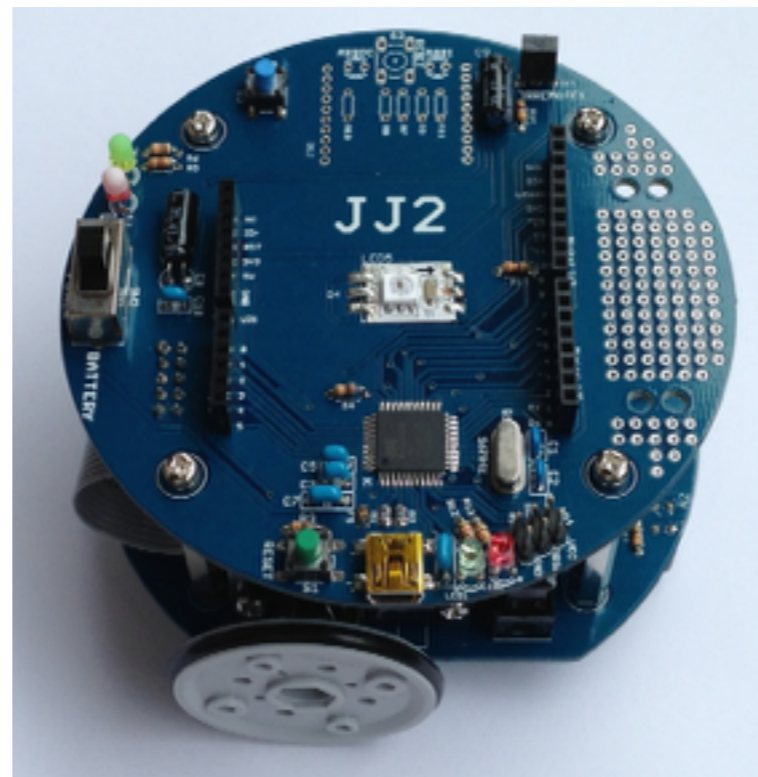


ロボットをつくらう

第6回 赤外線リモコンを使う



<http://cms.db.tokushima-u.ac.jp/DAV/person/S10723/>

川上 博

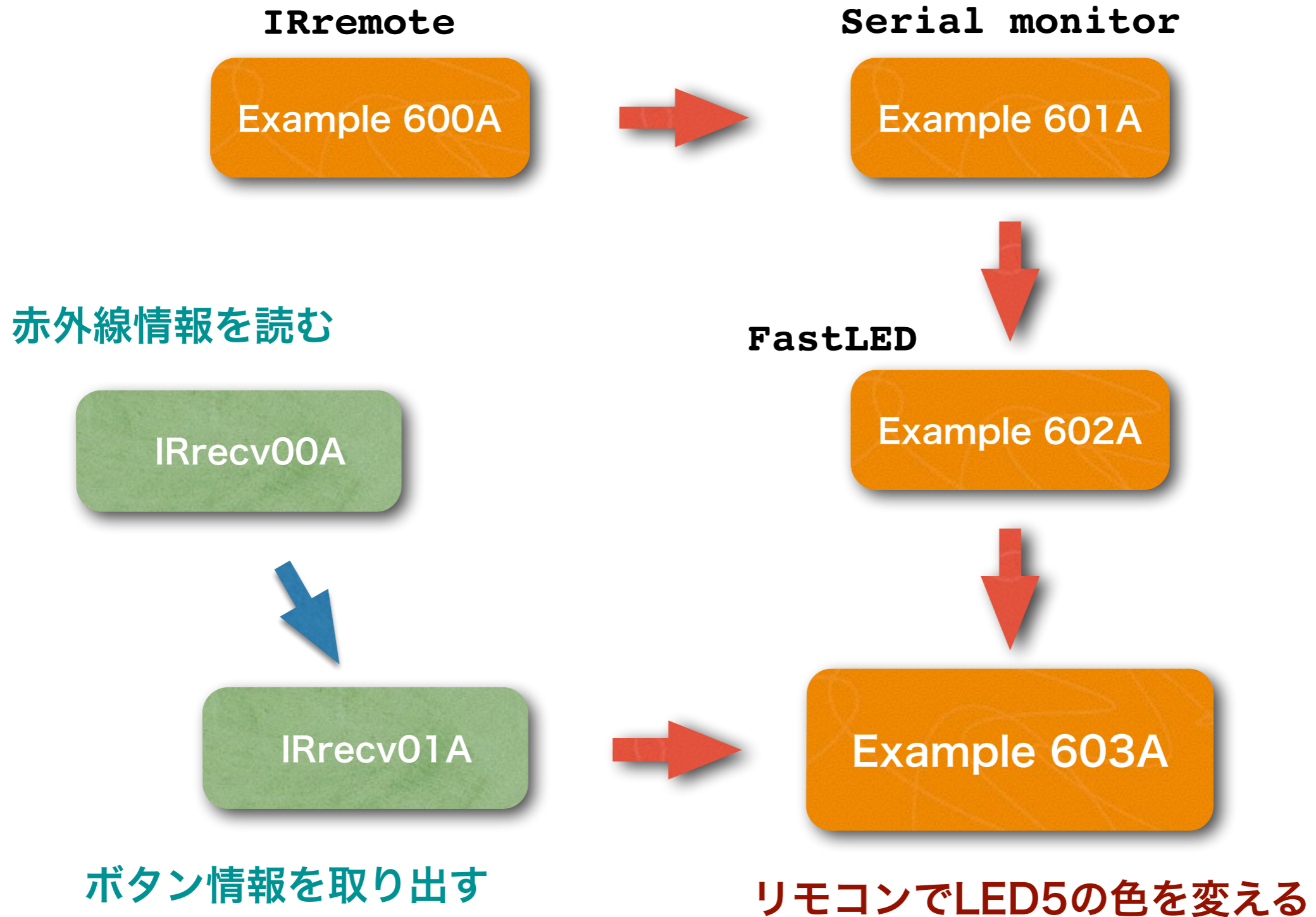
2014/08/30

今日のテーマ

Arduinoのlibrary - IR remote -を使う

IR リモコンでJJ2の上のLED5を点灯する

今日のスケッチ

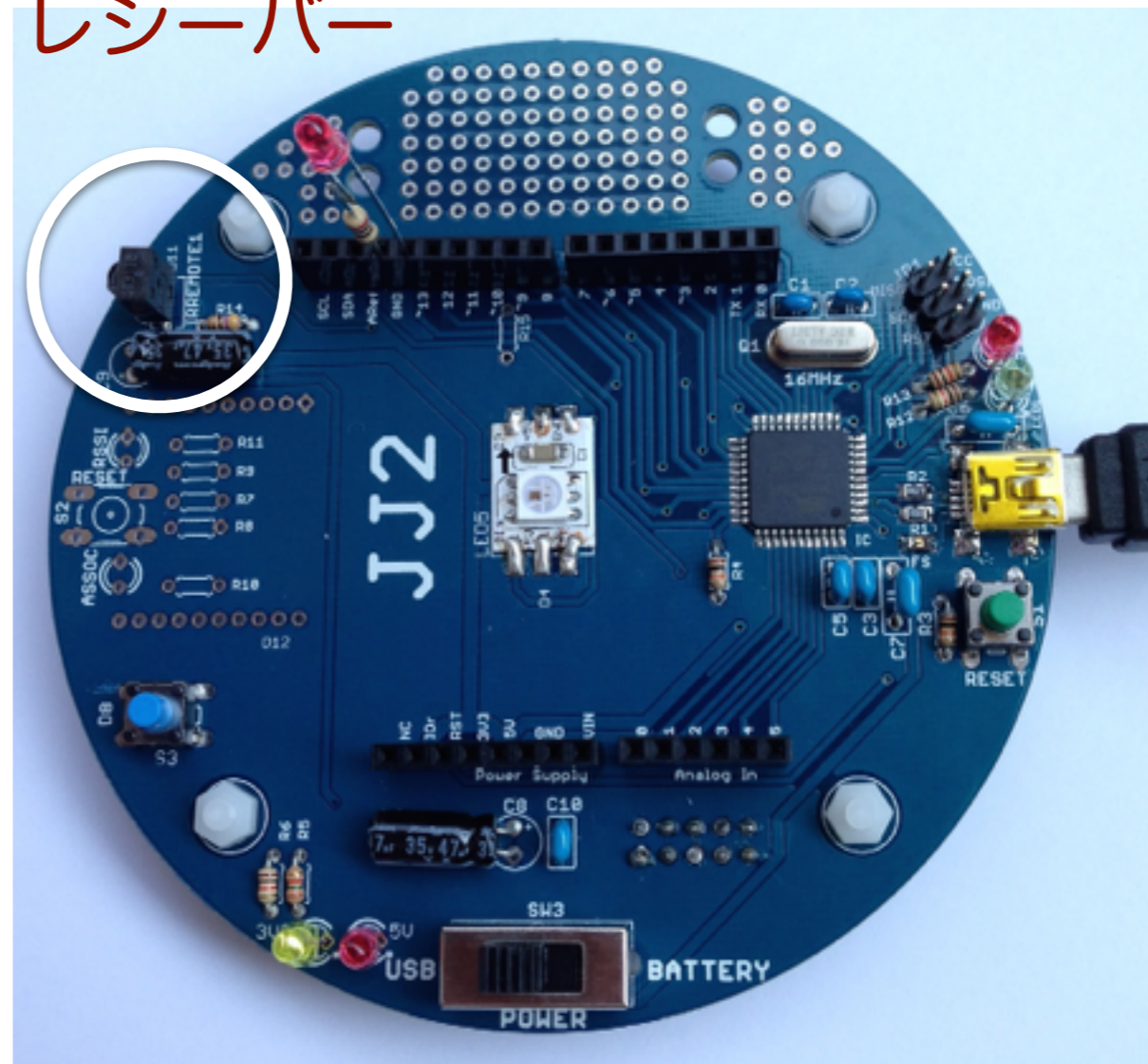


赤外線リモコン : IR remote

IR 送信器



IR レシーバー

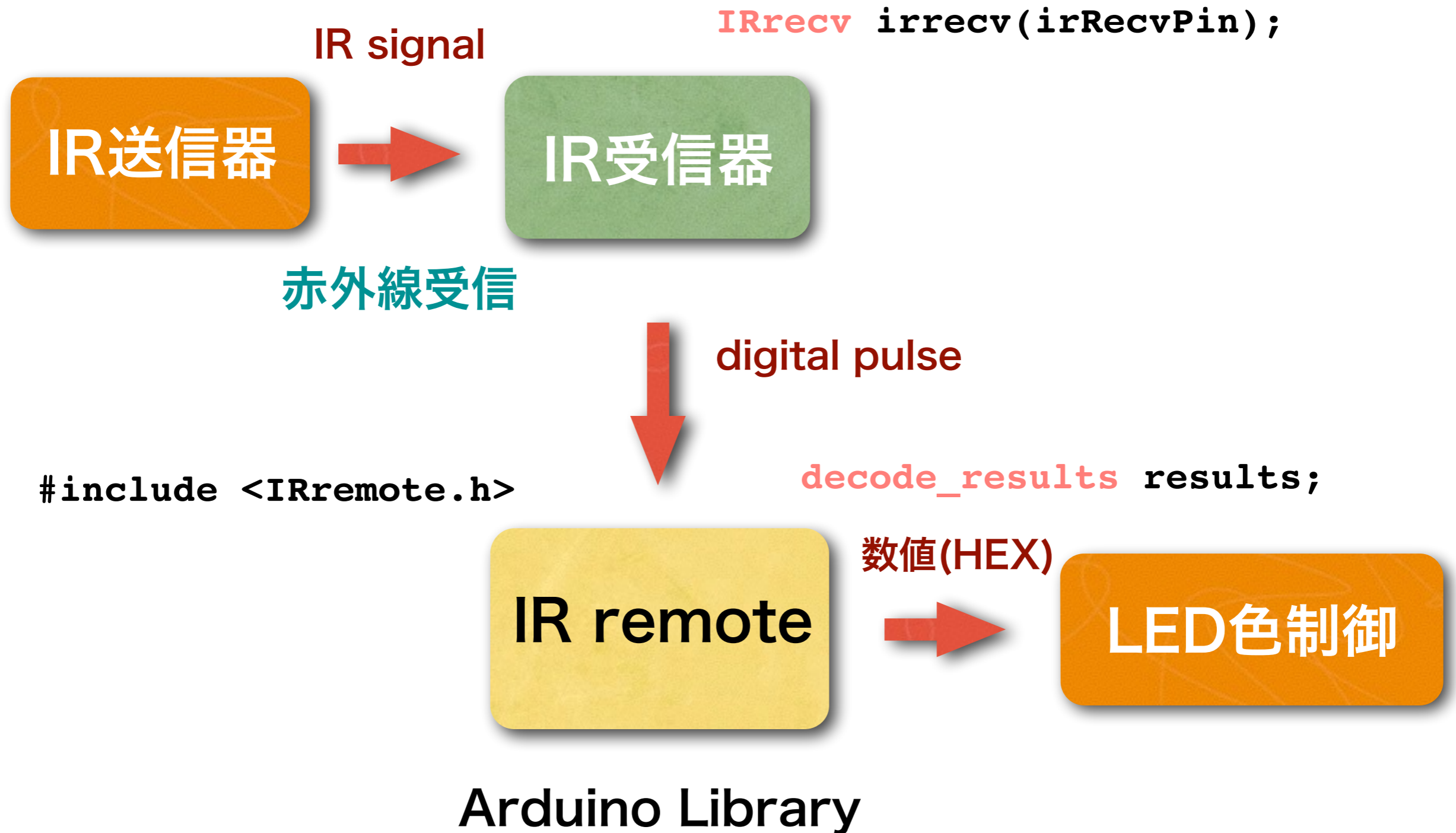


家電製品協会フォーマット

NECフォーマット

sonyフォーマット

赤外線通信 (受信の場合)



IRリモコンのボタンを押す毎に, LEDが点灯・消灯を繰り返す

```
/*
  IRremote:Example600A - detector sketch
*/
#include <IRremote.h>

const int irRecvPin = 11;
const int ledPin = 13;

IRrecv irrecv(irRecvPin); // IR receive object
decode_results results; // IR decode object

void setup(){
  pinMode(ledPin, OUTPUT);
  irrecv.enableIRIn();
}

int lightState = 0;

void loop(){
  if (irrecv.decode(&results)){
    delay(200);
    lightState = 1-lightState;
    digitalWrite(ledPin, lightState);
    irrecv.resume();
  }
}
```

`Serial.begin(9600);`

`Serial.println(results.value, HEX);`

```

IRrecv irrecv(irRecvPin); // IR receive object
decode_results results; // IR decode object

```

```

class IRrecv {
    IRrecv(int recvpin); // constructor
    void blink13(int blinkflag); // enable/disable blinking of pin 13
    int decode(decode_results *results); // decoder function
    void enableIRIn(); // initialization
    void resume(); // reset
};

class decode_results {
    int decode_type; // NEC, SONY, RC5, UNKNOWN
    unsigned int panasonicAddress; // Only used for decoding Panasonic data
    unsigned long value; // Decoded value
    int bits; // Number of bits in decoded value
    volatile unsigned int *rawbuf; // Raw intervals in .5 us ticks
    int rawlen; // Number of records in rawbuf.
};

```

IRリモコンからの情報を読む

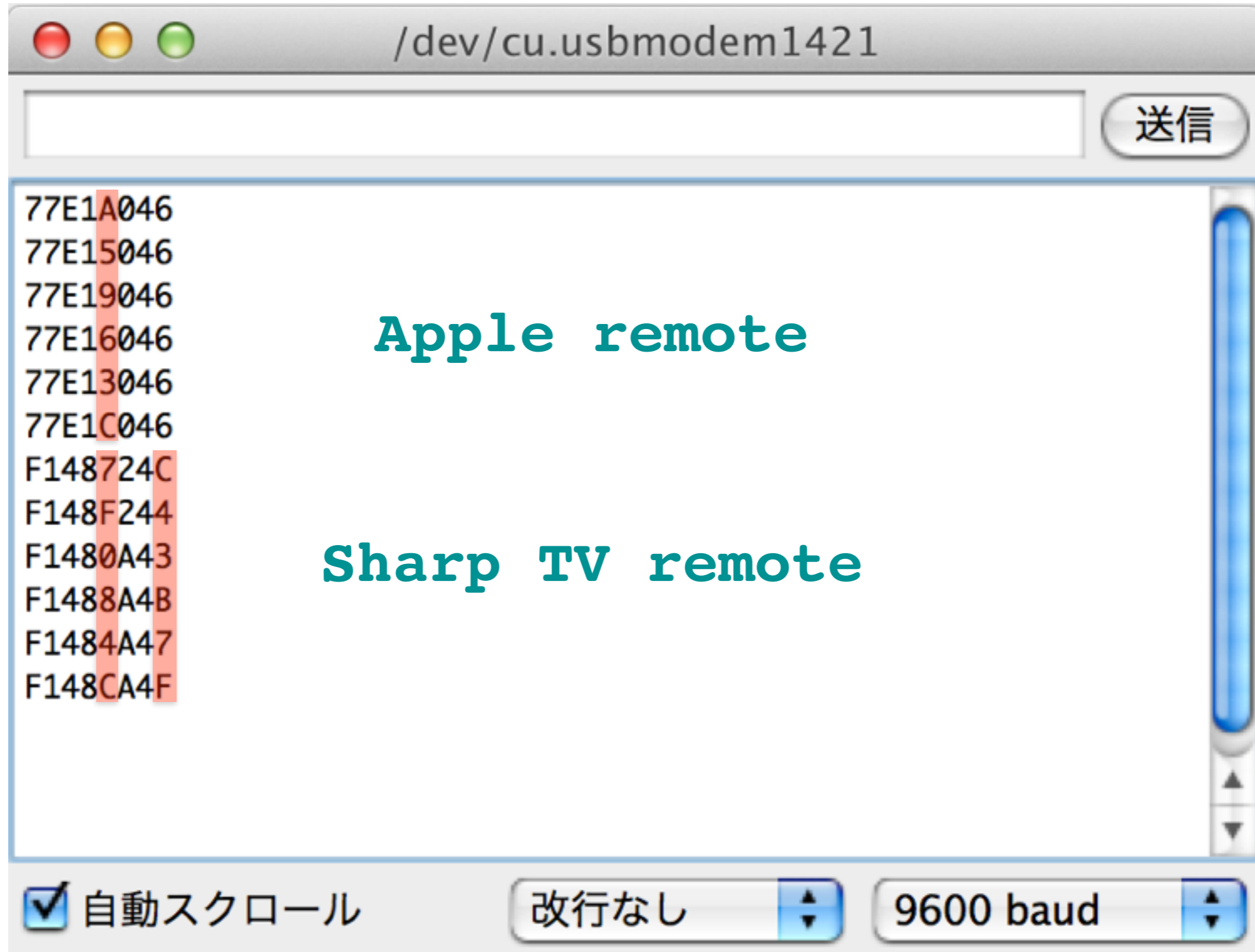
```
// IRrecvDemo00
#include <IRremote.h>

int RECV_PIN = 11;
IRrecv irrecv(RECV_PIN); // IR receiver object
decode_results results; // IR decoder object

void setup(){
  Serial.begin(9600);
  irrecv.enableIRIn(); // Start the receiver
}

void loop(){
  if (irrecv.decode(&results)) {
    delay(250);
    if(results.value!=0xffffffff){
      Serial.println(results.value, HEX);
      irrecv.resume(); // Receive the next value
    }
  }
}
```


`results.value(32bits)`



IRリモコンのボタン情報を読む

```
// IRrecvDemo01
#include <IRremote.h>

int RECV_PIN = 11;
IRrecv irrecv(RECV_PIN); // IR receiver object
decode_results results; // IR decoder object

unsigned getbits(unsigned long x, int p, int n){
    return (x>>(p+1-n))&~(~0<<n);
}

void setup(){
    Serial.begin(9600);
    irrecv.enableIRIn();
}

unsigned res2;

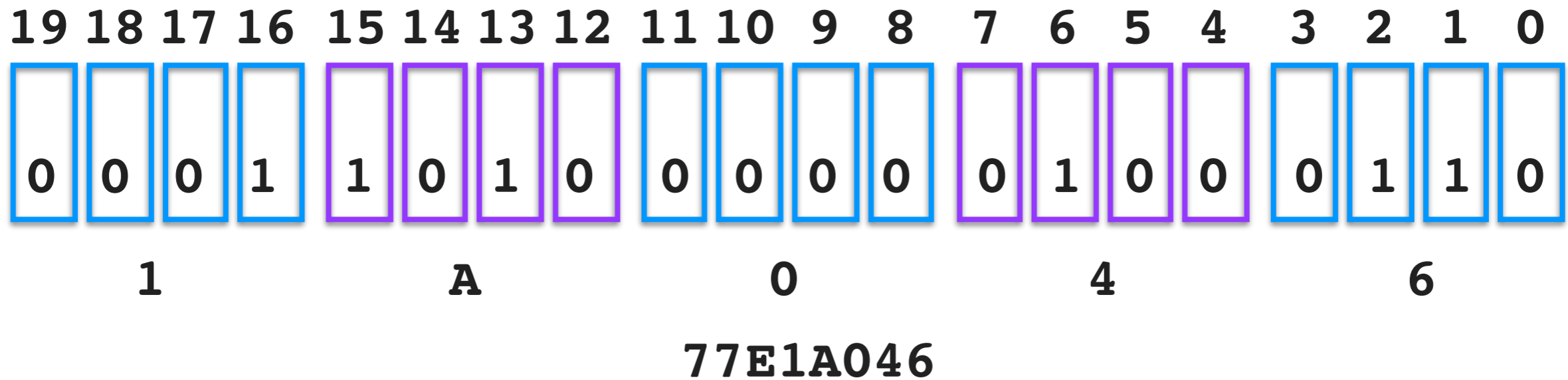
void loop(){
    if (irrecv.decode(&results)) {
        delay(250);
        res2=getbits(results.value, 15,4);
        Serial.println(res2, HEX);
        irrecv.resume();
    }
}
```



pの位置からnビットを取ってくる関数



```
res2 = getbits(results.value, 15, 4);
```



HEX	binary	HEX	binary
0	0000	8	1000
1	0001	9	1001
2	0010	A	1010
3	0011	B	1011
4	0100	C	1100
5	0101	D	1101
6	0110	E	1110
7	0111	F	1111



```
// IRremote:Example602A
#include <IRremote.h>
#include <FastLED.h>

const int irRecvPin = 11;
const int ledPin=4;
CRGB leds[1];
IRrecv irrecv(irRecvPin);
decode_results results;

void setup(){
  delay(2000);
  LEDS.setBrightness(127); // 0-255
  FastLED.addLeds<WS2811, ledPin, GRB>(leds, 1);
  Serial.begin(9600);
  irrecv.enableIRIn();
}

int ledState = 0;

void loop(){
  if (irrecv.decode(&results)){
    delay(200);
    Serial.println(results.value, HEX);
    ledState = 1-ledState;
    if(ledState){
      leds[0]= CRGB(0, 0, 255);
    }else{
      leds[0]= CRGB(0, 0, 0);
    }
    LEDS.show();
    irrecv.resume();
  }
}
```

LED5を点灯するために加えた命令

```
/*
 * Example603A: JJ2-LED5-Demo
 * H. kawakami July 8, 2014
 */

#include <IRremote.h>
#include <FastLED.h>

const int ledPin=4;
const int irRecvPin = 11;
CRGB leds[1];
IRrecv irrecv(irRecvPin);
decode_results results;

unsigned getbits(unsigned long x, int p, int n){
    return (x>>(p+1-n))&~(~0<<n);
}

void setup(){
    delay(200);
    LEDS.setBrightness(127); // 0-255
    FastLED.addLeds<WS2811, ledPin, GRB>(leds, 1);
    Serial.begin(9600);
    irrecv.enableIRIn();
}
```

```
unsigned res2;

void loop() {
  if (irrecv.decode(&results)) {
    delay(200);
    res2=getbits(results.value, 15,4);
    if(res2==0x5){
      leds[0]=CRGB(100, 0, 0);
    }
    if(res2==0x9){
      leds[0]=CRGB(0, 100, 0);
    }
    if(res2==0x6){
      leds[0]=CRGB(0, 0, 100);
    }
    if(res2==0x3){
      leds[0]=CRGB(100, 100, 0);
    }
    if(res2==0xa){
      leds[0]=CRGB(100, 100, 100);
    }
    if(res2==0xc){
      leds[0]=CRGB(0, 0, 0);
    }
    LEDS.show();
    Serial.println(results.value, HEX);
    Serial.println(res2, HEX);
    irrecv.resume();
  }
}
```

押されたボタンに従って色を変える