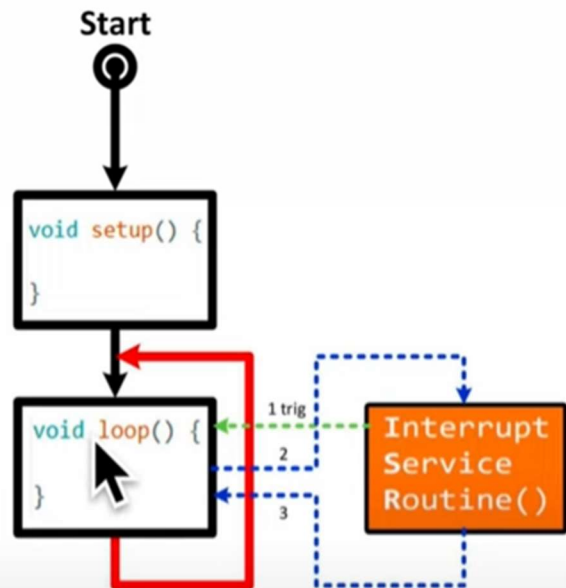
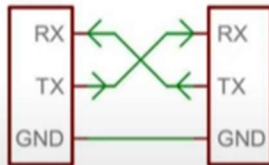
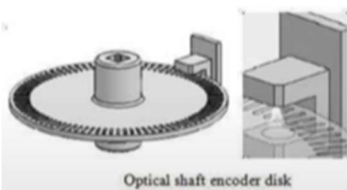


การขัดจังหวะการทำงาน (Interrupts) คืออะไร

การ Interrupts คือ การส่งสัญญาณจากอุปกรณ์ภายนอก (อาจจะเซ็นเซอร์ สวิตช์ หรือเส้นสัญญาณของการติดต่อสื่อสารไม่ว่าจะเป็น UART, I2C หรือ SPI) โดยเรามักจะเรียกสัญญาณนี้ว่า "Interrupt Request" เพื่อไปบอกไมโครคอนโทรลเลอร์ว่าต้องการให้ไมโครคอนโทรลเลอร์หยุดทำงานหลักก่อน เพื่อมาทำงานบางอย่างที่ได้กำหนดเอาไว้ **อย่างด่วนที่สุด**



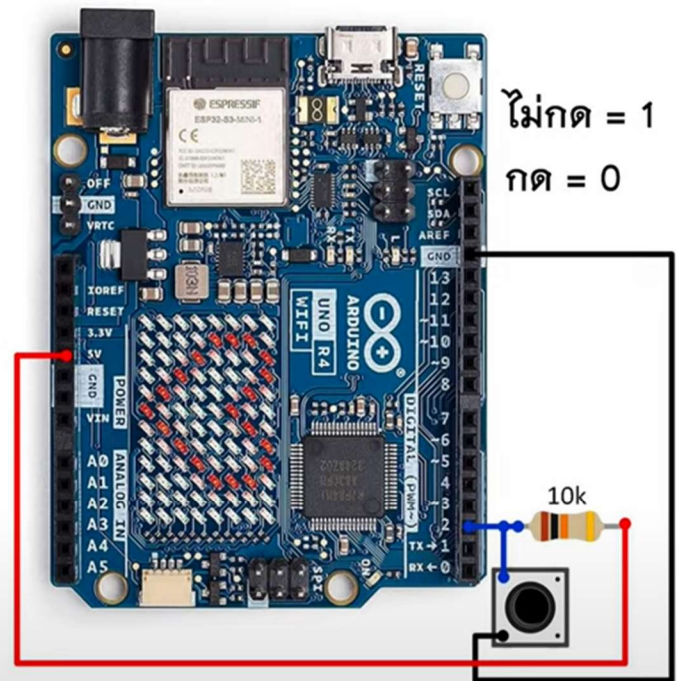
BOARD	DIGITAL PINS USABLE FOR INTERRUPTS	NOTES
Uno Rev3, Nano, Mini, other 328-based	2, 3	
UNO R4 Minima, UNO R4 WiFi	2, 3	
Uno WiFi Rev2, Nano Every	All digital pins	
Mega, Mega2560, MegaADK	2, 3, 18, 19, 20, 21	(pins 20 & 21 are not available to use for interrupts while they are used for I2C communication; they also have external pull-ups that cannot be disabled)
Micro, Leonardo	0, 1, 2, 3, 7	
Zero	0-3, 5-13, A0-A5	Pin 4 cannot be used as an interrupt.
MKR Family boards	0, 1, 4, 5, 6, 7, 8, 9, A1, A2	
Nano 33 IoT	2, 3, 9, 10, 11, 13, A1, A5, A7	
Nano 33 BLE, Nano 33 BLE Sense (rev 1 & 2)	all pins	
Nano RP2040 Connect	0-13, A0-A5	
Nano ESP32	all pins	
GIGA R1 WiFi	all pins	
Due	all digital pins	
101	all digital pins	(Only pins 2, 5, 7, 8, 10, 11, 12, 13 work with CHANGE)

โหมดของการ Interrupts ใน Arduino

- **LOW** to trigger the interrupt whenever the pin is low,
- **CHANGE** to trigger the interrupt whenever the pin changes value
- **RISING** to trigger when the pin goes from low to high,
- **FALLING** for when the pin goes from high to low.

The Due, Zero and MKR1000 boards allow also:

- **HIGH** to trigger the interrupt whenever the pin is high.



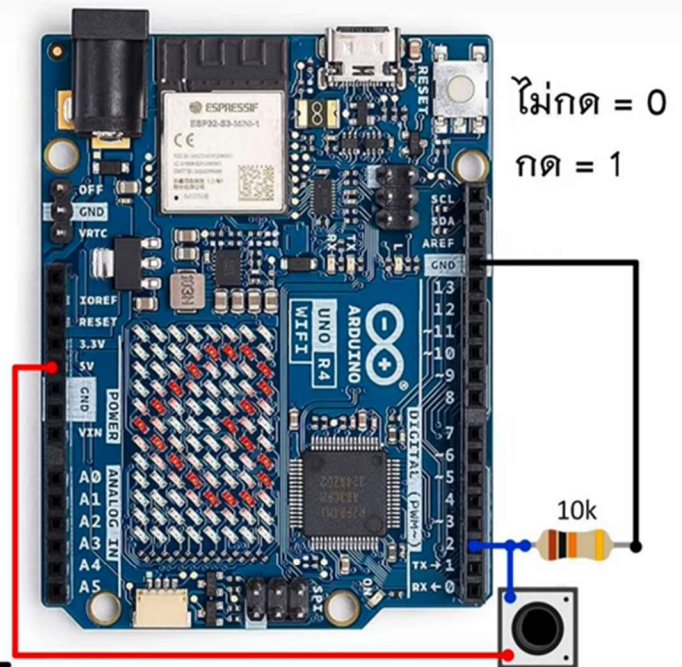
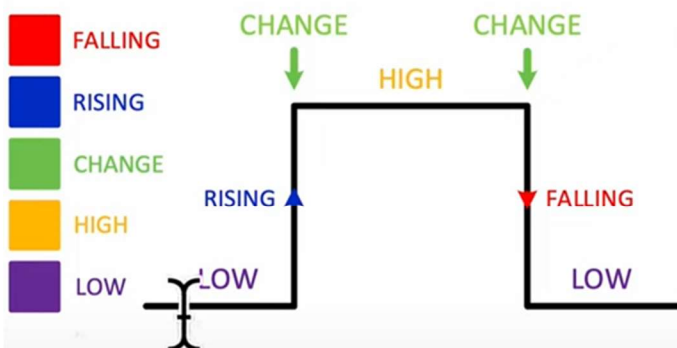
Pull-up Resistor Switch

โหมดของการ Interrupts ใน Arduino

- **LOW** to trigger the interrupt whenever the pin is low,
- **CHANGE** to trigger the interrupt whenever the pin changes value
- **RISING** to trigger when the pin goes from low to high,
- **FALLING** for when the pin goes from high to low.

The Due, Zero and MKR1000 boards allow also:

- **HIGH** to trigger the interrupt whenever the pin is high.



Pull-down Resistor Switch

```

1 #define BUTTON_PIN 3
2
3 volatile bool shouldMoveMotor = false;
4     I
5 void setup() {
6     pinMode(BUTTON_PIN, INPUT);
7     attachInterrupt(
8         digitalPinToInterrupt(BUTTON_PIN), triggerMoveMotor, RISING);
9 }
10
11 void loop() {
12     if (shouldMoveMotor) {
13         shouldMoveMotor = false;
14         moveMotor();
15     }
16 }
17
18 void triggerMoveMotor() {
19     shouldMoveMotor = true;
20 }
21
22 void moveMotor() {
23     // this function may contains code that
24     // requires heavy computation, or takes
25     // a long time to execute
26 }

```

1.การประกาศตัวแปรใด ๆ ที่จะใช้ในฟังก์ชัน ISR ควรจะประกาศเป็นแบบ Global Variable (คือการประกาศตัวแปรนอกทุกฟังก์ชัน)

2.จากกฎข้อที่ 1 ควรใช้ volatile สำหรับการประกาศตัวแปรแบบ Global Variable เพื่อจะทำการแชร์ตัวแปร Global Variable นั้น ๆ ระหว่างฟังก์ชัน setup(), loop(), และ ISR()

```

1 #define BUTTON_PIN 3
2
3 volatile bool shouldMoveMotor = false;
4     I
5 void setup() {
6     pinMode(BUTTON_PIN, INPUT);
7     attachInterrupt(
8         digitalPinToInterrupt(BUTTON_PIN), triggerMoveMotor, RISING);
9 }
10
11 void loop() {
12     if (shouldMoveMotor) {
13         shouldMoveMotor = false;
14         moveMotor();
15     }
16 }
17
18 void triggerMoveMotor() {
19     shouldMoveMotor = true;
20 }
21
22 void moveMotor() {
23     // this function may contains code that
24     // requires heavy computation, or takes
25     // a long time to execute
26 }

```

3.ไม่ควรมีค่าตัวแปรที่จะทำการโยนค่าเข้าหรือออกมาออกฟังก์ชันของ ISR

4.โปรแกรมที่อยู่ ในฟังก์ชัน ISR ควรจะสั้นที่สุดเท่าที่จะทำได้


```

1 #define BUTTON_PIN 3
2
3 volatile bool shouldMoveMotor = false;
4
5 void setup() {
6   pinMode(BUTTON_PIN, INPUT);
7   attachInterrupt(
8     digitalPinToInterrupt(BUTTON_PIN), triggerMoveMotor, RISING);
9 }
10
11 void loop() {
12   if (shouldMoveMotor) {
13     shouldMoveMotor = false;
14     moveMotor();
15   }
16 }
17
18 void triggerMoveMotor() {
19   unsigned long timeNow = millis(); // or micros()
20   // delay(100); --> NO
21   shouldMoveMotor = true;
22 }
23
24 void moveMotor() {
25   // this function may contains code that
26   // requires heavy computation, or takes

```

5. ไม่ควรใช้ฟังก์ชัน
delay(); ในฟังก์ชัน ISR

```

1 #define BUTTON_PIN 3
2
3 volatile bool shouldMoveMotor = false;
4
5 void setup() {
6   Serial.begin(9600);
7   pinMode(BUTTON_PIN, INPUT);
8   attachInterrupt(
9     digitalPinToInterrupt(BUTTON_PIN), triggerMoveMotor, RISING);
10 }
11
12 void loop() {
13   if (shouldMoveMotor) {
14     shouldMoveMotor = false;
15     moveMotor();
16   }
17 }
18
19 void triggerMoveMotor() {
20   // Serial.read(); --> NO
21   // Serial.println("Inside the interrupt!"); --> NO
22   shouldMoveMotor = true;
23 }
24
25 void moveMotor() {
26   // this function may contains code that

```

6. ห้ามใช้ฟังก์ชัน Serial
ต่าง ๆ ในฟังก์ชัน ISR

