

【B-1(4)】IchigoJam で信号機を作ろう (Ver.4)

●今回の目標

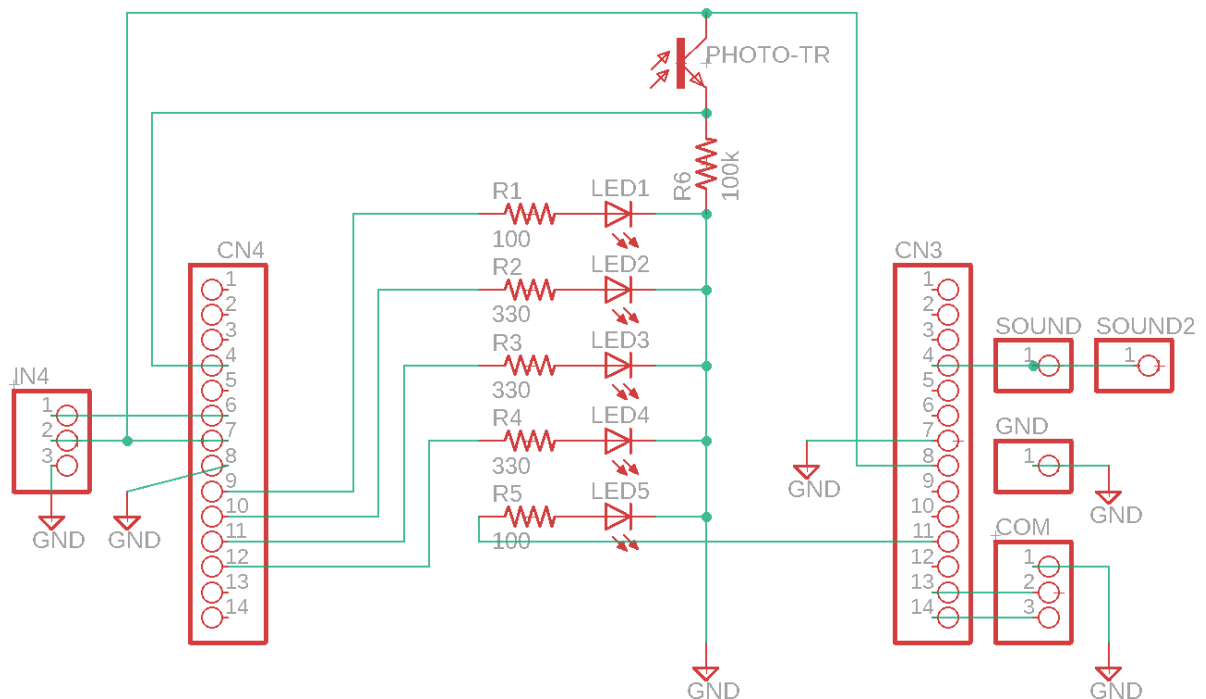
信号機基板を IchigoJam にさして、
BASIC プログラムで光らせます。



●信号機基板をさしこむ

IchigoJam のソケットに、信号機基板のピンをさします。
上下をまちがえないようにしてください。
14本のピン×2列が、IchigoJam のソケットにさります。

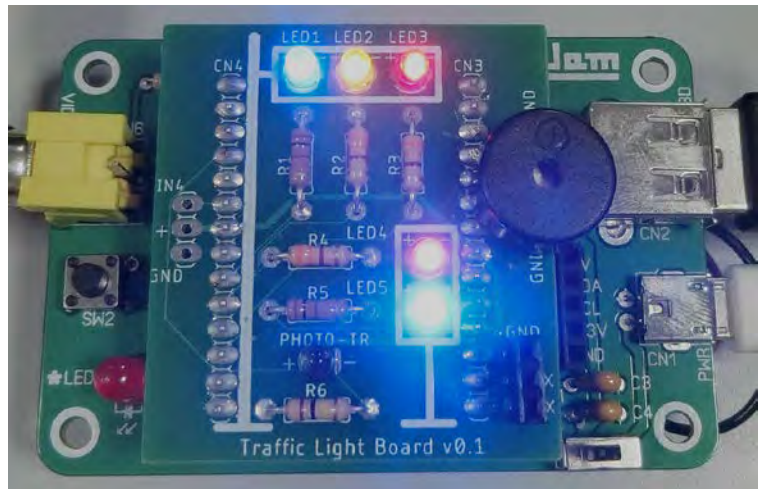
回路図は以下のとおりです。



信号機基板が取り付けられたら、ちゃんとLED が光るか、ダイレクトモードで確認しましょう。

```
OUT 1,1
OUT 2,1
OUT 3,1
OUT 4,1
OUT 5,1
```

車用と歩行者用の信号機の青・黄・赤のLED が光ります。もし光らなかったら、基板のさし方をまちがえています。よく見直しましょう。



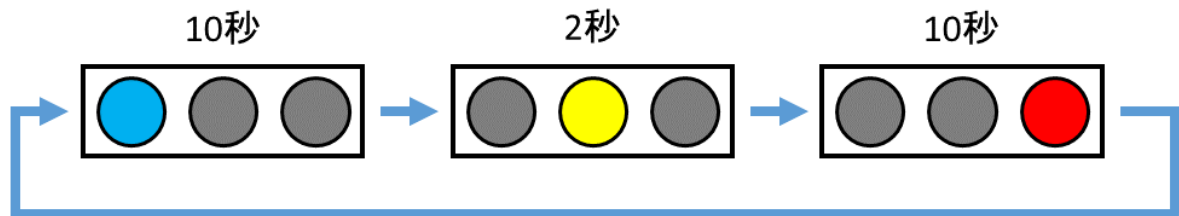
```
OUT 1,0
OUT 2,0
OUT 3,0
OUT 4,0
OUT 5,0
```

5 個の LED が消えます。

いろいろなパターンで信号機を光らせてみましょう。

● 車用信号機を光らせる

まず、車用の信号機を光らせるプログラムを作ります。
信号機の動きを考えてみます。



今回は、青・黄・赤の点灯時間を上のようにならしてみよう。

10	^*TRAFFIC LIGHT	プログラムのタイトル
20	OUT 0	LEDを全部消す
30	^*LOOP	ループの始め
40	OUT 1,1	青LEDを光らせる
50	WAIT 600	10秒待つ
60	OUT 1,0	青LEDを消す
70	OUT 2,1	黄LEDを光らせる
80	WAIT 120	2秒待つ
90	OUT 2,0	黄LEDを消す
100	OUT 3,1	赤LEDを光らせる
110	WAIT 600	10秒待つ
300	OUT 3,0	赤LEDを消す
310	GOTO 30	30行へもどる

「**RUN**」でプログラムを実行してみましょう。
信号機が、青→黄→赤→青→…と動きます。

WAIT (ウェイト) 命令の数値を変えると、それぞれの LED の点灯時間が変わります。
試してみましょう。

「**SAVE 0**」でプログラムを保存しておきましょう。

●歩行者用信号機を光らせる

車用信号機と歩行者用信号機を動かすプログラムを作ります。
2つの信号機の動きをどうすればいいか、考えてみましょう。

車用信号機							
歩行者用信号機							
時間	10秒	2秒	2秒	5秒	0.5秒	0.5秒	2秒

- 車用信号機を「青→黄→赤」と動かします。この間、歩行者用信号機はずっと「赤」です。
- 車用信号機を「赤」にして、2秒待ってから、歩行者用信号機を「青」にします。
(すぐに「青」にしてしまうと、まだ車が来るかもしれないので危ないです)
- 歩行者用信号機の「青」を5秒間続けます。
- 歩行者に注意をうながすため、歩行者用信号機の「青」を3回点滅させます。
- 歩行者用信号機を「赤」にして、2秒待ってから、最初にもどって車用信号機を「青」にします。
(すぐに「青」にしてしまうと、まだ歩行者が歩いているかもしれないので危ないです)

それでは、前ページの表と説明を見ながら、プログラムを改造してみましょう。

```

10  ^*TRAFFIC LIGHT
20  OUT 0
30  ^*LOOP
40  OUT 1,1
45  OUT 4,1
50  WAIT 600
60  OUT 1,0
70  OUT 2,1
80  WAIT 120
90  OUT 2,0
100 OUT 3,1
110 WAIT 120
120 OUT 4,0
130 OUT 5,1
140 WAIT 300
150 FOR I=1 TO 3
160 OUT 5,1
170 WAIT 30
180 OUT 5,0
190 WAIT 30
200 NEXT
210 OUT 4,1
220 WAIT 120
300 OUT 3,0
310 GOTO 30
    
```

歩行者用信号の赤を光らせる

車用信号の赤の時間を2秒に変更

歩行者用信号の赤を消して青を光らせる

5秒待つ

3回くりかえし

歩行者用信号の青を光らせる

0.5秒待つ

歩行者用信号の青を消す

0.5秒待つ

もどってくりかえし

歩行者用信号の赤を光らせる

2秒待つ

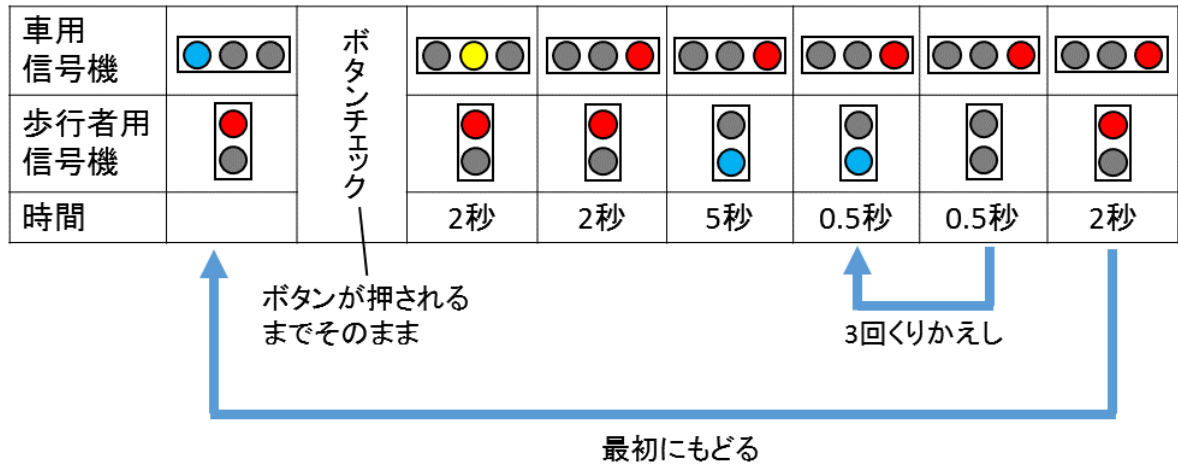
プログラムを実行してみましょう。
2つの信号機が連動して動きます。

WAIT (ウェイト) 命令の数値を変えると、それぞれの LED の点灯時間が変わります。
試してみましょう。

「SAVE 0」でプログラムを保存しておきましょう。

●押しボタン信号にする

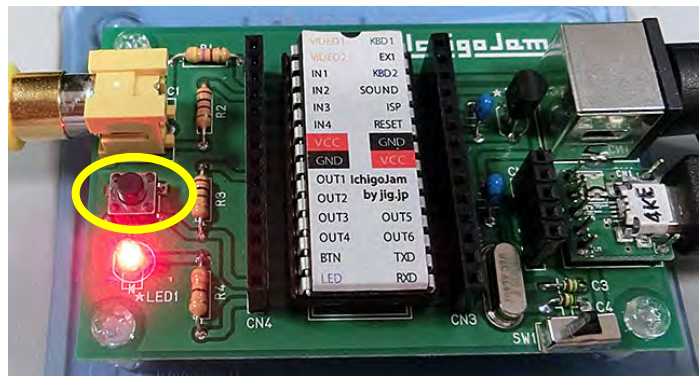
プログラムを改造して、「ボタンを押したら歩行者用信号が青になる」信号にしてみましょう。
信号機の動きを表で考えます。



車用信号機が青の時に、ボタンが押されているかをチェックして、押されるまではそのまま変化させません。

ボタンが押されたら、黄色以降の処理に移ります。それ以降の動きは同じです。

ボタンは、IchigoJam の左側に付いているボタンを使います。



プログラムを改造します。

車用信号機を青にした後の、時間待ち(WAIT)の行を書きかえます。

```
10  ^*TRAFFIC LIGHT
20  OUT 0
30  ^*LOOP
40  OUT 1,1
45  OUT 4,1
50  IF BTN( )=0 THEN GOTO 50
55  WAIT 180
60  OUT 1,0
70  OUT 2,1
80  WAIT 120
90  OUT 2,0
100 OUT 3,1
110 WAIT 120
120 OUT 4,0
130 OUT 5,1
140 WAIT 300
150 FOR I=1 TO 3
160 OUT 5,1
170 WAIT 30
180 OUT 5,0
190 WAIT 30
200 NEXT
210 OUT 4,1
220 WAIT 120
300 OUT 3,0
310 GOTO 30
```

ボタンが押されていないかったら、この行をくりかえし

3秒待つ

プログラムを実行してみましょう。

車用信号がずっと青のままになります。

ボタンを押すと、3秒後に黄色→赤に変わります。

※車用信号が青になった直後にボタンが押された場合、すぐに黄色になっても車は急に止まれないので、3秒待つようにしています。

●「おまちください」とお知らせする

ボタンを押してから3秒待たされると、「ちゃんとボタンが押せたのか？」と不安になります。現実の押しボタン信号では、「おまちください」のランプが点いて歩行者にお知らせします。IchigoJam についているLEDを、お知らせランプとして点灯させましょう。

```
10  ^*TRAFFIC LIGHT
20  OUT 0
30  ^*LOOP
40  OUT 1,1
45  OUT 4,1
50  IF BTN(<)<=0 THEN GOTO 50
52  LED 1 LEDを光らせる
55  WAIT 180
60  OUT 1,0
70  OUT 2,1
80  WAIT 120
90  OUT 2,0
100 OUT 3,1
110 WAIT 120
120 OUT 4,0
130 OUT 5,1
135 LED 0 LEDを消す
140 WAIT 300
150 FOR I=1 TO 3
160 OUT 5,1
170 WAIT 30
180 OUT 5,0
190 WAIT 30
200 NEXT
210 OUT 4,1
220 WAIT 120
300 OUT 3,0
310 GOTO 30
```

プログラムを実行してみましょう。

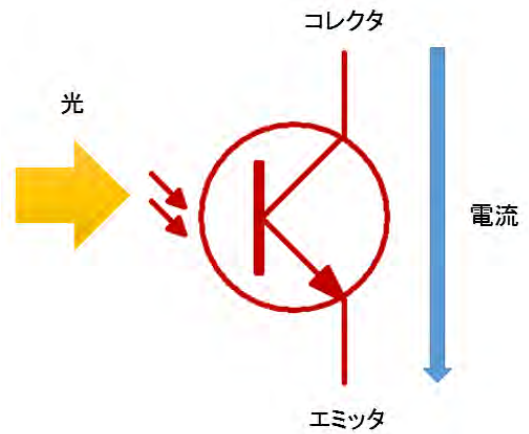
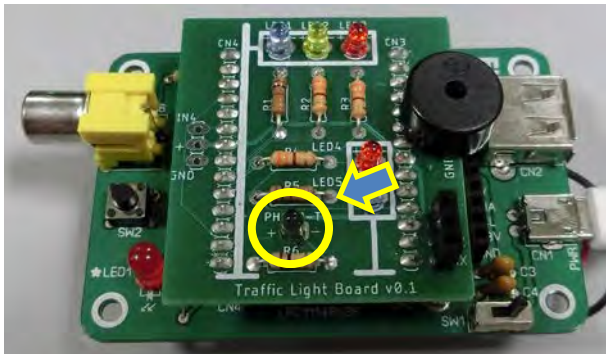
ボタンを押すと、IchigoJam の LED が光ってお知らせします。

歩行者用信号が青になると、LED が消えます。

「SAVE 0」でプログラムを保存しておきましょう。

●夜は黄色の点滅にする

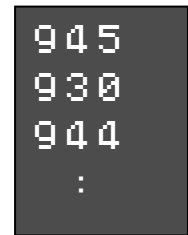
夜になると、黄色の点滅になる信号機があります。
 光センサーで明るさを読み取って、信号機を動かしてみましょ。う。
 今回は光センサーとして、**フォトランジスタ**を使います。
 フォトランジスタは、光が当たると電流が流れる部品です。



まず、フォトランジスタがちゃんと動作するか確認しましょう。
 以下のプログラムを入力します。

```
1 PRINT ANA(2):GOTO 1
```

プログラムを実行してみましょ。画面に数字が連続で表示されます。
 まわりの明るさによりますが、だいたい 900 を越えたくらいの値になるはずで。



フォトランジスタに指をおいて、かげにしてみましょ。
 数字が 100 以下くらいに小さくなります。
 (まわりの明るさによって数字は変わります)



確認ができたら、

```
1 (Enter)
```

と入力して、1 行目を消しましょ。

「ANA()」は、アナログ入力を読み取る関数で。

```
ANA( 2 )  
      ポート番号
```

ポート番号は、2(IN2 端子)、5~8(OUT1~4 端子)、0 か 9(BTN 端子)が使える。
 返ってくる値は、0(0V)~1023(3.3V)。

それでは、夜になったら黄色の点滅になる信号機を作りましょう。

```

10  ' *TRAFFIC LIGHT
20  OUT 0
30  ' *LOOP
40  OUT 1,1
45  OUT 4,1
47  IF ANA(2)<300 THEN GOSUB 400:GOTO 60
50  IF BTN( )=0 THEN GOTO 47
52  LED 1
55  WAIT 180
60  OUT 1,0
70  OUT 2,1
80  WAIT 120
90  OUT 2,0
100 OUT 3,1
110 WAIT 120
120 OUT 4,0
130 OUT 5,1
135 LED 0
140 WAIT 300
150 FOR I=1 TO 3
160 OUT 5,1
170 WAIT 30
180 OUT 5,0
190 WAIT 30
200 NEXT
210 OUT 4,1
220 WAIT 120
300 OUT 3,0
310 GOTO 30
400 ' *NIGHT
410 OUT 1,0
420 OUT 2,1
430 WAIT 30
440 OUT 2,0
450 WAIT 30
460 IF ANA(2)<300 THEN GOTO 420
470 RETURN

```

もし暗かったら、400 行のサブルーチンを呼ぶ
その後は 60 行へジャンプ

もどる行を変更

ここから夜間のサブルーチン

車用の青を消す

黄を光らせる

0.5 秒待つ

黄を消す

0.5 秒待つ

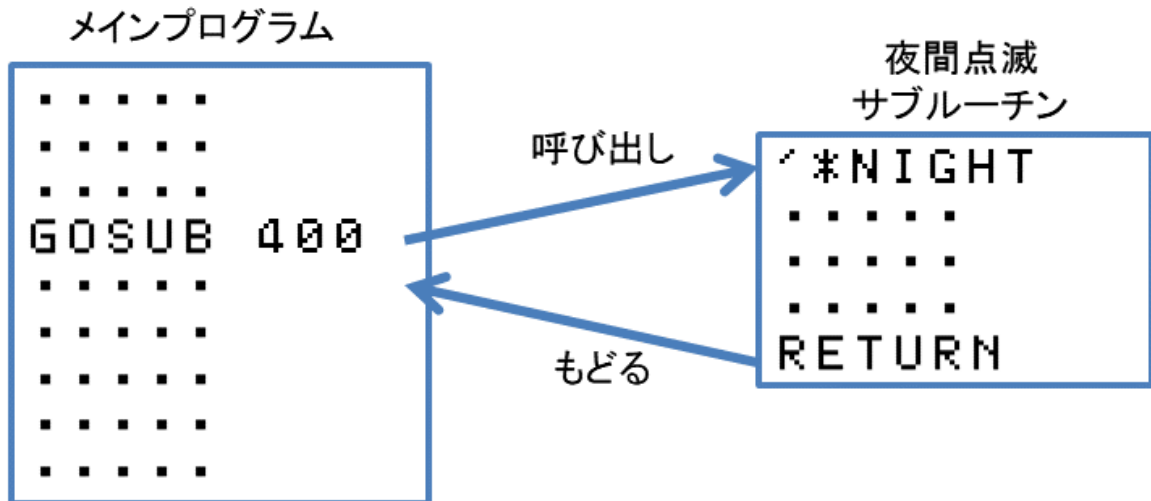
まだ暗かったら 420 行へジャンプ

メインプログラムへもどる

プログラムを実行してみましょう。

車用の青が光っている時にフォトランジスタを指で暗くすると、黄色の点滅になります。明るくすると、元に戻って黄色→赤になります。

47 行では、IF 命令でフォトランジスタからの入力・ANA(2)の値を判断して、もし 300 以下だったら夜間点滅のサブルーチンを呼んでいます。



メインプログラムからは「GOSUB」命令でサブルーチンへジャンプし、サブルーチンからは「RETURN」(リターン)命令でもどります。もどった後は、GOSUB 命令の続きへプログラムの処理が移ります。

サブルーチンに分けると、プログラムがすっきりしてわかりやすくなります。

また、何度も同じサブルーチンを呼び出して使うことができます。

「SAVE 0」でプログラムを保存しておきましょう。

★できる人は

歩行者用信号が青になる時に、メロディを流してみましょ。以下の部分を改造します。

```

120 OUT 4,0
130 OUT 5,1
135 LED 0
137 PLAY "04L8 E2E4D4E4ED>B2 <F4FFA4FE
FEDDE2 FFFF4AFEFEDDE2"
140 WAIT 720
150 FOR I=1 TO 3
160 OUT 5,1
165 PLAY "F4"
170 WAIT 30
180 OUT 5,0
185 PLAY "B4"
190 WAIT 30
200 NEXT
210 OUT 4,1
220 WAIT 120
300 OUT 3,0
310 GOTO 30
    
```

プログラムを実行してみましょ。歩行者用信号が青になると、メロディが流れます。

PLAY 命令は、音階を鳴らします。

音階はアルファベットで指定します。(MML、Music Macro Language)

ド	レ	ミ	ファ	ソ	ラ	シ
C	D	E	F	G	A	B

音階の前に「>」を付けると1オクターブ下げ、「<」を付けると1オクターブ上がります。音階の後の数字は、音符の長さを表します。(例:E4=ミの四分音符)最初に「L8」を指定しているので、数字の指定が無いと八分音符になります。

音階や曲をいろいろ変えてみましょ。