

File: MrLineTiny3.c, Date: 06/02/2011, Time: 14:53:57

/*

Project : PPE Robot suiveur de ligne "Carte MrLineTiny"
Version : 3
Date : 17/01/2011
Author : Philippe Mariano
Company : LYCEE Pierre Emile MARTIN
Comments: Suivi de ligne

Fichier : MrLineTiny2.c
Cross compilateur: CVAVR V2.04.08a

Chip type : ATtiny26(L)
Clock frequency : 16,000000 MHz
Memory model : Tiny
External SRAM size : 0
Data Stack size : 32

*/

Fichier : MrLineTiny2.c
Cross compilateur: CVAVR V2.04.08a

Chip type : ATtiny26(L)
Clock frequency : 16,000000 MHz
Memory model : Tiny
External SRAM size : 0
Data Stack size : 32

*/

Fichier : MrLineTiny2.c
Cross compilateur: CVAVR V2.04.08a

Chip type : ATtiny26(L)
Clock frequency : 16,000000 MHz
Memory model : Tiny
External SRAM size : 0
Data Stack size : 32

*/

*/

//=====

// Bibliothèques CVAVR

//-----

#include <tiny26.h>

#include <delay.h>

//-----

// Bibliothèque(s) développée(s) pour l'application

//-----

#include "MrLineTiny.h"

#include "RBELEVE.h"

//-----

//=====

// Déclarations

//-----

// Temporisations

//-----

#define t_delay 704 // Permet de régler la période d'échantillonnage (tech) de la position

// du robot par rapport à la ligne tech = 3*(t_IR + t_mux) + t_delay

//-----

//=====

//=====

//-----

// Début du programme principal

//-----

void main(void)

{

//=====

File: MrLineTiny3.c, Date: 06/02/2011, Time: 14:53:57

```
// Variables locales
//-----
// type      nom      Commentaires
//-----
unsigned char Visu_Position;           // Affichage de la position du robot sur 3 delts
unsigned char Position_Robot = HORS_LIGNE;
enum Etat_Commande_Moteur EtatCdeMot = DEPART;
//=====
// Initialisations
//-----
// Input/Output Ports initialization
// Port A initialization
// Func0=Out Func1=Out Func2=Out Func3=Out Func4=Out Func5=Out Func6=In Func7=In
// State0=1 State1=1 State2=1 State3=1 State4=1 State5=1 State6=T State7=T
PORTA=0x3F;
DDRA=0x3F;
//-----
// Port B initialization
// Func0=In Func1=Out Func2=In Func3=Out Func4=In Func5=In Func6=In Func7=In
// State0=T State1=1 State2=T State3=1 State4=T State5=T State6=P State7=T
PORTB=0x0A;
DDRB=0x0A;
//-----
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: 4000,000 kHz (fréquence d'entrée du timer FQ/4)
// Mode: PWMA & B top=OCR1C
// OC1A output: Non-Inv., /OC1A disconnected
// OC1B output: Non-Inv., /OC1B disconnected
PLLCSR=0x00;
TCCR1A=0xA3;
TCCR1B=0x03;
TCNT1=0x00;
OCR1A=0x00;
OCR1B=0x00;
OCR1C=0xFF;
//-----
// Analog Comparator initialization
// Analog Comparator: On
ACSR=0x00;
//-----
// Extinction des LED
PORTA = 0x3F;
//-----
//=====
```

File: MrLineTiny3.c, Date: 06/02/2011, Time: 14:53:57

```
// Boucle principale
//-----
while (1)
    { // début while
//-----
// Lecture de la position du robot par rapport à la ligne
//-----
        Position_Robot = AcquerirPositionRobot();
//-----
// Affichage de la position du robot par rapport à la ligne
//-----
        Visu_Position = AfficherPositionRobot(Position_Robot);
        PORTA = Visu_Position; // Fonction FS61 "Sélectionner les LED de visualisation"
//-----
// Correction de la trajectoire du robot
//-----
/*FS12 : "Choisir les valeurs des rapports cycliques"
//-----
Rôle: Régler le rapport cyclique des signaux de commande des moteurs droit et gauche pour corriger la
trajectoire du robot afin qu'il suive la ligne.

Entrée(s):      Position_Robot      (voir ci-dessus)

Sortie(s):      CDE_ROUE_G          octet      (OCR1A) règle le rapport cyclique du signal de commande
                                                    du moteur associé à la roue Gauche
                CDE_ROUE_D          octet      (OCR1B) règle le rapport cyclique du signal de commande
                                                    du moteur associé à la roue Droite
-----*/
switch (EtatCdeMot)
{ // Début switch (EtatCdeMot)
    case DEPART: Arret_Moteurs();
        if ((Position_Robot == CENTRE_1) || (Position_Robot == CENTRE_2) || (Position_Robot == CENTRE_3))
        {
            delay_ms(3000);
            Active_Timer();
            EtatCdeMot = CDE_CENTRE1;
        }
        else EtatCdeMot = DEPART;
    break;

    case CDE_CENTRE1: EtatCdeMot = CDE_CENTRE3;
    break;

    case CDE_CENTRE2: EtatCdeMot = CDE_CENTRE3;
```

File: MrLineTiny3.c, Date: 06/02/2011, Time: 14:53:57

```
break;

case CDE_CENTRE3: CDE_ROUE_G = _60PC; CDE_ROUE_D = _60PC;
    if (Position_Robot == GAUCHE) EtatCdeMot = CDE_DROITE;
    else if (Position_Robot == DROITE) EtatCdeMot = CDE_GAUCHE;
    else EtatCdeMot = CDE_CENTRE3;

break;

case CDE_DROITE: CDE_ROUE_G = _60PC; CDE_ROUE_D = _50PC;
    if (Position_Robot == CENTRE_3) EtatCdeMot = CDE_CENTRE3;
    else if (Position_Robot == GAUCHE_PLUS) EtatCdeMot = CDE_DROITE_PLUS;
    else EtatCdeMot = CDE_DROITE;

break;

// !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
// A compléter
// La liste des états est dans le fichier RBELEVE.h
// !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

default : EtatCdeMot = CDE_CENTRE3;

} // Fin switch (EtatCdeMot)
//-----
    delay_us(t_delay); // A ajuster en fonction de la vitesse du robot
} // fin while(1);
//-----
} // fin prog princ (main)
//=====
```