INT Interrupts

NDSU ECE 376

Lecture #22

Inst: Jake Glower

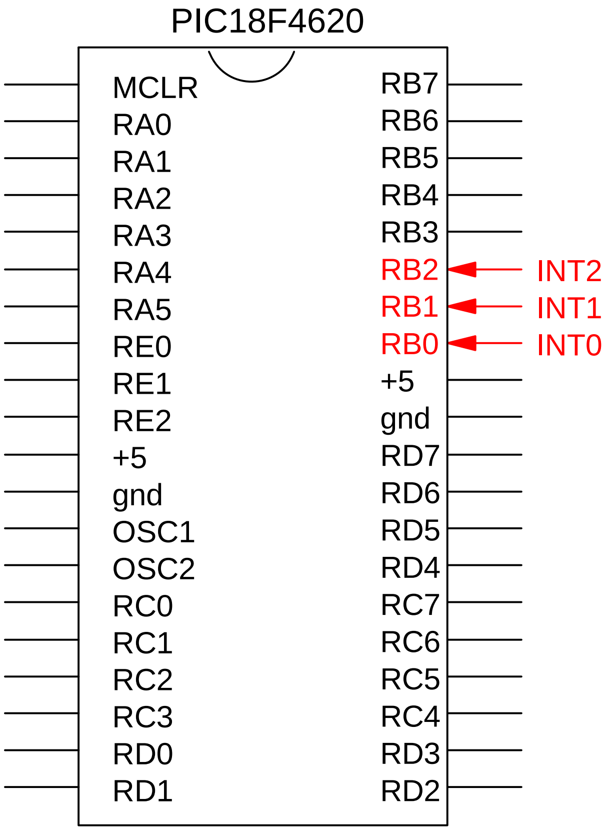
Please visit Bison Academy for corresponding lecture notes, homework sets, and solutions

INT Interrupts:

- Interrupt every rising edge or falling edge
- More efficient way to respond to button presses.

How: Hardware

- Make sure your device outputs 0V / 5V
- Connect to RB0/INT0 pin on the PIC, or
- Connect to RB1/INT1 pin on the PIC, or
- Connect to RB2/INT2 pin on the PIC



How: Software:

1. Set up RB0/RB1/RB2 as an input pin

2. Set up the conditions for the interrupt

- INTEDGx = 1: interrupt on a rising edge
- INTEDGx = 0: interupt on a falling edge

3. Enable the INT interrupt

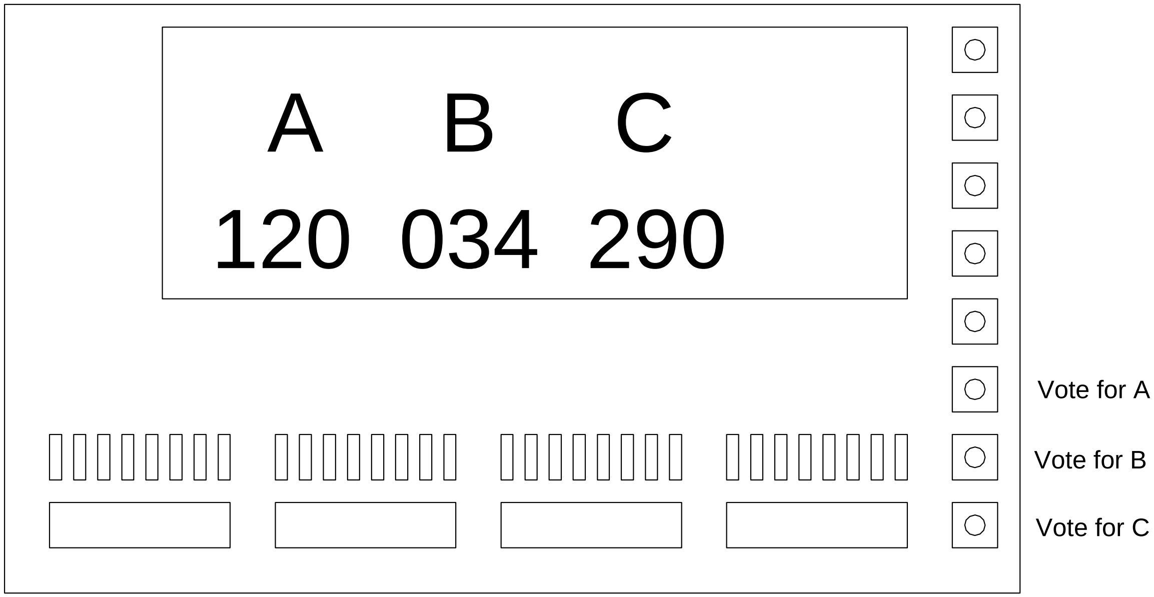
- INTxE = 1:   enable INTx interrupts
- INTxE = 0:   disable INTx interrupts

4. Enable all interrupts:

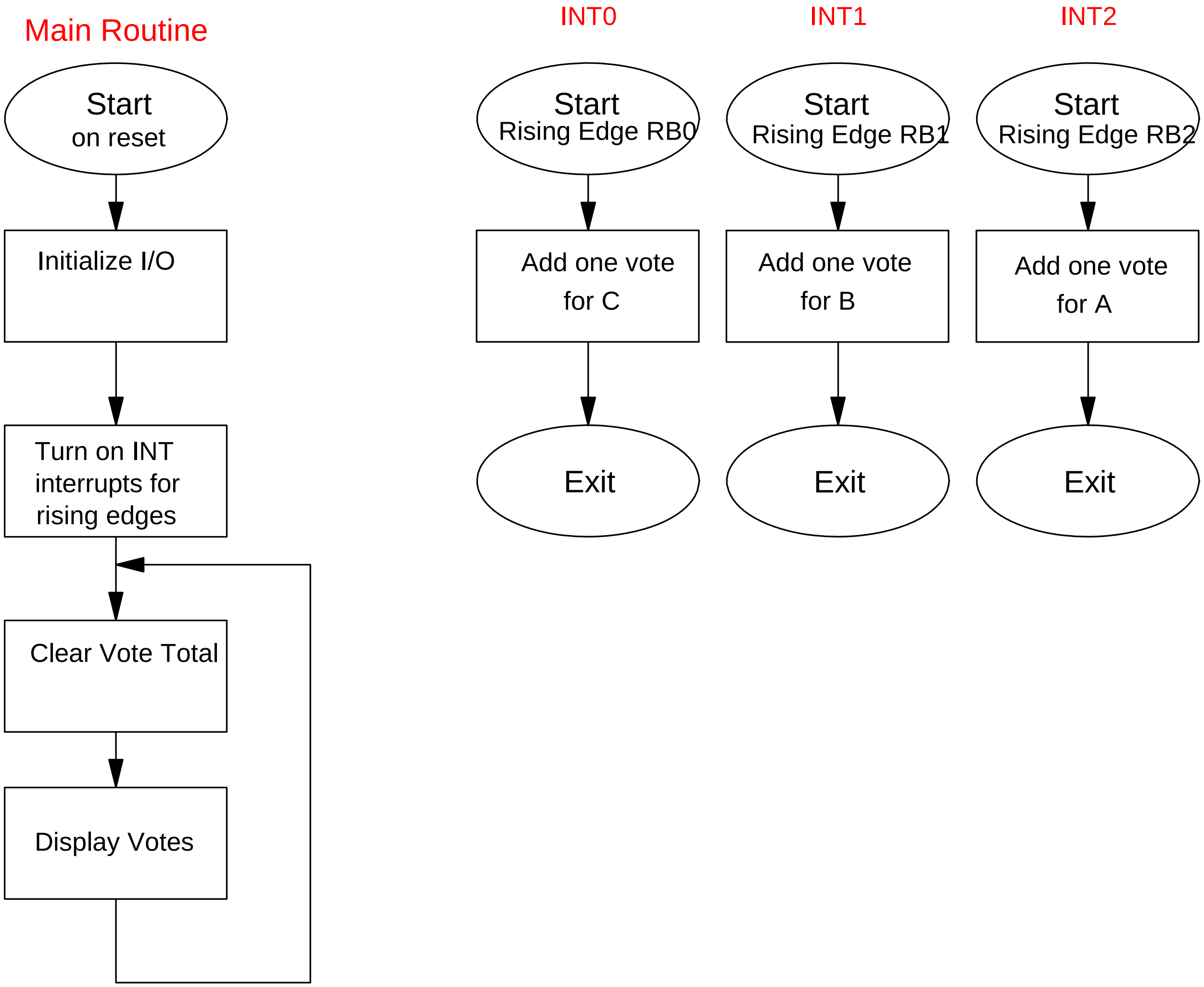
- GIE = 1:   enable all interrupts

Example 1: Voting Machine.

Count how many times you press buttons RB0, RB1, and RB2. Display the total number of button presses on the LCD display.



Software (Vote.C):



Increment three counters inside the interrupt service routine

```
// Global Variables
unsigned int N0, N1, N2;

void interrupt IntServe(void) {
   if (INT0IF) {
      N0 += 1;
      INT0IF = 0;
      }
   if (INT1IF) {
      N1 += 1;
      INT1IF = 0;
      }
   if (INT2IF) {
      N2 += 1;
      INT2IF = 0;
      }
   }
```

Set up INT0, 1, and 2 for rising edge interrupts.

```
// initialize INT0 interrupts for rising edges
   INT0IE = 1;
   TRISB0 = 1;
   INTEDG0 = 1;

// initialize INT1 interrupts for rising edges
   INT1IE = 1;
   TRISB1 = 1;
   INTEDG1 = 1;

// initialize INT2 interrupts for rising edges
   INT2IE = 1;
   TRISB2 = 1;
   INTEDG2 = 1;
```

Display the count total on the LCD display

```
// Main Loop

   while(1) {
      LCD_Move(1,0);   LCD_Out(N0, 0, 3);
      LCD_Move(1,5);   LCD_Out(N1, 0, 3);
      LCD_Move(1,10);  LCD_Out(N2, 0, 3);
      Wait_ms(100);
      }
```

Example 2: Measure the Time a Button was Pressed

- Have Timer0 measure time to 100ns
- Have INT0 record the time the button was pressed (rising edge)
- Have INT1 record the time the button was released (falling edge)

```
unsigned long int TIME, T0, T1;

void interrupt IntServe(void)  {
   if (TMR0IF) {
      TIME += 0x10000;
      TMR0IF = 0;
      }
   if (INT0IF) {
      T0 = TIME + TMR0;
      INT0IF = 0;
      }
   if (INT1IF) {
      T1 = TIME + TMR0;
      dT = T1 - T0;
      INT1IF = 0;
      }
```

Example 3: Random Number Generator

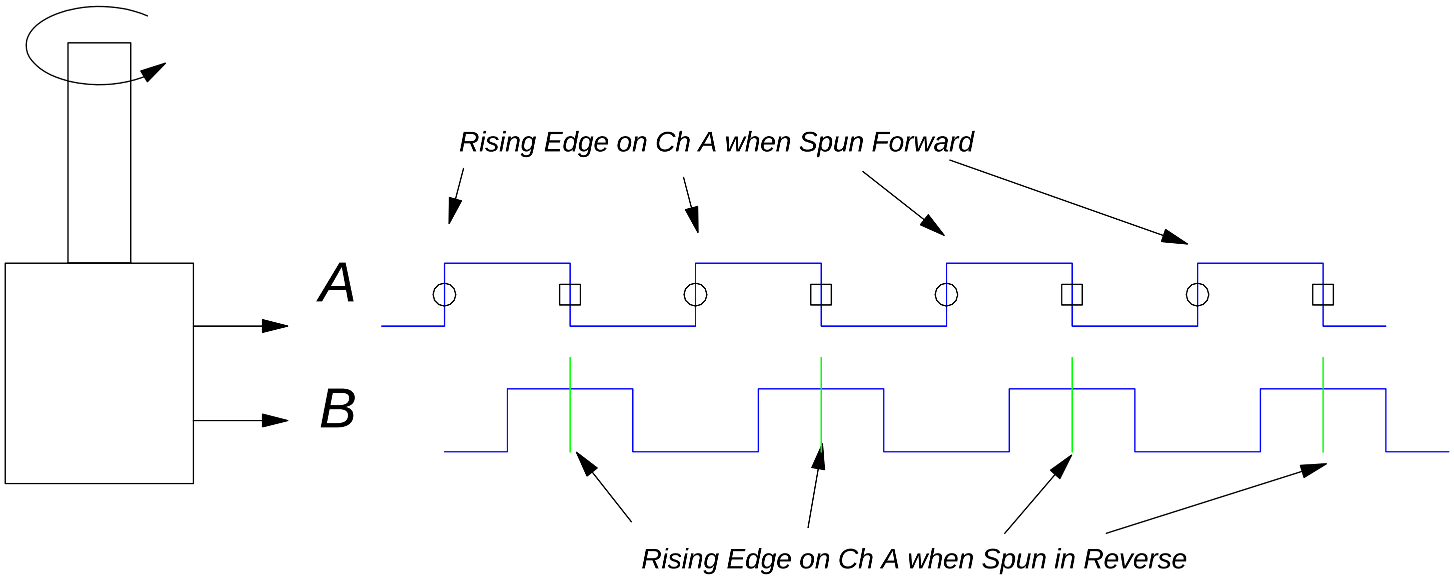
- Have Timer0 keep track of time to 100ns
- The time you press RB0 is the random number

```
unsigned long int TIME, d6;

void interrupt IntServe(void)  {
   if (TMR0IF) {
      TIME += 0x10000;
      TMR0IF = 0;
      }
   if (INT0IF) {
      d6 = TIME % 6;
      INT0IF = 0;
      }
```

Example 4: Optical Encoder

Determine the position of a digital potentiometer


Rising Edge on Ch A when Spun Forward
A
B
Rising Edge on Ch A when Spun in Reverse

Hardware:

You need to count edges on A and B. Connect these to the INTx inputs.

- Connect channel A to RB0 (INT0)
- Connect channel B to RB1 (INT1)

Software:

Code: Just initialize INT0 interrupts for rising egdges:

The interrupt service routine is

```
void interrupt IntServe(void) {
   if (INT0IF) {
      if (RB1) ANGLE += 1; else ANGLE -= 1;
      INT0IF = 0;
      }
   }
```