1705045 Iftekhar Hakim Ramim

1) It calculates y-2x

The calculation part will be—

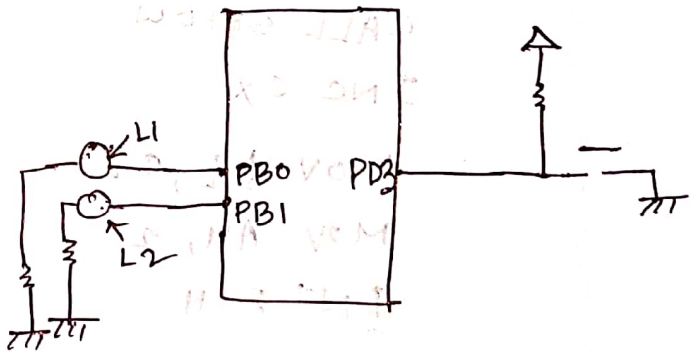
1. change first "MOV AX, X" to "MOV AX, Y"
2. change second "MOV AX, Y" to "MOV AX, X".
3. change "ADD DX, 0" to "ADD DX, '0'"

2)

```
CMP AX, BX
JLE LAST
XCHG AX, BX
LAST:
ADD BX, AX
```

(5) Work in falling edge of INT1

```
#include <avr/io.h>
#define F_CPU 1000000
#include <avr/interrupt.h>
volatile unsigned char c= 0b00000001;
ISR(INT1_vect) {
    c^= 0b00000011;
}
int main(void) {
    DDRB = 0xFF;
    MCUCR |= (1<<ISC11);
    MCUCR = 0b00001000;
    GICR = (1<<INT1)
    sei();
    while(1){
        PORTB = c;
    }
    return 0;
}
```



(6) If right justified and ADCH is read, we get only 4 equal steps.

So,

$$\text{max error} = \frac{2.56}{4} = 0.64V$$

(4)

```
; INPUT IN CX=N
;this is the recursive func
SHOW PROC

PUSH AX
PUSH BX
PUSH DX
CMP CX,0
JE LAST
DEC CX
CALL SHOW
INC CX
MOV DL,CL
MOV AH, 2
INT 21H
LAST:
RET
SHOW ENDP
```