

# Adaptació de kits d'aprenentatge de circuits electrònics a l'ensenyament d'enginyeria

**Nom:** Vicenç Nebot Parra

**Grau:** Enginyeria de Sistemes de Telecomunicació

**Any acadèmic:** 2024 - 2025

**Director:** Francesc Josep Robert Sanxis



Escola d'Enginyeria de Telecomunicació i  
Aeroespacial de Castelldefels

UNIVERSITAT POLITÈCNICA DE CATALUNYA BARCELONATECH

# ÍNDEX

---

- |                                 |                                       |
|---------------------------------|---------------------------------------|
| 1. Introducció                  | 5. Mode de radiocontrol per Bluetooth |
| 2. Justificació i Objectius     | 6. Sistema complet en funcionament    |
| 3. Metodologia                  | 7. Conclusions i possibles millores   |
| a. Microcontrolador escollit    | 8. Bibliografia                       |
| b. Funcionalitats               |                                       |
| 4. Robot amb evasió d'obstacles |                                       |
| a. Sensor frontal               |                                       |
| b. Sensor Panoràmic             |                                       |
| c. Decisió de moviments         |                                       |

# INTRODUCCIÓ

---



Es pot concebre com un projecte més de DigSys?

# JUSTIFICACIÓ I OBJECTIUS

---

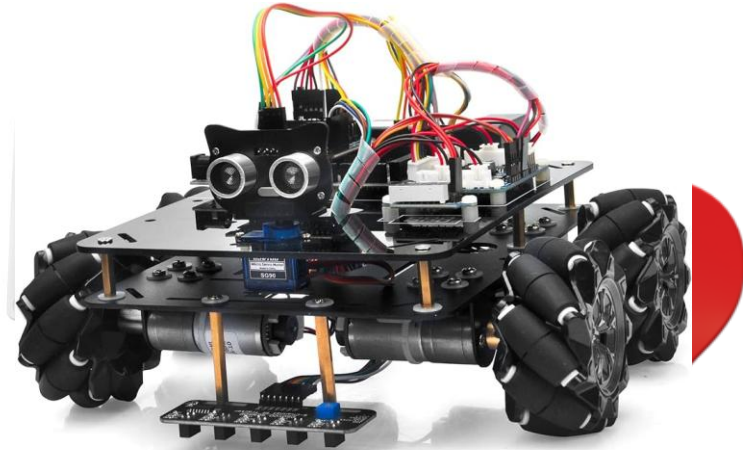


Modular. Progressiu. Accessible. Escalable.

# METODOLOGIA

---

1. Especificacions
2. Planificació
3. Desenvolupament i test
4. Prototipatge



# METODOLOGIA | Microcontrolador escollit

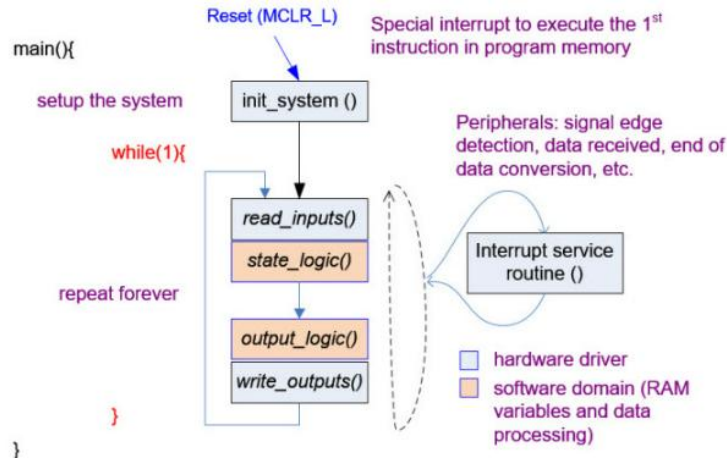
- Arduino UNO o Arduino Mega 2560?

|                                  | Arduino UNO                        | Arduino Mega 2560                  |
|----------------------------------|------------------------------------|------------------------------------|
| <b>Dimension</b>                 | 2.7 in x 2.1 in                    | 4 in x 2.1 in                      |
| <b>Processor</b>                 | Atmega328P                         | ATmega2560                         |
| <b>Clock Speed</b>               | 16MHz                              | 16MHz                              |
| <b>Flash Memory (kB)</b>         | 32                                 | 256                                |
| <b>EEPROM (kB)</b>               | 1                                  | 4                                  |
| <b>SRAM (kB)</b>                 | 2                                  | 8                                  |
| <b>Voltage Level</b>             | 5V                                 | 5V                                 |
| <b>Digital I/O Pins</b>          | 14                                 | 54                                 |
| <b>Digital I/O with PWM Pins</b> | 6                                  | 15                                 |
| <b>Analog Pins</b>               | 6                                  | 16                                 |
| <b>USB Connectivity</b>          | Standard A/B USB                   | Standard A/B USB                   |
| <b>Shield Compatibility</b>      | Yes                                | Yes                                |
| <b>Ethernet/Wi-Fi/Bluetooth</b>  | No (a Shield/module can enable it) | No (a Shield/module can enable it) |

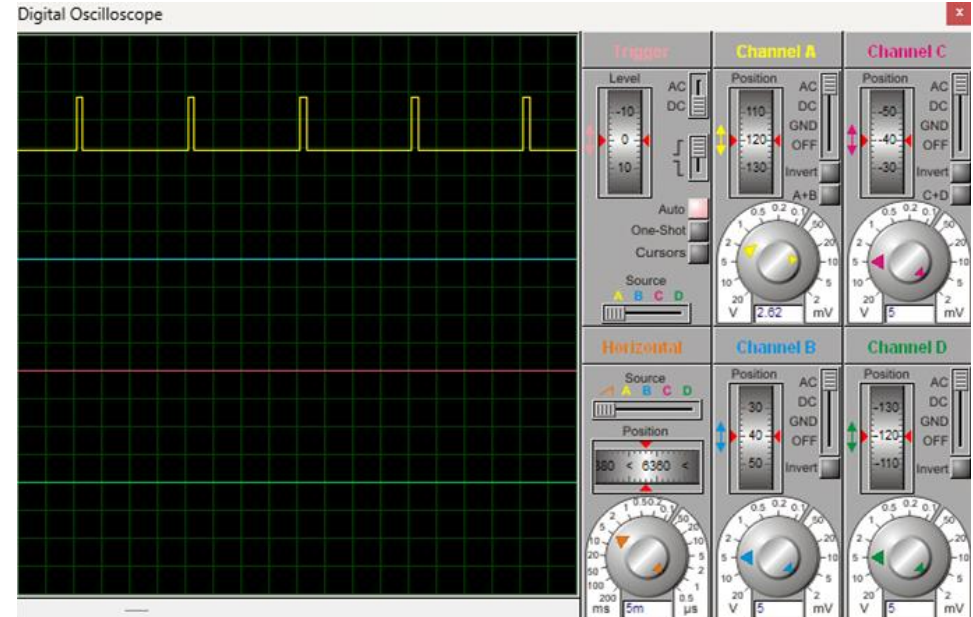
Taula extreta de la pàgina web *Arrow Electronics*, referenciada a la bibliografia

## METODOLOGIA | Funcionalitats

1. Moviment omnidireccional.
2. Evasió d'obstacles.
3. Control remot.
4. Seguiment d'una línia.



- Sense explicaciones:



# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor frontal

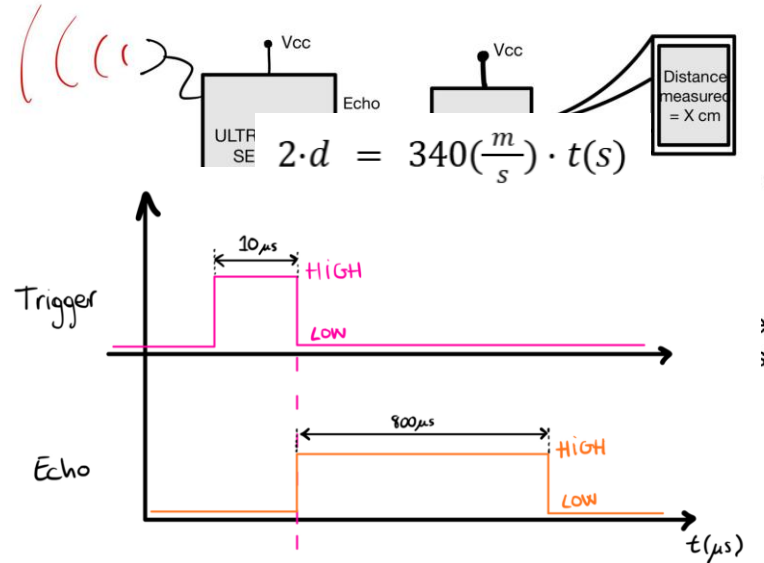
## 1. Especificacions: variable de mostreig, Start/Stop i Toggle LED

$\text{var\_SP\_flag} = 0 \rightarrow$  No ha passat el període de mostreig, no es pren mesures.

$\text{var\_SP\_flag} = 1 \rightarrow$  Ha passat el període de mostreig, ordre de prendre mesures.

$\text{var\_ST\_flag} = 0 \rightarrow$  Es manté a l'estat Idle

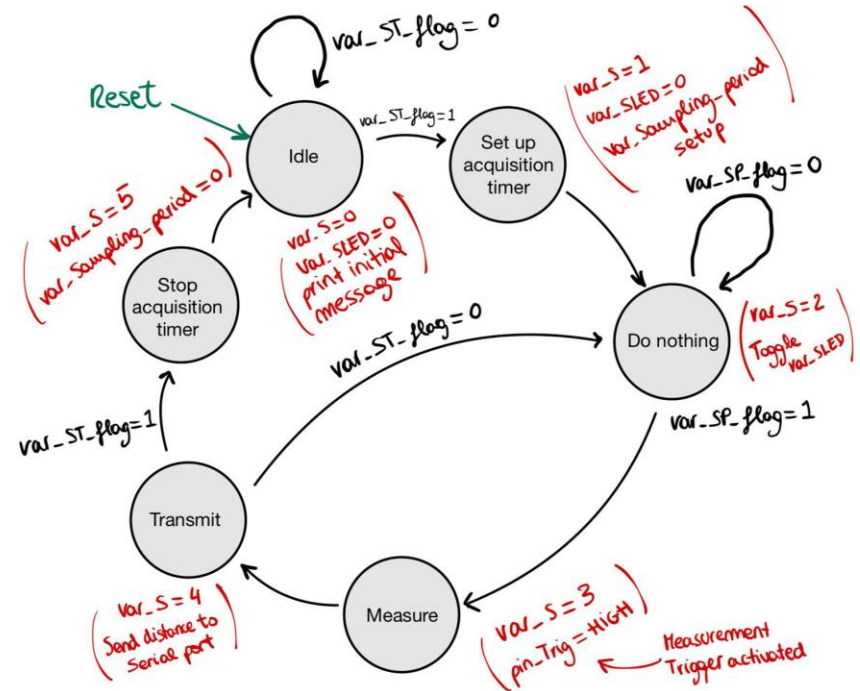
$\text{var\_ST\_flag} = 1 \rightarrow$  Comença a recórrer els estats





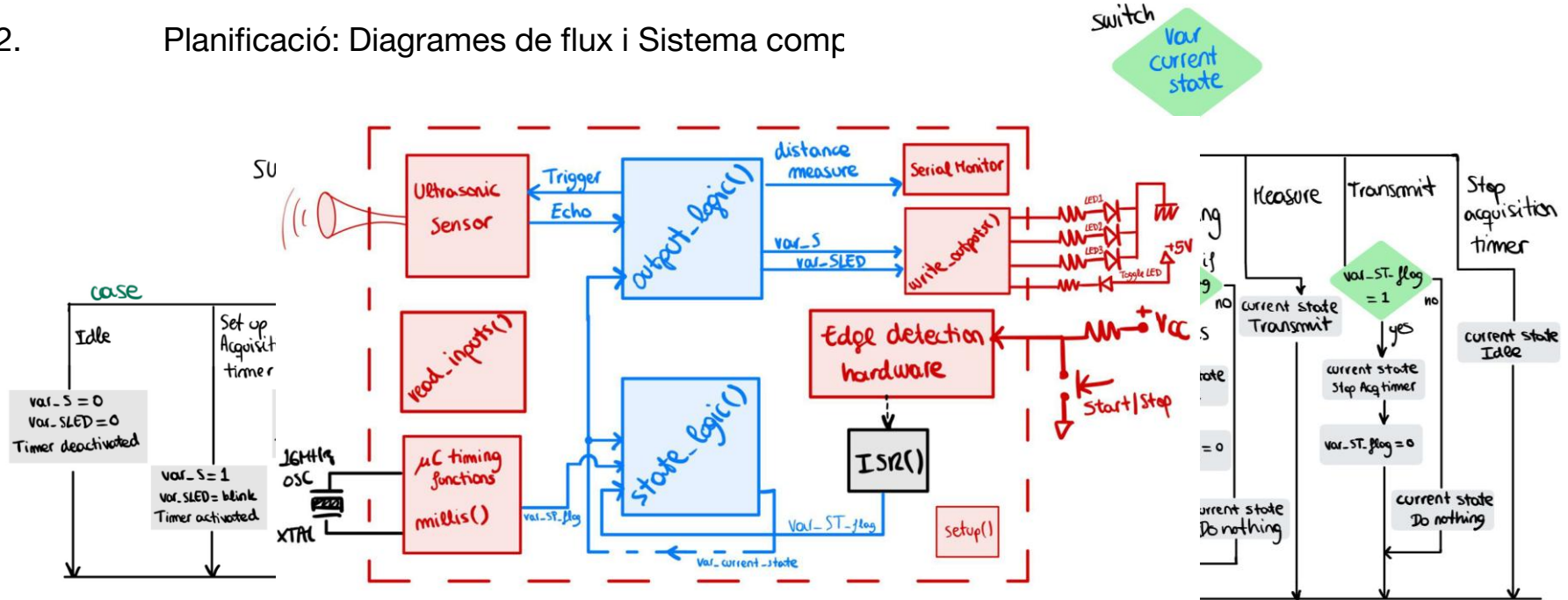
# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor frontal

## 2. Planificació: FSM i diagrama d'estats



# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor frontal

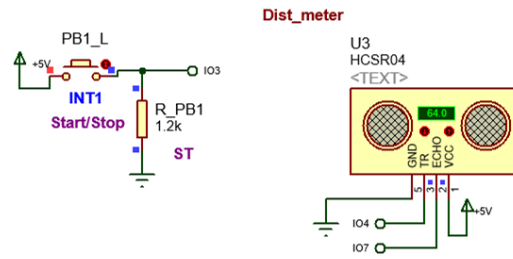
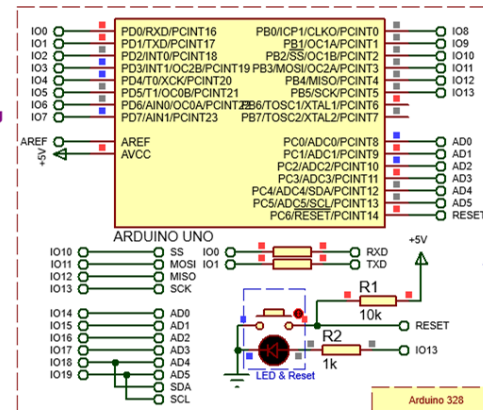
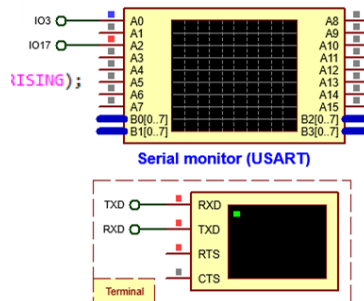
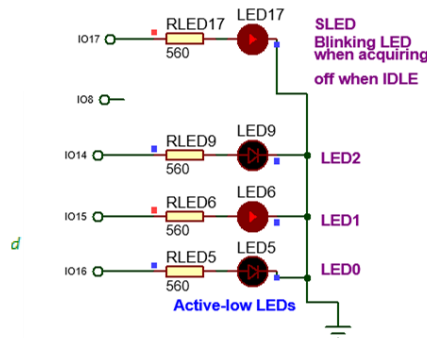
## 2. Planificació: Diagrames de flux i Sistema comp



# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor frontal

## 3. Desenvolupament i test

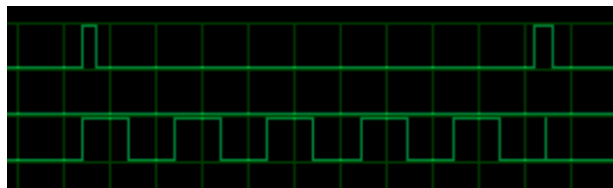
```
void update_time_flag(void){  
    // From internal CLK:  
    //-----  
    //-----  
    int watch(){  
        long echo_distance=0;  
        digitalWrite(pin_Trig,LOW);  
        delayMicroseconds(5);  
        digitalWrite(pin_Trig,HIGH);  
        delayMicroseconds(15);  
        digitalWrite(pin_Trig,LOW);  
        echo_distance=pulseIn(pin_Echo,HIGH);  
        echo_distance=echo_distance*0.01657; //how far away is the object in cm  
        //Serial.println((int)echo_distance);  
        return round(echo_distance);  
    }  
    //-----  
    //-----  
    var_ST_flag = 1;  
}
```



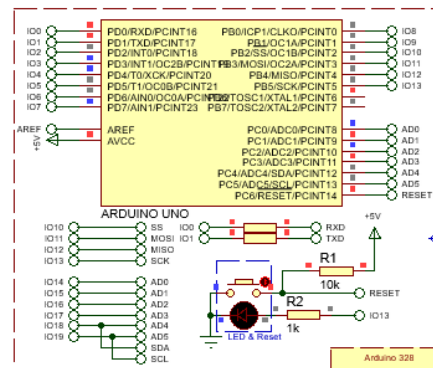
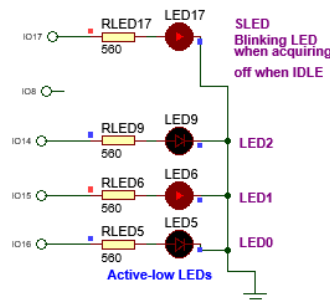
# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor frontal

## 3. Desenvolupament i test: En funcionament

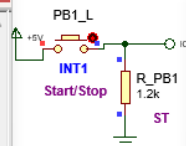
*var\_ST\_flag*



*var\_SLED*



```
Virtual Terminal
Project LABS - DEE
Distance calculator, acquiring data from servo
System IDLE: waiting for ST click to start
Distance measured = 61 cm
Distance measured = 61 cm
Distance measured = 61 cm
Distance measured = 61 cm
```

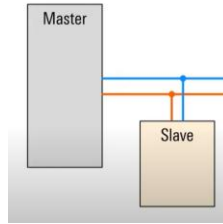


| Dist. meter        |         |       |       |  |
|--------------------|---------|-------|-------|--|
| Watch Window       |         |       |       |  |
| Name               | Address | Type  | Value |  |
| distance_M008001AA |         | word  | 61    |  |
| var_sampli008001AC |         | word  | 500   |  |
| var_time_e008001AE |         | dword | 493   |  |
| var_curr008001B2   |         | dword | 24532 |  |
| var_prev008001B6   |         | dword | 24039 |  |
| var_USART_00000000 |         | byte  | 0xCF  |  |
| var_ST_fla008001BA |         | byte  | '\0'  |  |
| var_SP_fla008001BB |         | byte  | '\0'  |  |
| var_curr008001BC   |         | byte  | 'C'   |  |
| var_SLED_008001BD  |         | byte  | 0x01  |  |
| var_S_008001BE     |         | byte  | 0x02  |  |

Arduino test

# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor panoràmic

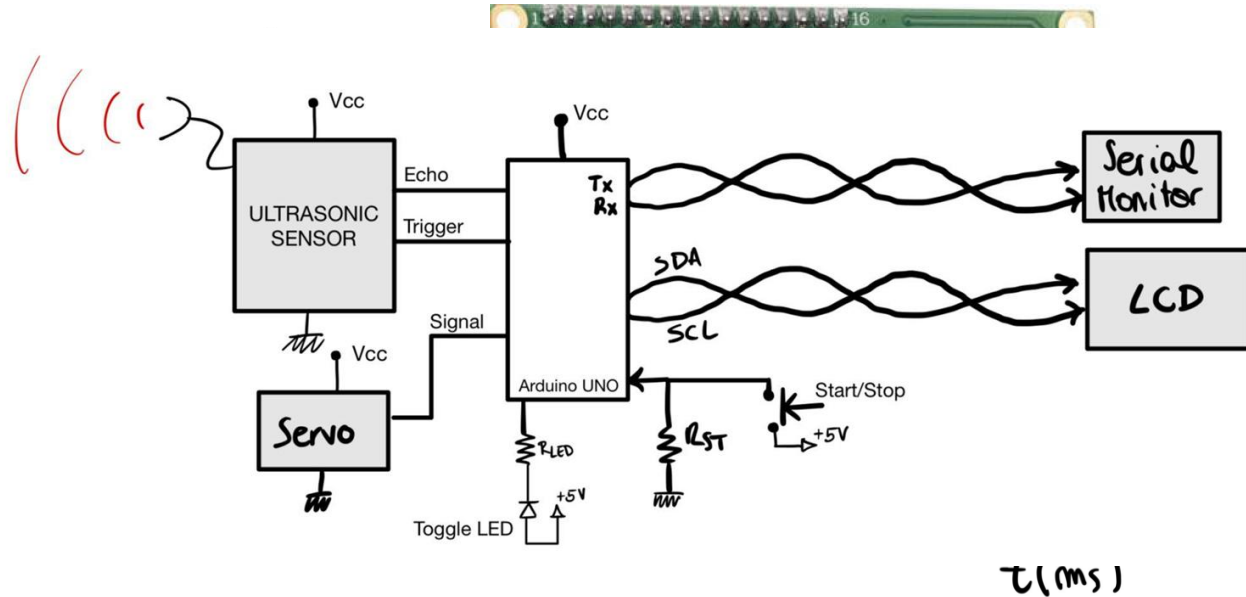
## 1. Especificacions: Elements i variables que afegim



`var_Servo_flag = 0`  
servo.

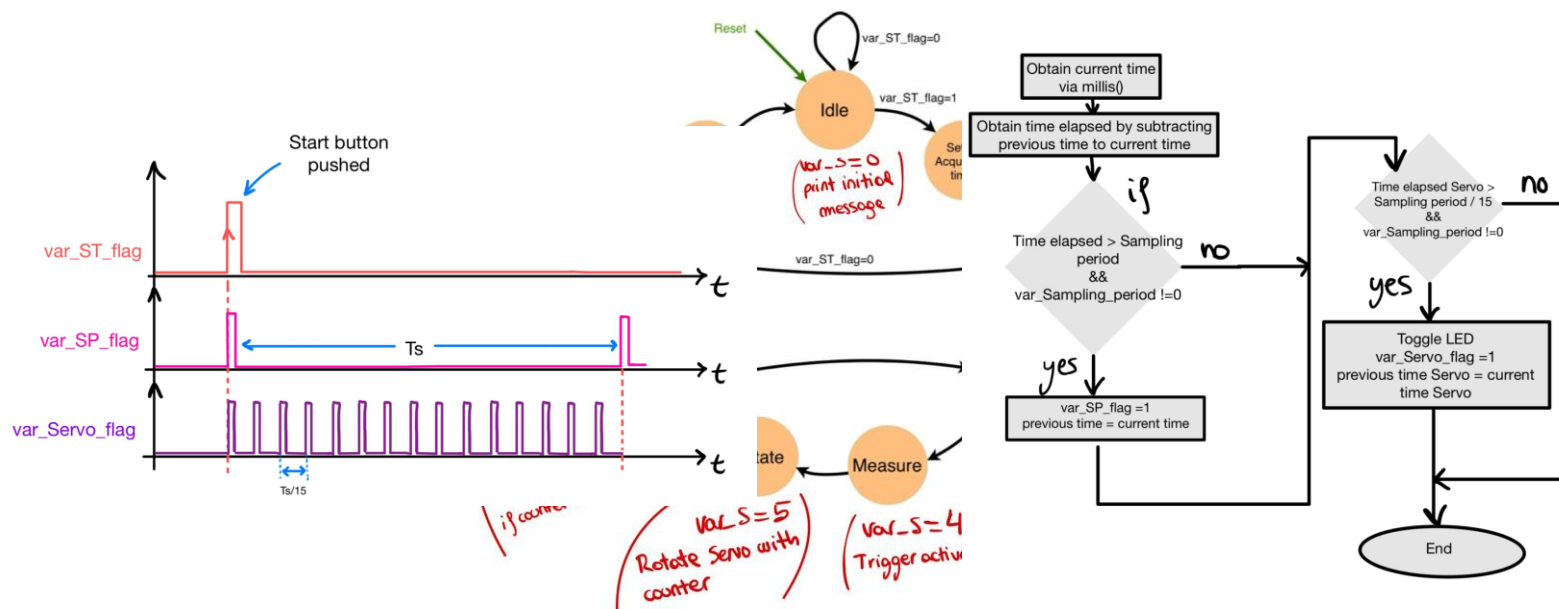
`var_Servo_flag = 1`

variable *counter*



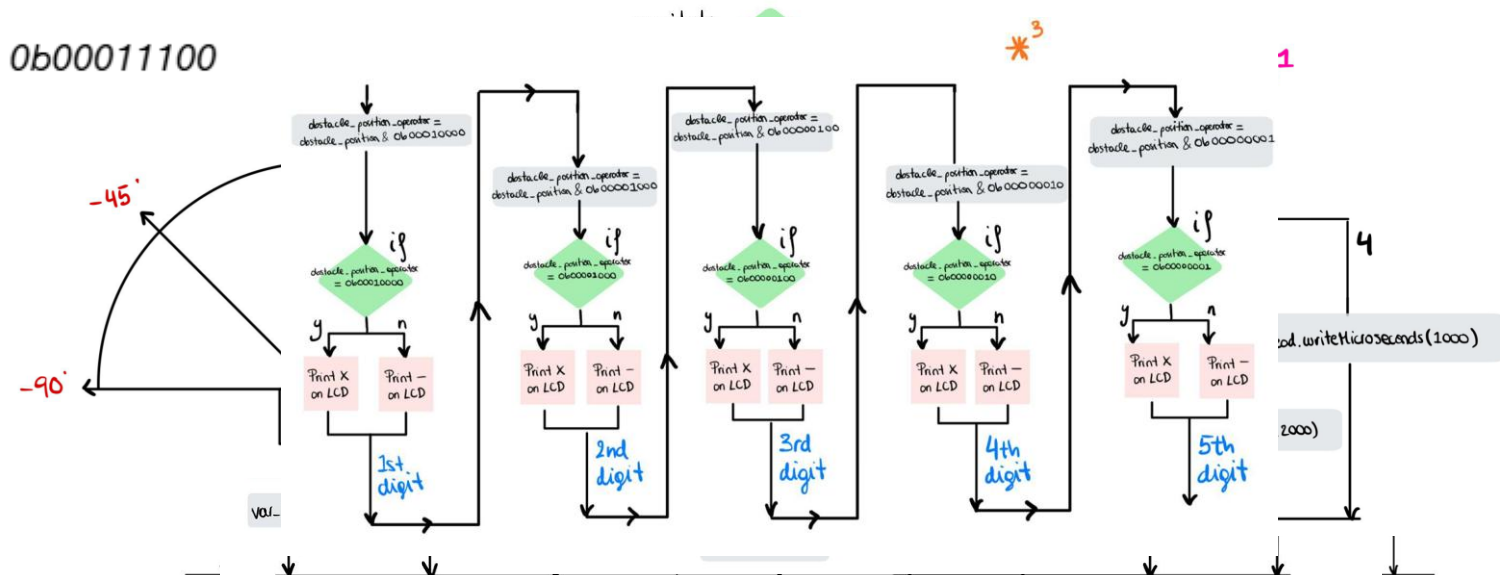
# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor panoràmic

## 2. Planificació: Diagrama d'estats actualitzat i funció de `var_Servo_flag`



# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor panoràmic

## 2. Planificació: Diagrames de flux de les funcions *state\_logic()* i *output\_logic()*



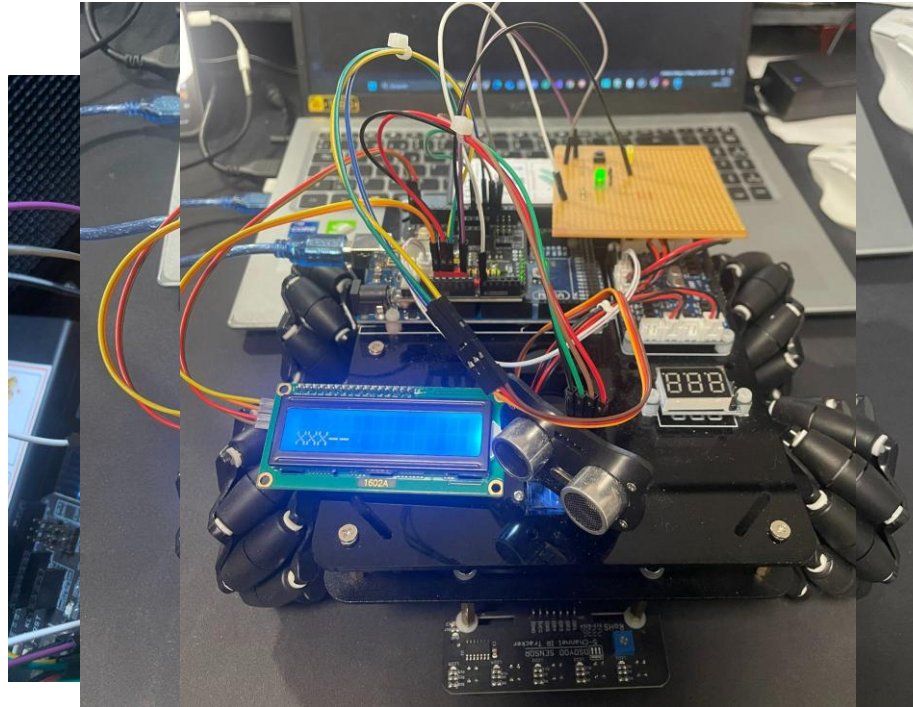






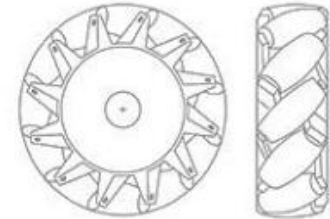
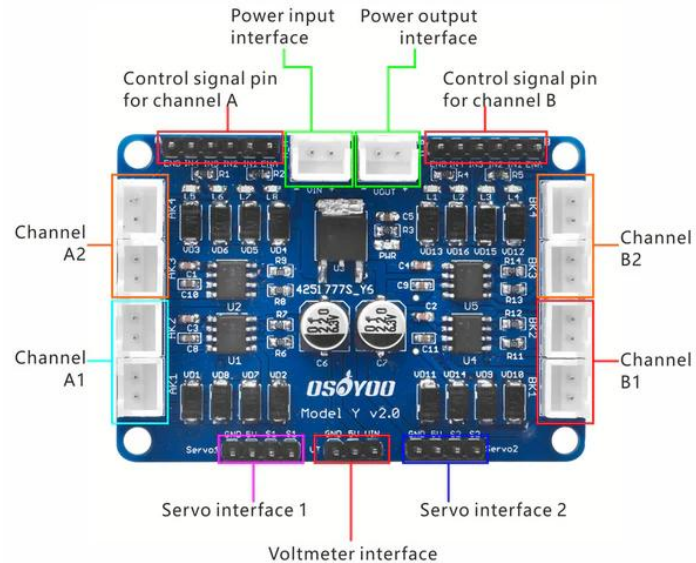
# ROBOT AMB EVASIÓ D'OBSTACLES | Sensor panoràmic

## 4. Prototipatge



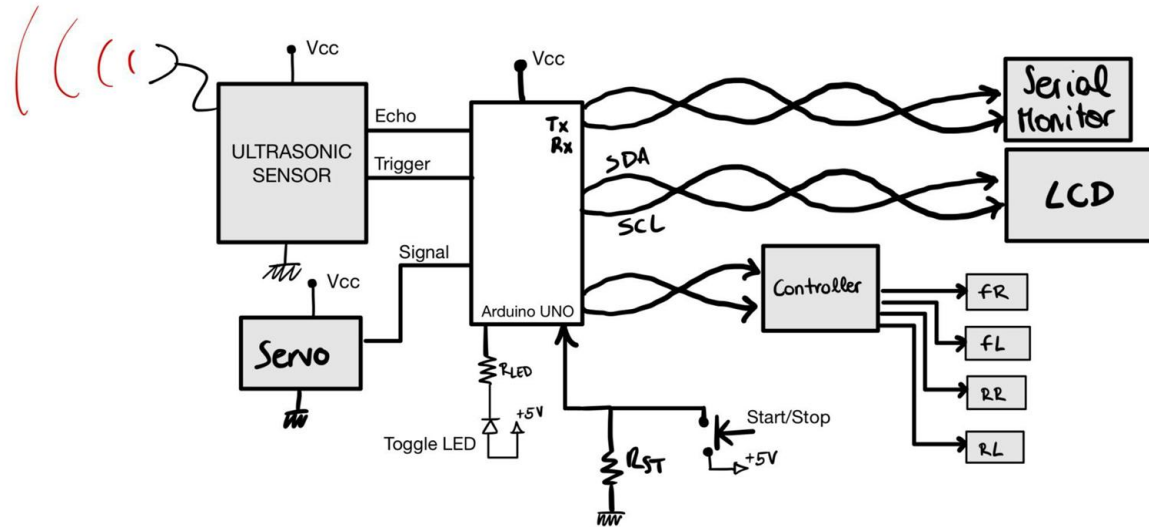
# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

## 1. Especificacions: Elements i variables que afegim



# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

## 1. Especificacions: Esquema general

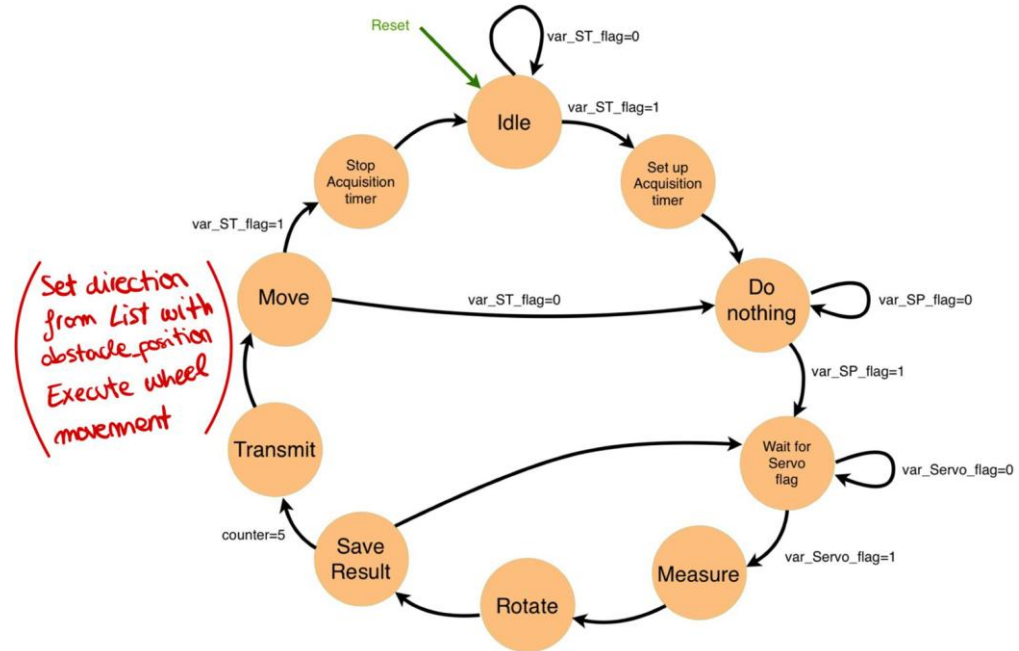


# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

## 2. Planificació: Nou diagrama d'estats i funcions de moviment

- *turnRight()*: S'activen les rodes FL i RL cap endavant.
- *turnLeft()*: S'activen les rodes FR i RR cap endavant.
- *moveForward()*: S'activen les quatre rodes cap endavant.
- *stopMotors()*: Es posa velocitat 0 a totes les rodes.

$$d_{TH} v_{TH} = 0.5 \frac{m}{s} t(s)$$



# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

## 2. Planificació: Velocitat màxima i direccions en funció de la variable *obstacle\_position*

| Motor JGA25-370-35.5KG |              |
|------------------------|--------------|
| Rated Voltage          | DC 6V        |
| Rated Voltage          | 170 ±10% RPM |
| No-load Current        | 0.1 A        |
| Rated Current          | 0.5 A        |
| Rated Speed            | 131 ±10% RPM |
| Rated Torque           | 0.6 Kg·cm    |
| Stall Torque           | 3.3 Kg·cm    |
| Stall Current          | 2.6 A        |

$$v_{TH} = 0.5 \frac{m}{s}$$

$$v_{max} = \frac{144 \text{ rpm} \cdot 2\pi r}{60 s} \simeq 0.6 \frac{m}{s}$$

$$v_{max} > v_{TH}$$



# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

## 3. Desenvolupament i test: Software

```
case MOVE_CAR:  
    var_S = 8;
```

```
    nextMove(obstacle_position, 125);
```

```
    break;
```

```
void nextMove(uint8_t obstacle_position, int speed) {  
    MOVEMENTS m = decision_table[obstacle_position];  
    switch(m) {  
        case FORWARD:  
            moveForward(speed);  
            delay(1000);  
            stopMotors();  
            break;  
  
        case BACKWARD:  
            turnLeft(speed);  
            delay(2000);  
            stopMotors();  
            moveForward(speed);  
            delay(1000);  
            stopMotors();  
            break;  
  
        case LEFT:  
            turnLeft(speed);  
            delay(1000);  
            stopMotors();  
            moveForward(speed);  
            delay(1000);  
            stopMotors();  
            break;
```

```
//WHEEL MOVEMENT DECISION=====
```

```
//=====
```

```
enum MOVEMENTS {  
    FORWARD, BACKWARD, LEFT, LEFTD, RIGHT, RIGHTD, STOP  
};
```

```
const MOVEMENTS decision_table[32] = {  
  
    FORWARD, // 0b00000000  
    LEFTD,   // 0b00000001  
    LEFTD,   // 0b00000010  
    LEFTD,   // 0b00000011  
    LEFT,    // 0b00000100  
    LEFT,    // 0b00000101  
    LEFT,    // 0b00000110  
    LEFT,    // 0b00000111  
    RIGHTD,  // 0b00001000  
    LEFT,    // 0b00001001  
    LEFT,    // 0b00001010  
    LEFT,    // 0b00001011  
    RIGHT,   // 0b00001100  
    LEFT,    // 0b00001101  
    LEFT,    // 0b00001110  
    LEFT,    // 0b00001111  
    RIGHTD,  // 0b00010000  
    FORWARD, // 0b00010001  
    RIGHT,   // 0b00010010  
    LEFTD,   // 0b00010011  
    RIGHT,   // 0b00010100  
    BACKWARD, // 0b00010101  
    RIGHT,   // 0b00010110  
    BACKWARD, // 0b00010111  
    RIGHTD,  // 0b00011000  
    RIGHTD,  // 0b00011001  
    LEFT,    // 0b00011010  
    BACKWARD, // 0b00011011  
    RIGHT,   // 0b00011100  
    BACKWARD, // 0b00011101  
    RIGHT,   // 0b00011110  
    BACKWARD // 0b00011111  
};
```

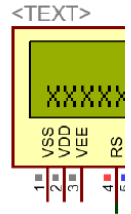
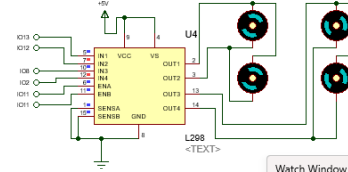
```
void moveForward(int speed){  
    RL_fwd(speed);  
    RR_fwd(speed);  
    FR_fwd(speed);  
    FL_fwd(speed);  
}
```

```
void FR_fwd(int speed) //front-right wheel forward turn  
{  
    digitalWrite(directionPinFR1,LOW);  
    digitalWrite(directionPinFR2,HIGH);  
    analogWrite(speedPinFR,speed);  
}
```



# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

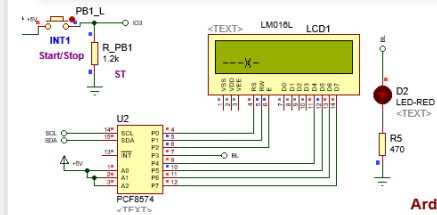
## 3. Desenvolupament i test: En funcionament



```
Virtual Terminal
Buscando dispositivos I2C...
Dispositivo encontrado en 0x27
Project LAB5 - DEE
Distance calculator, acquiring data from servo
System IDLE: waiting for ST click to start
: Ok. LCD found.
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
The nearest obstacle was at 49 cm
GO RIGHT DIAGONAL
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
No obstacle was found
MOVE FORWARD
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
The nearest obstacle was at 46 cm
MOVE BACKWARDS
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
No obstacle was found
MOVE FORWARD
```

```
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
The nearest obstacle was at 49 cm
GO RIGHT DIAGONAL
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
No obstacle was found
MOVE FORWARD
Counter: 0
Counter: 1
Counter: 2
Counter: 3
Counter: 4
The nearest obstacle was at 46 cm
GO LEFT
```

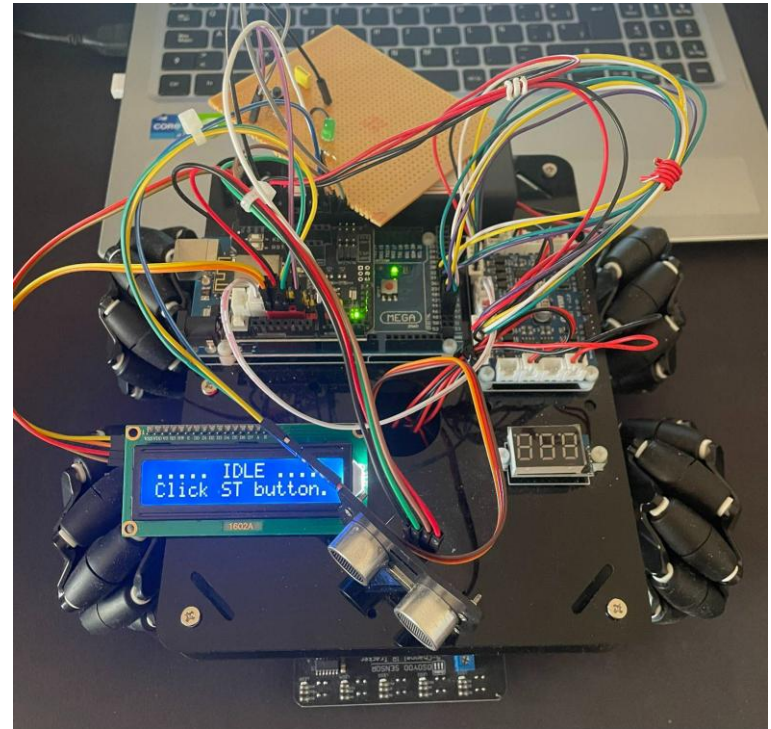
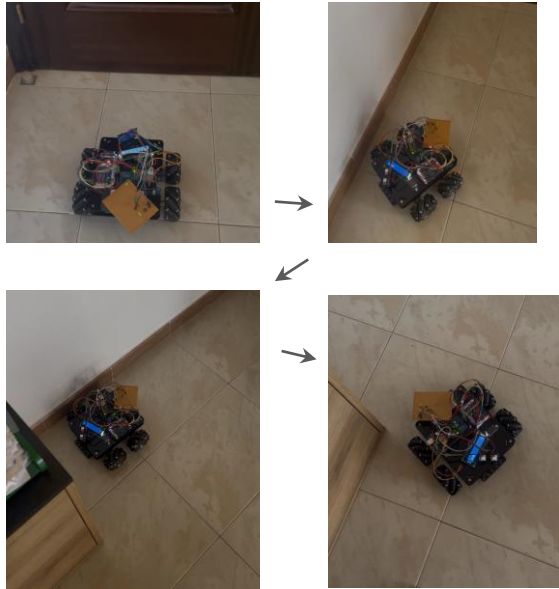
| Name                       | Address  | Type  | Value |
|----------------------------|----------|-------|-------|
| var_lcd_flag               | 00800312 | byte  | '\0'  |
| nearest_obstacle           | 00800104 | word  | 50    |
| distance_Measured          | 00800313 | word  | 50    |
| counter                    | 00800315 | word  | 0     |
| var_sampling_period        | 00800317 | word  | 10000 |
| var_time_elapsed_Servo     | 00800319 | dword | 584   |
| var_time_elapsed           | 0080031D | dword | 6666  |
| var_current_time_Servo     | 00800321 | dword | 16668 |
| var_current_time           | 00800325 | dword | 16668 |
| var_previous_time_Servo    | 00800329 | dword | 16084 |
| var_previous_time          | 0080032D | dword | 10002 |
| obstacle_position_operator | 00800331 | byte  | '\0'  |
| obstacle_position          | 00800332 | byte  | '\0'  |
| var_ST_flag                | 00800333 | byte  | '\0'  |
| var_Servo_flag             | 00800334 | byte  | 0x01  |
| var_SP_flag                | 00800335 | byte  | '\0'  |
| var_current_state          | 00800336 | byte  | 'c'   |
| var_SLED                   | 00800337 | byte  | 0x01  |
| var_S                      | 00800338 | byte  | 0x02  |



Arduino test b

# ROBOT AMB EVASIÓ D'OBSTACLES | Decisió de moviments

## 4. Prototipatge



# MODE DE RADIOCONTROL PER BLUETOOTH

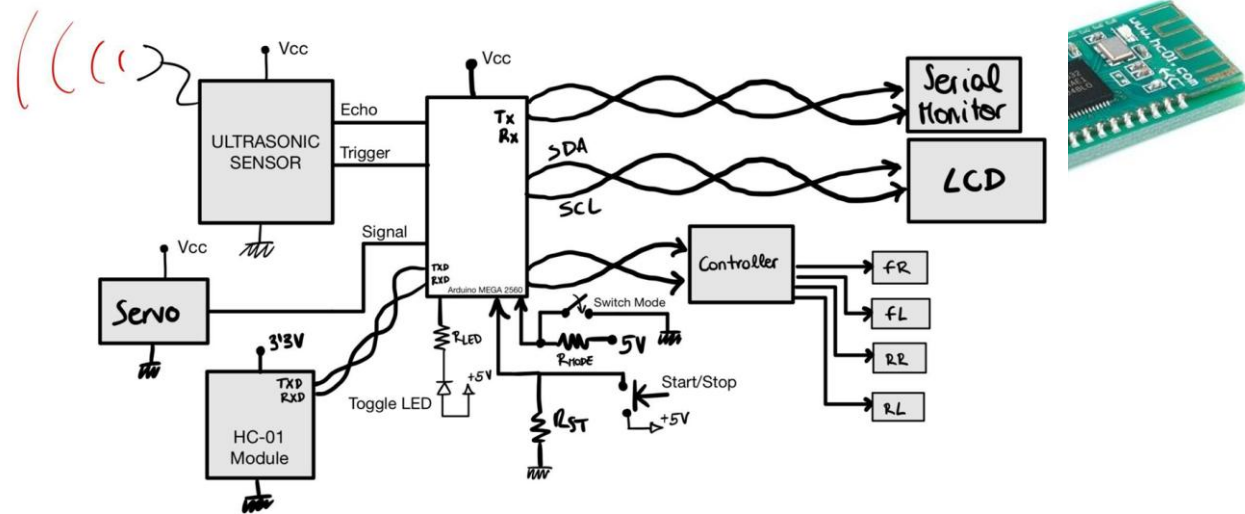
## 1. Especificacions: Elements i variables que afegim

Mode nou: *Bluetooth*

var\_Mode\_flag = 0 → Mo

var\_Mode\_flag = 1 → Mo

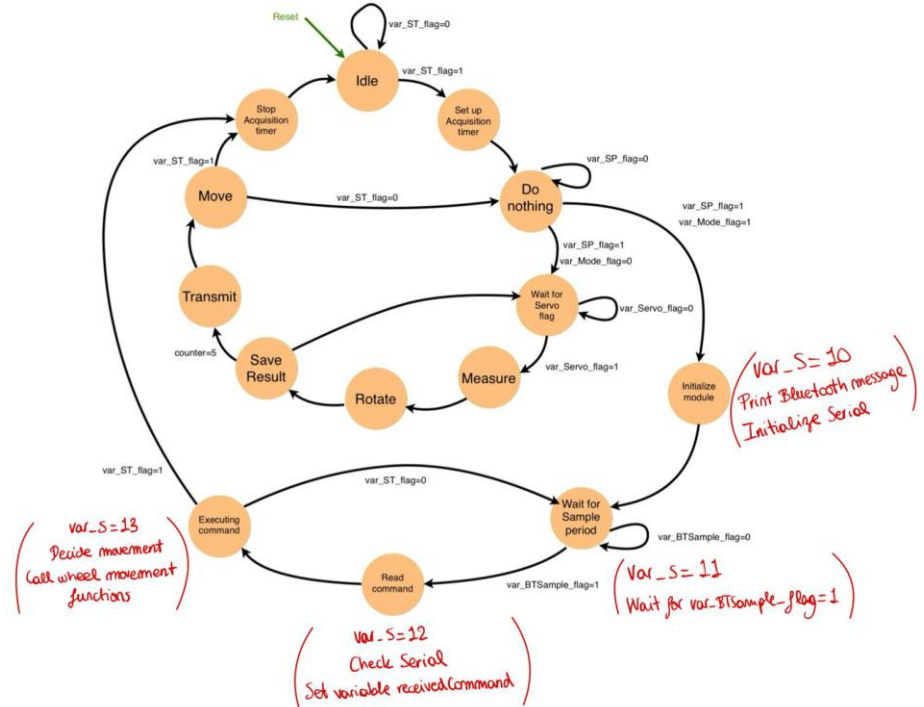
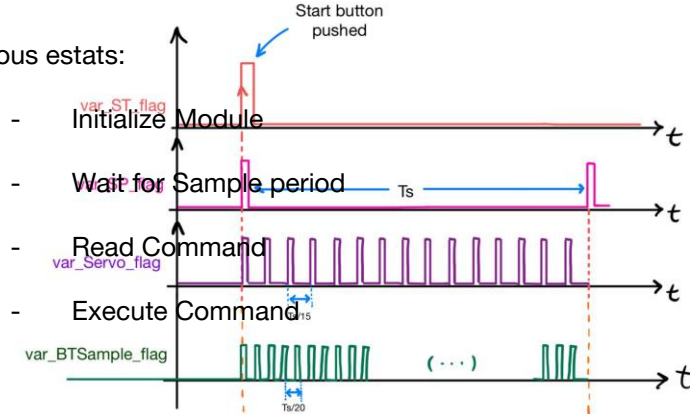
var\_BTsample\_flag



# MODE DE RADIOCONTROL PER BLUETOOTH

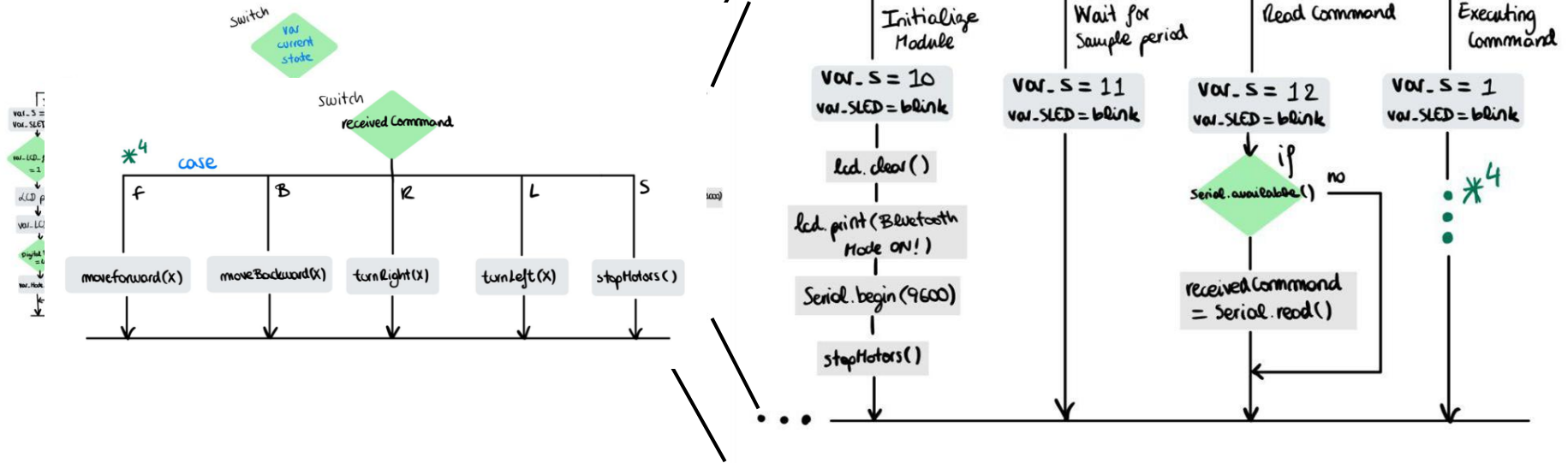
## 2. Planificació: Nou diagrama d'estats i funcions de moviment

Nous estats:



# MODE DE RADIOCONTROL PER BLUETOOTH

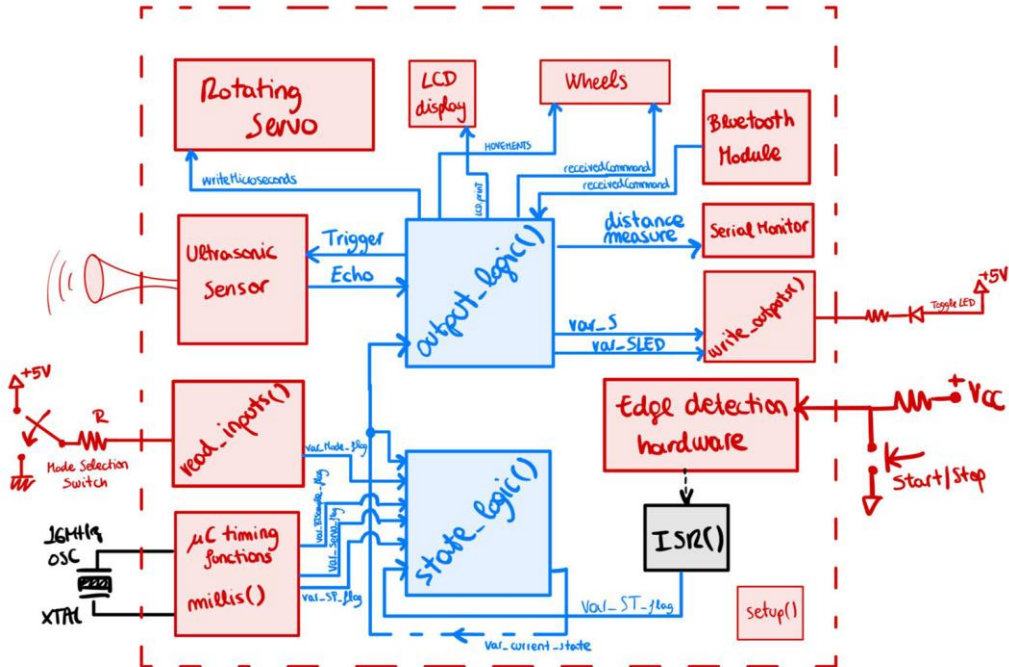
## 2. Planificació: Nous diagrames de flux



# MODE DE RADIOCONTROL PER BLUETOOTH

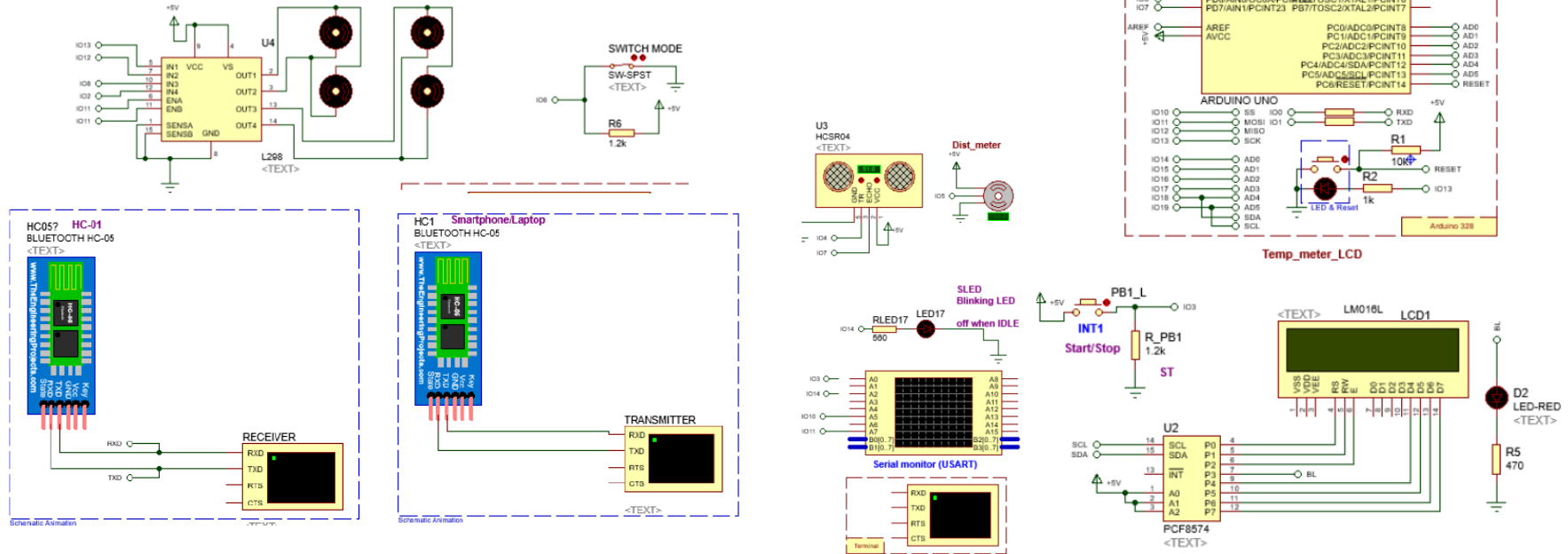
## 2. Planificació:

Diagrama hardware/software del sistema complet final



# MODE DE RADIOCONTROL PER BLUETOOTH

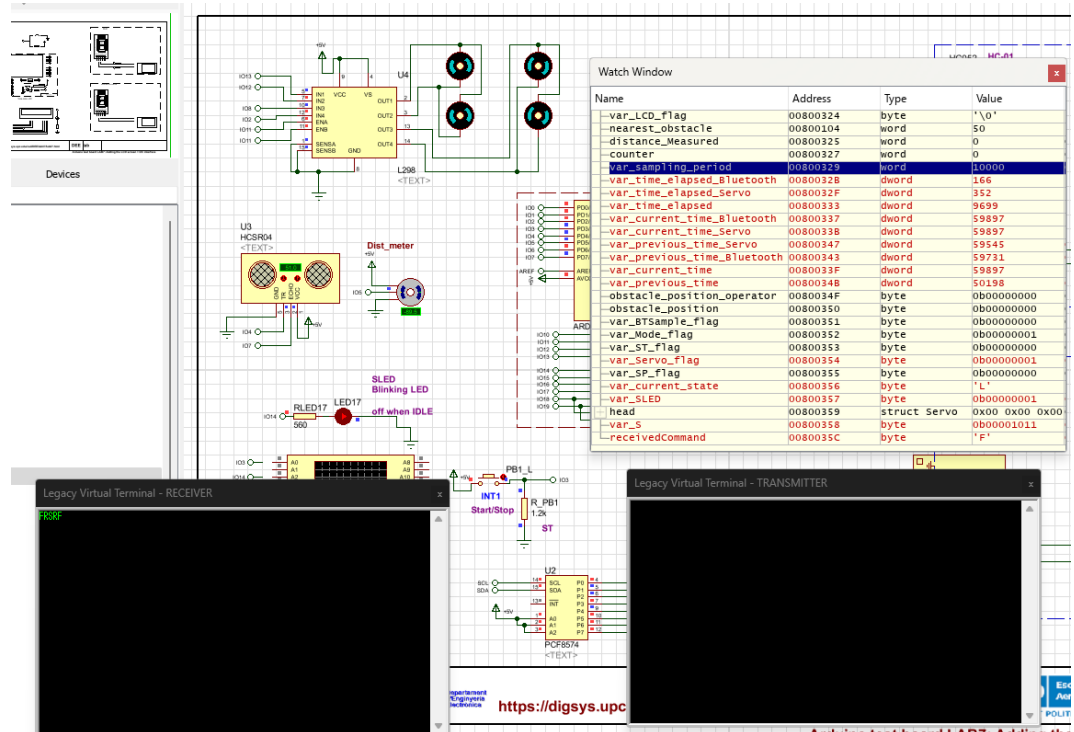
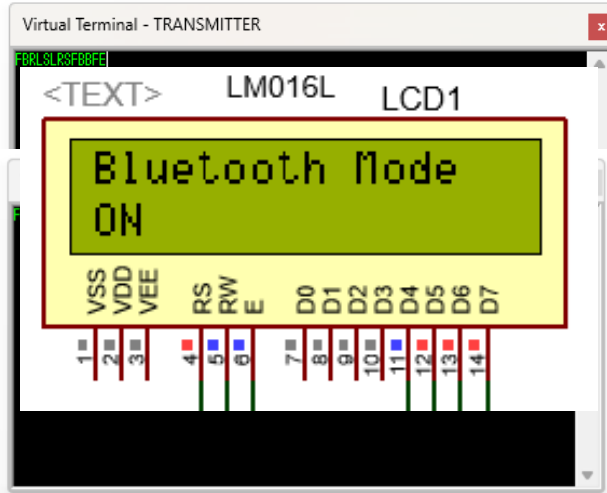
## 3. Desenvolupament i test: Rodes no independents





# MODE DE RADIOCONTROL PER BLUETOOTH

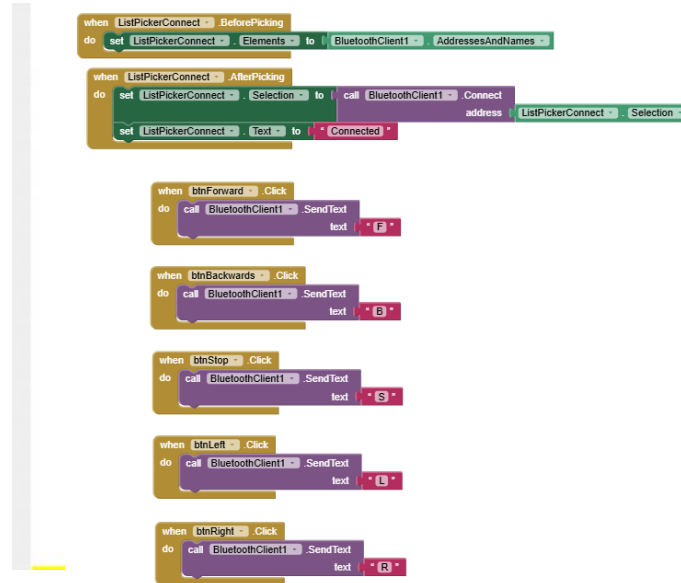
## 3. Desenvolupament i test: En simulació





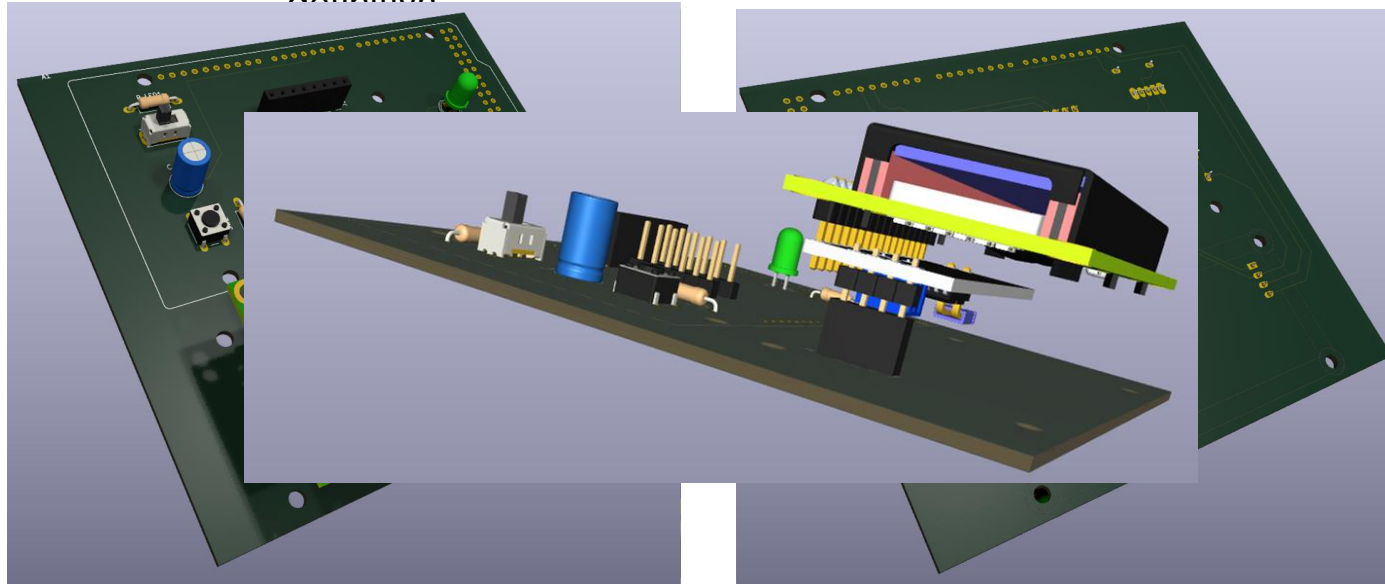
# MODE DE RADIOCONTROL PER BLUETOOTH

## 3. Desenvolupament i test: Programari



# MODE DE RADIOCONTROL PER BLUETOOTH

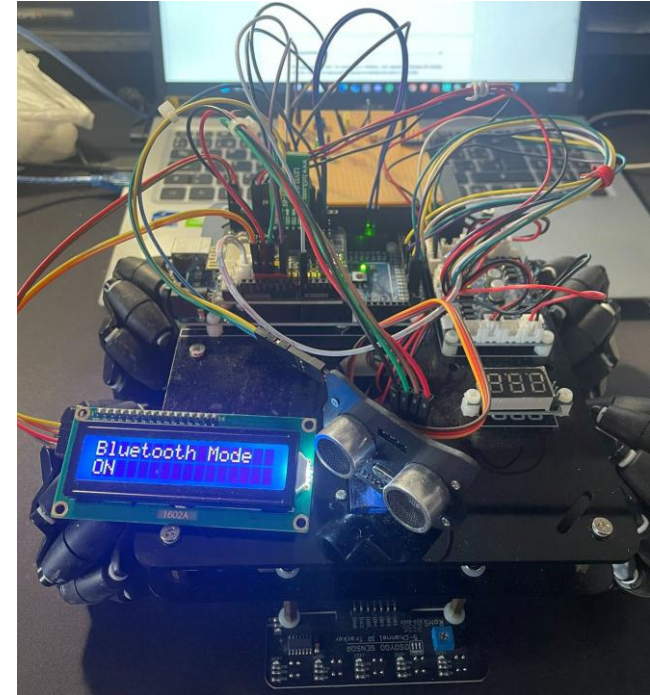
## 4. Prototipatge: Fase definitiva



# SISTEMA COMPLET EN FUNCIONAMENT

Vídeo de mostra del funcionament del robot:

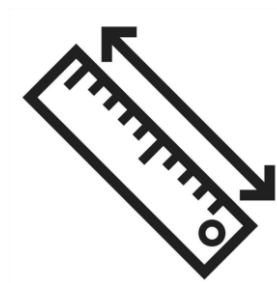
<https://www.youtube.com/watch?v=Ya-BwpK-UWY>



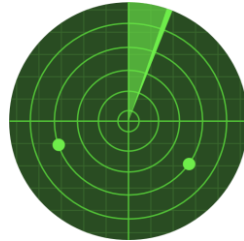
# CONCLUSIONS I POSSIBLES MILLORES

---

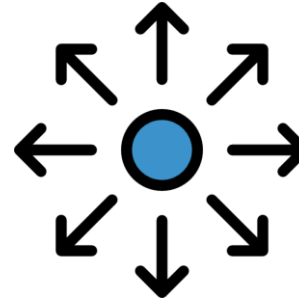
Sistema de  
mesura frontal



Sistema de mesura  
panoràmic



Moviment  
omnidireccional



Radiocontrol  
Bluetooth



# BIBLIOGRAFIA

---

- [8] Departament d'Enginyeria Electrònica - UPC. (n.d.). Computer Systems Design (CSD). Universitat Politècnica de Catalunya. <https://digsys.upc.edu/csd/index.html>
- [9] L1C02. (n.d.). HC-SR04 datasheet. AllDatasheet. <https://www.alldatasheet.com/datasheet-pdf/view/1192204/L1C02/HC-SR04.html>
- [9] Jameco Electronics. (n.d.). How Servo Motors Work. <https://www.jameco.com/Jameco/workshop/Howitworks/how-servo-motors-work.html>
- [21] Departament d'Enginyeria Electrònica - UPC. (n.d.). Digital Electronic Design (DEE). Universitat Politècnica de Catalunya. <https://digsys.upc.edu/csd/DEE/DEE.html>
- [10] Colvin, J. (n.d.). What is an H-Bridge? Digilent. Recuperado de <https://digilent.com/blog/what-is-an-h-bridge/>
- [3] Osoyoo. (2022, juliol). V2 Metal Chassis Mecanum Wheel Robotic for Arduino Mega2560 – Introduction. <https://osoyoo.com/2022/07/05/v2-metal-chassis-mecanum-wheel-robotic-for-arduino-mega2560-introduction-model-2021006600/>
- [11] Grabowiecki, A. (n.d.). Drawing of probably the first omni-directional wheel as described in Grabowiecki's US Patent [Figure]. ResearchGate.
- [41] Gudino, M. (2021, febrer 28). Arduino Uno vs. Mega vs. Micro. Arrow Electronics. <https://www.arrow.com/es-mx/research-and-events/articles/arduino-uno-vs-mega-vs-micro>
- <https://www.researchgate.net/figure/Drawing-of-probably-the-first-omni-directional-wheel-as-described-in-Grabowieckis-US-patent-224693506>
- [5] Generation Robots. (n.d.). Mecanum wheel application. <https://www.generationrobots.com/media/Mecanum-wheel-application.pdf>
- [12] STMicroelectronics. (n.d.). L298 datasheet. AllDatasheet. <https://www.alldatasheet.com/datasheet-pdf/view/22437/STMICROELECTRONICS/L298.html>
- [7] Jaffer, F. P. & Tahirad, H. D. (2015). Kinematic Model of a four mecanum wheeled mobile robot. ResearchGate. [https://www.researchgate.net/publication/276344731\\_Kinematic\\_Model\\_of\\_a\\_Four\\_Mecanum\\_Wheeled\\_Mobile\\_Robot](https://www.researchgate.net/publication/276344731_Kinematic_Model_of_a_Four_Mecanum_Wheeled_Mobile_Robot)
- [13] Wu, J. (2022, novembre). A Basic Guide to I2C (Application Note SBAA565). Texas Instruments. <https://www.ti.com/lit/an/sbaa565/sbaa565.pdf>
- [13] Zaragoza Makerspace (n.d.). Curso de Arduino y robótica. Motores paso a paso. <https://zazagomakerspace.com/courses/arduino-motores-paso-a-paso/>
- [14] Nebot Parra, V. (2025, juliol 2). Adaptació de kits d'aprenentatge de circuits electrònics a l'ensenyament d'enginyeria [Video]. YouTube. <https://www.youtube.com/watch?v=Ya-BwpK-UWY>

**MOLTES GRÀCIES  
PER LA VOSTRA ATENCIÓ!**