#include <stdio.h>

#include "xparameters.h"

#include "platform.h"

#include "xil\_printf.h"

#include "xil\_io.h"

#include "MY\_IP\_CONTADOR\_21\_22.h"

#include "MY\_IP\_FILTRO\_21\_22.h"

#include "MY\_IP\_HALL\_21\_22.h"

#include "MY\_IP\_STEPPER.h"

#include "MY\_IP\_HCSR04.h"

#include "my\_ip\_temperatura.h"

#include "MYIP\_MOTORDC\_DIR.h"

#define MOTORDC\_ADDRESS XPAR\_MYIP\_MOTORDC\_DIR\_0\_S00\_AXI\_BASEADDR

#define CONT\_ADDRESS XPAR\_MY\_IP\_CONTADOR\_21\_22\_0\_S00\_AXI\_BASEADDR

#define FILTRO\_ADDRESS XPAR\_MY\_IP\_FILTRO\_21\_22\_0\_S00\_AXI\_BASEADDR

#define HALL\_ADDRESS XPAR\_MY\_IP\_HALL\_21\_22\_0\_S00\_AXI\_BASEADDR

#define DIST\_ADDRESS XPAR\_MY\_IP\_HCSR04\_0\_S00\_AXI\_BASEADDR

#define STEPPER\_ADDRESS XPAR\_MY\_IP\_STEPPER\_0\_S00\_AXI\_BASEADDR

#define TEMP\_ADDRESS XPAR\_MY\_IP\_TEMPERATURA\_0\_S00\_AXI\_BASEADDR

#define REG0 0

#define REG1 4

#define REG2 8

#define REG3 12

#define REG4 16

int main()

{

init\_platform();

float valor\_milimetros=0;

int temp=0;

print("Hello World arriquitauntantaun\n\r");

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG0, 1);

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG0, 0);

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG2, 200);

MY\_IP\_HALL\_21\_22\_mWriteReg(HALL\_ADDRESS, REG0, 1);

MY\_IP\_HALL\_21\_22\_mWriteReg(HALL\_ADDRESS, REG0, 0);

MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG1,0);

while(1)

{

valor\_milimetros = MY\_IP\_HCSR04\_mReadReg(DIST\_ADDRESS, REG1);

xil\_printf("Distancia = %d", valor\_milimetros);

temp = MY\_IP\_TEMPERATURA\_mReadReg(TEMP\_ADDRESS, REG1);

xil\_printf("Temp = %d", temp);

if(valor\_milimetros < 100){

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG4, 0);

}

else{

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG4, 1);

}

if(temp < 30){

MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG2, 0);

}

else {

MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG2, 1);

}

for(int i=0; i<100; i+=10){

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG1, i);

xil\_printf("%d\n",i);

usleep(1000000);

}

}

cleanup\_platform();

return 0;

}

#include <stdio.h>

#include "xparameters.h"

#include "platform.h"

#include "xil\_printf.h"

#include "xil\_io.h"

#include "MY\_IP\_CONTADOR\_21\_22.h"

#include "MY\_IP\_FILTRO\_21\_22.h"

#include "MY\_IP\_HALL\_21\_22.h"

#include "MY\_IP\_STEPPER.h"

#include "MY\_IP\_HC\_SR04.h"

#include "my\_ip\_temperatura.h"

#include "MYIP\_MOTORDC\_DIR.h"

#define MOTORDC\_ADDRESS XPAR\_MYIP\_MOTORDC\_DIR\_0\_S00\_AXI\_BASEADDR

#define CONT\_ADDRESS XPAR\_MY\_IP\_CONTADOR\_21\_22\_0\_S00\_AXI\_BASEADDR

#define FILTRO\_ADDRESS XPAR\_MY\_IP\_FILTRO\_21\_22\_0\_S00\_AXI\_BASEADDR

#define HALL\_ADDRESS XPAR\_MY\_IP\_HALL\_21\_22\_0\_S00\_AXI\_BASEADDR

#define DIST\_ADDRESS XPAR\_MY\_IP\_HC\_SR04\_0\_S00\_AXI\_BASEADDR

#define STEPPER\_ADDRESS XPAR\_MY\_IP\_STEPPER\_0\_S00\_AXI\_BASEADDR

#define TEMP\_ADDRESS XPAR\_MY\_IP\_TEMPERATURA\_0\_S00\_AXI\_BASEADDR

#define REG0 0

#define REG1 4

#define REG2 8

#define REG3 12

#define REG4 16

int main()

{

int temp\_entera=0, temp\_decimal=0;

float temp=0;

float temp\_decimal2=0;

print("porongo\n\r");

while(1)

{

temp\_entera=MY\_IP\_TEMPERATURA\_mReadReg(TEMP\_ADDRESS, REG1);

xil\_printf("la parte entera es %d \n", temp\_entera);

temp\_decimal2=0.0;

temp+=temp\_entera;

temp\_decimal=MY\_IP\_TEMPERATURA\_mReadReg(TEMP\_ADDRESS, REG2);

xil\_printf("la parte decimal es %d \n", temp\_decimal);

temp\_decimal2+=(temp\_decimal & 0x00000008)\*0.5/8;

temp\_decimal2+=(temp\_decimal & 0x00000004)\*0.25/4;

temp\_decimal2+=(temp\_decimal & 0x00000002)\*0.125/2;

temp\_decimal2+=(temp\_decimal & 0x00000001)\*0.0625/1;

// xil\_printf("temp entera %d \n", temp\_entera);

printf("temp decimal %f \n\n", temp\_decimal2+temp\_entera);

// xil\_printf("la parte decimal es %d \n", MY\_IP\_TEMPERATURA\_mReadReg(TEMP\_ADDRESS, 4));

usleep(1000000);

}

cleanup\_platform();

return 0;

}

//

// int main()

//{

// init\_platform();

//

// print("Hello World arriquitauntantaun\n\r");

//

// while(1)

// {

// MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG1,1);

// MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG3, 100);

// MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG2, 0);

// }

// cleanup\_platform();

// return 0;

//}

//

//int main()

//{

// u32 valor\_microsegundos=0;

// float valor\_milimetros=0;

//

// MY\_IP\_HCSR04\_mWriteReg(DIST\_ADDRESS, REG0, 1);

// MY\_IP\_HCSR04\_mWriteReg(DIST\_ADDRESS, REG0, 0);

//

// while (1)

// {

// valor\_microsegundos= MY\_IP\_HCSR04\_mReadReg(DIST\_ADDRESS, REG1);

// printf("valor microsegundos: %d\n", valor\_microsegundos);

// valor\_milimetros=valor\_microsegundos/5.8;

// printf("valor en milimetros: %f\n", valor\_milimetros);

// usleep(300000);

// }

//

//

// cleanup\_platform();

// return 0;

//}

//int main()

//{

// init\_platform();

//

// print("Hello World arriquitauntantaun\n\r");

//

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG0, 1);

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG0, 0);

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG2, 200);

//

// MY\_IP\_HALL\_21\_22\_mWriteReg(HALL\_ADDRESS, REG0, 1);

// MY\_IP\_HALL\_21\_22\_mWriteReg(HALL\_ADDRESS, REG0, 0);

//

//

// while(1)

// {

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG4, 0);

// for(int i=0; i<100; i+=10){

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG1, i);

// xil\_printf("%d\n",i);

// usleep(1000000);

// }

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG4, 1);

// for(int i=0; i<100; i+=10){

// MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG1, i);

// xil\_printf("%d\n",i);

// usleep(1000000);

// }

//

// usleep(1000000);

//

// }

// cleanup\_platform();

// return 0;

//}

CODIGO

**int** **main**()

{

init\_platform();

**float** dist\_mm;

**int** temp=0,dist,giro;

**printf**("Hello World\n\r");

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG0, 1);

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG0, 0);

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG2, 200);

MY\_IP\_HALL\_21\_22\_mWriteReg(HALL\_ADDRESS, REG0, 1);

MY\_IP\_HALL\_21\_22\_mWriteReg(HALL\_ADDRESS, REG0, 0);

MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG0,0);

**while**(1){

dist = MY\_IP\_HC\_SR04\_mReadReg(DIST\_ADDRESS, REG1);

xil\_printf("Distancia: %d \n\r", dist);

**if**(dist>1500){

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG1, 30);

MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG2,0);

giro=0;

}

**if**(dist<600){

MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG1, 80);

MY\_IP\_STEPPER\_mWriteReg(STEPPER\_ADDRESS, REG2,1);

giro=1;

}

xil\_printf("Sentido giro %d \n\r", giro);

sleep(1);

}

//MYIP\_MOTORDC\_DIR\_mWriteReg(MOTORDC\_ADDRESS, REG1, 50);

cleanup\_platform();

**return** 0;

}