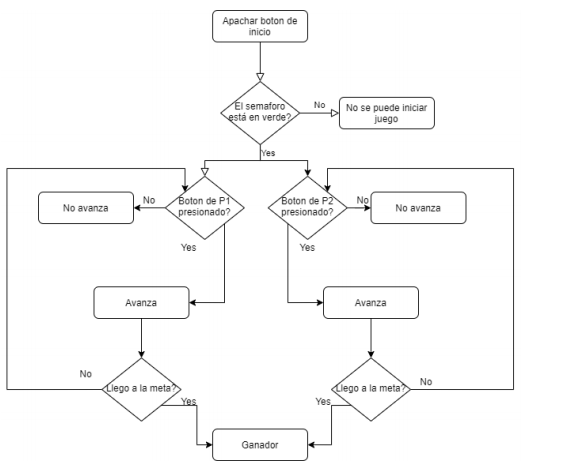
Laboratorio No. 4

Juego de carrera Tiva C

**Link de GitHub:** <https://github.com/mon19379/DIGITAL2.git>

**Link Video:** <https://youtu.be/jzLWDiLGYb4>

**Pseudocódigo:**

****

**Código:**

#include <stdint.h>

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//variables

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

uint8\_t PL1;

uint8\_t PL2;

uint8\_t SEM;

uint8\_t CONT;

uint8\_t CONT1;

uint8\_t BO1;

uint8\_t BO2;

uint8\_t BO3;

uint8\_t FLAG;

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//prototipos de funciones

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void setup (void);

void semaforo(void);

void player1 (void);

void player2 (void);

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Configuración

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void setup() {

// put your setup code here, to run once:

//LEDS J1

pinMode(PB\_5, OUTPUT);

pinMode(PB\_0, OUTPUT);

pinMode(PB\_1, OUTPUT);

pinMode(PE\_4, OUTPUT);

pinMode(PE\_5, OUTPUT);

pinMode(PB\_4, OUTPUT);

pinMode(PA\_5, OUTPUT);

pinMode(PA\_6, OUTPUT);

//LEDS J2

pinMode(PD\_0, OUTPUT);

pinMode(PD\_1, OUTPUT);

pinMode(PD\_2, OUTPUT);

pinMode(PD\_3, OUTPUT);

pinMode(PE\_1, OUTPUT);

pinMode(PE\_2, OUTPUT);

pinMode(PE\_3, OUTPUT);

pinMode(PA\_7, OUTPUT);

//SEMAFORO

pinMode(PF\_3, OUTPUT);

pinMode(PB\_3, OUTPUT);

pinMode(PC\_4, OUTPUT);

//BOTONES

pinMode(PB\_2, INPUT\_PULLUP);

pinMode(PE\_0, INPUT\_PULLUP);

pinMode(PF\_0, INPUT\_PULLUP);

//INDICADOR

pinMode(RED\_LED, OUTPUT);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Ciclo principal

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void loop() {

// put your main code here, to run repeatedly:

BO1 = digitalRead(PB\_2);

BO2 = digitalRead(PE\_0);

BO3 = digitalRead(PF\_0);

player1();

player2();

if (BO1 == HIGH) {

SEM = 0;

}

else {

if (BO1 == LOW) {

SEM = 1;

semaforo();

}

}

if (FLAG == 1) {

if (BO2 == LOW) {

PL1 = 1;

}

if (PL1 == 1 && BO2 == HIGH) {

PL1 = 0;

CONT ++;

}

}

if (FLAG == 1) {

if (BO3 == LOW) {

PL2 = 1;

}

if (PL2 == 1 && BO3 == HIGH) {

PL2 = 0;

CONT1 ++;

}

}

delay(100);

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

//Subrutinas

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

void semaforo (void) {

digitalWrite(RED\_LED, LOW);

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

CONT = 0;

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

CONT1 = 0;

digitalWrite (PC\_4, HIGH);

digitalWrite (PB\_3, LOW);

digitalWrite (PF\_3, LOW);

delay(500);

digitalWrite (PC\_4, LOW);

digitalWrite (PB\_3, HIGH);

digitalWrite (PF\_3, LOW);

delay(500);

digitalWrite (PC\_4, LOW);

digitalWrite (PB\_3, LOW);

digitalWrite (PF\_3, HIGH);

FLAG = 1;

delay(500);

digitalWrite (PC\_4, LOW);

digitalWrite (PB\_3, LOW);

digitalWrite (PF\_3, LOW);

}

void player1 (void) {

if (CONT == 1) {

digitalWrite(PB\_5, HIGH);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 2) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, HIGH);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 3) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, HIGH);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 4) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, HIGH);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 5) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, HIGH);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 6) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, HIGH);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 7) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, HIGH);

digitalWrite(PA\_6, LOW);

}

else if (CONT == 8) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, HIGH);

}

else if (CONT == 9) {

digitalWrite(PB\_5, LOW);

digitalWrite(PB\_0, LOW);

digitalWrite(PB\_1, LOW);

digitalWrite(PE\_4, LOW);

digitalWrite(PE\_5, LOW);

digitalWrite(PB\_4, LOW);

digitalWrite(PA\_5, LOW);

digitalWrite(PA\_6, LOW);

digitalWrite(RED\_LED, HIGH);

FLAG = 0;

CONT = 0;

}

}

void player2 (void) {

if (CONT1 == 1) {

digitalWrite(PD\_0, HIGH);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 2) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, HIGH);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 3) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, HIGH);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 4) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, HIGH);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 5) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, HIGH);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 6) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, HIGH);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 7) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, HIGH);

digitalWrite(PA\_7, LOW);

}

else if (CONT1 == 8) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, HIGH);

}

else if (CONT1 == 9) {

digitalWrite(PD\_0, LOW);

digitalWrite(PD\_1, LOW);

digitalWrite(PD\_2, LOW);

digitalWrite(PD\_3, LOW);

digitalWrite(PE\_1, LOW);

digitalWrite(PE\_2, LOW);

digitalWrite(PE\_3, LOW);

digitalWrite(PA\_7, LOW);

digitalWrite(RED\_LED, HIGH);

FLAG = 0;

CONT1 = 0;

}

}