mount
Package / Case	32-TQFP
Supplier Device Package	32-TQFP (7x7)
Purchase URL	https://www.e-xfl.com/product-detail/microchip-technology/atmega48-20ai

```
EEPROM_write:
  ; Wait for completion of previous write
  sbic EECR,EEPE
  rjmp EEPROM_write
  ; Set up address (r18:r17) in address register
  out  EEARH, r18
  out  EEARL, r17
  ; Write data (r16) to Data Register
  out  EEDR,r16
  ; Write logical one to EEMPE
  sbi  EECR,EEMPE
  ; Start eeprom write by setting EEPE
  sbi  EECR,EEPE
  ret
```

```
void EEPROM_write(unsigned int uiAddress, unsigned char ucData)
{
  /* Wait for completion of previous write */
  while(EECR & (1<<EEPE))
    ;
  /* Set up address and Data Registers */
  EEAR = uiAddress;
  EEDR = ucData;
  /* Write logical one to EEMPE */
  EECR |= (1<<EEMPE);
  /* Start eeprom write by setting EEPE */
  EECR |= (1<<EEPE);
}
```

Table 6-4.

Oscillator Source / Power Conditions	Start-up Time from Power-down and Power-save	Additional Delay from Reset (V_{CC} = 5.0V)	CKSEL0	SUT1..0
		1		
		1		
		1		

1.
2.

6.4 Full Swing Crystal Oscillator

Table 6-5.

Frequency Range[1] (MHz)	Recommended Range for Capacitors C1 and C2 (pF)	CKSEL3..1

1.
2.

[redacted]

1 [redacted]

2 [redacted]

[redacted]

6.11.1 Clock Prescale Register – CLKPR

Bit	7	6	5	4	3	2	1	0	
	CLKPCE	–	–	–	CLKPS3	CLKPS2	CLKPS1	CLKPS0	CLKPR
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	[redacted]				

- **Bit 7 – CLKPCE: Clock Prescaler Change Enable**

[redacted]

- **Bits 3..0 – CLKPS3..0: Clock Prescaler Select Bits 3 - 0**

[redacted]

$$R_{PCPWM} = \frac{\log(TOP+1)}{\log(2)}$$

Figure 13-8.

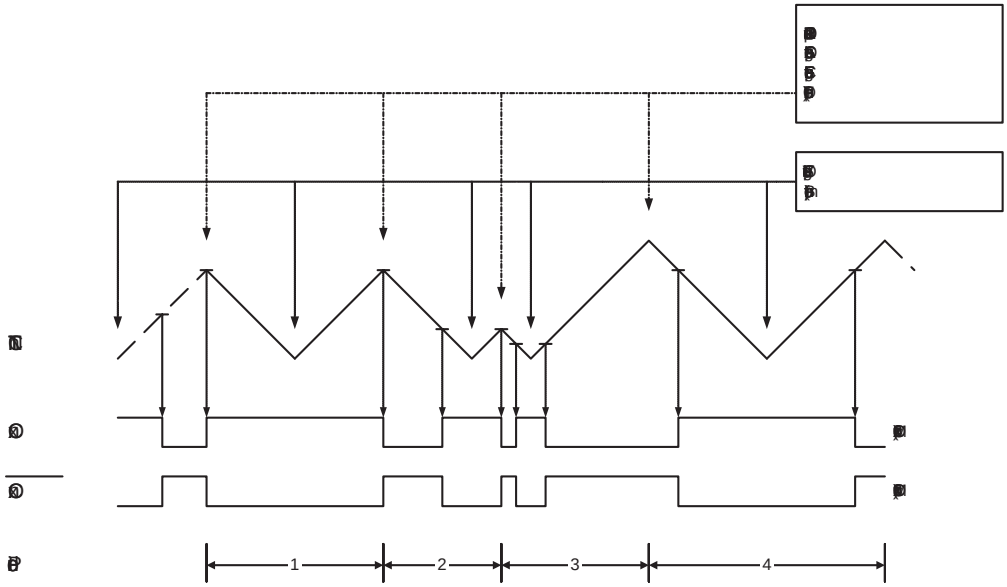


Figure 13-11.

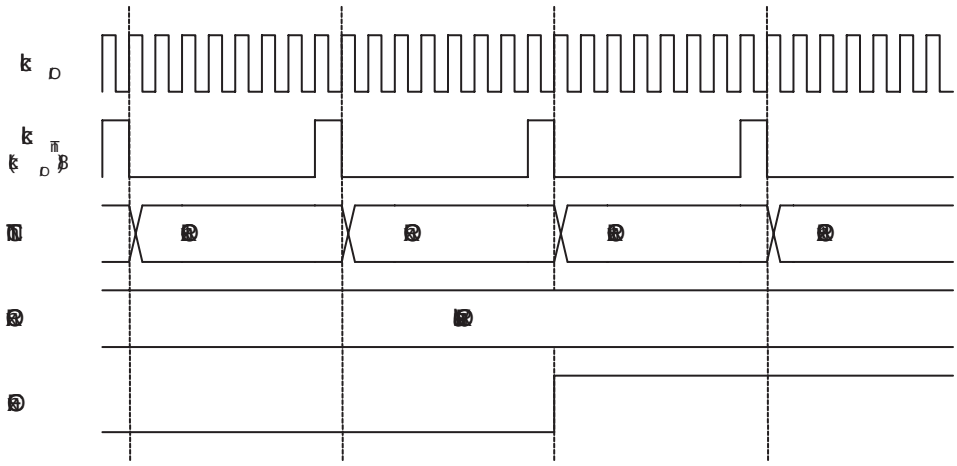
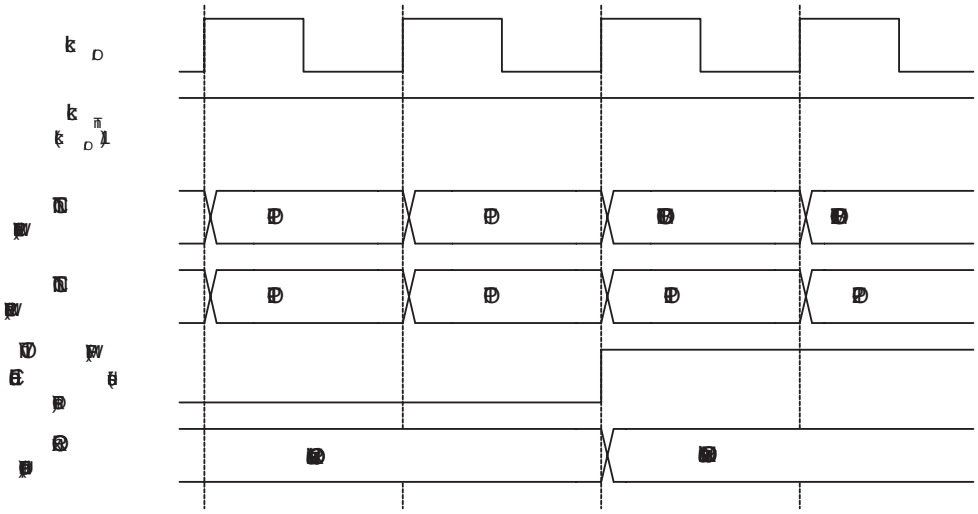


Figure 13-12.



13.10 16-bit Timer/Counter Register Description

13.10.1 Timer/Counter1 Control Register A – TCCR1A

Bit	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	–	–	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	R	R	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

- **Bit 7:6 – COM1A1:0: Compare Output Mode for Channel A**

- **Bit 5:4 – COM1B1:0: Compare Output Mode for Channel B**

[redacted] *Data Direction Register* [redacted]

[redacted]

Table 13-1. [redacted]

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	[redacted]
0	1	[redacted]
1	0	[redacted]
1	1	[redacted]

[redacted]

Table 13-2. [redacted]

COM1A1/COM1B1	COM1A0/COM1B0	Description
0	0	[redacted]
0	1	[redacted]
1	0	[redacted]
1	1	[redacted]

[redacted]

Table 13-4. [illegible]

Mode	WGM13	WGM12 (CTC1)	WGM11 (PWM11)	WGM10 (PWM10)	Timer/Counter Mode of Operation	TOP	Update of OCR1x at	TOV1 Flag Set on
0	0	0	0	0	[illegible]	[illegible]		
1	0	0	0	1	[illegible]	[illegible]		
2	0	0	1	0	[illegible]	[illegible]		
3	0	0	1	1	[illegible]	[illegible]		
4	0	1	0	0	[illegible]	[illegible]		
5	0	1	0	1	[illegible]	[illegible]		
6	0	1	1	0	[illegible]	[illegible]		
7	0	1	1	1	[illegible]	[illegible]		
8	1	0	0	0	[illegible]	[illegible]	[illegible]	
9	1	0	0	1	[illegible]	[illegible]		
10	1	0	1	0	[illegible]	[illegible]		
11	1	0	1	1	[illegible]	[illegible]		
12	1	1	0	0	[illegible]	[illegible]		
13	1	1	0	1	[illegible]	[illegible]		
14	1	1	1	0	[illegible]	[illegible]		
15	1	1	1	1	[illegible]	[illegible]		

[illegible]

13.10.2 Timer/Counter1 Control Register B – TCCR1B

Bit	7	6	5	4	3	2	1	0	
	ICNC1	ICES1	–	WGM13	WGM12	CS12	CS11	CS10	TCCR1B
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial value	0	0	0	0	0	0	0	0	

- **Bit 7 – ICNC1: Input Capture Noise Canceler**

[illegible]

- **Bit 6 – ICES1: Input Capture Edge Select**

[illegible]

[illegible]

[illegible]

Table 15-4. [illegible]

COM2A1	COM2A0	Description
0	0	[illegible]
0	1	[illegible]
1	0	[illegible]
1	1	[illegible]

[illegible]

- **Bits 5:4 – COM2B1:0: Compare Match Output B Mode**

[illegible]

[illegible]

[illegible]

Table 15-5. [illegible]

COM2B1	COM2B0	Description
0	0	[illegible]
0	1	[illegible]
1	0	[illegible]
1	1	[illegible]

[illegible]

Table 15-6. [illegible]

COM2B1	COM2B0	Description
0	0	[illegible]
0	1	[illegible]
1	0	[illegible]
1	1	[illegible]

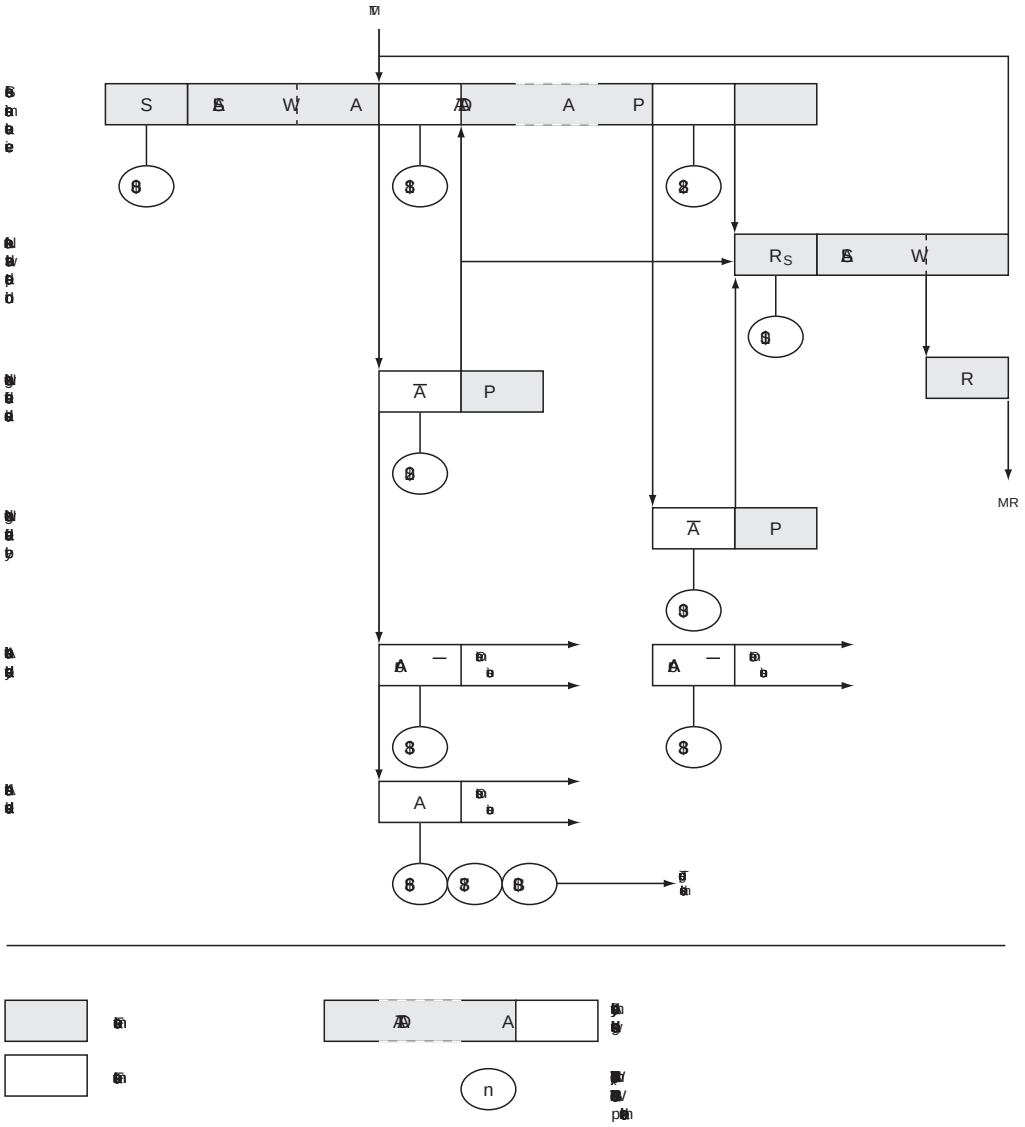
[illegible]

```
SPI_SlaveInit:
  ; Set MISO output, all others input
  ldi  r17,(1<<DD_MISO)
  out  DDR_SPI,r17
  ; Enable SPI
  ldi  r17,(1<<SPE)
  out  SPCR,r17
  ret

SPI_SlaveReceive:
  ; Wait for reception complete
  sbis SPSR,SPIF
  rjmp SPI_SlaveReceive
  ; Read received data and return
  in   r16,SPDR
  ret
```

```
void SPI_SlaveInit(void)
{
  /* Set MISO output, all others input */
  DDR_SPI = (1<<DD_MISO);
  /* Enable SPI */
  SPCR = (1<<SPE);
}

char SPI_SlaveReceive(void)
{
  /* Wait for reception complete */
  while(!(SPSR & (1<<SPIF)))
    ;
  /* Return Data Register */
  return SPDR;
}
```

1

Figure 19-13.



19.8.2 Master Receiver Mode

- **Bit 5 – ADATE: ADC Auto Trigger Enable**

- **Bit 4 – ADIF: ADC Interrupt Flag**

- **Bit 3 – ADIE: ADC Interrupt Enable**

- **Bits 2:0 – ADPS2:0: ADC Prescaler Select Bits**

Table 21-4.

ADPS2	ADPS1	ADPS0	Division Factor
0	0	0	2
0	0	1	2
0	1	0	4
0	1	1	8
1	0	0	16
1	0	1	32
1	1	0	64
1	1	1	128

23.0.1 Performing Page Erase by SPM

23.0.2 Filling the Temporary Buffer (Page Loading)

Table 24-2.

BLB0 Mode	BLB02	BLB01	Protection
1	1	1	
2	1	0	
3	0	0	
4	0	1	

Table 24-3.

BLB1 Mode	BLB12	BLB11	Protection
1	1	1	
2	1	0	
3	0	0	
4	0	1	

24.5 Entering the Boot Loader Program

Table 24-4.

BOOTRST	Reset Address
1	
0	)

Table 24-11. [illegible]

Variable		Corresponding Z-value[1]	Description
[illegible]	[illegible]		[illegible]
[illegible]	[illegible]		[illegible]
[illegible]	[illegible]		[illegible]
[illegible]	[illegible]		[illegible]
[illegible]	[illegible]		[illegible]
[illegible]	[illegible]		[illegible]

[illegible]

Table 25-12.

XA1	XA0	Action when XTAL1 is Pulsed
0	0	
0	1	
1	0	
1	1	

Table 25-13.

Command Byte	Command Executed

25.7 Parallel Programming

25.7.1 Enter Programming Mode

1 V_{CC}

2 $_{CC}$

$_{CC}$

3

4

5

6 $_{CC}$

1 V_{CC}

2 $_{CC}$

3 $_{CC}$ $_{CC}$

26.8 ADC Characteristics – Preliminary Data

Table 26-4.

Symbol	Parameter	Condition	Min	Typ	Max	Units
				2		
				2		
				2		
				2		

Figure 27-23. [redacted] C [redacted]

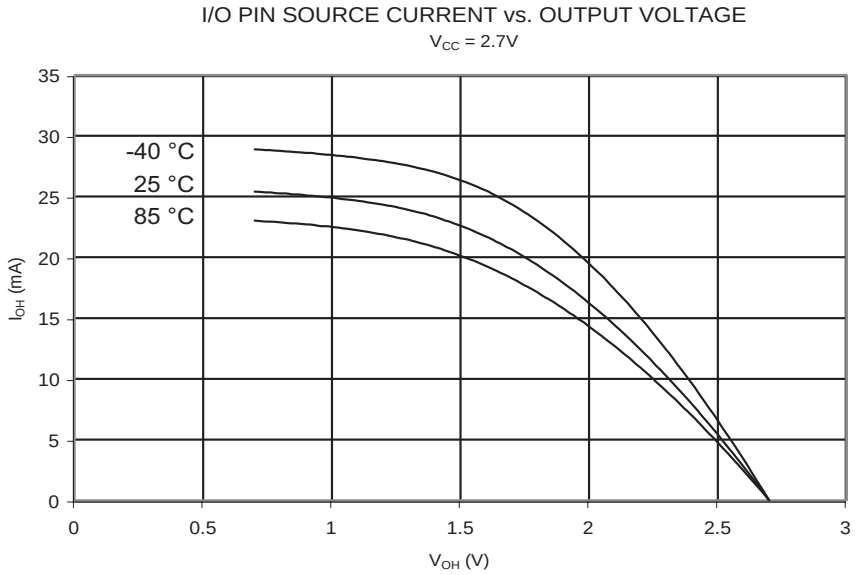
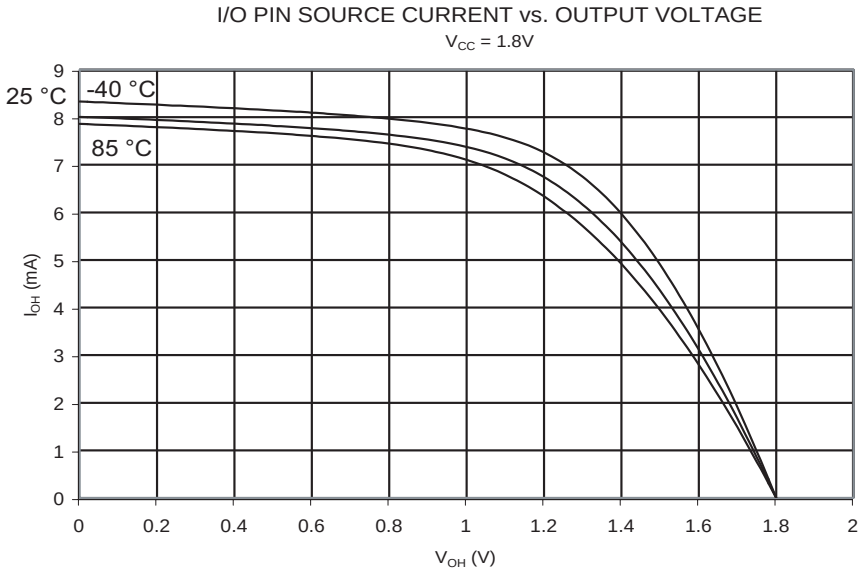


Figure 27-24. [redacted] C [redacted]



7 Power Management and Sleep Modes 37

8 System Control and Reset 43

9 Interrupts 54

10 I/O-Ports 64

11 External Interrupts 83

12 8-bit Timer/Counter0 with PWM 88

13 16-bit Timer/Counter1 with PWM 106

13.1Overview 106
13.2Accessing 16-bit Registers 108
13.3Timer/Counter Clock Sources 111
13.4Counter Unit 112
13.5Input Capture Unit 113
13.6Output Compare Units 115
13.7Compare Match Output Unit 117
13.8Modes of Operation 118
13.9Timer/Counter Timing Diagrams 125
13.1016-bit Timer/Counter Register Description 128

14 Timer/Counter0 and Time r/Counter1 Prescalers 135

15 8-bit Timer/Counter2 with PW M and Asynchronous Operation 138

15.1Overview 138
15.2Timer/Counter Clock Sources 139
15.3Counter Unit 139
15.4Output Compare Unit 140
15.5Compare Match Output Unit 142
15.6Modes of Operation 143
15.7Timer/Counter Timing Diagrams 147
15.88-bit Timer/Counter Register Description 149