#include <msp430.h>

#include "ADC.h"

#include "Afficheur.h"

#include "Action.h"

#define BUTTON BIT3

int appui = 1;

int compteur = 0;

int seconde = 0;

unsigned int capteurDevant=0;

unsigned int capteurDroite=0;

unsigned int capteurGauche=0;

int main( void )

{

// Stop watchdog timer to prevent time out reset

WDTCTL = WDTPW + WDTHOLD;

BCSCTL2 = CALBC1\_1MHZ;

DCOCTL = CALDCO\_1MHZ;

P1DIR &= ~(BIT0 ); //capteur de lumiere droit gauche

P1DIR |= BUTTON;

P2DIR |= (BIT2|BIT1|BIT5|BIT4);

P2DIR &= ~(BIT0|BIT3);

P2SEL |= (BIT2|BIT4|BIT0|BIT3);

P2SEL2 &= ~(BIT2|BIT4|BIT0|BIT3);

P2OUT|=BIT2; //on envoie une tension au moteur droite

P2OUT|=BIT4; //on envoie une tension au moteur gauche

P1IE|=BIT3;//on applique l'interruption sur le bouton P1.3

P1IES|=BIT3;//front descendant de P1.3

P1REN |= BIT3;

P1IFG &= ~(BIT3);

TA1CTL = 0|(TASSEL\_2|ID\_0);

TA1CTL |= MC\_1;

TA1CCTL1 |= OUTMOD\_7;

TA1CCTL2 |= OUTMOD\_7;

TA1CCR0 = 500; // this value to change

TA1CCR1 = 354;

TA1CCR2 = 350;

P2OUT &= ~BIT5; //0

P2OUT |= BIT1; //1

/\*

TACTL = TASSEL\_2 + ID\_3 + MC\_2; //période de 1s

TA0CCTL0 = CCIE; //interruption au bout de 60s

TA0CCR0 = 62500; //TAOCCRO pour 1s

\*/

ADC\_init();

Aff\_Init();

while(1){

ADC\_Demarrer\_conversion(0); //on démarre la conversion

capteurDevant=ADC\_Lire\_resultat(); //on lit le résultat

Aff\_valeur(convert\_Hex\_Dec(capteurDevant));

/\*ADC\_Demarrer\_conversion(0); //on démarre la conversion

capteurDroite=ADC\_Lire\_resultat(); //on lit le résultat

Aff\_valeur(convert\_Hex\_Dec(capteurDroite));\*/

//capteurDroite = (capteurDroite -360)/6;

// ADC\_Demarrer\_conversion(7); //on démarre la conversion

//capteurGauche=ADC\_Lire\_resultat(); //on lit le résultat

// capteurGauche = (capteurGauche -360)/6;

/\* ADC\_Demarrer\_conversion(5); //on démarre la conversion

capteurligneGauche=ADC\_Lire\_resultat(); //on lit le résultat

ADC\_Demarrer\_conversion(4); //on démarre la conversion

capteurligneDroite=ADC\_Lire\_resultat(); //on lit le résultat\*/

if(capteurDevant<420){

Straight();

//\_\_delay\_cycles(200000);

//SpeedUp();

}

else{

Stop();

/\*if(capteurDroite<600){

TurnRight();

Stop();

}

else{

TurnLeft();

Stop();

}\*/

}

/\* ADC\_Demarrer\_conversion(0x01);

mesure = ADC\_Lire\_resultat();

lumiere = (mesure -360)/6;

if (lumiere<20)

Aff\_4carac("nuit");

else Aff\_4carac("jour");

for (i=0; i<40000; i++);\*/

}

}

#pragma vector=PORT1\_VECTOR

\_\_interrupt void Port\_1(void){

if(appui){

appui = 0;

TA1CCTL1 = OUTMOD\_0; //moteurs éteints

TA1CCTL2 = OUTMOD\_0;

}

P1IFG&=~BIT3;

}

#pragma vector=TIMER0\_A0\_VECTOR

\_\_interrupt void ma\_capture(void)

{

if(seconde >= 62){ //condition pour le timer du temps

TACTL = MC\_0;

TA1CCTL1 = OUTMOD\_0; //moteurs éteints

TA1CCTL2 = OUTMOD\_0;

}else{

seconde++; //on incrémente

}

}