

# SYNTHIAM

[synthiam.com](http://synthiam.com)

## Use Arduino for Wheel Encoder Counter

Have your Arduino count motor encoder rotations. Connect the Arduino to your EZ-B UART #0 port to send the encoder counts.

Last Updated: 5/26/2016

There are two sets of wires which will need to be connected:

1. The Arduino to the encoder
2. The Arduino to the EZ-B

### **Encoder**

Connect the encoder wires to your Arduino pins 2 and 3. These pins are interrupt pins. The code will count the number of changes between HIGH and LOW state of these pins.

- pin 2: Encoder #1
- pin 3: Encoder #2

### **EZ-B**

Connect the Arduino's TX and RX ports to the UART #0 of the EZ-B. Consult the ez-b datasheet to locate the UART #0 port.

- Connect EZB UART 0 TX to ARDUINO RX
- Connect EZB UART 0 RX to ARDUINO TX
- Connect Arduino GND to EZ-B GND

# Program Arduino

---

Here is the program which you will upload to your Arduino. The program has 2 interrupt counters for the pins connected to the encoders. The main program will loop until the letter 'a' is received from the EZ-B over the UART. Once the 'a' is received, the Arduino will transmit 2 bytes, which are the two encoder values, respectively. Once the data is transmitted, the count of the 2 bytes is reset.

Resetting the values on each query will make it much easier in EZ-Builder to identify what wheel is spinning quicker/slower.

## Arduino Code:

```
...
```

```
// just reading the encoder, nothing else.
```

```
int encoderValue_A = 0;
```

```
int encoderValue_B = 0;
```

```
void setup() {
```

```
  Serial.begin(9600);
```

```
  pinMode(2, INPUT);
```

```
  pinMode(3, INPUT);
```

```
  attachInterrupt(digitalPinToInterrupt(2), countA, FALLING);
```

```
  attachInterrupt(digitalPinToInterrupt(3), countB, FALLING);
```

```
}
```

```
void loop() {
```

```
  // if there is data to read, read it
```

```
  if (Serial.available() > 0) {
```

```
    // read the incoming data from the ezB
```

```
    int incomingByte = Serial.read();
```

```
    // command 'a' means transmit the current values and reset the counters
```

```
    if (incomingByte == 'a') {
```

```
      // send the values
```

```
      Serial.write(encoderValue_A);
```

```
      Serial.write(encoderValue_B);
```

```
      // reset the values
```

```
      encoderValue_A = 0;
```

```
      encoderValue_B = 0;
```

```
    }
```

```
}
```

```
void countA() {
```

```
  encoderValue_A++;
```

```
}
```

```
void countB() {
```

```
  encoderValue_B++;
```

```
}
```

```
...
```

In EZ-Builder, add a new EZ-Script control.

This code will be pasted into the EZ-Script control. The code will loop and continue to request the interrupt counts of the encoders from the Arduino.

The values of the encoder are in the array \$encoders[0] and \$encoders[1]. You may use these two values globally within the project to identify what the encoder values are.

#### **EZB Code**

```
...`
```

```
uartinit(0, 0, 9600)  
:loop
```

## **Send a request for data to the arduino**

```
uartwrite(0, 0, "a")
```

## **wait a bit for the data to be transmitted**

```
sleep(50)  
$dataAvail = UartAvailable(0, 0)  
IF ($dataAvail != 2)  
print("The arduino didn't send us anything. Exiting")  
halt()  
ELSE  
UartReadBinary(0, 0, 2, $encoders)  
print("Encoder A: " + $encoders[0])  
print("Encoder B: " + $encoders[1])  
ENDIF  
sleep(100)  
goto(loop)
```

```
...`
```

This tutorial was inspired from this thread: <https://www.ez-robot.com/Community/Forum/Thread?threadId=5603&page=3>