

ラジコン 赤外線受信

```
#define IR_PIN          2 // 赤外線受信モジュール接続ピン
/*****
IRrecv - 赤外線リモコンの送信データを受信する関数
        DATA_POINT で指定した位置のデータ(8ビット)を読み取ります。
*
        戻り: 読み取った整数値を返します
*****/
int IRrecv()
{
    unsigned long t ;
    int i , j ;
    int cnt , ans ;
    char IRbit[64] ;

    ans = 0 ;
    t = 0 ;
    if (digitalRead(IR_PIN) == LOW) {
        // リーダ部のチェックを行う
        t = micros() ;                // 現在の時刻(us)を得る
        while (digitalRead(IR_PIN) == LOW) ; // HIGH(ON)になるまで待つ
        t = micros() - t ;           // LOW(OFF)の部分をはかる
    }
    // リーダ部有りなら処理する(3.4ms 以上の LOW にて判断する)
    if (t >= 3400) {
        i = 0 ;
        while (digitalRead(IR_PIN) == HIGH) ; // ここまでがリーダ部(ON 部分)読み飛ばす
        // データ部の読み込み
        while (1) {
            while (digitalRead(IR_PIN) == LOW) ; // OFF 部分は読み飛ばす
            t = micros() ;
            cnt = 0 ;
            while (digitalRead(IR_PIN) == HIGH) { // LOW(OFF)になるまで待つ
                delayMicroseconds(10) ;
                cnt++ ;
                if (cnt >= 1200) break ; // 12ms 以上 HIGH のままなら中断
            }
            t = micros() - t ;
            if (t >= 10000) break ; // ストップデータ
            if (t >= 1000) IRbit[i] = (char)0x31 ; // ON 部分が長い
            else          IRbit[i] = (char)0x30 ; // ON 部分が短い
            i++ ;
        }
    }
}
```

```
// データ有りなら指定位置のデータを取り出す
if (i != 0) {
    i = (DATA_POINT - 1) * 8;
    for (j = 0; j < 8; j++) {
        if (IRbit[i + j] == 0x31) bitSet(ans, j);
    }
}
return (ans);
}
```

赤外線確認

```
#define IRpin      2    // 赤外線受信モジュール接続ピン番号

// 初期化処理
void setup()
{
    Serial.begin(9600); // パソコン(ArduinoIDE)とシリアル通信の準備を行う
    pinMode(IRpin,INPUT); // 赤外線受信モジュールに接続ピンをデジタル入力に設定
}

// メインの処理
void loop()
{
    unsigned long t;
    int i, cnt;
    char IRbit[64];

    t = 0;
    if (digitalRead(IRpin) == LOW) {
        // リーダ部のチェックを行う
        t = micros(); // 現在の時刻(us)を得る
        while (digitalRead(IRpin) == LOW); // HIGH(ON)になるまで待つ
        t = micros() - t; // LOW(OFF)の部分をはかる
    }
    // リーダ部有りなら処理する(3.4ms 以上の LOW にて判断する)
    if (t >= 3400) {
        i = 0;
        while(digitalRead(IRpin) == HIGH); // ここまでがリーダ部(ON 部分)読み飛ばす
        // データ部の読み込み
        while (1) {
            while(digitalRead(IRpin) == LOW); // OFF 部分は読み飛ばす
            t = micros();
            cnt = 0;
            while(digitalRead(IRpin) == HIGH) { // LOW(OFF)になるまで待つ
                delayMicroseconds(10);
                cnt++;
                if (cnt >= 1200) break; // 12ms 以上 HIGH のままなら中断
            }
            t = micros() - t;
            if (t >= 10000) break; // ストップデータ
            if (t >= 1000) IRbit[i] = (char)0x31; // ON 部分が長い
            else IRbit[i] = (char)0x30; // ON 部分が短い
            i++;
        }
    }
}
```

```

// データ有りなら表示を行う
if (i != 0) {
    IRbit[i] = 0 ;
    DspData(i,IRbit) ;
}
}
}
// 受信データをシリアルモニタ画面に送り表示をさせる処理
void DspData(int num,char *data)
{
    int i , j , x , dt ;

    Serial.print(data) ; // ビットデータで表示
    Serial.write(" ( " ) ;
    x = num / 8 ;
    // ビット文字列データから数値に変換する
    for (j=0 ; j < x ; j++) {
        dt = 0 ;
        for (i=0 ; i < 8 ; i++) {
            if (*data++ == 0x31) bitSet(dt,i) ;
        }
        Serial.print(dt,HEX) ; // H E X (16 進数) で表示
        Serial.write(' ' ) ;
    }
    Serial.println(')') ;
}
}

```