BIT ADDRESSABILITY

Hoda Roodaki

hroodaki@kntu.ac.ir

Manipulating the bits of general purpose registers

- ***Copying* a *bit***
- One of the bits in the SREG (status register) is named T (temporary), which is used when we want to copy a bit of data from one GPR to another GPR.
- The BST (Bit Store from register to T) and BLD (Bit Load from T to register) instructions can be used to copy a bit of a register to a specific bit of another register.
  - The "BST Rd, b" instruction stores bit b from Rd to the T flag.
  - The “BLD Rr, b" instruction copies the T flag to bit bin register Rr.

```
BST R17,3   ;store bit 3 from R17 to the T flag
BLD R19,5   ;copy the T flag to bit 5 in R19
```

Example 6-21

A switch is connected to pin PB4. Write a program to get the status of the switch and save it in D0 of internal RAM location 0x200.

Solution:

```
.EQU MYREG = 0x200      ;set aside loc 0x200
     CBI   DDRB,0       ;make PB0 an input
     IN    R17,PINB     ;R17 = PINB
     BST   R17,4        ;T = PINB.4
     LDI   R16,0x00     ;R16 = 0
     BLD   R16,0        ;R16.0 = T
     STS   MYREG,R16    ;copy R16 to location $200
HERE: JMP  HERE
```

Manipulating the bits of general purpose registers

- ***Checking a bit***
  - To see if a bit of a general purpose register is set or cleared, we can use the SBRS (Skip next instruction if Bit in Register is Set) and SBRC (Skip next instruction if Bit in register is Cleared) instructions.
    - The SBRS instruction tests a bit of a register and skips the instruction right below it if the bit is HIGH.

Example 6-22

A switch is connected to pin PC7. Using the SBRS instruction, write a program to check the status of the switch and perform the following:
(a) If switch = 0, send letter 'N' to Port D.
(b) If switch = 1, send letter 'Y' to Port D.

Solution:

```
.INCLUDE "M32DEF.INC"   ;include a file according to the IC you use
      CBI   DDRC,7      ;make PC7 an input
      LDI   R16,0xFF
      OUT   DDRD,R16    ;make Port D an output port
AGAIN:IN    R20,PINC    ;R20 = PINC
      SBRS  R20,7       ;skip next line if Bit PC7 is set
      RJMP  OVER        ;it must be LOW
      LDI   R16,'Y'     ;R16 = 'Y' ASCII letter Y
      OUT   PORTD,R16   ;issue R16 to PD
      RJMP  AGAIN       ;we could use JMP instead
OVER: LDI   R16,'N'     ;R16 = 'N' ASCII letter N
      OUT   PORTD,R16   ;issue R16 to PORTD
      RJMP  AGAIN       ;we can use JMP too
```

Status register bit-addressability

- To set a flag we can use the BSET instruction

  BSET s

- To clear a flag we can use the BCLR (flag bit clear) instruction

  BCLR s

  – *Where s* is a number between 0 and 7, and represents the bit to be set in the status register.

Status register bit-addressability

Instruction	Action		Instruction	Action	
SEC	Set Carry	C = 1	CLC	Clear Carry	C = 0
SEZ	Set Zero	Z = 1	CLZ	Clear Zero	Z = 0
SEN	Set Negative	N = 1	CLN	Clear Negative	N = 0
SEV	Set overflow	V = 1	CLV	Clear overflow	V = 0
SES	Set Sign	S = 1	CLS	Clear Sign	S = 0
SEH	Set Half carry	H = 1	CLH	Clear Half carry	H = 0
SET	Set Temporary	T = 1	CLT	Clear Temporary	T = 0
SEI	Set Interrupt	I = 1	CLI	Clear Interrupt	I = 0

Internal RAM bit-addressability

- The internal RAM is not bit-addressable.
  - In order to manipulate a bit of the internal RAM location, you should bring it into the general purpose register and then manipulate it.

Example 6-26

Write a program to see if the internal RAM location $195 contains an even value. If so, send it to Port B. If not, make it even and then send it to Port B.

Solution 1:

```
.EQU   MYREG = 0x195           ;set aside loc 0x195
       LDI   R16,0xFF
       OUT   DDRB,R16          ;make Port B an output port
AGAIN:LDS    R16,MYREG
       SBRS  R16,0             ;bit test D0, skip if set
       RJMP  OVER              ;it must be LOW
       CBR   R16,0b00000001    ;clear bit D0 = 0
OVER: OUT    PORTB,R16         ;copy it to Port B
       JMP   AGAIN             ;we can use RJMP too
```

Solution 2:

```
.EQU   MYREG = 0x195           ;set aside loc 0x195
       LDI   R16,0xFF
       OUT   DDRB,R16          ;make Port B an output port
AGAIN:LDS    R16,MYREG
       CBR   R16,0b00000001    ;clear bit D0 = 0
OVER: OUT    PORTB,R16         ;copy it to Port B
       JMP   AGAIN             ;we can use RJMP too
```

Example 6-27

Write a program to see if the internal RAM location $137 contains an even value. If so, write 0x55 into location $200. If not, write 0x63 into location $200.

Solution:

```
.EQU   MYREG = 0x137      ;set aside location 0x137
.EQU   RESULT= 0x200
       LDS    R16,MYREG
       SBRC   R16,0        ;skip if clear Bit D0 of R16 register is clr
       RJMP   OVER         ;it is odd
       LDI    R16,0x55
       STS    RESULT,R16
       RJMP   HERE
OVER:  LDI    R16,0x63
       STS    RESULT,R16
HERE:  RJMP   HERE
```