

EIE3105 Quiz 2

Interrupt

1. Set a Bit

```
PORTB |= (1<<4);  
TCCR0B = (1<<CS02) | (1<<CS01) | (1<<CS00);
```

2. Clear a Bit

```
PORTB &= ~(1<<4);
```

3. Check a Bit

```
PINB & (1<<5);
```

4. Set as Input

```
DDRB = 0x00;  
PORTB = 0xFF;    //Pull Up
```

5. Read Input

```
unsigned char var = PINB;
```

6. Set as Output

```
DDRB = 0xFF;  
PORTB = 0x00;    // Output 0;  
PORTB = 0xFF;    // Output 1;
```

7. Toggle a Bit

```
PORTB ^= (1<<0);  
PORTB ^= (1<<0) | (1<<1) | (1<<2);
```

8. TIMER POLLING

- a. Timer 0: P106
- b. Timer 1: P134
- c. Timer 2: P158

a. TIMER0 Normal Mode

```
TCNT0 = 0xFF;           // 256 - Time Count  
TCCR0A = 0x00;          // Normal Mode  
TCCR0B = 0xFF;          // Pre-scalar  
while((TIFR0 & (1<<TOV0)) == 0);    // Polling  
TCCR0B = 0x00;          // Stop Timer  
TIFR0 = (1<<TOV0);      // Reset Flag
```

b. TIMER0 CTC Mode

```
OCR0A = 0xFF;           // Time Count - 1  
TCCR0A |= (1<<WGM01);   // CTC Mode  
TCCR0B = 0xFF;          // Pre-scalar  
while((TIFR0 & (1<<OCF0A)) == 0);    // Polling  
TCCR0B = 0x00;          // Stop Timer  
TIFR0 = (1<<OCF0A);     // Reset Flag
```

```

c. TIMER1 Normal Mode
TCNT1H = 0xFF;           // High Byte
TCNT1L = 0xFF;           // Low Byte
TCCR1A = 0x00;           // Normal Mode
TCCR1B = 0xFF;           // Pre-scaler
while((TIFR1 & (1<<TOV1) == 0); // Polling
TCCR1B = 0x00;           // Stop
TIFR1 = (1<<TOV1);       // Reset Flag

```

```

d. TIMER1 CTC Mode
OCR1AH = 0xFF;           // High Byte
OCR1AL = 0xFF;           // Low Byte
TCCR1A = 0x00;           // CTC Mode
TCCR1B = 0xFF;           // WGM12 = 1, CTC Mode
while((TIFR1 & (1<<OCF1A) == 0); // Polling
TCCR1B = 0x00;           // Stop
TIFR1 = (1<<OCF1A);      // Reset Flag

```

9. TIMER INTERRUPT

```

a. TIMER1 Interrupt
OCR1AH = 0xFF;
OCR1AL = 0xFF;
TCCR1A = 0x00;
TCCR1B = (1<<WGM12) | (1<<CS12) | (1<<CS10);
TIMSK1 = (1<<OCIE1A);

```

```

b. TIMER0/COUNTER Interrupt
OCR0A = 0xFF;
TCCR0A = (1<<WGM01);
TCCR0B = (1<<CS02) | (1<<CS01) | (1<<CS00);
TIMSK0 = (1<<OCIE0A);

```

10. EXTERNAL INTERRUPT

```

EIMSK = 0xFF;           // INT0/INT1
EICRA = 0xFF;           // Trigger

```