



Registry →
mikroprocesoru

R00	\$00	
R01	\$01	
R02	\$02	
...		
R13	\$0D	
R14	\$0E	
R15	\$0F	
R16	\$10	
R17	\$11	
...		
R26	\$1A	X – registr Low
R27	\$1B	X – registr High
R28	\$1C	Y – registr Low
R29	\$1D	Y – registr High
R30	\$1E	Z – registr Low
R31	\$1F	Z – registr High

Aritmetické a logické instrukce

ADD	Rd, Rr	Add two Registers	$Rd \leftarrow Rd + Rr$	Z,C,N,V,H	1
ADC	Rd, Rr	Add with Carry two Registers	$Rd \leftarrow Rd + Rr + C$	Z,C,N,V,H	1
ADIW	RdI, K	Add Immediate to Word	$Rdh:Rdl \leftarrow Rdh:Rdl + K$	Z,C,N,V,S	2
SUB	Rd, Rr	Subtract two Registers	$Rd \leftarrow Rd - Rr$	Z,C,N,V,H	1
SUBI	Rd, K	Subtract Constant from Register	$Rd \leftarrow Rd - K$	Z,C,N,V,H	1
SBC	Rd, Rr	Subtract with Carry two Registers	$Rd \leftarrow Rd - Rr - C$	Z,C,N,V,H	1
SBCI	Rd, K	Subtract with Carry Constant from Rd	$Rd \leftarrow Rd - K - C$	Z,C,N,V,H	1
SBIW	RdI, K	Subtract Immediate from Word	$Rdh:Rdl \leftarrow Rdh:Rdl - K$	Z,C,N,V,S	2
AND	Rd, Rr	Logical AND Registers	$Rd \leftarrow Rd \cdot Rr$	Z,N,V	1
ANDI	Rd, K	Logical AND Register and Constant	$Rd \leftarrow Rd \cdot K$	Z,N,V	1
OR	Rd, Rr	Logical OR Registers	$Rd \leftarrow Rd \vee Rr$	Z,N,V	1
ORI	Rd, K	Logical OR Register and Constant	$Rd \leftarrow Rd \vee K$	Z,N,V	1
EOR	Rd, Rr	Exclusive OR Registers	$Rd \leftarrow Rd \oplus Rr$	Z,N,V	1
COM	Rd	One's Complement	$Rd \leftarrow \$FF - Rd$	Z,C,N,V	1
NEG	Rd	Two's Complement	$Rd \leftarrow \$00 - Rd$	Z,C,N,V,H	1
SBR	Rd, K	Set Bit(s) in Register	$Rd \leftarrow Rd \vee K$	Z,N,V	1
CBR	Rd, K	Clear Bit(s) in Register	$Rd \leftarrow Rd \cdot (\$FF - K)$	Z,N,V	1
INC	Rd	Increment	$Rd \leftarrow Rd + 1$	Z,N,V	1
DEC	Rd	Decrement	$Rd \leftarrow Rd - 1$	Z,N,V	1
TST	Rd	Test for Zero or Minus	$Rd \leftarrow Rd \cdot Rd$	Z,N,V	1
CLR	Rd	Clear Register	$Rd \leftarrow Rd \oplus Rd$	Z,N,V	1
SER	Rd	Set Register	$Rd \leftarrow \$FF$	None	1
MUL	Rd, Rr	Multiply Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULS	Rd, Rr	Multiply Signed	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
MULSU	Rd, Rr	Multiply Signed with Unsigned	$R1:R0 \leftarrow Rd \times Rr$	Z,C	2
FMUL	Rd, Rr	Fractional Multiply Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z,C	2
FMULS	Rd, Rr	Fractional Multiply Signed	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z,C	2
FMULSU	Rd, Rr	Fractional Multiply Signed Unsigned	$R1:R0 \leftarrow (Rd \times Rr) \ll 1$	Z,C	2

Instrukce skoku (Branch)

RJMP	k	Relative Jump	$PC \leftarrow PC + k + 1$	None	2
IJMP		Indirect Jump to (Z)	$PC \leftarrow Z$	None	2
JMP	k	Direct Jump	$PC \leftarrow k$	None	3
RCALL	k	Relative Subroutine Call	$PC \leftarrow PC + k + 1$	None	3
ICALL		Indirect Call to (Z)	$PC \leftarrow Z$	None	3
CALL	k	Direct Subroutine Call	$PC \leftarrow k$	None	4
RET		Subroutine Return	$PC \leftarrow \text{Stack}$	None	4
RETI		Interrupt Return	$PC \leftarrow \text{Stack}$	I	4
CPSE	Rd, Rr	Compare, Skip if Equal ($Rd = Rr$)	$PC \leftarrow PC + 2$ or 3	None	1-3
CP	Rd, Rr	Compare	$Rd - Rr$	Z, N, V, C, H	1
CPC	Rd, Rr	Compare with Carry	$Rd - Rr - C$	Z, N, V, C, H	1
CPI	Rd, K	Compare Register with Immediate	$Rd - K$	Z, N, V, C, H	1
SBRC	Rr, b	Skip if Bit in Rr Cleared	$(Rr(b)=0): PC \leftarrow PC + 2$ or 3	None	1-3
SBRS	Rr, b	Skip if Bit in Rr is Set	$(Rr(b)=1): PC \leftarrow PC + 2$ or 3	None	1-3
SBIC	P, b	Skip if Bit in I/O Rr Cleared	$(P(b)=0): PC \leftarrow PC + 2$ or 3	None	1-3
SBIS	P, b	Skip if Bit in I/O Rr is Set	$(P(b)=1): PC \leftarrow PC + 2$ or 3	None	1-3
BRBS	s, k	Branch if Status Flag Set	$(SREG(s) = 1): PC \leftarrow PC + k + 1$	None	1-2
BRBC	s, k	Branch if Status Flag Cleared	$(SREG(s) = 0): PC \leftarrow PC + k + 1$	None	1-2
BREQ	k	Branch if Equal	$(Z = 1): PC \leftarrow PC + k + 1$	None	1-2
BRNE	k	Branch if Not Equal	$(Z = 0): PC \leftarrow PC + k + 1$	None	1-2
BRCS	k	Branch if Carry Set	$(C = 1): PC \leftarrow PC + k + 1$	None	1-2
BRCC	k	Branch if Carry Cleared	$(C = 0): PC \leftarrow PC + k + 1$	None	1-2
BRSH	k	Branch if Same or Higher	$(C = 0): PC \leftarrow PC + k + 1$	None	1-2
BRLO	k	Branch if Lower	$(C = 1): PC \leftarrow PC + k + 1$	None	1-2
BRMI	k	Branch if Minus	$(N = 1): PC \leftarrow PC + k + 1$	None	1-2
BRPL	k	Branch if Plus	$(N = 0): PC \leftarrow PC + k + 1$	None	1-2
BRGE	k	Branch if Greater or Equal, Signed	$(N \oplus V = 0): PC \leftarrow PC + k + 1$	None	1-2
BRLT	k	Branch if Less Than Zero, Signed	$(N \oplus V = 1): PC \leftarrow PC + k + 1$	None	1-2
BRHS	k	Branch if Half Carry Flag Set	$(H = 1): PC \leftarrow PC + k + 1$	None	1-2
BRHC	k	Branch if Half Carry Flag Cleared	$(H = 0): PC \leftarrow PC + k + 1$	None	1-2
BRTS	k	Branch if T Flag Set	$(T = 1): PC \leftarrow PC + k + 1$	None	1-2
BRTC	k	Branch if T Flag Cleared	$(T = 0): PC \leftarrow PC + k + 1$	None	1-2
BRVS	k	Branch if Overflow Flag is Set	$(V = 1): PC \leftarrow PC + k + 1$	None	1-2
BRVC	k	Branch if Overflow Flag is Cleared	$(V = 0): PC \leftarrow PC + k + 1$	None	1-2
BRIE	k	Branch if Interrupt Enabled	$(I = 1): PC \leftarrow PC + k + 1$	None	1-2
BRID	k	Branch if Interrupt Disabled	$(I = 0): PC \leftarrow PC + k + 1$	None	1-2

Řídící instrukce mikroprocesoru

NOP		No Operation		None	1
SLEEP		Sleep		None	1
WDR		Watchdog Reset		None	1
BREAK		Break For		None	N

Instrukce přesunů

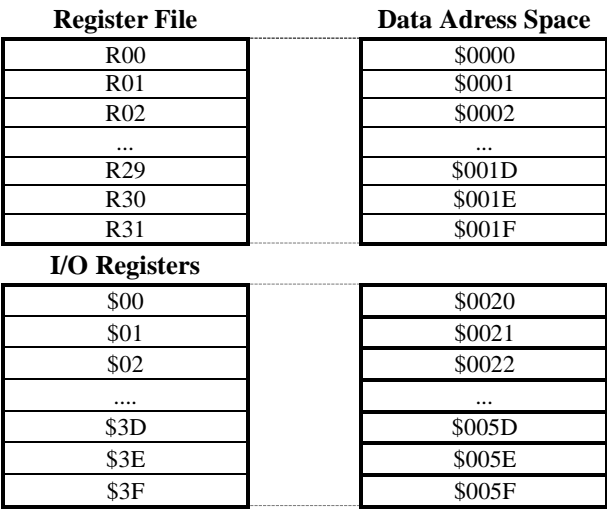
MOV	Rd, Rr	Move Between Registers	$Rd \leftarrow Rr$	None	1
MOVW	Rd, Rr	Copy Register Word	$Rd+1:Rd \leftarrow Rr+1:Rr$	None	1
LDI	Rd, K	Load Immediate	$Rd \leftarrow K$	None	1
LD	Rd, X	Load Indirect	$Rd \leftarrow (X)$	None	2
LD	Rd, X+	Load Indirect and Post-Inc.	$Rd \leftarrow (X), X \leftarrow X + 1$	None	2
LD	Rd, - X	Load Indirect and Pre-Dec.	$X \leftarrow X - 1, Rd \leftarrow (X)$	None	2
LD	Rd, Y	Load Indirect	$Rd \leftarrow (Y)$	None	2
LD	Rd, Y+	Load Indirect and Post-Inc.	$Rd \leftarrow (Y), Y \leftarrow Y + 1$	None	2
LD	Rd, - Y	Load Indirect and Pre-Dec.	$Y \leftarrow Y - 1, Rd \leftarrow (Y)$	None	2
LDD	Rd,Y+q	Load Indirect with Displacement	$Rd \leftarrow (Y + q)$	None	2
LD	Rd, Z	Load Indirect	$Rd \leftarrow (Z)$	None	2
LD	Rd, Z+	Load Indirect and Post-Inc.	$Rd \leftarrow (Z), Z \leftarrow Z+1$	None	2
LD	Rd, -Z	Load Indirect and Pre-Dec.	$Z \leftarrow Z - 1, Rd \leftarrow (Z)$	None	2
LDD	Rd, Z+q	Load Indirect with Displacement	$Rd \leftarrow (Z + q)$	None	2
LDS	Rd, k	Load Direct from SRAM	$Rd \leftarrow (k)$	None	2
ST	X, Rr	Store Indirect	$(X) \leftarrow Rr$	None	2
ST	X+, Rr	Store Indirect and Post-Inc.	$(X) \leftarrow Rr, X \leftarrow X + 1$	None	2
ST	- X, Rr	Store Indirect and Pre-Dec.	$X \leftarrow X - 1, (X) \leftarrow Rr$	None	2
ST	Y, Rr	Store Indirect	$(Y) \leftarrow Rr$	None	2
ST	Y+, Rr	Store Indirect and Post-Inc.	$(Y) \leftarrow Rr, Y \leftarrow Y + 1$	None	2
ST	- Y, Rr	Store Indirect and Pre-Dec.	$Y \leftarrow Y - 1, (Y) \leftarrow Rr$	None	2
STD	Y+q,Rr	Store Indirect with Displacement	$(Y + q) \leftarrow Rr$	None	2
ST	Z, Rr	Store Indirect	$(Z) \leftarrow Rr$	None	2
ST	Z+, Rr	Store Indirect and Post-Inc.	$(Z) \leftarrow Rr, Z \leftarrow Z + 1$	None	2
ST	-Z, Rr	Store Indirect and Pre-Dec.	$Z \leftarrow Z - 1, (Z) \leftarrow Rr$	None	2
STD	Z+q,Rr	Store Indirect with Displacement	$(Z + q) \leftarrow Rr$	None	2
STS	k, Rr	Store Direct to SRAM	$(k) \leftarrow Rr$	None	2
LPM		Load Program Memory	$R0 \leftarrow (Z)$	None	3
LPM	Rd, Z	Load Program Memory	$Rd \leftarrow (Z)$	None	3
LPM	Rd, Z+	Load Program Memory and Post-Inc	$Rd \leftarrow (Z), Z \leftarrow Z+1$	None	3
SPM		Store Program Memory	$(Z) \leftarrow R1:R0$	None	-
IN	Rd, P	In Port	$Rd \leftarrow P$	None	1
OUT	P, Rr	Out Port	$P \leftarrow Rr$	None	1
PUSH	Rr	Push Register on Stack	$Stack \leftarrow Rr$	None	2
POP	Rd	Pop Register from Stack	$Rd \leftarrow Stack$	None	2

Instrukce bitové (testování a rotace)

SBI	P,b	Set Bit in I/O Register	$I/O(P,b) \leftarrow 1$	None	2
CBI	P,b	Clear Bit in I/O Register	$I/O(P,b) \leftarrow 0$	None	2
LSL	Rd	Logical Shift Left	$Rd(n+1) \leftarrow Rd(n), Rd(0) \leftarrow 0$	Z,C,N,V	1
LSR	Rd	Logical Shift Right	$Rd(n) \leftarrow Rd(n+1), Rd(7) \leftarrow 0$	Z,C,N,V	1
ROL	Rd	Rotate Left Through Carry	$Rd(0) \leftarrow C, Rd(n+1) \leftarrow Rd(n), C \leftarrow Rd(7)$	Z,C,N,V	1
ROR	Rd	Rotate Right Through Carry	$Rd(7) \leftarrow C, Rd(n) \leftarrow Rd(n+1), C \leftarrow Rd(0)$	Z,C,N,V	1
ASR	Rd	Arithmetic Shift Right	$Rd(n) \leftarrow Rd(n+1), n=0..6$	Z,C,N,V	1
SWAP	Rd	Swap Nibbles	$Rd(3..0) \leftarrow Rd(7..4), Rd(7..4) \leftarrow Rd(3..0)$	None	1

Instrukce bitové (nastavování)

BSET	s	Flag Set	$SREG(s) \leftarrow 1$	$SREG(s)$	1
BCLR	s	Flag Clear	$SREG(s) \leftarrow 0$	$SREG(s)$	1
BST	Rr, b	Bit Store from Register to T	$T \leftarrow Rr(b)$	T	1
BLD	Rd, b	Bit load from T to Register	$Rd(b) \leftarrow T$	None	1
SEC		Set Carry	$C \leftarrow 1$	C	1
CLC		Clear Carry	$C \leftarrow 0$	C	1
SEN		Set Negative Flag	$N \leftarrow 1$	N	1
CLN		Clear Negative Flag	$N \leftarrow 0$	N	1
SEZ		Set Zero Flag	$Z \leftarrow 1$	Z	1
CLZ		Clear Zero Flag	$Z \leftarrow 0$	Z	1
SEI		Global Interrupt Enable	$I \leftarrow 1$	I	1
CLI		Global Interrupt Disable	$I \leftarrow 0$	I	1
SES		Set Signed Test Flag	$S \leftarrow 1$	S	1
CLS		Clear Signed Test Flag	$S \leftarrow 0$	S	1
SEV		Set Twos Compl. Overflow.	$V \leftarrow 1$	V	1
CLV		Clear Twos Compl. Overflow	$V \leftarrow 0$	V	1
SET		Set T in SREG	$T \leftarrow 1$	T	1
CLT		Clear T in SREG	$T \leftarrow 0$	T	1
SEH		Set Half Carry Flag in SREG	$H \leftarrow 1$	H	1
CLH		Clear Half Carry Flag in SREG	$H \leftarrow 0$	H	1



I/O Registers

\$00
\$01
\$02
....
\$3D
\$3E
\$3F

Internal SRAM

\$0060
\$0061
\$085E
\$085F

Mapa paměti

- Příznaky**

I (7)
Global Interrupt Enable

T (6)
Bit Copy Storage

H (5)
Half Carry Flag

S (4)
Sign Bit, $S = N \oplus V$

V (3)
Two’s Complement Overflow Flag

N (2)
Negative Flag

Z (1)
Zero Flag

C (0)
Carry Flag

Direktivy a funkce překladače assembleru

.include	připojení externího souboru, například s definicemi konstant <code>.include "m32def.inc" ;pro ATmega32</code>
.device	definice typu mikroprocesoru <code>.device ATmega32</code>
.org	definuje adresu v paměti, na jakou má být program přeložen <code>.org 0x0100 ;překlad od adresy 100h</code>
.cseg	programový segment, paměť Flash
.dseg	datový segment, paměť SRAM
.eseg	EEPROM segment
.equ	přířazení hodnoty – definice konstant <code>.equ SREG = 0x3f ;stavový registr</code> <code>.equ SPL = 0x3d ;ukazatel zásobníku</code> <code>.equ SPH = 0x3e</code>
.set	přířazení hodnoty; možno měnit v průběžně ve zdrojovém textu
.def .undef	definice alternativního názvu registru <code>.def ior=R0</code> <code>.undef ior</code>
.byte	definice proměnných v paměti <code>prom1: .BYTE ;rezervování 1 byte</code> <code>tab: .BYTE 20 ;rezervování 20 bytů pro tabulku</code>
.db .dw	definice konstanty o velikosti 1B, 2B <code>consts: .db 0, 255, 0b01010101, -128, 0xaa</code> <code>const1: .dw 0x458A</code>
.macro .endmacro	definice makroinstrukce <code>@0, @1, @2,... parametry makroinstrukce</code> <code>.MACRO SUBI16 ;začátek definice makra</code> <code>subi @0,low(@2) ;odčítání dolní byte</code> <code>sbc @1,high(@2) ;odčítání horní byte</code> <code>.ENDMACRO ;konec definice makra</code> <code>.CSEG</code> <code>SUBI16 r16,r17,0x1234 ;odečtení</code> <code>;konstanty 0x1234 od dvojregistru r17:r16</code>

ATMega32 – piny

(XCK/T0) PB0	1	40	PA0 (ADC0)
(T1) PB1	2	39	PA1 (ADC1)
(INT2/AIN0) PB2	3	38	PA2 (ADC2)
(OC0/AIN1) PB3	4	37	PA3 (ADC3)
(SS) PB4	5	36	PA4 (ADC4)
(MOSI) PB5	6	35	PA5 (ADC5)
(MISO) PB6	7	34	PA6 (ADC6)
(SCK) PB7	8	33	PA7 (ADC7)
RESET	9	32	AREF
VCC	10	31	GND
GND	11	30	AVCC
XTAL2	12	29	PC7 (TOSC2)
XTAL1	13	28	PC6 (TOSC1)
(RXD) PD0	14	27	PC5 (TDI)
(TXD) PD1	15	26	PC4 (TDO)
(INT0) PD2	16	25	PC3 (TMS)
(INT1) PD3	17	24	PC2 (TCK)
(OC1B) PD4	18	23	PC1 (SDA)
(OC1A) PD5	19	22	PC0 (SCL)
(ICP1) PD6	20	21	PD7 (OC2)

- 04 -

Bit	7	6	5	4	3	2	1	0	
	I	T	H	S	V	N	Z	C	SREG
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	

SREG (\$3F) – status registr: stavový registr

Bit	15	14	13	12	11	10	9	8
	SP15	SP14	SP13	SP12	SP11	SP10	SP9	SP8
	SP7	SP6	SP5	SP4	SP3	SP2	SP1	SP0
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

SP – Stack Pointer: ukazatel zásobníku: **SPH (\$3E)**, **SPL (\$3D)**

Bit	7	6	5	4	3	2	1	0	
	CAL7	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	OSCCAL
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	Device Specific Calibration Value								

OSCCAL (\$31) – Oscillator Calibration Register

Bit	7	6	5	4	3	2	1	0
	SE	SM2	SM1	SM0	ISC11	ISC10	ISC01	ISC00
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

MCUCR

MCUCR (\$35) - MCU Control Register

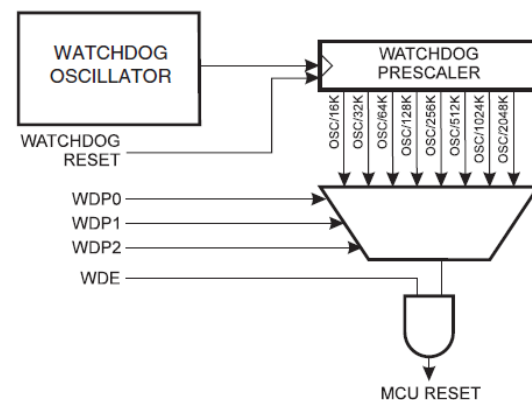
- **SE** – Sleep Enable: povolení úsporných režimů
- **SM2..0** – Sleep Modes: výběr módu.
- **ISCxx**: Interrupt Sense Control: řízení chování přerušovacích vstupů INT0 a INT1

Bit	7	6	5	4	3	2	1	0	
	JTD	ISC2	—	JTRF	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	See Bit Description					

MCUCSR (\$34) – MCU Control and Status Register:

- **JTD:** JTAG Interface Disable
- **ISC2:** řízení chování přerušovacího vstup INT2
- **Zdroje RESETu:**
JTAG Reset, Watchdog, Brown-out Detection, External Reset, Power-On Reset

Watchdog



Bit	7	6	5	4	3	2	1	0
	-	-	-	WDOE	WDE	WDP2	WDP1	WDP0
Read/Write	R	R	R	R/W	R/W	R/W	R/W	R/W
Initial Value	0	0	0	0	0	0	0	0

WDTCR (\$21) – Watchdog Timer Control Register

- **WDTOE** – Watchdog Turn-off Enable (používá se pro vypnutí Watchdogu)
- **WDE** – Watchdog Enable: povolení Watchdogu
- **WDP2..0** – Nastavení intervalu Watchdogu a zpoždění po zapnutí či RESETu

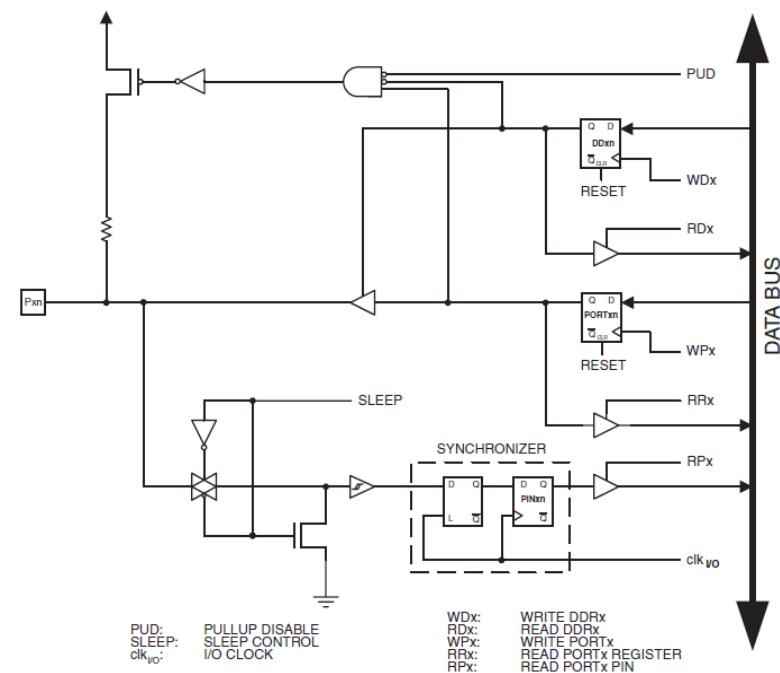
PORTA	\$1B
DDRA	\$1A
PINA	\$19

PORTB	\$18
DDRB	\$17
PINB	\$16

PORTC	\$15
DDRC	\$14
PINC	\$13

PORTD	\$12
DDRD	\$11
PIND	\$10

DDR _x :
0 - vstupní
1 - výstupní



Konfigurace pinů PAX, PBx, PCx, PDx

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)

SFIOR (\$30) - Special Function I/O Register

➤ **PUD:** Pull-up disable – řízení Pull-up rezistorů

Přerušení ATmega 32

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

GICR (\$3B) – General Interrupt Control Register: povolování přerušení

- **INT1:** povolení vnějšího přerušení INT1
- **INT0:** povolení vnějšího přerušení INT0
- **INT2:** povolení vnějšího přerušení INT2
- **IVSEL, IVCE:** povolení změn vektorů přerušení v paměti programu FLASH

Bit	7	6	5	4	3	2	1	0	
	JTD	ISC2	–	JTRF	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	See Bit Description					

MCUCSR (\$34) – MCU Control and Status Register:

- **ISC2:** řízení chování přerušovacího vstup INT2: 0 – sestupná hrana, 1 – náběžná hrana

Bit	7	6	5	4	3	2	1	0	
	SE	SM2	SM1	SM0	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	

MCUCR (\$35) - MCU Control Register

- **ISCxx:** Interrupt Sense Control: řízení chování přerušovacích vstupů INT0 a INT1

Tabulka přerušení ATmega 32

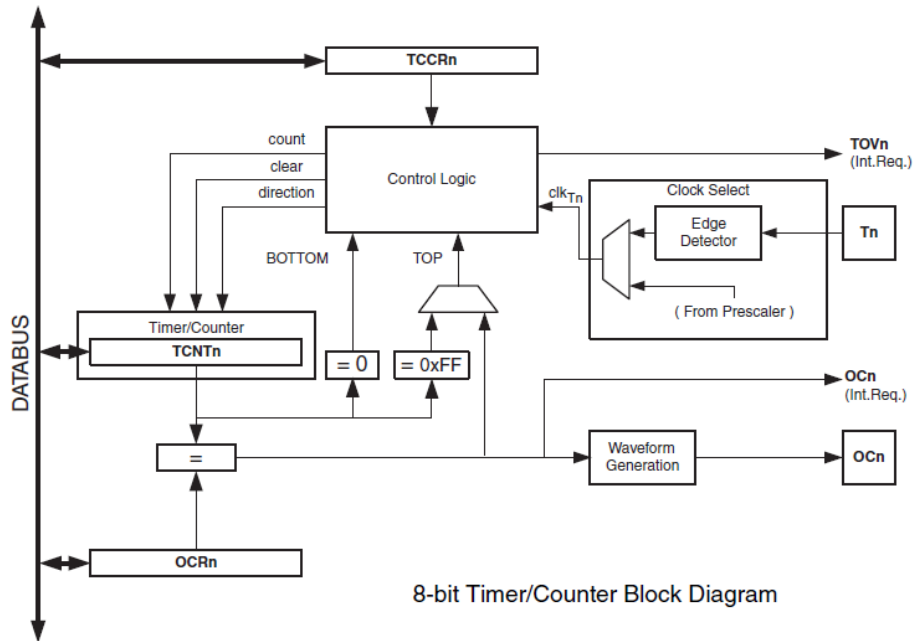
- 05 -

Vector No.	Program Address ⁽²⁾	Source	Interrupt Definition
1	\$000 ⁽¹⁾	RESET	External Pin, Power-on Reset, Brown-out Reset, Watchdog Reset, and JTAG AVR Reset
2	\$002	INT0	External Interrupt Request 0
3	\$004	INT1	External Interrupt Request 1
4	\$006	INT2	External Interrupt Request 2
5	\$008	TIMER2 COMP	Timer/Counter2 Compare Match
6	\$00A	TIMER2 OVF	Timer/Counter2 Overflow
7	\$00C	TIMER1 CAPT	Timer/Counter1 Capture Event
8	\$00E	TIMER1 COMPA	Timer/Counter1 Compare Match A
9	\$010	TIMER1 COMPB	Timer/Counter1 Compare Match B
10	\$012	TIMER1 OVF	Timer/Counter1 Overflow
11	\$014	TIMER0 COMP	Timer/Counter0 Compare Match
12	\$016	TIMER0 OVF	Timer/Counter0 Overflow
13	\$018	SPI, STC	Serial Transfer Complete
14	\$01A	USART, RXC	USART, Rx Complete
15	\$01C	USART, UDRE	USART Data Register Empty
16	\$01E	USART, TXC	USART, Tx Complete
17	\$020	ADC	ADC Conversion Complete
18	\$022	EE_RDY	EEPROM Ready
19	\$024	ANA_COMP	Analog Comparator
20	\$026	TWI	Two-wire Serial Interface
21	\$028	SPM_RDY	Store Program Memory Ready

Chování přerušovacích vstupů INT0 a INT1

ISC11	ISC10	Description
0	0	The low level of INT1 generates an interrupt request.
0	1	Any logical change on INT1 generates an interrupt request.
1	0	The falling edge of INT1 generates an interrupt request.
1	1	The rising edge of INT1 generates an interrupt request.

Timer/Counter 0



8-bit Timer/Counter Block Diagram

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	

TCNT0 (\$32) – Timer/Counter Register 0

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	

OCR0 (\$3C) – Output Compare Register 0

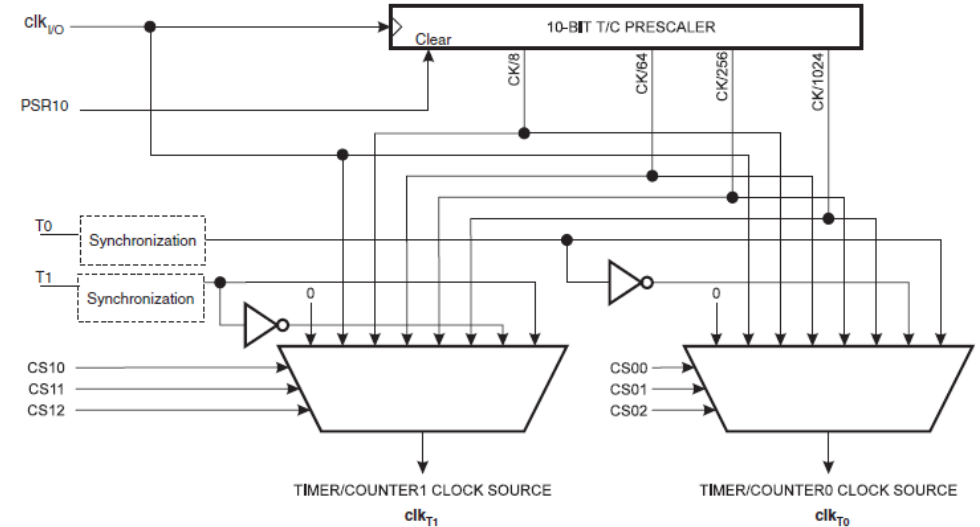
Bit	7	6	5	4	3	2	1	0	
	FOC0	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

TCCR0 (\$33) – Timer/Counter Control Register 0

- **FOC0, WGM00, COM01, COM00, WGM01:** řízení módů komparátoru a generátoru průběhů.
- **CS02, CS01, CS00:** řízení dělicího poměru předděličky 0.

Prescaler 0 a 1

Prescaler for Timer/Counter0 and Timer/Counter1



Bit	7	6	5	4	3	2	1	0	
	ADTS2	ADTS1	ADTS0	–	ACME	PUD	PSR2	PSR10	SFIOR
Read/Write	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

SFIOR (\$30) – Special Function IO Register

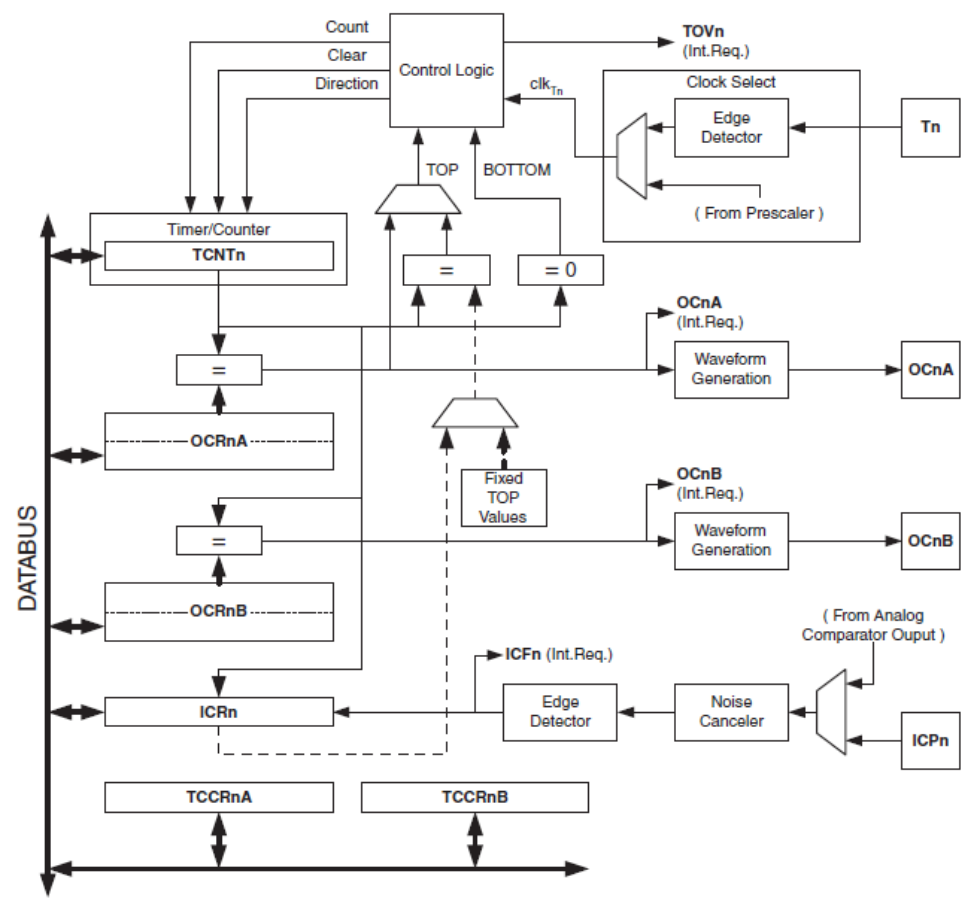
- **PSR10:** resetování předděličky 1 a 0.
- **PSR2:** resetování předděličky 2.

Význam bitů CSxx pro řízení předděličky 0 a 1

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk _{IO} /(No prescaling)
0	1	0	clk _{IO} /8 (From prescaler)
0	1	1	clk _{IO} /64 (From prescaler)
1	0	0	clk _{IO} /256 (From prescaler)
1	0	1	clk _{IO} /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.

Timer/Counter 1

16-bit Timer/Counter Block Diagram



Frekvence výstupního signálu OCn:

$$f_{OCn} = \frac{f_{CLK_IO}}{2 \cdot N \cdot (1 + OCRn)}$$

- **fCLK_IO**: hodinový signál pro časovače
- **OCRn**: přednastavená hodnota v registru OCR0
- **N**: nastavení předděličky (1, 8, 64, 256, 1024)

Bit	7	6	5	4	3	2	1	0	
	TCNT1[15:8]								TCNT1H
	TCNT1[7:0]								TCNT1L
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	

TCNT1 – Timer/Counter 1: TCNT1H (\$2D), TCNT1L (\$2C)

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	

ICR1 – Input Capture Register 1: ICR1H (\$27) ICR1L (\$26)

Bit	7	6	5	4	3	2	1	0	
	OCR1A[15:8]								OCR1AH
	OCR1A[7:0]								OCR1AL
	OCR1B[15:8]								OCR1BH
	OCR1B[7:0]								OCR1BL
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	

OCR1x – Output Compare Register 1A a 1B:
OCR1AH (\$2B), OCR1BH (\$29), OCR1AL (\$2A), OCR1BL (\$28)

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	

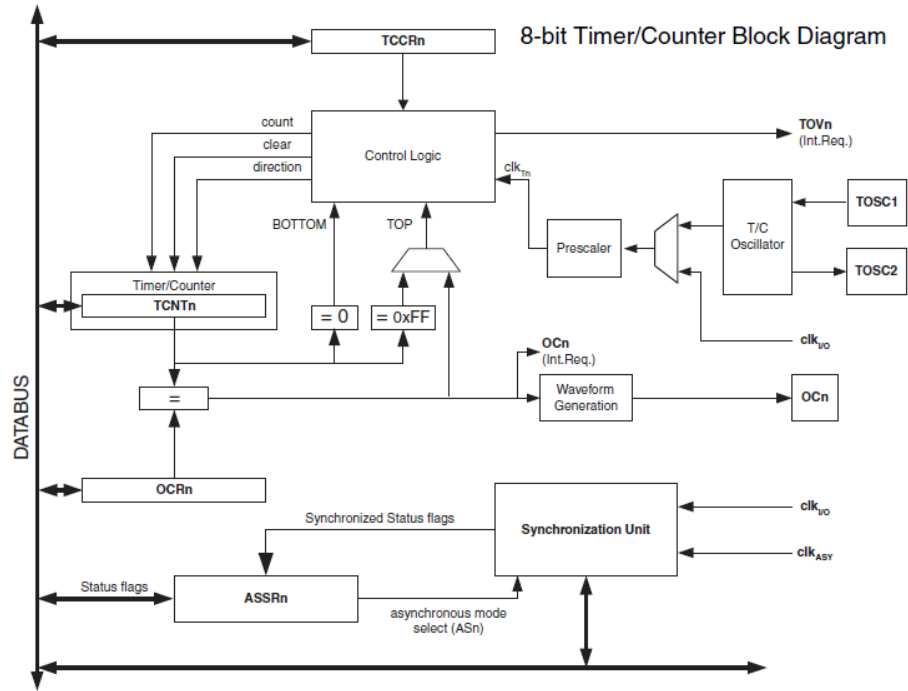
TCCR1A (\$2F) – Timer/Counter Control Register 1A

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	

TCCR1B (\$2E) – Timer/Counter Control Register 1B

- **ICNC1**: povolení jednotky pro potlačení zámkitů
- **ICES1**: volba detekce hrany na vstupu IC1 (0 – sestupná hrana, 1 – náběžná hrana)
- **CS12, CS11, CS10**: řízení dělicího poměru předděličky 1

Timer/Counter 2



Bit	7	6	5	4	3	2	1	0	
	TCNT2[7:0]								TCNT2
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	

TCNT2 (\$24) – Timer/Counter 2

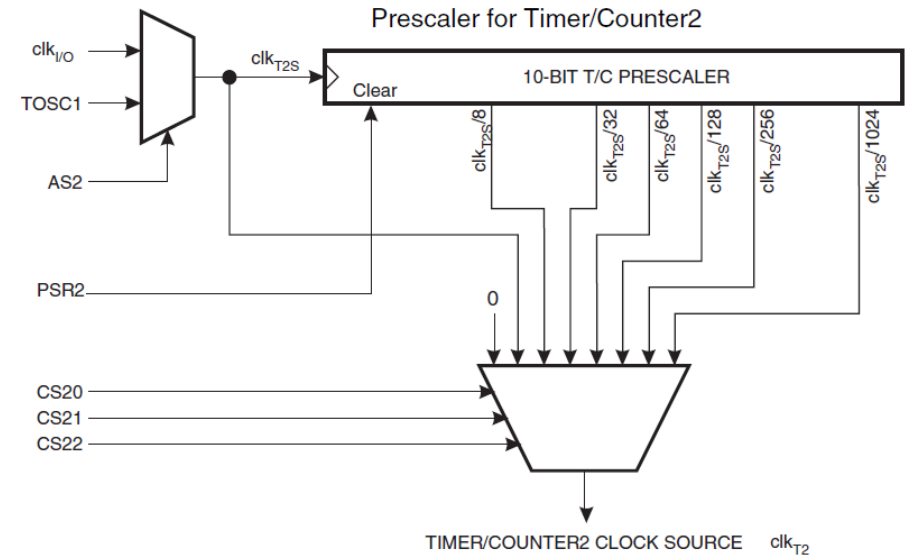
Bit	7	6	5	4	3	2	1	0	
	OCR2[7:0]								OCR2
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

OCR2 (\$23) – Output Compare Register 2

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	

TCCR2 (\$25) – Timer/Counter Control Register 2

- **FOC2, WGM20, COM21, COM20, WGM21:** módy komparátoru a generátoru průb.
- **CS22, CS21, CS20:** řízení dělicího poměru předděličky 2



Bit	7	6	5	4	3	2	1	0	
	-	-	-	-	AS2	TCN2UB	OCR2UB	TCR2UB	ASSR
Read/Write	R	R	R	R	R/W	R	R	R	
Initial Value	0	0	0	0	0	0	0	0	

ASSR (\$22) – Asynchronous Status Register

- **AS2:** Asynchronous Timer/Counter2. AS2=0: $clk_{I/O}$ AS2=1: clk_{ASY}
- **TCN2UB, OCR2UB, TCR2UB:** bity využívané pro práci v asynchronním režimu

Bit	7	6	5	4	3	2	1	0	
	OCIE2	TOIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0	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	

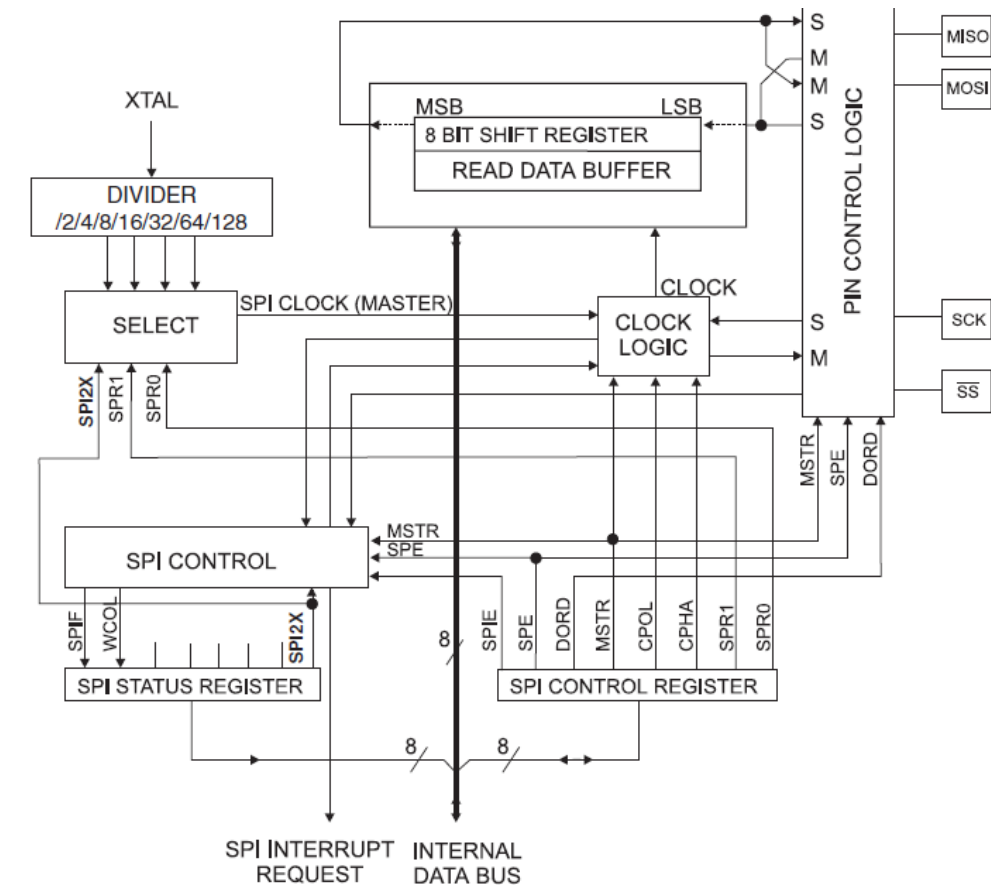
TIMSK (\$39) – Timer/Counter Interrupt Mask: povolování přerušení

- **OCIExx** – komparační registr
- **TICIE1** – záchytný registr
- **TOIEx** – přetečení čítače

Bit	7	6	5	4	3	2	1	0	
	OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0	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	

TIFR (\$38) – Timer/Counter Interrupt Flag: bit nastavovaný při události

- **OCFxx** – komparační registr
- **ICF1** – záchytný registr
- **TOVx** – přetečení čítače



Bit	7	6	5	4	3	2	1	0	
	SPIE	SPE	DORD	MSTR	CPOL	CPHA	SPR1	SPR0	SPCR
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	

SPCR (\$0D) – SPI Control Register

- **SPIE**: SPI Interrupt Enable: povolení přerušení po dokončení přenosu dat
- **SPE**: SPI Enable: povolení činnosti rozhraní SPI
- **DORD**: Data Order – řízení směru přenosu: 1: LSB first, 0: MSB first
- **MSTR**: Master/Slave Select: v režimu master mikroprocesor řídí signál /SS
- **CPOL**: Clock Polarity
- **CPHA**: Clock Phase
- **SPR1, SPR0**: SPI Clock Rate Select 1 and 0: volba přenosové rychlosti

Bit	7	6	5	4	3	2	1	0	
	SPIF	WCOL	–	–	–	–	–	SPI2X	SPSR
Read/Write	R	R	R	R	R	R	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

SPSR (\$0E): SPI Status Register

- **SPIF**: SPI Interrupt Flag: bit je nastaven na 1 po ukončení přenosu.
- **WCOL**: Write COLLision Flag
- **SPI2X**: Double SPI Speed Bit

Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	SPDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	X	X	X	X	X	X	X	X	Undefined

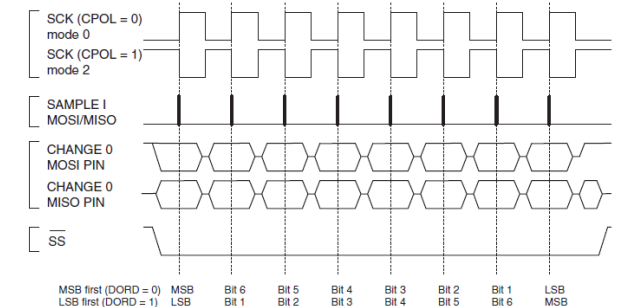
SPDR (\$0F): SPI Data Register

SPI – volba přenosové rychlosti

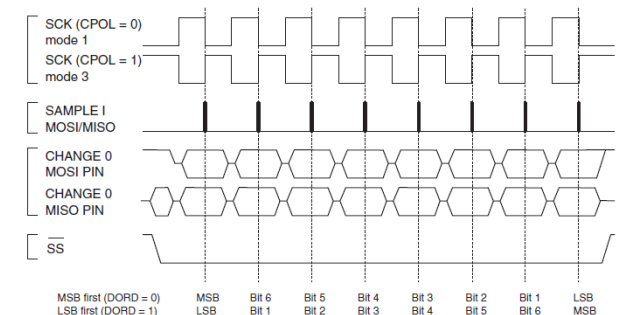
SP12X	SPR1	SPR0	f SCK
0	0	0	$f_{osc}/4$
0	0	1	$f_{osc}/16$
0	1	0	$f_{osc}/64$
0	1	1	$f_{osc}/128$
1	0	0	$f_{osc}/2$
1	1	0	$f_{osc}/32$
1	1	1	$f_{osc}/64$

SPI módy

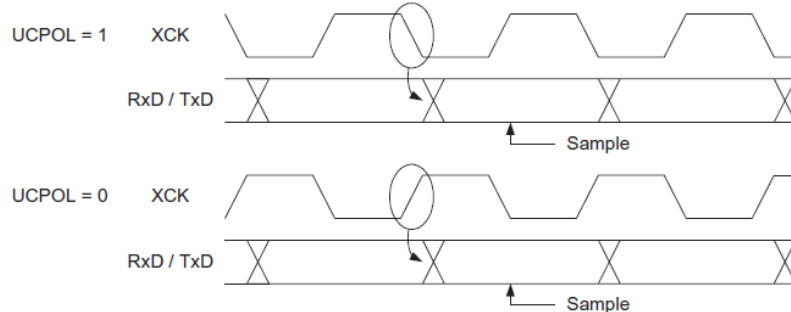
SPI Transfer Format with CPHA = 0



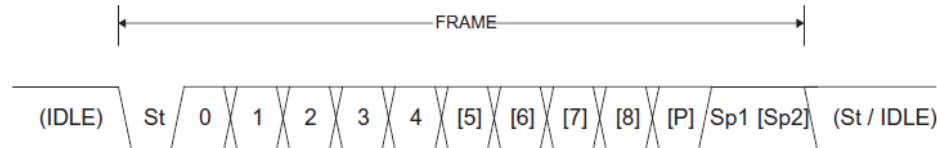
SPI Transfer Format with CPHA = 1



Synchronní režim



Asynchronní režim



- St** Start bit, always low.
(n) Data bits (0 to 8).
P Parity bit. Can be odd or even.
Sp Stop bit, always high.

Registry

Bit	7	6	5	4	3	2	1	0	
	RXB[7:0]								UDR (Read)
	TXB[7:0]								UDR (Write)
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	

UDR (\$0C): USART I/O Data Register – datový registr rozhraní USART

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



Bit	7	6	5	4	3	2	1	0	
	RXC	TXC	UDRE	FE	DOR	PE	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	

UCSRA (\$0B): USART Control and Status Register A

- **RXC:** USART Receive Complete – bit je nastaven po ukončení příjmu dat z RxD a je automaticky nulován při přečtení dat z UDR.
- **TXC:** USART Transmit Complete – bit je nastaven po ukončení vysílání dat na TxD a je automaticky nulován při provedení přerušení vyvolaného ukončením vysílání.
- **UDRE:** USART Data Register Empty – bit je nastaven, jestliže datový registr je připravený přijmout další data pro vysílání.
- **FE:** Frame Error: chyba při přenosu dat (první Stop bit je 0)
- **DOR:** Data OverRun – byl přijat druhý znak dříve než byl předchozí vyjmut z datového registru. Znak čeká v přijímacím posuvném registru.
- **PE:** Parity Error – chyba při přenosu dat - parita
- **U2X:** Double the USART Transmission Speed – dvojnásobná přenosová rychlost
- **MPCM:** Multi-processor Communication Mode – povolení více procesorového přenosového módu

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	

UCSRB (\$0A): USART Control and Status Register B

- **RXCIE:** RX Complete Interrupt Enable – povolení přer. č. 14 po ukončení příjmu dat.
- **TXCIE:** TX Complete Interrupt Enable – povolení přer. č. 16 po ukončení vysílání dat.
- **UDRIE:** USART Data Register Empty Int. Enable – povol. přer. č. 15 vyprázdnění UDR
- **RXEN:** Receiver Enable
- **TXEN:** Transmitter Enable – povolení alternativní funkce portu TxD a RxD.
- **UCSZ2:** Character Size (viz USCR)
- **RXB8 / TXB8:** Receive / Transmit Data Bit 8 – bit 8 při přenosu devíti bitového slova

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	

UCSRC (\$20): USART Control and Status Register C

- **URSEL:** Register Select – volba mezi prací s registrem UCSRC (1) a UBRRH (0).
- **UMSEL:** USART Mode Select volba přenosového módu: 0: asynchronní; 1: synchronní
- **UPM1, UPM0:** Parity Mode
- **UCSZ1, UCSZ0:** charakter size
- **UCPOL:** Clock Polarity: nastavení polarity hodinového signálu XCK

USART – ATmega32

Nastavení délky přenášeného slova

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

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	

UBRR: USART Baud Rate Registers, **UBRRH** (\$20), **UBRRL** (\$09)

- **URSEL:** Register Select – volba mezi prací s registrem UCSRC (1) a UBRRH (0).
- **UBRR11.. 0:** nastavení přenosové rychlosti

Operating Mode	Equation for Calculating Baud Rate ⁽¹⁾	Equation for Calculating UBRR Value
Asynchronous Normal Mode (U2X = 0)	$BAUD = \frac{f_{OSC}}{16(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = 1)	$BAUD = \frac{f_{OSC}}{8(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{OSC}}{2(UBRR + 1)}$	$UBRR = \frac{f_{OSC}}{2BAUD} - 1$

EEPROM – ATmega32

- 11 -

1024 byte

Bit	15	14	13	12	11	10	9	8	
	–	–	–	–	–	–	EEAR9	EEAR8	EEARH
	EEAR7	EEAR6	EEAR5	EEAR4	EEAR3	EEAR2	EEAR1	EEAR0	EEARL
	7	6	5	4	3	2	1	0	
Read/Write	R	R	R	R	R	R	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	X	
	X	X	X	X	X	X	X	X	

EEAR: EEPROM Address Register: **EEARH** (\$1F), **EEARL** (\$1E)

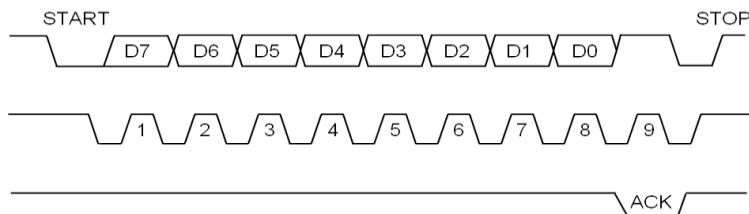
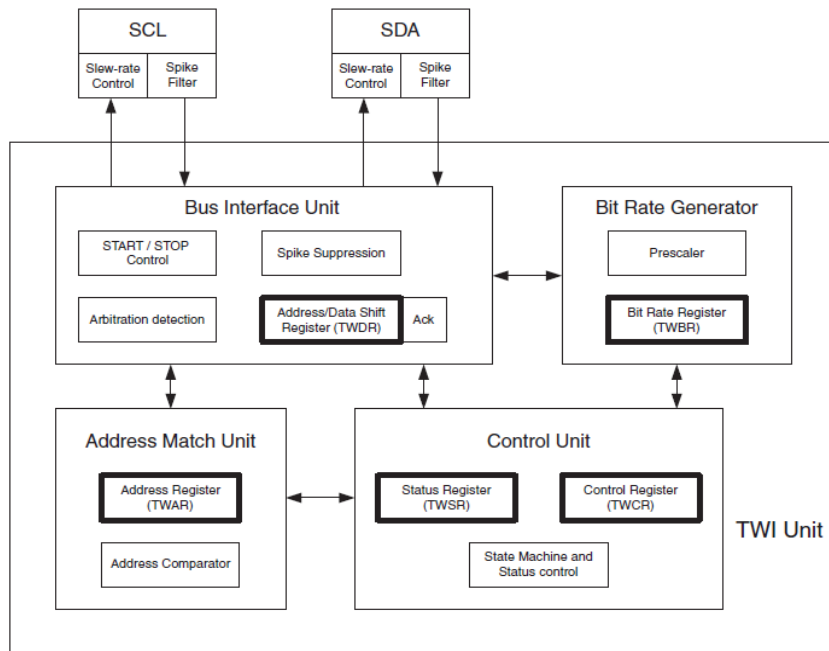
Bit	7	6	5	4	3	2	1	0	
	MSB							LSB	EEDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

EEDR (\$1D): EEPROM Data Register

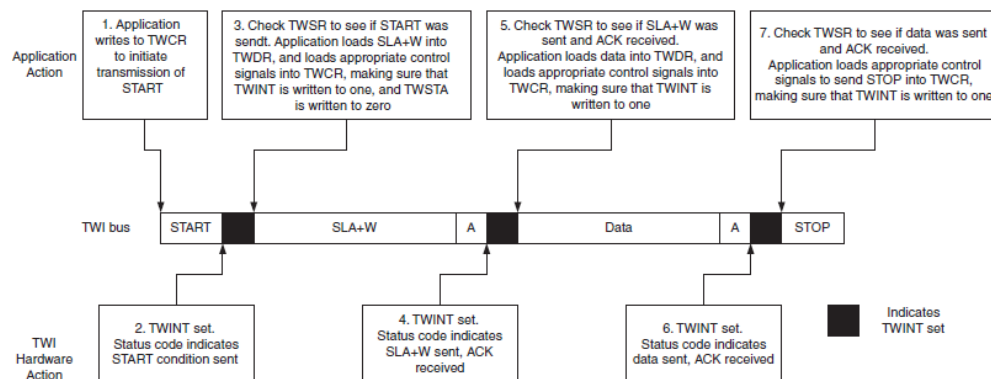
Bit	7	6	5	4	3	2	1	0	
	–	–	–	–	EERIE	EEMWE	EEWE	EERE	EECR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	X	0	

EECR – Řídící (control) registr

- **EERIE:** EEPROM Ready Interrupt Enable – povolení přerušení od paměti EEPROM.
- **EEMWE:** EEPROM Master Write Enable - „hlavní“ povolení zápisu do paměti EEPROM pokud tento bit není nastaven na 1, není možné zapsat a bit EEWE je ignorován.
- **EEWE:** EEPROM Write Enable
 - Nastavení bitu EEWE spustí cyklus zápisu do paměti EEPROM.
 - Po ukončení zápisu je hardwarově nulován.
 - Zápis trvá přibližně 8,5 ms (8500 cyklů), po tuto dobu nelze s EEPROM pracovat.
 - Před zápisem či čtením je nutné otestovat pomocí bitu EEWE, zda je paměť EEPROM připravená nebo zda probíhá cyklus zápisu.
- **EERE:** EEPROM Read Enable – signál pro přečtení dat z paměti EEPROM.



Interfacing the Application to the TWI in a Typical Transmission



Bit	7	6	5	4	3	2	1	0	
	TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0	TWDR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	1	1	1	1	1	1	1	

TWDR (\$03): TWI Data Register: datový registr

Bit	7	6	5	4	3	2	1	0	
	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE	TWAR
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	1	1	1	1	1	1	1	0	

TWAR (\$04): TWI (slave) Address Register

➤ **TWGCE:** TWI General Call Recognition Enable – Bit povolení žádosti o sběrnici

Bit	7	6	5	4	3	2	1	0	
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0	TWBR
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	

TWBR (\$00):

TWI Bit Rate Register

$$\text{SCL frequency} = \frac{\text{CPU Clock frequency}}{16 + 2(\text{TWBR}) \cdot 4^{\text{TWPS}}}$$

TWBR = Value of the TWI Bit Rate Register

TWPS = Value of the prescaler bits in the TWI Status Register

Bit	7	6	5	4	3	2	1	0	
	TWS7	TWS6	TWS5	TWS4	TWS3	–	TWPS1	TWPS0	TWSR
Read/Write	R	R	R	R	R	R	R/W	R/W	
Initial Value	1	1	1	1	1	0	0	0	

TWSR (\$01): TWI Status Register

➤ **TWS7..3:** stavové bity (*jsou pouze ke čtení*) popisují aktuální stav přenosu dat po TWI

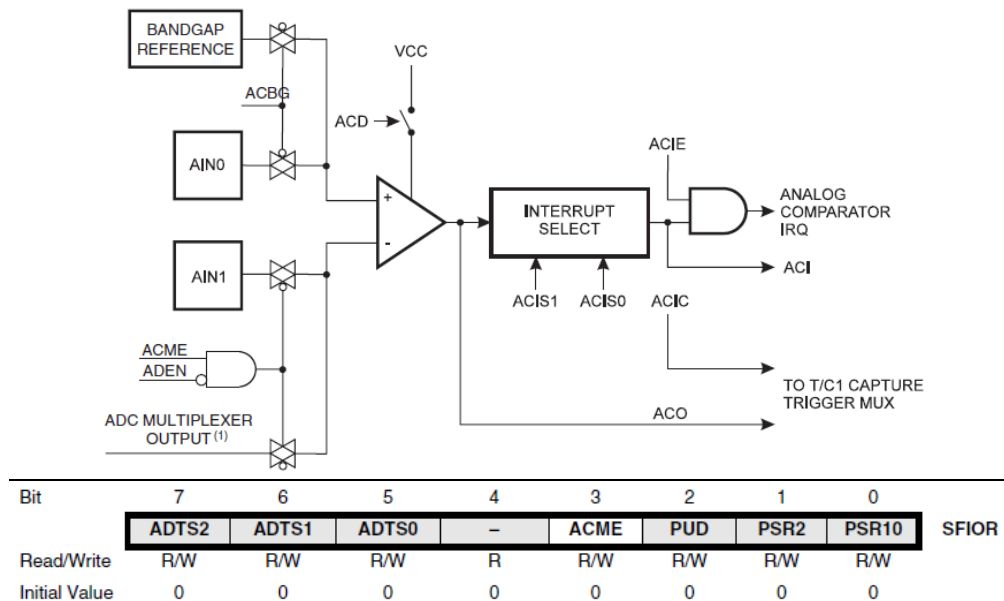
➤ **TWPS1..0:** nastavení předděličky (00 – 1, 01 – 4, 10 – 16, 11 – 64)

Bit	7	6	5	4	3	2	1	0	
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	–	TWIE	TWCR
Read/Write	R/W	R/W	R/W	R/W	R	R/W	R	R/W	
Initial Value	0	0	0	0	0	0	0	0	

TWCR (\$36): TWI Control Register

- **TWINT:** TWI Interrupt Flag – bit je nastaven při generování přerušení TWI (č. 20)
- **TWEA:** TWI Enable Acknowledge Bit – povolení generování potvrzovacího bitu ACK
- **TWSTA:** TWI START Condition Bit – nastavením bitu je generován START bit
- **TWSTO:** TWI STOP Condition Bit – nastavením bitu je generován STOP bit
- **TWWC:** TWI Write Collision Flag: – příznak kolize na sběrnici
- **TWEN:** TWI Enable Bit – povolení TWI
- **TWIE:** TWI Interrupt Enable – povolení přerušení.

Analogový komparátor – ATmega32



SFIOR (\$30): Special Function IO Register

- **ACME:** Analog Comparator Multiplexer Enable

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	

ACSR (\$08): Analog Comparator Control and Status Register:

- **ACD:** Analog Comparator Disable – zakázání funkce komparátoru
- **ACBG:** Analog Comparator Bandgap Select (0=AN0 1=BANDGAP)
- **ACO:** Analog Comparator Output – bit je nastavován výstupem komparátoru
- **ACI:** Analog Comparator Interrupt Flag – přerušení č. 19 – analogový komparátor.
- **ACIE:** Analog Comparator Interrupt Enable – povolení přerušení analog. kompar.
- **ACIC:** Analog Comparator Input Capture Enable
výstup komparátoru je veden na vstupní záchytnou jednotku čítače 1
- **ACIS1..0:** Analog Comparator Interrupt Mode Select

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

Výběr signálu přiváděného na záporný vstup komparátoru

- 13 -

ACME	ADEN	MUX2..0	Analog Comparator Negative Input
0	x	xxx	AIN1
1	1	xxx	AIN1
1	0	000	ADC0
1	0	001	ADC1
1	0	010	ADC2
1	0	011	ADC3
1	0	100	ADC4
1	0	101	ADC5
1	0	110	ADC6
1	0	111	ADC7

JTAG – ATmega32

Bit	7	6	5	4	3	2	1	0	
	MSB/IDRD							LSB	OCDR
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	

OCDR (\$31): On-chip Debug Register

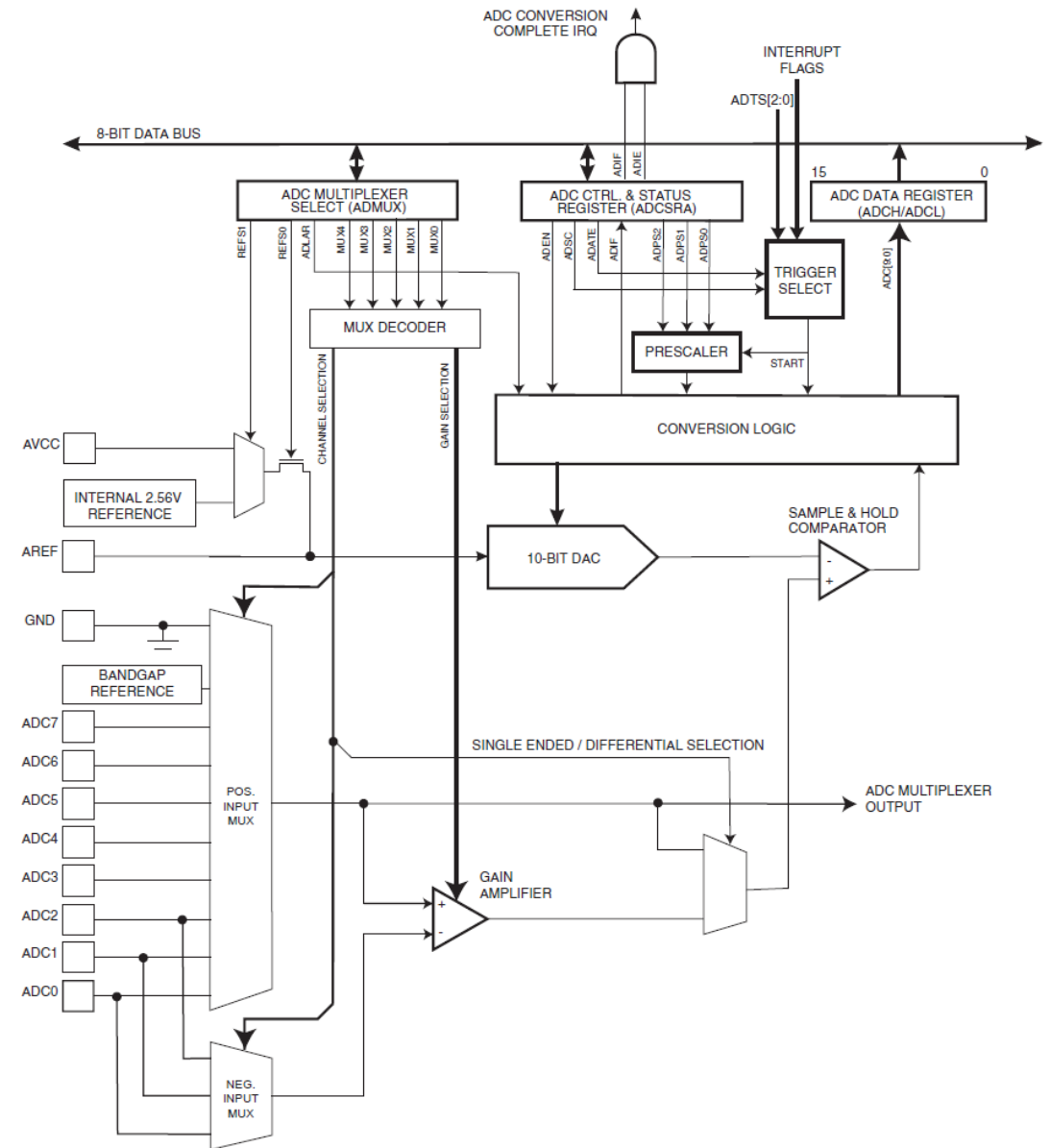
- Komunikační kanál při ladění programu
- Adresa je sdílená s OSCCAL, přístup k registru je pouze při ladění

Bit	7	6	5	4	3	2	1	0	
	JTD	ISC2	–	JTRF	WDRF	BORF	EXTRF	PORF	MCUCSR
Read/Write	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0						See Bit Description

MCUCSR (\$34) – MCU Control and Status Register:

- **JTD:** JTAG Interface Disable – zakázání / povolení JTAG (současně musí být povolená pojistka JTAGEN)
- **JTRF:** JTAG Reset Flag

MUX4..0	Single Ended Input	Positive Differential Input	Negative Differential Input	Gain
00000	ADC0	N/A		
00001	ADC1			
00010	ADC2			
00011	ADC3			
00100	ADC4			
00101	ADC5			
00110	ADC6			
00111	ADC7			
01000	N/A	ADC0	ADC0	10x
01001		ADC1	ADC0	10x
01010 ⁽¹⁾		ADC0	ADC0	200x
01011 ⁽¹⁾		ADC1	ADC0	200x
01100		ADC2	ADC2	10x
01101		ADC3	ADC2	10x
01110 ⁽¹⁾		ADC2	ADC2	200x
01111 ⁽¹⁾		ADC3	ADC2	200x
10000		ADC0	ADC1	1x
10001		ADC1	ADC1	1x
10010		ADC2	ADC1	1x
10011		ADC3	ADC1	1x
10100		ADC4	ADC1	1x
10101		ADC5	ADC1	1x
10110		ADC6	ADC1	1x
10111		ADC7	ADC1	1x
11000		ADC0	ADC2	1x
11001		ADC1	ADC2	1x
11010		ADC2	ADC2	1x
11011		ADC3	ADC2	1x
11100		ADC4	ADC2	1x
11101		ADC5	ADC2	1x
11110	1.22 V (V_{BG})	N/A		
11111	0 V (GND)			



A/D převodník – ATmega32

Bit	15	14	13	12	11	10	9	8	
ADLAR=0								ADC9	ADC8
								ADC7	ADC6
								ADC5	ADC4
								ADC3	ADC2
ADLAR=1								ADC1	ADC0
								ADC9	ADC8
								ADC7	ADC6
								ADC5	ADC4
Read/Write								ADC3	ADC2
								ADC1	ADC0
								ADC9	ADC8
								ADC7	ADC6
Initial Value								ADC5	ADC4
								ADC3	ADC2
								ADC1	ADC0
								ADC9	ADC8

ADCH (\$05), ADCL (\$04): datový registr

Bit	7	6	5	4	3	2	1	0	
Read/Write								REFS1	REFS0
								ADLAR	MUX4
								MUX3	MUX2
								MUX1	MUX0
Initial Value								0	0
								0	0
								0	0
								0	0

ADMUX (\$07): ADC Multiplexer Selection

- **REFS1:0:** Reference Selection Bits – volba zdroje referenčního napětí
- **ADLAR:** ADC Left Adjust Result: formát dat v registru ADC
- **MUX4:0:** Analog Channel and Gain Selection Bits – nastavování multiplexorů a zesilovače signálu přiváděného na vstup kompenzačního A/D převodníku.

REFS1	REFS0	Voltage Reference Selection
0	0	AREF, Internal Vref turned off
0	1	AVCC with external capacitor at AREF pin
1	0	Reserved
1	1	Internal 2.56V Voltage Reference with external capacitor at AREF pin

Bit	7	6	5	4	3	2	1	0	
Read/Write								ADTS2	ADTS1
								ADTS0	–
								ACME	PUD
								PSR2	PSR10
Initial Value								0	0
								0	0
								0	0
								0	0

SFIOR (\$30) – Special Function IO Register

- **ADTS2:0:** ADC Auto Trigger Source
 - Vybrání zdroj, který způsobí start A/D převodu.
 - Převod je spuštěn s náběžnou hranou signálu.

Bit	7	6	5	4	3	2	1	0	
Read/Write								ADEN	ADSC
								ADATE	ADIF
								ADIE	ADPS2
								ADPS1	ADPS0
Initial Value								0	0
								0	0
								0	0
								0	0

ADCSRA (\$06): ADC Control and Status Register A

- **ADEN:** ADC Enable – povolení (zapnutí) A/D
- **ADSC:** ADC Start Conversion v režimu Single Conversion zápis „1“ spustí převod.
- **ADATE:** ADC Auto Trigger Enable
 - povolení spouštění převodu při události na vybraném zdroji.
- **ADIF:** ADC Interrupt Flag – přerušení č. 17 při dokončení převodu
- **ADIE:** ADC Interrupt Enable – povolení přerušení č. 17 – dokončení A/D převodu
- **ADPS2:0:** ADC Prescaler Select – nastavení předděličky pro A/D převodník

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

Pro kvalitní 10 bitový převod se frekvence volí mezi 50 kHz a 200 kHz.

Zdroje spouštění A/D převodu:

ADTS2	ADTS1	ADTS0	Trigger Source
0	0	0	Free Running mode
0	0	1	Analog Comparator
0	1	0	External Interrupt Request 0
0	1	1	Timer/Counter0 Compare Match
1	0	0	Timer/Counter0 Overflow
1	0	1	Timer/Counter Compare Match B
1	1	0	Timer/Counter1 Overflow
1	1	1	Timer/Counter1 Capture Event