

## IR 4bit Receiver

3/4/19

```
int IRrx = 3;
int indicator = 13;           //The indicator will light when a valid
int Red = 4;                 //signal is received.
int Yellow = 5;
int Green = 6;
int Blue = 7;
int startCount = 0;
int stopCount = 0;
int RedSig;
int YellowSig;
int GreenSig;
int BlueSig;
int checkSig;               //The checkSig will always be low, thus preventing
                             //a false reading on a constant HIGH.

void setup() {
  pinMode(IRrx, INPUT);
  pinMode(indicator, OUTPUT);
  pinMode(Red, OUTPUT);
  pinMode(Yellow, OUTPUT);
  pinMode(Green, OUTPUT);
  pinMode(Blue, OUTPUT);
}

void loop() {
  startCount = 0;
  stopCount = 0;
  while(digitalRead(IRrx) == HIGH) { //The IRrx is holding the output HIGH
    digitalWrite(indicator, LOW); //until it receives a signal.
    delay(1); }
  while(digitalRead(IRrx) == LOW) { //This loop counts how long the
    startCount++; //start pulse is.
    delay(10); }
  if(startCount == 6) {
    digitalWrite(indicator, HIGH);
    delay(10);
    RedSig = digitalRead(IRrx);
    delay(10);
    YellowSig = digitalRead(IRrx);
    delay(10);
    GreenSig = digitalRead(IRrx);
    delay(10);
    BlueSig = digitalRead(IRrx);
    delay(10);
    checkSig = digitalRead(IRrx);
    delay(3);
  }
  while(digitalRead(IRrx) == HIGH) { //This is waiting for the beginning
    delay(1); } //of the stop pulse.
  while(digitalRead(IRrx) == LOW) { //This loop counts how long the
    stopCount++; //stop pulse is.
    delay(10); }
  if(stopCount == 7) {
    digitalWrite(Red, !RedSig); //The signals will be saved as the
    digitalWrite(Yellow, !YellowSig); //inverse of their state, so we use
    digitalWrite(Green, !GreenSig); //the exclamation mark to "invert"
    digitalWrite(Blue, !BlueSig); //the signal from LOW to HIGH.
    delay(500);
  }
}
```