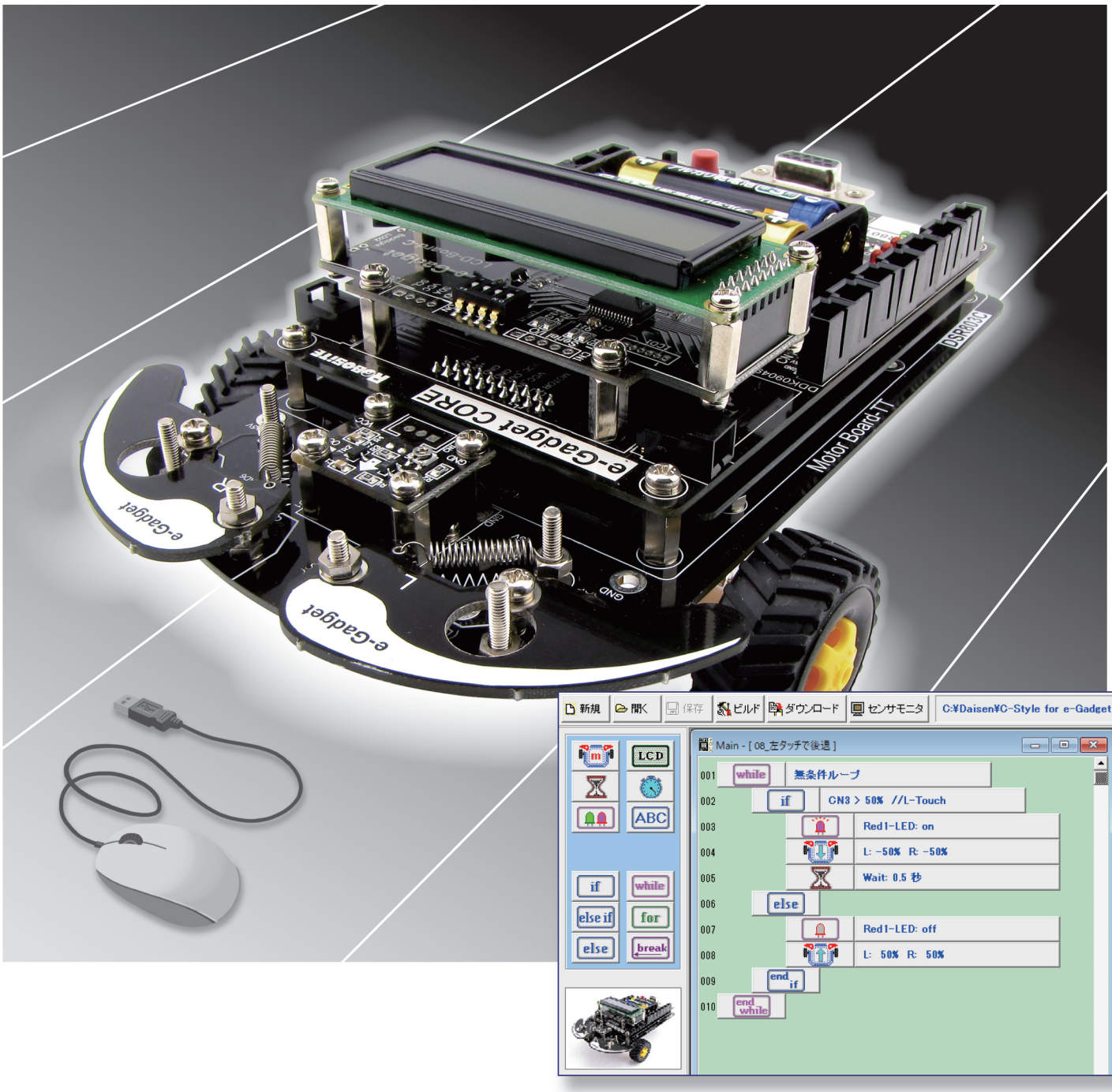
自立型ロボット製作キット
Autonomous Robot D.I.Y. Kit

NIPPONBASHI ROBOT DREAM LAB
ROBOSITE

イー・ガジェット
e-Gadget

C-Style 操作編



DAISEN
株式会社ダイセン電子工業

Rev.190220

目　次

C-Style 操作編（本書）

1. プログラムボタンの説明 ---------- 2
   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. ＬＣＤ表示ボタン ---------- 6
   1-6. タイマスタートボタン ---------- 7
   1-7. 変数ボタン ---------- 8
   1-8. 条件分岐 ---------- 9
   1-9. 条件付き繰返し ---------- １０
   1-10. 回数指定の繰返し ---------- １１
   1-11. タイマチェック ---------- １２
   1-12. 変数チェック ---------- １３

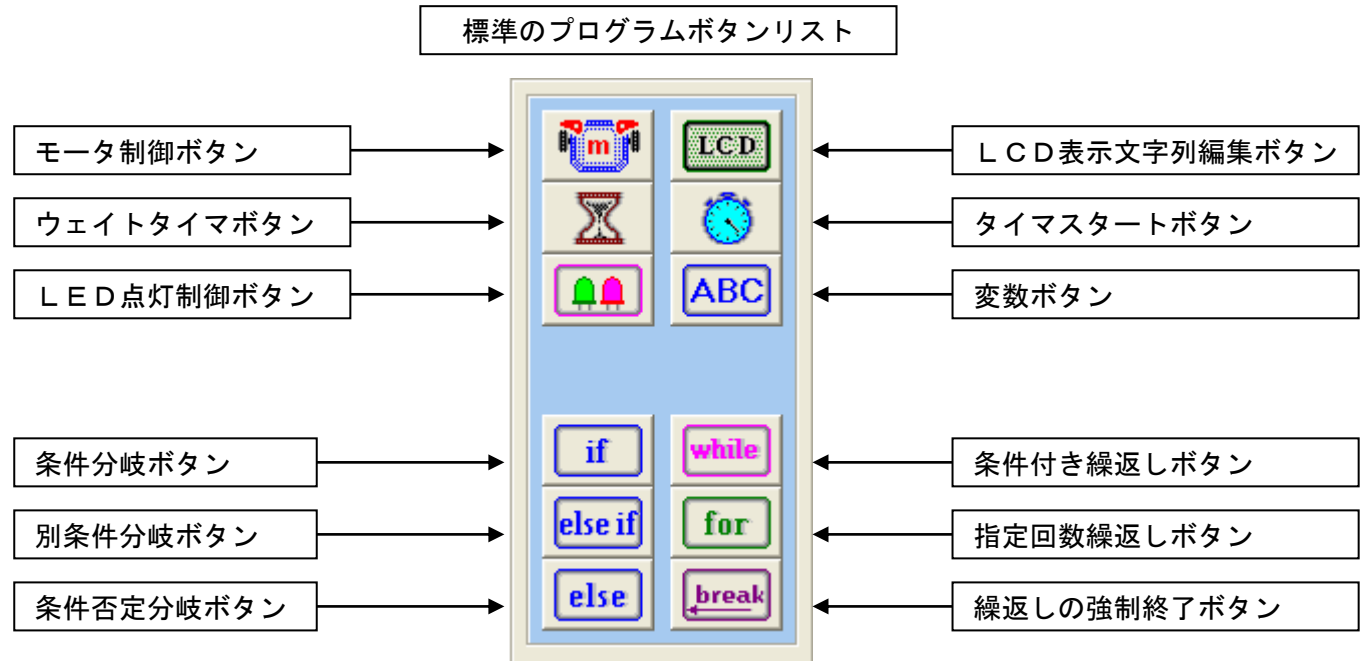
2. プログラムボタンの挿入、削除、コピー、貼付けの説明 ---------- １４
   2-1. ボタンの挿入 ---------- １４
   2-2. ボタンの削除 ---------- １５
   2-3. ボタンのコピーと貼付け ---------- １６

3. 入出力設定と拡張機能の設定 ---------- １７
   3-1. Setup ボタンを表示 ---------- １７
   3-2. I/O Setup（入出力設定）---------- １８
   3-3. 拡張機能の設定 ---------- １９
   3-4. 超音波距離センサを使う ---------- ２０
   3-5. サーボモータを使う ---------- ２２
   3-6. 6D/9D-Compass(多機能電子コンパス：DSR1401/1603)を使う --- ２５
   3-7. PixyCam を使う ---------- ２７
   3-8. 複数台の e-Gadget を接続する ---------- ３０
   3-9. ＥＧシリーズを使う ---------- ３１

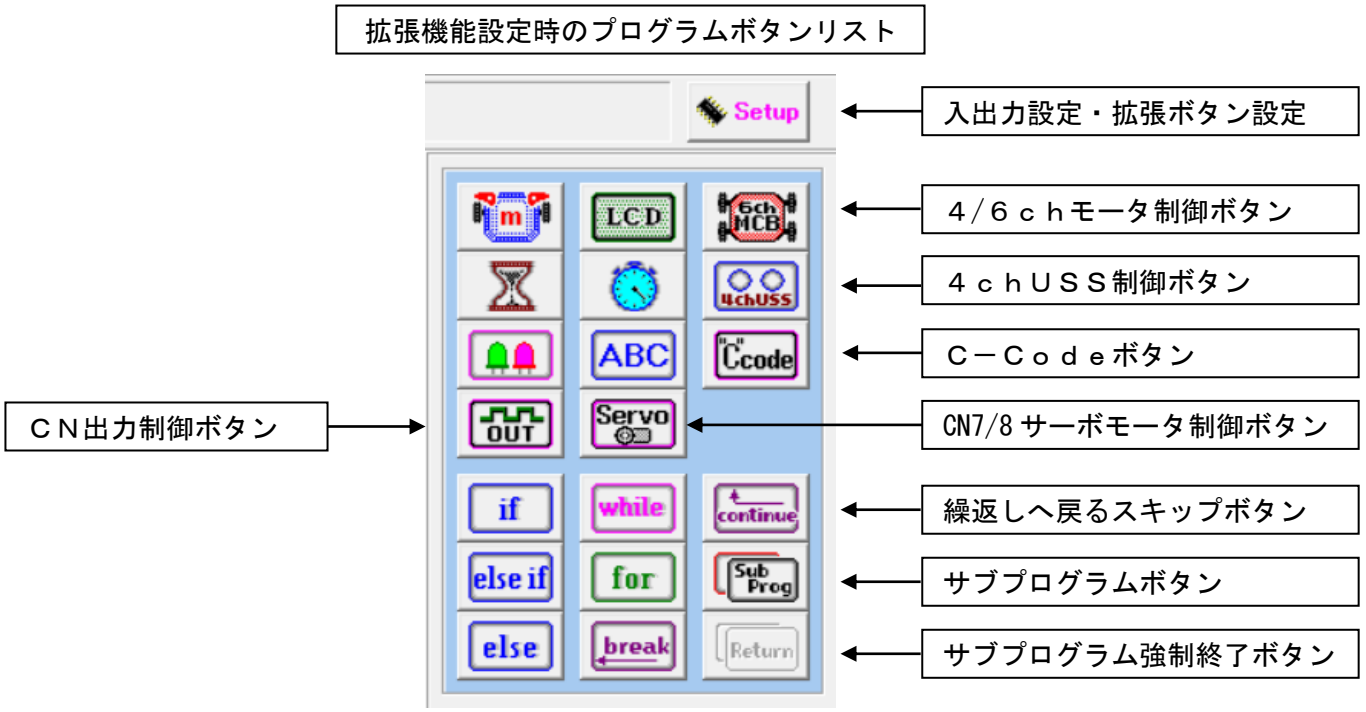
4. サブプログラムの説明 ---------- ３２
   4-1. サブプログラムボタンを置く ---------- ３２
   4-2. サブプログラムの編集 ---------- ３３
   4-3. タイマ割込み内で実行するサブプログラム ---------- ３４

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



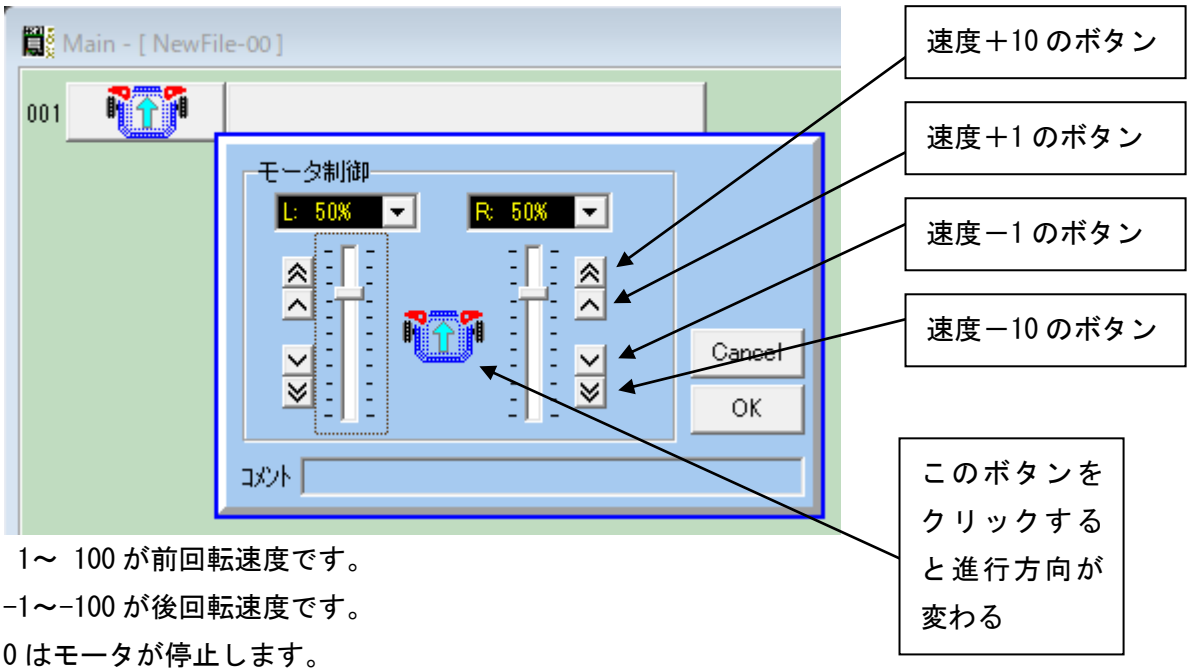
◆拡張機能を設定するとプログラムボタンリストも下記のように変わります。



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）



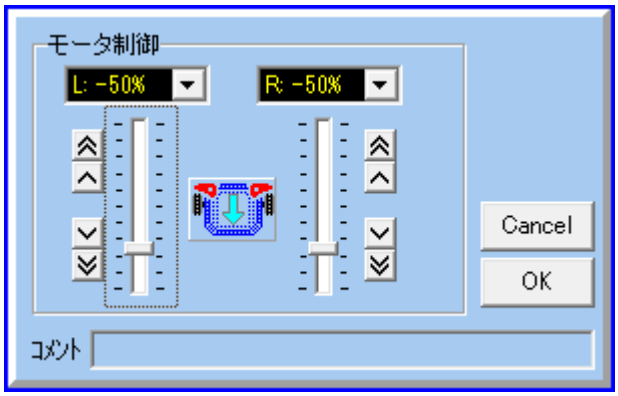
1～ 100 が前回転速度です。

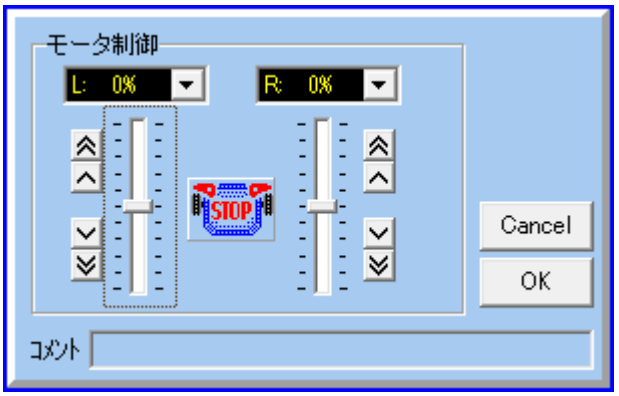
-1～-100 が後回転速度です。

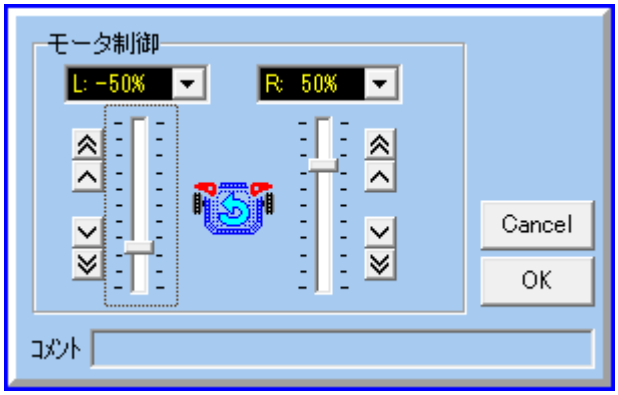
0 はモータが停止します。

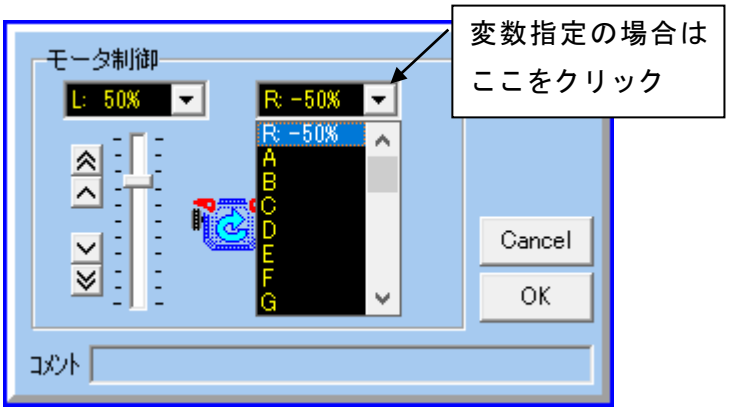
左右を+/-逆に設定するとロボットは回転します。

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。





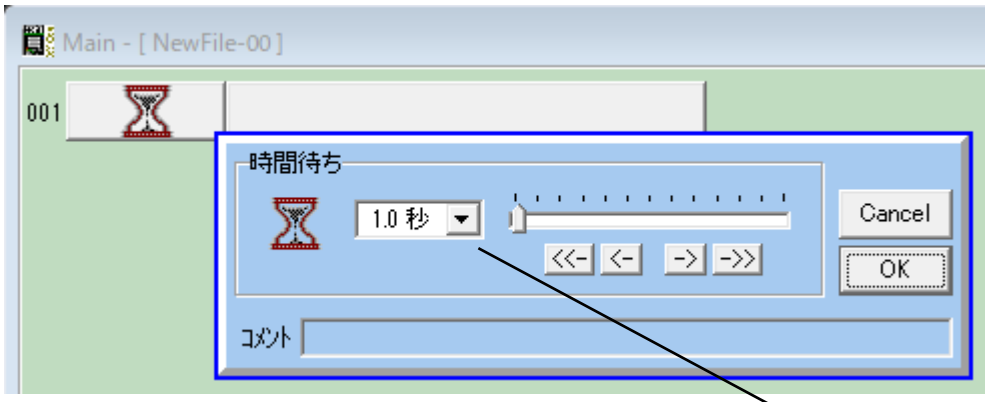




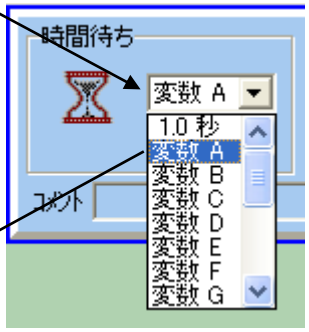
前方向に旋回または、後方向に旋回にするには、速度ボタンまたはスライダを操作して決めます。

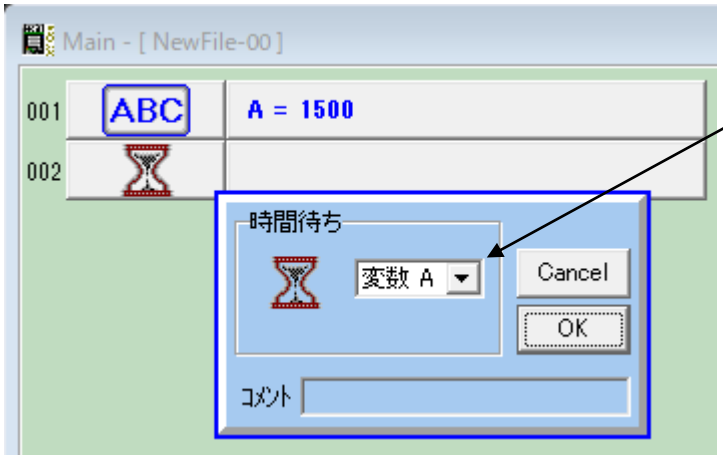
速度表示のプルダウンメニューを表示して変数 A～Z で指定することも出来ます。

1-3. ウェイトタイマボタン



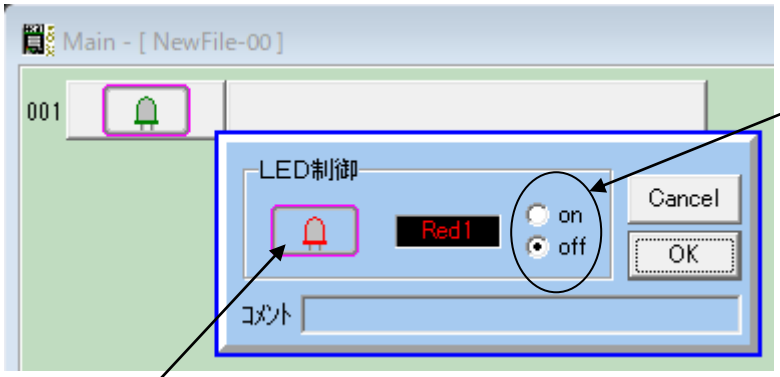
0.1 秒から 60.0 秒までの時間待ちを設定します。
待ち時間を変数に設定することも出来ます。
変数へ待ち時間を設定する場合は、1000 倍したミリ秒の単位で
設定します。
例えば 1.5 秒の待ち時間を変数 A に設定する場合は、A=1500 として
待ち時間表示をクリックして「変数 A」を選択します。





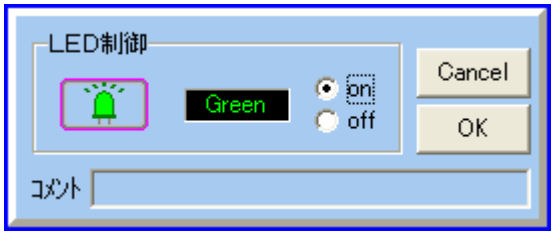
ウェイトタイマボタンは、簡単に待ち時間をプログラムすることができますが、待っている間は、他のプログラムボタンを置いて制御することが出来ません。待っている間他のプログラム制御したい場合は、タイマスタートボタンとタイムチェックを組み合わせて行います。

1-4. ＬＥＤ制御ボタン

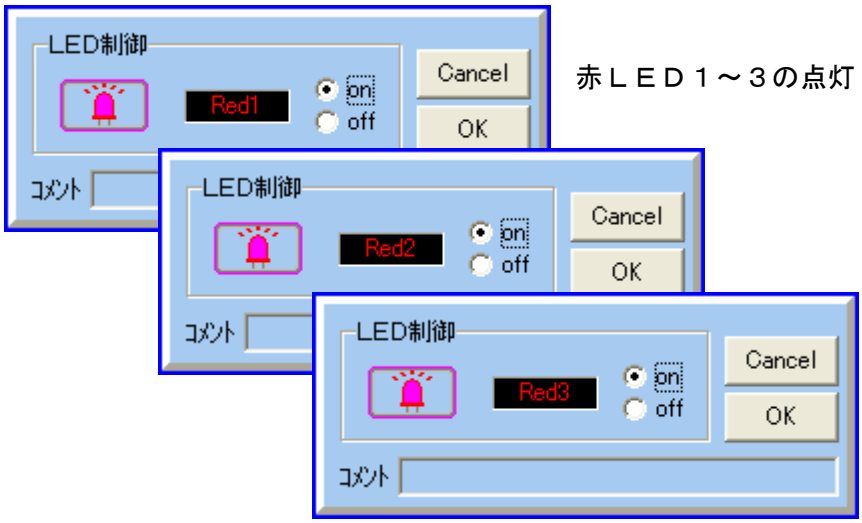


ここをクリックして点灯、消灯を選択します。

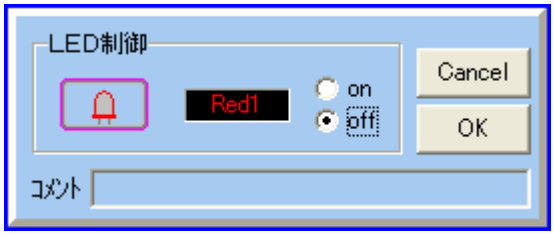
ここをクリックして赤と緑のＬＥＤを選択します。



緑ＬＥＤの点灯

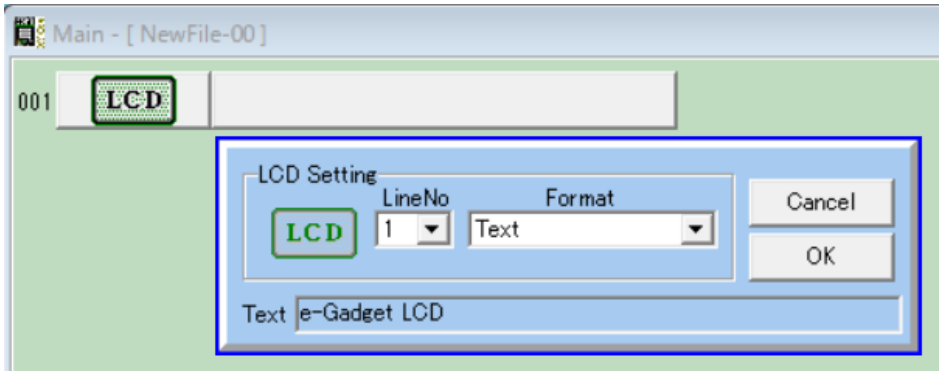


赤ＬＥＤ１～３の点灯



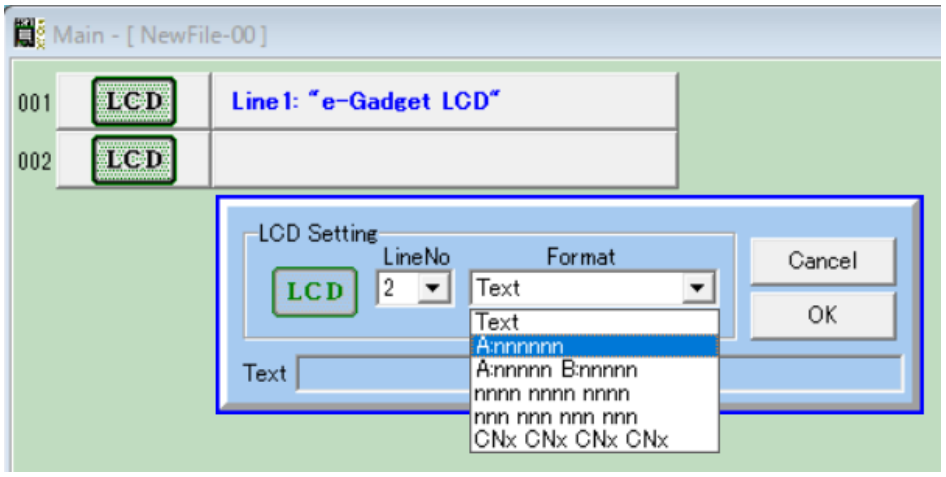
赤ＬＥＤ１の消灯

1-5. ＬＣＤ表示文字列の編集ボタン



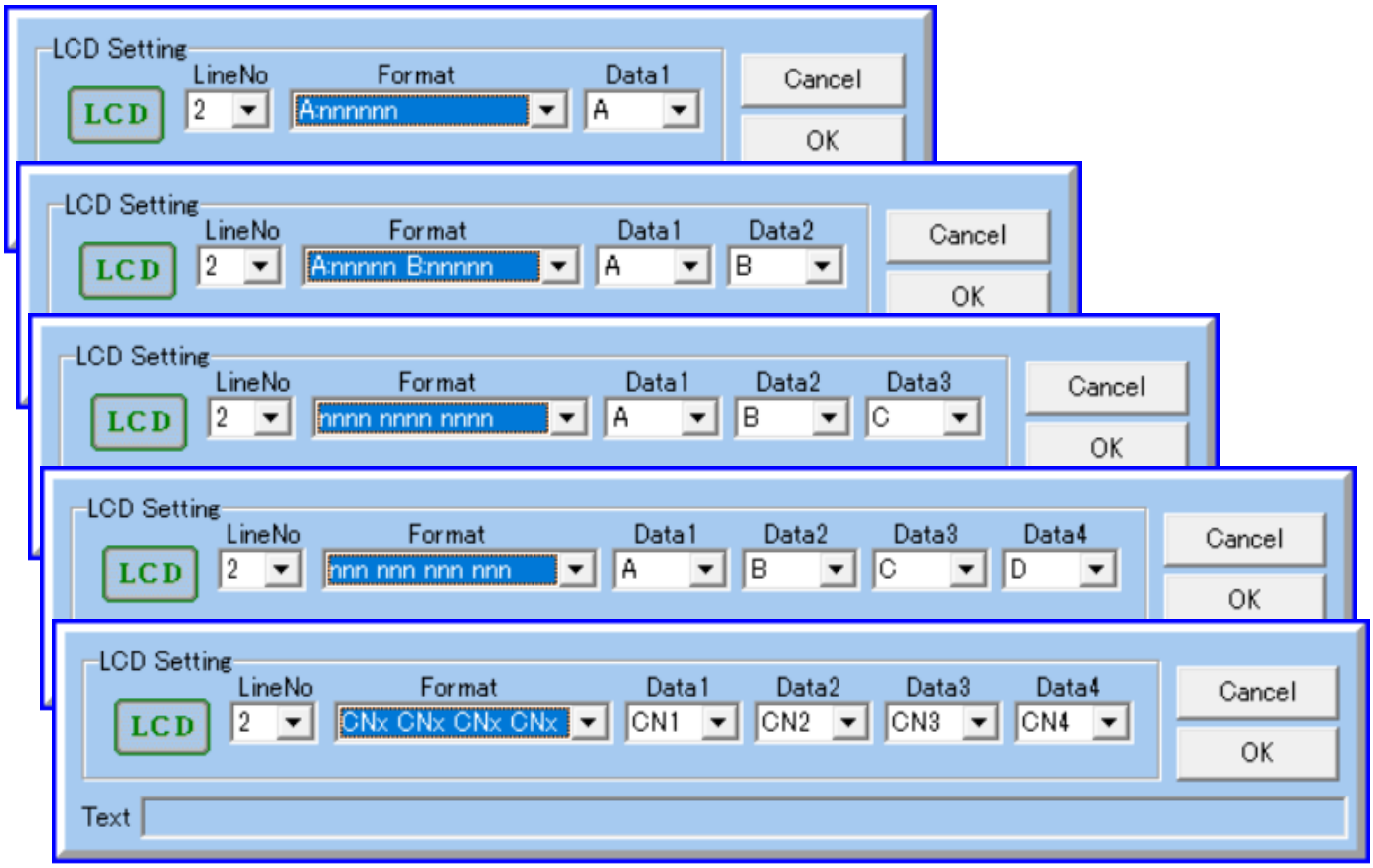
2 行 x 16 桁のＬＣＤへの文字列は行番号を指定して編集することが出来ます。

行番号（LineNo）は１、または２を選択します。

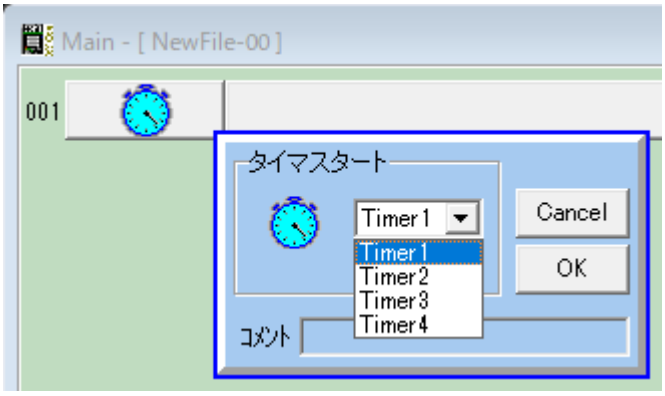


Format 欄の Text では、コメント欄に表示したい文字列を入力します。
半角 16 文字までの英数字と記号が入力できます。

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、変数値やＣＮのセンサ値を表示できます。



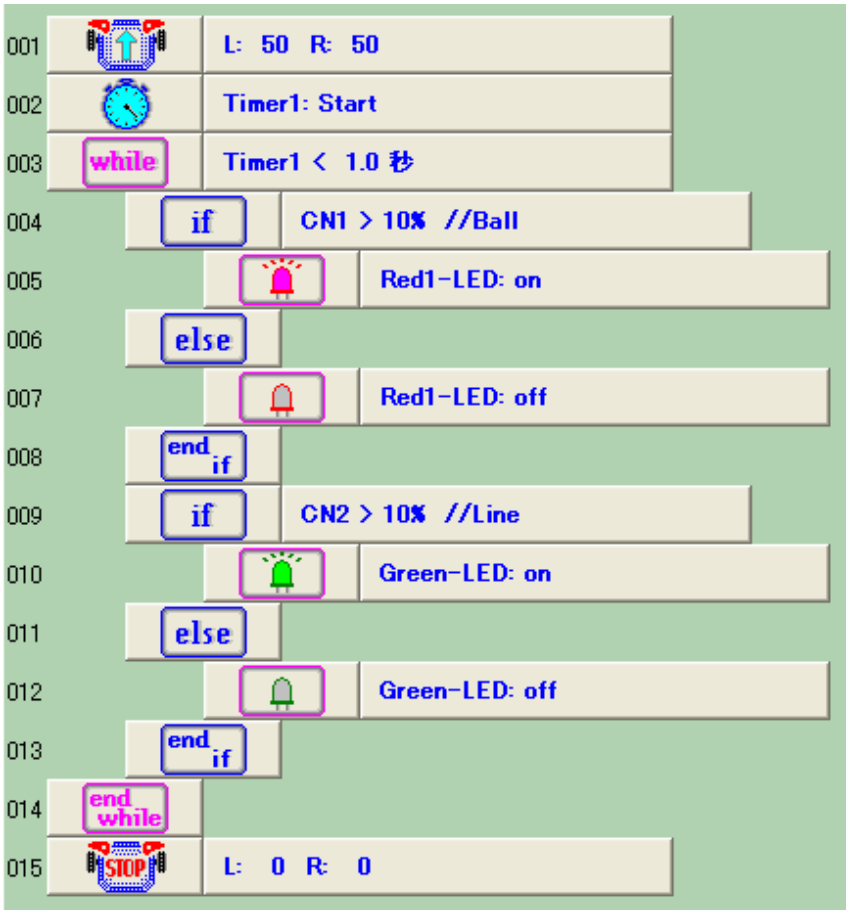
1-6. タイマスタートボタン



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､停止する機能はありませんので､再計測する場合は再スタートさせます。

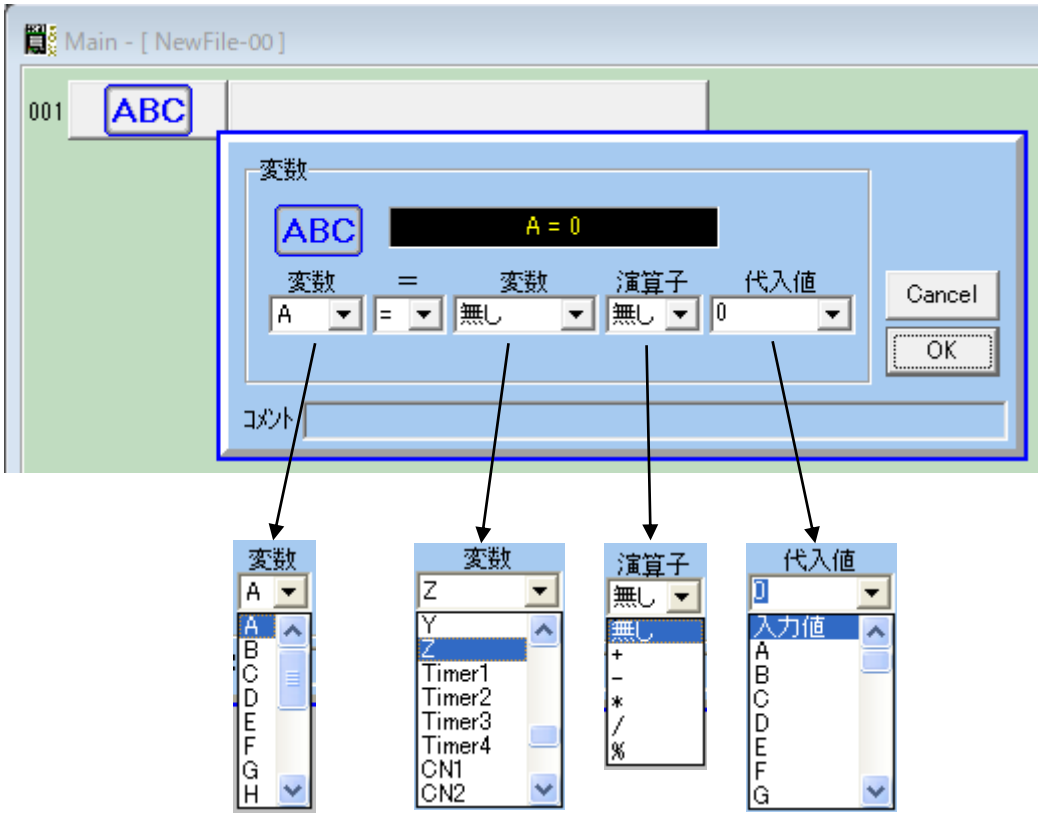
タイマを使ったプログラム例



この例は、導入編「３．ロボットを動作させるまでの手順」で説明した１秒間前進して停止させるプログラムの ウェイトタイマの代わりに タイマを使って同じ動作を実現しています。
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､1.0 秒間の前進の間に CN1 のボールセンサが 10%以上の判定で RED1 の LED を点灯し、CN2 のラインセンサが 10%以上の判定で緑 LED の点灯することが出来ます。

1-7. 変数ボタン ABC

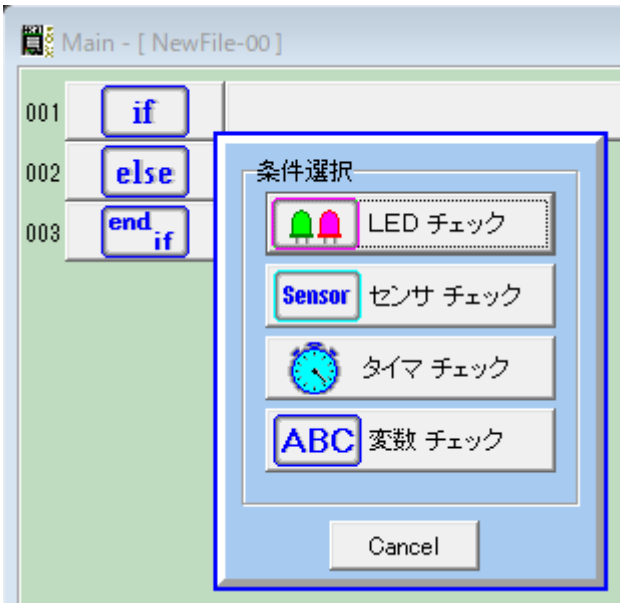


変数は、Ａ～Ｚの２６個が使用出来ます。演算子と＋、－、×、÷、余り計算などが行えます。
またタイマの値やCN1からCN10に接続されたセンサ値も変数へ代入することが出来ます。

◇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（変数Aは0～65535の値になる）
“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 – CN6”	変数Aに変数BからCN6に接続されたセンサ値を引いた値を代入

1-8. 条件分岐



if イフと呼びます

else if エルス イフと呼びます。

else エルスと呼びます。

end if エンド イフと呼びます。

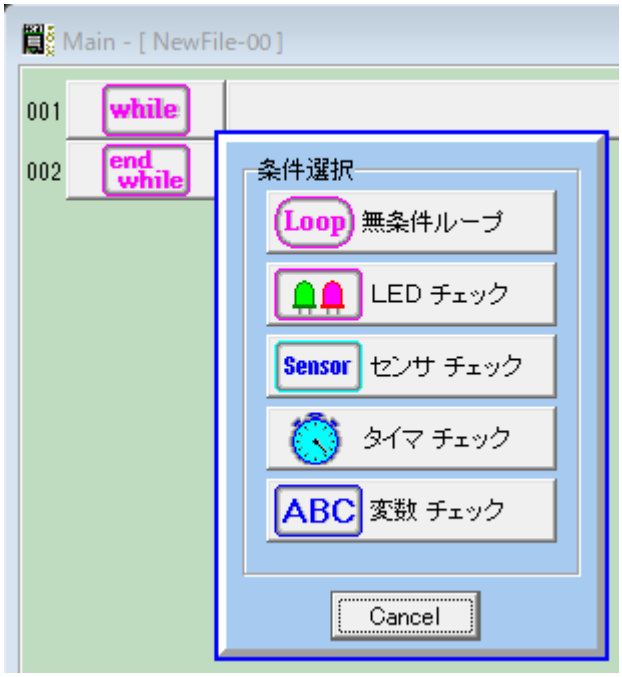
「if」と「else」と「end if」の組合せで必ず配置されます。
「else」は必要に応じて削除出来ます。挿入の場合は「end if」の前に１個だけ置けます。
「else if」は「if」と「else」または「end if」の間に幾つでも置けます。
「if」または「else if」を置いた時に、LED 点灯状態、ボールセンサ、ラインセンサ、タッチセンサ、タイムチェック、変数チェック等の判定を行う為の条件選択が表示されます。

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 行目のボタンをクリックします。

「if」を切取りまた削除する場合は「end if」までのブロック単位となります。

1-9. 条件付き繰返し

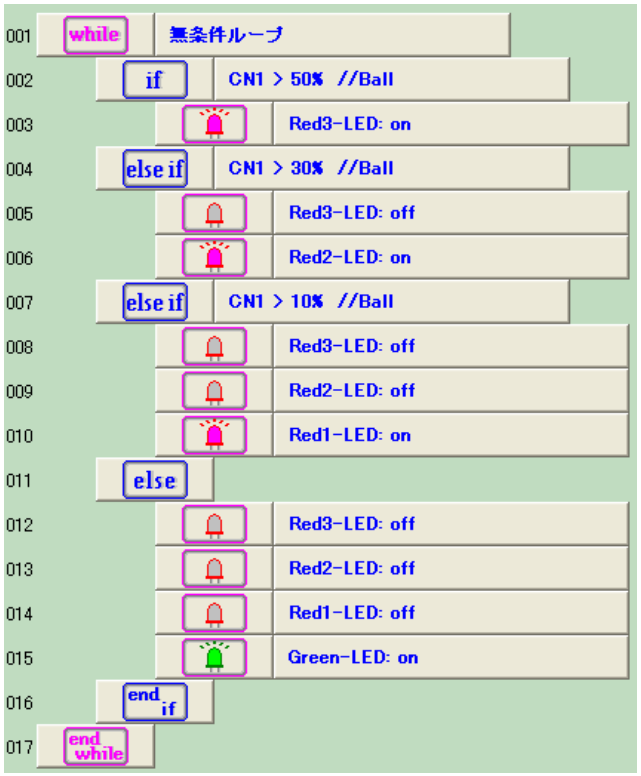


while ホワイルと呼びます。
end while エンド ホワイルと呼びます。
条件が成立している間、「while」と「end while」の間に置かれたプログラムボタンが繰返し実行されます。
「while」と「end while」ボタンの組合せで必ず配置されます。

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その位置から繰返しは強制終了され「end while」の次の行から実行されます。

continue コンティニューと呼びます。 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)
このボタンは拡張機能の設定（Setup 画面の“Advanced Mode”にチェックを付ける）で表示されます。

while, if, elseif, else プログラム例



while(001)～end while(017)の範囲を無条件で繰り返す
002～003：CN1 のボールセンサが 50%以上で
RED3 の LED を点灯
004～006：CN1 のボールセンサが 30%以上で
RED3 の LED を消灯 RED2 の LED を点灯

007～010：CN1 のボールセンサが 10%以上で
RED3, RED2 の LED を消灯 RED1 の LED を点灯

011～015：上記の条件以外の場合は
RED3, RED2, RED1 の LED を消灯 緑 LED を点灯

1-10. 回数指定の繰返し

for フォアと呼びます。

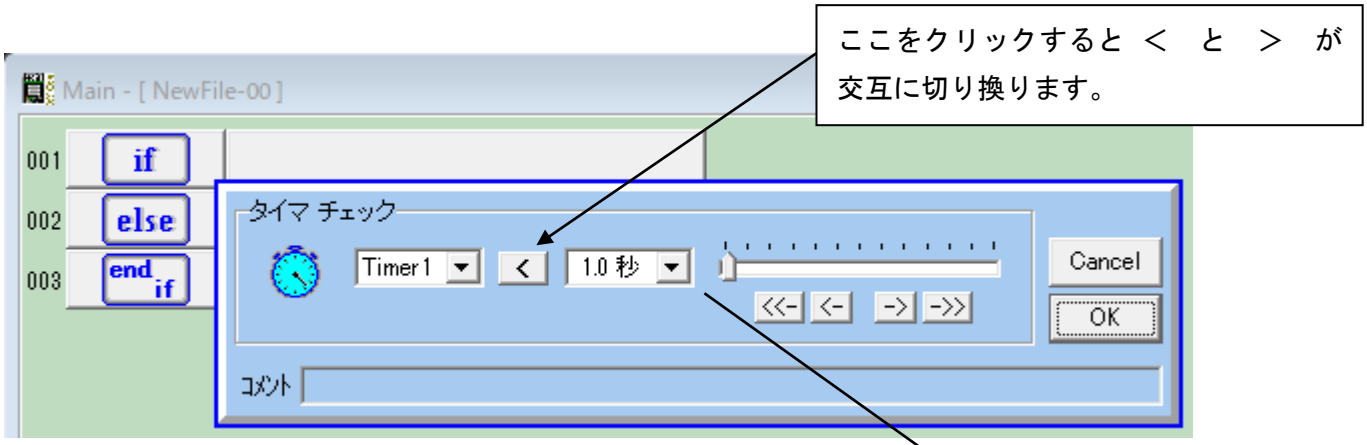
end for エンド フォアと呼びます。

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。

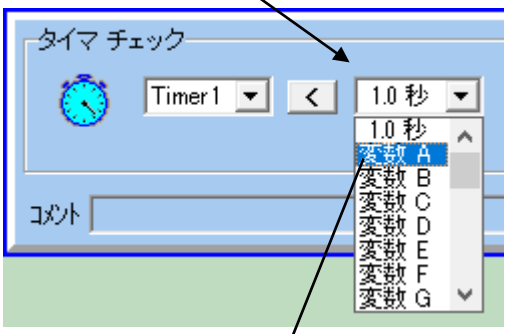
break ブレイクと呼びます。　繰返しの中にだけ置けるボタンで、このボタンに出会うと、繰返しを強制終了し、その位置から「end for」の次の行まで実行移されます。
「break」から「end for」までのプログラムボタンは実行されません。

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)
このボタンは拡張機能の設定（Setup 画面の“Advanced Mode”にチェックを付ける）で表示されます。

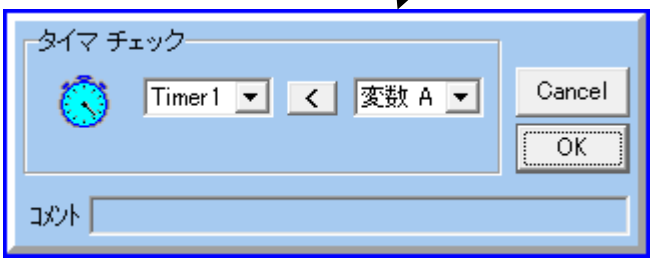
1-11. タイマチェック



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。

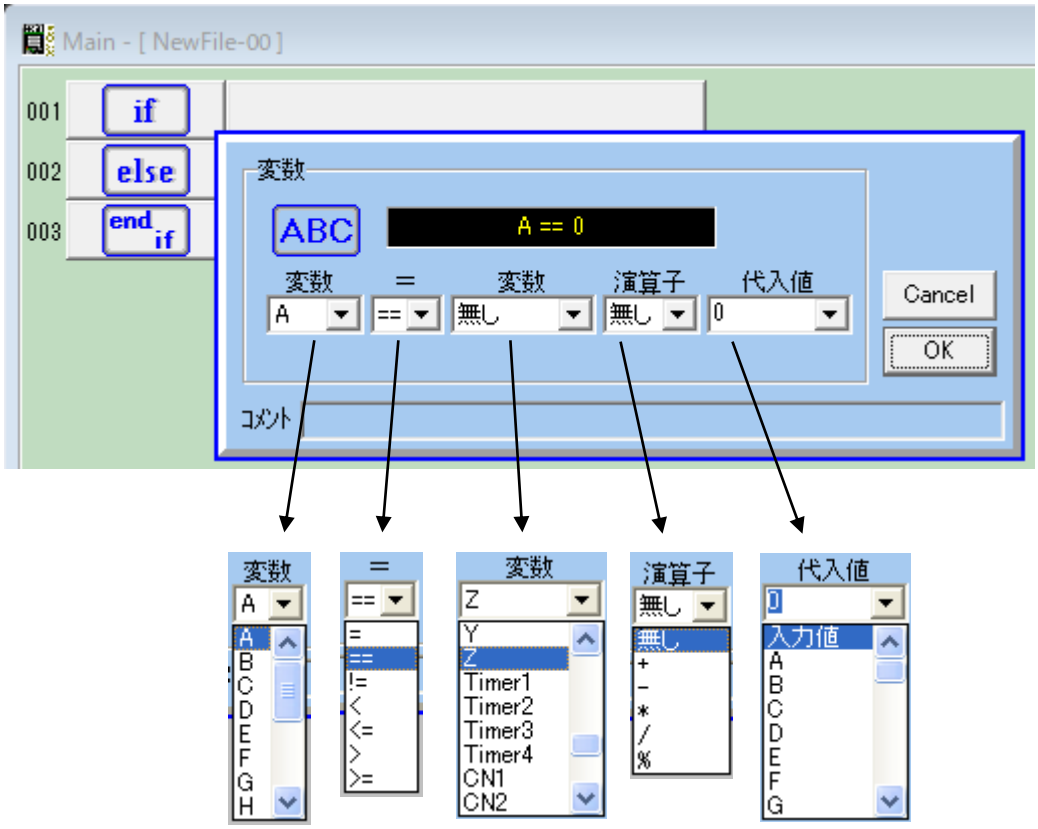


ボタン名称の例
“Timer1 ＜ 0.1 秒以下”
“Timer2 ＞ 1.5 秒以上”
“Timer1 ＞ 変数Ａ 以上”
“Timer2 ＜ 変数Ａ 以下”



◇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

1-12. 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。

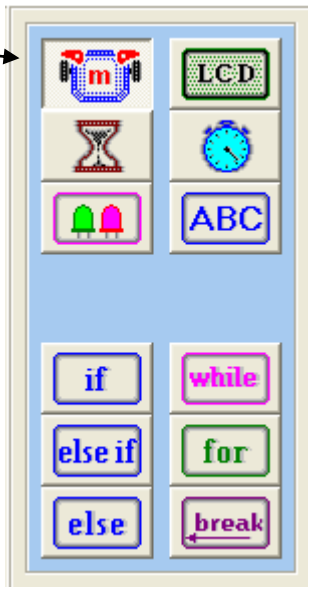
ボタン名称の例

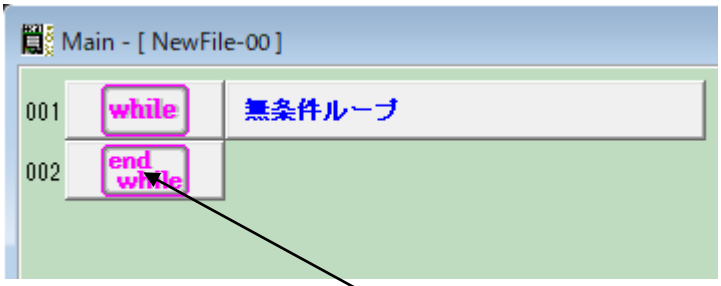
“A == 0”	変数 A が 0 の場合
“A ＜ 5”	変数 A が 5 以下の場合
“A ＞ B”	変数 A が変数 B より大きき場合
“A ＜ CN1”	変数 A より CN1 のセンサ値の方が大きい場合
“A ＜ B － CN2”	変数 A より変数 B から CN2 のセンサ値を引いた値の方が大きい場合
“A == B ％ 2”	変数 A が変数 B を 2 で割った余りと等しい場合（変数 A が奇数か？の判定）
“A ＞ CN3”	変数 A より CN3 のセンサ値の方が小さい場合
“A ＜ Timer3”	変数 A よりタイマ３の値の方が大きい場合

２． プログラムボタンの挿入と削除

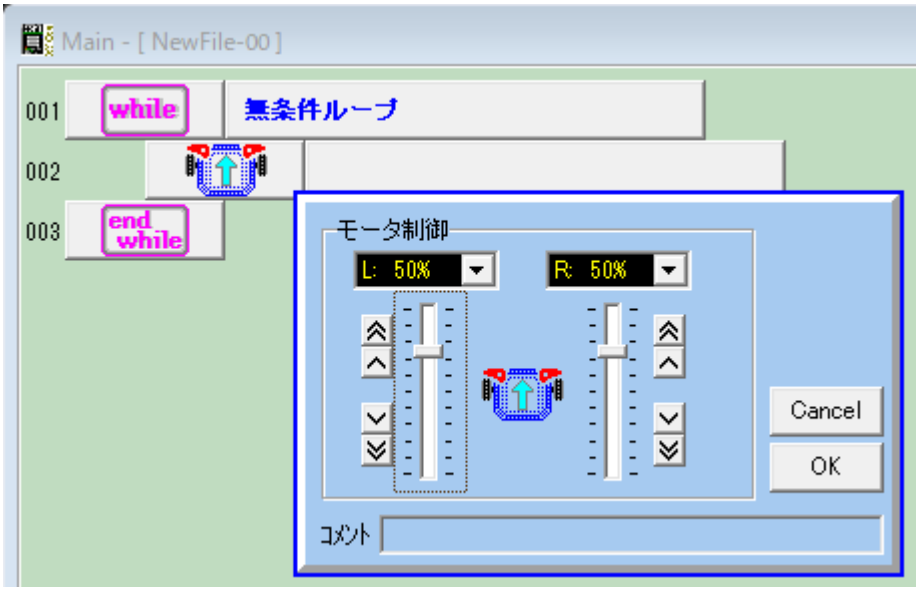
2-1. ボタンの挿入

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。



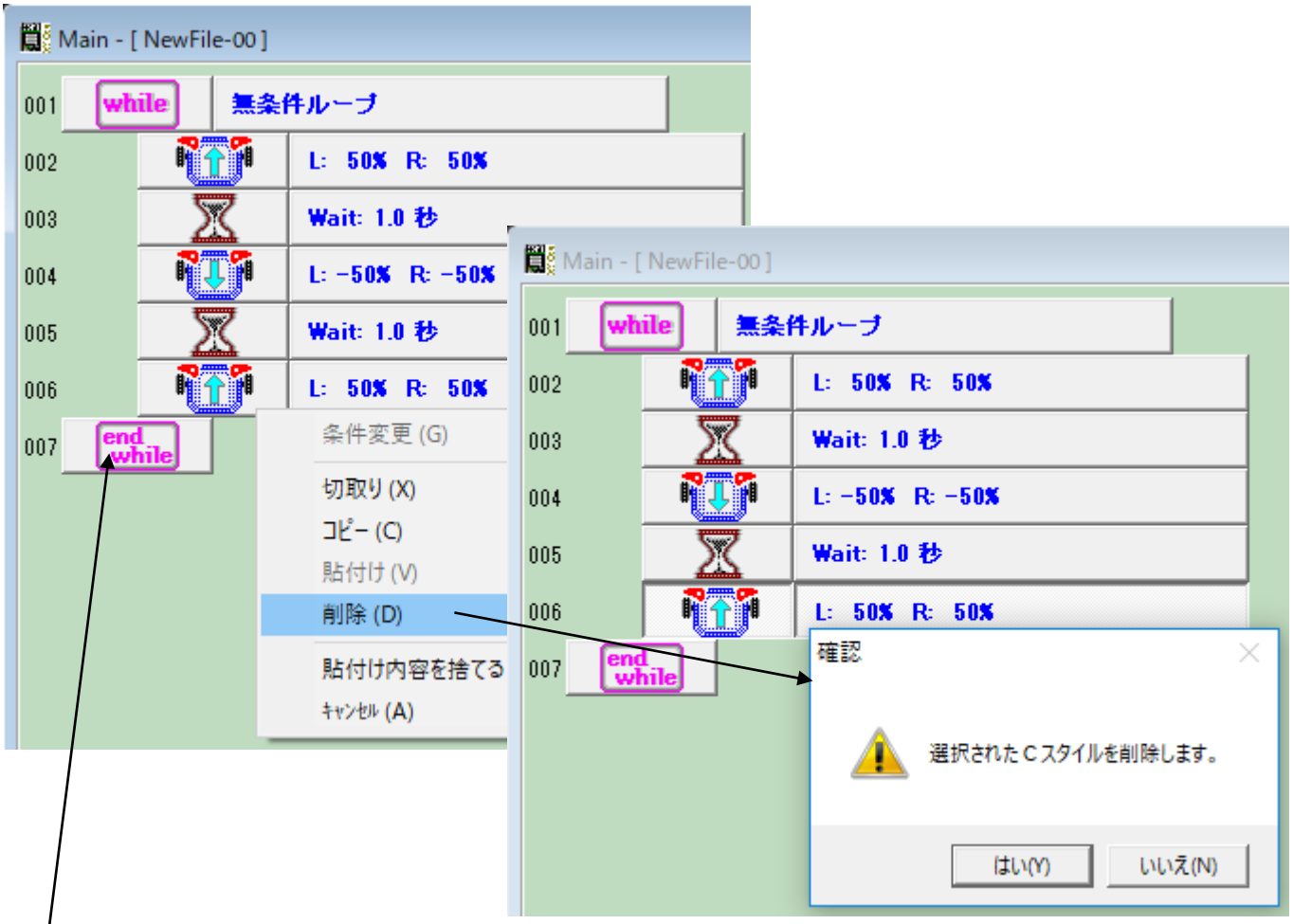


② 次に、挿入したい行の先頭ボタンをクリックすると挿入されます。



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



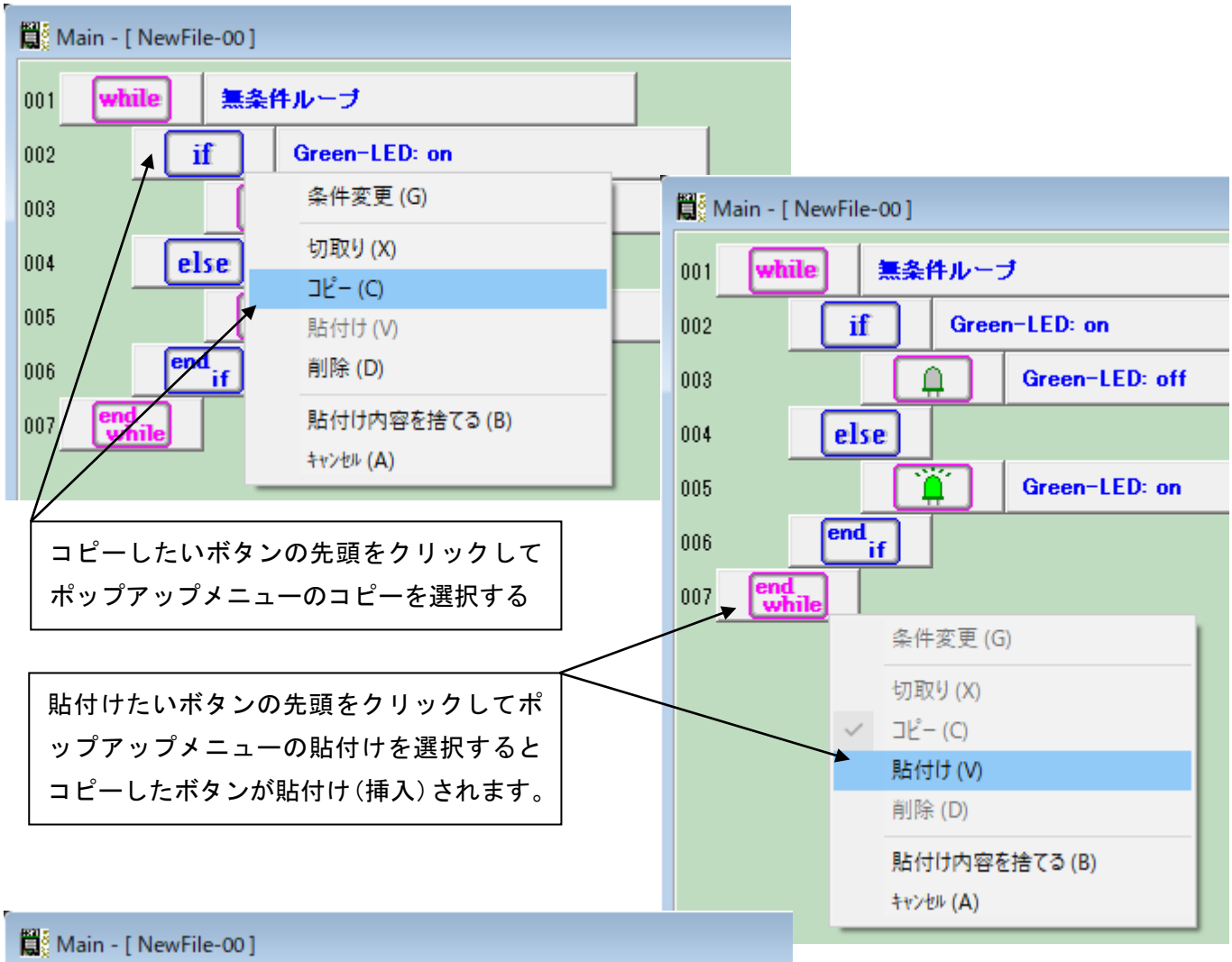
① 削除したい行の前側のボタンをクリックするとポップアップメニューが表示されますので、「削除」を選択します。

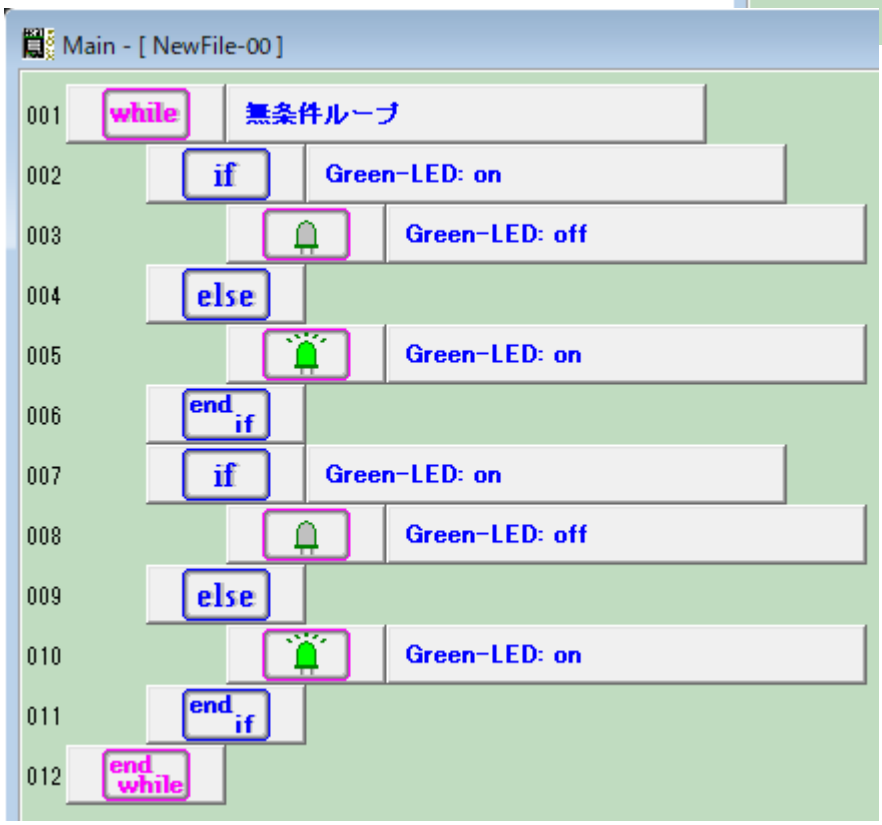
② ボタン削除の確認ダイアログが表示されますので、「はい」をクリックすると、その行のボタンが削除され、行が詰められます。

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





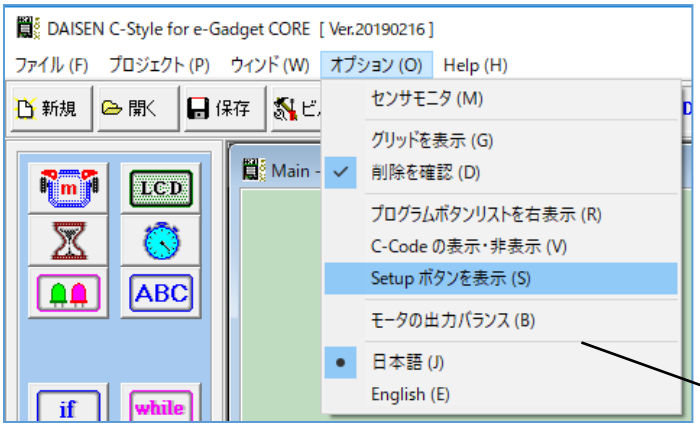
コピーを連続して行うと、以前にコピーした内容の後に追加され、貼付け時に全て挿入されます。
この機能は、切取りにおいても同じ働きをします。
コピーや切取りで蓄積された貼付け内容がいらなくなった場合は、ポップアップメニューを表示させて、「貼付け内容を捨てる」を選択します。

３． 入出力設定と拡張機能の設定

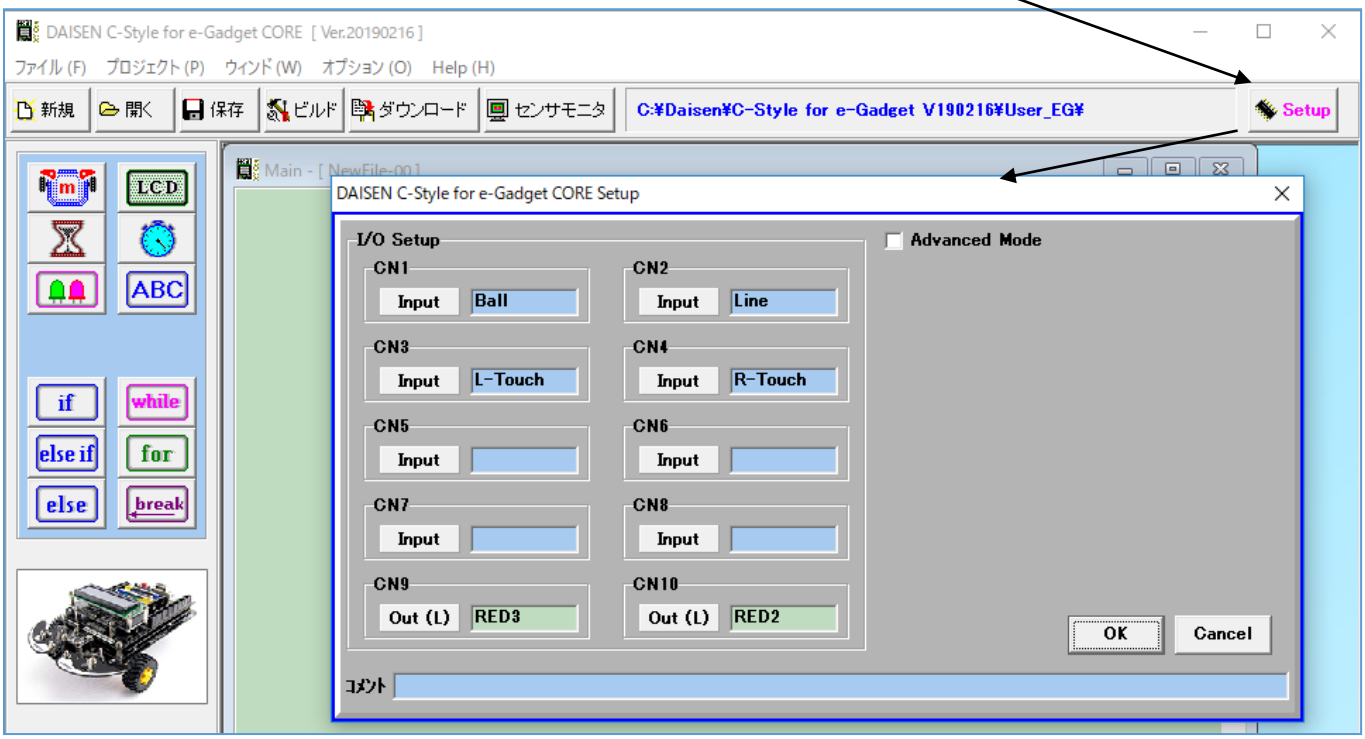
e-Gadget CORE には、CN1 から CN10 までの入出力端子があります。
出荷時の状態では、CN1～CN4 は赤外ボールセンサ、ラインセンサ、左右タッチセンサに接続され、CN5 から CN8 までは予備で、これらはアナログ入力の設定となっています。また CN9 と CN10 はデジタル出力でロボット本体のＲＥＤ３、ＲＥＤ２のＬＥＤに接続されています。これらの端子は設定を変えることで Output 設定にも出来ます。また CN7 と CN8 はサーボモータ制御、CN7～CN10 は超音波センサを接続することも出来ます。

その他に 4ch・6ch モータコントールボード、電子コンパス(6D/9D-COMPASS/HMC6352)、PixyCam.、複数台の e-Gadget CORE をＩ２Ｃ接続出来る機能など拡張機能の設定が行えます。

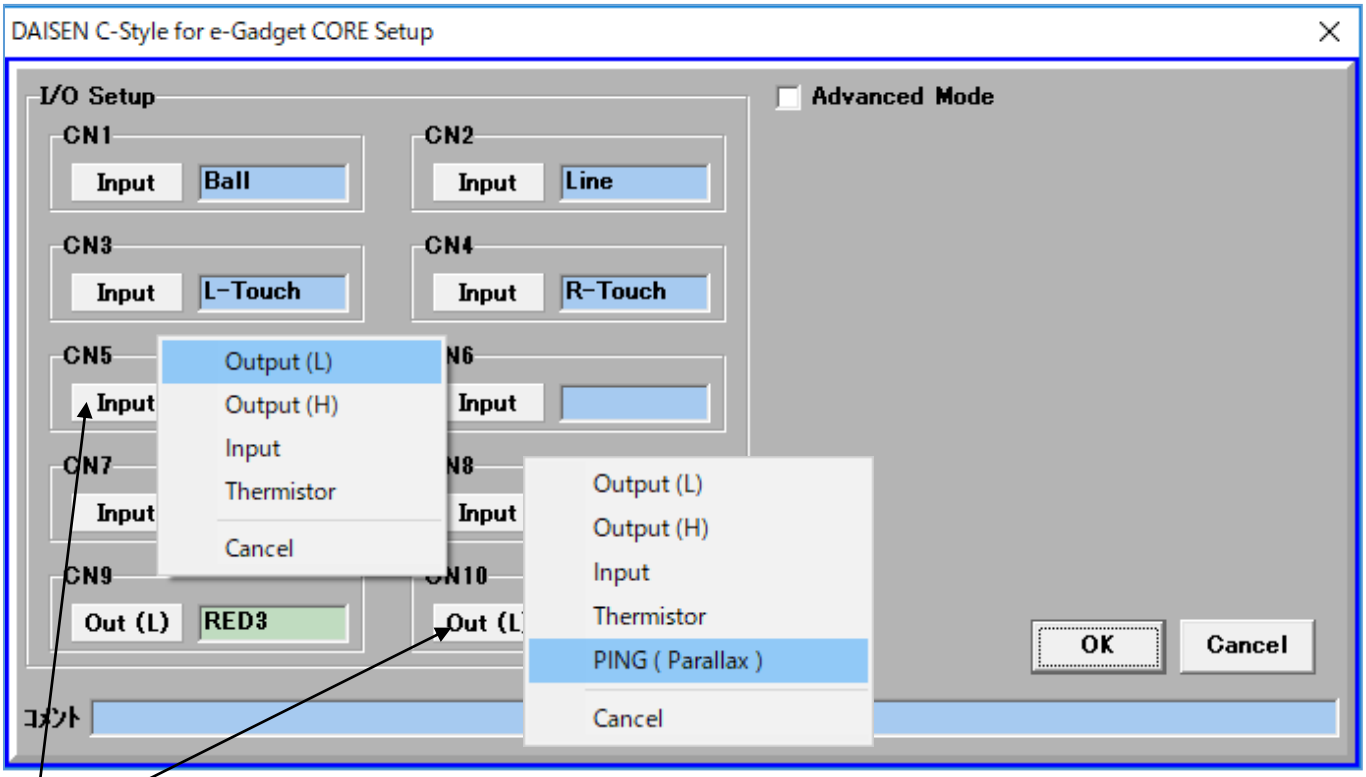
3-1. Setup ボタンを表示



オプションメニューの「Setup ボタンを表示」を選択しますと、画面右側に Setup ボタンが表示されます。
初めてボタン表示選択をした場合は、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。



3-2. I/O Setup（入出力設定）



「 Input 」/「 Output 」ボタンをクリックすると入出力設定用のポップアップメニューが表示されます。

ＣＮ７とＣＮ８はサーボモータ制御の設定も出来ます。
またＣＮ７からＣＮ１０は超音波距離センサ（Parallax 製の PING）を接続できる機能もあります。
（この設定は、ＣＮ１０からＣＮ７へと逆に設定していく必要があります。）

Output 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。
Input は標準のアナログ入力センサです。

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

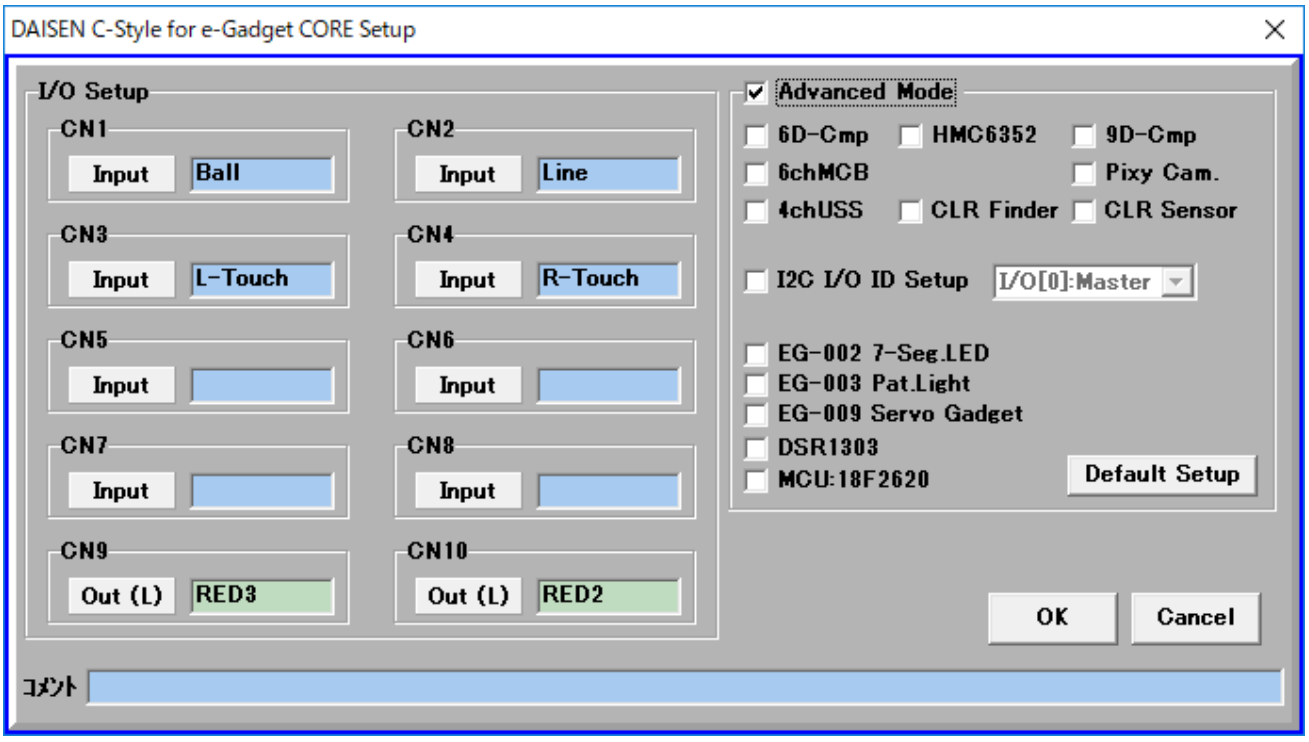
入出力の設定を変更した場合は、C-Style プログラムをビルドしてロボットにダウンロードすることで、はじめて有効となります。

3-3. 拡張機能の設定

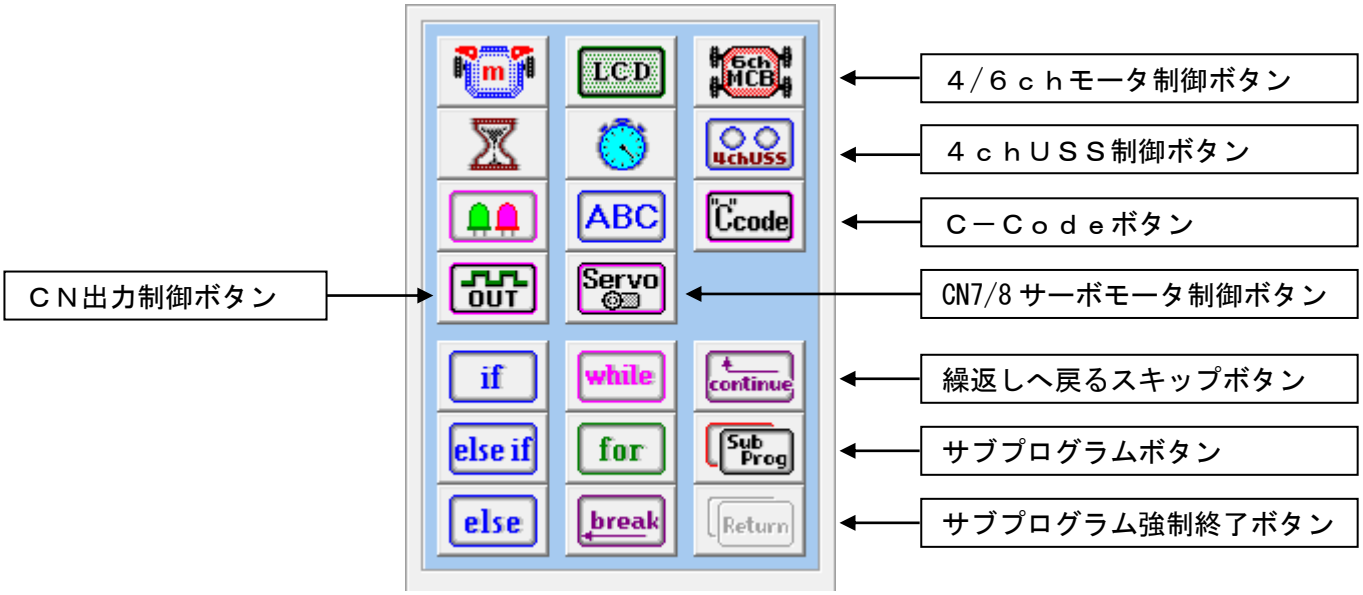
入出力設定ダイアログ内の右側にある　**□Advanced Mode**　にチェックを付けると拡張機能を設定するチェックボックスが表示されます。

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)、4ch/6ch モータコントロールボード設定、4chＵＳＳ(4ch 超音波センサボード：DSR1410)、PixyCam.設定、複数台の e-GadgetCORE を I2C 接続設定で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、EG シリーズ接続設定などの機能拡張の設定が行えます。

「 □ MCU:18F2620 」は標準仕様の MCU:18F2520 をグレードアップした場合に設定します。(プログラムサイズが３２ＫＢから６４ＫＢに増やすことができます。)



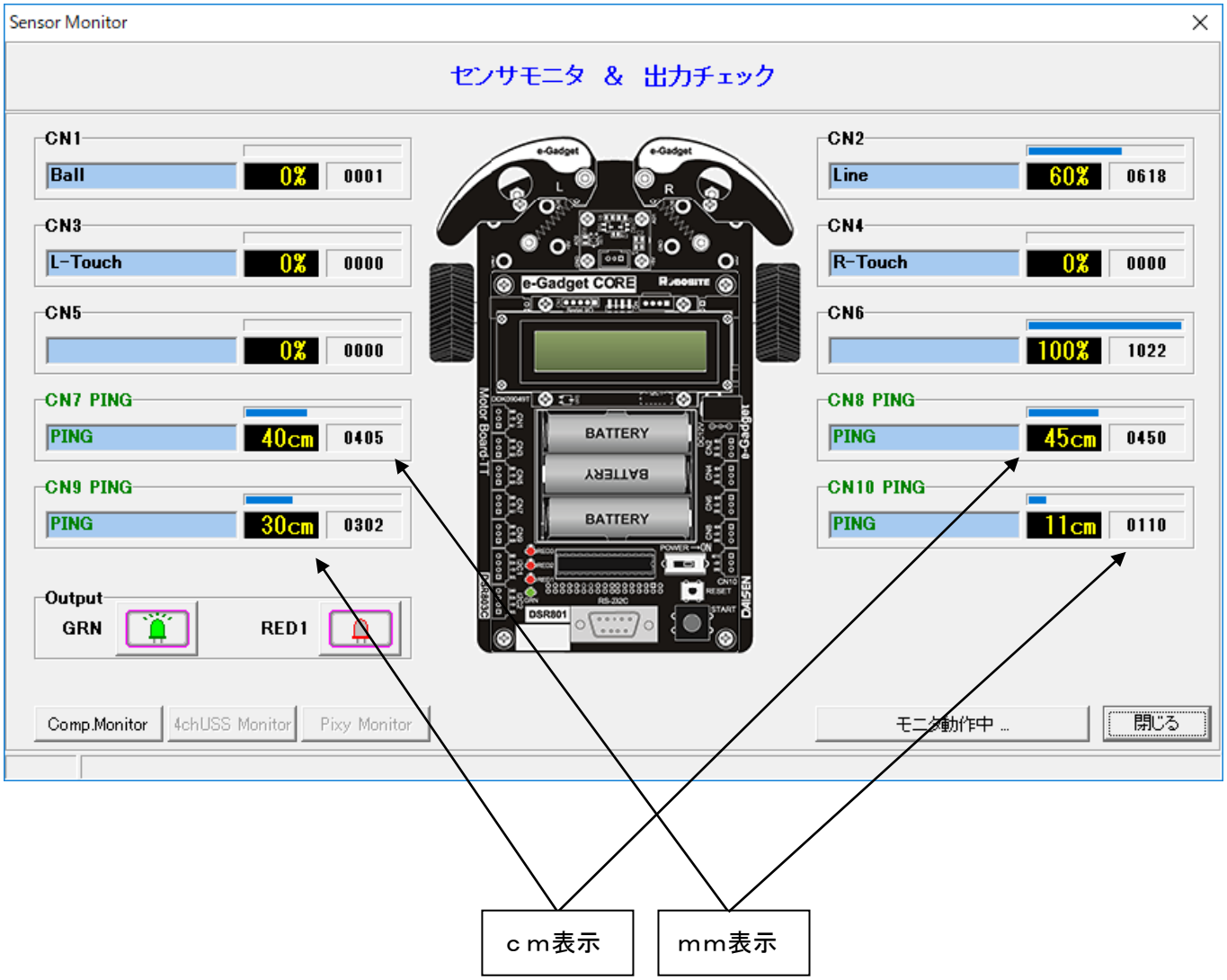
拡張機能を設定するとプログラムボタンリストも下記のように変わります。



3-4. 超音波距離センサを使う

Parallax 製の PING 超音波距離センサを接続する場合は、
入出力設定で、CN10～CN7 の Output/Input ボタンをクリックして PING に設定し、該当する CN 番号の PING センサのチェックプログラムを作成してビルド、ダウンロードすることでセンサモニタの値は%表示からｃｍ表示に変わります。（約３ｃｍから３ｍまで計測可能です）

ＣＮ１０、ＣＮ９、ＣＮ８、ＣＮ７を PING 接続に設定したモニター画面



Setup 画面で PING 設定する場合は、CN10 から CN7 への順に設定して下さい。

ＣＮ１０に接続したPINGが３０ｃｍ以下で左回転するプログラム例

通常のセンサ入力の判定プログラムと同様に編集できます。
上記の例の様に直接判定する場合は、ｃｍの単位となります。
ＣＮ１０＜３０ｃｍ

変数と比較する場合は、mmの単位で変数へ代入します。
変数Ａ＝３００　　（３００mm）
ＣＮ１０＜変数Ａ　　（ＣＮ１０＜３０ｃｍと同じ内容となります）

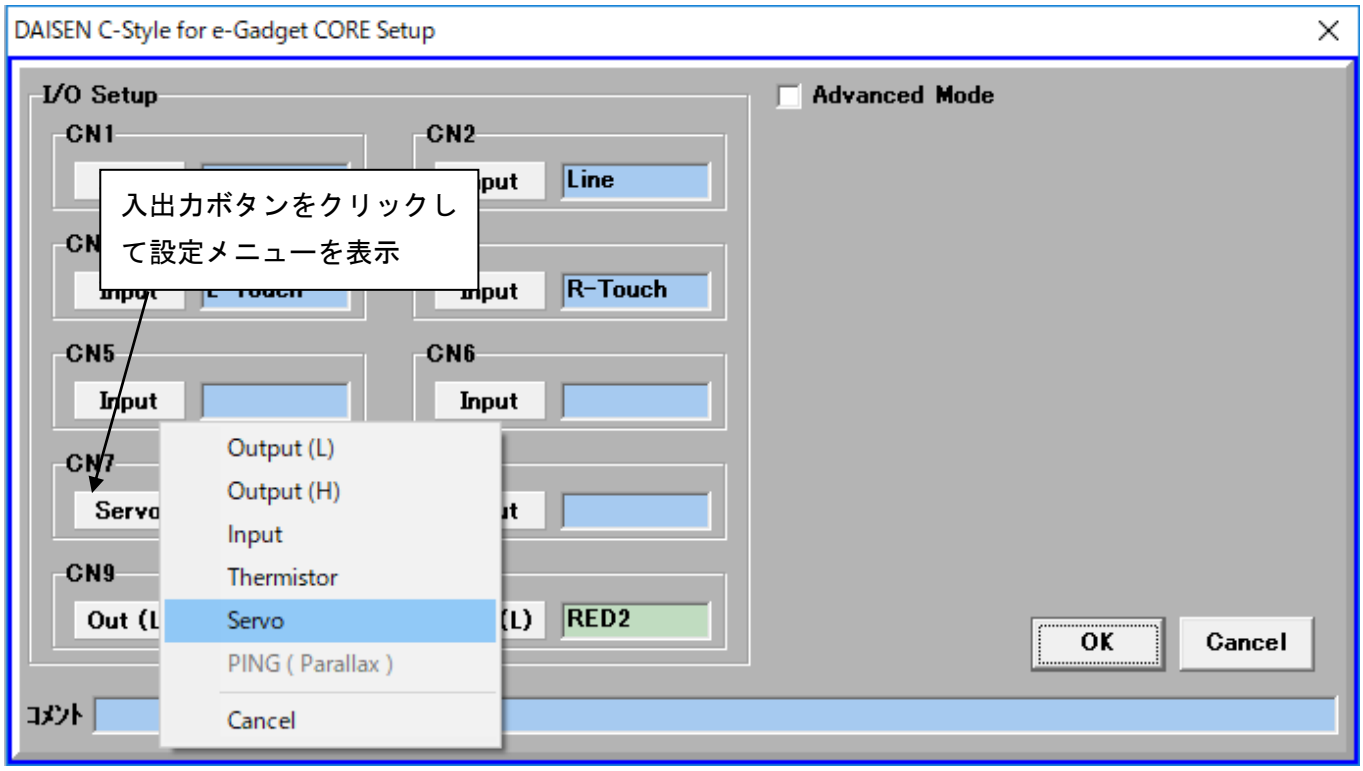
変数への代入の場合も中身はmmの単位となります。
変数Ａ＝ＣＮ１０
変数Ａ＜３００　　（３００mm：３０ｃｍ）

インストールしたC-Styleの User_EG フォルダにPINGのサンプルプログラムがありますので参考にして下さい。

Parallax製のPING 超音波距離センサ（CN10～CN7に接続可能）

3-5. サーボモータを使う

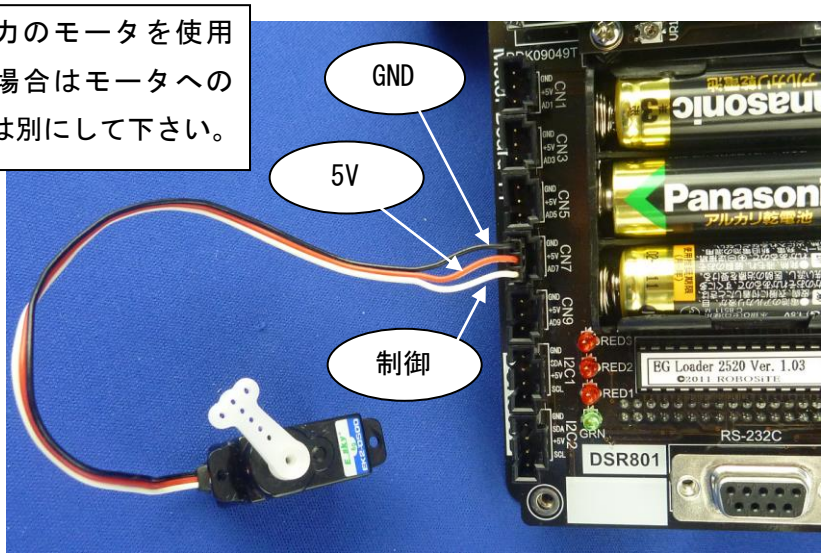
入出力設定で、CN7 と CN8 にはサーボモータ接続用の設定メニューがあります。



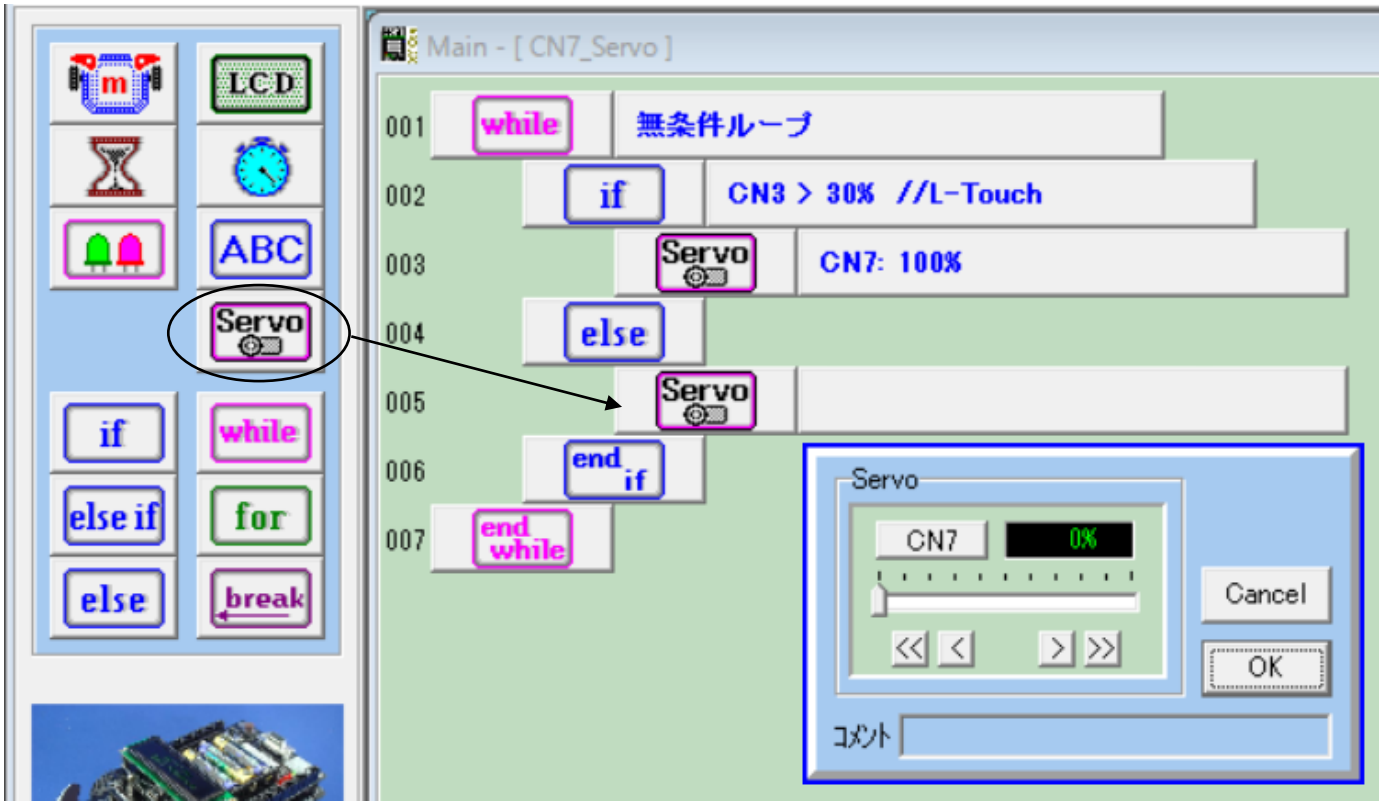
サーボモータは５Ｖで動作するものが使えます。コネクタの３ピンの内２番目（中央）が電源端子になっていれば、ロボット本体の CN7(CN8)に直接差し込むだけです。１番ピンと３番ピンは制御信号とＧＮＤですので、ロボット側のコネクタに一致するように接続します。

サーボモータの設定をするとプログラムボタンリストにサーボ制御ボタンが追加されます。

高出力のモータを使用する場合はモータへの電源は別にして下さい。



サーボモータ制御プログラム例



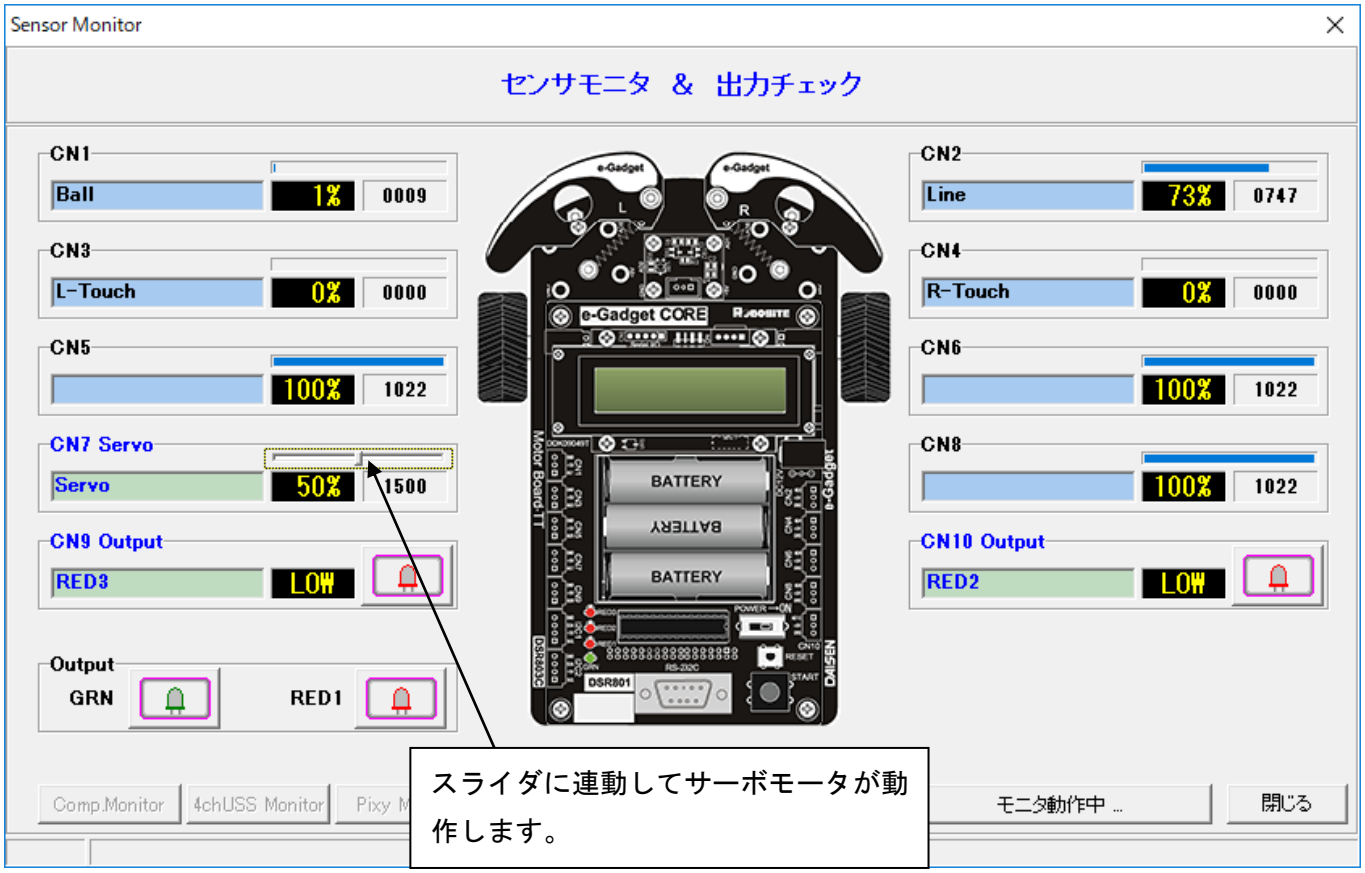
この例では、左側のタッチセンサを使ってサーボモータを動作させています。
サーボ制御は０％（５００μ秒）から１００％（２５００μ秒）で設定できます。
中央の位置が５０％（１５００μ秒）です。

使用するサーボモータによっては、最小値、最大値が上記の例とは異なる場合があります。
限界を超えた値を与え続けると故障の原因となりますので、センサモニタで調べて下さい。

CN7 と CN8 の両方を使用する設定にしている場合は、設定ダイアログ内の「CN7」のボタンをクリックすると「CN8」の表示に変わり制御するサーボを選択することが出来ます。

ロボットからの電源でサーボモータを動かすとロボットの電源回路が破損する恐れがありますので、高出力のサーボモータを使用する場合は、必ず別の電源でサーボモータへ電源供給して下さい。

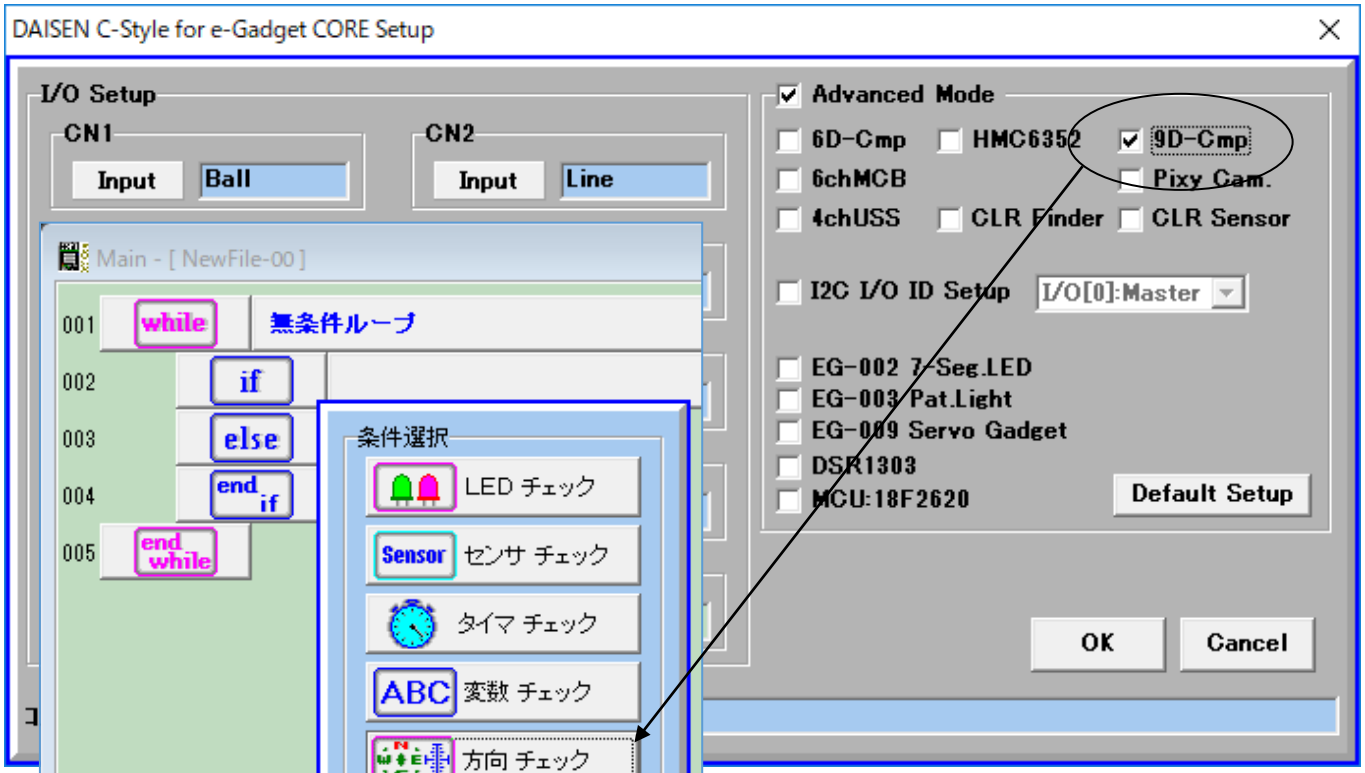
サーボモータ設定時のセンサモニタ画面



サーボモータ制御機能を設定したプログラムをダウンロードすればセンサモニタでの動作確認が行えます。

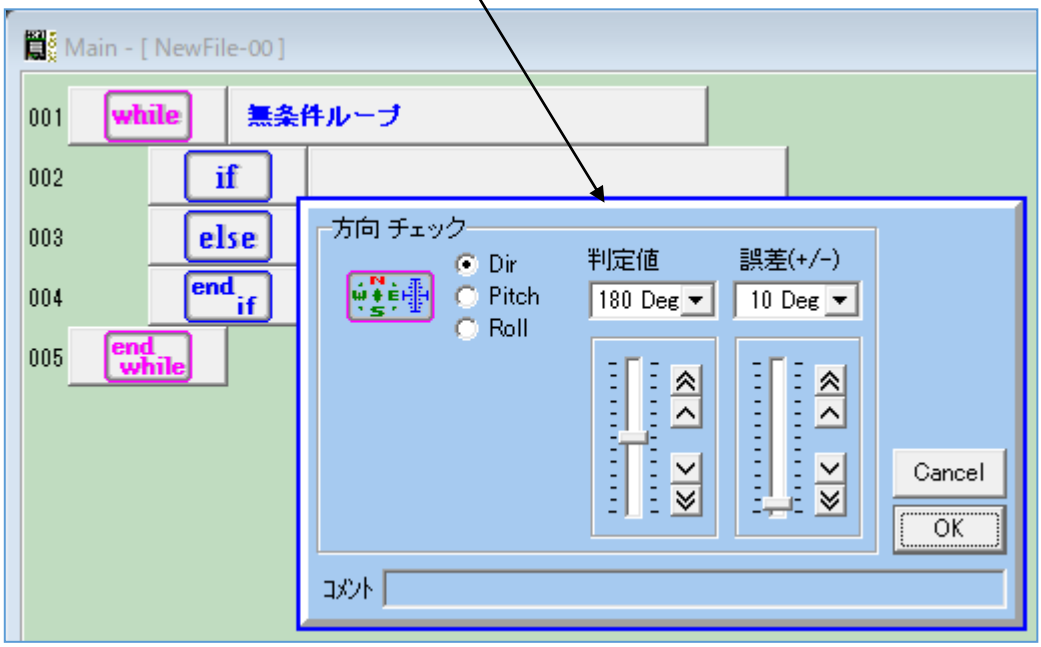
3-6. 6D/9D-Compass（多機能電子コンパス：DSR1401/DSR1603）を使う

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

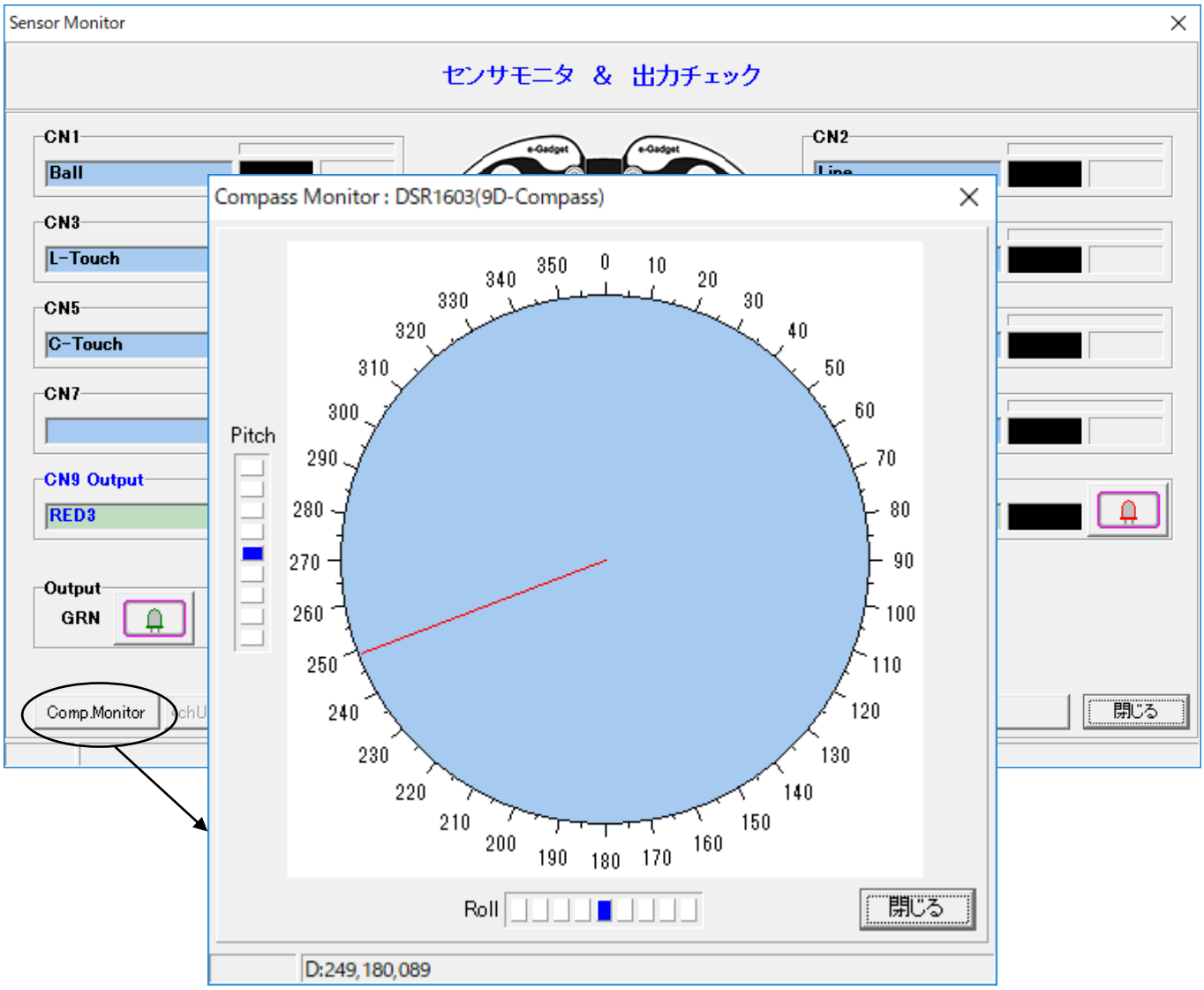


6D-Cmp または **9D-Cmp** にチェックを付けると条件判断のボタンリストに方向チェックのボタンが追加表示されます。

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。



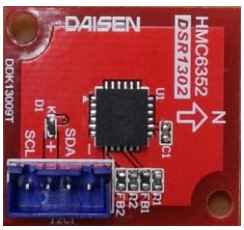
設定画面の 6D/9D-Cmp または HMC6352 にチェックの付いたプログラムをダウンロードすると、センサモニタ画面の左下の「Comp.Monitor」ボタンが有効となります。



センサモニタの「Comp.Monitor」ボタンをクリックすると現在の方向を表示するダイアログを表示することが出来ます。

HMC6352

単機能電子コンパス：DSR1302



6D-Compass

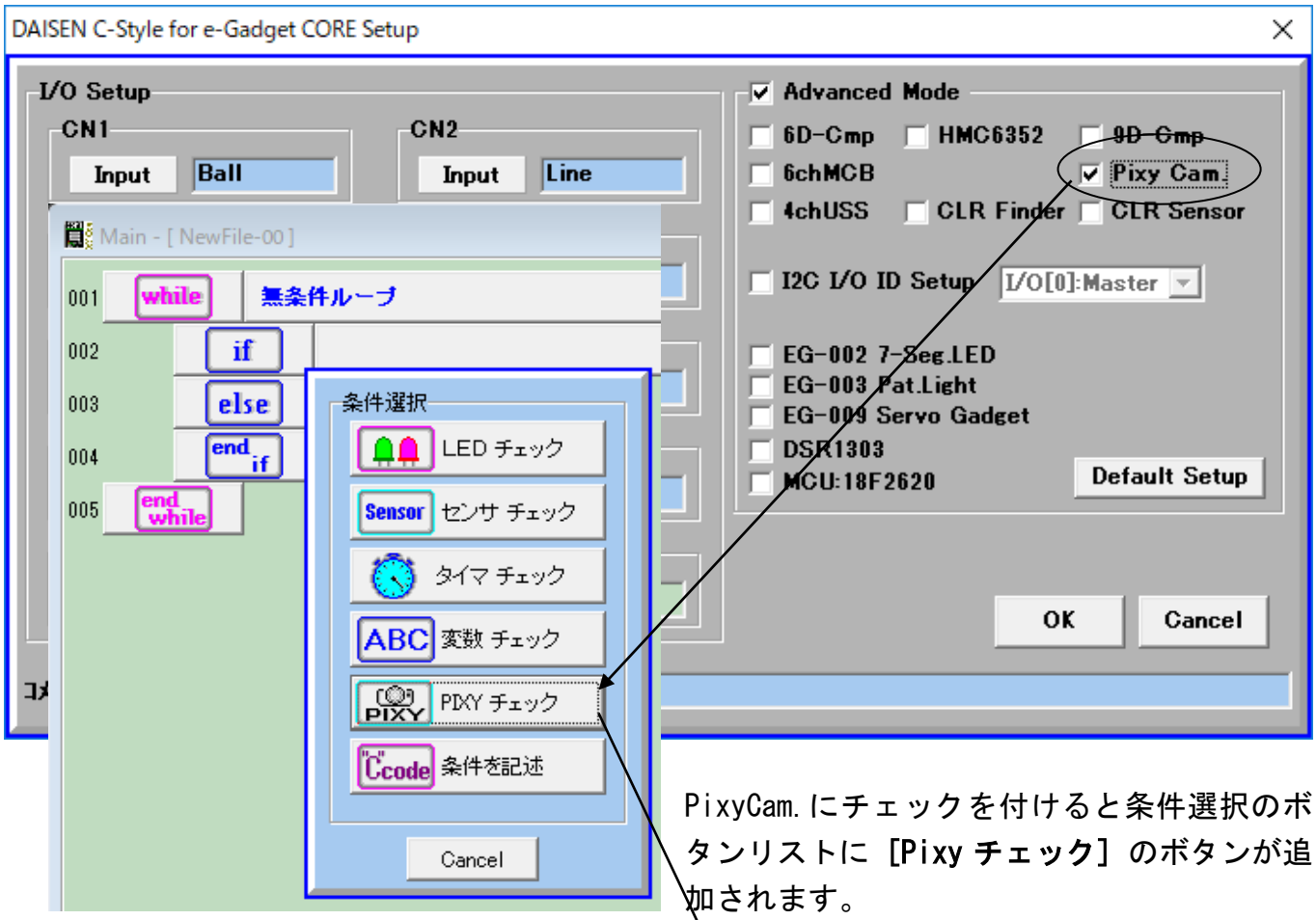
多機能電子コンパス：DSR1401

9D-Compass

多機能電子コンパス：DSR1603

3-7. PixyCam.を使う

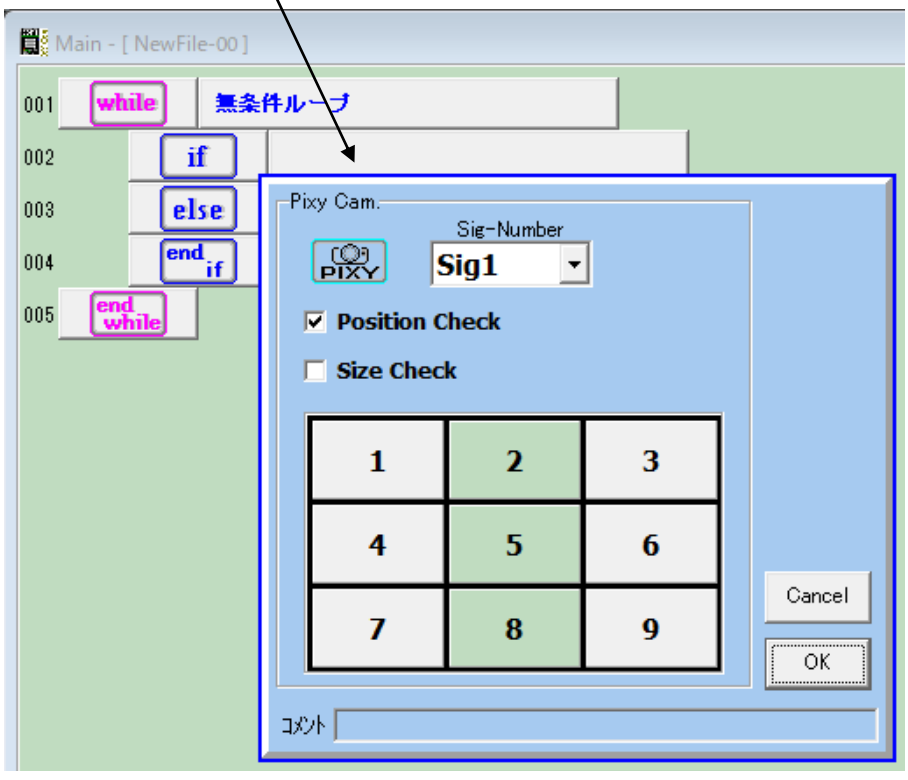
PixyCam.を使用するには、Setup 画面の[Advanced Mode]－[Pixy Cam.]にチェックを付けます。



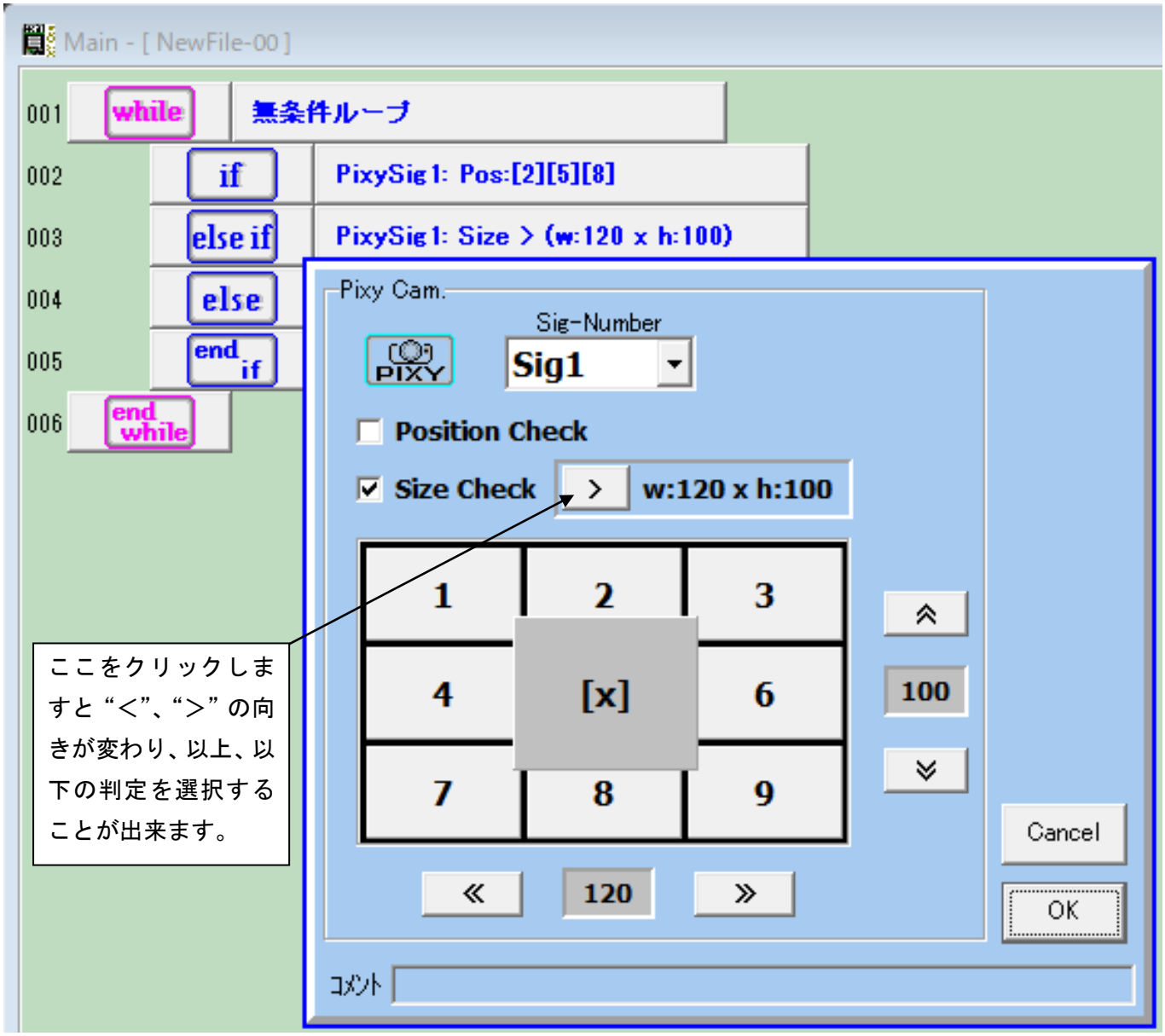
PixyCam.にチェックを付けると条件選択のボタンリストに［Pixy チェック］のボタンが追加されます。

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。

PixyCam 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)



PixyCam.のサイズ指定

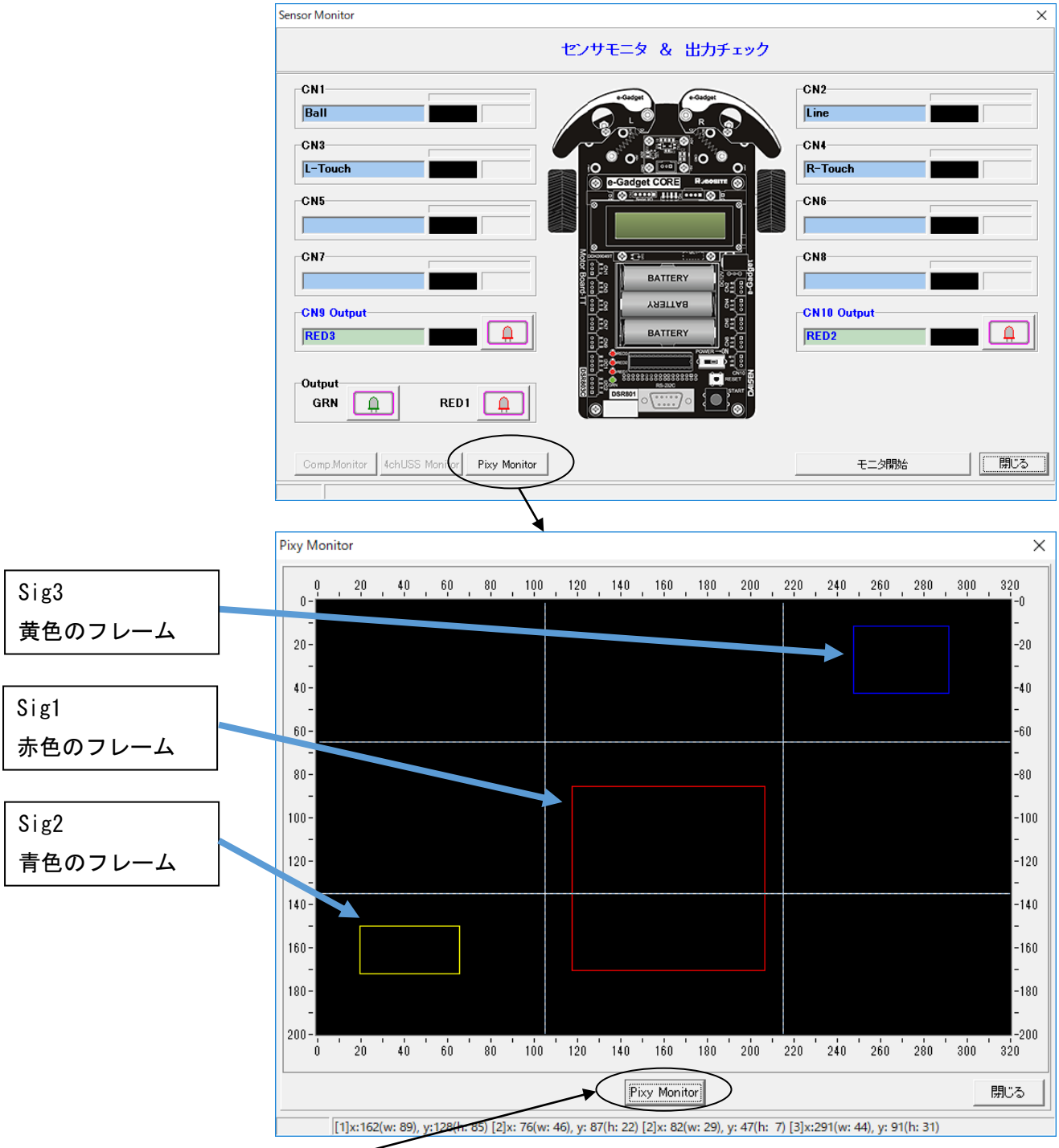


サイズはｈ：１０～１９０、ｗ：１０～３１０の範囲で設定できます。
画面中央でサイズ表示をしていますが、[Position Check] の指定が無い場合は、指定サイズが画面の何処に現れても判定条件となります。

[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。

PixyCam.のセンサモニタ

Setup画面でPixyCamの使用の設定を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の下側にある［Pixy Monitor］のボタンが有効となりますので、それをクリックしますとPixy Monitor画面が表示されます。



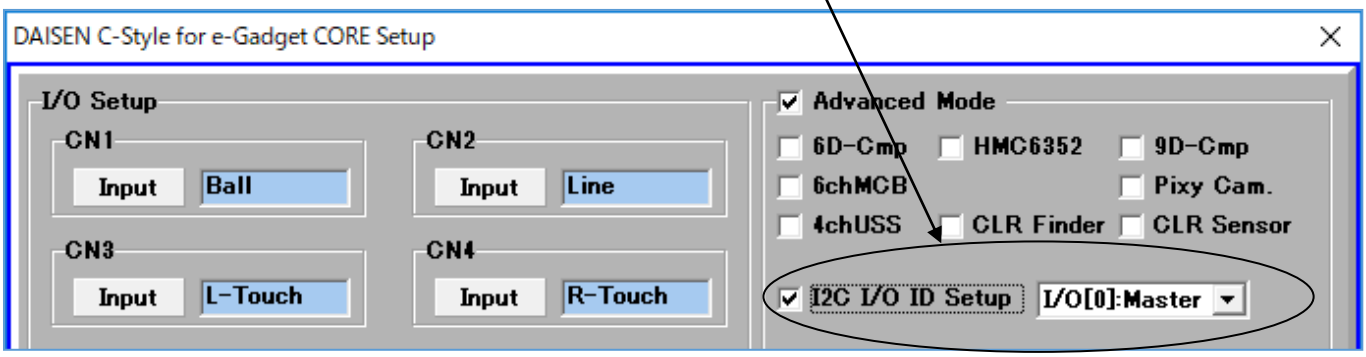
画面下側の［Pixy Monitor］ボタンをクリックするとモニタ動作を開始します。

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。

※ご注意：フレームの色は学習された色とは関係なく固定された色です。

3-8. 複数台の e-Gadget を接続する

入出力設定画面の **I2C I/O ID Setup にチェックを付ける**と最大８台までの e-Gadget を接続して I/O センサボードして使用できるようになります。

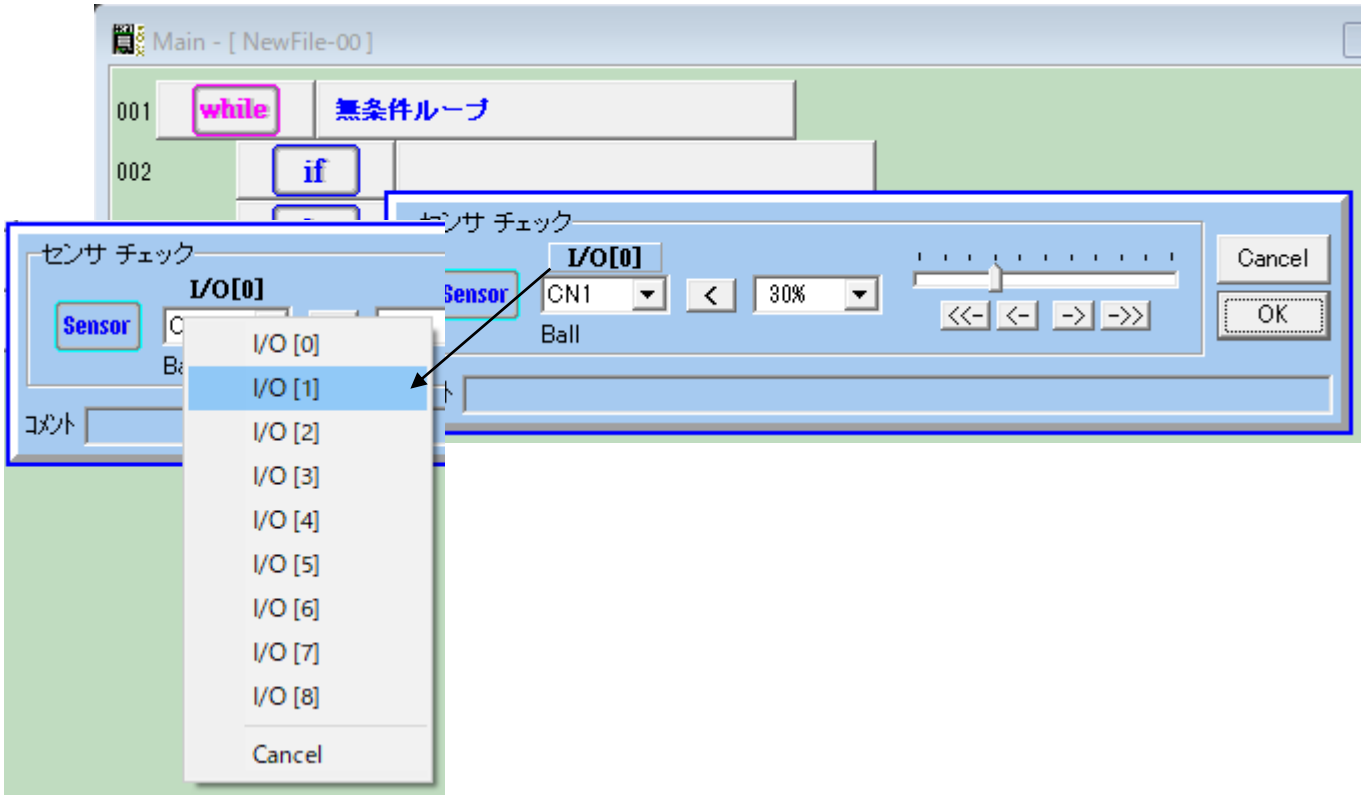


この機能は、通常 e-Gadget は 1 台当たり最大 10 個のセンサ(CN1～CN10)を使うことができますが、I2C という通信仕様のコネクタ(4pin)を使って別の e-Gadget と接続してセンサを増やすことが出来ます。

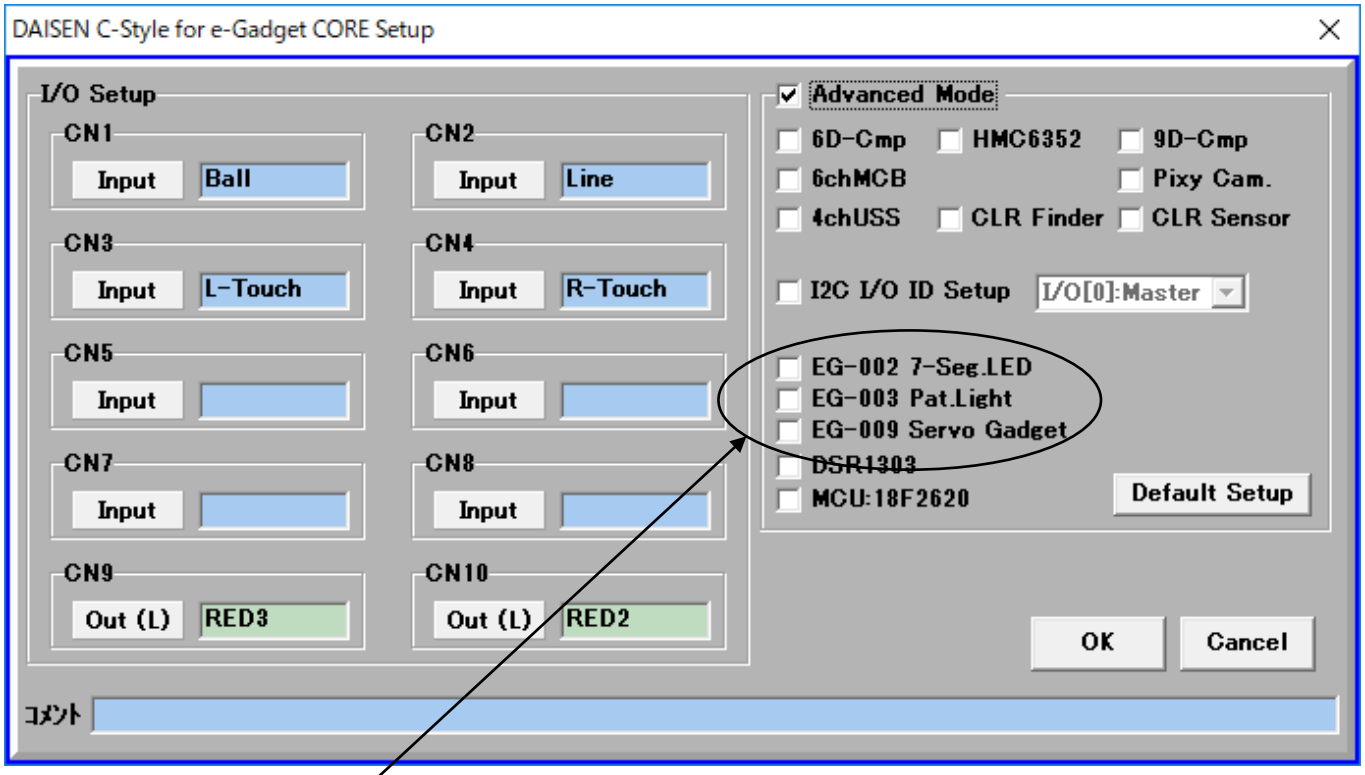
センサ情報を一括管理する親の e-Gadget のことを Master(I/O[0])と呼び、センサ情報を親へ引き渡す e-Gadget のことを SubI/O(I/O[1]～I/O[8])と呼びます。

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 e-Gadget のプログラムは、この 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 のセンサ情報を取出して条件判定することが出来ます。



3-9. EG シリーズを使う

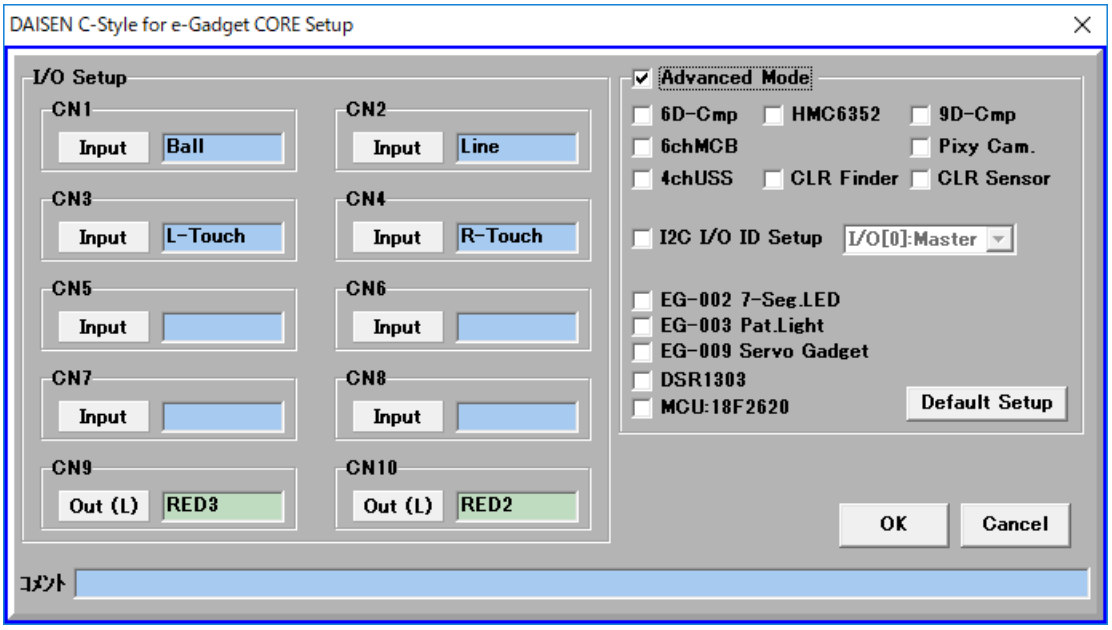


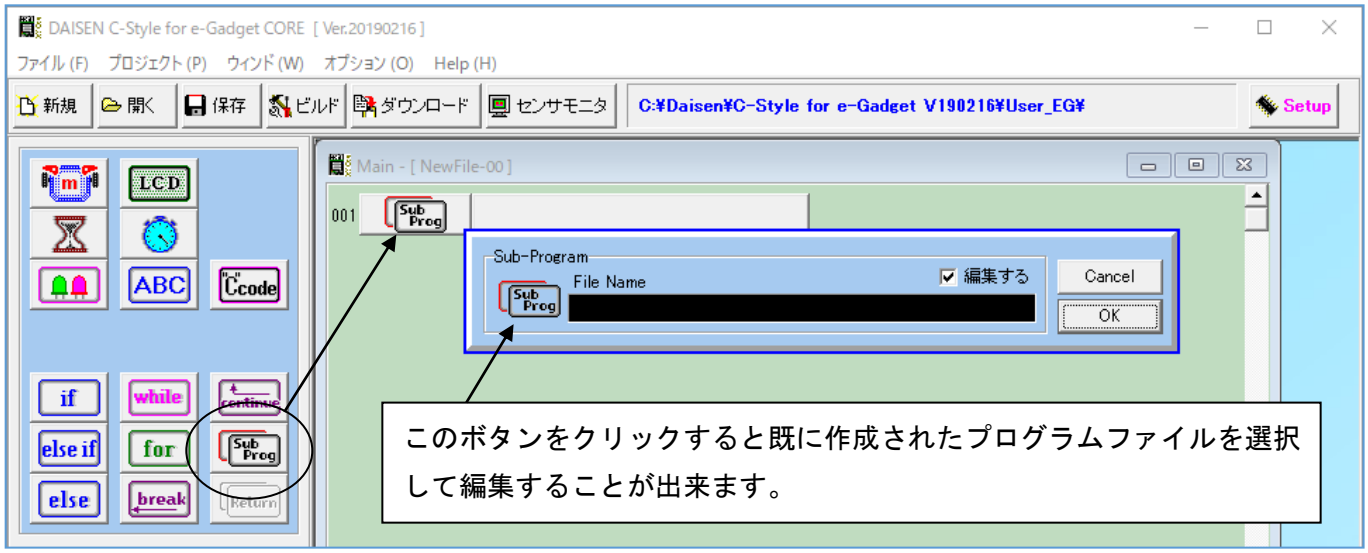
e-Gadget には、EG シリーズと称して拡張パーツが使える機能が実装されています。
現在は、３機種のパーツが用意されています。
EG-002：4 桁の 7Seg.LED 表示器
EG-003：赤、橙、緑色の３色 LED ミニパトライト
EG-009：１軸サーボモータコントローラ
これらの制御コードは、C-Code で記述する必要があります。
サンプルプログラムは、“User_EG¥EG-Sample”のフォルダ内にあります。

４．サブプログラムの説明

4-1. サブプログラムボタンを置く

Setup 画面の［Advanced Mode］にチェックを付けて「ＯＫ」ボタンで画面を閉じます。





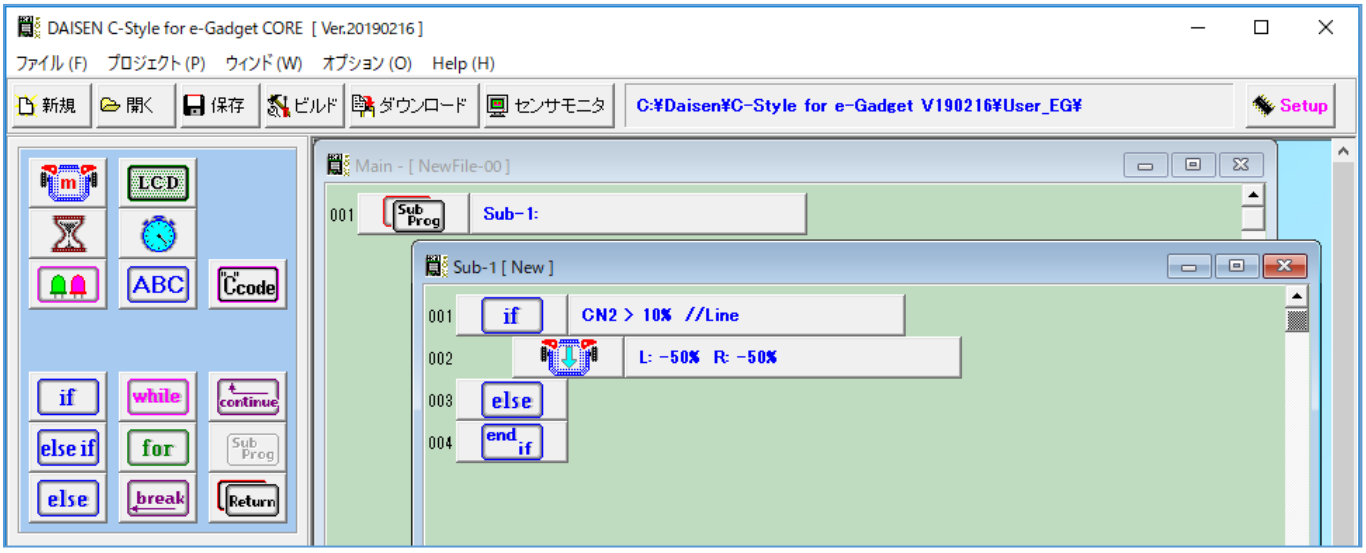
プログラムボタンリストが３列表示になり

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）

サブプログラムは３０種類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出来ません。

4-2. サブプログラムの編集



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、サブプログラムの編集を行います。

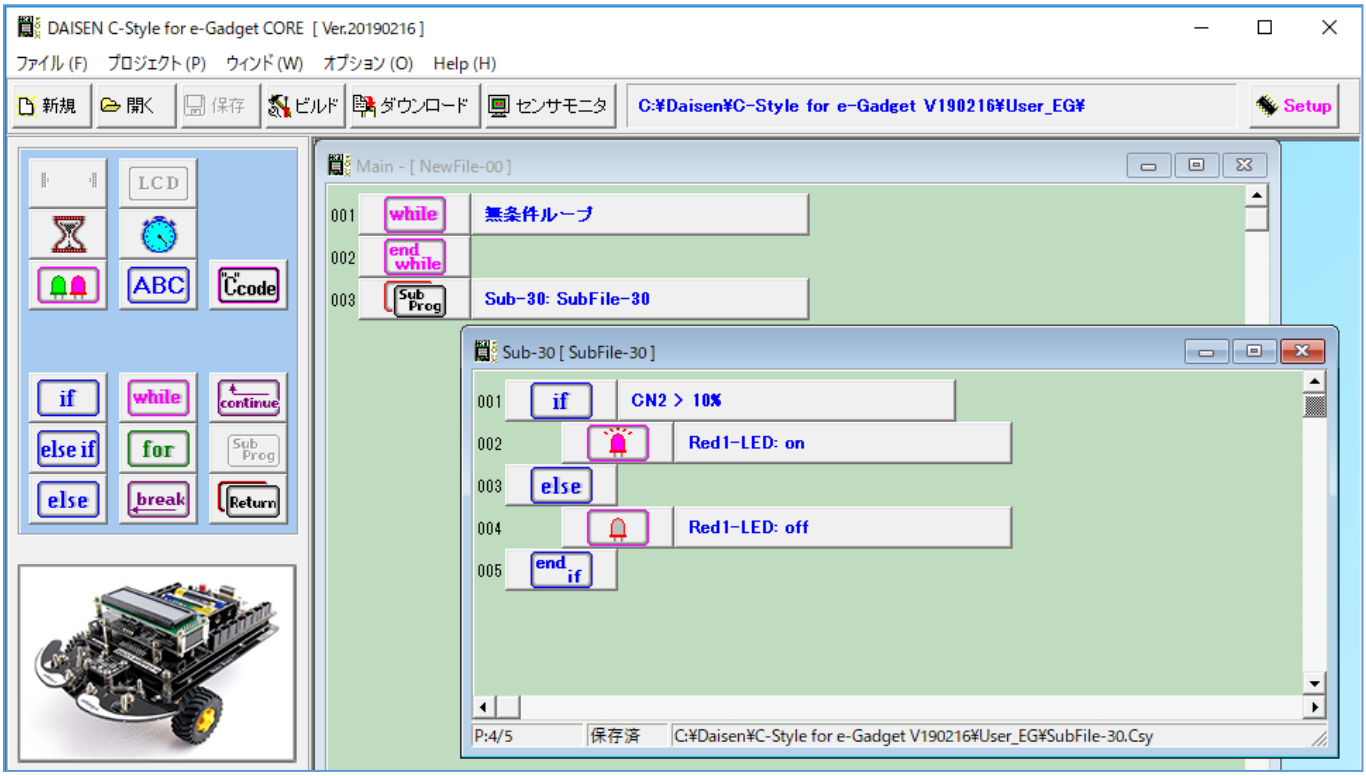
編集されたサブプログラムを保存するには、サブプログラムフォームをアクティブな状態してから「保存」ボタンをクリックします。
アクティブな状態にするには、保存したいフォームのタイトルバーをマウスでクリックしると他のフォームより最前面に表示されます。
サブプログラムはメインプログラムが保存されている同じフォルダ内に保存して下さい。

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。

Setup 画面で設定する入出力設定およびオプションパーツの設定はメインとサブプログラムで全て同じ設定をして下さい。
メインとサブで異なる設定をした場合、ビルド失敗やビルドが成功しても正しく動作しません。

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%以上になると 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 の点灯制御などが使用可能ですが、モータ、PING、サーボモータ、I2C 等の制御は出来ません。

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://daisendenshi.com
Email ddk@daisendenshi.com