

Principios de mecatrónica

Clase 4

SDI-11561-004, SDI-11561-002

Algunos ejemplos

Ejemplo1 .

```
LDI R16, 0x99  
STS 0x212, R16  
LDI R16, 0x85  
STS 0x213, R16  
LDI R16, 0x3F  
STS 0x214, R16  
LDI R16, 0x63
```

Ejemplo2.

```
LDI R20, 5  
LDI R21, 2  
ADD R20, R21  
ADD R20, R21  
STS 0x120, R20
```

1. Usando instrucciones con la memoria de datos (Registros I/O)

Instrucción IN

La instrucción *IN* llama al CPU para cargar un byte de un registro *I/O* al GPR. Después esta es ejecutada; el GPR tendrá el mismo valor como el registro *I/O*.

Por ejemplo, la siguiente instrucción copiará el contenido de la localidad 16 (en hexadecimal) de la memoria *I/O* dentro del registro 20.

IN R20, 0x16

En la siguiente figura cada localidad en la memoria *I/O* tiene dos direcciones: dirección *I/O* y dirección de datos de memoria.

Address		Name
Mem.	I/O	
\$20	\$00	TWBR
\$21	\$01	TWSR
\$22	\$02	TWAR
\$23	\$03	TWDR
\$24	\$04	ADCL
\$25	\$05	ADCH
\$26	\$06	ADCSRA
\$27	\$07	ADMUX
\$28	\$08	ACSR
\$29	\$09	UBRRL
\$2A	\$0A	UCSRB
\$2B	\$0B	UCSRA
\$2C	\$0C	UDR
\$2D	\$0D	SPCR
\$2E	\$0E	SPSR
\$2F	\$0F	SPDR
\$30	\$10	PIND
\$31	\$11	DDRD
\$32	\$12	PORTD
\$33	\$13	PINC
\$34	\$14	DDRC
\$35	\$15	PORTC

Address		Name
Mem.	I/O	
\$36	\$16	PINB
\$37	\$17	DDRB
\$38	\$18	PORTB
\$39	\$19	PINA
\$3A	\$1A	DDRA
\$3B	\$1B	PORTA
\$3C	\$1C	EECR
\$3D	\$1D	EEDR
\$3E	\$1E	EEARL
\$3F	\$1F	EEARH
\$40	\$20	UBRRC
		UBRRH
\$41	\$21	WDTCR
\$42	\$22	ASSR
\$43	\$23	OCR2
\$44	\$24	TCNT2
\$45	\$25	TCCR2
\$46	\$26	ICR1L
\$47	\$27	ICR1H
\$48	\$28	OCR1BL
\$49	\$29	OCR1BH
\$4A	\$2A	OCR1AL

Address		Name
Mem.	I/O	
\$4B	\$2B	OCR1AH
\$4C	\$2C	TCNT1L
\$4D	\$2D	TCNT1H
\$4E	\$2E	TCCR1B
\$4F	\$2F	TCCR1A
\$50	\$30	SFIOR
\$51	\$31	OCDR
		OSCCAL
\$52	\$32	TCNT0
\$53	\$33	TCCR0
\$54	\$34	MCUCSR
\$55	\$35	MCUCR
\$56	\$36	TWCR
\$57	\$37	SPMCR
\$58	\$38	TIFR
\$59	\$39	TIMSK
\$5A	\$3A	GIFR
\$5B	\$3B	GICR
\$5C	\$3C	OCR0
\$5D	\$3D	SPL
\$5E	\$3E	SPH
\$5F	\$3F	SREG

Para trabajar fácilmente con los registros de *I/O*, es posible usar los nombres de la dirección de los registros. Por ejemplo, la siguiente instrucción carga R19 con el contenido de PIND.

IN R19, PIND

Ejemplo3. Realizar un programa (usando la instrucción *IN*) que sume el contenido de *PIND* a *PINB* y almacene el resultado en la localización 0x300 del dato de memoria

Instrucción OUT

La instrucción *OUT* llama al CPU para almacenar el GRP en el registro *I/O*. Después de ejecutar la instrucción, el registro *I/O* tendrá el mismo valor como el GRP. Por ejemplo, la siguiente instrucción copiara el contenido de *R10* dentro del PORTD (localización 12 de la memoria *I/O*).

OUT PORTD, R10

Es posible ver que en la instrucción *OUT*, los registro *I/O* son referidos por sus direcciones *I/O*

Ejemplo 4. Realizar un programa que copie *0xE6* al registro SPL usando las instrucciones *LDI* y *OUT*.

```
LDI   R20, 0xE6
OUT   SPL, R20
```

Ejemplo 5. Copiar PIND hacia el PORTA usando las instrucciones *IN* y *OUT*

Instrucción MOV

La instrucción *MOV* es usada para copiar datos entre los registros R0-R31 del GPR.

MOV R_d, R_r ;R_d = R_r (copia R_r a R_d)
;R_d y R_r pueden ser cualquier registro del GRPS

Instrucción INC

La instrucción *INC* incrementa el contenido de *R_d* por uno.

Por ejemplo el siguiente programa incrementa el contenido de dato de memoria de la localización $0x430$ por uno.

LDS R20, 0x430

INC R20

STS 0x430, R20

Instrucción SUB

La instrucción *SUB* tiene el siguiente formato

SUB R_d, R_r ;R_d = R_d - R_r

Cómo se realiza la resta de los datos de memoria 0x25 y 0x34 ?

Cómo se resta 5 a la localización 0x300 y el resultado sea almacenado en 0x320 ?

Instrucción DEC

La instrucción *DEC* tiene el siguiente formato

DEC R_d ;R_d = R_d - 1 La instrucción DEC, decremento (resta 1 de

) el contenido de R_d y pone el resultado de regreso dentro del registro R_d .

Por ejemplo

LDI R30, 3

DEC R30

DEC R30

Instrucción COM

La instrucción *COM R_d* complementa (invierte) el contenido de R_d y deja el resultado en el registro R_d . En el siguiente programa, nosotros ponemos $0x55$ dentro del registro $R16$ y después se envía la información al PORTB. Entonces el contenido de $R16$ es complementado, el cual es en hexadecimal AA.

LDI R16, 0x01
OUT PORTB, R16
COM R16
OUT PORTB, R16