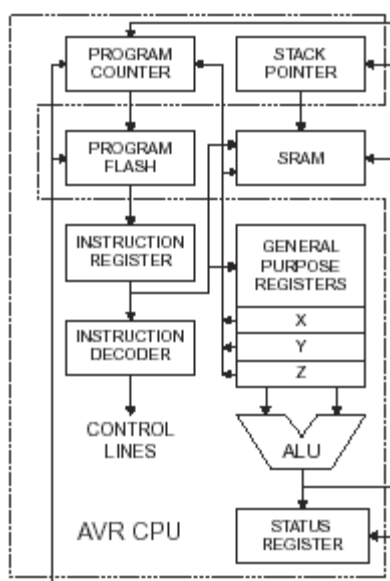


1.	PRINCIP FUNKCE MIKROPROCESORU	2
2.	MIKROPROCESOR ATMEGA162.....	4
3.	STAVOVÝ REGISTR.....	6
4.	PAMĚŤ DAT	7
4.1	REGISTRY A PAMĚŤOVÁ MÍSTA (SRAM)	7
4.2	ČTENÍ A ZÁPIS 16-TI BITOVÝCH REGISTRŮ	8
4.2.1	Zápis do 16-ti bitových registrů	8
4.2.2	Čtení 16-ti bitových registrů.....	8
4.3	ZÁSOBNÍK	9
4.4	PAMĚŤ DAT EEPROM	9
5.	ČASOVÁNÍ MIKROPROCESORU.....	11
6.	RESETOVACÍ SYSTÉM MIKROPROCESORU	13
7.	REŽIMY SNÍŽENÉ SPOTŘEBY	14
7.1	REGISTRY PRO NASTAVENÍ A POVOLENÍ REŽIMU SNÍŽENÉ SPOTŘEBY	15
7.2	NASTAVENÍ PRO REŽIM POWER-DOWN	15
8.	PORTY MIKROPROCESORU	16
8.1	ZÁKLADNÍ FUNKCE PORTU	16
8.2	ALTERNATIVNÍ FUNKCE PORTU.....	19
9.	PŘERUŠOVACÍ SYSTÉM	20
9.1	VNĚJŠÍ PŘERUŠENÍ.....	22
9.1.1	Přerušení INT0.....	23
9.1.2	Přerušení INT1	23
9.1.3	Přerušení INT2.....	24
9.1.4	Přerušení PCINT0.....	24
9.1.5	Přerušení PCINT1.....	25
10.	ČÍTAČE/ČASOVAČE	26
10.1	MÓDY ČASOVAČŮ.....	26
10.1.1	Normal mode	26
10.1.2	CTC mode.....	26
10.1.3	Fast PWM mode	27
10.1.4	Phase Correct PWM mode	27
10.1.5	Záchytný režim	28
10.2	ČASOVAČ 0.....	28
10.2.1	Nastavení nejpoužívanějšího módu CTC:.....	30
10.3	ČASOVAČ 1	31
10.3.1	Nastavení nejpoužívanějšího módu CTC:.....	33
10.4	ČASOVAČ 3	34
10.4.1	Nastavení nejpoužívanějšího módu CTC:.....	34
10.5	ČASOVAČ 2.....	35
10.5.1	Nastavení nejpoužívanějšího módu CTC:.....	35
10.5.2	Nastavení módu Fast PWM pro regulaci jasu displeje	35
10.5.3	Nastavení záchytného režimu pro měření časového intervalu.....	37
11.	SÉRIOVÝ KANÁL USART	40
11.1	SYNCHRONNÍ REŽIM	41
11.2	ASYNCHRONNÍ REŽIM	42
11.3	REGISTRY SÉRIOVÉHO KANÁLU	42
11.4	PŘÍKLADY NASTAVENÍ REGISTRŮ USART	47
12.	ANALOGOVÝ KOMPARÁTOR	48
13.	NASTAVENÍ PROCESORU POMOCÍ PROGRAMOVACÍCH PROPOJEK	50

1. Princip funkce mikroprocesoru

Mikroprocesor je logický obvod, jehož funkce je řízena programem. Program je uložen v paměti programu typu EEPROM. Při běhu programu čítač programu (PROGRAM COUNTER) vybere instrukci z paměti programu, přesune ji do dekodéru instrukcí a podle typu instrukce provádí následující činnost (přesun dat, aritmeticko-logické operace, bitové operace, skoky apod.). Paralelně s vykonáváním instrukcí jsou prováděny další činnosti nezávislé na programu – kontrola napájení, hlídání zdrojů přerušení, analogový komparátor, A/D převody, sériový kanál a další. Výsledky těchto činností mohou ovlivnit činnost programu, případně ji přerušit nebo dokonce resetovat procesor.



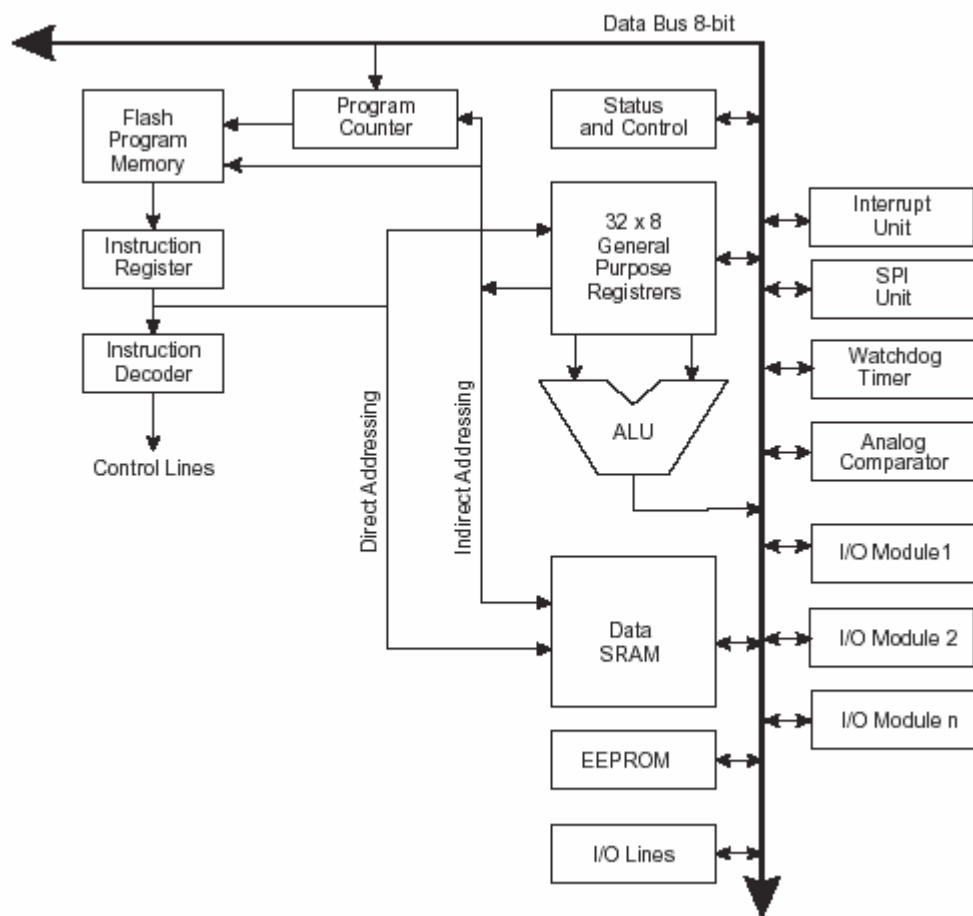
CPU (centrální procesorová jednotka) obsahuje ALU (aritmeticko-logická jednotka), PC (program counter), dekodér instrukcí, SREG (stavový registr), sadu registrů a SP (ukazatel vrcholu zásobníku). ALU vykonává pomocí registrů většinu instrukcí (aritmetické, logické a bitové instrukce a podmíněné skoky). SREG obsahuje 8 bitů-příznaků, které se nastavují po provedení instrukcí a je možné je testovat a větvit podle výsledku program. SP je ukazatel vrcholu zásobníku, což je paměť LIFO fyzicky umístěná v operační paměti SRAM a slouží k zálohování dat. SP obsahuje adresu naposled uložených dat.

V blokovém diagramu mikroprocesoru je zobrazeno propojení již zmiňovaných částí CPU s dalšími částmi mikroprocesoru. Mikroprocesor bude určitě obsahovat několik I/O portů (jejich počet se bude u každého typu lišit – 6 ÷ 35), operační paměť např. 1024 bytů, resetovací obvod, časovací obvody, obvody pro řízení přerušení. Dále může obsahovat paměť dat EEPROM, analogový komparátor, A/D převodník, čítače/časovače, obvod reálného času a další dle vkusu každého soudruha.

Pomocí I/O portů mikroprocesor komunikuje se svým okolím, porty jsou obousměrné a většina z nich má ještě další funkci (vstup ext. přerušení, vstup čítače, analog. komparátor, sériový kanál atd.).

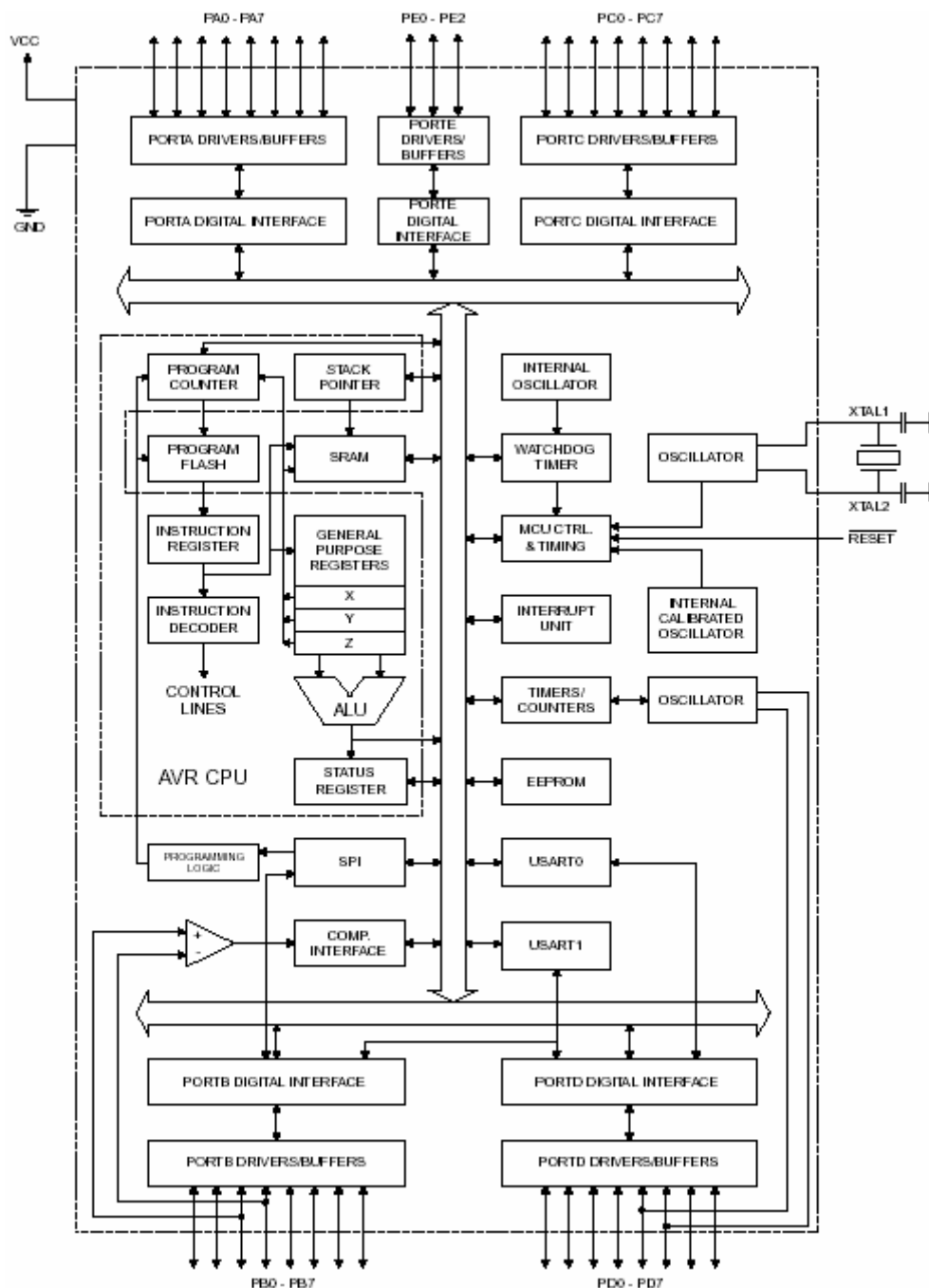
Resetovací obvod hlídá nestandardní stavy a v takovém případě provede reset mikroprocesoru, kdy program začíná od adresy 0000h a některé registry nastaví na určitou definovanou úroveň. Nestandardními stavy se myslí např. pokles napájecího napětí, impuls na resetovacím vstupu watchdog timer reset.

Časovací obvody řídí rychlost vykonávání jednotlivých instrukcí a čítání časovačů, jsou řízeny vnějším nebo vnitřním oscilátorem, jehož max. frekvence je 16MHz. Vykonávání jedné instrukce pak trvá 62,5 ns.



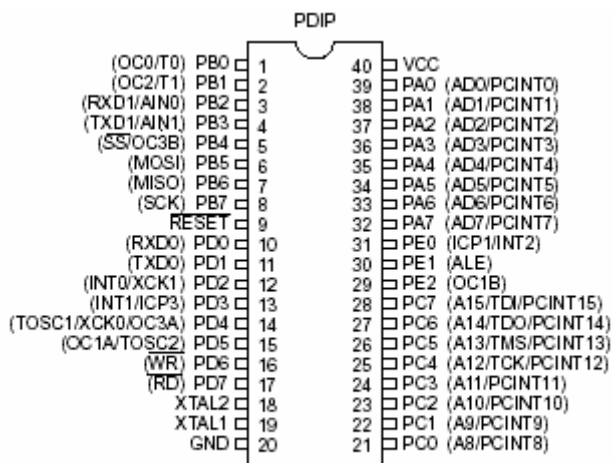
2. Mikroprocesor ATmega162

ATmega162 je 8-bitový mikroprocesor založený na architektuře RISC (omezená instrukční sada, rychlé vykonávání instrukcí). ALU provádí většinu instrukcí ve spojení s 32 registry uloženými v paměti [SRAM](#), z nichž 6 je organizováno jako 3 16-ti bitové registry. Paměť SRAM má kapacitu 1024 bytů. Paměť dat [EEPROM](#) má kapacitu 512 bytů.



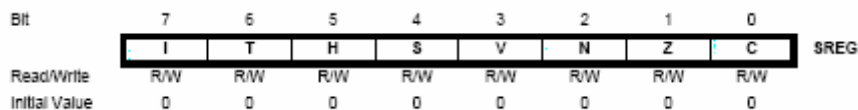
S okolím mikroprocesor komunikuje pomocí [35 I/O linek](#), u kterých je možno jednotlivě volit vstupní a výstupní režim. Všechny tyto linky mají ještě další funkce – [analogový komparátor](#), 2x [sériový kanál](#), 4x vstupy a výstupy čítačů/časovačů, 3+16x vnější přerušení. Některé I/O linky mají i čtyři funkce. Dále mikroprocesor nabízí dva osmibitové a dva šestnáctibitové

[čítače/časovače](#) s výstupy PWM a velkým množstvím režimů, z nichž jeden je čítání v reálném čase i při režimu snížené spotřeby. [Režimů snížené spotřeby](#) je celkem 5. [Časování](#) může probíhat z vnitřního nebo vnějšího krystalu. [Reset](#) můžou vyvolat čtyři zdroje – vnější resetovací vstup, watchdog timer, power-on (připojení napájení), Brown-out (pokles napětí pod nastavenou úroveň. Napájecí napětí je $1,8 \div 5,5V$. Frekvence krystalu max. 16 MHz.



3. Stavový registr

Stavový registr obsahuje informace o výsledcích posledních aritmetických operací. Podle těchto výsledků, které jsou obsaženy v příznakových bitech, je možno pomocí podmíněných skoků větvit program.



Bit 7 – I : Global Interrupt Enable

Tento bit v log. součinu s povolovacími bity jednotlivých přerušení povoluje dané [přerušení](#).

Bit 6 – T : Bit Copy Storage

Slouží pro bitové přesuny dat. Data přesuneme z výchozího bitu do T a z T do cílového bitu.

Bit 5 - H : Half Carry Flag

Indikuje přetečení mezi 3. a 4. bitem při některých aritmetických operacích. Využívá se hlavně s BCD aritmetikou.

Bit 4 – S : Sign Flag

Znaménkový příznak.

Bit 3 – V : Two's Komplement Overflow Flag

Je nastaven dojde-li k přetečení při aritmetických operacích.

Bit 2 – N : Negativ Flag

Indikuje záporný výsledek aritmetické nebo logické operace.

Bit 1 – Z : Zero Flag

Indikuje nulový výsledek aritmetické nebo logické operace.

Bit 0 – C : Carry Flag

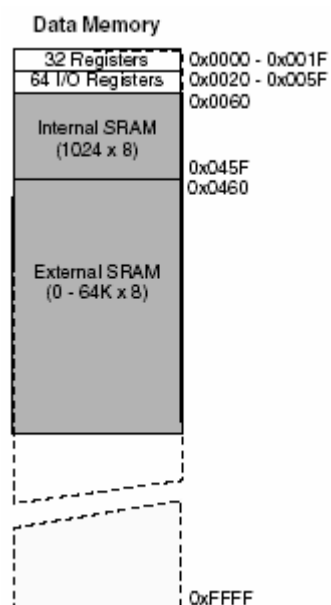
Indikuje přenos při aritmetických nebo logických operacích.

4. Paměť dat

4.1 Registry a paměťová místa (SRAM)

Paměť dat se skládá ze třech částí:

1. 0÷31 – 32 registrů
2. 32÷95 – 64 SFR (I/O registry)
3. 96÷1119 – 1024 paměťových míst



Paměťová místa jsou číslována od nuly, i když paměťové místo 0 leží na adrese 96. Stejně tak I/O registry jsou číslovány od nuly. Toto je umožněno tím, že pro přesuny z nebo do jednotlivých třech částí SRAM se používají jiné instrukce.

Registry se značí R0÷R31, přičemž registry R26÷R31 mohou být použity jako tři šestnáctibitové registry X,Y,Z. Registry jsou využívány pro většinu instrukcí.

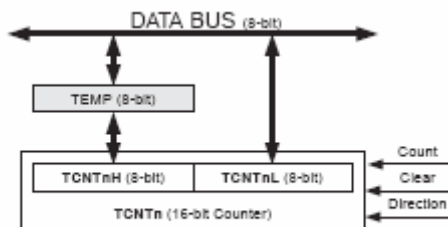
7	0	Addr.	
	R0	0x00	
	R1	0x01	
	R2	0x02	
	...		
	R13	0x0D	
	R14	0x0E	
	R15	0x0F	
	R16	0x10	
	R17	0x11	
	...		
	R26	0x1A	X-register Low Byte
	R27	0x1B	X-register High Byte
	R28	0x1C	Y-register Low Byte
	R29	0x1D	Y-register High Byte
	R30	0x1E	Z-register Low Byte
	R31	0x1F	Z-register High Byte

4.2 Čtení a zápis 16-ti bitových registrů

V programu můžeme v jednom strojovém cyklu zapisovat nebo číst pouze jeden osmibitový registr. S druhým osmibitovým registrem komunikujeme v dalším strojovém cyklu. Mezi tím může dojít ke změně (např. čítáním) jednoho z registrů a tím vznikne chyba celého řádu. Pro ošetření této situace procesor využívá při komunikaci pomocný registr TEMP, díky kterému se komunikuje se všemi 16-ti bity najednou.

4.2.1 Zápis do 16-ti bitových registrů

Zápis si vysvětlíme na čítacím registru čítače 3 TCNT3. Tento registr se skládá z registrů TCNT3H a TCNT3L. Zápis provádíme ve dvou osmibitových operacích, přičemž jako první zapisujeme data do horního bytu, tato data se uloží do registru TEMP a při zápisu do dolního bytu se přepíše TEMP do TCNT3H – tedy všech 16 bitů najednou.



Nulování čítacích registrů:

```
LDI    R17,0
OUT    TCNT3H,R17
OUT    TCNT3L,R17
```

4.2.2 Čtení 16-ti bitových registrů

Při čtení postupujeme obráceně. Nejdříve přečteme dolní byte a při tom se zkopíruje obsah horního bytu do registru TEMP, ze kterého ho přečteme při následující operaci při čtení horního bytu.

```
IN     R16,TCNT3L
IN     R17,TCNT3H
```


4.3 Zásobník

V paměti SRAM je uložen zásobník. Je to paměť typu LIFO, to znamená, že se bude využívat ke krátkodobému zálohování dat, kdy musí platit, že data, která si zálohujeme jako poslední budeme potřebovat jako první. To je hlavně při ukládání návratových adres a důležitých registrů při skocích do podprogramů. Při návratech z podprogramů se tyto data znovu v opačném pořadí obnoví.

4.4 Paměť dat EEPROM

S pamětí dat EEPROM (512B) komunikujeme pomocí registru EEDR (EEPROM DATA REGISTR). Adresu bytu se kterým chceme komunikovat zapíšeme do registrů EARL (8bitů), EARH (1bit). Pro přesun dat je nutné nastavit některé bity v reg. EECR.

Postup pro zápis:

1. Testujeme bit EEWB v reg. EECR, který log.0 signalizuje dokončený předchozí zápis a čekáme dokud není log.0.
2. Zapíšeme novou adresu do EEARH, EEARL.
3. Zapíšeme data do EEDR.
4. Zapíšeme log.1 do bitu EEMWB v reg. EECR.
5. Do čtyř strojových cyklů po nastavení EEMWB musíme zapsat log.1 do bitu EEWB v reg. EECR, pak se provede uložení dat. Jinak se bit EEMWB nuluje a nastavení EEWB uložení neprovede.

```
LDI    R16,0           ; adresa dolní byt
OUT    EEARL,R16

LDI    R16,0           ; adresa horní 1 bit
OUT    EEARH,R16

LDS    R16,ZAPISOVANADATA ; do EEDR data
OUT    EEDR,R16
RCALL  ZAPIS

.....

ZAPIS: SBIC    EECR,EEWB    ; test jestli je ukončený zápis
      RJMP    ZAPIS

      SBI     EECR,EEMWB    ; nastavení eeprom master write enable
      SBI     EECR,EEWB    ; nastavení eeprom write enable

ZAPIS1: SBIC   EECR,EEWB    ; čekání na ukončení zápisu
      RJMP    ZAPIS1

      RET
```

Postup pro čtení:

1. Zapišeme novou adresu do EEARH, EEARL.
2. Nastavíme bit EERE v reg. EECR do log.1.
3. Čtení trvá čtyři strojové cykly, během čtení zastavena CPU, data se načtou do reg. EEDR.

```
LDI    R16,0           ; adresa dolní byt
OUT    EEARL,R16

LDI    R16,0           ; adresa horní 1 bit
OUT    EEARH,R16

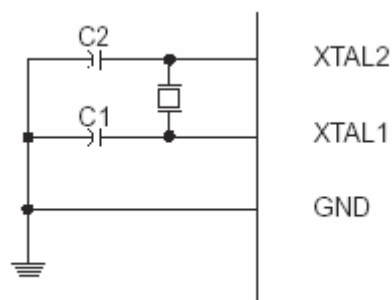
SBI    EECR,EERE       ; nastavení eeprom read enable
IN     R16,EEDR
```

5. Časování mikroprocesoru

Časovací obvody mikroprocesoru řídí rychlost všech důležitých částí. Zdroje frekvence pro časování je možno rozdělit do dvou částí:

1. Oscilátor řízený vnějším krystalem.

K procesoru můžeme připojit krystal max. frekvence 16 MHz a je nutné ho zapojit podle následujícího schématu. Kondenzátory C1 a C2 mají kapacitu 12 – 22 pF.



2. Vnitřní kalibrovaný oscilátor

Vnitřní kalibrovaný RC oscilátor je nastaven na frekvenci 8 MHz. Kalibrační konstanta je do reg. OSCCAL zapsána vždy po resetu a je možno ji v programu měnit.

Oscillator Calibration Register – OSCCAL

Bit	7	6	5	4	3	2	1	0	
	–	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	Device Specific Calibration Value							

Volba oscilátoru se provádí při programování programovacími propojkami. **Jeden strojový cyklus** (doba vykonávání jedné instrukce) trvá jednu periodu oscilátoru, tzn. že při použití krystalu 8 MHz bude trvat vykonávání jedné instrukce 125 ns.

Frekvenci oscilátoru je možno ještě dělit pomocí vnitřní děličky 1, 2, 4, 8, 16, 32, 64, 128, 256. Volba dělicího poměru se provádí nastavením registru CLKPR.

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	See Bit Description				

Změna se provádí následujícím způsobem:

1. Bit CLKPE nastavíme do log1 a ostatní bity do log0.
2. Do čtyřech strojových cyklů zapíšeme do bitů CLKPS3÷0 novou hodnotu podle tabulky 15, jinak bude bit CLKPE nulován. Bit CLKPE se nuluje i po zápisu do CLKPS3÷0.

Table 15. Clock Prescaler Select

CLKPS3	CLKPS2	CLKPS1	CLKPS0	Clock Division Factor
0	0	0	0	1
0	0	0	1	2
0	0	1	0	4
0	0	1	1	8
0	1	0	0	16
0	1	0	1	32
0	1	1	0	64
0	1	1	1	128
1	0	0	0	256
1	0	0	1	Reserved
1	0	1	0	Reserved
1	0	1	1	Reserved
1	1	0	0	Reserved
1	1	0	1	Reserved
1	1	1	0	Reserved
1	1	1	1	Reserved

6. Resetovací systém mikroprocesoru

Dojde-li k resetu mikroprocesoru, všechny SFR registry budou nastaveny na danou inicializační hodnotu a program se přesune na adresu 00h. [Porty](#) jsou ve stavu vysoké impedance.

Zdroje resetu:

1. Power-on reset – k resetu dojde tehdy, když napájecí napětí je pod úrovní nastaveného napětí Power-on reset.
2. Externí reset - k resetu dojde tehdy, když na resetovacím vstupu je log.0 po dobu
3. Watchdog reset – je-li spuštěn watchdog timer, musí program pravidelně do nastavené doby ($16\text{ms} \div 2, 1\text{s}$) provést instrukci WDR (watchdog reset). Pokud se program zacyklí a tato instrukce se neprovede, dojde k resetu.
4. Brown-out reset – je-li aktivován, dojde k resetu při poklesu napájecího napětí pod nastavenou úroveň Brown-out reset. Tato úroveň může být nastavena v rozsahu $1,7 \div 4,3\text{V}$. K resetu nedojde je-li pokles kratší než $2\mu\text{s}$.

7. Režimy snížené spotřeby

Mikroprocesor má pět režimů snížené spotřeby, které se od sebe liší spotřebou v daném režimu a možnostmi probuzení procesoru z jednotlivých režimů. Režim zvolíme pomocí bitů SM2:0 podle tab.13. Dále musíme nastavením bitu SE (sleep enable) do log.1 povolit usnutí procesoru a k usnutí dojde provedením instrukce SLEEP.

Table 16. Sleep Mode Select

SM2	SM1	SM0	Sleep Mode
0	0	0	Idle
0	0	1	Reserved
0	1	0	Power-down
0	1	1	Power-save
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Standby ⁽¹⁾
1	1	1	Extended Standby ⁽¹⁾

Note: 1. Standby mode and Extended Standby mode are only available with external crystals or resonators.

Probuzení procesoru může nastat různými zdroji dle tab.17

Table 17. Active Clock domains and Wake up sources in the different sleep modes

Sleep Mode	Active Clock domains				Oscillators		Wake-up Sources			
	clk _{CPU}	clk _{FLASH}	clk _{IO}	clk _{ASY}	Main Clock Source Enabled	Timer Osc Enabled	INT2 INT1 INT0 and Pin Change	Timer2	SPM/ EEPROM Ready	Other I/O
Idle			X	X	X	X ⁽²⁾	X	X	X	X
Power-down							X ⁽³⁾			
Power-save				X ⁽²⁾		X ⁽²⁾	X ⁽³⁾	X ⁽²⁾		
Standby ⁽¹⁾					X		X ⁽³⁾			
Extended Standby ⁽¹⁾				X ⁽²⁾	X	X ⁽²⁾	X ⁽³⁾	X ⁽²⁾		

Notes: 1. External Crystal or resonator selected as clock source
2. If AS2 bit in ASSR is set
3. For INT1 and INT0, only level interrupt

Pro nás bude nejzajímavější mód Power-down, který má spotřebu $I_{CC} < 1\mu A$ a processor můžeme probudit vnějším přerušením. Režim Idle má více zdrojů probuzení, ale vysokou spotřebu, režim Power-save umožňuje navíc probuzení časovačem T2 který ovšem pracuje s externím krystalem 32kHz (vhodné využít pro čítání reálného času), režimy Standby a Extended Standby jsou podobné jako Power-down a Power-save, ale pracují jen s externím krystalem.

7.1 Registry pro nastavení a povolení režimu snížené spotřeby

MCU Control Register – MCUCR

Bit	7	6	5	4	3	2	1	0	
	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit 5 – SE – Sleep Enable – povolení režimu snížené spotřeby

Bity SM2:0 – bity pro nastavení režimu

MCU Control and Status Register – MCUCSR

Bit	7	6	5	4	3	2	1	0	
	JTD	–	SM2	JTRF	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Extended MCU Control Register – EMCUCR

Bit	7	6	5	4	3	2	1	0	
	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	EMCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

7.2 Nastavení pro režim Power-down

Bit SE nastavíme do log.1 a bity SM2:0 kombinací 010. Potom kdekoli v programu instrukcí SLEEP uvedeme procesor do režimu snížené spotřeby. Probuzení nastane [vnějším přerušením](#) INT0, INT1, které musí být nastaveny aby reagovaly na úroveň, INT2 nebo PCINT0, PCINT1. Po probuzení se nejdříve obslouží přerušení a potom program pokračuje za instrukcí SLEEP, která procesor uspala.

```
LDI    R16,0B00001000 ; povolení přerušení PCINT0
OUT    GICR,R16

LDI    R16,0B11111111 ; výběrbitů které budou vyvolávat přerušení (všechny)
STS    PCMSK0,R16

IN     R16,MCUCR      ; povolení sleep módu -bit5 - SE- log1, bit4 - SM1- log1
ORI    R16,0B00110000
OUT    MCUCR,R16

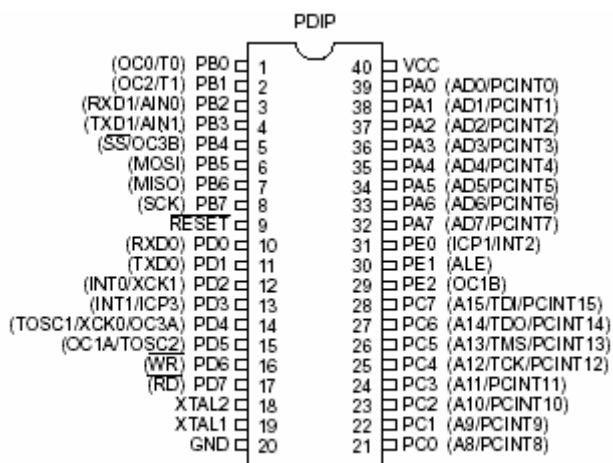
IN     R16,MCUCSR     ; bit5 - SM2 - log0
ANDI   R16,0B11011111
OUT    MCUCSR,R16

IN     R16,EMCUCR     ; bit7 - SM0 - log0
ANDI   R16,0B01111111
OUT    EMCUCR,R16

SEI
.
.
.
SLEEP
```

8. Porty mikroprocesoru

Mikroprocesor využívá ke komunikaci s okolím 35 I/O linek, které jsou organizovány v pěti portech PA÷PE. Všechny tyto linky mají ještě další funkce – [analogový komparátor](#), 2x [sériový kanál](#), 4x vstupy a výstupy čítačů/časovačů, 3+16x vnější přerušení. Některé I/O linky mají i čtyři funkce.

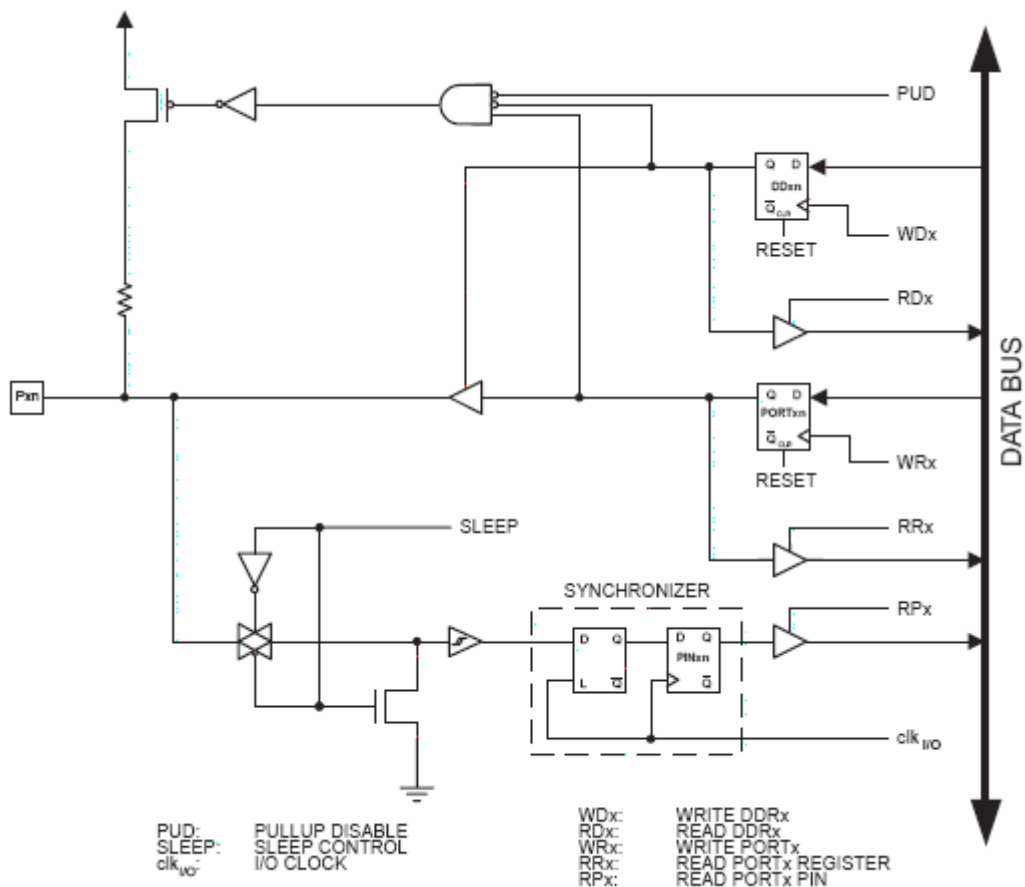


Každý bit portu má tedy základní funkci – vstup dat (PINx.y) a výstup dat (PORTx.y). Bit PINx.y můžeme pouze číst a udává nám stav bitu portu. Bit PORTx.y můžeme zapisovat (i číst) a tímto bitem ovládáme výstupní budič - nastavujeme logickou úroveň na výstup. Vedle této základní funkce mají bity portu alternativní funkce, které je nutné povolit.

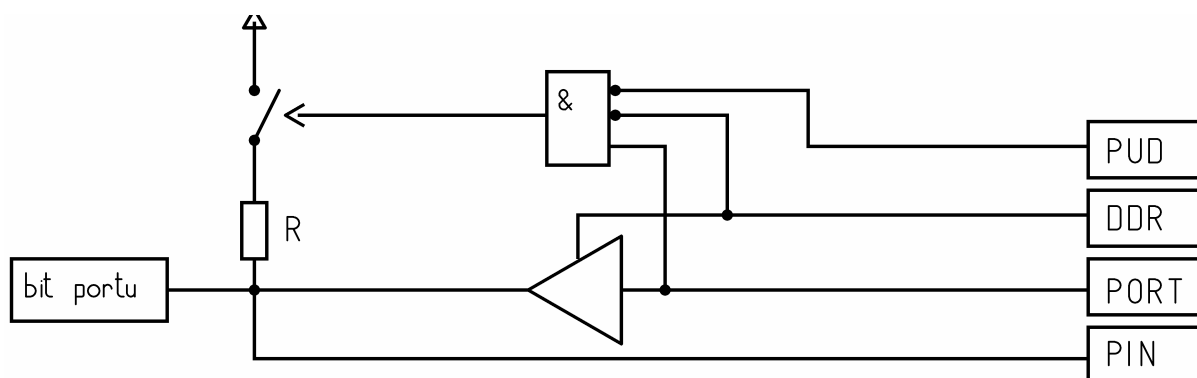
- budeme-li chtít testovat stav bitu portu, musíme nastavit bit jako vstupní a podle potřeby zapnout nebo vypnout PU-rezistor.
- při použití bitu pro spínání nastavíme bit jako výstupní.
- při využití alternativní funkce tuto funkci povolíme a potom je nastavení portu (vstup/výstup, výst. data) řízeno touto funkcí, která má vyšší prioritu.

8.1 Základní funkce portu

Každý bit je možno samostatně přepnout do vstupního, nebo výstupního režimu. Všem najednou můžeme zapnout nebo vypnout PULL-UP rezistor (20÷50kΩ) bitem PUD v registru SFIOR. Nezávisle na tom se rezistor při přepnutí do výstupního režimu automaticky vypíná.



Zjednodušené schéma portu:



Každý pin portu se skládá ze třech bitů + jeden společný pro všechny piny:

1. DDxn – tento bit přepíná vstupní a výstupní režim
2. PORTxn – toto je bit registru portu, do kterého zapisujeme
3. PINxn – skutečný stav pinu portu, nemusí odpovídat zapsanému stavu
4. PUD – tento bit zapíná nebo vzpíná PULL-UP rezistor všem pinům společně.

Potom může nastat několik kombinací, které jsou shrnuty v tabulce:

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

Je-li DDxn v log0 jedná se o vstupní režim:

1. Je-li na PORTxn zapsána log0, nezávisle na PUD je na výstupu stav vysoké impedance Z.
2. Je-li na PORTxn zapsána log1 a PUD log0, je PULL-UP rezistor zapnut, na výstupu bude log1 přes PULL-UP rezistor.
3. Je-li na PORTxn zapsána log1 a PUD log1, je PULL-UP rezistor vypnut, na výstupu bude stav vysoké impedance Z.

Je-li DDxn v log1 jedná se o výstupní režim (PULL-UP rezistor nezávisle na PUD vypnut):

1. Je-li na PORTxn zapsána log0, na výstupu bude log0 z výstupního budiče.
2. Je-li na PORTxn zapsána log1, na výstupu bude log1 z výstupního budiče.

Port A Data Register – PORTA

Bit	7	6	5	4	3	2	1	0	
	PORTA7	PORTA6	PORTA5	PORTA4	PORTA3	PORTA2	PORTA1	PORTA0	PORTA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Port A Data Direction Register – DDRA

Bit	7	6	5	4	3	2	1	0	
	DDA7	DDA6	DDA5	DDA4	DDA3	DDA2	DDA1	DDA0	DDRA
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Port A Input Pins Address – PINA

Bit	7	6	5	4	3	2	1	0	
	PINA7	PINA6	PINA5	PINA4	PINA3	PINA2	PINA1	PINA0	PINA
Read/Write	R	R	R	R	R	R	R	R	
Initial Value	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	

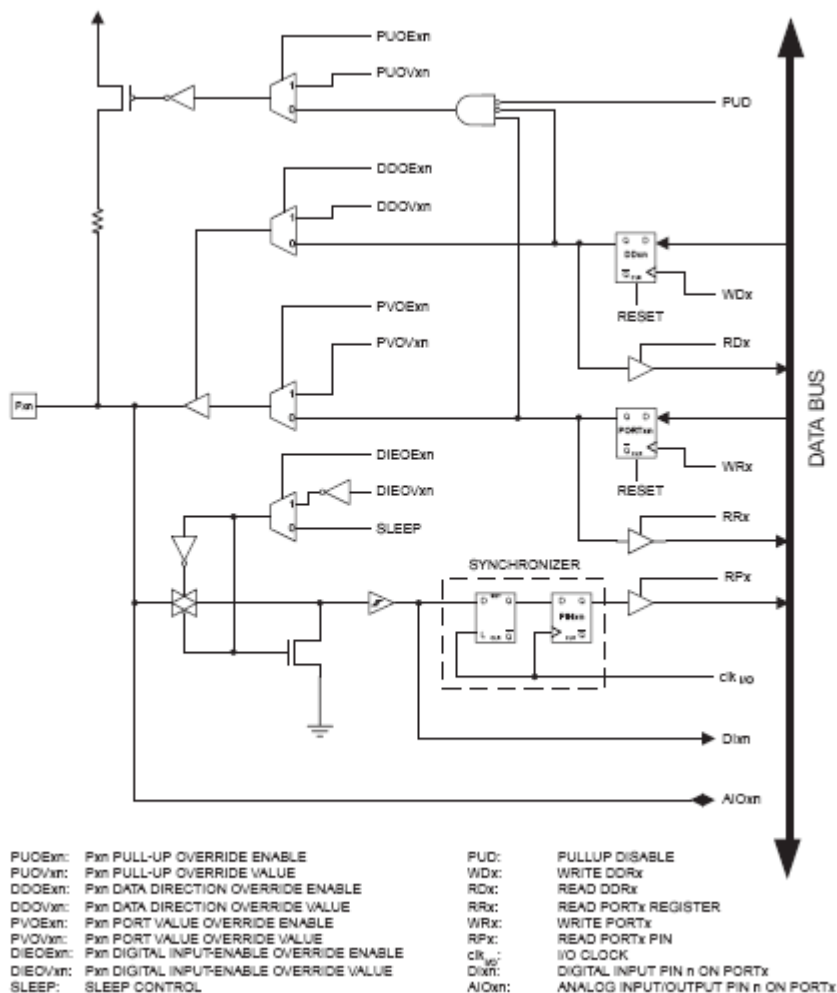
Maximální proud ve výstupním režimu je 40mA, ve vstupním režimu je omezen PULL-UP rezistorem.

Po [resetu](#) jsou všechny porty ve vstupním režimu, bit PUD je v log.0 (rezistor povolen) a všechny registry PORT jsou v log.0 (jednotlivé PU rezistory zakázané) , to znamená, že porty jsou ve stavu vysoké impedance.

8.2 Alternativní funkce portu

Na obr. 32 je celkové schéma portu s prvky které umožňují ovládat port (PU- rezistor, DD- vstup/výstup, PV- výstupní logická úroveň, DI- ovládání vstupního vodiče). Pomocí těchto prvků se při aktivování alternativní funkce provede nastavení portu tak, aby vyhovovalo této funkci (neplatí pro všechny funkce –např. neplatí pro vnější přerušovací vstupy).

Figure 32. Alternate Port Functions⁽¹⁾



9. Prerušovací systém

Mikroprocesor obsahuje 28 zdrojů přerušení a je možno je rozdělit do těchto skupin:

- [reset](#)
- [vnější přerušení](#)
- [přerušení od časovačů](#)
- [přerušení od sériových kanálů](#)
- [zápis do paměti EEPROM](#) a pam. Programu
- [analogový komparátor](#)

Vector No.	Program Address ⁽²⁾	Source	Interrupt Definition
1	0x000 ⁽¹⁾	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	0x002	INT0	External Interrupt Request 0
3	0x004	INT1	External Interrupt Request 1
4	0x006	INT2	External Interrupt Request 2
5	0x008	PCINT0	Pin Change Interrupt Request 0
6	0x00A	PCINT1	Pin Change Interrupt Request 1
7	0x00C	TIMER3 CAPT	Timer/Counter3 Capture Event
8	0x00E	TIMER3 COMPA	Timer/Counter3 Compare Match A
9	0x010	TIMER3 COMPB	Timer/Counter3 Compare Match B
10	0x012	TIMER3 OVF	Timer/Counter3 Overflow
11	0x014	TIMER2 COMP	Timer/Counter2 Compare Match
12	0x016	TIMER2 OVF	Timer/Counter2 Overflow
13	0x018	TIMER1 CAPT	Timer/Counter1 Capture Event
14	0x01A	TIMER1 COMPA	Timer/Counter1 Compare Match A
15	0x01C	TIMER1 COMPB	Timer/Counter1 Compare Match B
16	0x01E	TIMER1 OVF	Timer/Counter1 Overflow
17	0x020	TIMER0 COMP	Timer/Counter0 Compare Match
18	0x022	TIMER0 OVF	Timer/Counter0 Overflow
19	0x024	SPI, STC	Serial Transfer Complete
20	0x026	USART0, RXC	USART0, Rx Complete
21	0x028	USART1, RXC	USART1, Rx Complete
22	0x02A	USART0, UDRE	USART0 Data Register Empty
23	0x02C	USART1, UDRE	USART1 Data Register Empty
24	0x02E	USART0, TXC	USART0, Tx Complete
25	0x030	USART1, TXC	USART1, Tx Complete
26	0x032	EE_RDY	EEPROM Ready
27	0x034	ANA_COMP	Analog Comparator
28	0x036	SPM_RDY	Store Program Memory Ready

V následující tabulce jsou uvedeny vektory přerušení a názvy vektorů, které můžeme použít v programu.

***** INTERRUPT VECTORS *****

INT0addr	= 0x0002	; External Interrupt Request 0
INT1addr	= 0x0004	; External Interrupt Request 1
INT2addr	= 0x0006	; External Interrupt Request 2
PCI0addr	= 0x0008	; Pin Change Interrupt Request 0
PCI1addr	= 0x000a	; Pin Change Interrupt Request 1
ICP3addr	= 0x000c	; Timer/Counter3 Capture Event
OC3Aaddr	= 0x000e	; Timer/Counter3 Compare Match A
OC3Baddr	= 0x0010	; Timer/Counter3 Compare Match B
OVF3addr	= 0x0012	; Timer/Counter3 Overflow
OC2addr	= 0x0014	; Timer/Counter2 Compare Match
OVF2addr	= 0x0016	; Timer/Counter2 Overflow
ICP1addr	= 0x0018	; Timer/Counter1 Capture Event
OC1Aaddr	= 0x001a	; Timer/Counter1 Compare Match A
OC1Baddr	= 0x001c	; Timer/Counter Compare Match B
OVF1addr	= 0x001e	; Timer/Counter1 Overflow
OC0addr	= 0x0020	; Timer/Counter0 Compare Match
OVF0addr	= 0x0022	; Timer/Counter0 Overflow
SPIaddr	= 0x0024	; SPI Serial Transfer Complete
URXC0addr	= 0x0026	; USART0, Rx Complete
URXC1addr	= 0x0028	; USART1, Rx Complete
UDRE0addr	= 0x002a	; USART0 Data register Empty
UDRE1addr	= 0x002c	; USART1, Data register Empty
UTXC0addr	= 0x002e	; USART0, Tx Complete
UTXC1addr	= 0x0030	; USART1, Tx Complete
ERDYaddr	= 0x0032	; EEPROM Ready
ACIaddr	= 0x0034	; Analog Comparator
SPMRaddr	= 0x0036	; Store Program Memory Read

Při vzniku žádosti o přerušení (událost na přerušovacích vstupech, v čítači/časovači, v sériovém kanálu ...) dojde nejdříve k nastavení odpovídajícího příznaku přerušení. Je-li přerušení povoleno a neobsluhuje-li se jiné přerušení - dojde k obsluze tohoto přerušení, kdy nemůže být obsluhováno jiné přerušení:

- do zásobníku se uloží návratová adresa
- přeruší se vykonávání hlavního programu a program pokračuje na odpovídajícím vektoru přerušení, kde je skok na obslužný program
- nuluje se příznak přerušení
- obsluha končí instrukcí RETI, která vybere návratovou adresu ze zásobníku a vrátí se do hlavního programu a ze systému odhlásí obsluhu přerušení – od této doby může být obsluženo další přerušení, jehož žádost vznikla při obsluze tohoto přerušení

Zásady a chyby při používání přerušení:

- Obslužný program přerušení musí být co nejkratší aby neblokoval obsluhu dalších přerušení. Vznikne-li při obsluze přerušení další žádost o přerušení, nastaví se odpovídající příznak, který zajistí, že k obsluze dojde po skončení obsluhy prvního přerušení. Pokud by však byl obslužný program dlouhý, může při tom dojít k několika událostem od jednoho zdroje přerušení, ale příznak se nastaví pouze jeden. Tím nebude několik událostí zpracováno.
- Má-li obsluha přerušení vykonat časově náročný program, nastavíme v obsluze přerušení pouze příznak, který budeme testovat v hlavním programu.
- Neskončí-li obslužný program instrukcí RETI (návrat instrukcí RET) systém je stále v režimu obsluhy přerušení a tím se nemůže obsluhovat další přerušení.
- Je-li přerušení zakázané, při žádosti o přerušení se nastaví příznak. Pokud přerušení povolíme, okamžitě se vyvolá obsluha přerušení. Pokud tomu chceme zabránit musíme před povolením přerušení příznak programově vynulovat.

9.1 Vnější přerušení

Vnější přerušení má pět zdrojů

- INT0
- INT1
- INT2
- PCINT0
- PCINT1

Přerušení INT0 a INT1 jsou shodné a jsou vyvolávané signály na pinech PD2 a PD3. Je možno nastavit reakci na úroveň log0, sestupnou, nástupnou hranu nebo oba typy hran.

Přerušení INT2 reaguje na signál na pinu PE0 a je možno ho nastavit na sestupnou nebo nástupnou hranu.

Přerušení PCINT0 reaguje na libovolnou změnu na portu PA. Přerušení PCINT1 reaguje na libovolnou změnu na portu PC. Přičemž je možno vybrat piny na obou portech, které budou přerušení vyvolávat.

Při žádosti o přerušení se nastaví příznak přerušení v registru GIFR.

General Interrupt Flag
Register – GIFR

Bit	7	6	5	4	3	2	1	0	
	INTF1	INTF0	INTF2	PCIF1	PCIF0	–	–	–	GIFR
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

Je-li přerušení povoleno a neobsluhuje-li se právě jiné přerušení, dojde k přerušení hlavního programu, uloží se návratová adresa do zásobníku a program pokračuje na odpovídajícím vektoru přerušení. V tento okamžik se nuluje příznak. Na vektoru přerušení je skok na

- 0,0 – úroveň log0
- 0,1 – likelihood $1 < \infty$

- Register –

Bit	7	6
	SRE	SRW

Nastavení typu signálu na který bude přerušení reagovat – bity ISC11, ISC10 v registru MCUCR.

- 0,0 – úroveň log0
- 0,1 – libovolná změna
- 1,0 – sestupná hrana
- 1,1 – nástupná hrana
-

MCU Control Register – MCUCR

Bit	7	6	5	4	3	2	1	0	
	SRE	SRW10	SE	SM1	ISC11	ISC10	ISC01	ISC00	MCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

9.1.3 Přerušení INT2

Vektor přerušení je 006 (*INT2addr*), povolení přerušení – INT2 - bit 5 v registru GICR.

General Interrupt Control Register – GICR

Bit	7	6	5	4	3	2	1	0	
	INT1	INT0	INT2	PCIE1	PCIE0	–	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Nastavení typu signálu na který bude přerušení reagovat – bit ISC2 v registru EMCUCR.

- 0 – sestupná hrana
- 1 – nástupná hrana

Extended MCU Control Register – EMCUCR

Bit	7	6	5	4	3	2	1	0	
	SM0	SRL2	SRL1	SRL0	SRW01	SRW00	SRW11	ISC2	EMCUCR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

9.1.4 Přerušení PCINT0

Vektor přerušení je 008(*PCI0addr*), povolení přerušení – PCIE0 - bit 3 v registru GICR.

General Interrupt Control Register – GICR

Bit	7	6	5	4	3	2	1	0	
	INT1	INT0	INT2	PCIE1	PCIE0	–	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Nastavení bitů portu PA, na kterých změna stavu vyvolá přerušení – registr PCMSK0.

Pin Change Mask Register 0 – PCMSK0

Bit	7	6	5	4	3	2	1	0	
	PCINT7	PCINT6	PCINT5	PCINT4	PCINT3	PCINT2	PCINT1	PCINT0	PCMSK0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Nastavením bitu v tomto registru do log1 a povolením přerušení bude libovolná změna na odpovídajícím bitu portu PA vyvolávat přerušení.

9.1.5 Přerušení PCINT1

Vektor přerušení je 00A (*PC11addr*), povolení přerušení – PCIE1 - bit 4 v registru GICR.

General Interrupt Control Register – GICR

Bit	7	6	5	4	3	2	1	0	
	INT1	INT0	INT2	PCIE1	PCIE0	–	IVSEL	IVCE	GICR
Read/Write	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Nastavení bitů portu PC, na kterých změna stavu vyvolá přerušení – registr PCMSK1.

Pin Change Mask Register 1 – PCMSK1

Bit	7	6	5	4	3	2	1	0	
	PCINT15	PCINT14	PCINT13	PCINT12	PCINT11	PCINT10	PCINT9	PCINT9	PCMSK1
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Nastavením bitu v tomto registru do log1 a povolením přerušení bude libovolná změna na odpovídajícím bitu portu PC vyvolávat přerušení.

10. Čítače/časovače

Mikroprocesor obsahuje dva osmibitové čítače/časovače (TIMER0, TIMER2), a dva šestnáctibitové čítače/časovače (TIMER1, TIMER3). Tyto čtyři časovače nabízí 12 zdrojů přerušení – přetečení časovačů, shoda s porovnávacími registry, dále umožňují ovládat 6 PWM kanálů, generování frekvence na výstupních bitech a časování RTC, mohou čítat vnější impulzy. Všechny časovače mohou pracovat v různých režimech z nichž nejjednodušší je CTC mód, který inkrementuje časovač od hodnoty 0 do hodnoty uložené v porovnávacím registru, kdy vyvolá přerušení a vynuluje časovač. Je vhodný pro generování časového intervalu (při 8 MHz krystalu až 8s).

10.1 Módy časovačů

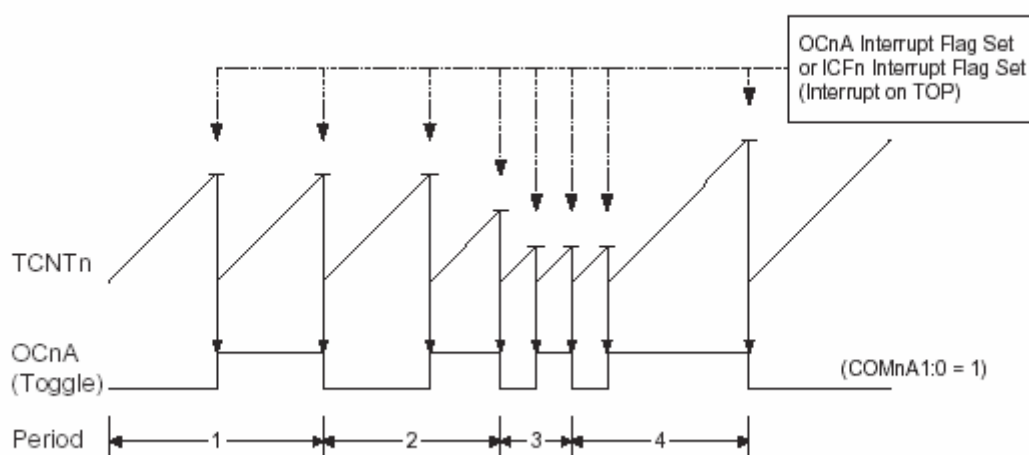
10.1.1 Normal mode

Při tomto módu čítá vždy nahoru, při přetečení (FFFF→0, FF→0) nastavuje příznak a může být vyvoláno přerušení. Čítací registr časovače TCNT může být kdykoli přednastaven.

10.1.2 CTC mode

Čítač čítá vždy nahoru a při shodě čítacího registru TCNT s komparačním registrem OCR dojde k vynulování čítacího registru TCNT, nastavení příznaku a může být vyvoláno přerušení.

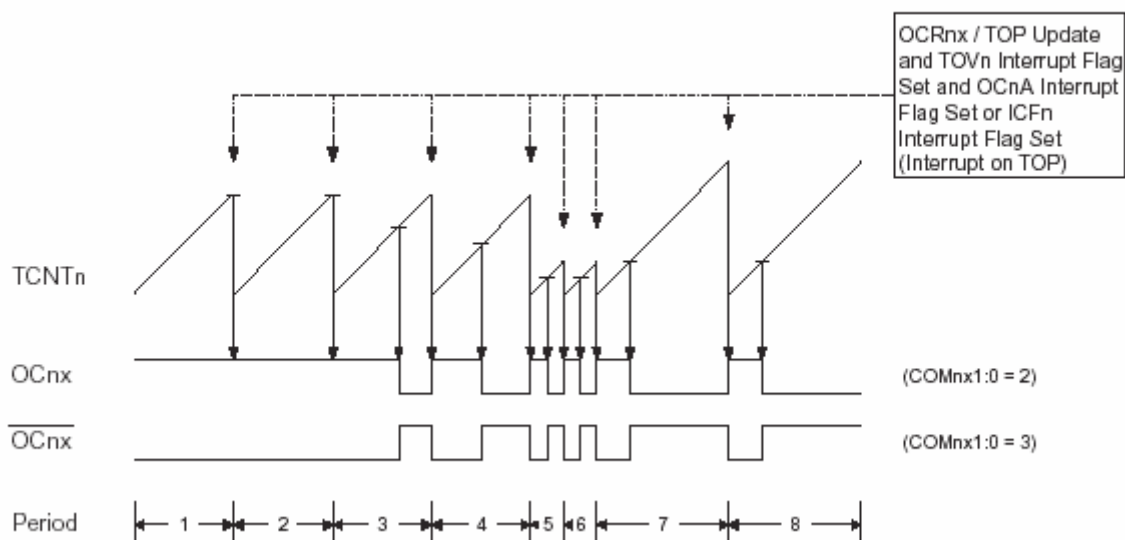
Figure 51. CTC Mode, Timing Diagram



10.1.3 Fast PWM mode

Čítač čítá vždy nahoru do maximálnej hodnoty (0FF, 1FF, 3FF, registry OCR, ICR), poté přeteče do nuly. Přitom se nastaví příznak a může být vyvoláno přerušení. Bit OC se při shodě s registrem OC nuluje a při přetečení nastaví do log1. Tento bit je možné připojit na výstup mikroprocesoru a tím vygenerovat pulzně-šířkově modulovaný signál.

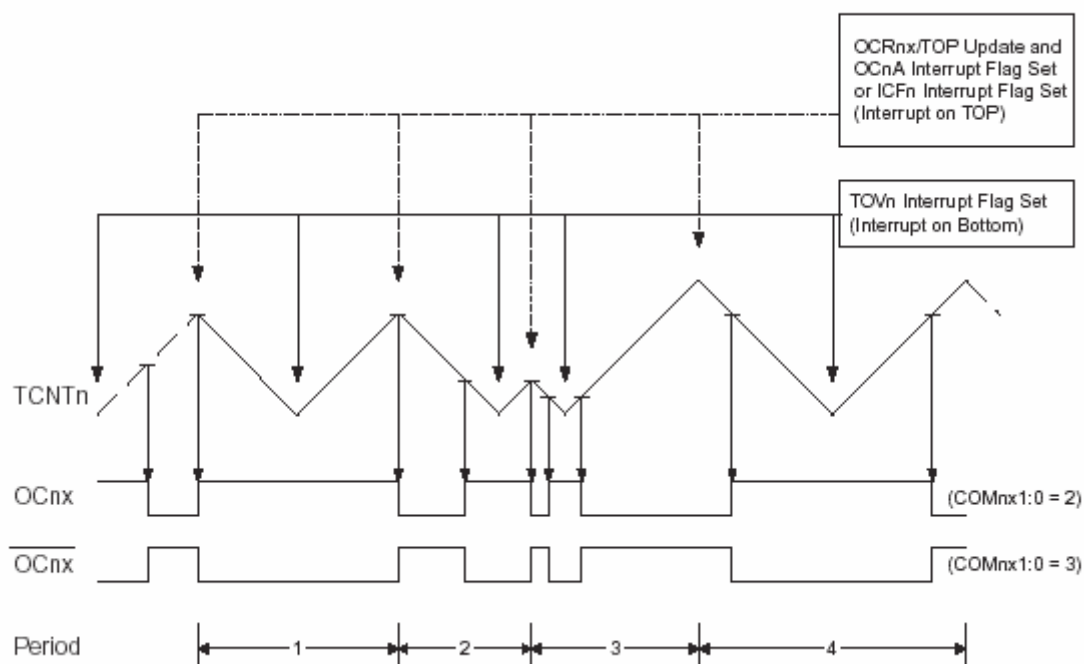
Figure 52. Fast PWM Mode, Timing Diagram



10.1.4 Phase Correct PWM mode

Čítač čítá nahoru do maximálnej hodnoty (0FF, 1FF, 3FF, registry OCR, ICR), poté čítá dolů do hodnoty 0. Přitom se nastaví příznak a může být vyvoláno přerušení. Bit OC se při shodě s registrem OCR nuluje a při přetečení nastaví do log1. Tento bit je možné připojit na výstup mikroprocesoru a tím vygenerovat pulzně-šířkově modulovaný signál.

Figure 53. Phase Correct PWM Mode, Timing Diagram



10.1.5 Záchytný režim

Tento režim umožňuje zapsat aktuální stav čítacího registru TCNT do záchytného registru ICR při zvolené události (sestupná hrana na vnějším vstupu, nástupná hrana na vnějším vstupu, překlopení analogového komparátoru).

10.2 Časovač 0

Je to osmibitový časovač, nastavení fce se provádí pomocí bitů reg. TCCR0

Timer/Counter Control
Register – TCCR0

Bit	7	6	5	4	3	2	1	0	
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Nastavení módu určují bity WGM01, WGM00:

Table 47. Waveform Generation Mode Bit Description⁽¹⁾

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0 at	TOV0 Flag Set on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	TOP	MAX

Bity CS01-3 slouží k nastavení předděličky a zdroje signálu (vnitřní osc/x, vnější hodiny nástupná nebo sestupná hrana) :

Table 51. Clock Select Bit Description

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	$\text{clk}_{\text{I/O}}$ /(No prescaling)
0	1	0	$\text{clk}_{\text{I/O}}/8$ (From prescaler)
0	1	1	$\text{clk}_{\text{I/O}}/64$ (From prescaler)
1	0	0	$\text{clk}_{\text{I/O}}/256$ (From prescaler)
1	0	1	$\text{clk}_{\text{I/O}}/1024$ (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

Bity COM01, COM00 slouží k nastavení reakce výstupního bitu OC0 (PB0) na přetečení, shodu TCNT0 s OCR0. Tyto reakce jsou různé pro jednotlivé módy, ale při nastavení COM01, COM00 do log0 nebude OC0 reagovat vůbec, PB0 je normální bit portu. Ostatní v katalogovém listu.

Čítací registr TCNT0:

Timer/Counter Register – TCNT0

Bit	7	6	5	4	3	2	1	0	
	TCNT0[7:0]								TCNT0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Komparační registr OCR0:

Output Compare Register – OCR0

Bit	7	6	5	4	3	2	1	0	
	OCR0[7:0]								OCR0
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Registr povolení přerušení:

Timer/Counter Interrupt Mask Register – TIMSK

Bit	7	6	5	4	3	2	1	0	
	TOIE1	OCIE1A	OCIE1B	OCIE2	TICIE1	TOIE2	TOIE0	OCIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit TOIE0 – povolení přerušení při přetečení

Bit OCIE0 – povolení přerušení při rovnosti TCNT0 a OCR0

Registr příznaků přerušení:

Timer/Counter Interrupt Flag Register – TIFR

Bit	7	6	5	4	3	2	1	0	
	TOV1	OCF1A	OCF1B	OCF2	ICF1	TOV2	TOV0	OCF0	TIFR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

10.2.1 Nastavení nejpoužívanějšího módu CTC:

Bit	7	6	5	4	3	2	1	0	
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	

TCCR0 – 00001XXX - bity XXX nastavíme dělicí poměr dle tabulky 57

Hodnotu při které se bude čítač nulovat nastavíme do registru OCR0. Výsledná frekvence pak bude:

$$f_{OCn} = \frac{f_{clk\ I/O}}{2 \cdot N \cdot (1 + OCRn)}$$

- kde
- N je dělicí poměr predděličky
 - f_{CLK} je frekvence oscilátoru
 - OCRn je komparační registr OCR0

Pro povolení přerušení je nutné nastavit bit I v SREG a OCIE0 v reg. TIMSK.
Vektor přerušení je 0x0020 (OC0addr).

10.3 Časovač 1

Je to šestnáctibitový časovač, nastavení fce se provádí pomocí bitů reg. TCCR1A a TCCR1B.

**Timer/Counter1 Control
Register A – TCCR1A**

Bit	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	W	W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

**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	

Nastavení módu určují bity WGM10-3 :

Table 56. Waveform Generation Mode Bit Description⁽¹⁾

Mode	WGMn3	WGMn2 (CTCn)	WGMn1 (PWMn1)	WGMn0 (PWMn0)	Timer/Counter Mode of Operation	TOP	Update of OCRnX at	TOVn Flag Set on
0	0	0	0	0	Normal	0xFFFF	Immediate	MAX
1	0	0	0	1	PWM, Phase Correct, 8-bit	0x00FF	TOP	BOTTOM
2	0	0	1	0	PWM, Phase Correct, 9-bit	0x01FF	TOP	BOTTOM
3	0	0	1	1	PWM, Phase Correct, 10-bit	0x03FF	TOP	BOTTOM
4	0	1	0	0	CTC	OCRnA	Immediate	MAX
5	0	1	0	1	Fast PWM, 8-bit	0x00FF	TOP	TOP
6	0	1	1	0	Fast PWM, 9-bit	0x01FF	TOP	TOP
7	0	1	1	1	Fast PWM, 10-bit	0x03FF	TOP	TOP
8	1	0	0	0	PWM, Phase and Frequency Correct	ICRn	BOTTOM	BOTTOM
9	1	0	0	1	PWM, Phase and Frequency Correct	OCRnA	BOTTOM	BOTTOM
10	1	0	1	0	PWM, Phase Correct	ICRn	TOP	BOTTOM
11	1	0	1	1	PWM, Phase Correct	OCRnA	TOP	BOTTOM
12	1	1	0	0	CTC	ICRn	Immediate	MAX
13	1	1	0	1	Reserved	–	–	–
14	1	1	1	0	Fast PWM	ICRn	TOP	TOP
15	1	1	1	1	Fast PWM	OCRnA	TOP	TOP

Note: 1. The CTCn and PWMn1:0 bit definition names are obsolete. Use the WGMn2:0 definitions. However, the functionality and location of these bits are compatible with previous versions of the timer.

Bits CS11-3 slouží k nastavení předděličky a zdroje signálu (vnitřní osc/x, vnější hodiny nástupná nebo sestupná hrana) :

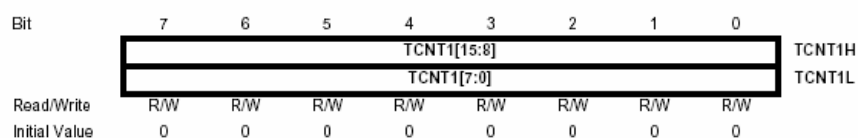
Table 57. Clock Select Bit Description Timer/Counter1

CS12	CS11	CS10	Description
0	0	0	No clock source. (Timer/Counter stopped).
0	0	1	$\text{clk}_{\text{I/O}}/1$ (No prescaling)
0	1	0	$\text{clk}_{\text{I/O}}/8$ (From prescaler)
0	1	1	$\text{clk}_{\text{I/O}}/64$ (From prescaler)
1	0	0	$\text{clk}_{\text{I/O}}/256$ (From prescaler)
1	0	1	$\text{clk}_{\text{I/O}}/1024$ (From prescaler)
1	1	0	External clock source on T1 pin. Clock on falling edge.
1	1	1	External clock source on T1 pin. Clock on rising edge.

Bity COM1A1-0 slouží k nastavení reakce výstupního bitu OC1A (PD5) na přetečení, shodu TCNT1 s OCR1A. Tyto reakce jsou různé pro jednotlivé módy, ale při nastavení COM1A1, COM1A0 do log0 nebude OC1A reagovat vůbec, PD5 je normální bit portu. Ostatní v katalogovém listu. Totéž platí pro bity COM1B1-0, výstupní bit OC1B(PE2).

Čítací registr :

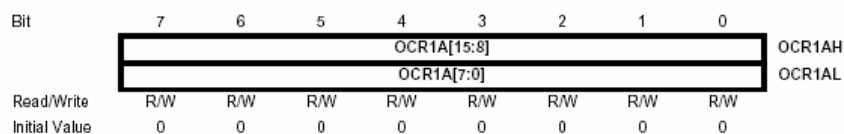
Timer/Counter1 – TCNT1H and TCNT1L



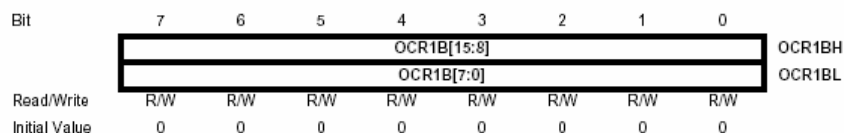
Komparační registry:

Dosáhne-li TCNT1 hodnoty OCR1A,B je nastaven příznak a může být vyvoláno přerušení.

Output Compare Register 1 A – OCR1AH and OCR1AL



Output Compare Register 1 B – OCR1BH and OCR1BL



Záchytné registry:

Při události na bitu ICP1 a na výstupu analogového komparátoru dojde k zachycení aktuální hodnoty TCNT1.

Input Capture Register 1 – ICR1H and ICR1L

Bit	7	6	5	4	3	2	1	0	
	ICR1[15:8]								ICR1H
	ICR1[7:0]								ICR1L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Input Capture Register 3 – ICR3H and ICR3L

Bit	7	6	5	4	3	2	1	0	
	ICR3[15:8]								ICR3H
	ICR3[7:0]								ICR3L
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Timer/Counter Interrupt Mask Register – TIMSK⁽¹⁾

Bit	7	6	5	4	3	2	1	0	
	TOIE1	OCIE1A	OCIE1B	OCIE2	TICIE1	TOIE2	TOIE0	OCIE0	TIMSK
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

10.3.1 Nastavení nepoužívanějšího módu CTC:

Bit	7	6	5	4	3	2	1	0	
	COM1A1	COM1A0	COM1B1	COM1B0	FOC1A	FOC1B	WGM11	WGM10	TCCR1A
Read/Write	R/W	R/W	R/W	R/W	W	W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
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	

TCCR1A – 00000000

TCCR1B – 00001XXX - bity XXX nastavíme dělicí poměr dle tabulky 51

Hodnotu při které se bude čítač nulovat nastavíme do registru OCR1. Výsledná frekvence pak bude:

$$f_{OCn} = \frac{f_{clk \ I/O}}{2 \cdot N \cdot (1 + OCRn)}$$

- kde
- N je dělicí poměr predděličky
 - f_{CLK} je frekvence oscilátoru
 - $OCRn$ je číslo v komparačních registrech OCR1A

Pro povolení přerušení je nutné nastavit bit I v SREG a OCIE1A v reg. TIMSK.
Vektor přerušení je 0x001A (*OC1addr*).

10.4 Časovač 3

Časovač 3 pracuje stejně jako časovač1, liší se pouze v registrech a bitech (místo čísla 1 mají číslo 3) a neumí čítat vnější frekvenci, takže tabulka nastavení předděličky a zdroje signálu je následující.

Table 58. Clock Select Bit Description Timer/Counter3

CS32	CS31	CS30	Description
0	0	0	No clock source. (Timer/Counter stopped).
0	0	1	$\text{clk}_{\text{I/O}} / 1$ (No prescaling)
0	1	0	$\text{clk}_{\text{I/O}} / 8$ (From prescaler).
0	1	1	$\text{clk}_{\text{I/O}} / 64$ (From prescaler).
1	0	0	$\text{clk}_{\text{I/O}} / 256$ (From prescaler).
1	0	1	$\text{clk}_{\text{I/O}} / 1024$ (From prescaler).
1	1	0	$\text{clk}_{\text{I/O}} / 16$ (From prescaler).
1	1	1	$\text{clk}_{\text{I/O}} / 32$ (From prescaler).

10.4.1 Nastavení nejpoužívanějšího módu CTC:

Bit	7	6	5	4	3	2	1	0	
	COM3A1	COM3A0	COM3B1	COM3B0	FOC3A	FOC3B	WGM31	WGM30	TCCR3A
Read/Write	R/W	R/W	R/W	R/W	W	W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit	7	6	5	4	3	2	1	0	
	ICNC3	ICES3	–	WGM33	WGM32	CS32	CS31	CS30	TCCR3B
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	

TCCR3A – 00000000

TCCR3B – 00001XXX - bity XXX nastavíme dělicí poměr dle tabulky 51

Hodnotu při které se bude čítač nulovat nastavíme do registru OCR1. Výsledná frekvence pak bude:

$$f_{OCn} = \frac{f_{\text{clk I/O}}}{2 \cdot N \cdot (1 + OCRn)}$$

- kde
- N je dělicí poměr předděličky
 - f_{CLK} je frekvence oscilátoru
 - OCRn je číslo v komparačních registrech OCR3A

Pro povolení přerušení je nutné nastavit bit I v SREG a OCIE3A v reg. ETIMSK.
Vektor přerušení je 0x000E (*OC3addr*).

Extended Timer/Counter
Interrupt Mask Register –
ETIMSK⁽¹⁾

Bit	7	6	5	4	3	2	1	0	
			TICIE3	OCIE3A	OCIE3B	TOIE3	–	–	ETIMSK
Read/Write	R	R	R/W	R/W	R/W	R/W	R	R	
Initial Value	0	0	0	0	0	0	0	0	

10.5 Časovač 2

Časovač 2 je osmibitový časovač, který je možno použít stejným způsobem jako časovač 0, případně jako časovač pro reálný čas, kdy pracuje s vnějším krystalem 32 kHz.

10.5.1 Nastavení nejpoužívanějšího módu CTC:

Timer/Counter Control
Register – TCCR2

Bit	7	6	5	4	3	2	1	0	
	FOC2	WGM20	COM21	COM20	WGM21	CS22	CS21	CS20	TCCR2
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

TCCR0 – 00001XXX - bity XXX nastavíme dělicí poměr dle tabulky 64

Hodnotu při které se bude čítač nulovat nastavíme do registru OCR2. Výsledná frekvence pak bude:

$$f_{OCn} = \frac{f_{clk\ I/O}}{2 \cdot N \cdot (1 + OCRn)}$$

- kde
- N je dělicí poměr předděličky
 - f_{CLK} je frekvence oscilátoru
 - OCRn je komparační registr OCR2

Pro povolení přerušení je nutné nastavit bit I v SREG a OCIE0 v reg. TIMSK.
Vektor přerušení je 0x0014 (*OC2addr*).

10.5.2 Nastavení módu Fast PWM pro regulaci jasu displeje


```

LDI    R16,0B01101100 ; nastavení PWM pro timer2 - výstupní bit připojený k
OUT    TCCR2,R16      ; disp- EN

LDI    R16,255
OUT    OCR2,R16      ; nastavení porovnávacího registru

LDI    R20,0          ; do R20 číslo 0 ( - na displeji)

;*****hlavní program *****

LOOP:  LDS    R17,POC          ; v počítadle je hodnota 0-255 kterou zapisujeme do
DEC    R17                  ; porovnávacího registru
STS    POC,R17
OUT    OCR2,R17

STS    OP0,R17              ; převod 1 byte do BCD
RCALL  BINBCD1

LDS    R16,OP0
STS    SEG0,R16

LDS    R16,OP1
STS    SEG1,R16

LDS    R16,OP2
STS    SEG2,R16

KINC:  RCALL  DISP595
CALL  DELAY
RJMP  LOOP

```

10.5.3 Nastavení záchytného režimu pro měření časového intervalu

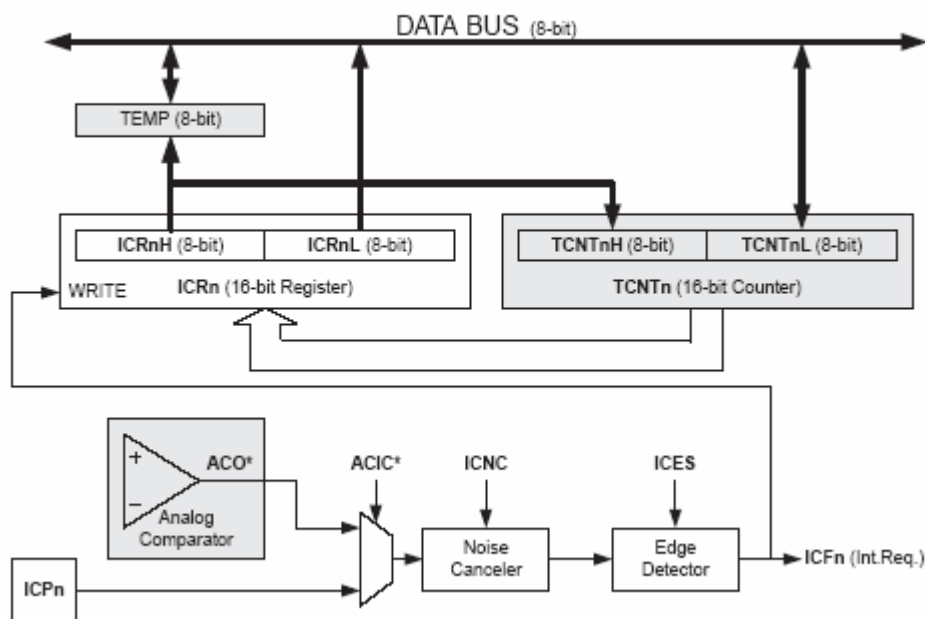
Na obr.48 je schéma popisující funkci čítače časovače v záchytném režimu. Při vzniku události (sestupná nebo nástupná hrana na bitu portu, nebo u čítače/časovače 1 překlopení analogového komparátoru) dojde k zapsání aktuálního stavu čítacích registrů do registrů záchytných. Toto můžeme využít k měření časového intervalu mezi dvěma hranami signálu. Při první hraně vynulujeme čítací registry TCNT (nejdříve nulujeme vyšší byt a potom nižší – viz [Čtení a zápis 16-ti bitových registrů](#)). Při hraně iniciující funkci zachycení dojde k přepisu čítacího registru TCNT do záchytného registru ICR. Zároveň dojde k vyvolání přerušení, při kterém zpracujeme hodnotu ze záchytného registru ICR. Výhoda automatického zachycení čítané hodnoty spočívá v tom, že nastane právě v okamžik hrany a nezáleží na případném zpoždění vyvolání přerušení.

Měření časového intervalu můžeme provádět i mezi dvěma různými hranami – měření doby log.0 nebo log.1. Potom musíme nulování provádět při jednom typu hrany a zachycení při druhém typu.

Frekvenci čítání můžeme měnit dělením frekvence oscilátoru v předděliči čítače ($f_{osc}/1-f_{osc}/1024$). Tím můžeme měřit časový interval řádu mikrosekund až desítek sekund. Výsledek je nutné přepočítat, protože tvar výsledku je závislý na dělicím poměru.

Mód časovače zvolíme normál (0-0FFFF), můžeme zapnout filtr pro spouštěcí pin, který vyžaduje impuls na pinu v trvání minimálně čtyř strojových cyklů aby byla spuštěna funkce zachycení.

Figure 48. Input Capture Unit Block Diagram⁽¹⁾



Bit	7	6	5	4	3	2	1	0	
	COM3A1	COM3A0	COM3B1	COM3B0	FOC3A	FOC3B	WGM31	WGM30	TCCR3A
Read/Write	R/W	R/W	R/W	R/W	W	W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit	7	6	5	4	3	2	1	0	
	ICNC3	ICES3	—	WGM33	WGM32	CS32	CS31	CS30	TCCR3B
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	

TCCR3A – 00000000

TCCR3B – 10000XXX - bity XXX nastavíme dělicí poměr dle tabulky 51, zapneme filtr, sestupná hrana, mód časovače normal

.CSEG

JMP INIT

```

        .ORG    ICP3addr
        JMP     ICEVENT

ICEVENT:    LDI     R16,0
            STS     TCNT3H,R16    ; nulování čítacích registrů
            STS     TCNT3L,R16

            LDS     R16,ICR3L    ; přepis záchytných registrů
            LDS     R17,ICR3H

            *
            *
            *

            RETI

;*****

INIT:      *
            *
            *

            LDI     R16,0B00000000    ;režim normal
            STS     TCCR3A,R16

            LDI     R16,0B10000101    ;filtr zap, sest. hrana, dělení fosc/1024
            STS     TCCR3B,R16

            LDI     R16,0B00100000    ; povolení přerušení časovače od záchytné události
            STS     ETIMSK,R16

            SEI                      ; globální povolení      přerušení

            *
            *
            *

```

11. Sériový kanál USART

Mikroprocesor obsahuje dva stejné kanály plně duplexní sériové komunikace USART0 a USART1, které mohou pracovat ve dvou základních režimech:

- v synchronní režim (master nebo slave) - bity TxD – vysílací bit, RxD – přijímací bit a XCK hodiny (master režim vysílá hodiny, slave režim přijímá hodiny)
- asynchronní režim - bity TxD – vysílací bit, RxD – přijímací bit

Pro vysílání a přijímání je nutné toto povolit bity RXEN a TXEN, což má za následek zakázání normální činnosti pinů RxD a TxD. Nepovolení má za následek nepřijetí dat a nastavení chybových bitů FE, DOR a UPE.

Přijímací registr je tvořen dvěma registry:

- přijímací registr UDR ke kterému máme přístup a můžeme ho číst
- sériový přijímací registr do kterého se data během příjmu posouvají

Do registru UDR se data přepisují v okamžiku, kdy přijmeme celý byt a procesor detekuje startbit dalšího bytu. Přijmeme-li první byt, při startbitu druhého bytu se přepíše do UDR. Potom přijmeme celý druhý byt, který se přepíše do UDR při detekování startbitu třetího bytu. Do té doby musíme mít z registru UDR přečten první byt, jinak se přepíše druhým bytem. V tomto případě by se nastavil chybový příznak DOR.

Při vysílání nebo přijímání dochází k nastavení různých příznaků podle prováděné činnosti a tyto příznaky mohou vyvolat přerušení:

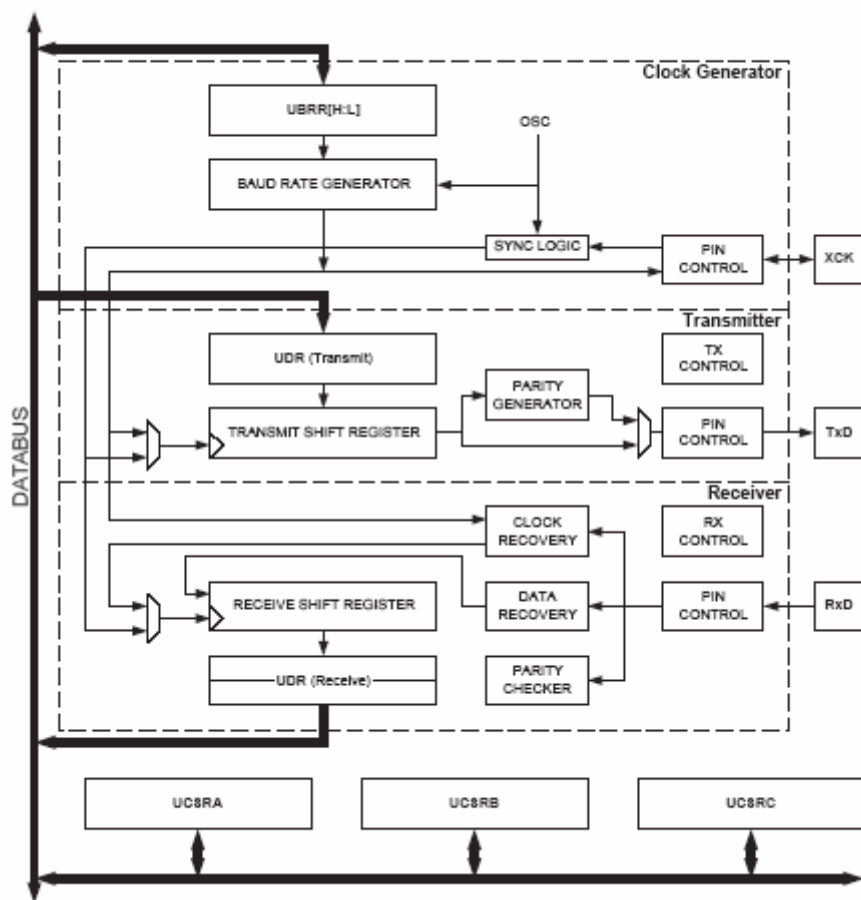
1. RXC – kompletní příjem
2. TXC – kompletní vyslání
3. UDRE – vysílací registr prázdný – připravený pro zápis nových dat

Dále dochází k nastavení chybových příznaků:

1. FE – stopbit byl přijat jako log0
2. DOR – přijímací registr nebyl přečten, druhý sériový přijímací registr je plný a byl detekován další startbit.
3. UPE – chyba parity

Vysílání začíná zápisem do registru UDR. Po přijetí dat se data objeví také v registru UDR, který má stejnou adresu, ale fyzicky jsou to dva registry RXB a TXB.

Figure 75. USART Block Diagram⁽¹⁾



11.1 Synchronní režim

Synchronní režim nastavíme bitem UMSEL do log1. Využívá I/O bity RxD – přijímací bit, TxD – vysílací bit a XCK synchronizační hodinový signál.

Hodinový signál může pracovat ve dvou režimech nastavením bitu UCPOL:

1. UCPOL=log1, data jsou čtena při nástupné hraně a při sestupné hraně nastává výměna dat.
2. UCPOL=log0, data jsou čtena při sestupné hraně a při nástupné hraně nastává výměna dat.

**USART Control and Status
Register A – UCSRA**

Bit	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	UPE	U2X	MPCM	UCSRA
Read/Write	R	R/W	R	R	R	R	R/W	R/W	
Initial Value	0	0	1	0	0	0	0	0	

Bit 7 – RXC – nastaví se při dokončení příjmu a nuluje při čtení přijatých dat z reg. UDR instrukcí `IN Rd,UDR0`. Může generovat přerušení.

Bit 6 – TXC – nastaví se při dokončení vysílání (při vyprázdnění sériového vysílacího registru) pokud nejsou v reg UDR zapsaná další data. Nuluje se automaticky při vyvolání přerušení nebo programově. Může generovat přerušení.

Bit 5 – UDRE – nastaví se je-li vysílací reg. UDR prázdný (data jsou přesunuta do sériového vysílacího reg.), připravený na zápis nových dat. Může generovat přerušení.

Bit 4 – FE chybový příznak - nastaví se v případě, že stopbit byl přijat jako log0

Bit 3 – DOR chybový příznak – nastaví se v případě, kdy přijímací registr nebyl přečten, druhý sériový přijímací registr je plný a byl detekován další startbit

Bit 2 – UPE chybový příznak – nastaví se v případě, že došlo k chybě parity

Bit 1 – využívá se při asynchronní komunikaci (při synchronní vždy log0) log1 nastaví dvojnásobnou přenosovou rychlost.

Bit 0 – MPCM – Povoluje multiprocessorovou komunikaci

**USART Control and Status
Register B – UCSRB**

Bit	7	6	5	4	3	2	1	0	
	RXCIE	TXCIE	UDRIE	RXEN	TXEN	UCSZ2	RXB8	TXB8	UCSRB
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Bit 7 – RXCIE - povolení přerušení přijímače

Bit 6 – TXCIE - povolení přerušení vysílače

Bit 5 – UDRIE - povolení přerušení vyprázdnění reg UDR

Bit 4 – RXEN - povolení příjmu – zakáže normální fci portu

Bit 3 – TXEN - povolení vysílání - zakáže normální fci portu

Bit 2 – UCSZ2 – společně s bity UCSZ1,0 nastavuje formát dat – viz tab. 76

Bit 1 – RXB8 - devátý bit přijímacího reg.

Bit 0 – TXB8 - devátý bit vysílacího reg.

Table 76. UCSZ Bits Settings

UCSZ2	UCSZ1	UCSZ0	Character Size
0	0	0	5-bit
0	0	1	6-bit
0	1	0	7-bit
0	1	1	8-bit
1	0	0	Reserved
1	0	1	Reserved
1	1	0	Reserved
1	1	1	9-bit

USART Control and Status Register C – UCSRC⁽¹⁾

Bit	7	6	5	4	3	2	1	0	
	URSEL	UMSEL	UPM1	UPM0	USBS	UCSZ1	UCSZ0	UCPOL	UCSRC
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	0	0	0	0	1	1	0	

Bit 7 – URSEL slouží k výběru registru UBRR/UCSRC. Tyto dva registry mají stejnou adresu a k rozlišení se kterým komunikujeme musíme dodržet následující pravidla:

- je-li bit URSEL = log0 – následujících 7 bitů se zapíše do reg. UBRRH
- je-li bit URSEL = log1 – následujících 7 bitů se zapíše do reg. UCSRC
- nebyl-li v předchozím strojovém cyklu čten tento reg., čteme UBRRH
- byl-li v předchozím strojovém cyklu čten tento reg., čteme UCSRC

Bit 6 – UMSEL – volí asynchronní nebo synchronní komunikaci

Table 73. UMSEL Bit Settings

UMSEL	Mode
0	Asynchronous Operation
1	Synchronous Operation

Bit 5 – UPM1

Bit 4 – UPM0 – volí mód parity

Table 74. UPM Bits Settings

UPM1	UPM0	Parity Mode
0	0	Disabled
0	1	Reserved
1	0	Enabled, Even Parity
1	1	Enabled, Odd Parity

Bit 3 – USBS - volí počet stopbitů

Table 75. USBS Bit Settings

USBS	Stop Bit(s)
0	1-bit
1	2-bit

Bit 2 – UCSZ1

Bit 1 – UCSZ0 viz tab. 76

Bit 0 – UCPOL - volí polaritu hodinového impulsu pro synchronní režim

Table 77. UCPOL Bit Settings

UCPOL	Transmitted Data Changed (Output of TxD Pin)	Received Data Sampled (Input on RxD Pin)
0	Rising XCK Edge	Falling XCK Edge
1	Falling XCK Edge	Rising XCK Edge

**USART Baud Rate Registers –
UBRRL and UBRRH⁽¹⁾**

Bit	15	14	13	12	11	10	9	8	
	URSEL	–	–	–	UBRR[11:8]				UBRRH
	UBRR[7:0]								UBRRL
	7	6	5	4	3	2	1	0	
Read/Write	R/W	R	R	R	R/W	R/W	R/W	R/W	
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	
	0	0	0	0	0	0	0	0	

Registr UBRR nastavuje přenosovou rychlost (bity 11÷0) Bit URSEL slouží k výběru registru UBRR/UCSRC. Tyto dva registry mají stejnou adresu a k rozlišení se kterým komunikujeme musíme dodržet následující pravidla:

- je-li bit URSEL = log0 – následujících 7 bitů se zapíše do reg. UBRRH
- je-li bit URSEL = log1 – následujících 7 bitů se zapíše do reg. UCSRC
- nebyl-li v předchozím strojovém cyklu čten tento reg., čteme UBRH
- byl-li v předchozím strojovém cyklu čten tento reg., čteme UCSRC

Table 78. Examples of UBRR Settings for Commonly Used Oscillator Frequencies

Baud Rate (bps)	$f_{osc} = 1.0000 \text{ MHz}$				$f_{osc} = 1.8432 \text{ MHz}$				$f_{osc} = 2.0000 \text{ MHz}$			
	U2X = 0		U2X = 1		U2X = 0		U2X = 1		U2X = 0		U2X = 1	
	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error
2400	25	0.2%	51	0.2%	47	0.0%	95	0.0%	51	0.2%	103	0.2%
4800	12	0.2%	25	0.2%	23	0.0%	47	0.0%	25	0.2%	51	0.2%
9600	6	-7.0%	12	0.2%	11	0.0%	23	0.0%	12	0.2%	25	0.2%
14.4k	3	8.5%	8	-3.5%	7	0.0%	15	0.0%	8	-3.5%	16	2.1%
19.2k	2	8.5%	6	-7.0%	5	0.0%	11	0.0%	6	-7.0%	12	0.2%
28.8k	1	8.5%	3	8.5%	3	0.0%	7	0.0%	3	8.5%	8	-3.5%
38.4k	1	-18.6%	2	8.5%	2	0.0%	5	0.0%	2	8.5%	6	-7.0%
57.6k	0	8.5%	1	8.5%	1	0.0%	3	0.0%	1	8.5%	3	8.5%
76.8k	–	–	1	-18.6%	1	-25.0%	2	0.0%	1	-18.6%	2	8.5%
115.2k	–	–	0	8.5%	0	0.0%	1	0.0%	0	8.5%	1	8.5%
230.4k	–	–	–	–	–	–	0	0.0%	–	–	–	–
250k	–	–	–	–	–	–	–	–	–	–	0	0.0%
Max. ⁽¹⁾	62.5 kbps		125 kbps		115.2 kbps		230.4 kbps		125 kbps		250 kbps	

1. UBRR = 0, Error = 0.0%

Table 80. Examples of UBRR Settings for Commonly Used Oscillator Frequencies (Continued)

Baud Rate (bps)	$f_{osc} = 8.0000 \text{ MHz}$				$f_{osc} = 11.0592 \text{ MHz}$				$f_{osc} = 14.7456 \text{ MHz}$			
	U2X = 0		U2X = 1		U2X = 0		U2X = 1		U2X = 0		U2X = 1	
	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error	UBRR	Error
2400	207	0.2%	416	-0.1%	287	0.0%	575	0.0%	383	0.0%	767	0.0%
4800	103	0.2%	207	0.2%	143	0.0%	287	0.0%	191	0.0%	383	0.0%
9600	51	0.2%	103	0.2%	71	0.0%	143	0.0%	95	0.0%	191	0.0%
14.4k	34	-0.8%	68	0.6%	47	0.0%	95	0.0%	63	0.0%	127	0.0%
19.2k	25	0.2%	51	0.2%	35	0.0%	71	0.0%	47	0.0%	95	0.0%
28.8k	16	2.1%	34	-0.8%	23	0.0%	47	0.0%	31	0.0%	63	0.0%
38.4k	12	0.2%	25	0.2%	17	0.0%	35	0.0%	23	0.0%	47	0.0%
57.6k	8	-3.5%	16	2.1%	11	0.0%	23	0.0%	15	0.0%	31	0.0%
76.8k	6	-7.0%	12	0.2%	8	0.0%	17	0.0%	11	0.0%	23	0.0%
115.2k	3	8.5%	8	-3.5%	5	0.0%	11	0.0%	7	0.0%	15	0.0%
230.4k	1	8.5%	3	8.5%	2	0.0%	5	0.0%	3	0.0%	7	0.0%
250k	1	0.0%	3	0.0%	2	-7.8%	5	-7.8%	3	-7.8%	6	5.3%
0.5M	0	0.0%	1	0.0%	–	–	2	-7.8%	1	-7.8%	3	-7.8%
1M	–	–	0	0.0%	–	–	–	–	0	-7.8%	1	-7.8%
Max. ⁽¹⁾	0.5 Mbps		1 Mbps		691.2 kbps		1.3824 Mbps		921.6 kbps		1.8432 Mbps	

1. UBRR = 0, Error = 0.0%

11.4 Příklady nastavení registrů USART

Toto je nastavení pro 9600 bitů/s při 8MHz frekvence oscilátoru – UBR – 51 a bit UX2 v UCSRA je v log. 0. V UCSRB nastavíme bit 3 a 4 nastavíme do log.1 pro povolení vysílání a příjmu – zakáže normální fci portu. Přerušení je zakázané, vysíláme 8 bitů dat.

```
LDI R16,0B00000000 ; příznaky , bit1 U2X přenos. rychlost
OUT UCSRA,R16

LDI R16,0B00011000 ; povolení příjmu a vysílání
OUT UCSRB,R16

LDI R16,0B10000110 ; 8 bitů dat
OUT UCSRC,R16

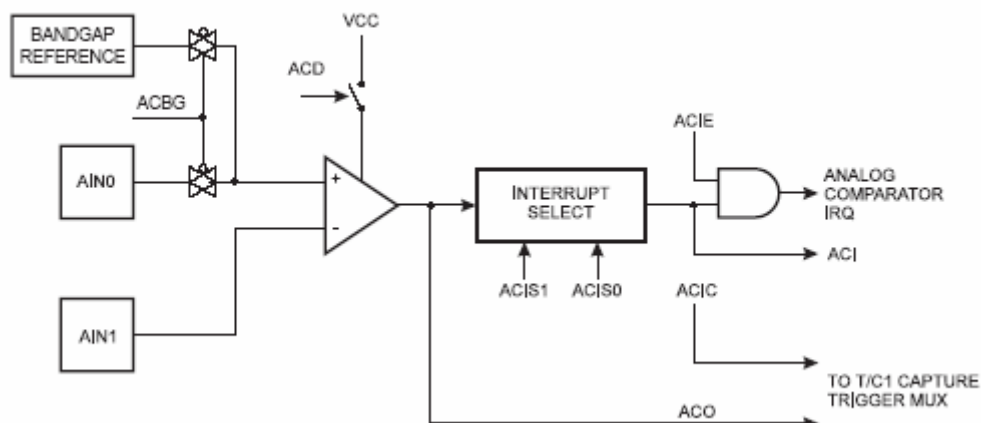
LDI R16,LO(51) ;nastavení přenosové rychlosti 9600 pro 8 Mhz
OUT UBRR0L,R16

LDI R16,HI(51)
OUT UBRR0H,R16
```

Pro použití přerušení (pro příjem) musíme povolit přerušení v reg. UCSRB bit 7 a na adresu vektoru přerušení 0x026 napsat skok na podprogram pro obsluhu přerušení od příjmu sériového kanálu.

12. Analogový komparátor

Figure 82. Analog Comparator Block Diagram⁽¹⁾



Analogový komparátor porovnává napětí na vstupech AIN0 (PB2) nebo Uref a AIN1 (PB3), přičemž vstup AIN1 funguje jako negativní a AIN0 jako pozitivní vstup komparátoru. Výstup komparátoru je vyveden na bit ACO a změna (nástupná, sestupná nebo obě hrany) na tomto bitu může vyvolat přerušení – pokud bude povoleno. Dále může mít za následek zachycení stavu časovače 1 do záchytného registru.

Všechny funkce se nastavují v registru ACSR

Analog Comparator Control and Status Register – ACSR

Bit	7	6	5	4	3	2	1	0	
	ACD	ACBG	ACO	ACI	ACIE	ACIC	ACIS1	ACIS0	ACSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	N/A	0	0	0	0	0	

Bit 7 - ACD – Analog Comparator Disable - je-li nastaven do log.1 je funkce komparátoru zakázána.

Bit 6 - ACBG – Analog Comparator Bandgap Select – je-li nastaven do log.1, je na pozitivní vstup připojen vnitřní ref. zdroj. Je-li v log.0 je na pozitivní vstup připojen vstup AIN0.

Bit 5 - ACO – Analog Comparator Output – výstup analogového komparátoru

Bit 4 – ACI – Analog Comparator Interrupt Flag – příznak přerušení analogového komparátoru. Je nastaven dojde-li k změně na výstupu analogového komparátoru definované bity ACIS1, ACIS0.

Bit 3 – ACIE – Analog Comparator Interrupt Enable – povolení přerušení znakového komparátoru.

Bit 2 - ACIC – Analog Comparator Input Capture – je-li nastaven do log.1 umožňuje zachycení stavu časovače 1 do záchytného registru pokud bude tato funkce nastavena u časovače 1.

Bity 1,0 – ACIS1,0 – bity pro nastavení typu změny na výstupu komparátoru na kterou bude reagovat přerušovací systém.

Table 82. ACIS1/ACIS0 Settings

ACIS1	ACIS0	Interrupt Mode
0	0	Comparator Interrupt on Output Toggle.
0	1	Reserved
1	0	Comparator Interrupt on Falling Output Edge.
1	1	Comparator Interrupt on Rising Output Edge.

Nastavení v programu bez použití přerušení a záchytného režimu časovače:

```
LDI    R16,0B01010000 ; port B jako vstupní kromě PB4 a PB6 - LED
OUT    DDRB,R16        ; nastaví bit0 portu D jako výstupní
OUT    PORTB,R16

LDI    R16,0B00000000 ; nastaví analogový komparátor,zakázané přerušení
OUT    ACSR,R16        ; i záchytný režim - pouze se výstup objeví v bitu ACO

;*****hlavní program*****

ZAC:   SBIC    ACSR,ACO ; test bitu ACO - výstupu komparátoru
       RJMP   JEDNA
       SBI    PORTB,6 ; rozsvítí červenou, zhasne zelenou
       CBI    PORTB,4
       RJMP   ZAC

JEDNA: SBI    PORTB,4 ; zhasne červenou, rozsvítí zelenou
       CBI    PORTB,6
       JMP    ZAC
```

13. Nastavení procesoru pomocí programovacích propojek

Při programování procesoru můžeme nastavit některé funkce procesoru pomocí 20 bitů, které jsou ve třech registrech **Fuse High Byt, Fuse Low Byte, Extended Fuse Byt**.

1. Fuse Low Byte

Table 101. Fuse Low Byte

Fuse Low Byte	Bit no	Description	Default value
CKDIV8 ⁽⁴⁾	7	Divide clock by 8	0 (programmed)
CKOUT ⁽³⁾	6	Clock Output	1 (unprogrammed)
SUT1	5	Select start-up time	1 (unprogrammed) ⁽¹⁾
SUT0	4	Select start-up time	0 (programmed) ⁽¹⁾
CKSEL3	3	Select Clock source	0 (programmed) ⁽²⁾
CKSEL2	2	Select Clock source	0 (programmed) ⁽²⁾
CKSEL1	1	Select Clock source	1 (unprogrammed) ⁽²⁾
CKSEL0	0	Select Clock source	0 (programmed) ⁽²⁾

Bity CKSEL3÷0 slouží k nastavení zdroje hodinového signálu k časování procesoru. Podle následující tabulky můžeme zvolit vnitřní kalibrovaný oscilátor s frekvencí 8MHz (původní nastavení), vnější krystal, vnější nízkofrekvenční krystal, vnější oscilátor.

Table 5. Device Clocking Options Select

Device Clocking Option	CKSEL3..0
External Crystal/Ceramic Resonator	1111 - 1000
External Low-frequency Crystal	0111 - 0100
Calibrated Internal RC Oscillator	0010
External Clock	0000
Reserved	0011, 0001

Bity SUT1,0 slouží k nastavení startovacího času po resetu, režimech snížené spotřeby apod.

Bit CKOUT umožňuje připojit frekvenci oscilátoru na bit PB.0.

Bit CKDIV8 nastaví dělení frekvence oscilátoru 8 (původní nastavení – dělení zapnuto).

To znamená, že původní nastavení je vnitřní oscilátor dělený 8 (frekvence 1MHz), frekvence na výstup zakázaná a startovací čas maximální.

2. Fuse High Byte

Bit JTAGEN povoluje systém JTAG

Bit SPIEN zakáže sériové programování – SPI (původní nastavení povoleno). Po zakázání je možno povolit pouze v paralelním programátoru.

Bit WDTON nastaví Watchdog timer vždy zapnutý (původní nastavení vypnuto).

Bit EESAVE nastaví zachovávání obsahu datové EEPROM při programování (původní nastavení nezachovávat obsah).

Bit BOOTSZ1,0 nastaví prostor paměti programu pro samoprogramování.

Bit BOOTRST nastaví resetovací vektor.

Table 100. Fuse High Byte

Fuse Low Byte	Bit no	Description	Default Value
OCDEN ⁽³⁾	7	Enable OCD	1 (unprogrammed, OCD disabled)
JTAGEN ⁽⁴⁾	6	Enable JTAG	0 (programmed, JTAG enabled)
SPIEN ⁽¹⁾	5	Enable Serial Program and Data Downloading	0 (programmed, SPI prog. enabled)
WDTON	4	Watchdog Timer always on	1 (unprogrammed)
EESAVE	3	EEPROM memory is preserved through the Chip Erase	1 (unprogrammed, EEPROM not preserved)
BOOTSZ1	2	Select Boot Size (see Table 94 for details)	0 (programmed) ⁽²⁾
BOOTSZ0	1	Select Boot Size (see Table 94 for details)	0 (programmed) ⁽²⁾
BOOTRST	0	Select Reset Vector	1 (unprogrammed)

2. Extended Fuse Byte

Bit M161C nastavuje kompatibilitu s typem Atmega161 (původní nastavení nekompatibilní).

Ostatní bity nastavují úroveň napájecího napětí při kterém se vyvolá Brown-out reset.

Table 99. Extended Fuse Byte⁽¹⁾⁽²⁾

Fuse Low Byte	Bit no	Description	Default Value
–	7	–	1
–	6	–	1
–	5	–	1
M161C	4	ATmega161 compatibility mode	1 (unprogrammed)
BODLEVEL2 ⁽²⁾	3	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL1 ⁽²⁾	2	Brown-out Detector trigger level	1 (unprogrammed)
BODLEVEL0 ⁽²⁾	1	Brown-out Detector trigger level	1 (unprogrammed)
–	0	–	1

V programu nastavení provedeme následujícím způsobem:

.LOFUSE 0B11100010

Toto je stejné nastavení jako původní pouze nebude signál dělen 8. Pokud tento příkaz vůbec nepoužijeme, bude platit původní nastavení. Pokud příkaz použijeme a nastavení změníme platí toto změněné nastavení i když příkaz při dalších programováních používat nebudeme.

.HIFUSE 0B11010001

Tímto zakážeme JTAG (blokuje některé bity PORTU C), povolíme SPI, zakážeme WATCDOG, zachovááme obsah datové EEPROM.