```
;***** THIS IS A MACHINE GENERATED FILE - DO NOT EDIT ********************
;***** Created: 2007-02-28 07:32 ******* Source: ATtiny2313.xml ***********
;*************************************************************************
;* A P P L I C A T I O N   N O T E   F O R   T H E   A V R   F A M I L Y
;*
;* Number           : AVR000
;* File Name        : "tn2313def.inc"
;* Title            : Register/Bit Definitions for the ATtiny2313
;* Date             : 2007-02-28
;* Version          : 2.24
;* Support E-mail   : avr@atmel.com
;* Target MCU       : ATtiny2313
;*
;* DESCRIPTION
;* When including this file in the assembly program file, all I/O register
;* names and I/O register bit names appearing in the data book can be used.
;* In addition, the six registers forming the three data pointers X, Y and
;* Z have been assigned names XL - ZH. Highest RAM address for Internal
;* SRAM is also defined
;*
;* The Register names are represented by their hexadecimal address.
;*
;* The Register Bit names are represented by their bit number (0-7).
;*
;* Please observe the difference in using the bit names with instructions
;* such as "sbr"/"cbr" (set/clear bit in register) and "sbrs"/"sbrc"
;* (skip if bit in register set/cleared). The following example illustrates
;* this:
;*
;* in    r16,PORTB             ;read PORTB latch
;* sbr   r16,(1<<PB6)+(1<<PB5) ;set PB6 and PB5 (use masks, not bit#)
;* out   PORTB,r16             ;output to PORTB
;*
;* in    r16,TIFR              ;read the Timer Interrupt Flag Register
;* sbrc  r16,TOV0              ;test the overflow flag (use bit#)
;* rjmp  TOV0_is_set           ;jump if set
```

```
;* ...                          ;otherwise do something else
;*****************************************************************

#ifndef _TN2313DEF_INC_
#define _TN2313DEF_INC_


#pragma partinc 0

; ***** SPECIFY DEVICE ************************************************
.device ATtiny2313
#pragma AVRPART ADMIN PART_NAME ATtiny2313
.equ   SIGNATURE_000 = 0x1e
.equ   SIGNATURE_001 = 0x91
.equ   SIGNATURE_002 = 0x0a

#pragma AVRPART CORE CORE_VERSION V2
#pragma AVRPART CORE NEW_INSTRUCTIONS lpm rd,z+


; ***** I/O REGISTER DEFINITIONS **************************************
; NOTE:
; Definitions marked "MEMORY MAPPED"are extended I/O ports
; and cannot be used with IN/OUT instructions
.equ   SREG   = 0x3f
.equ   SPL    = 0x3d
.equ   OCR0B  = 0x3c
.equ   GIMSK  = 0x3b
.equ   EIFR   = 0x3a
.equ   TIMSK  = 0x39
.equ   TIFR   = 0x38
.equ   SPMCSR = 0x37
.equ   OCR0A  = 0x36
.equ   MCUCR  = 0x35
.equ   MCUSR  = 0x34
.equ   TCCR0B = 0x33
```

```
.equ   TCNT0  = 0x32
.equ   OSCCAL = 0x31
.equ   TCCR0A = 0x30
.equ   TCCR1A = 0x2f
.equ   TCCR1B = 0x2e
.equ   TCNT1L = 0x2c
.equ   TCNT1H = 0x2d
.equ   OCR1AL = 0x2a
.equ   OCR1AH = 0x2b
.equ   OCR1BL = 0x28
.equ   OCR1BH = 0x29
.equ   CLKPR  = 0x26
.equ   ICR1L  = 0x24
.equ   ICR1H  = 0x25
.equ   GTCCR  = 0x23
.equ   TCCR1C = 0x22
.equ   WDTCR  = 0x21
.equ   PCMSK  = 0x20
.equ   EEAR   = 0x1e
.equ   EEDR   = 0x1d
.equ   EECR   = 0x1c
.equ   PORTA  = 0x1b
.equ   DDRA   = 0x1a
.equ   PINA   = 0x19
.equ   PORTB  = 0x18
.equ   DDRB   = 0x17
.equ   PINB   = 0x16
.equ   GPIOR2 = 0x15
.equ   GPIOR1 = 0x14
.equ   GPIOR0 = 0x13
.equ   PORTD  = 0x12
.equ   DDRD   = 0x11
.equ   PIND   = 0x10
.equ   USIDR  = 0x0f
.equ   USISR  = 0x0e
.equ   USICR  = 0x0d
```

```
.equ   UDR    = 0x0c
.equ   UCSRA  = 0x0b
.equ   UCSRB  = 0x0a
.equ   UBRRL  = 0x09
.equ   ACSR   = 0x08
.equ   UCSRC  = 0x03
.equ   UBRRH  = 0x02
.equ   DIDR   = 0x01


; ***** BIT DEFINITIONS ************************************************

; ***** PORTB ************************
; PORTB - Port B Data Register
.equ   PORTB0 = 0    ; Port B Data Register bit 0
.equ   PB0    = 0    ; For compatibility
.equ   PORTB1 = 1    ; Port B Data Register bit 1
.equ   PB1    = 1    ; For compatibility
.equ   PORTB2 = 2    ; Port B Data Register bit 2
.equ   PB2    = 2    ; For compatibility
.equ   PORTB3 = 3    ; Port B Data Register bit 3
.equ   PB3    = 3    ; For compatibility
.equ   PORTB4 = 4    ; Port B Data Register bit 4
.equ   PB4    = 4    ; For compatibility
.equ   PORTB5 = 5    ; Port B Data Register bit 5
.equ   PB5    = 5    ; For compatibility
.equ   PORTB6 = 6    ; Port B Data Register bit 6
.equ   PB6    = 6    ; For compatibility
.equ   PORTB7 = 7    ; Port B Data Register bit 7
.equ   PB7    = 7    ; For compatibility

; DDRB - Port B Data Direction Register
.equ   DDB0   = 0    ; Port B Data Direction Register bit 0
.equ   DDB1   = 1    ; Port B Data Direction Register bit 1
.equ   DDB2   = 2    ; Port B Data Direction Register bit 2
.equ   DDB3   = 3    ; Port B Data Direction Register bit 3
```

```
.equ   DDB4   = 4    ; Port B Data Direction Register bit 4
.equ   DDB5   = 5    ; Port B Data Direction Register bit 5
.equ   DDB6   = 6    ; Port B Data Direction Register bit 6
.equ   DDB7   = 7    ; Port B Data Direction Register bit 7

; PINB - Port B Input Pins
.equ   PINB0  = 0    ; Port B Input Pins bit 0
.equ   PINB1  = 1    ; Port B Input Pins bit 1
.equ   PINB2  = 2    ; Port B Input Pins bit 2
.equ   PINB3  = 3    ; Port B Input Pins bit 3
.equ   PINB4  = 4    ; Port B Input Pins bit 4
.equ   PINB5  = 5    ; Port B Input Pins bit 5
.equ   PINB6  = 6    ; Port B Input Pins bit 6
.equ   PINB7  = 7    ; Port B Input Pins bit 7


; ***** TIMER_COUNTER_0 **************
; TIMSK - Timer/Counter Interrupt Mask Register
.equ   OCIE0A = 0    ; Timer/Counter0 Output Compare Match A Interrupt Enable
.equ   TOIE0  = 1    ; Timer/Counter0 Overflow Interrupt Enable
.equ   OCIE0B = 2    ; Timer/Counter0 Output Compare Match B Interrupt Enable

; TIFR - Timer/Counter Interrupt Flag register
.equ   OCF0A  = 0    ; Timer/Counter0 Output Compare Flag 0A
.equ   TOV0   = 1    ; Timer/Counter0 Overflow Flag
.equ   OCF0B  = 2    ; Timer/Counter0 Output Compare Flag 0B

; OCR0B - Timer/Counter0 Output Compare Register
.equ   OCR0_0 = 0    ;
.equ   OCR0_1 = 1    ;
.equ   OCR0_2 = 2    ;
.equ   OCR0_3 = 3    ;
.equ   OCR0_4 = 4    ;
.equ   OCR0_5 = 5    ;
.equ   OCR0_6 = 6    ;
.equ   OCR0_7 = 7    ;
```

```
; OCR0A - Timer/Counter0 Output Compare Register
;.equ  OCR0_0 = 0    ;
;.equ  OCR0_1 = 1    ;
;.equ  OCR0_2 = 2    ;
;.equ  OCR0_3 = 3    ;
;.equ  OCR0_4 = 4    ;
;.equ  OCR0_5 = 5    ;
;.equ  OCR0_6 = 6    ;
;.equ  OCR0_7 = 7    ;

; TCCR0A - Timer/Counter  Control Register A
.equ   WGM00  = 0    ; Waveform Generation Mode
.equ   WGM01  = 1    ; Waveform Generation Mode
.equ   COM0B0 = 4    ; Compare Match Output B Mode
.equ   COM0B1 = 5    ; Compare Match Output B Mode
.equ   COM0A0 = 6    ; Compare Match Output A Mode
.equ   COM0A1 = 7    ; Compare Match Output A Mode

; TCNT0 - Timer/Counter0
.equ   TCNT0_0      = 0    ;
.equ   TCNT0_1      = 1    ;
.equ   TCNT0_2      = 2    ;
.equ   TCNT0_3      = 3    ;
.equ   TCNT0_4      = 4    ;
.equ   TCNT0_5      = 5    ;
.equ   TCNT0_6      = 6    ;
.equ   TCNT0_7      = 7    ;

; TCCR0B - Timer/Counter Control Register B
.equ   TCCR0  = TCCR0B     ; For compatibility
.equ   CS00   = 0    ; Clock Select
.equ   CS01   = 1    ; Clock Select
.equ   CS02   = 2    ; Clock Select
.equ   WGM02  = 3    ;
.equ   FOC0B  = 6    ; Force Output Compare B
```

```
.equ   FOC0A  = 7    ; Force Output Compare B


; ***** TIMER_COUNTER_1 **************
; TIMSK - Timer/Counter Interrupt Mask Register
.equ   ICIE1  = 3    ; Timer/Counter1 Input Capture Interrupt Enable
.equ   TICIE  = ICIE1      ; For compatibility
.equ   OCIE1B = 5    ; Timer/Counter1 Output CompareB Match Interrupt Enable
.equ   OCIE1A = 6    ; Timer/Counter1 Output CompareA Match Interrupt Enable
.equ   TOIE1  = 7    ; Timer/Counter1 Overflow Interrupt Enable

; TIFR - Timer/Counter Interrupt Flag register
.equ   ICF1   = 3    ; Input Capture Flag 1
.equ   OCF1B  = 5    ; Output Compare Flag 1B
.equ   OCF1A  = 6    ; Output Compare Flag 1A
.equ   TOV1   = 7    ; Timer/Counter1 Overflow Flag

; TCCR1A - Timer/Counter1 Control Register A
.equ   WGM10  = 0    ; Pulse Width Modulator Select Bit 0
.equ   PWM10  = WGM10      ; For compatibility
.equ   WGM11  = 1    ; Pulse Width Modulator Select Bit 1
.equ   PWM11  = WGM11      ; For compatibility
.equ   COM1B0 = 4    ; Comparet Ouput Mode 1B, bit 0
.equ   COM1B1 = 5    ; Compare Output Mode 1B, bit 1
.equ   COM1A0 = 6    ; Comparet Ouput Mode 1A, bit 0
.equ   COM1A1 = 7    ; Compare Output Mode 1A, bit 1

; TCCR1B - Timer/Counter1 Control Register B
.equ   CS10   = 0    ; Clock Select bit 0
.equ   CS11   = 1    ; Clock Select 1 bit 1
.equ   CS12   = 2    ; Clock Select1 bit 2
.equ   WGM12  = 3    ; Waveform Generation Mode Bit 2
.equ   CTC1   = WGM12      ; For compatibility
.equ   WGM13  = 4    ; Waveform Generation Mode Bit 3
.equ   ICES1  = 6    ; Input Capture 1 Edge Select
.equ   ICNC1  = 7    ; Input Capture 1 Noise Canceler
```

```
; TCCR1C - Timer/Counter1 Control Register C
.equ   FOC1B  = 6    ; Force Output Compare for Channel B
.equ   FOC1A  = 7    ; Force Output Compare for Channel A

; OCR1BH - Timer/Counter1 Outbut Compare Register High Byte
.equ   OCR1AH0      = 0    ; Timer/Counter1 Outbut Compare Register High Byte bit 0
.equ   OCR1AH1      = 1    ; Timer/Counter1 Outbut Compare Register High Byte bit 1
.equ   OCR1AH2      = 2    ; Timer/Counter1 Outbut Compare Register High Byte bit 2
.equ   OCR1AH3      = 3    ; Timer/Counter1 Outbut Compare Register High Byte bit 3
.equ   OCR1AH4      = 4    ; Timer/Counter1 Outbut Compare Register High Byte bit 4
.equ   OCR1AH5      = 5    ; Timer/Counter1 Outbut Compare Register High Byte bit 5
.equ   OCR1AH6      = 6    ; Timer/Counter1 Outbut Compare Register High Byte bit 6
.equ   OCR1AH7      = 7    ; Timer/Counter1 Outbut Compare Register High Byte bit 7

; OCR1BL - Timer/Counter1 Output Compare Register Low Byte
.equ   OCR1AL0      = 0    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 0
.equ   OCR1AL1      = 1    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 1
.equ   OCR1AL2      = 2    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 2
.equ   OCR1AL3      = 3    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 3
.equ   OCR1AL4      = 4    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 4
.equ   OCR1AL5      = 5    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 5
.equ   OCR1AL6      = 6    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 6
.equ   OCR1AL7      = 7    ; Timer/Counter1 Outbut Compare Register Low Byte Bit 7


; ***** WATCHDOG *********************
; WDTCR - Watchdog Timer Control Register
.equ   WDTCSR = WDTCR      ; For compatibility
.equ   WDP0   = 0    ; Watch Dog Timer Prescaler bit 0
.equ   WDP1   = 1    ; Watch Dog Timer Prescaler bit 1
.equ   WDP2   = 2    ; Watch Dog Timer Prescaler bit 2
.equ   WDE    = 3    ; Watch Dog Enable
.equ   WDCE   = 4    ; Watchdog Change Enable
.equ   WDTOE  = WDCE ; For compatibility
.equ   WDP3   = 5    ; Watchdog Timer Prescaler Bit 3
```

```
.equ   WDIE   = 6    ; Watchdog Timeout Interrupt Enable
.equ   WDIF   = 7    ; Watchdog Timeout Interrupt Flag


; ***** EXTERNAL_INTERRUPT ***********
; GIMSK - General Interrupt Mask Register
.equ   PCIE   = 5    ;
.equ   INT0   = 6    ; External Interrupt Request 0 Enable
.equ   INT1   = 7    ; External Interrupt Request 1 Enable

; EIFR - Extended Interrupt Flag Register
.equ   GIFR   = EIFR ; For compatibility
.equ   PCIF   = 5    ;
.equ   INTF0  = 6    ; External Interrupt Flag 0
.equ   INTF1  = 7    ; External Interrupt Flag 1


; ***** USART ************************
; UDR - USART I/O Data Register
.equ   UDR0   = 0    ; USART I/O Data Register bit 0
.equ   UDR1   = 1    ; USART I/O Data Register bit 1
.equ   UDR2   = 2    ; USART I/O Data Register bit 2
.equ   UDR3   = 3    ; USART I/O Data Register bit 3
.equ   UDR4   = 4    ; USART I/O Data Register bit 4
.equ   UDR5   = 5    ; USART I/O Data Register bit 5
.equ   UDR6   = 6    ; USART I/O Data Register bit 6
.equ   UDR7   = 7    ; USART I/O Data Register bit 7

; UCSRA - USART Control and Status Register A
.equ   USR    = UCSRA      ; For compatibility
.equ   MPCM   = 0    ; Multi-processor Communication Mode
.equ   U2X    = 1    ; Double the USART Transmission Speed
.equ   UPE    = 2    ; USART Parity Error
.equ   PE     = UPE  ; For compatibility
.equ   DOR    = 3    ; Data overRun
.equ   FE     = 4    ; Framing Error
```

```
.equ   UDRE   = 5    ; USART Data Register Empty
.equ   TXC    = 6    ; USART Transmitt Complete
.equ   RXC    = 7    ; USART Receive Complete

; UCSRB - USART Control and Status Register B
.equ   UCR    = UCSRB      ; For compatibility
.equ   TXB8   = 0    ; Transmit Data Bit 8
.equ   RXB8   = 1    ; Receive Data Bit 8
.equ   UCSZ2  = 2    ; Character Size
.equ   CHR9   = UCSZ2      ; For compatibility
.equ   TXEN   = 3    ; Transmitter Enable
.equ   RXEN   = 4    ; Receiver Enable
.equ   UDRIE  = 5    ; USART Data register Empty Interrupt Enable
.equ   TXCIE  = 6    ; TX Complete Interrupt Enable
.equ   RXCIE  = 7    ; RX Complete Interrupt Enable

; UCSRC - USART Control and Status Register C
.equ   UCPOL  = 0    ; Clock Polarity
.equ   UCSZ0  = 1    ; Character Size Bit 0
.equ   UCSZ1  = 2    ; Character Size Bit 1
.equ   USBS   = 3    ; Stop Bit Select
.equ   UPM0   = 4    ; Parity Mode Bit 0
.equ   UPM1   = 5    ; Parity Mode Bit 1
.equ   UMSEL  = 6    ; USART Mode Select

.equ   UBRR   = UBRRL      ; For compatibility

; ***** ANALOG_COMPARATOR ************
; ACSR - Analog Comparator Control And Status Register
.equ   ACIS0  = 0    ; Analog Comparator Interrupt Mode Select bit 0
.equ   ACIS1  = 1    ; Analog Comparator Interrupt Mode Select bit 1
.equ   ACIC   = 2    ;
.equ   ACIE   = 3    ; Analog Comparator Interrupt Enable
.equ   ACI    = 4    ; Analog Comparator Interrupt Flag
.equ   ACO    = 5    ; Analog Compare Output
.equ   ACBG   = 6    ; Analog Comparator Bandgap Select
```

```
.equ   ACD    = 7    ; Analog Comparator Disable

; DIDR - Digital Input Disable Register 1
.equ   AIN0D  = 0    ; AIN0 Digital Input Disable
.equ   AIN1D  = 1    ; AIN1 Digital Input Disable


; ***** PORTD ************************
; PORTD - Data Register, Port D
.equ   PORTD0 = 0    ;
.equ   PD0    = 0    ; For compatibility
.equ   PORTD1 = 1    ;
.equ   PD1    = 1    ; For compatibility
.equ   PORTD2 = 2    ;
.equ   PD2    = 2    ; For compatibility
.equ   PORTD3 = 3    ;
.equ   PD3    = 3    ; For compatibility
.equ   PORTD4 = 4    ;
.equ   PD4    = 4    ; For compatibility
.equ   PORTD5 = 5    ;
.equ   PD5    = 5    ; For compatibility
.equ   PORTD6 = 6    ;
.equ   PD6    = 6    ; For compatibility

; DDRD
.equ   DDD0   = 0    ;
.equ   DDD1   = 1    ;
.equ   DDD2   = 2    ;
.equ   DDD3   = 3    ;
.equ   DDD4   = 4    ;
.equ   DDD5   = 5    ;
.equ   DDD6   = 6    ;

; PIND - Input Pins, Port D
.equ   PIND0  = 0    ;
.equ   PIND1  = 1    ;
```

```
.equ   PIND2  = 2    ;
.equ   PIND3  = 3    ;
.equ   PIND4  = 4    ;
.equ   PIND5  = 5    ;
.equ   PIND6  = 6    ;


; ***** EEPROM ***********************
; EEAR - EEPROM Read/Write Access
.equ   EEARL  = EEAR ; For compatibility
.equ   EEAR0  = 0    ; EEPROM Read/Write Access bit 0
.equ   EEAR1  = 1    ; EEPROM Read/Write Access bit 1
.equ   EEAR2  = 2    ; EEPROM Read/Write Access bit 2
.equ   EEAR3  = 3    ; EEPROM Read/Write Access bit 3
.equ   EEAR4  = 4    ; EEPROM Read/Write Access bit 4
.equ   EEAR5  = 5    ; EEPROM Read/Write Access bit 5
.equ   EEAR6  = 6    ; EEPROM Read/Write Access bit 6

; EEDR - EEPROM Data Register
.equ   EEDR0  = 0    ; EEPROM Data Register bit 0
.equ   EEDR1  = 1    ; EEPROM Data Register bit 1
.equ   EEDR2  = 2    ; EEPROM Data Register bit 2
.equ   EEDR3  = 3    ; EEPROM Data Register bit 3
.equ   EEDR4  = 4    ; EEPROM Data Register bit 4
.equ   EEDR5  = 5    ; EEPROM Data Register bit 5
.equ   EEDR6  = 6    ; EEPROM Data Register bit 6
.equ   EEDR7  = 7    ; EEPROM Data Register bit 7

; EECR - EEPROM Control Register
.equ   EERE   = 0    ; EEPROM Read Enable
.equ   EEPE   = 1    ; EEPROM Write Enable
.equ   EEWE   = EEPE ; For compatibility
.equ   EEMPE  = 2    ; EEPROM Master Write Enable
.equ   EEMWE  = EEMPE      ; For compatibility
.equ   EERIE  = 3    ; EEProm Ready Interrupt Enable
.equ   EEPM0  = 4    ;
```

```
.equ   EEPM1  = 5    ;


; ***** PORTA ************************
; PORTA - Port A Data Register
.equ   PORTA0 = 0    ; Port A Data Register bit 0
.equ   PA0    = 0    ; For compatibility
.equ   PORTA1 = 1    ; Port A Data Register bit 1
.equ   PA1    = 1    ; For compatibility
.equ   PORTA2 = 2    ; Port A Data Register bit 2
.equ   PA2    = 2    ; For compatibility

; DDRA - Port A Data Direction Register
.equ   DDA0   = 0    ; Data Direction Register, Port A, bit 0
.equ   DDA1   = 1    ; Data Direction Register, Port A, bit 1
.equ   DDA2   = 2    ; Data Direction Register, Port A, bit 2

; PINA - Port A Input Pins
.equ   PINA0  = 0    ; Input Pins, Port A bit 0
.equ   PINA1  = 1    ; Input Pins, Port A bit 1
.equ   PINA2  = 2    ; Input Pins, Port A bit 2


; ***** CPU **************************
; SREG - Status Register
.equ   SREG_C = 0    ; Carry Flag
.equ   SREG_Z = 1    ; Zero Flag
.equ   SREG_N = 2    ; Negative Flag
.equ   SREG_V = 3    ; Two's Complement Overflow Flag
.equ   SREG_S = 4    ; Sign Bit
.equ   SREG_H = 5    ; Half Carry Flag
.equ   SREG_T = 6    ; Bit Copy Storage
.equ   SREG_I = 7    ; Global Interrupt Enable

; SPMCSR - Store Program Memory Control and Status register
.equ   SPMEN  = 0    ; Store Program Memory Enable
```

```
.equ   PGERS  = 1    ; Page Erase
.equ   PGWRT  = 2    ; Page Write
.equ   RFLB   = 3    ; Read Fuse and Lock Bits
.equ   CTPB   = 4

; MCUCR - MCU Control Register
.equ   ISC00  = 0    ; Interrupt Sense Control 0 bit 0
.equ   ISC01  = 1    ; Interrupt Sense Control 0 bit 1
.equ   ISC10  = 2    ; Interrupt Sense Control 1 bit 0
.equ   ISC11  = 3    ; Interrupt Sense Control 1 bit 1
.equ   SM0    = 4    ; Sleep Mode Select Bit 0
.equ   SM     = SM0  ; For compatibility
.equ   SE     = 5    ; Sleep Enable
.equ   SM1    = 6    ; Sleep Mode Select Bit 1
.equ   PUD    = 7    ; Pull-up Disable

; CLKPR - Clock Prescale Register
.equ   CLKPS0 = 0    ; Clock Prescaler Select Bit 0
.equ   CLKPS1 = 1    ; Clock Prescaler Select Bit 1
.equ   CLKPS2 = 2    ; Clock Prescaler Select Bit 2
.equ   CLKPS3 = 3    ; Clock Prescaler Select Bit 3
.equ   CLKPCE = 7    ; Clock Prescaler Change Enable

; MCUSR - MCU Status register
.equ   PORF   = 0    ; Power-On Reset Flag
.equ   EXTRF  = 1    ; External Reset Flag
.equ   BORF   = 2    ; Brown-out Reset Flag
.equ   WDRF   = 3    ; Watchdog Reset Flag

; OSCCAL - Oscillator Calibration Register
.equ   CAL0   = 0    ; Oscillatro Calibration Value Bit 0
.equ   CAL1   = 1    ; Oscillatro Calibration Value Bit 1
.equ   CAL2   = 2    ; Oscillatro Calibration Value Bit 2
.equ   CAL3   = 3    ; Oscillatro Calibration Value Bit 3
.equ   CAL4   = 4    ; Oscillatro Calibration Value Bit 4
.equ   CAL5   = 5    ; Oscillatro Calibration Value Bit 5
```

```
.equ   CAL6   = 6    ; Oscillatro Calibration Value Bit 6

; GTCCR - General Timer Counter Control Register
.equ   SFIOR  = GTCCR      ; For compatibility
.equ   PSR10  = 0    ;

; PCMSK - Pin-Change Mask register
.equ   PCINT0 = 0    ; Pin-Change Interrupt 0
.equ   PCINT1 = 1    ; Pin-Change Interrupt 1
.equ   PCINT2 = 2    ; Pin-Change Interrupt 2
.equ   PCINT3 = 3    ; Pin-Change Interrupt 3
.equ   PCINT4 = 4    ; Pin-Change Interrupt 4
.equ   PCINT5 = 5    ; Pin-Change Interrupt 5
.equ   PCINT6 = 6    ; Pin-Change Interrupt 6
.equ   PCINT7 = 7    ; Pin-Change Interrupt 7

; GPIOR2 - General Purpose I/O Register 2
.equ   GPIOR20      = 0    ; General Purpose I/O Register 2 bit 0
.equ   GPIOR21      = 1    ; General Purpose I/O Register 2 bit 1
.equ   GPIOR22      = 2    ; General Purpose I/O Register 2 bit 2
.equ   GPIOR23      = 3    ; General Purpose I/O Register 2 bit 3
.equ   GPIOR24      = 4    ; General Purpose I/O Register 2 bit 4
.equ   GPIOR25      = 5    ; General Purpose I/O Register 2 bit 5
.equ   GPIOR26      = 6    ; General Purpose I/O Register 2 bit 6
.equ   GPIOR27      = 7    ; General Purpose I/O Register 2 bit 7

; GPIOR1 - General Purpose I/O Register 1
.equ   GPIOR10      = 0    ; General Purpose I/O Register 1 bit 0
.equ   GPIOR11      = 1    ; General Purpose I/O Register 1 bit 1
.equ   GPIOR12      = 2    ; General Purpose I/O Register 1 bit 2
.equ   GPIOR13      = 3    ; General Purpose I/O Register 1 bit 3
.equ   GPIOR14      = 4    ; General Purpose I/O Register 1 bit 4
.equ   GPIOR15      = 5    ; General Purpose I/O Register 1 bit 5
.equ   GPIOR16      = 6    ; General Purpose I/O Register 1 bit 6
.equ   GPIOR17      = 7    ; General Purpose I/O Register 1 bit 7
```

```
; GPIOR0 - General Purpose I/O Register 0
.equ   GPIOR00      = 0    ; General Purpose I/O Register 0 bit 0
.equ   GPIOR01      = 1    ; General Purpose I/O Register 0 bit 1
.equ   GPIOR02      = 2    ; General Purpose I/O Register 0 bit 2
.equ   GPIOR03      = 3    ; General Purpose I/O Register 0 bit 3
.equ   GPIOR04      = 4    ; General Purpose I/O Register 0 bit 4
.equ   GPIOR05      = 5    ; General Purpose I/O Register 0 bit 5
.equ   GPIOR06      = 6    ; General Purpose I/O Register 0 bit 6
.equ   GPIOR07      = 7    ; General Purpose I/O Register 0 bit 7


; ***** USI *************************
; USIDR - USI Data Register
.equ   USIDR0 = 0    ; USI Data Register bit 0
.equ   USIDR1 = 1    ; USI Data Register bit 1
.equ   USIDR2 = 2    ; USI Data Register bit 2
.equ   USIDR3 = 3    ; USI Data Register bit 3
.equ   USIDR4 = 4    ; USI Data Register bit 4
.equ   USIDR5 = 5    ; USI Data Register bit 5
.equ   USIDR6 = 6    ; USI Data Register bit 6
.equ   USIDR7 = 7    ; USI Data Register bit 7

; USISR - USI Status Register
.equ   USICNT0      = 0    ; USI Counter Value Bit 0
.equ   USICNT1      = 1    ; USI Counter Value Bit 1
.equ   USICNT2      = 2    ; USI Counter Value Bit 2
.equ   USICNT3      = 3    ; USI Counter Value Bit 3
.equ   USIDC  = 4    ; Data Output Collision
.equ   USIPF  = 5    ; Stop Condition Flag
.equ   USIOIF = 6    ; Counter Overflow Interrupt Flag
.equ   USISIF = 7    ; Start Condition Interrupt Flag

; USICR - USI Control Register
.equ   USITC  = 0    ; Toggle Clock Port Pin
.equ   USICLK = 1    ; Clock Strobe
.equ   USICS0 = 2    ; USI Clock Source Select Bit 0
```

```
.equ   USICS1 = 3    ; USI Clock Source Select Bit 1
.equ   USIWM0 = 4    ; USI Wire Mode Bit 0
.equ   USIWM1 = 5    ; USI Wire Mode Bit 1
.equ   USIOIE = 6    ; Counter Overflow Interrupt Enable
.equ   USISIE = 7    ; Start Condition Interrupt Enable



; ***** LOCKSBITS **********************************************************
.equ   LB1    = 0    ; Lockbit
.equ   LB2    = 1    ; Lockbit


; ***** FUSES ***************************************************************
; LOW fuse bits
.equ   CKSEL0 = 0    ; Select Clock Source
.equ   CKSEL1 = 1    ; Select Clock Source
.equ   CKSEL2 = 2    ; Select Clock Source
.equ   CKSEL3 = 3    ; Select Clock Source
.equ   SUT0   = 4    ; Select start-up time
.equ   SUT1   = 5    ; Select start-up time
.equ   CKOUT  = 6    ; Clock output
.equ   CKDIV8 = 7    ; Divide clock by 8

; HIGH fuse bits
.equ   BODLEVEL0    = 0    ; Brown-out Detector trigger level
.equ   BODLEVEL1    = 1    ; Brown-out Detector trigger level
.equ   BODLEVEL2    = 2    ; Brown-out Detector trigger level
.equ   EESAVE = 3    ; EEPROM memory is preserved through chip erase
.equ   WDTON  = 4    ; Watchdog Timer Always On
.equ   SPIEN  = 5    ; Enable Serial programming and Data Downloading
.equ   DWEN   = 6    ; debugWIRE Enable
.equ   RSTDISBL     = 7    ; External reset disable

; EXTENDED fuse bits
.equ   SELFPRGEN    = 0    ; Self Programming Enable
```

```
; ***** CPU REGISTER DEFINITIONS ***************************************
.def   XH     = r27
.def   XL     = r26
.def   YH     = r29
.def   YL     = r28
.def   ZH     = r31
.def   ZL     = r30



; ***** DATA MEMORY DECLARATIONS ****************************************
.equ   FLASHEND      = 0x03ff      ; Note: Word address
.equ   IOEND  = 0x003f
.equ   SRAM_START    = 0x0060
.equ   SRAM_SIZE     = 128
.equ   RAMEND = 0x00df
.equ   XRAMEND       = 0x0000
.equ   E2END  = 0x007f
.equ   EEPROMEND     = 0x007f
.equ   EEADRBITS     = 7
#pragma AVRPART MEMORY PROG_FLASH 2048
#pragma AVRPART MEMORY EEPROM 128
#pragma AVRPART MEMORY INT_SRAM SIZE 128
#pragma AVRPART MEMORY INT_SRAM START_ADDR 0x60



; ***** BOOTLOADER DECLARATIONS *****************************************
.equ   NRWW_START_ADDR      = 0x0
.equ   NRWW_STOP_ADDR       = 0x3ff
.equ   RWW_START_ADDR       = 0x0
.equ   RWW_STOP_ADDR= 0x0
.equ   PAGESIZE      = 16
```

```
; ***** INTERRUPT VECTORS ********************************************
.equ   INT0addr      = 0x0001     ; External Interrupt Request 0
.equ   INT1addr      = 0x0002     ; External Interrupt Request 1
.equ   ICP1addr      = 0x0003     ; Timer/Counter1 Capture Event
.equ   OC1Aaddr      = 0x0004     ; Timer/Counter1 Compare Match A
.equ   OC1addr       = 0x0004     ; For compatibility
.equ   OVF1addr      = 0x0005     ; Timer/Counter1 Overflow
.equ   OVF0addr      = 0x0006     ; Timer/Counter0 Overflow
.equ   URXCaddr      = 0x0007     ; USART, Rx Complete
.equ   URXC0addr     = 0x0007     ; For compatibility
.equ   UDREaddr      = 0x0008     ; USART Data Register Empty
.equ   UDRE0addr     = 0x0008     ; For compatibility
.equ   UTXCaddr      = 0x0009     ; USART, Tx Complete
.equ   UTXC0addr     = 0x0009     ; For compatibility
.equ   ACIaddr       = 0x000a     ; Analog Comparator
.equ   PCIaddr       = 0x000b     ;
.equ   OC1Baddr      = 0x000c     ;
.equ   OC0Aaddr      = 0x000d     ;
.equ   OC0Baddr      = 0x000e     ;
.equ   USI_STARTaddr = 0x000f     ; USI Start Condition
.equ   USI_OVFaddr   = 0x0010     ; USI Overflow
.equ   ERDYaddr      = 0x0011     ;
.equ   WDTaddr       = 0x0012     ; Watchdog Timer Overflow

.equ   INT_VECTORS_SIZE    = 19   ; size in words

#endif  /* _TN2313DEF_INC_ */

; ***** END OF FILE ****************************************************
```