# RTOS Task 1 : Interrupt Handling

**Name : Omkar Sulakhe**

**USN: 2GI17EC065**

**Div: B**

Interrupt is a signal emitted by hardware or software when a process needs immediate attention. It alerts the processor to a high priority process requiring interruption of the current working process.

**Hardware interrupt**

There are many internal components in a microcontroller like timers, counters etc. that require attention of the processor. Since all the devices can’t obtain the attention of the processor at all times, the concept of “Interrupts” comes in to picture. **An Interrupt**, as the name suggests, interrupts the microcontroller from whatever it is doing and draws its attention to perform a special task.

In the event of an interrupt, the source of the interrupt (like a Timer, Counter etc.) sends a special request to the processor called **Interrupt Request (IRQ)** in order to run a special piece of code. The special code or function is called as **Interrupt Service Routine (ISR).**

**Code**

const int ledPin = 13;

const int buttonPin = 2;

int ledToggle = LOW;

void setup()

{

pinMode(ledPin, OUTPUT);

pinMode(buttonPin, INPUT\_PULLUP);

attachInterrupt(digitalPinToInterrupt(buttonPin), button\_ISR, CHANGE);

}

void loop()

{

delay(300);

}

void button\_ISR()

{

ledToggle = !ledToggle;

digitalWrite(ledPin, ledToggle);

}