5197 - Sistema Digitais

Bacharelado de Informática

UEM – DIN - Prof. Elvio

2016

Roteiro

- ATmega328 (Relógio)
- ATmega328 (Modos *Sleep*)
- ATmega328 (*Reset*)
- ATmega328 (*Watchdog*)

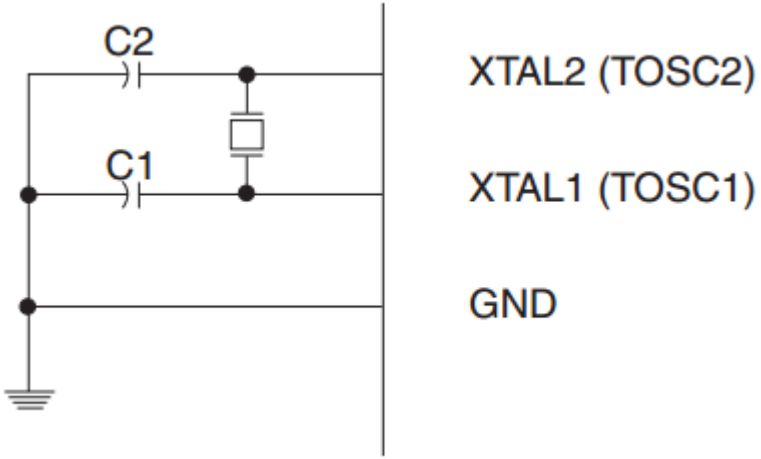
ATmega328 (Relógio)

- Fontes de relógio
  - Default valores:
    - CKSEL = 0010 (oscilador RC interno de 8 MHz)
    - SUT = 10 (máximo tempo de *start-up*)
    - CKDIV8 = 0 (relógio dividido por 8, portanto relógio de 1 MHz)

Device Clocking Option	CKSEL3...0
Low Power Crystal Oscillator	1111 - 1000
Full Swing Crystal Oscillator	0111 - 0110
Low Frequency Crystal Oscillator	0101 - 0100
Internal 128kHz RC Oscillator	0011
Calibrated Internal RC Oscillator	0010
External Clock	0000
Reserved	0001

ATmega328 (Relógio)

- Fontes de relógio:
  Oscilador a cristal de baixo consumo
  - Usa pinos XTAL1 e XTAL2
  - CKSEL0 e SUT[1-0] são usados para definir tempo de *start-up*



Frequency Range (MHz)	Recommended Range for Capacitors C1 and C2 (pF)	CKSEL3...1[1]
0.4 - 0.9	–	100[2]
0.9 - 3.0	12 - 22	101
3.0 - 8.0	12 - 22	110
8.0 - 16.0	12 - 22	111

ATmega328 (Relógio)

- Fontes de relógio: Oscilador a cristal padrão
  - Usa pinos XTAL1 e XTAL2
  - CKSEL0 e SUT[1-0] são usados para definir tempo de *start-up*

Frequency Range[1] (MHz)	Recommended Range for Capacitors C1 and C2 (pF)	CKSEL3...1
0.4 - 20	12 - 22	011

ATmega328 (Relógio)

- Fontes de relógio: Oscilador a cristal de baixa frequência
  - Usa pinos XTAL1 e XTAL2
  - Otimizado para relógio de 32,768 kHz
  - CKSEL0 e SUT[1-0] são usados para definir tempo de *start-up*
- Fontes de relógio: Oscilador RC interno

Frequency Range[2] (MHz)	CKSEL3...0
7.3 - 8.1	0010[1]

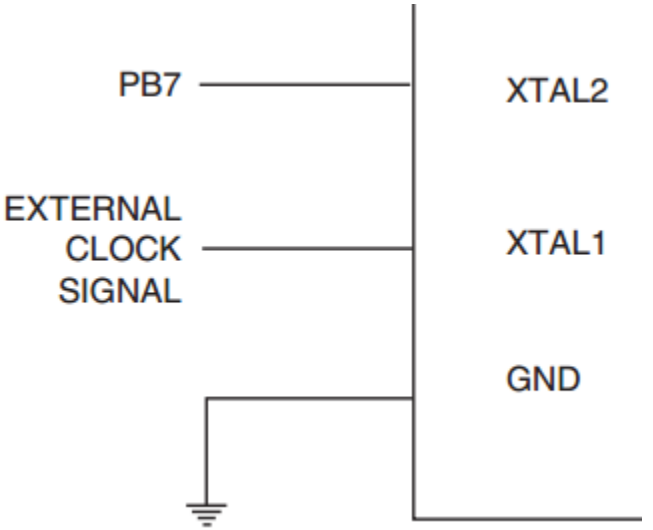
ATmega328 (Relógio)

- Fontes de relógio: Oscilador interno de 128 kHz

Nominal Frequency[1]	CKSEL3...0
128kHz	0011

- Fontes de relógio: Oscilador externo
  - Utiliza pino XTAL1

Frequency	CKSEL3...0
0 - 20MHz	0000



ATmega328 (Modos *Sleep*)

- Usado para economizar energia

BOD: *Brown-out Detector*

	Active Clock Domains					Oscillators		Wake-up Sources							
	clk_{CPU}	clk_{FLASH}	clk_{IO}	clk_{ADC}	clk_{ASY}	Main Clock Source Enabled	Timer Oscillator Enabled	INT1, INT0 and Pin Change	TWI Address Match	Timer2	SPM/EEPROM Ready	ADC	WDT	Other I/O	Software BOD Disable
Idle			X	X	X	X	X(2)	X	X	X	X	X	X	X	
ADC Noise Reduction				X	X	X	X(2)	X(3)	X	X(2)	X	X	X		
Power-down								X(3)	X				X		X
Power-save					X		X(2)	X(3)	X	X			X		X
Standby(1)						X		X(3)	X				X		X
Extended Standby					X(2)	X	X(2)	X(3)	X	X			X		X

ATmega328 (Modos *Sleep*)

- *Idle*: basicamente para a CPU mas mantém os periféricos funcionando
- ADC *Noise Reduction*: para CPU e a maioria dos blocos de E/S
  - Diminui o ruído e permite ADC mais precisa
- *Power Down*: para todos os relógios, mantendo apenas em operação blocos que operam em modo assíncrono

ATmega328 (Modos *Sleep*)

- *Power Save*: igual a *Power Down*, exceto que Timer 2 permanece ativo
- *Standyby*: igual a *Power Down*, exceto que oscilador permanece ativo
- *Extended Standyby*: igual a *Power Save*, exceto que oscilador permanece ativo
- Redução de consumo: permite desligar o relógio de alguns blocos, diminuindo o consumo

ATmega328 (Modos *Sleep*)

- Registrador *Sleep Mode Control* (SMCR)

Bit	7	6	5	4	3	2	1	0	
0x33 (0x53)	–	–	–	–	SM2	SM1	SM0	SE	SMCR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

  - *Sleep Enable* (SE)
    - Deve estar setado quando da execução da instrução SLEEP para uC entrar em modo *sleep*
    - Recomenda-se:
      - setar este bit imediatamente antes da instrução SLEEP
      - limpar este bit imediatamente após acordar

ATmega328 (Modos *Sleep*)

- Registrador *Sleep Mode Control* (SMCR)
  - SM[2:0]: *Sleep Mode Select Bits*

SM2	SM1	SM0	Sleep Mode
0	0	0	Idle
0	0	1	ADC Noise Reduction
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	External Standby(1)

ATmega328 (Modos *Sleep*)

- Registrador *Power Reduction* (PRR)
  - Cada bit setado desliga bloco correspondente
    - *PRTWI: Power Reduction TWI (Twin Wire Interface)*
    - *PRTIM[2-0]: Power Reduction Timer/Counter [2-0]*
    - *PRSPI: Power Reduction Serial Peripheral Interface*
    - *PRUSART0: Power Reduction USART0*
    - *PRADC: Power Reduction ADC*

Bit	7	6	5	4	3	2	1	0	
(0x64)	**PRTWI**	**PRTIM2**	**PRTIM0**	**–**	**PRTIM1**	**PRSPI**	**PRUSART0**	**PRADC**	**PRR**
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	

ATmega328 (*Reset*)

- Causas de *Reset*
  - *Power-on Reset*: *reset* quando a tensão de alimentação cai abaixo do limiar *Power-on Reset*
  - *Reset* Externo: *reset* quando "0" é aplicado ao pino RESET por tempo suficientemente longo
  - *Watchdog System Reset*: reset quando *Watchdog* expira e o reset de *Watchdog* está habilitado
  - *Brown-out Reset*: reset quando a tensão de alimentação cai abaixo do limiar *Brown-out Reset* e o *Brown-out Detector* está habilitado

ATmega328 (*Reset*)

- Registrador MCU Status (MCUSR)
  - Informa a causa do reset
    - WDRF: *Watchdog System Reset Flag*
      - Reset causado por *Watchdog*
      - Bit é ressetado em Power-on Reset ou escrevendo 0 neste *flag*
    - BORF: *Brown-out Reset Flag*
      - Reset causado por *Brown-out*
      - Bit é ressetado em *Power-on Reset* ou escrevendo 0 neste *flag*
    - EXTRF: *External Reset Flag*
      - Reset causado por *External Reset*
      - Bit é ressetado em *Power-on Reset* ou escrevendo 0 neste *flag*
    - PORF: *Power-on Reset Flag*
      - Bit é setado em *Power-on Reset*
      - Bit é ressetado escrevendo 0 neste *flag*

Bit	7	6	5	4	3	2	1	0	
0x34 (0x54)	–	–	–	–	WDRF	BORF	EXTRF	PORF	MCUSR
Read/Write	R	R	R	R	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	See Bit Description				

ATmega328 (*Watchdog*)

- *Watchdog*
  - Procura manter sanidade na execução (detecta falhas na execução ou no *hardware*)
  - Execução deve regularmente reiniciar o temporizador de *Watchdog*
  - Caso contrário:
    - Provoca interrupção: para retirar de *sleep* ou limitar tempo de execução
    - Provoca *reset*: em caso de falhas (sistema *hang-up*)
    - Provoca interrupção e *reset*: primeiro interrupção, mudando para *reset*
  - Programável de 16 ms a 8 s

ATmega328 (*Watchdog*)

- Registrador *Watchdog Timer Control* (WDTCSR)
  - WDIF: *Watchdog Interrupt Flag*
    - Setado quando o temporizador expira e ele está configurado para interrupção
    - Ressetado por hardware quando a interrupção é tratada ou por software na escrita deste *flag*
  - WDIE: *Watchdog Interrupt Enable*
    - Habilita a interrupção (*Global Interrupt Enable* deve estar habilitado)
  - WDCE: *Watchdog Change Enable*
    - Deve ser setado se bit WDE é ressetado e/ou bits WDP (*prescaler*) são alterados
    - Ressetado por hardware
  - WDE: *Watchdog System Reset Enable*
    - Habilita *reset*
  - WDP[3-0]: *Watchdog Timer Prescaler*
    - Determina o tempo do temporizador

Bit	7	6	5	4	3	2	1	0	
(0x60)	**WDIF**	**WDIE**	**WDP3**	**WDCE**	**WDE**	**WDP2**	**WDP1**	**WDP0**	**WDTCSR**
Read/Write	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	X	0	0	0	

ATmega328 (*Watchdog*)

- WDTCSR – *Watchdog Timer Control Register*

  WDTON: *Watchdog Timer always on*

WDTON[(1)]	WDE	WDIE	Mode	Action on Time-out
1	0	0	Stopped	None
1	0	1	Interrupt Mode	Interrupt
1	1	0	System Reset Mode	Reset
1	1	1	Interrupt and System Reset Mode	Interrupt, then go to System Reset Mode
0	x	x	System Reset Mode	Reset

ATmega328 (*Watchdog*)

- WDTCSR – *Watchdog Timer Control Register*

WDP3	WDP2	WDP1	WDP0	Number of WDT Oscillator Cycles	Typical Time-out at V_{CC} = 5.0V
0	0	0	0	2K (2048) cycles	16ms
0	0	0	1	4K (4096) cycles	32ms
0	0	1	0	8K (8192) cycles	64ms
0	0	1	1	16K (16384) cycles	0.125 s
0	1	0	0	32K (32768) cycles	0.25 s
0	1	0	1	64K (65536) cycles	0.5 s
0	1	1	0	128K (131072) cycles	1.0 s
0	1	1	1	256K (262144) cycles	2.0 s
1	0	0	0	512K (524288) cycles	4.0 s
1	0	0	1	1024K (1048576) cycles	8.0 s
1	0	1	x	Reserved	
1	1	x	x		

ATmega328 (*Watchdog*)

Assembly Code Example[1]

```
WDT_off:
  ; Turn off global interrupt
  cli
  ; Reset Watchdog Timer
  wdr
  ; Clear WDRF in MCUSR
  in    r16, MCUSR
  andi  r16, (0xff & (0<<WDRF))
  out   MCUSR, r16
  ; Write logical one to WDCE and WDE
  ; Keep old prescaler setting to prevent unintentional time-out
  lds r16, WDTCSR
  ori   r16, (1<<WDCE) | (1<<WDE)
  sts WDTCSR, r16
  ; Turn off WDT
  ldi   r16, (0<<WDE)
  sts WDTCSR, r16
  ; Turn on global interrupt
  sei
  ret
```

C Code Example[1]

```
void WDT_off(void)
{
  __disable_interrupt();
  __watchdog_reset();
  /* Clear WDRF in MCUSR */
  MCUSR &= ~(1<<WDRF);
  /* Write logical one to WDCE and WDE */
  /* Keep old prescaler setting to prevent unintentional time-out */
  WDTCSR |= (1<<WDCE) | (1<<WDE);
  /* Turn off WDT */
  WDTCSR = 0x00;
  __enable_interrupt();
}
```

ATmega328 (*Watchdog*)

Assembly Code Example[1]

```
WDT_Prescaler_Change:
  ; Turn off global interrupt
  cli
  ; Reset Watchdog Timer
  wdr
  ; Start timed sequence
  lds r16, WDTCSR
  ori   r16, (1<<WDCE) | (1<<WDE)
  sts WDTCSR, r16
  ; --  Got four cycles to set the new values from here -
  ; Set new prescaler(time-out) value = 64K cycles (~0.5 s)
  ldi   r16, (1<<WDE) | (1<<WDP2) | (1<<WDP0)
  sts WDTCSR, r16
  ; --  Finished setting new values, used 2 cycles -
  ; Turn on global interrupt
  sei
  ret
```

C Code Example[1]

```
void WDT_Prescaler_Change(void)
{
  __disable_interrupt();
  __watchdog_reset();
  /* Start timed  sequence */
  WDTCSR |= (1<<WDCE) | (1<<WDE);
  /* Set new prescaler(time-out) value = 64K cycles (~0.5 s) */
  WDTCSR  = (1<<WDE) | (1<<WDP2) | (1<<WDP0);
  __enable_interrupt();
}
```