Alpha

α-Xplorer

ロボットプログラミングキット

アルファ・エクスプローラ

C-Style 操作編

Rev.190615

DAISEN

株式会社ダイセン電子工業

目　次

C-Style 操作ガイド（本書）

1. プログラムボタンの説明
   1-1. プログラムボタンリスト ---------- ２
   1-2. モータ制御ボタン ---------- ３
   1-3. ウェイトタイマボタン ---------- ４
   1-4. ＬＥＤ制御ボタン ---------- ５
   1-5. タイマスタートボタン ---------- ６
   1-6. 変数ボタン ---------- ７
   1-7. 条件分岐 ---------- ８
   1-8. 条件付き繰返し ---------- ９
   1-9. 回数指定の繰返し ---------- １０
   1-10. タイマチェック ---------- １１
   1-11. 変数チェック ---------- １２

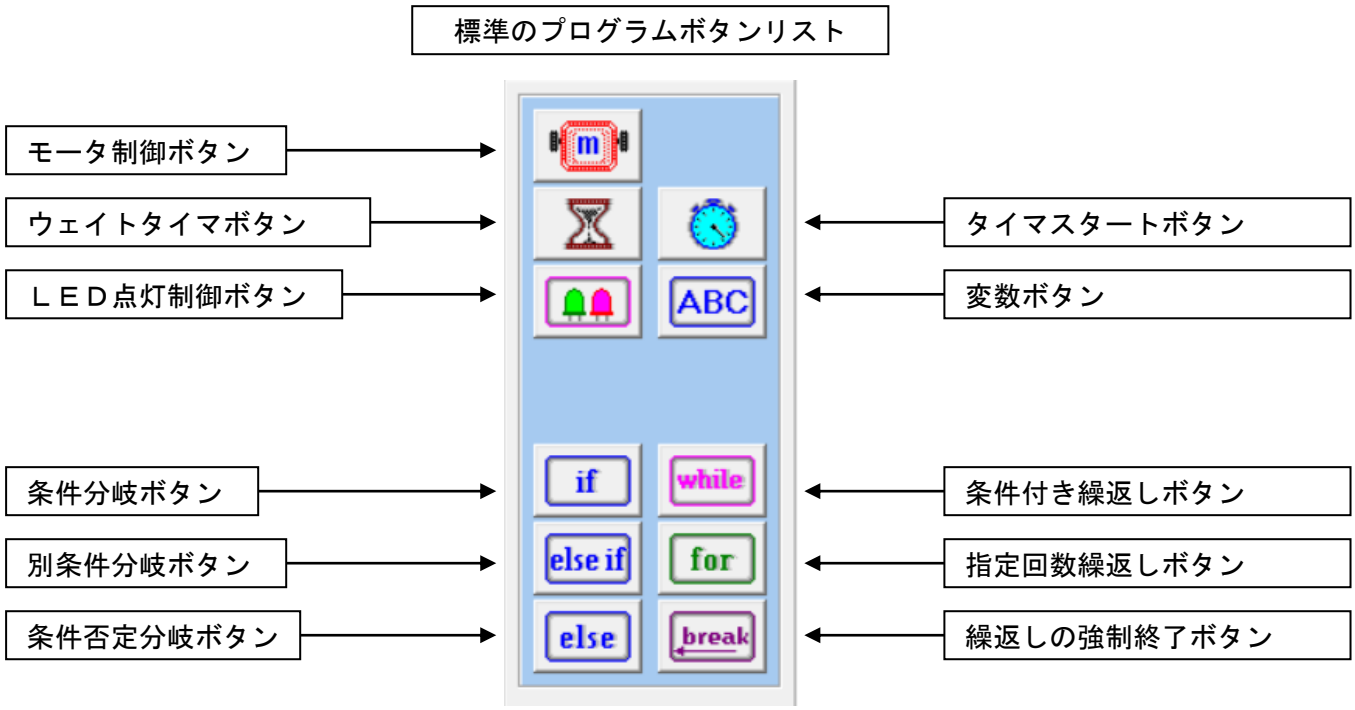
2. プログラムボタンの挿入、削除、コピー、貼付けの説明
   2-1. ボタンの挿入 ---------- １３
   2-2. ボタンの削除 ---------- １４
   2-3. ボタンのコピーと貼付け ---------- １５

3. 入出力設定と拡張機能の設定
   3-1. Setup ボタンの表示と入出力設定 ---------- １６
   3-2. 拡張機能の設定 ---------- １８
   3-3. サーボモータを使う ---------- １９
   3-4. ブザーボード(DSR1801)を使う ---------- ２０
   3-5. 16x2 LCD ボード(DSR1416)を使う ---------- ２１
   3-6. 4ch/6ch モータボードを使う ---------- ２２
   3-7. 超音波センサ(4chUSS：DSR1608)を使う ---------- ２３
   3-8. 多機能電子コンパス(9D-Compass：DSR1603)を使う ---------- ２５
   3-9. Pixy Cam.を使う ---------- ２７
   3-10. 複数台の Alpha-Xplorer を接続する ---------- ３０

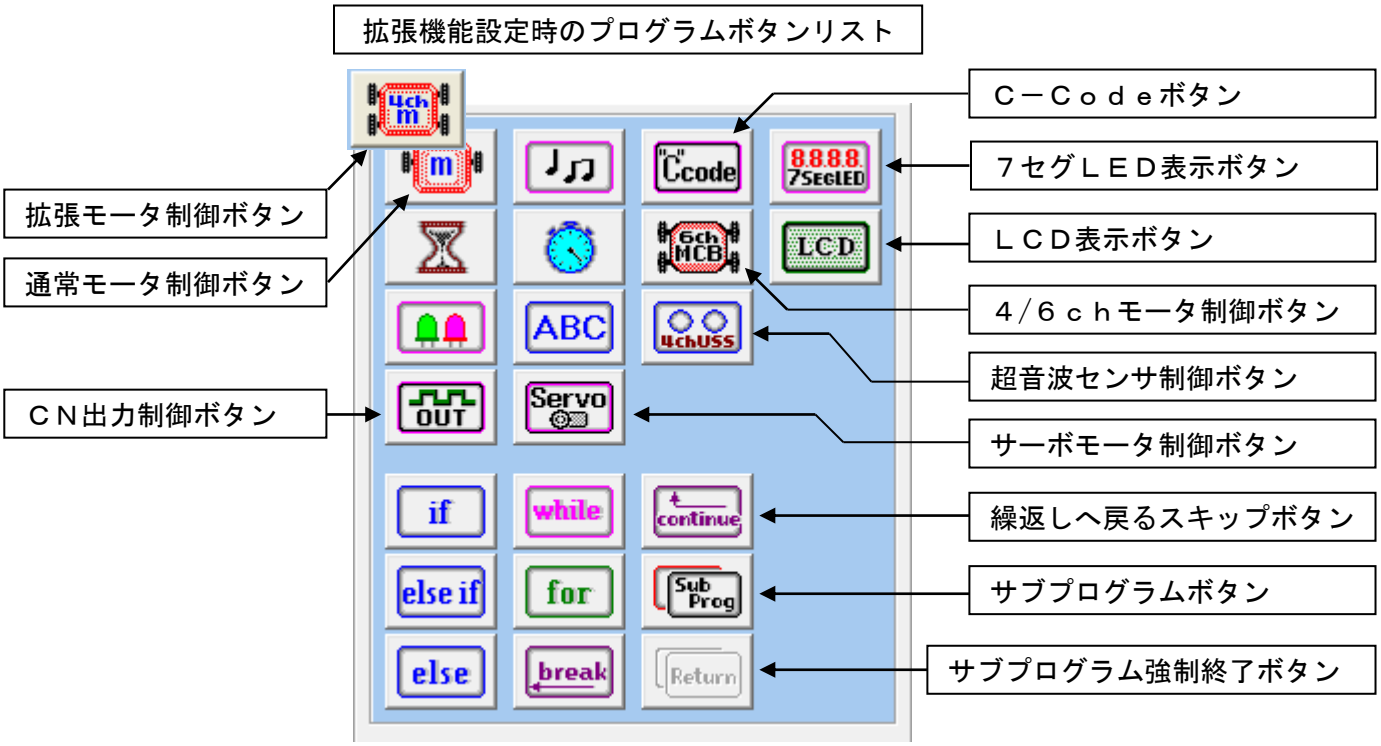
4. サブプログラムの説明
   4-1. サブプログラムボタンの表示 ---------- ３１
   4-2. サブプログラムの編集 ---------- ３２
   4-3. タイマ割込み内で実行するサブプログラム ---------- ３３

１．　プログラムボタンの説明

1-1. プログラムボタンリスト



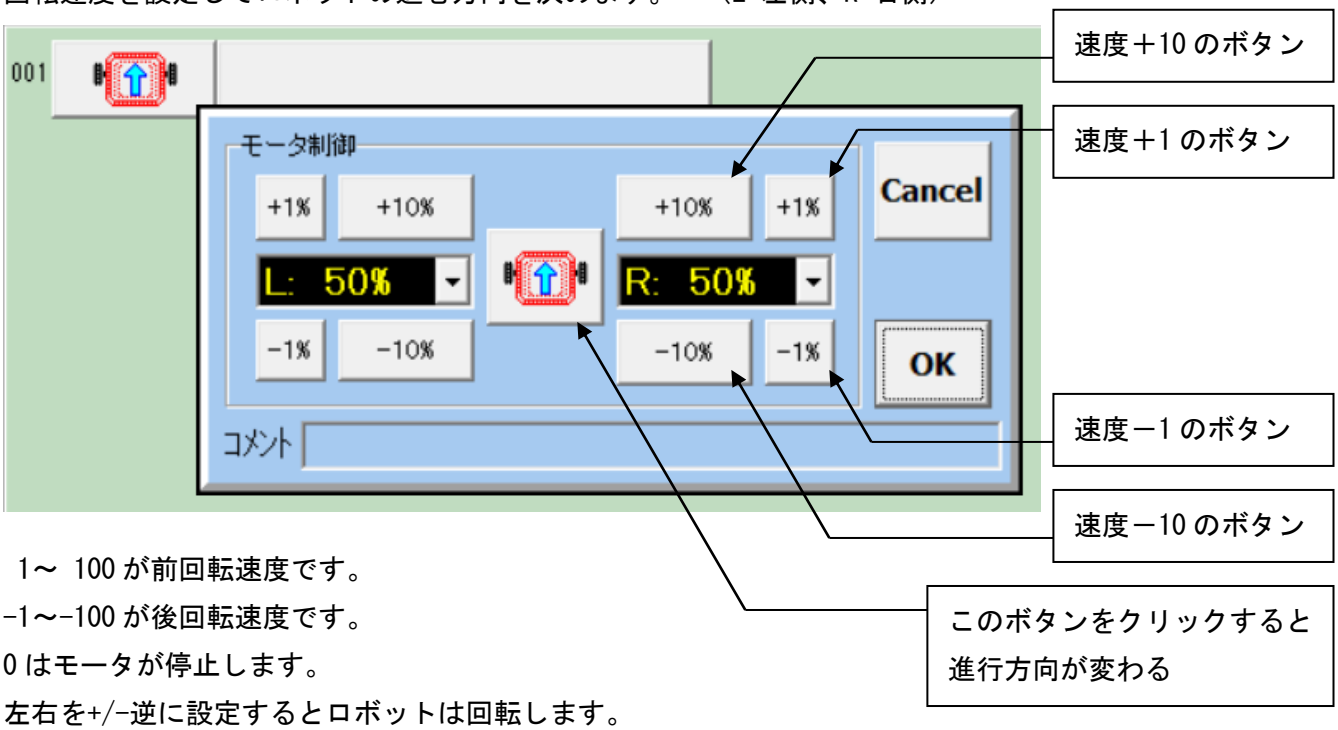
◆拡張機能を設定するとプログラムボタンリストも下記のように変わります。



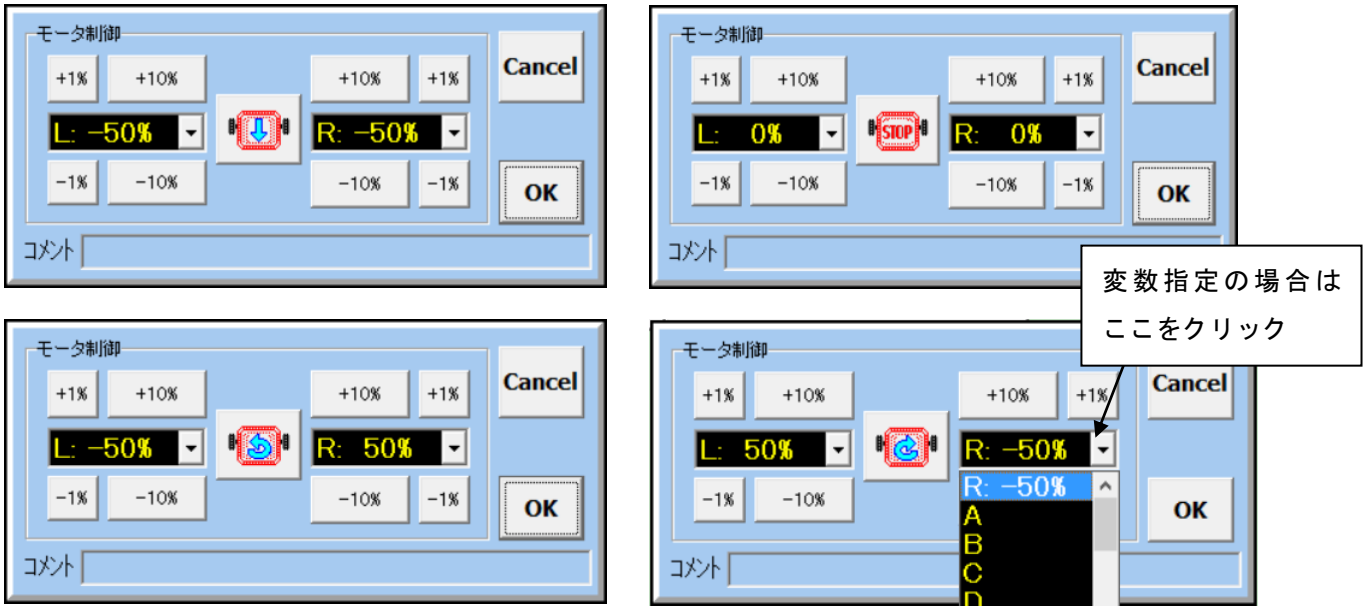
詳しくは、「３. 入出力設定と拡張機能の設定」のページを参照して下さい。

1-2. モータ制御ボタン

回転速度を設定してロボットの進む方向を決めます。　（L:左側、R:右側）

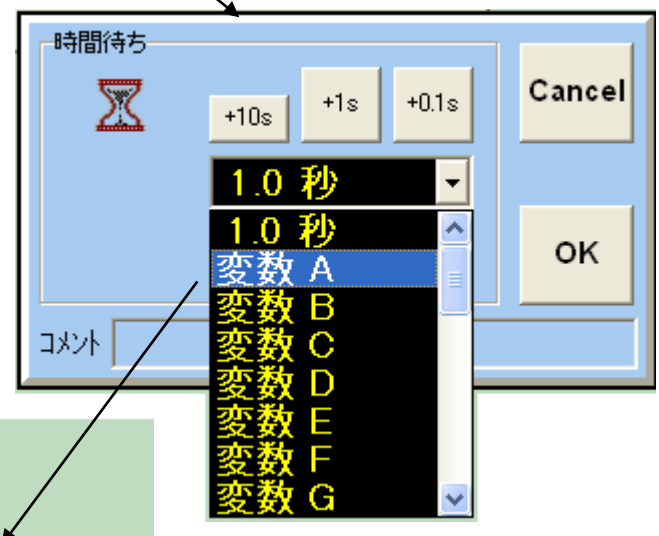


1～ 100 が前回転速度です。
-1～-100 が後回転速度です。
0 はモータが停止します。
左右を+/-逆に設定するとロボットは回転します。
中央のモータボタンをクリックすると前進、後退、停止、左回転、右回転の順に進行方向が簡単に設定することが出来ます。左右どちらかの回転速度を設定してから中央のモータボタンをクリックすると、反対側のモータの回転速度も同じ値に設定されます。

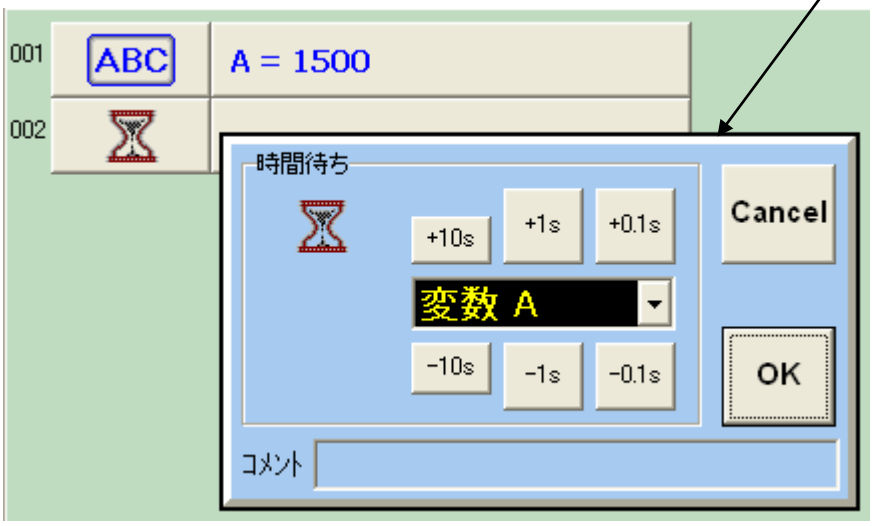


前方向に旋回または、後方向に旋回にするには、速度ボタンを操作して決めます。
速度表示のプルダウンメニューを表示して変数 A～Z で指定することも出来ます。

1-3. ウェイトタイマボタン

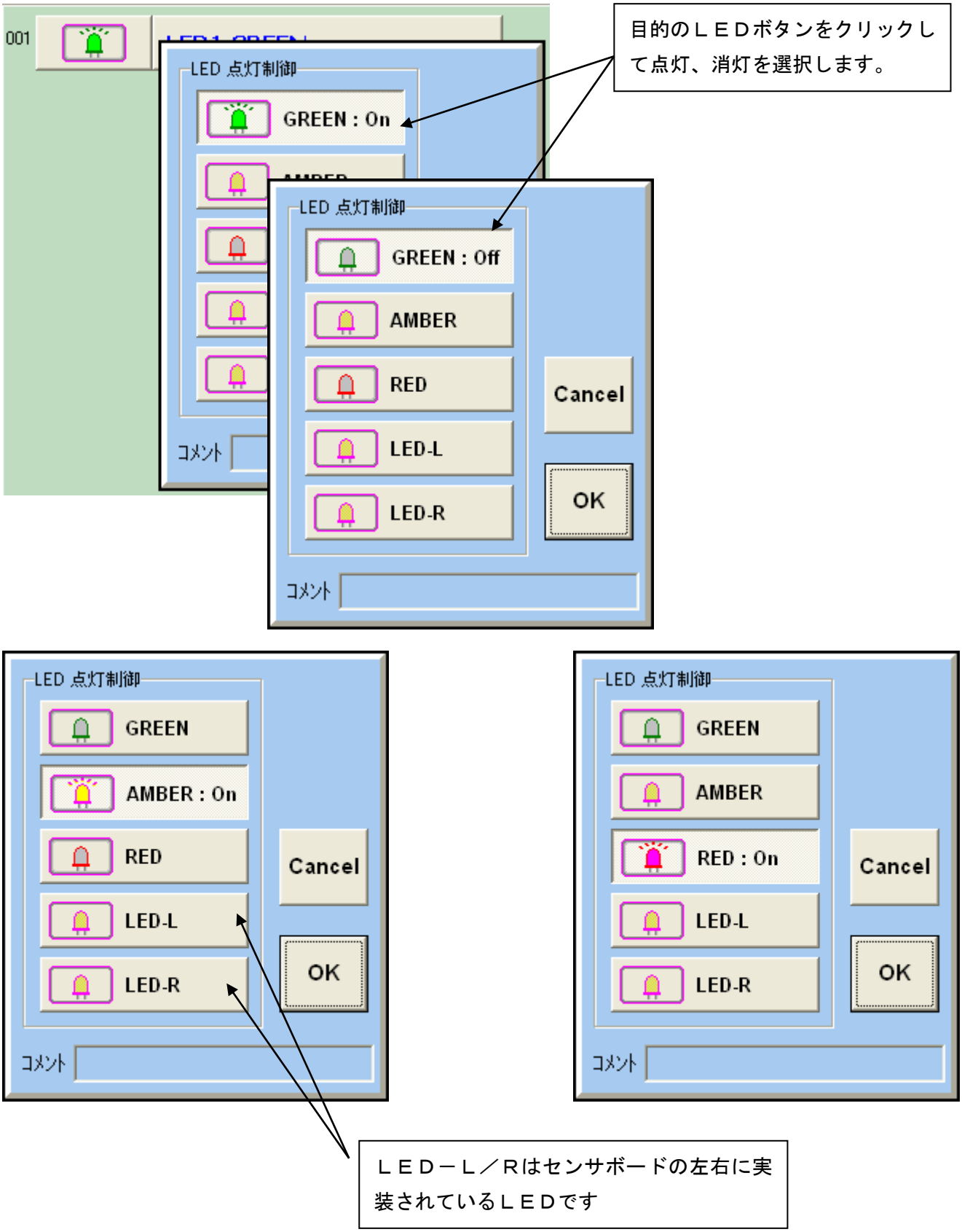


0.1 秒から 60.0 秒までの時間待ちを設定します。
待ち時間を変数に設定することも出来ます。
変数へ待ち時間を設定する場合は、1000 倍したミリ秒の単位で設定します。
例えば、1.5 秒の待ち時間を変数へ設定する場合は、下図のようにします。

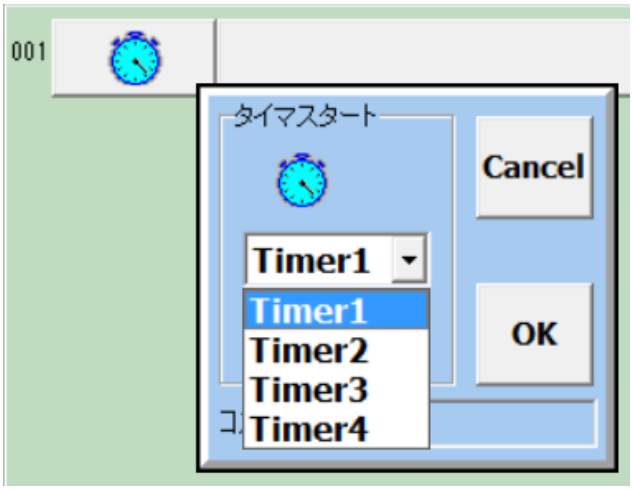


ウェイトタイマボタンは、簡単に待ち時間をプログラムすることができますが、待っている間は、他のプログラムボタンを置いて制御することが出来ません。
待っている間に他のプログラムボタンを置いて制御したい場合は、while if を組み合わせて行います。 「1-5. タイマスタートボタン」項の例題で説明しています。

1-4. ＬＥＤ制御ボタン


001
LED 点灯制御
GREEN : On
LED 点灯制御
GREEN : Off
AMBER
RED
LED-L
LED-R
Cancel
OK
コメント
LED 点灯制御
GREEN
AMBER : On
RED
LED-L
LED-R
Cancel
OK
コメント
LED 点灯制御
GREEN
AMBER
RED : On
LED-L
LED-R
Cancel
OK
コメント
目的のＬＥＤボタンをクリックして点灯、消灯を選択します。
ＬＥＤ－Ｌ／Ｒはセンサボードの左右に実装されているＬＥＤです

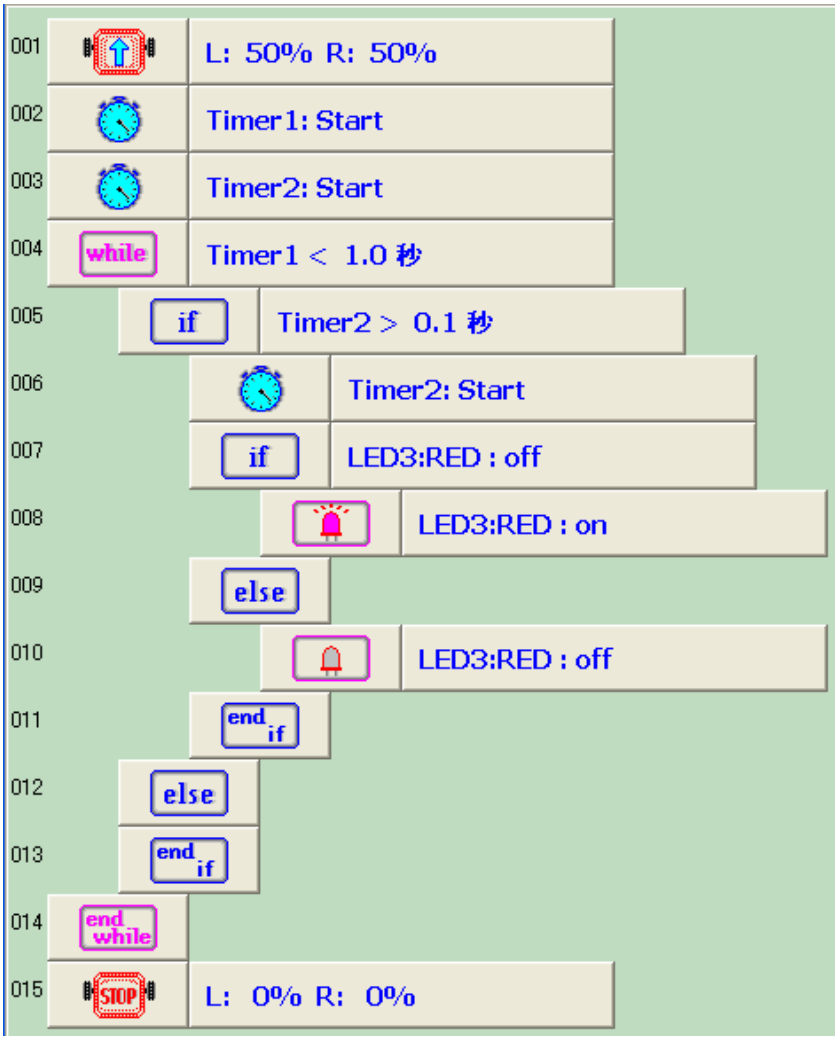
1-5. タイマスタートボタン



条件判定のタイムチェックで使用するタイマをスタートさせます。
４個のタイマを選択して使用します。

ストップウォッチのスタートボタンを押す感じと同じですが、停止はありません。

タイマを使ったプログラム例



この例は、導入編の説明で１秒間前進して停止させるプログラムでウェイトタイマの代わりにタイマボタンを使って同じ動作を実現しながらＬＥＤの点滅をしています。

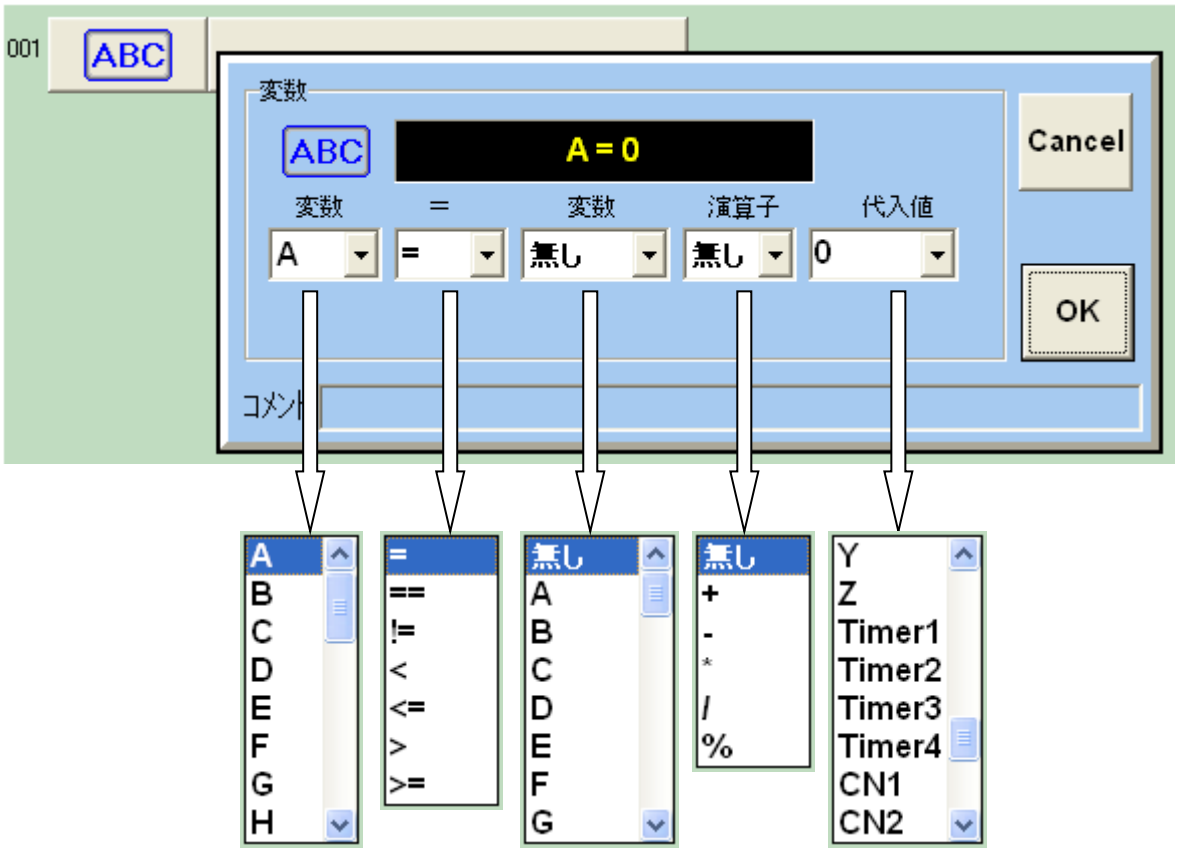
ウェイトタイマを使用した場合、ただ 1.0 秒間待つだけで、他のプログラムを実行することが出来ませんでしたが。タイマを使った場合 1.0 秒間に他のプログラムを実行することが出来ます。

この例では、Timer1 で 1.0 秒間前進する間に Timer2 を使って 0.1 秒間隔で LED を点滅することが出来ます。

4 行目、5 行目のタイマチェックについては、「1-10. タイマチェック」の項で説明しています。

1-6. 変数ボタン ABC

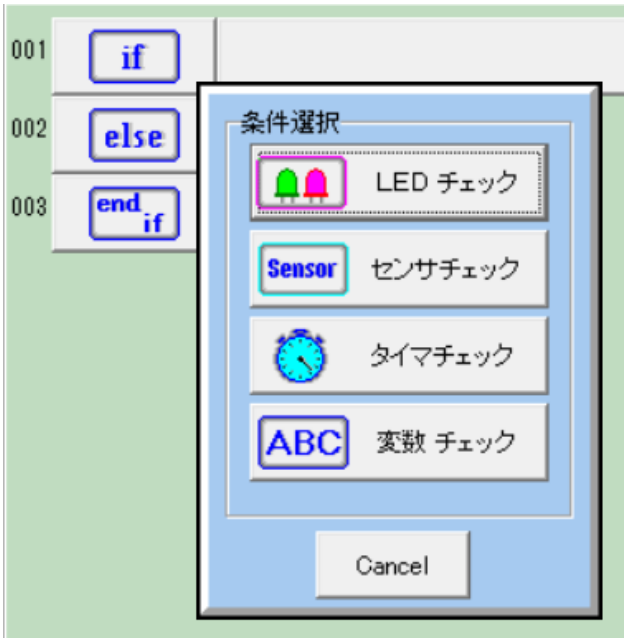
変数は、Ａ～Ｚの２６個が使用出来ます。
演算子と＋、－、×、÷、余り計算などが行えます。
またタイマの値や CN1 から CN10 に接続されたセンサ値も変数へ代入することが出来ます。



◇変数への代入例

“A = 0”	変数 A に 0 を代入
“A = 100”	変数 A に 100 を代入
“B = A”	変数 B に変数 A を代入
“A = Timer1	変数 A にタイマ１の値を代入
“A = A + 1”	変数 A に変数 A の値に１を足した値を代入（変数 A は以前の値より１増す）
“A = A * 2”	変数 A に変数 A を 2 倍にした値を代入（変数 A は以前の値の倍になる）
“A = A / 2”	変数 A に変数 A を 2 で割った値を代入（変数 A は以前の値の半部になる）
“A = B % 2”	変数 A に変数 B を 2 で割った余りの値を代入（変数 A は 0 また 1 となる）
“A = B - CN2”	変数 A に変数 B から CN2 に接続されたセンサ値を引いた値を代入

1-7. 条件分岐



if イフと呼びます。（ 条件成立で次の行からプログラムを実行します。）
条件成立で、この間に置かれたプログラムボタンが実行されます

else if エルス イフと読みます。（条件成立で次の行からプログラムを実行します。）
条件成立で、この間に置かれたプログラムボタンが実行されます。

else エルスと呼びます。（条件不正立の場合に次の行からプログラムを実行します。）
どの条件も成立しなかった時、この間に置かれたプログラムボタンが実行されます。

end if エンド イフと呼びます。

「if」と「end if」の組合せで必ず配置されます。
「if」と「else if」ボタンを置いた時に、ＬＥＤチェック、センサチェック、タイマチェック、変数チェックの判定条件を選択することが出来ます。
「else if」は「if」ボタンの後ろに幾つでも置けます。

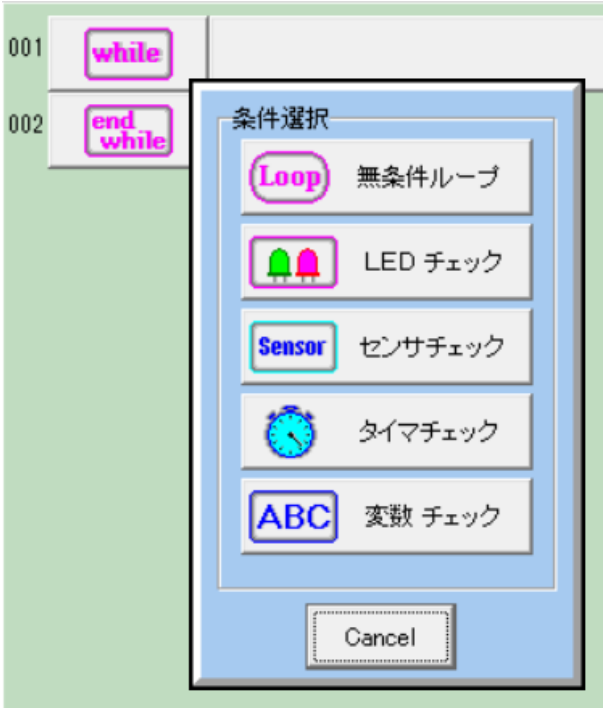
「else」ボタンは「end if」ボタンの前に１個だけ置けます。

条件が成立した場合は、次の行からプログラムは実行され、「else if」または「else」ボタンに出会うと「end if」までスキップします。
条件が不正立の場合は、次の「else if」か「else」までスキップし、無ければ「end if」までプログラムはスキップします。

条件分岐ボタンの間にその他のボタン（さらに条件分岐ボタンでも良い）を挿入するには、
挿入したいプログラムボタンをボタンリストから選択しておいてから、挿入したい行のプログラムボタンをクリックすると挿入されます。
例えば、１行目と 2 行目の間にプログラムボタンを挿入する場合は 2 行目の else ボタンをクリックします。

プログラムボタンの編集方法は、「２．プログラムボタンの挿入と削除」で説明しています。

1-8. 条件付き繰返し



while ホワイルと呼びます。
条件成立の間、この間に置かれたプログラムボタンが繰返し実行されます。
end while エンド ホワイルと呼びます。

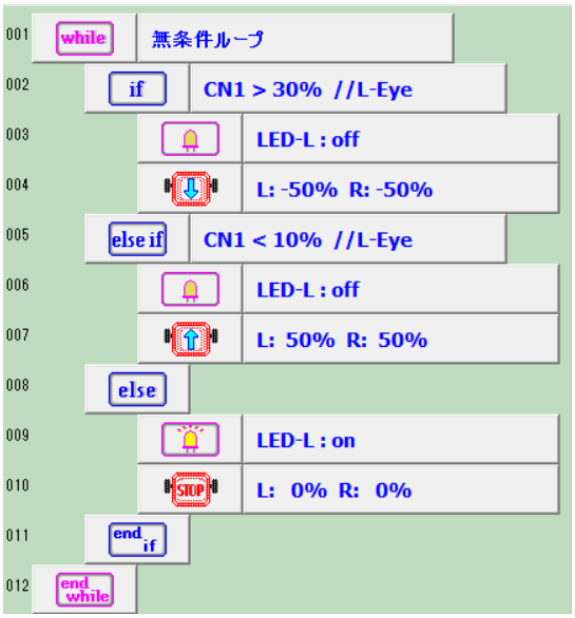
「while」と「end while」ボタンの組合せで必ず配置されます。

別のプログラムボタンを挿入する場合は、先にプログラムボタンリストから挿入したボタンを選択しておいてから、挿入したい行のボタンをクリックします。(条件分岐の場合と同じ)

break ブレイクと呼びます。 繰返しの中にだけ置けるボタンで、このボタンに出会うと、繰返し条件を無視して強制終了します。(「end while」までのプログラムボタンは実行されません)

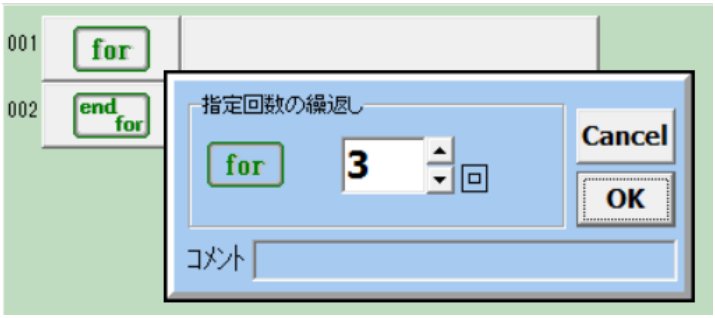
continue コンティニュウと呼びます。 繰返しの中にだけ置けるボタンでこのボタンに出会うと、繰返しの条件判定に戻ります。(「end while」ボタンまでのプログラムボタンは実行されません)
このボタンは拡張プログラムボタンの説明で“Advanced Mode”の設定で表示されます。

while, if, elseif, else プログラム例



while(001)～end while(012)の範囲を無条件で繰り返す
002～004：CN1 の L-Eye センサが 30%より大きい場合
　　LED-L を消灯
　　モータを後退
005～007：CN1 の L-Eye センサが 10%より小さい場合
　　LED-L を消灯
　　モータを前進
008～010：上記の条件以外の場合は
　　LED-L を点灯
　　モータは停止

1-9. 回数指定の繰返し



for フォアと呼びます。
指定回数、この間に置かれたプログラムボタンを実行します。

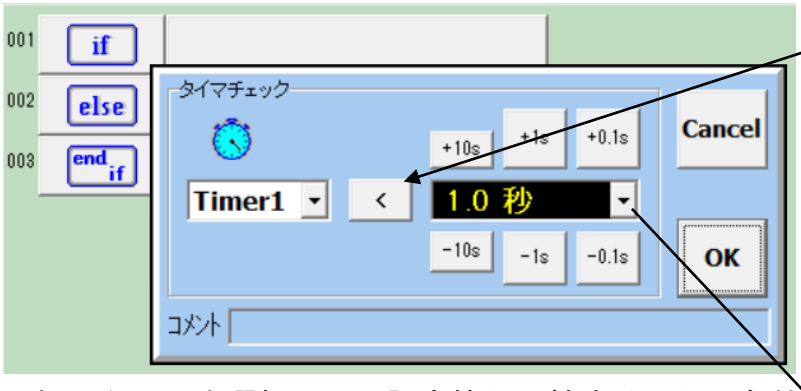
end for エンド フォアと呼びます。

「for」と「end for」の間にプログラムボタンを挿入する場合は、先にプログラムボタンリストから挿入したボタンを選択しておいてから、挿入したい行の先頭ボタンをクリックします。(条件分岐の場合と同じ)

break ブレイクと呼びます。　　繰返しの中にだけ置けるボタンで、このボタンに出会うと、繰返し回数を無視して強制終了します。(「end for」までのプログラムボタンは実行されません)

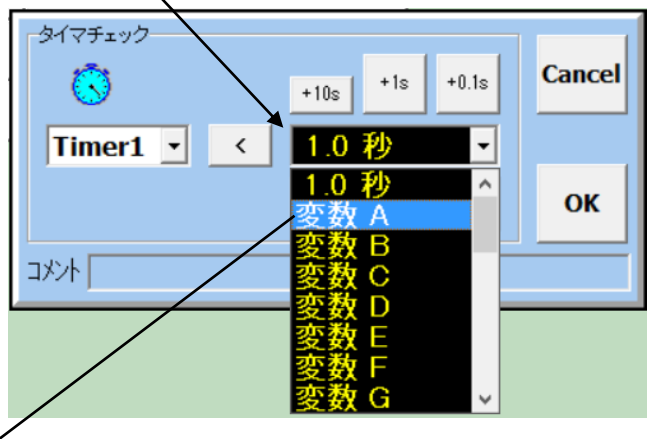
continue コンティニュウと呼びます。 繰返しの中にだけ置けるボタンでこのボタンに出会うと、繰返し回数の判定に戻ります。(「end for」ボタンまでのプログラムボタンは実行されません)
このボタンは拡張プログラムボタンの説明で“Advanced Mode”の設定で表示されます。

1-10. タイマチェック

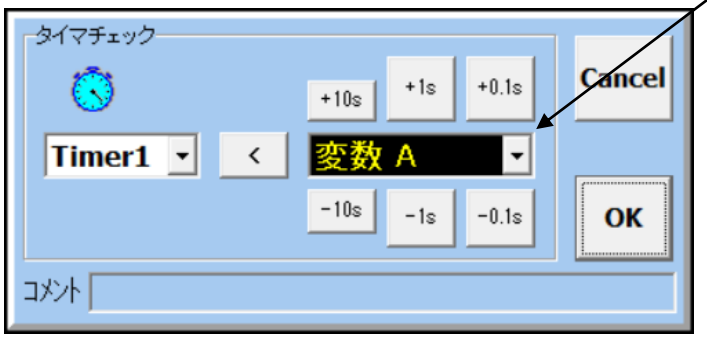


ここをクリックすると ＜ と ＞ が交互に切り換ります。

４個のタイマを選択して、設定値と比較することで条件判定することが出来ます。
設定値のかわりに、変数（Ａ～Ｚ）との比較判定も出来ます。



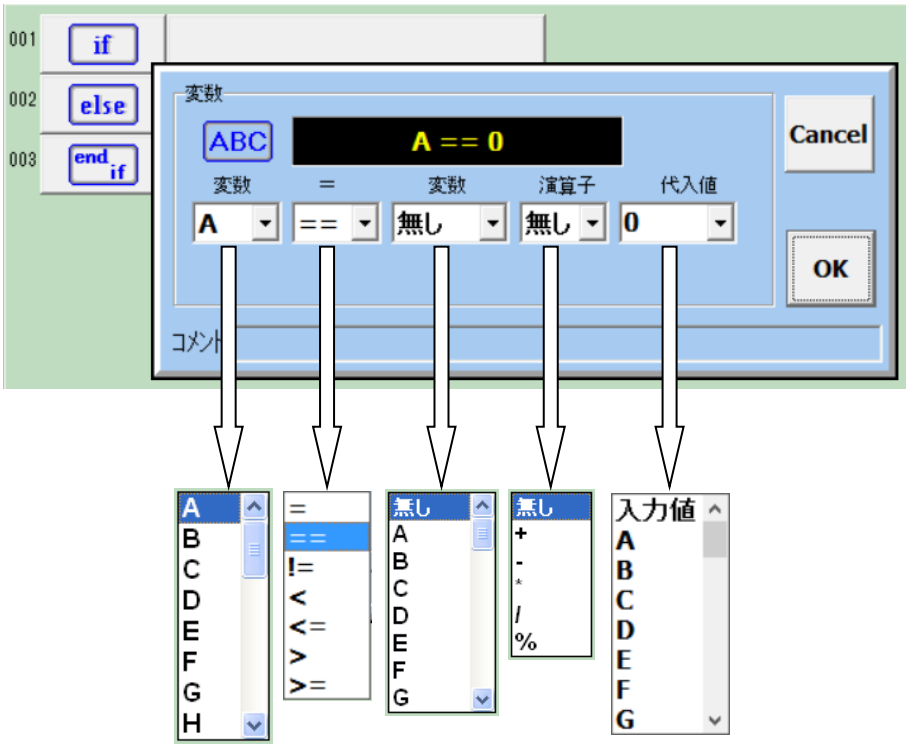
ボタン名称の例
“Timer1 ＜ 0.1 秒より短い”
“Timer2 ＞ 1.5 秒より長い”
“Timer1 ＜ 変数Ａの値より小さい”
“Timer2 ＞ 変数Ａの値より大きい”



◆変数と比較する場合の注意
　変数へは秒単位を１０００倍したミリ秒の単位で値を代入する必要があります。
例えば 1.5 秒の値を変数に代入して比較したい時は、１０００倍して１５００を代入します。

変数Ａ＝１５００
if　Timer1 ＜ 変数Ａ
　　・・・
endif

1-11. 変数チェック



変数は、Ａ～Ｚの２６個が使用出来ます。
演算子と＋、－、×、÷、余り計算などが行えます。
また、タイマの値や CN1 から CN10 に接続されたセンサの値も変数へ代入し比較することが出来ます。

ボタン名称の例

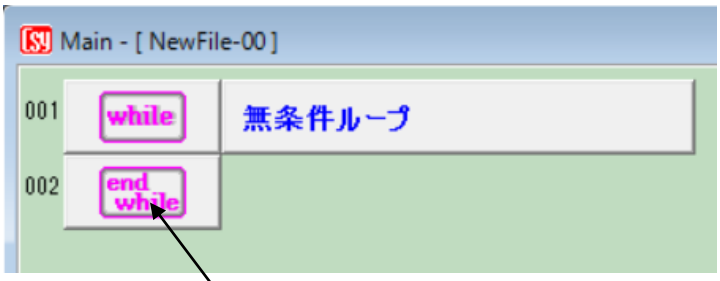
“A == 0”	変数 A が 0 の場合
“A != 0”	変数 A が 0 で無い場合
“A <= 5”	変数 A が 5 以下の場合
“A >= B”	変数 A が変数 B 以上の場合
“A < CN1”	変数 A が CN1 のセンサ値より小さい場合
“A > CN3”	変数 A が CN3 のセンサ値より大きい場合
“A < B - CN2”	変数 A が変数 B から CN2 のセンサ値を引いた値より小さい場合
“A < Timer3”	変数 A がタイマ３の値より小さい場合
“A == B + 2”	変数 A が変数 B に 2 を足した値と等しい場合
“A == B - 2”	変数 A が変数 B から 2 を引いた値と等しい場合
“A == B * 2”	変数 A が変数 B に 2 を掛けた値と等しい場合
“A == B / 2”	変数 A が変数 B を 2 で割った値と等しい場合
“A == B % 2”	変数 A が変数 B を 2 で割った余りと等しい場合

変数への入力値範囲は -2147483648～+2147483647（32 ビット幅の符号付き整数）です。

２． プログラムボタンの挿入、削除、コピー、貼付けの説明

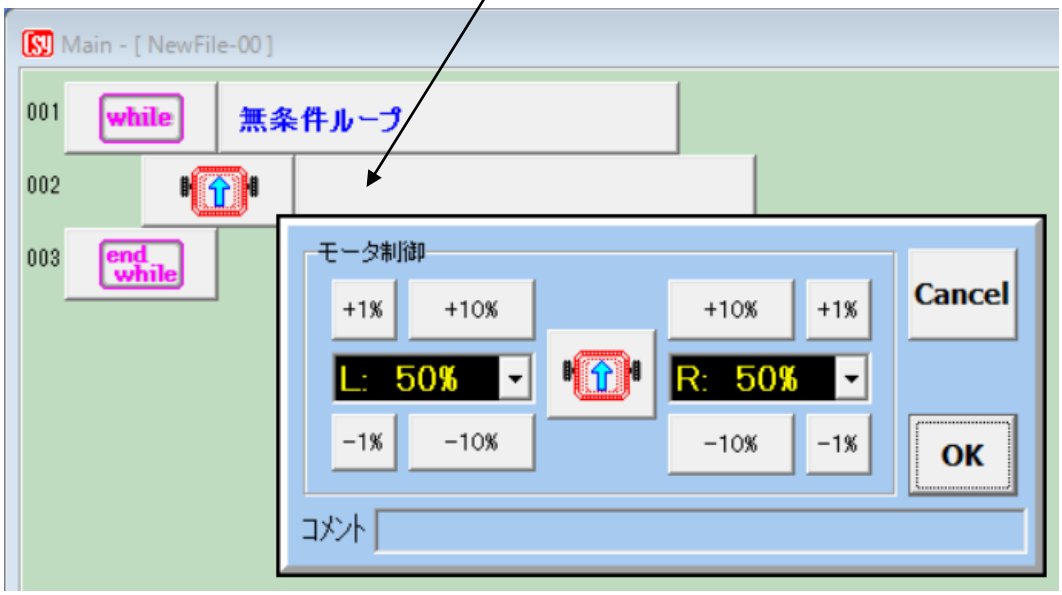
2-1. ボタンの挿入

① 先に、挿入したいボタンをプログラムボタンリストから選択しておきます。



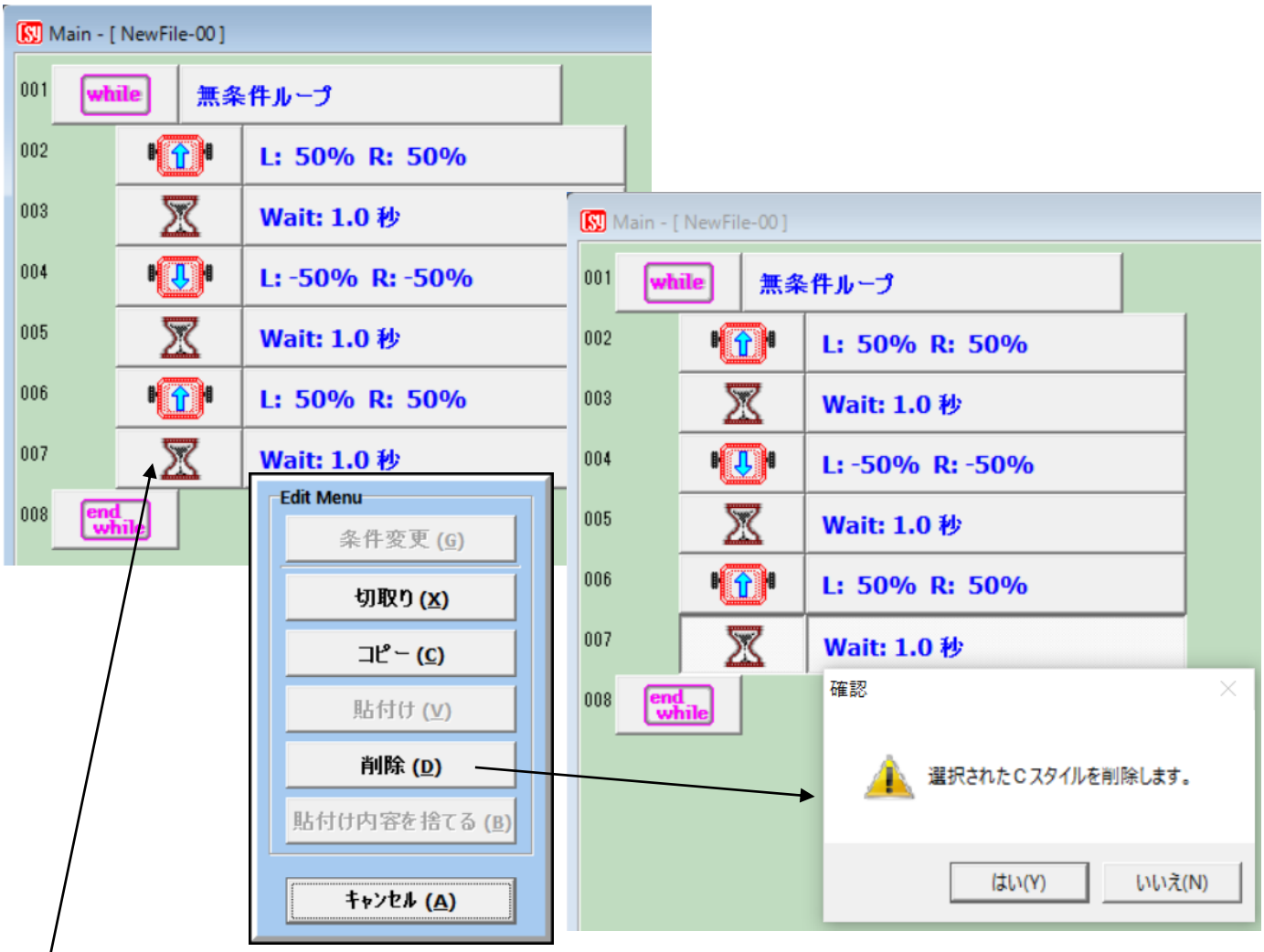
② 次に、挿入したい行の先頭ボタンをクリックすると挿入されます。

③ 挿入されたボタンのダイアログが表示されます。



この例では、モータボタンをプログラムボタンリストから選択してから、「end while」ボタンをクリックすると前進ボタンが挿入されたところです。

2-2. ボタンの削除



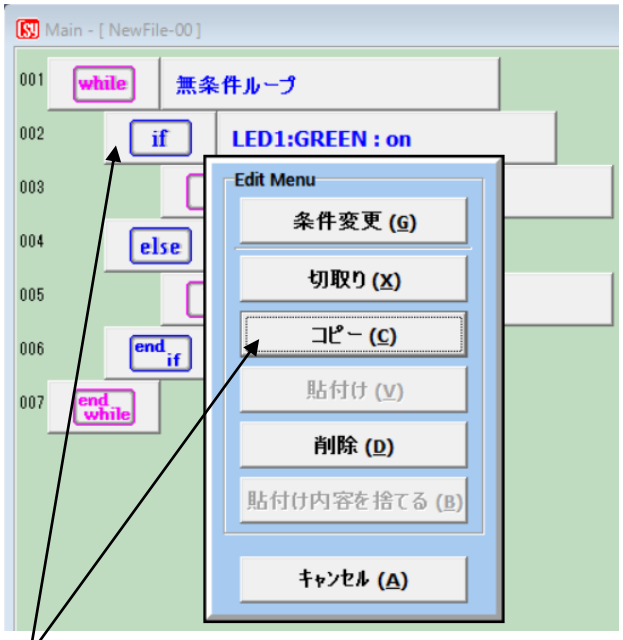
① 削除したい行の前側のボタンをクリックするとポップアップメニューが表示されますので、「削除」を選択します。

② ボタン削除の確認ダイアログが表示されますので、「はい」をクリックすると、その行のボタンが削除され、行が詰められます。

ポップアップメニューには、「切取り」、「コピー」、「貼付け」などの機能もあります。

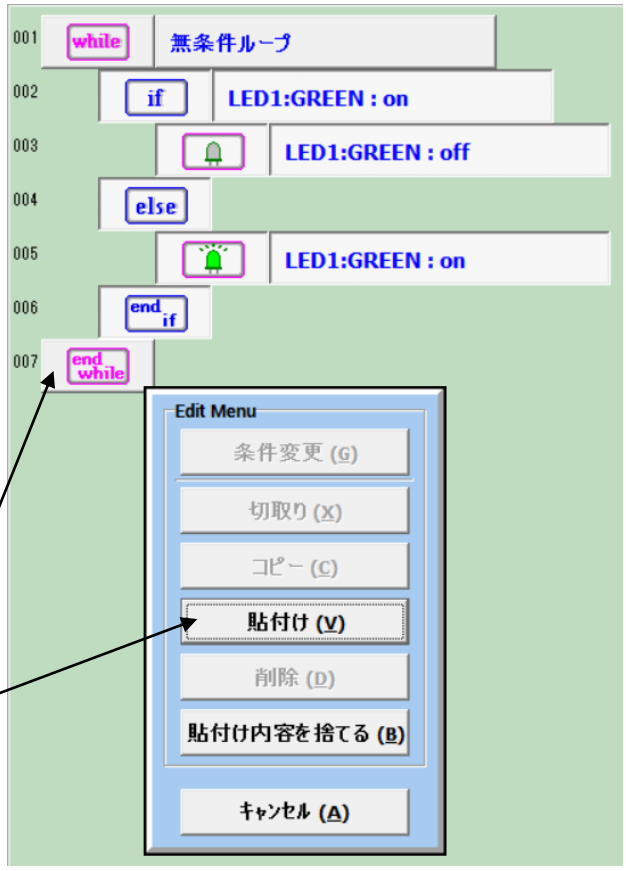
「while」や「for」または「if」ボタンを選択した場合は、「end while」や「end for」または「end if」までの範囲が削除などの対象となります。

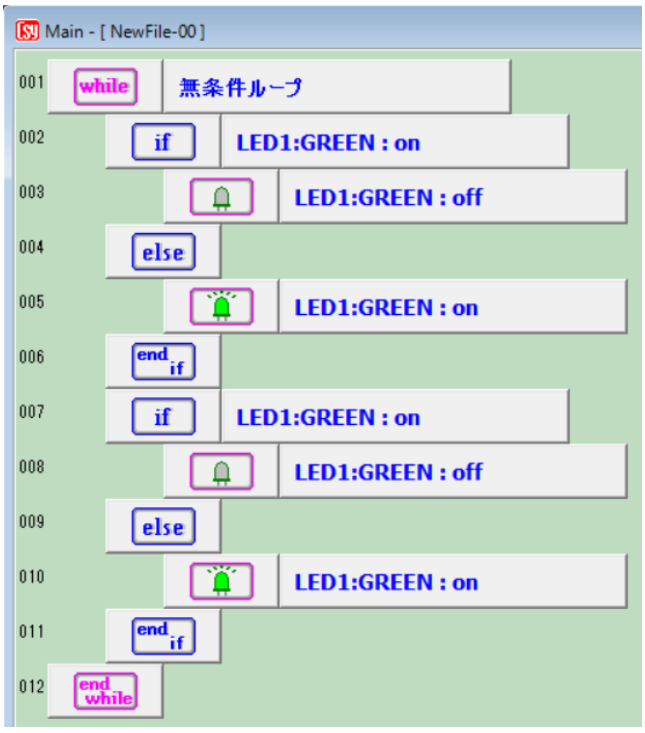
2-3. ボタンのコピーと貼付け



コピーしたいボタンの先頭をクリックして
ポップアップメニューのコピーを選択する

貼付けたいボタンの先頭をクリックしてポ
ップアップメニューの貼付けを選択すると
コピーしたボタンが貼付け（挿入）されます。





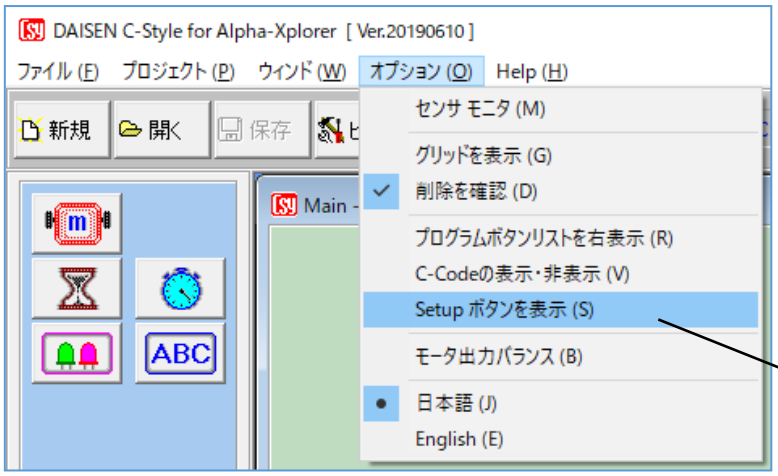
コピーを連続して行うと、以前にコピーした内容の後に追加され、貼付け時に全て挿入されます。この機能は、切取りにおいても同じ働きをします。コピーや切取りで蓄積された貼付け内容がいらなくなった場合は、ポップアップメニューを表示させて、「貼付け内容を捨てる」を選択します。

３． 入出力設定と拡張機能の設定

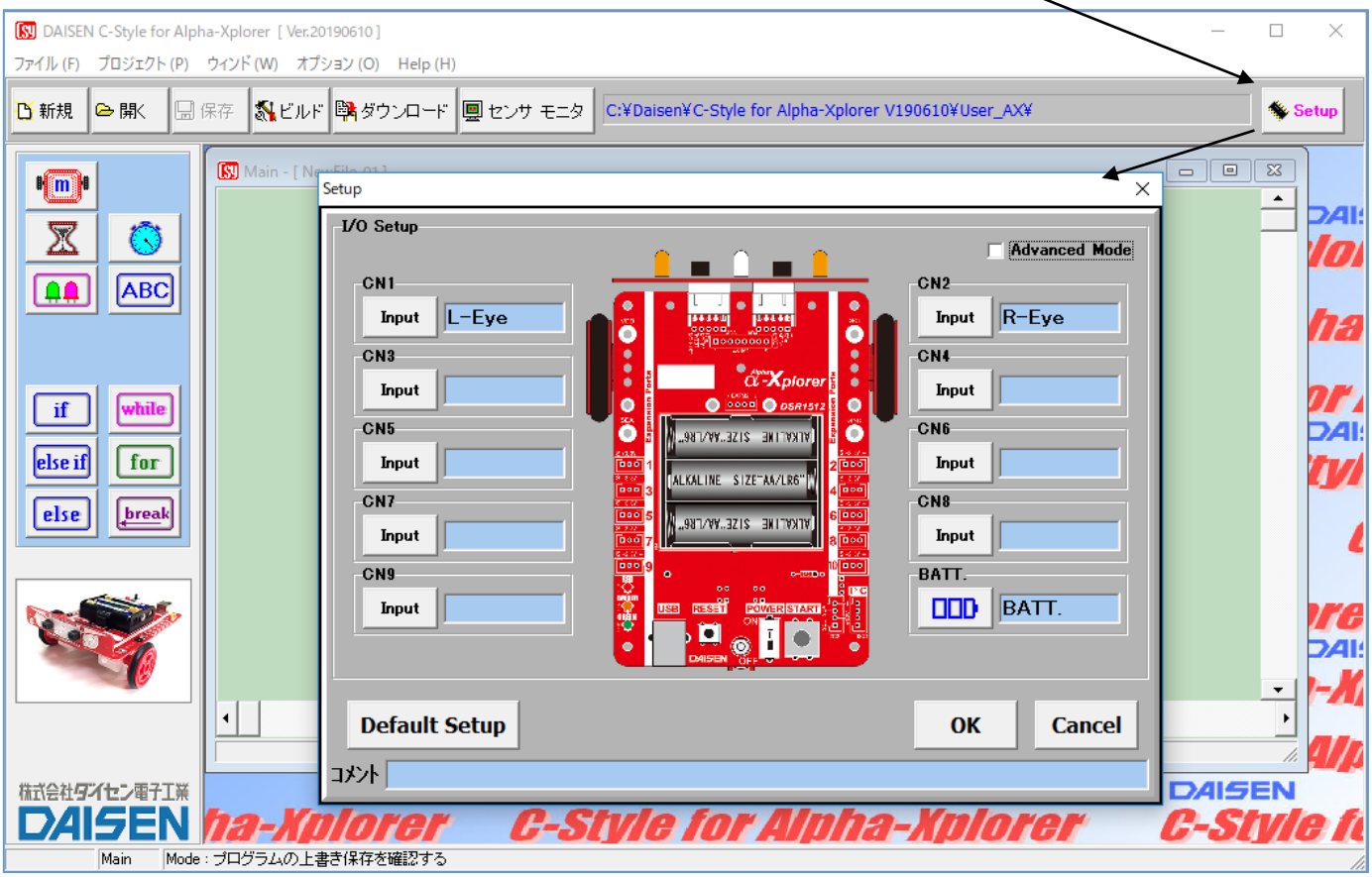
Alpha-Xplorer には、CN1 から CN10 までの入出力端子があります。
出荷時の状態では、CN1 と CN2 は近距離赤外センサが接続され、CN3～CN9 までは予備で、これらはアナログ入力の設定となっています。また CN10 は電池電圧を計測する為に配線されています。これらの入出力設定を変えることが出来ます。更に CN5～CN9 はサーボモータを接続することが出来ます。

また「Advanced Mode」にチェックを付けると「I2C Device」のオプション製品を接続する機能もあります。

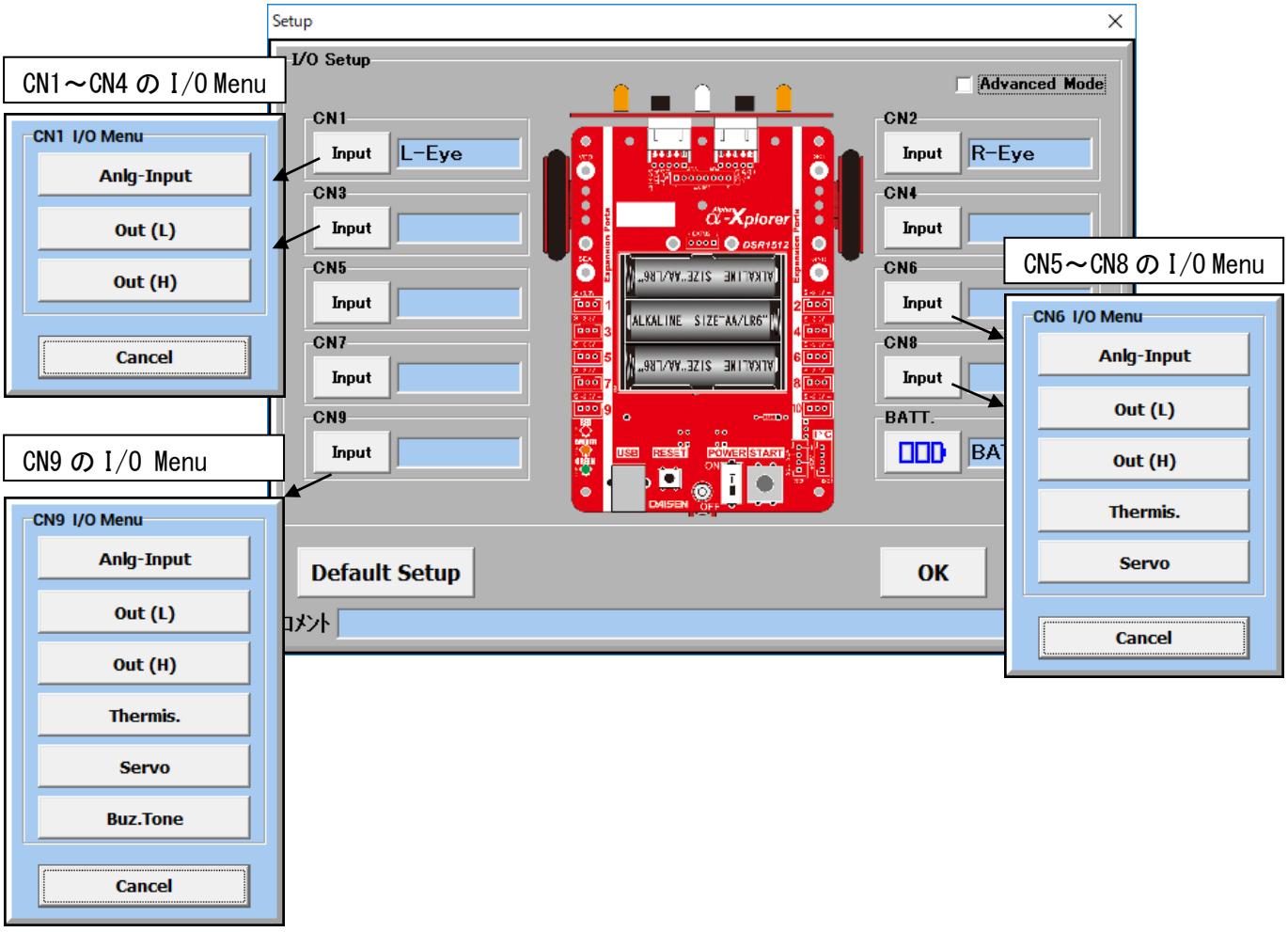
3-1. Setup ボタンの表示と入出力設定



① オプションメニューの「Setup ボタンの表示」を選択します。
② プログラムボタンリストの上側に Setup ボタンが表示され入出力の設定が行える「I/O Setup」ダイアログが表示されます。



入出力設定（I/O Setup）



CN1～CN9 の［Input］ボタンをクリックすると I/O Menu が表示され入出力設定を変更することが出来ます。CN 番号によって選択できる機能がことなります。

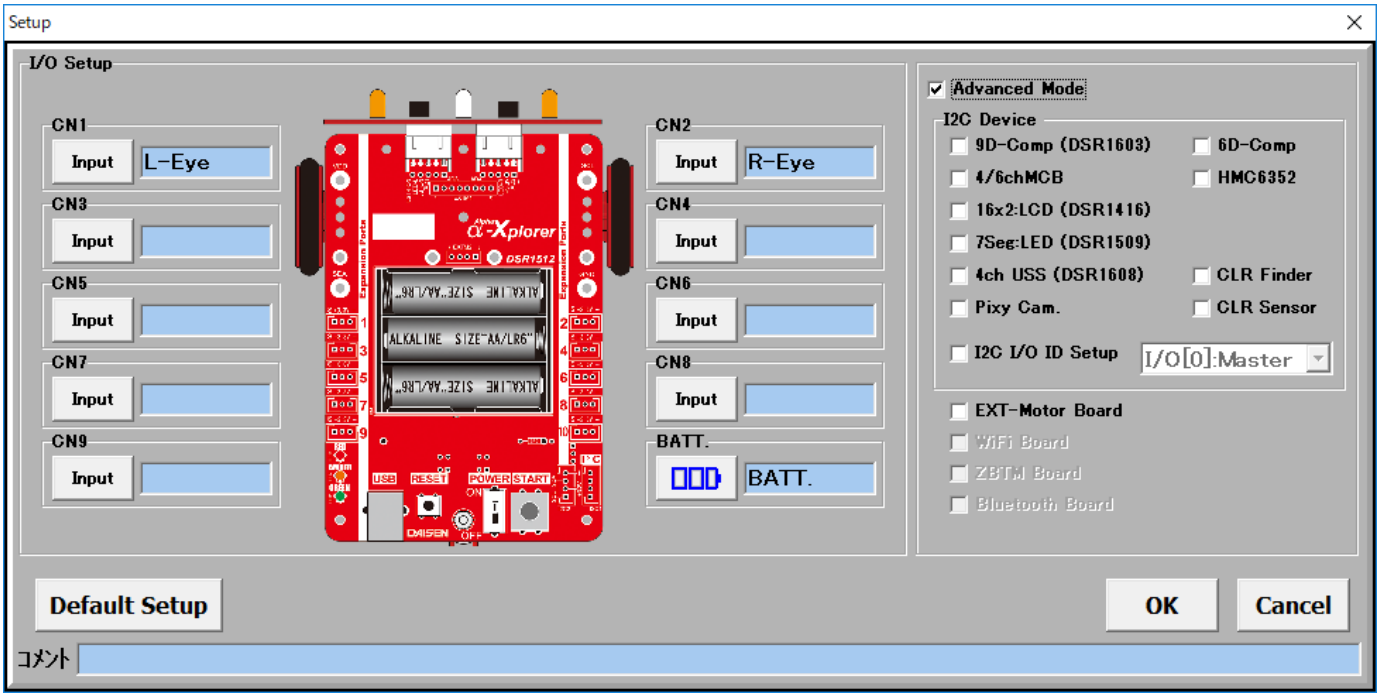
Out(L/H)設定をすると、プログラムボタンリストに ボタンが追加表示されます。
Thermis.設定をすると、サーミスタボード(DSR1802)を該当する CN に接続して温度情報を得ることが出来ます。動作範囲は-10℃～＋50℃です。
Servo 設定をすると、プログラムボタンリストに ボタンが追加表示されます。
Buz.Tone 設定をすると、ブザーボード(DSR1801)を CN9 に接続してブザー音で音階を奏でることが出来ます。プログラムボタンリストに ボタンが追加表示されます。

［ **Default Setup** ］ボタンは入出力設定を初期の状態に戻す時に操作します。

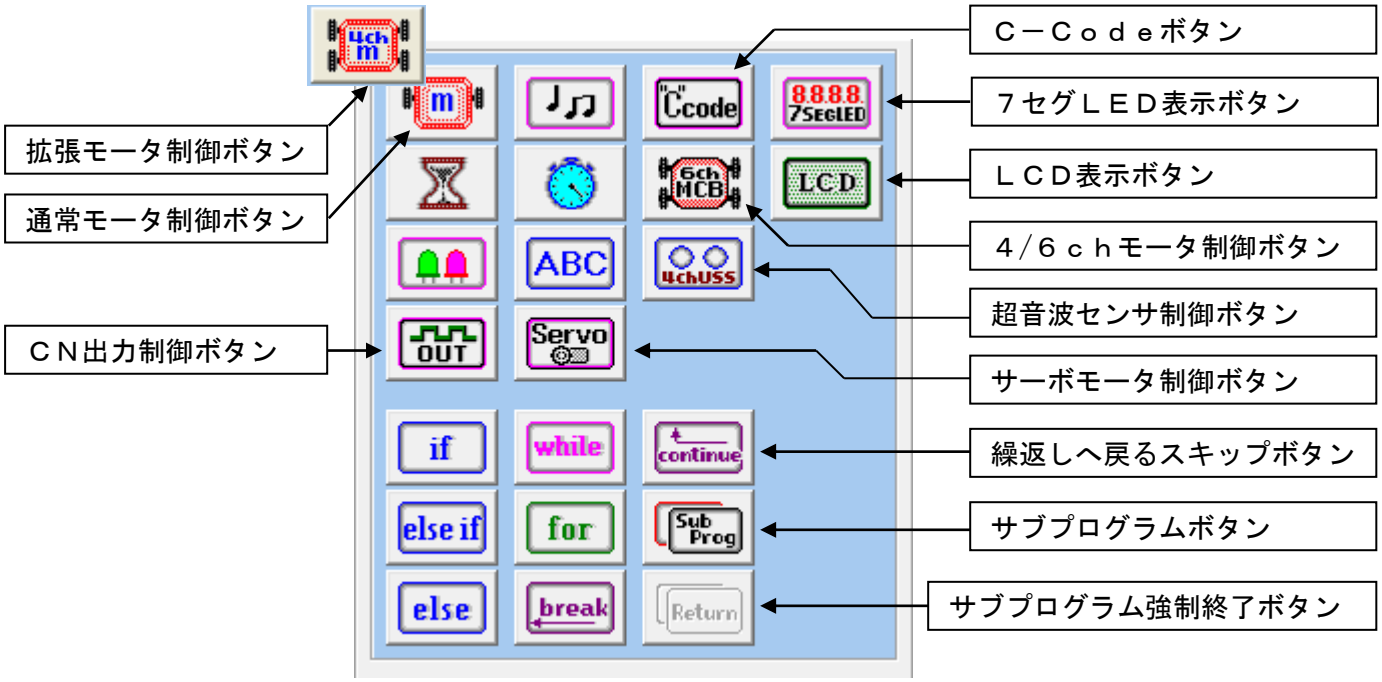
入出力の設定を変更した場合は、C-Style プログラムをビルドしてロボットにダウンロードすることで、はじめて有効となります。

3-2. 拡張機能の設定

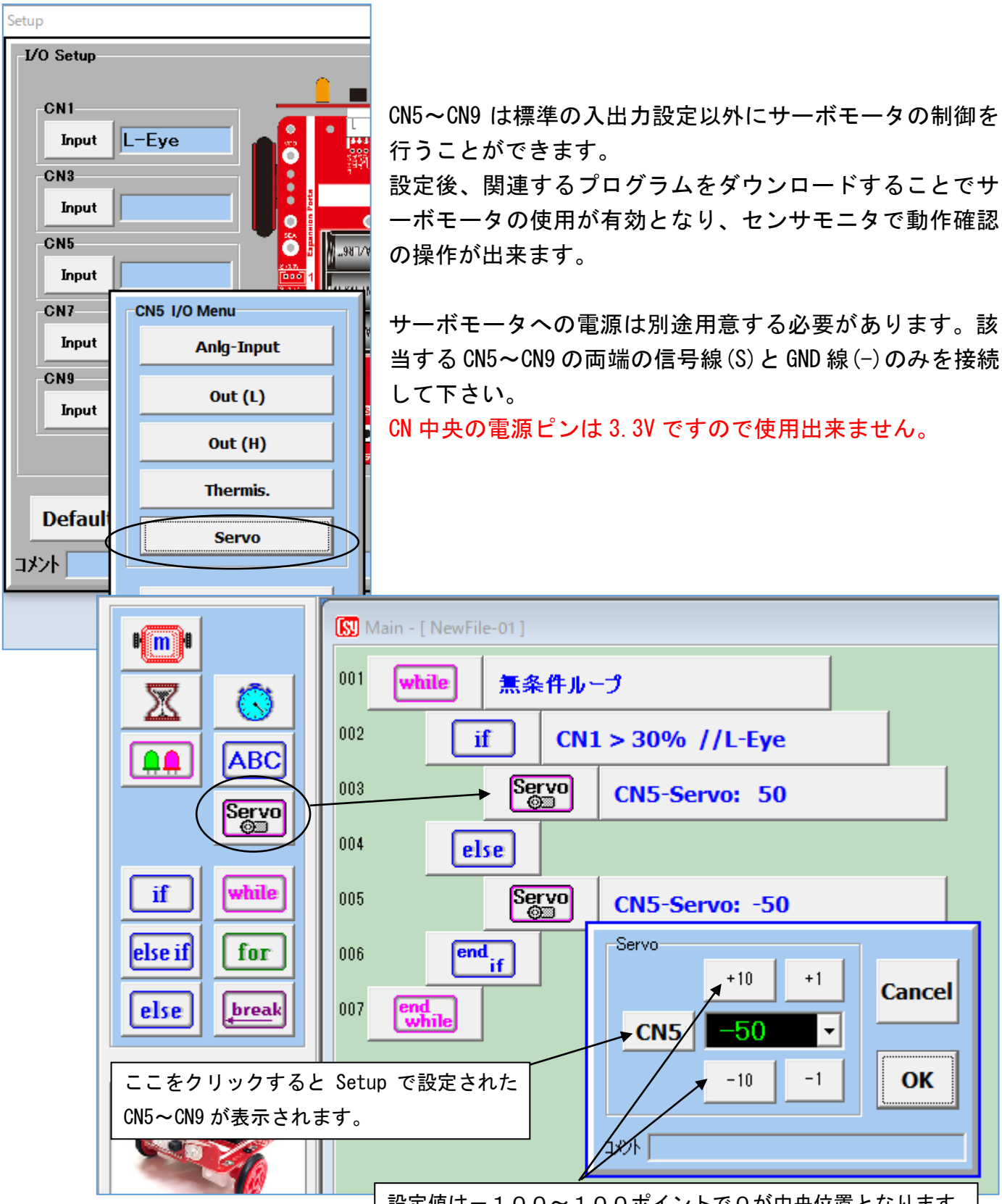
I/O Setup（入出力設定）ダイアログ内の上側にある「Advanced Mode」にチェックを付けると拡張機能を設定するチェックボックスが表示されます。
6D/9D-Compass(多機能電子コンパス：DSR1401/1603)、HMC6352(単機能電子コンパス：DSR1302)、4/6ch モータコントロールボード、16x2 LCD の表示装置、4chUSS 超音波距離センサ, 複数台の Alpha-Xplorer を I2C 接続で拡張センサボードとして使用する機能設定、サブプログラムボタンの表示など、機能拡張の設定が行えます。



I/O Setup の入出力設定や拡張機能の設定を行うとプログラムボタンリスト表示も変わります。



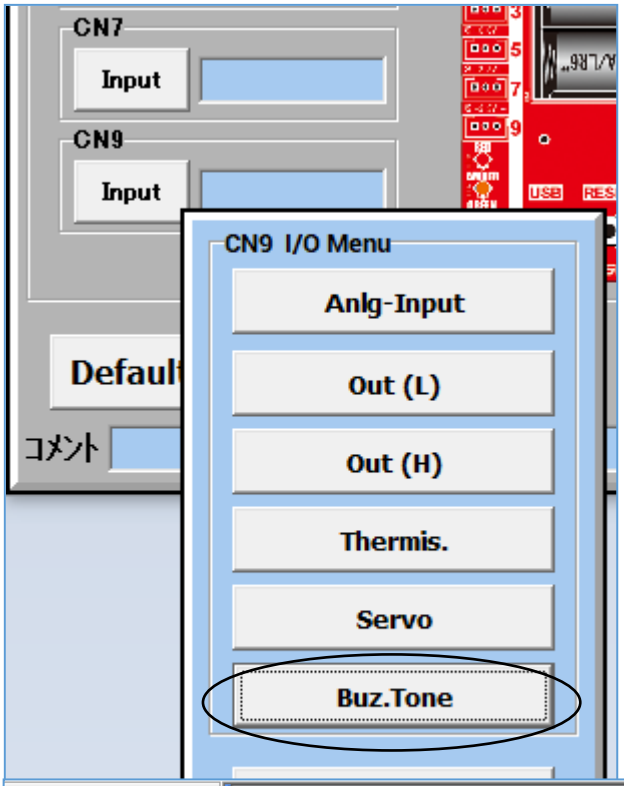
3-3. サーボモータを使う



CN5～CN9 は標準の入出力設定以外にサーボモータの制御を行うことができます。
設定後、関連するプログラムをダウンロードすることでサーボモータの使用が有効となり、センサモニタで動作確認の操作が出来ます。

サーボモータへの電源は別途用意する必要があります。該当する CN5～CN9 の両端の信号線(S)と GND 線(-)のみを接続して下さい。
CN 中央の電源ピンは 3.3V ですので使用出来ません。

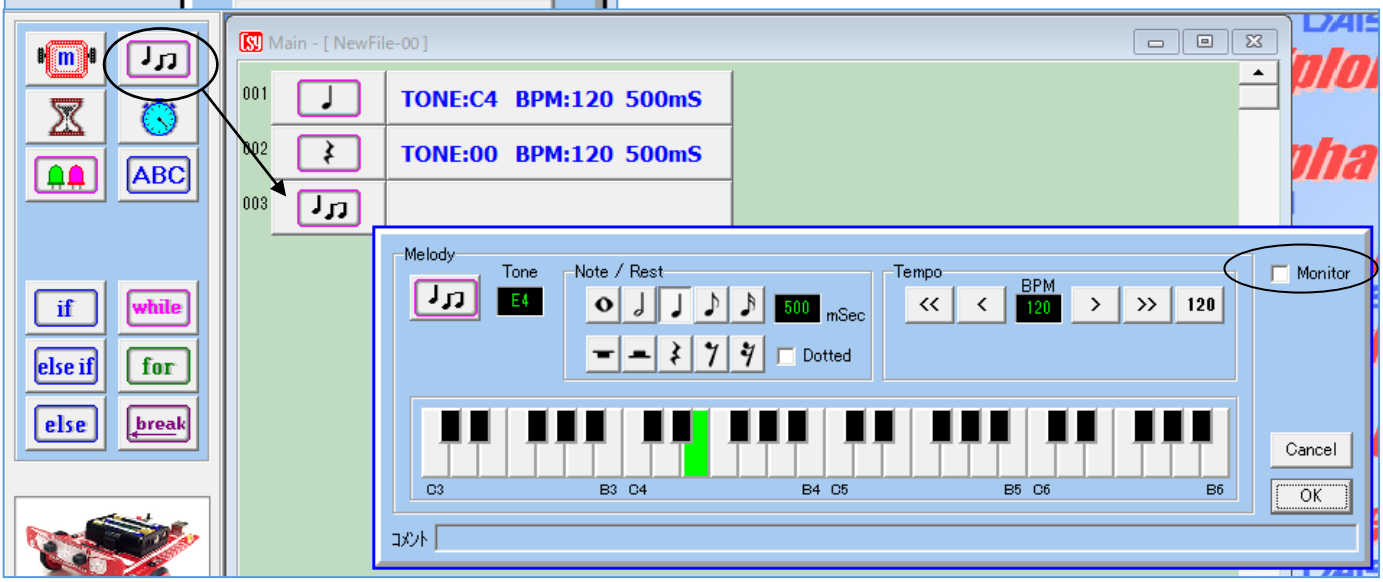
3-4. ブザーボード（DSR1801）を使う



CN9 の I/O Menu で Buz.Tone を選択するとブザーボード(DSR1801)を使って音階を出力することが出来ます。

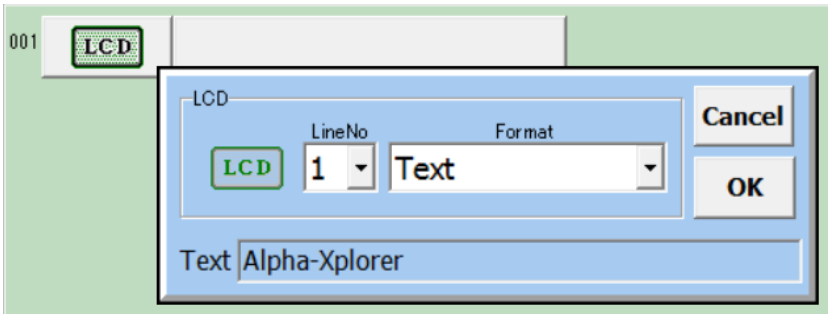
プログラムボタンリストにメロディボタンが追加表示されメロディプログラムを作成することが出来ます。

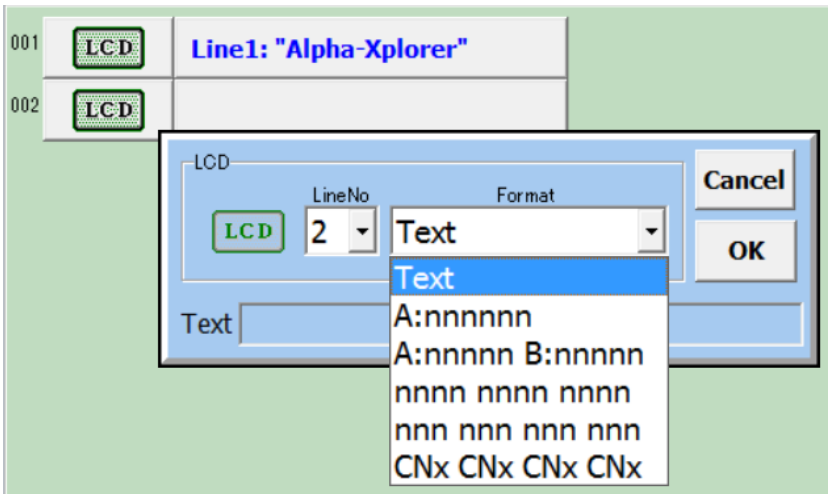
センサ入力と組み合わせてメロディを奏でるプログラムを作成することも出来ます。



メロディ設定ダイアログで以下の設定が行えます。
　音階は C3～B6 の 4 オクターブまで設定できます。
　音符は全音符から１６分音符まで設定できます。
　休符は全休符から１６分休符まで設定できます。
　テンポ（BPM） ２０BPM から設定できます。
またダイアログ内の「Monitor」にチェックを付けた状態で
鍵盤をクリックするとブザートーンをモニタリングすることが出来ます。

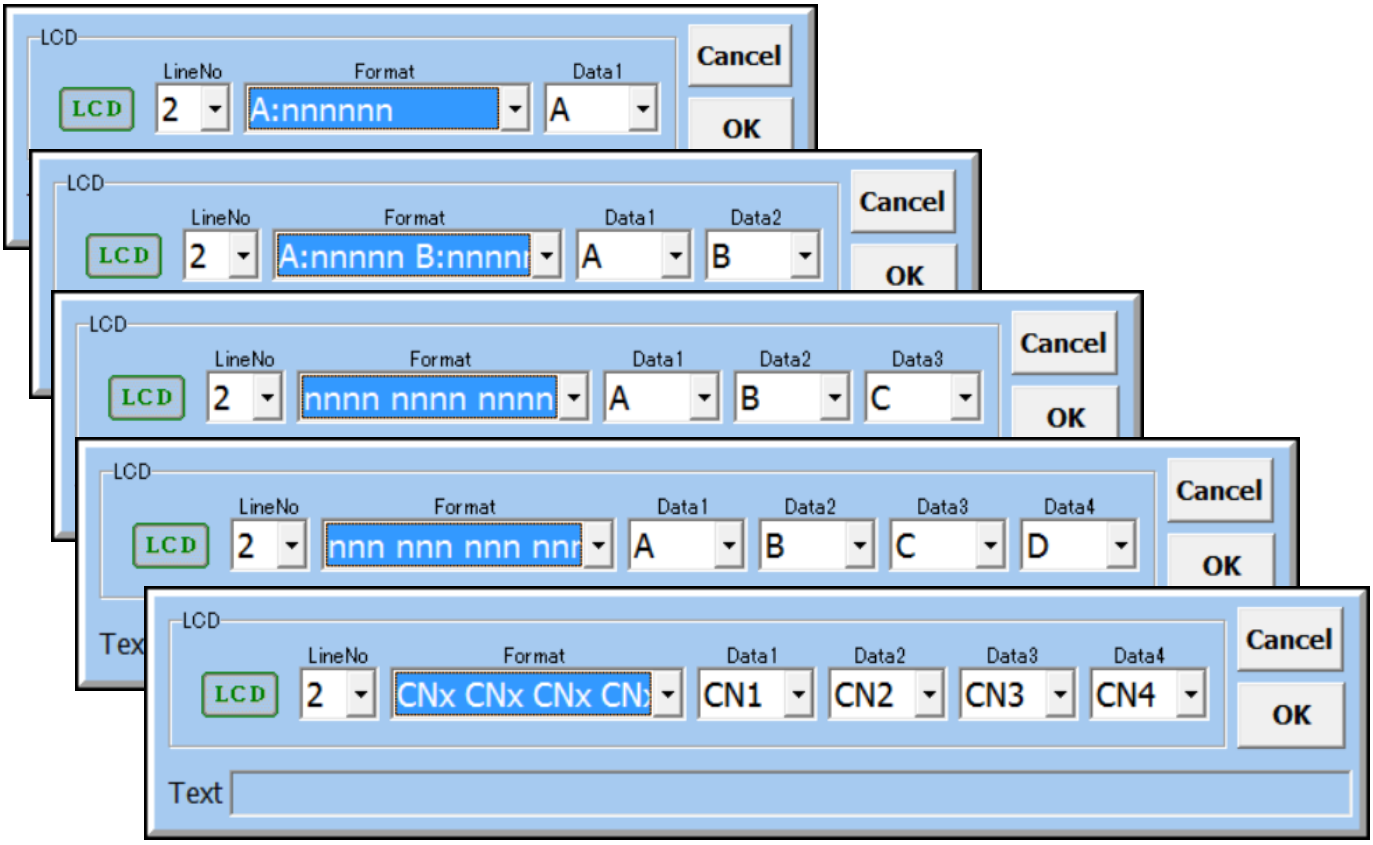
3-5. 16x2 LCD ボード(DSR1416)を使う





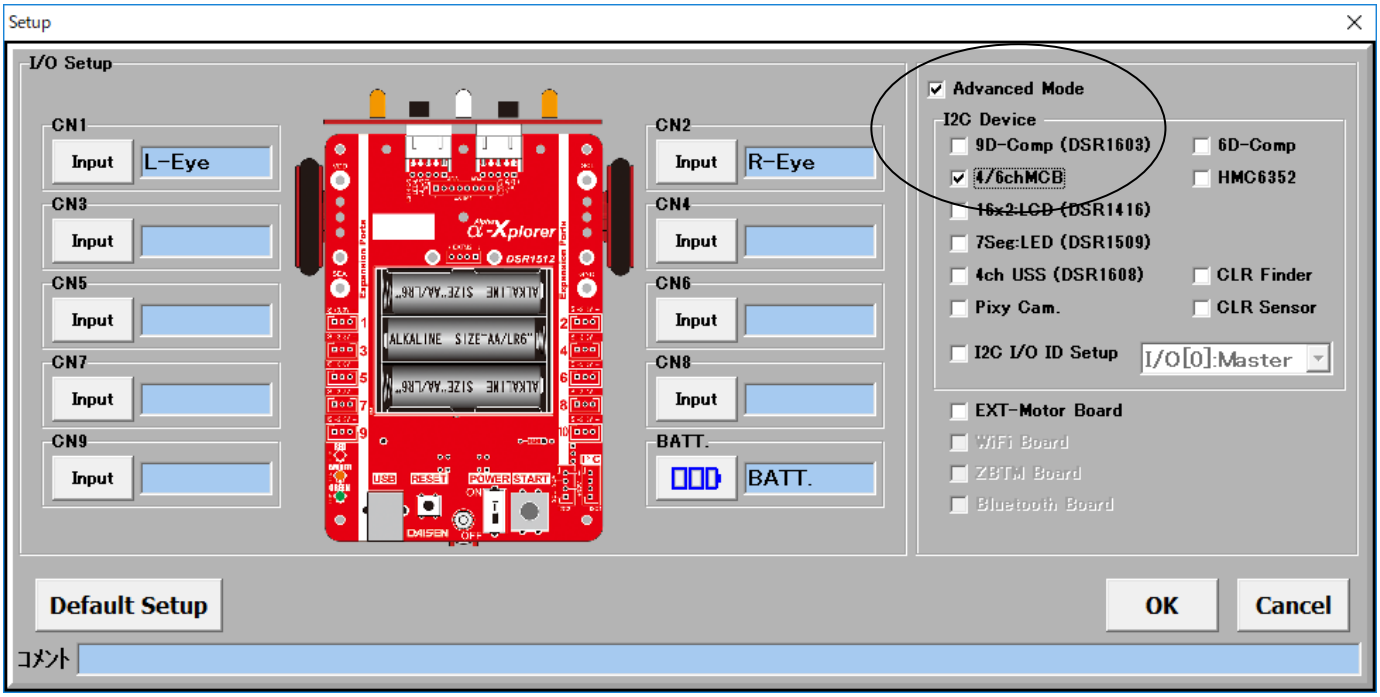
この LCD ボタンは Setup ダイアログ内の「Advanced Mode」設定から I2C Device リストの「16x2 : LCD」を選択すると拡張プログラムボタンリストに表示されます。
行番号(LineNo)に１か２を選択し、Format 欄が Text の場合、コメント欄に表示したい文字列を入力します。
半角 16 文字までの英数字と記号が入力できます。
但し、ダブルコーテーション「”」の記号は入力できません。(自動的に付加されます)

その他の表示フォーマットを選択すると、変数の値やＣＮの値を表示できます。



3-6. 4ch/6ch モータボードを使う

I/O Setup 画面を表示させて「Advanced Mode」にチェックを付けると「I2C Device」が表示されます。その中に「4ch/6ch MCB」にチェックを付けて「ＯＫ」ボタンをクリックします。

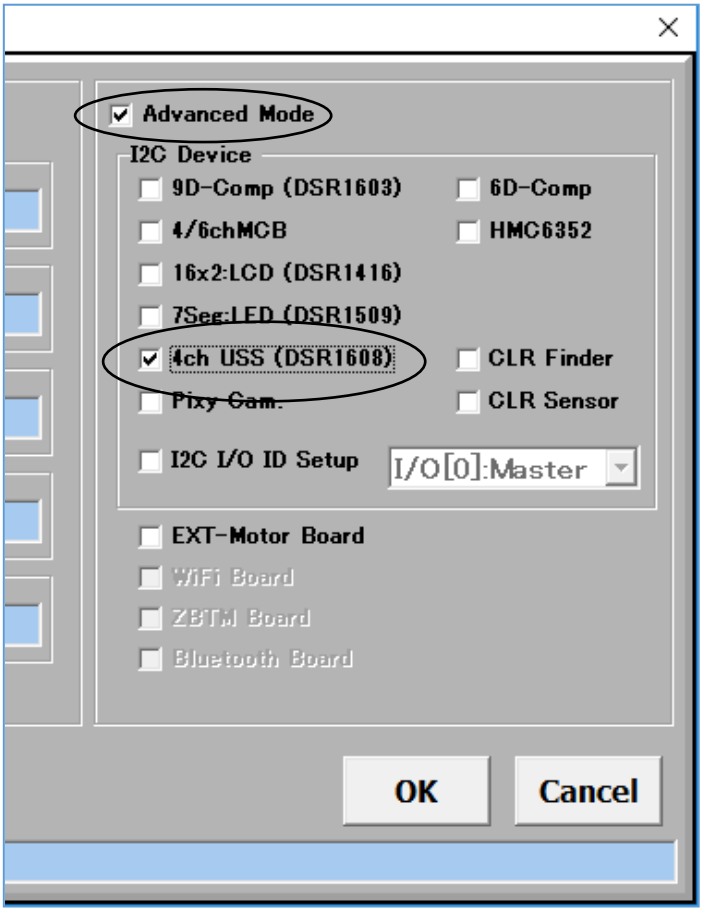


プログラムボタンリストに が追加表示されます。

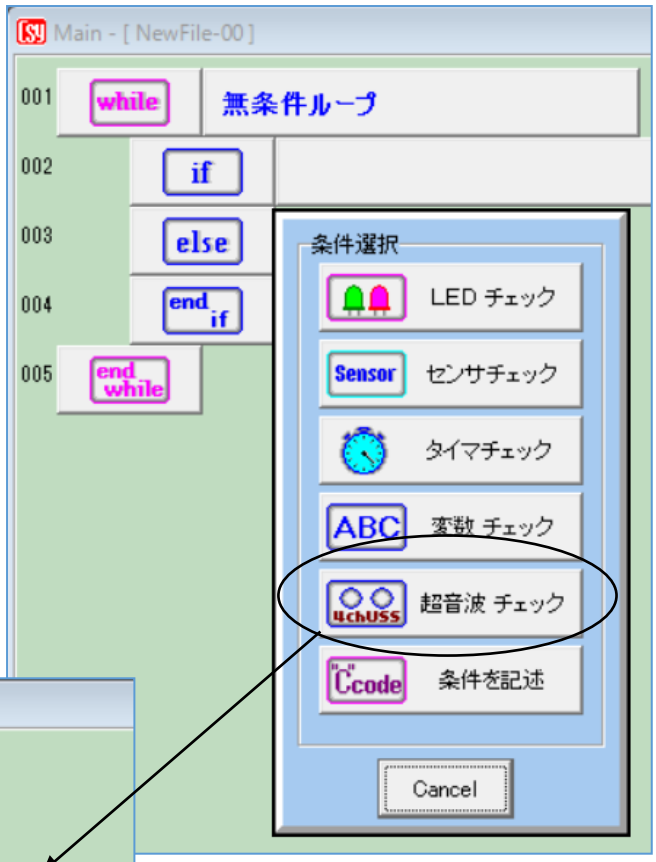
モータ速度を０%から１００%までの範囲で設定します。
回転方向は「Rev/Brk」にチェック付けます。速度が０%の場合はブレーキとなります。
使用しないまたは以前の設定を変更しない場合は「Modify」のチェックを外します。

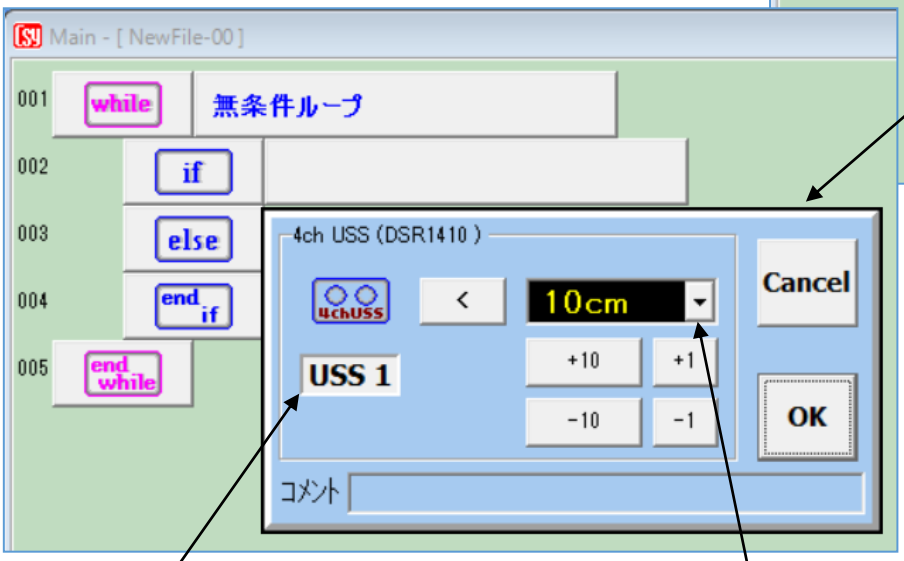
3-7. 超音波距離センサ(4chUSS:DSR1608)を使う

I/O Setup 画面を表示させて「Advanced Mode」にチェックを付けると「I2C Device」が表示されます。その中に「4ch Uss」にチェックを付けて「ＯＫ」ボタンをクリックします。



while と if の条件選択に超音波チェック「4chUSS」のボタンが追加されます。

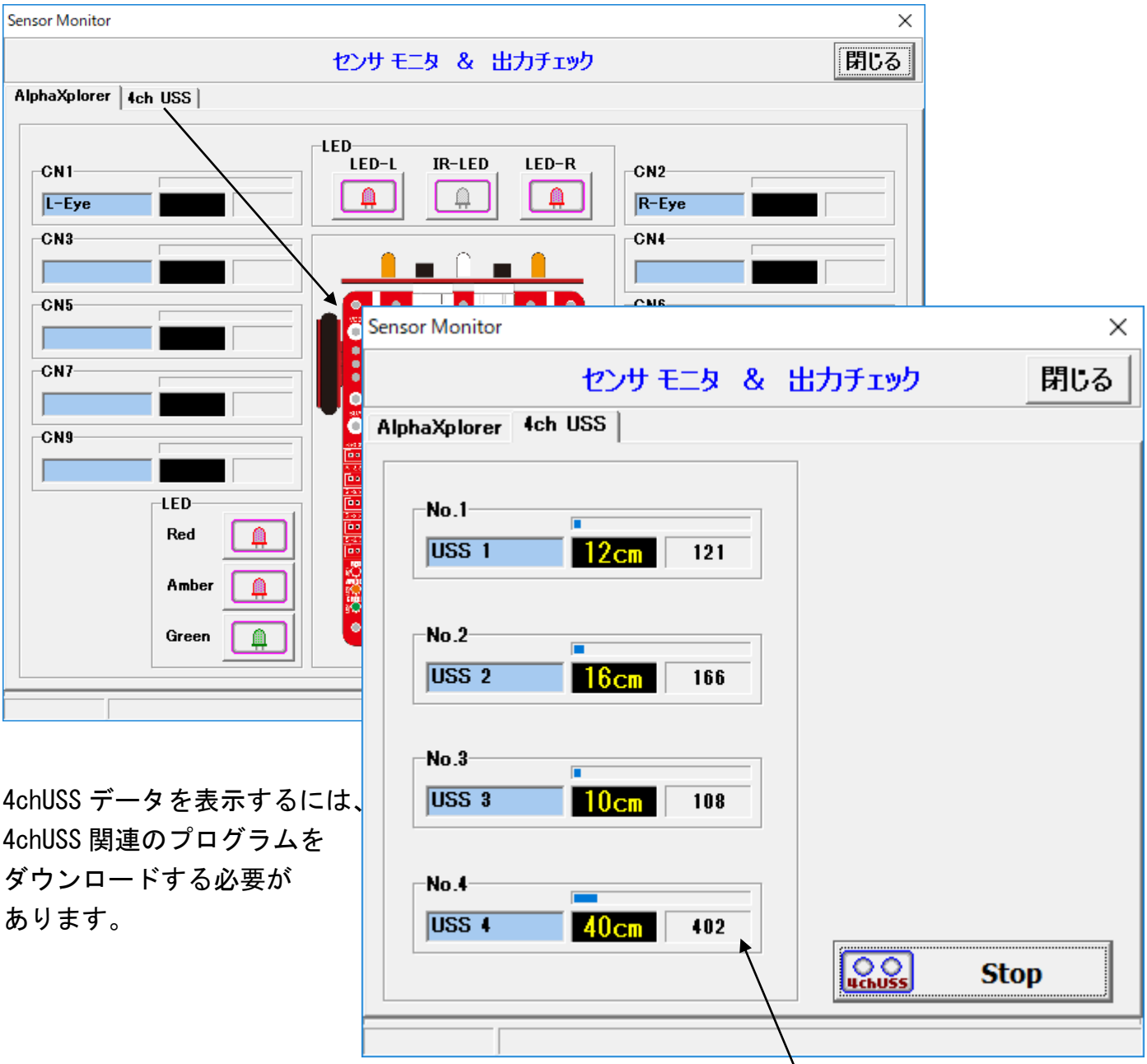




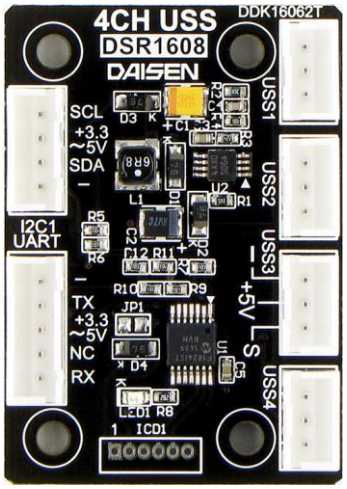
ここクリックするとＵＳＳ１からＵＳＳ４の選択が出来ます。

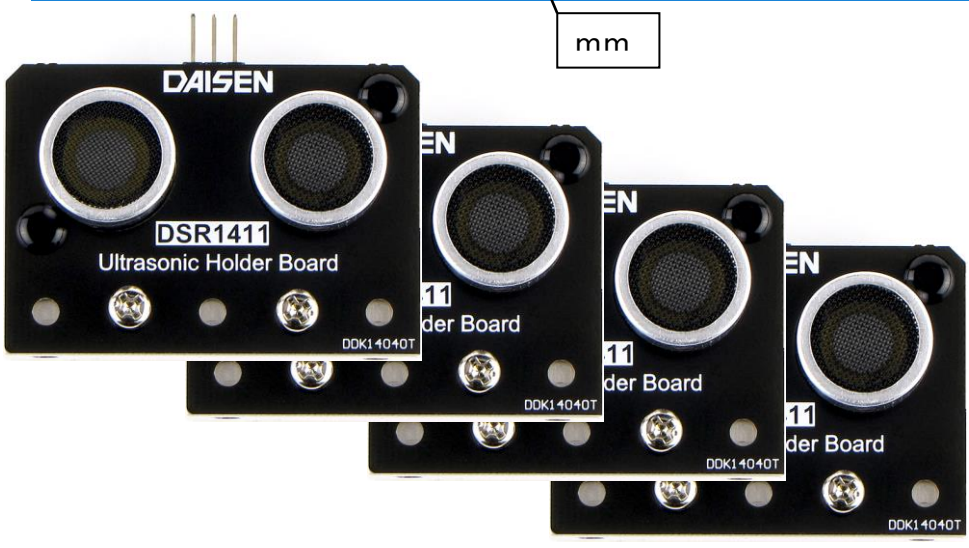
プルダウンメニューで変数指定も行えます。
変数へは単位をmmで指定します。

I/O Setup 画面で「4chUSS」を設定すると、センサモニタ画面に「4chUSS」タブが追加されます。



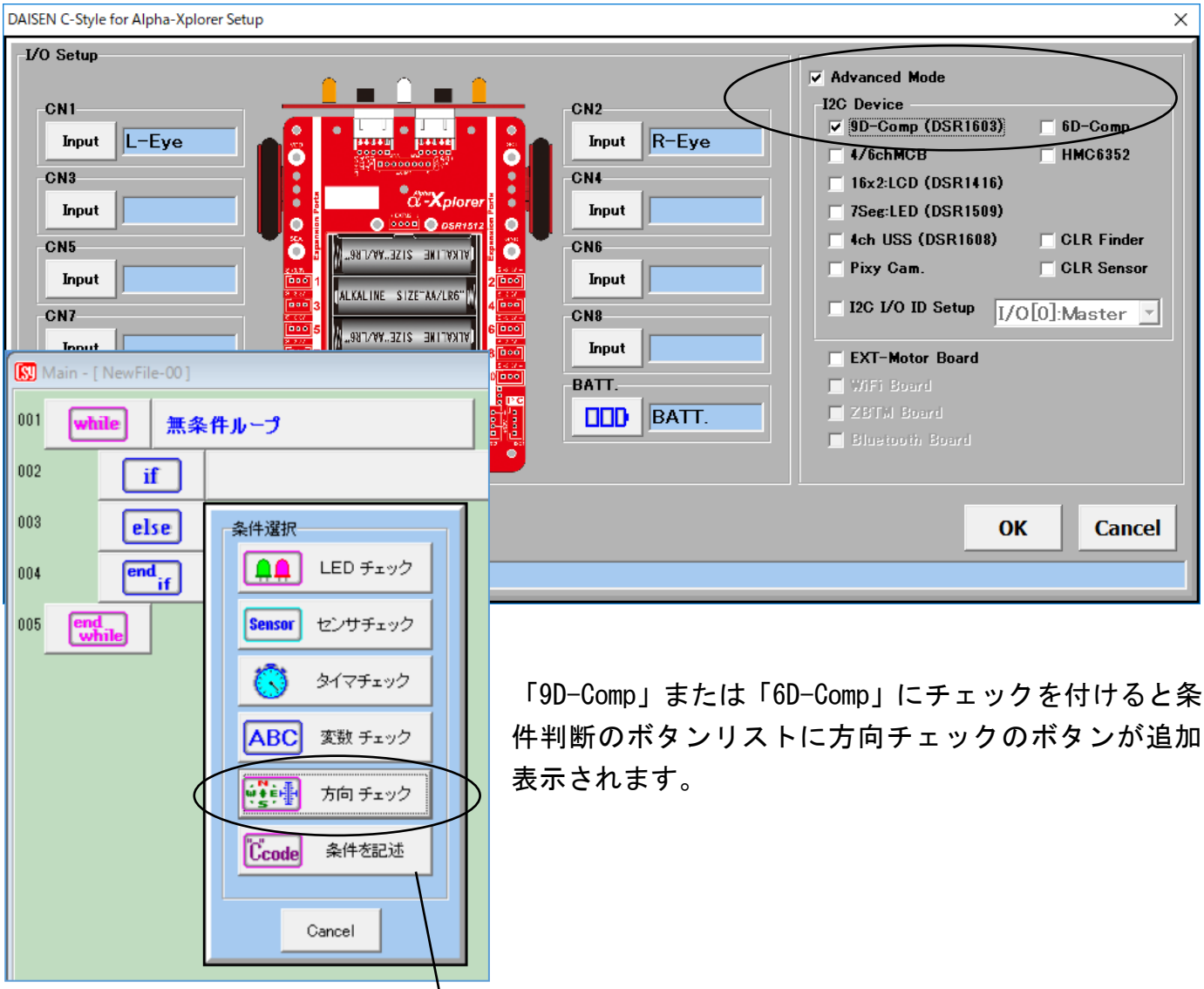
4chUSS データを表示するには、
4chUSS 関連のプログラムを
ダウンロードする必要が
あります。





3-8. 多機能電子コンパス（9D/6D-Compass：DSR1603/DSR1401）を使う

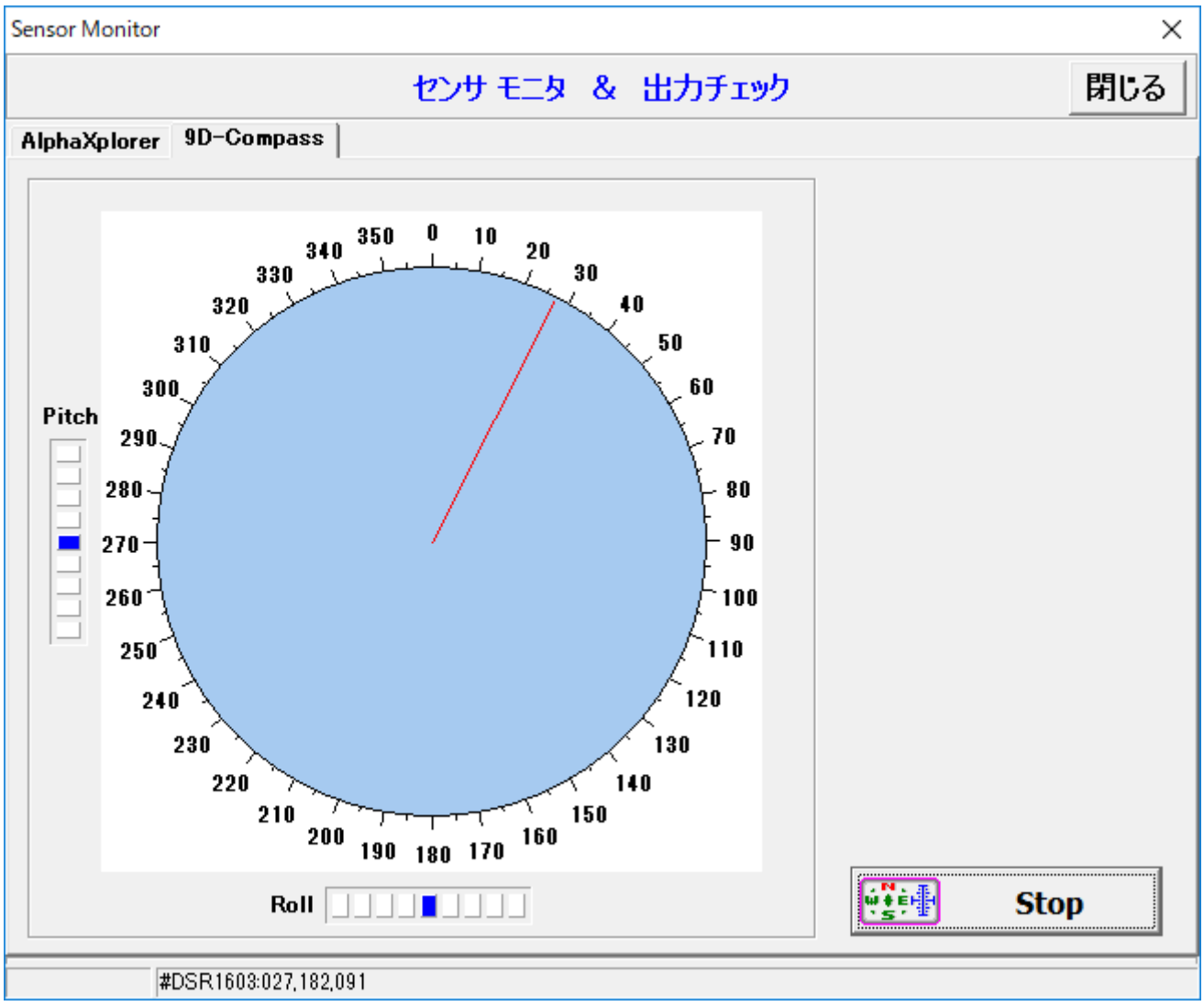
多機能電子コンパスを使用する場合は、「9D-Comp」または「6D-Comp」にチェックを付けます。



「9D-Comp」または「6D-Comp」にチェックを付けると条件判断のボタンリストに方向チェックのボタンが追加表示されます。

方向チェックボタンを選択しますと、Dir(地磁気方向)、Pitch(前後の傾き)、Roll(左右の傾き)
と判定角度、誤差範囲の設定が行えます。

設定画面の「9D-Comp」または「6D-Comp」にチェックの付いたプログラムをダウンロードすると、センサモニタ画面に追加された「9D-Compas」または「6D-Compass」のタブをクリックします。

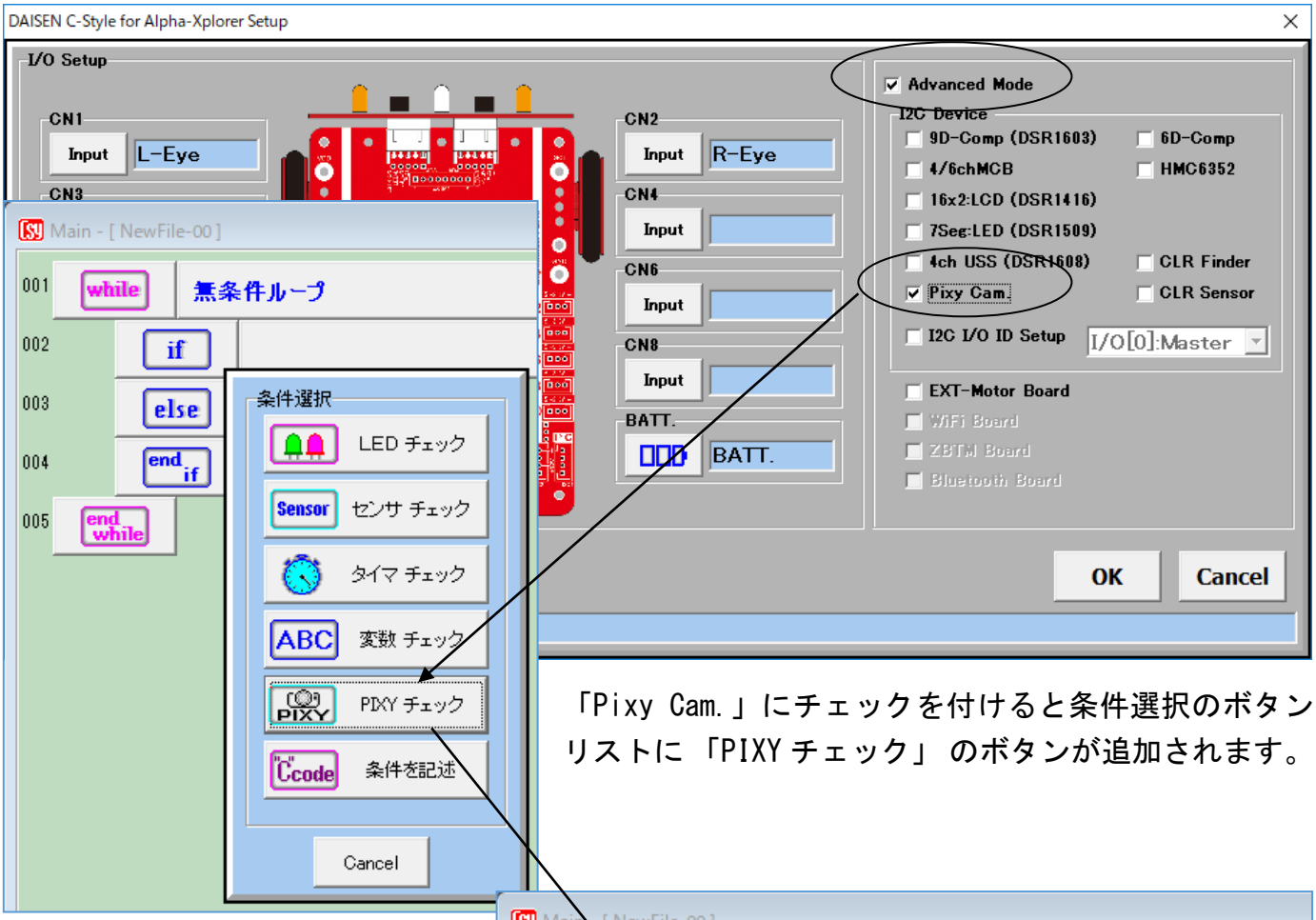


「モニタ開始」ボタンをクリックすると現在の方向を示す針が表示されます。電子コンパスを傾けると Pitch や Roll の青色(水平位置)インジケータが緑色に変化し移動します。

電源投入直後は正確な北の方角を示しません。
基板を数回８の字旋回することで自己補正機能が働き徐々に正確な北の方角を示す様になります。

3-9. Pixy Cam.を使う

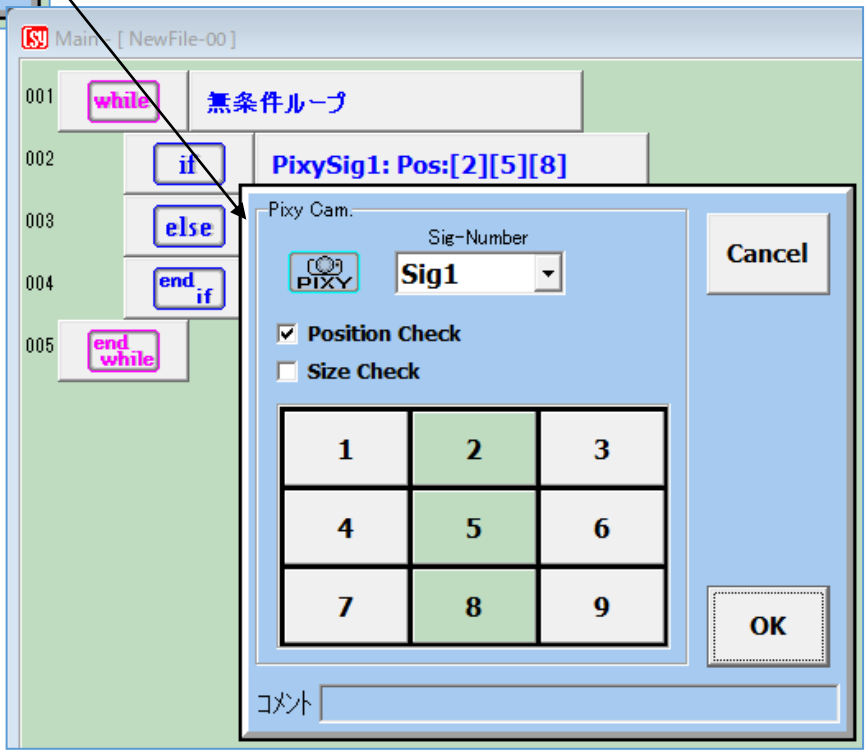
I/O Setup 画面の「Advanced Mode」-「Pixy Cam.」にチェックを付けます。



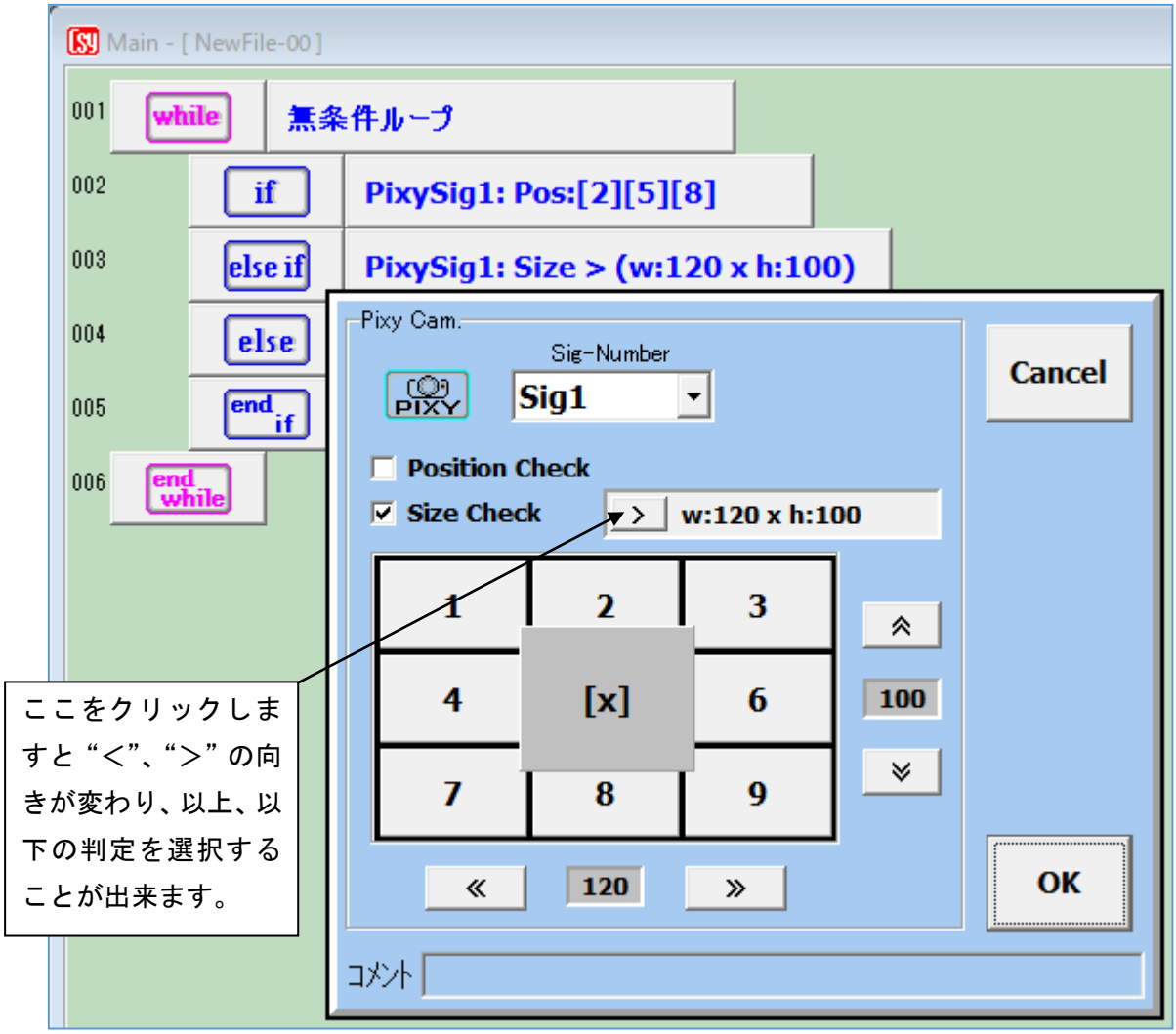
「Pixy Cam.」にチェックを付けると条件選択のボタンリストに「PIXY チェック」のボタンが追加されます。

PIXY チェック画面では、
Sig1～7 の学習色番号
Position１～９の座標位置
Size は
ｈ：１０～１９０
ｗ：１０～３１０のピクセル値
をそれぞれ指定して判定条件とします。

Pixy Cam が認識する最小サイズは PixyMon のツールで設定します。（10～20 程度）

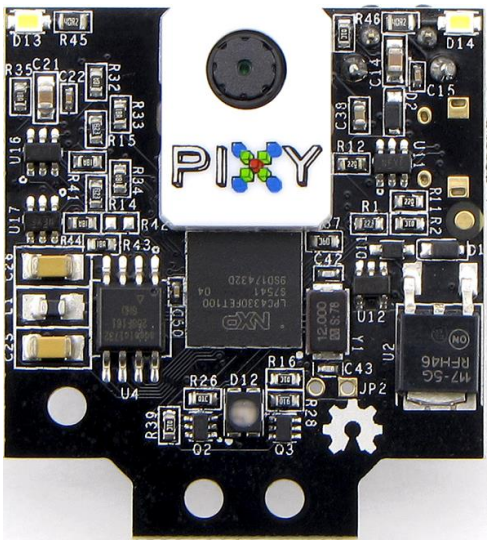


Pixy Cam.のサイズ指定



サイズはｈ：１０～１９０、ｗ：１０～３１０の範囲で設定できます。
画面中央でサイズ表示をしていますが、「Position Check」 の指定が無い場合は、指定サイズが画面の何処に現れても判定条件となります。

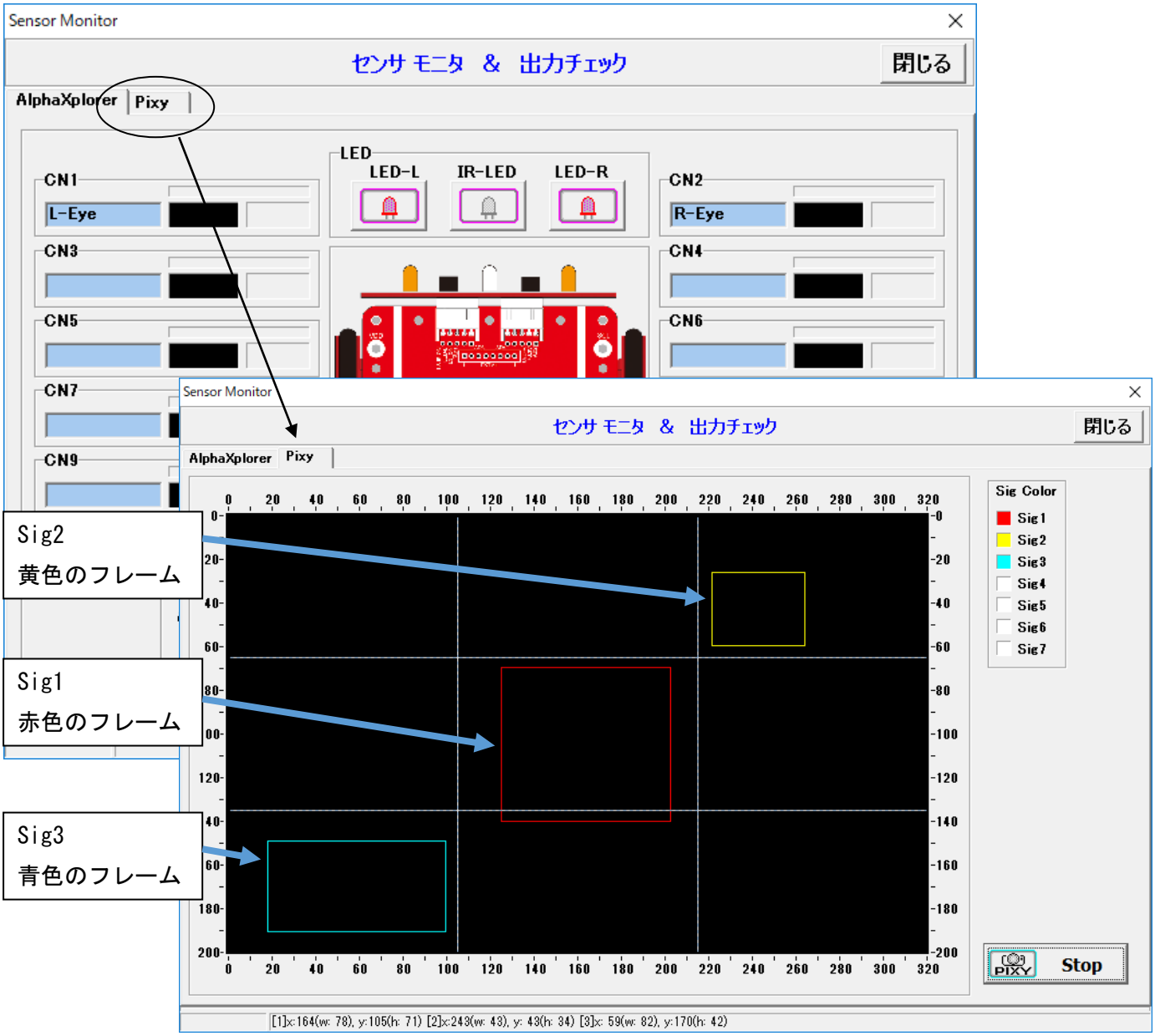
「Position Check」と「Size Check」の両方にチェックがついている場合、画像の中心座標が指定された 1～9(複数指定可能)の位置で指定サイズ以上または以下が判定条件となります。



C-Style でプログラムする前に Pixy サイトより調整用のアプリ“PixyMon V2”をダウンロードして初期設定や該当するカラーの学習など調整する必要があります。

Pixy Cam.のセンサモニタ

I/O Setup 画面で Pixy Cam の使用を設定したプログラムをダウンロードすることでセンサモニタ画面に「Pixy」のタブが追加表示され、それをクリックしますと Pixy Monitor 画面になります

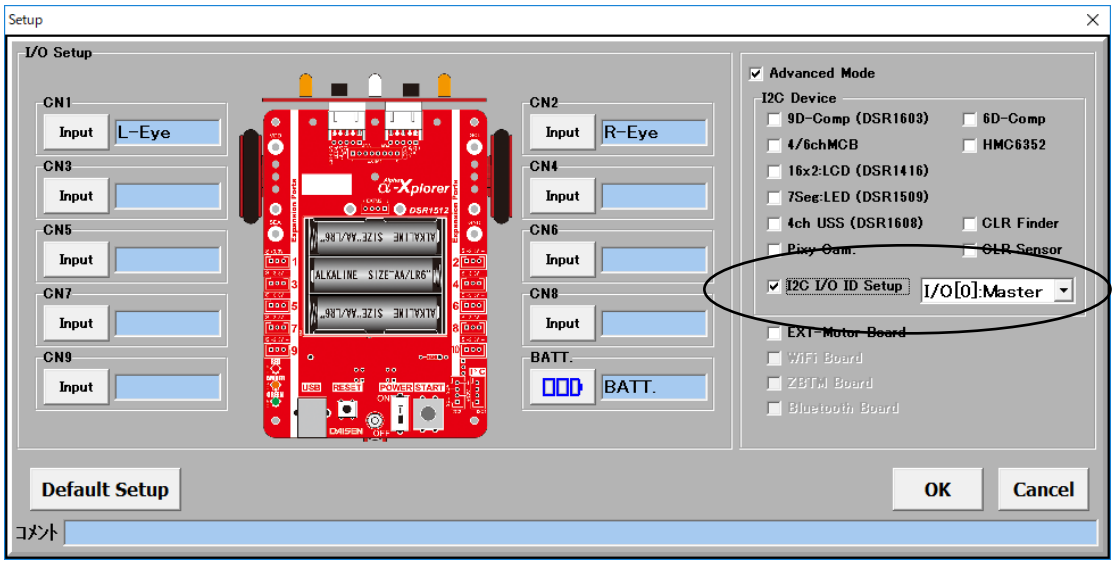


画面右下側の「PIXY Star」ボタンをクリックすると「PIXY Stop」の表示に変わりモニタ動作を開始します。

Sig1 が赤色のフレーム、Sig2 が黄色のフレーム、Sig3 が青色のフレーム、Sig4～7 は白色のフレームで表示されます。Sig Color はダブルクリックすることで任意の色に変更できます。

3-10. 複数台の Alpha-Xplorer を接続する

「I2C I/O ID Setup」にチェックを付けると最大８台までの Alpha-Xplorer 接続して I/O センサボードして使用できるようになります。

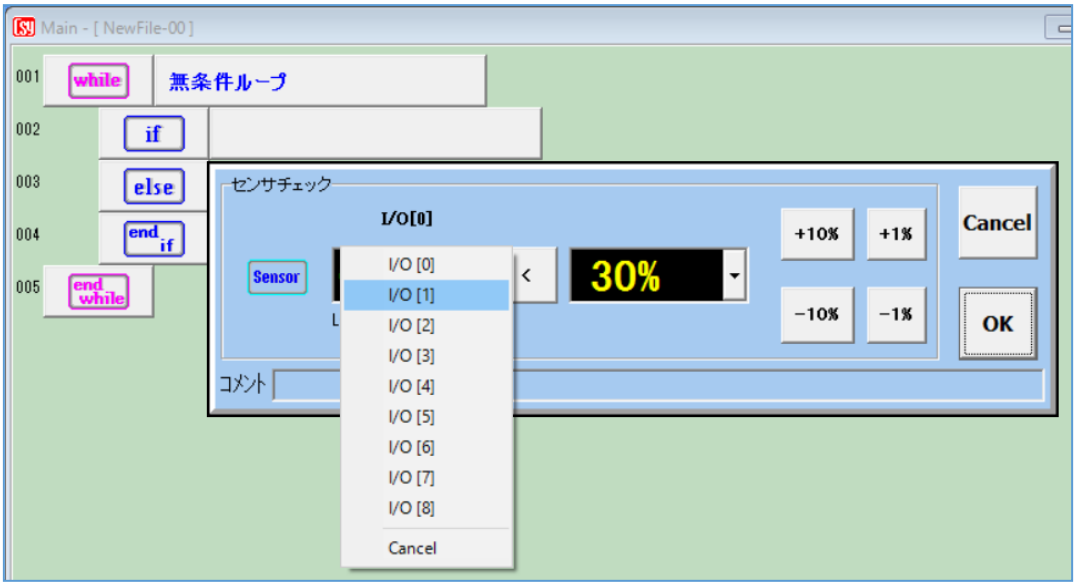


この機能は、通常は 1 台当たり最大 9 個のセンサ(CN1～CN9)を使うことができますが、I2C という通信仕様のコネクタ(4pin)を使って別の Alpha-Xplorer と接続してセンサを増やすことが出来ます。

センサ情報を一括管理する親の Alpha-Xplorer のことを Master(I/O[0])と呼び、センサ情報を親へ引き渡す Alpha-Xplorer のことを SubI/O(I/O[1]～I/O[8])と呼びます。

I/O Setup 画面では、Master の設定か、SubI/O の何番の ID：1～8 に設定するかを決めてからプログラム編集を行います。

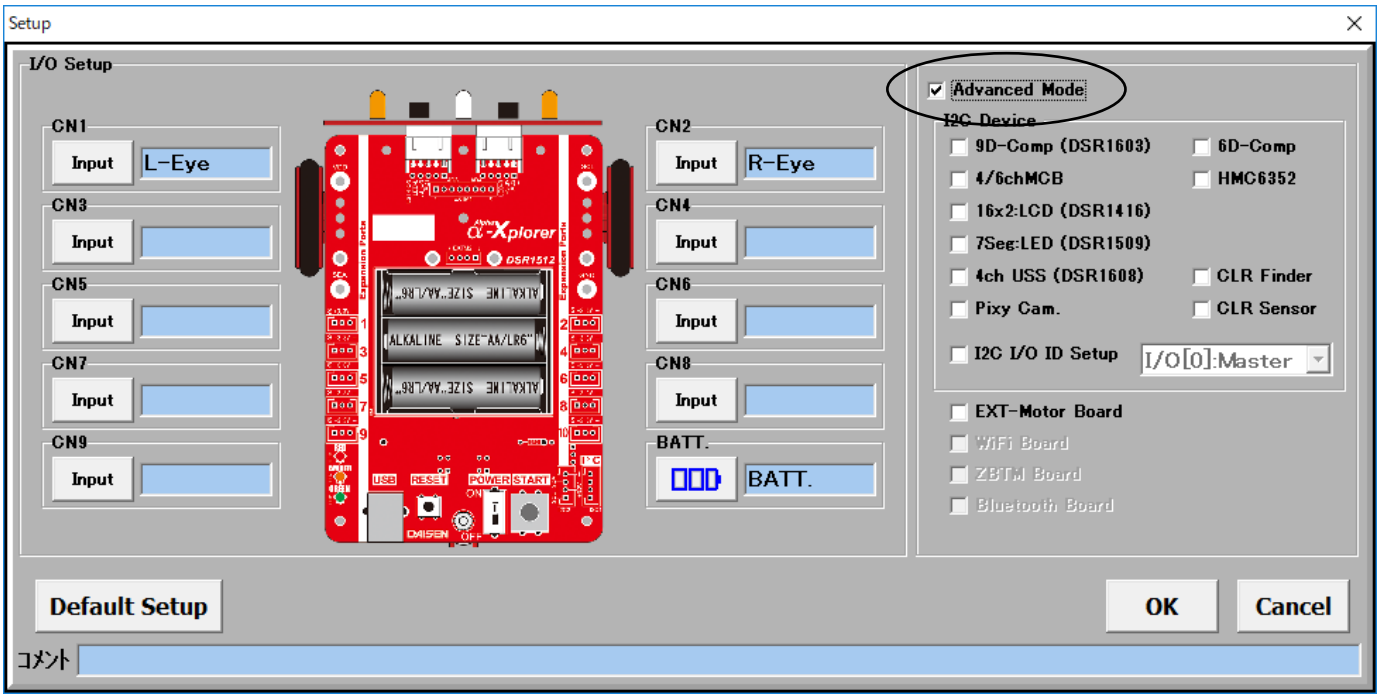
通常、SubI/O のプログラムは、この I/O Setup 画面で設定した ID(1～8)だけでビルドしてダウンロードすれば、電源を入れるだけ（スタートボタンは押さなくても良い）で、Master 側のプログラムによって SubI/O のセンサ情報を取得して条件判定することが出来ます。



４． サブプログラムの説明

4-1. サブプログラムボタンの表示

I/O Setup（入出力設定）のダイアログ内にある「Advanced Mode」にチェックを付けて「ＯＫ」ボタンでダイアログを閉じます。



拡張機能の設定をするとサブプログラムボタンが使用出来るようになります。

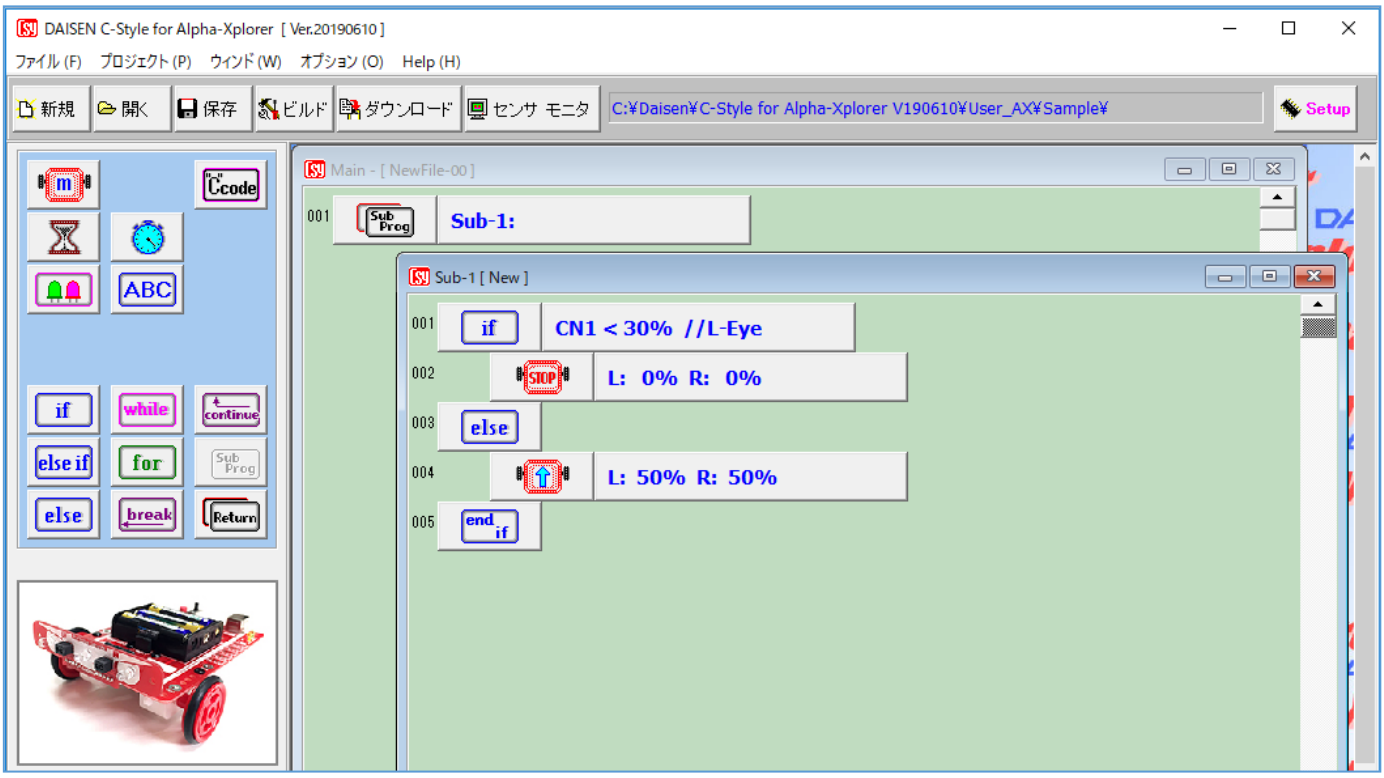
同じ処理を何回も再利用したい場合や、プログラムを見やすくする為に、一定の処理プログラムを一つのプログラムボタンにまとめることが出来ます。（サブルーチンと言います）

4-2. サブプログラムの編集

サブプログラムは３０個まで作成することが出来ます。
但し、サブプログラムの中にサブプログラムボタンを置くことは C-Style では出来ません。

新規にサブプログラムを作成する場合はボタンを置いてから「ＯＫ」をクリックします。
既に作成されたサブプログラムを指定したい場合は、「サブプログラム」ボタンをクリックするとファイルから選択出来ます。

サブプログラムはメインプログラムがあるフォルダーに移動しておいて下さい。



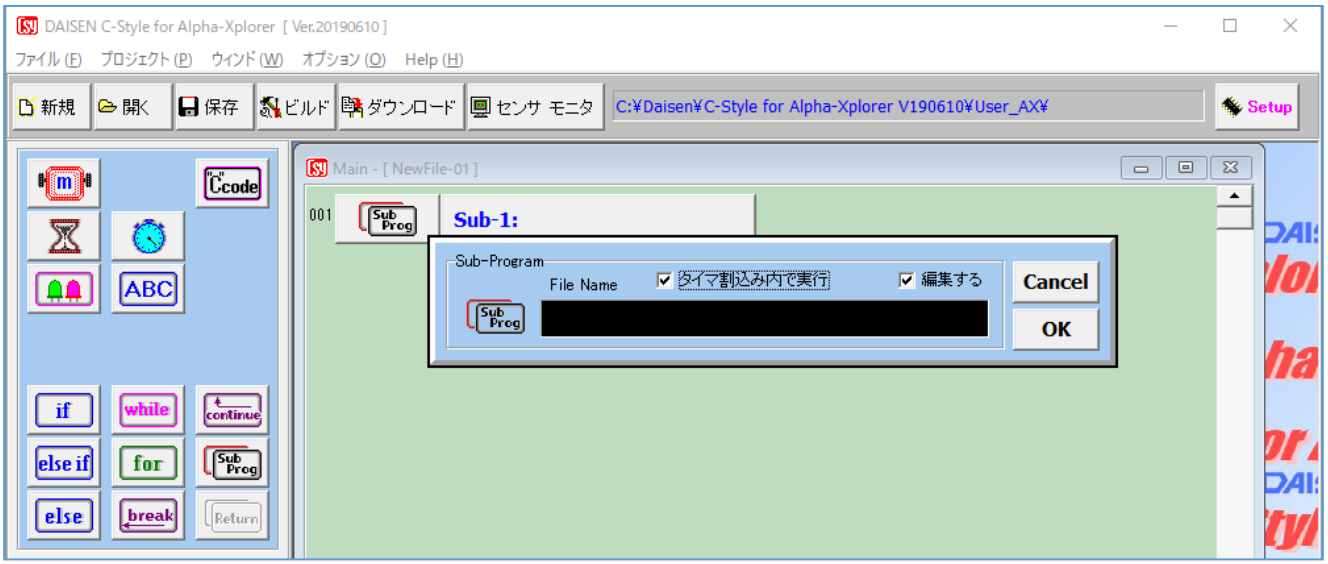
サブプログラムの編集フォームが表示されたら、メインフォーム同様にプログラムボタンリストからプログラムボタンを選択して、サブプログラムを編集して行きます。

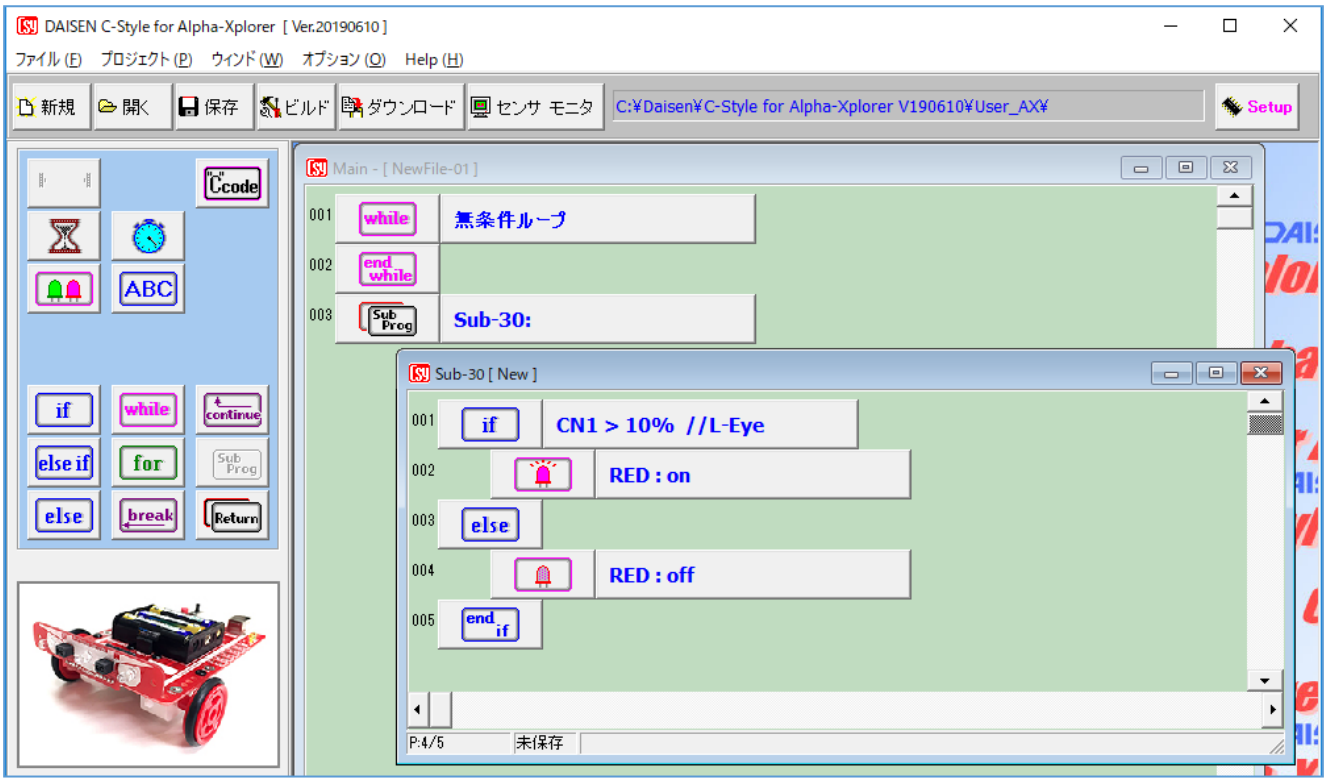
編集されたサブプログラムを保存するには、サブプログラムフォームをアクティブな状態（タイトルバーをマウスでクリックして濃いブルーの表示になった状態）にしてから、「保存」ボタンをクリックします。
サブプログラムファイルは、メインプログラムファイルと同じフォルダー内に保存して下さい。

保存したファイル名が、メインフォームのサブプログラムボタンのコメント表示となります。

4-3. タイマ割込み内で実行するサブプログラム

通常のサブプログラム作成手順で、サブプログラムボタンを置いた時に表示される「File Name」のラベルをマウスでダブルクリックしますと「タイマ割込み内で実行」のチェックボックスが現れますのでチェックを付けます。一度チェックを付けたサブプログラムは ”Sub-30” と表記が変わり特別なサブプログラムとなります。





この例では、サブプログラムが無条件ループの外側に置かれているので、通常実行されないのですが、実際はタイマ割込み内で常に実行されているので、CN2 のラインセンサが１０%よりおおきくになると RED1 の LED が点灯します。

この特別なサブプログラムは、変数演算、CN1～CN10 に接続されたアナログセンサの判定、LED の点灯制御のみが使用可能です。（モータ、PING、サーボモータ、I2C 等の制御は出来ません。）

メモ

株式会社ダイセン電子工業
DAISEN

〒556-0005 大阪市浪速区日本橋４丁目 9-24
TEL 06-6631-5553　(FAX 06-6631-6886)
URL http://www.daisendenshi.com
Email ddk@daisendenshi.com