

Embedded Systems Lab Experiment 4

# Hardware Interrupts in Microcontroller ATMEGA32

Program an ATmega 32 such that it accepts interrupts through push button using interrupt vector

Jitbitan Baroi  
2-5-2024

# INDIAN INSTITUTE OF TECHNOLOGY

DATE

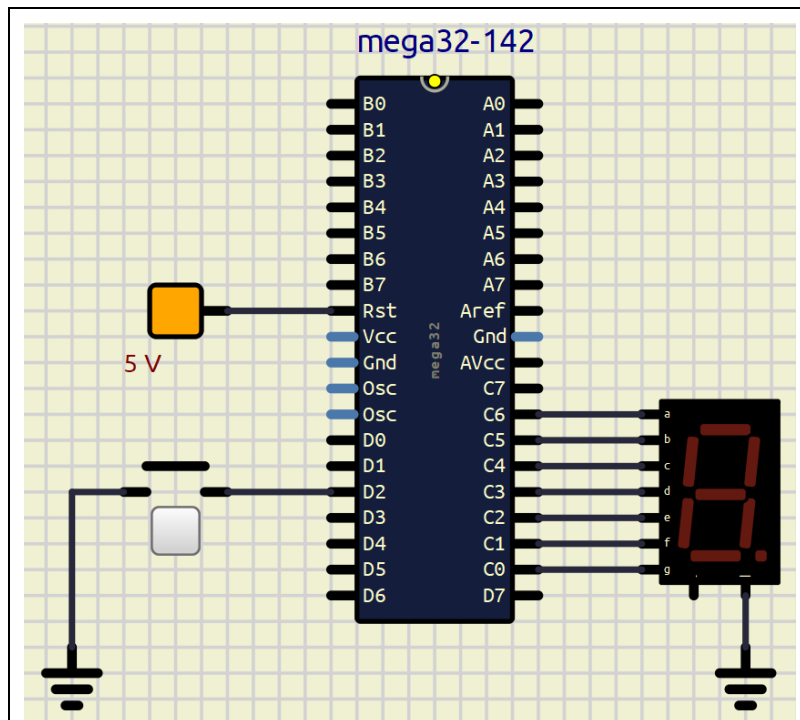
Exp 4 Hardware Interrupts in AT Mega 32

SHEET NO.

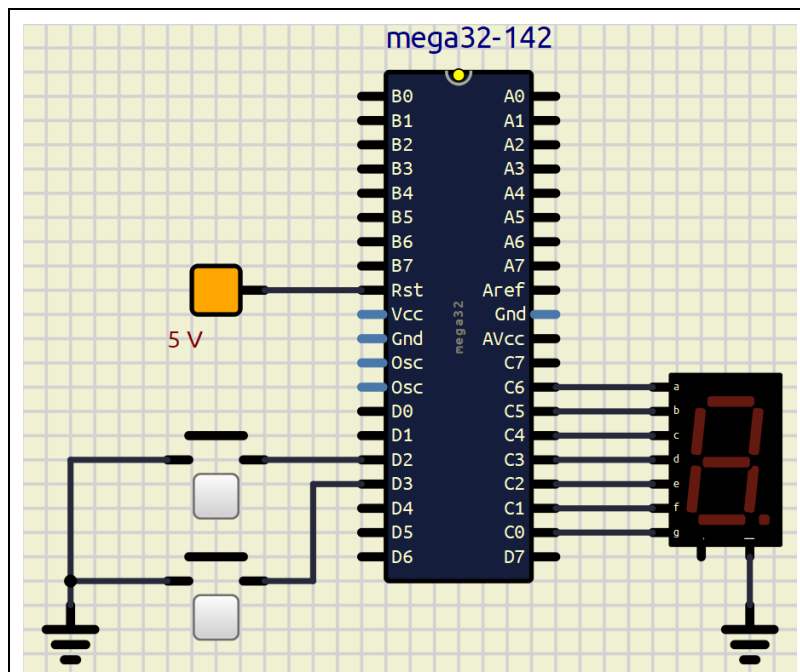
•> Aim: To program AT Mega 32 such that it accepts interrupts through push buttons using interrupt vector.

•> Components reqd.:

Name	Specification	Quantity
1. AT Mega 32. $\mu$ C	—	1
2. Push button	—	2.
3. LED	—	2.
4. $\Phi$ Resistors	100 $\Omega$	8
5. Display	7 segment.	1



*Schematic used for Assignment 1  
Free running UP and DOWN Counter (0 to 9)*

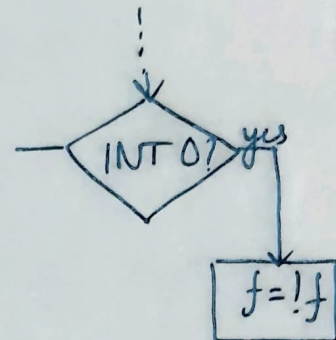
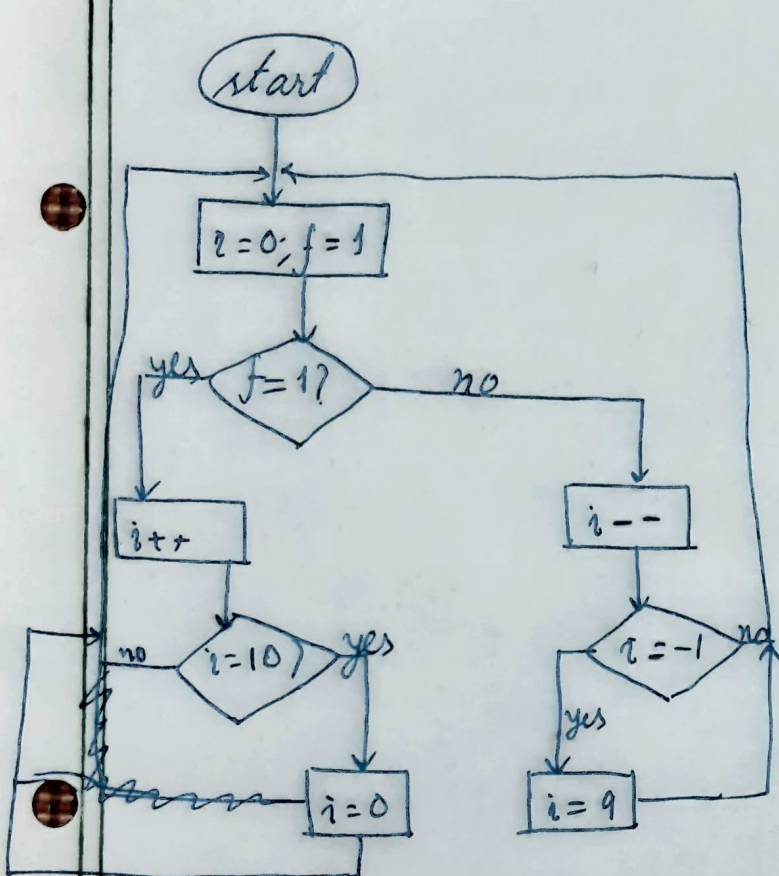


*Schematic used for Assignment 2  
Increment and Decrement using Push Button*

DATE

SHEET NO.

• Flow of code 1:

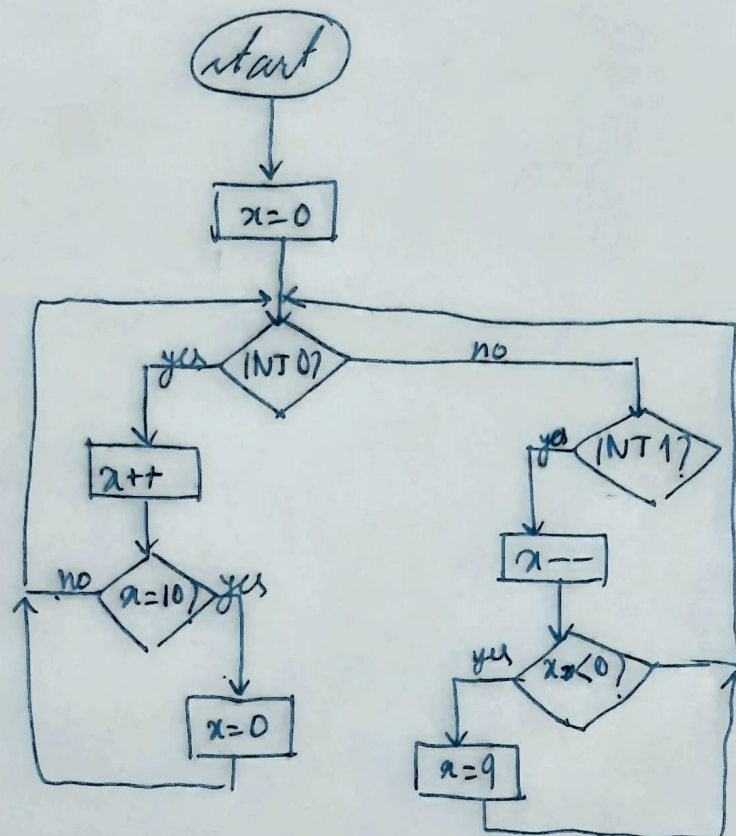




DATE

SHEET NO.

Flow of Code 2:



# INDIAN INSTITUTE OF TECHNOLOGY

DATE

SHEET NO.

## • Assignment 1:

#include <avr/io.h>	→ basic include header file
#define F_CPU 1000000UL	→ include.
#include <util/delay.h>	→
#include <avr/interrupt.h>	→
void init_intr() {	→ an interrupt initializer
GICR  = 0x40;	→ for INTO.
MCUCR  = 0x02;	→ falling edge.
sei(); }	→ set interrupt.
int f = 0;	→ a global flag.
int main(void) {	
int a[] = {....}	→ an array of fragments code.
DDRC = 0xFF;	→ PORTC: output
DDRD = 0x00;	→ PORTD: input
PORTD = 0xFF;	→ PORTD: internal pullup.
init_intr();	
int i = 0;	→ the counter
while (1) {	→ inf loop.
if (f) {	→ if flag is set incr.
i++;	



# INDIAN INSTITUTE OF TECHNOLOGY

DATE

SHEET NO.

<code>if (i == 10) i = 0;</code>	→	<code>if i &gt; 10; i ← 0.</code>
<code>PORTC = a[i];</code>	→	<code>set PORTC according with 7</code>
<code>delay_ms(500);</code>	→	<code>segment code to display i and wait 50.</code>
<code>} else {</code>	→	<code>else. decr i</code>
<code>i--;</code>	→	<code>,</code>
<code>if (i &lt; 0)</code>	→	<code>if i &lt; 0 -&gt; i = 9.</code>
<code>i = 9;</code>	→	<code>set PORTC accordingly and display</code>
<code>PORTC = a[i];</code>	→	<code>i and wait 50.</code>
<code>delay_ms(500); }</code>	→	
<code>ISR(INT0_vect) {</code>	→	<code>if an intr is detected at 0, INT0</code>
<code>f = !f;</code>	→	<code>→ invert the flag.</code>



DATE

SHEET NO

## • Assignment 2 code:

```
#include <avr/io.h>
#include <util/delay.h>
#include <avr/interrupt.h>

void init_intr() {
    GICR = 1 = 0x0001;
    MCUCR |= 0x0A;
    sei();
}

int x = 0;

ISR(INT0_vect) { x++; }
ISR(INT1_vect) { x--; }

int main(void) {
    int a[] = { ... };
    DDRC = 0xFF; DDRD = 0; PORTD = ~0;
    init_intr();
    while(1) {
        PORTC = a[x];
        check();
    }

    void check() {
        if (x > 9) x = 0;
        if (x < 0) x = 9;
    }
}
```

basic include header  
file for delay and interrupt  
for falling edge.  
initialise INT0, INT1 intr.  
a global counter;  
if INT0 intr => incr x.  
if INT1 intr => decr x.  
7 segment codes for all digits  
C and D are op/ip pins (internal pull)  
init interrupt.  
inf loop.  
display x on 7 segment display  
P.R.E.