

NRobEdit1 プログラム生成についての説明

2023/11/01 中津川ロボカップジュニア

1.概要

NRob のプログラムより Arduino スケッチを生成するソフト。NRobEdit0 と同じ形で作ったプログラムを Arduino スケッチに変え、ArduinoIDE から NRob メインマイコンの ArduinoNano にプログラムを書き込んで実行できるようになる。

メリット:

- ・プログラムの実行が速くなる
- ・長いプログラムを書き込める

デメリット:

- ・手元キーでプログラムの変更が出来なくなる。
- ・元に戻すには、ArduinoIDE から NRob プログラムを書込む必要

2.準備

| | |
|----|---|
| 1. | ArduinoIDE を PC にインストールする |
| 2. | ドキュメントフォルダの Arduino フォルダに[nprg000]をフォルダごとコピーする(プログラムはこれをコピーして生成される) |

3.生成されるフォルダ内のファイル

| | |
|--------------|----------------------------------|
| NRob.cpp | NRob 基板用ライブラリのファイル。 |
| NRob.h | .h は宣言用のヘッダファイル、.cpp はプログラムのファイル |
| NRobDisp.cpp | NRob の LCD 表示関係のライブラリ。 |
| NRobDisp.h | .h は宣言用のヘッダファイル、.cpp はプログラムのファイル |
| sub.ino | 生成するプログラムの補助用関数のファイル |
| [フォルダ名.ino] | 生成されたプログラム本体のメインファイル |

4.生成メインファイルのヘッダ部

| | |
|--|--|
| <pre>#include "NRob.h" #include "NRobDisp.h" //****クラスの宣言 **** NRob3 nrob; NRobDsp ndsp(nrob); //----- プログラム実行部変数 int Tim[3];</pre> | <p>プログラムで使用するファイルの宣言 (Arduino では必要)</p> <p>上記同様に使用するクラス(変数と実行プログラムを合わせたもの)の宣言</p> <p>実行変数は NRob で使っている</p> |
|--|--|

| | |
|--|---|
| int X[10]; | T、X をそれぞれ 3 個、10 個用意 |
| <pre> //**** メイン **** void setup() { nrob.init(50); ndsp.dsp_modeStr(0); prg_init(); } </pre> | <p>初期動作する部分 初期化と初期表示の設定</p> |
| <pre> void loop() { Byte mode=0; mode = ndsp.chg_Mode_main(); switch (mode) { case 1: do_prog(); break; case 2: do_Setting_main(); break; } } </pre> | <p>メニュー切替</p> <p>プログラムの実行か センサーモニターか 切り替える</p> <p>生成したプログラムは do_prog()以下に作られている</p> |

5.各コマンドごとの生成内容

| コマンド | 生成 |
|------|--|
| G | do_Go(①,②,③); 例)G+30+30:10 ⇒ do_Go(+30,+30,10); |
| M | do_Motor(①,②); 例)M+30-30 ⇒ do_Motor(+30,-30); |
| V | do_Servo(①,②); 例)v1:50 ⇒ do_Servo(1,50); |
| L | do_LEDLCD(①,②); 例)L1:01 ⇒ do_LEDLCD(1,01); |
| W | act_Wait(①); 例)w:10 ⇒ act_Wait(10); |
| B | void sub①(void){ 例)B:01 ⇒ void sub01(void){ |
| # | #Tの時 Tim[①]②=③; |

| | |
|---|--|
| | 例) #T1=+00 ⇒ Tim[1]=+00; #Xの時 X[0]=②; または X[0]=①②③; 例) #x1=00+00 ⇒ x[1]= 00; 例) #x2=x3-05 ⇒ x[2]=x[3]-05; |
| J | sub①(); 例) JB:01 ⇒ sub01(); |

その他、+-などのレベルの変化や、サブプログラムの終わりなどを解析し、“}”を生成し適時追加。

6.主な関数

メインファイル(フォルダ名 ino)

| | |
|--------------|------------------------|
| setup() | Arduino の決まりの関数 初期設定用 |
| loop() | Arduino の決まりの関数 実行用 |
| prg_init() | NRobEditI 生成関数 初期設定用 |
| do_prog() | NRobEditI 生成関数 実行部 |
| sub01 ~ 99() | NRobEditI 生成関数 サブプログラム |

sub.ino

| | |
|-------------------|-------------------------|
| do_Setting_main() | センサーモニター 動作メイン |
| moni_AD() | センサーモニター AD 入力モニター |
| moni_Sonar() | センサーモニター 超音波センサー |
| moni_Compas() | センサーモニター コンパスモニター |
| moni_Servo() | センサーモニター サーボ動作テスト |
| do_Action_main() | プログラム実行動作メイン |
| chk_loop() | 基本タイマーとセンサチェック |
| act_Wait(c) | 動作ウエイト c:待ち時間(x100ms) |
| Timer_Count() | タイマーカウント動作 |
| do_Go(a,b,c) | G コマンド動作 a=左/b=右/c=動作時間 |
| do_Motor(a,b) | M コマンド動作 a=左/b=右 |
| do_Servo(n,p) | V コマンド動作 n=サーボ番号/p=位置 |
| do_LEDLCD(n,d) | L コマンド動作 n=番号/d=表示内容 |
| get_AD(n) | S の読込 n=番号 センサーの値などを返す |
| get_SubInTenth(n) | I の読込 n=番号 拡張入力の値を返す |

NRob.h NRob 基板対応用関数など

| | |
|--------|--------------------------------|
| loop() | 基本ループ 50ms 毎正を返す センサー等常時入力しメモリ |
| init() | NRob 基板の初期化 |

| | |
|----------------------|--------------------------|
| get_AD(n) | メモリしたAD値を返す n=番号 |
| in_SW(n) | SW が押されたか n=SW の番号 |
| in_LongSW(n) | SW の長押し確認 |
| key_SW(n) | SW の状態を返す |
| act_motorAB(a,b) | モータの駆動 AB 共 a,b 各モーターパワー |
| act_motorA(a) | モータ駆動 A のみ |
| act_motorB(b) | モータ駆動 B のみ |
| act_ServoDir(n,p) | サーボの駆動 n:番号/p:位置 |
| servo_ON() | サーボの ON のみ |
| servo_OFF() | サーボの OFF のみ |
| get_ServoPos(n) | サーボの現在位置を返す |
| get_US(n) | 超音波センサー読込 n:センサー番号 |
| use_US(n) | 超音波センサー使用宣言 |
| use_SubIn(n) | 入力子機使用 1=あり/0=無し |
| get_SubIn(n) | 入力子機値を返す |
| write_SubOut(n,d) | 出力子機に書き出し |
| chk_Compass() | コンパスの接続を確認 |
| init_Compass() | コンパスモジュール初期化 |
| read_Compass() | コンパス読み込み動作 |
| get_Angle() | コンパス読み込み |
| get_RelAng() | 相対角度読み込み |
| get_OffsetCompas() | コンパスオフセット調整 |
| LED_ON(n)/LED_OFF(n) | LED 点灯/消灯 |
| LED_AlloFF() | LED 全消灯 |
| LCD_clear() | LCD 全消し |
| LCD_print(w) | 現在のカーソルから文字列表示 |
| LCD_write(c) | 現在のカーソルから1文字表示 |
| LCD_setCursor(x,y) | カーソル位置の移動 x:横/y:縦 |
| LCD_Cursor() | カーソルを表示する |
| LCD_nocursor() | カーソル表示を消す |
| LCD_Blink() | カーソルブリンクをする |
| LCD_noblink() | カーソルブリンクを止める |
| dsp_Number2(x,y,n) | 2桁数字表示 x,y:表示位置/n:表示数字 |
| dsp_NumberB(x,y,n) | 1バイト表示(3桁) |
| dsp_Number4(x,y,n) | 4桁数字表示 |
| I2C_Check(ad) | I2C 子機が接続されているかチェック |
| I2C_Send(n) | I2C 送信 n:送信バイト数 |
| I2C_Recv(n) | I2C 受信 n:受信バイト数 |