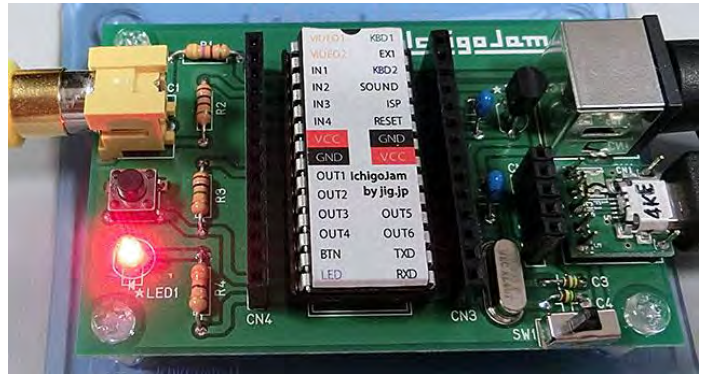


【B-0(1)】IchigoJam 入門・LED を光らせよう（１）

●今回の目標

IchigoJam に付いている LED を、いろいろに光らせます。
LED を光らせるための BASIC プログラムを作っていきます。



まずは、以下のプログラムを打ちます。

```
LED 1 <Enter キーを押す>
```

LED が光ります。

```
LED 0 <Enter キーを押す>
```

LED が消えます。

LED(エルイーディー)命令は、内蔵の LED を光らせる命令です。

LED 1
スイッチ

スイッチ	1=ON、0=OFF
------	------------

このままでは一度に 1 つの命令しか使えません。
もう少し長いプログラムを作ってみましょう。

```
10 LED 1 <Enter>
20 LED 0 <Enter>
```

LED を ON

LED を OFF

先頭に行番号をつけてプログラムを入力すると、すぐには実行されず、IchigoJam のメモリに記憶されます。

入力できたら、プログラムを実行してみましょう。

```
RUN <Enter>
```

RUN(ラン)命令はプログラムを実行する命令です。行の順番にプログラムが実行されます。
実行すると、どうになりましたか？

…LED が光りません。

これは、10 行の「LED 1」で LED が光るのですが、すぐに 20 行の「LED 0」で LED が消えてしまうので、速すぎて目に見えないのです。

●時間待ちをする

LED を光らせた後、しばらく時間を置いてから消すプログラムにしましょう。
間に 1 行追加します。

10	LED	1	※この行は打たなくてよい
15	WAIT	30	0.5 秒待つ
20	LED	0	※この行は打たなくてよい

画面上のどの場所でも「**15 WAIT 30** <Enter>」と入力すれば、15 行に追加されます。

プログラムを実行してみましょう。LED が 0.5 秒光って消えます。

WAIT (ウェイト) 命令は、時間を待つ命令です。

WAIT 30
待ち時間

待ち時間	1/60 秒単位で指定。60 で 1 秒。
------	-----------------------

待ち時間をいろいろ変えて、試してみましょう。

プログラムを 4 行追加して、LED を 2 回光らせてみましょう。

10	LED	1	※この行は打たなくてよい
15	WAIT	30	
20	LED	0	
30	WAIT	30	0.5 秒待つ
40	LED	1	LED を ON
50	WAIT	30	0.5 秒待つ
60	LED	0	LED を OFF

プログラムを実行してみましょう。LED が 2 回光ります。
WAIT の待ち時間をいろいろ変えて、試してみましょう。

入力されたプログラムの内容を確認するには、**LIST** (リスト) 命令を使います。

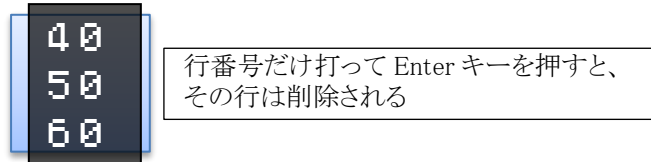
LIST

プログラムがうまく動かない時は、リストを表示して確認しましょう。

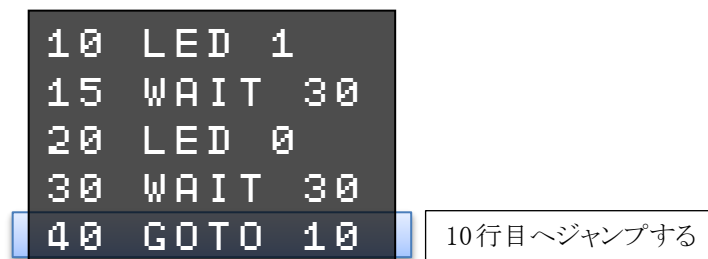
●くり返し

LED を 2 回光らせましたが、何度もくり返し光らせたい場合はどうすればいいでしょうか。
 こんな時は、プログラムの別の行へジャンプさせる命令を使います。

まず、LED を 2 回目に光らせるプログラム部分、40 行、50 行、60 行を消します。



以下のように、新しい 40 行を追加します。



プログラムを実行してみましょう。LED がずっとくり返し光ります。

そのままだと永久にくり返してしまうので、キーボードの ESC (エスケープ) キーを押して、プログラムを止めましょう。

新しく追加した GOTO (ゴートゥー) 命令について説明します。

GOTO 10
 ジャンプ先の
 行番号

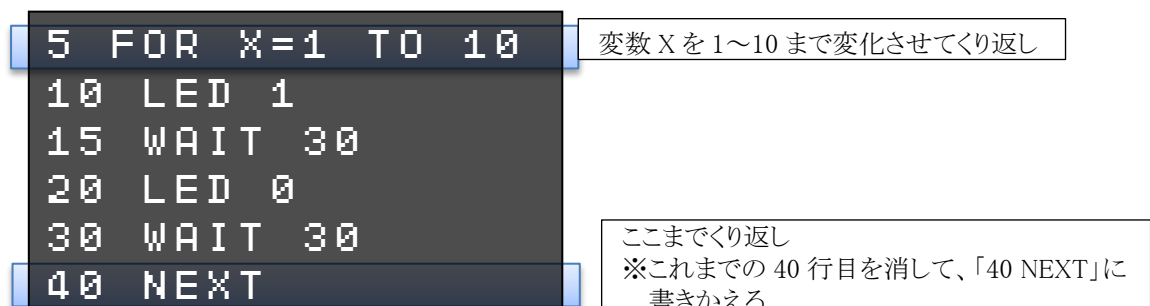
「**GOTO 10**」と書くと、そこから 10 行目へジャンプします。

プログラムの先頭にもどるので、LED を光らせる動作をずっとくり返すことになります。

では、LED を 10 回光らせて止めるにはどうすればいいでしょうか。

こんな時は、くり返しの命令を使います。

以下のプログラムを入力します。



プログラムを実行してみましょう。LED が 10 回光ります。

新しく追加した **FOR**(フォー)、**NEXT**(ネクスト) 命令について説明します。

FOR X=1 TO 10 STEP 1
 変数の 最後の値 変化させる値
 最初の値

変数の最初の値	指定した変数へ最初にセットする値。
最後の値	この値まで変数を変化させる。
変化させる値	この値ずつ変数を変化させる。 STEP 以下を省略すると1 ずつ増やす。

NEXT ここまで繰り返し

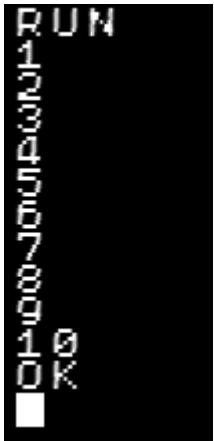
「変数」(へんすう)は、数字を入れる箱(いれもの)だと思ってください。
小学生の人は算数で習う「□」(四角)、中学生以上の人は「*x*」などの文字と思えばいいです。
「**FOR X=1 TO 10**」とすると、変数Xを1から始めて 10 になるまで1ずつ増やして、「**NEXT**」までのプログラムを繰り返し実行します。

くり返しの動きを確認するために、変数Xの値を画面に表示するようにしてみます。

```
5 FOR X=1 TO 10
7 PRINT X
10 LED 1
15 WAIT 30
20 LED 0
30 WAIT 30
40 NEXT
```

変数 X の値を画面に表示する

プログラムを実行してみましょう。
LED が光ると同時に、画面にXの値が順に表示されます。



FOR 命令の最初の値、最後の値、変化させる値をいろいろ変えて、どんな動きになるか試してみましょう。

●条件判断

IchigoJam には押しボタンスイッチが
付いています。
ボタンを押したら LED が光るようにし
てみましょう。



プログラムの 10 行目を改造します。

```
5 FOR X=1 TO 10
7 PRINT X
10 IF BTN()=1 THEN LED 1
15 WAIT 30
20 LED 0
30 WAIT 30
40 NEXT
```

もしボタンが押されていたら
LED を光らせる

プログラムを実行してみましょう。ボタンを押している時だけ LED が光ります。

「もしボタンが押されていたら」という条件判断には、**IF** (イフ)、**THEN** (ゼン)、**ELSE** (エルス) 命令を使います。

IF	BTN()=1	THEN	~	ELSE	~
	条件式		条件が成り立 つ時に実行		条件が成り立た ない時に実行

条件式	判断する条件。
条件が成り立つ 時に実行	条件式が成り立つ時に、この部分を実行する。
条件が成り立た ない時に実行	条件式が成り立たない時に、この部分を実行する。 ELSE 以下は省略可能。

今回は条件式で「ボタンが押されているかどうか」を判断するために、**BTN**(ボタン) 関数を使っています。

BTN()

関数の値	ボタンが押されている=1 ボタンが押されていない=0
------	-------------------------------

LED を光らせる条件をいろいろ変えてみましょう。

◇逆に、ボタンが押されていない時だけ LED を光らせる

```
10 IF BTN()=0 THEN LED 1
```

もしボタンが押されていないかったら
LED を光らせる

先のプログラムと逆になるので、ボタンを押すと LED が光らなくなります。

◇X が 1～5 の時だけ LED を光らせる

```
10 IF X<6 THEN LED 1
```

もし X が 6 より小さければ
LED を光らせる

最初の 5 回は LED が光りますが、後半の 5 回は光らなくなります。

不等号「<」「>」を使うと、「～より小さい」「～より大きい」という条件判断ができます。

◇X が奇数の時だけ LED を光らせる

```
10 IF X%2=1 THEN LED 1
```

もし X を 2 で割った余りが 1 ならば
LED を光らせる

X が 1,3,5,7,9 の奇数の時だけ、LED が光ります。

わり算の余りを出すには「%」を使います。

条件式をいろいろ変えて、試してみましょう。

●プログラムを保存する

このままだと、IchigoJam の電源を切ると、入力したプログラムは消えてしまいます。
消えないように、プログラムを保存しましょう。

★プログラムの保存

プログラムを保存するには、SAVE (セーブ) 命令を使います。

```
SAVE 0 <Enter>
Saved 62byte
OK
```

SAVE 命令の文法は以下のとおりです。

```
SAVE    0
      保存領域番号
```

保存番号	保存領域の番号(0～3)。省略すると、前回使った番号になる。
------	--------------------------------

保存領域は 0～3 の 4 つあり、最大 4 個までプログラムを保存できます。

ファンクションキー 3 番 (F3) に「SAVE」が設定されているので、ワンタッチで入力できます。

保存されているプログラムを確認するには、FILES (ファイルズ) 命令を使います。

```
FILES <Enter>
0 FOR X=1 TO 10
OK
```

保存領域の番号と、プログラム 1 行目の内容が表示されます。

★保存したプログラムを呼び出す

保存したプログラムを呼び出すには、LOAD (ロード) 命令を使います。

```
LOAD 0 <Enter>
Loaded 62byte
OK
```

```
LOAD    0
      保存領域番号
```

保存番号	保存領域の番号(0～3)。省略すると、前回使った番号になる。
------	--------------------------------

ファンクションキー 2 番 (F2) に「LOAD」が設定されているので、ワンタッチで入力できます。