

**UNIVERSIDAD NACIONAL DE CÓRDOBA**  
FACULTAD DE CIENCIAS EXACTAS, FÍSICAS Y NATURALES



ELECTRÓNICA DIGITAL 2

---

**TRABAJO PRACTICO FINAL**

**Terreneitor:  
Auto controlado por Bluetooth**

---

Autor:  
Baccino, Luca

Profesor:  
Gomez, Mauro Gastón

2025

# Índice

<b>1. Introducción</b>	<b>3</b>
<b>2. Objetivos</b>	<b>3</b>
<b>3. Descripción del sistema</b>	<b>3</b>
3.1. Motores . . . . .	3
3.2. Comunicación PIC-CELULAR . . . . .	4
3.3. Sensor Ultrasónico HCRS04 y módulo ADC . . . . .	5
3.4. Displays 7 segmentos . . . . .	6
3.5. Pulsador e interruptores adicionales . . . . .	6
<b>4. Diagrama circuital</b>	<b>7</b>
<b>5. Fragmentos relevantes del código</b>	<b>7</b>
5.1. Inicialización de periféricos . . . . .	8
5.2. Subrutina de interrupción . . . . .	9
5.3. Control de motores . . . . .	9
5.4. Lectura de distancia con sensor ultrasónico . . . . .	9
5.5. Lectura del umbral desde potenciómetro . . . . .	9
5.6. Visualización en displays . . . . .	10
5.7. Frenado por botón físico . . . . .	10
<b>6. Pruebas, problemas y resultados</b>	<b>10</b>
6.1. Pruebas de movimiento por Bluetooth . . . . .	10
6.2. Frenado automático por proximidad . . . . .	10
6.3. Visualización en displays . . . . .	11
6.4. Interacción con pulsador e interruptores físicos . . . . .	11
6.5. Resumen de resultados . . . . .	11
6.6. Problemas . . . . .	11
6.6.1. 1. Consumo de energía y comportamiento errático de los motores . . . . .	11
6.6.2. 2. Fallas en la lectura de botones . . . . .	11
<b>7. Montaje</b>	<b>12</b>
<b>8. Conclusiones</b>	<b>13</b>
<b>A. Anexo - Repositorio y documentación técnica</b>	<b>15</b>
A.1. Repositorio del proyecto . . . . .	15
A.2. Hojas de datos de componentes . . . . .	15

# 1. Introducción

El presente informe describe el diseño e implementación de un vehículo robótico, denominado **Terre-neitor**, controlado de forma remota mediante comunicación Bluetooth y basado en el microcontrolador PIC16F887.

El sistema fue desarrollado en lenguaje ensamblador, se realizó un control preciso de los distintos periféricos utilizados. Se implementaron las siguientes funcionalidades principales: control de movimiento en cuatro direcciones via Bluetooth (adelante, atrás, izquierda, derecha), control de movimiento en dos direcciones (izquierda, derecha) con dos interruptores, visualización de la distancia de frenado en dos displays de 7 segmentos mediante multiplexado, pulsador de emergencia para frenado, regulación del umbral de frenado mediante un potenciómetro conectado al módulo ADC, y monitoreo de la distancia en tiempo real mediante un sensor ultrasónico.

Este documento detalla tanto la lógica de funcionamiento como los aspectos técnicos más relevantes, incluyendo el esquema de conexiones, las rutinas de interrupción, el uso de temporizadores, el manejo del módulo USART y las estrategias de multiplexado aplicadas. Se incluyen además fragmentos del código ensamblador comentado para una mejor comprensión del diseño.

## 2. Objetivos

En la sección detallada a continuación se plantearán los objetivos para la realización del proyecto:

- Implementar una interfaz de control remoto por Bluetooth, capaz de recibir comandos en tiempo real desde un celular.
- Desarrollo de un sistema de visualización mediante displays de 7 segmentos para mostrar valores relevantes del sistema realizando un multiplexado (distancia de frenado).
- Uso del módulo de conversión Analógica-Digital del microcontrolador con un potenciómetro para regular la distancia de frenado.
- Incorporación un sensor ultrasónico para la medición de distancia frente a obstáculos y frenar en el caso que se detecte una distancia menor a la regulada con el potenciómetro.
- Botón físico como mecanismo alternativo de frenado de emergencia.
- Implementar dos interruptores para ordenar un giro a la izquierda o derecha.
- Desarrollar rutinas de interrupción para manejar eventos, como la recepción de datos por USART o el conteo de tiempo por Timer0.

## 3. Descripción del sistema

El sistema implementado se compone de un microcontrolador PIC16F887, cuya programación fue desarrollada en lenguaje ensamblador. El control del vehículo se realiza mediante comunicación Bluetooth, utilizando el módulo MLT-BT05 conectado a los pines RX y TX del microcontrolador. Se tiene además un sensor ultrasónico para las mediciones de distancia, un regulador de voltaje AMS1117 para alimentar la protoboard con 5Volts, un potenciómetro controlando la distancia de frenado, 2 displays 7 segmentos, resistencias varias, transistores y una batería de 9Volts.

### 3.1. Motores

Para ejecutar los desplazamientos, se controlan dos motores conectados a los pines RA0, RA1, RA6 y RA7, permitiendo avanzar, retroceder y girar hacia ambos lados. La lógica de control incluye funciones de parada total y restauración del movimiento anterior cuando se frena por estar a una distancia menor a la distancia umbral, lo que garantiza una respuesta coherente del sistema frente a eventos de interrupción.

```

1  ADELANTE
2  BCF PORTA,0
3  BCF PORTA,1
4  BCF PORTA,6
5  BCF PORTA,7
6  BSF PORTA,1
7  BSF PORTA,7
8  RETURN

```

Listing 1: Rutina para movimiento hacia adelante

### 3.2. Comunicación PIC-CELULAR

Se hizo uso de un módulo Bluetooth MLT-BT05. Este es un módulo BLE (Bluetooth Low Energy) que permite la recepción de comandos enviados desde una aplicación móvil (LightBlue) y traducirlos en movimientos del vehículo.

```

1  RECEPCION
2  CALL    DELAY5MS
3  CALL    DELAY5MS
4  MOVF    RCREG,W      ; Leo el dato recibido
5  MOVWF   DATO_RX
6  MOVWF   ESTADO_MOVIMIENTO
7  CALL    DECODIFICACION
8  MOVLW   'A'
9  MOVWF   TXREG
10 BCF     PIR1,RCIF    ; Limpio bandera
11 CALL    DELAY5MS
12 CALL    DELAY5MS
13 CALL    DELAY5MS
14 CALL    DELAY5MS
15 FINALIZAR
16 RETFIE

17
18 RESTAURAR_MOVIMIENTO
19 MOVF     ESTADO_MOVIMIENTO, W
20 CALL     DECODIFICACION
21 RETURN

22
23 DECODIFICACION
24 MOVF     DATO_RX,W
25 XORLW    'A'        ; El dato recibido es Adelante?
26 BTFSC    STATUS,Z
27 GOTO     ADELANTE

28
29 MOVF     DATO_RX,W
30 XORLW    'R'        ; El dato recibido es Retroceder?
31 BTFSC    STATUS,Z
32 GOTO     RETROCEDER

33
34 MOVF     DATO_RX,W
35 XORLW    'D'        ; El dato recibido es Derecha?
36 BTFSC    STATUS,Z
37 GOTO     DERECHA

38
39 MOVF     DATO_RX,W
40 XORLW    'I'        ; El dato recibido es Izquierda?
41 BTFSC    STATUS,Z
42 GOTO     IZQUIERDA

43
44 MOVF     DATO_RX,W
45 XORLW    'S'        ; El dato recibido es Stop?
46 BTFSC    STATUS,Z
47 GOTO     STOP

48
49 RETURN

```

Listing 2: Rutina de recepción y decodificación

### 3.3. Sensor Ultrasónico HCRS04 y módulo ADC

El monitoreo de la distancia hacia posibles obstáculos se realiza mediante un sensor ultrasónico, el cual emplea el pin RA3 como salida de disparo (TRIGGER) y el pin RA4 como entrada de eco (ECHO). El tiempo entre la señal de disparo y la recepción del eco es medido utilizando el temporizador TMRO, y se convierte en una distancia expresada en centímetros. Este valor se compara con un umbral ajustable por el usuario a través de un potenciómetro conectado al pin RE0, configurado como entrada analógica y leído mediante el módulo ADC del microcontrolador. Si la distancia medida es menor que el umbral definido, el vehículo se detiene automáticamente.

```

1  START
2  BCF      PIR1, RCIF
3  CLRF     VAR_DISTANCIA ; limpio el registro donde guardo la distancia
4  BSF      TRIGGER
5  CALL     DELAY_10_MICROS
6  BCF      TRIGGER ; enciendo el trigger durante 10us (visto en datasheet)
7
8  ECHO_ES_1
9  BTFSS    ECHO ; si echo es uno se sigue midiendo la distancia
10 GOTO     ECHO_ES_1 ; bucle para cuando echo es 0
11 MOVLW    TMRO_CARGA58 ; cargo tmr0 para 58us (segun datasheet)
12 MOVWF    TMRO
13 BSF      INTCON,5 ; Habilito interrupcion por timer 0
14
15 ECHO_ES_0
16 BTFSC    ECHO ; si echo es 0 ya termine la medicion de distancia
17 GOTO     ECHO_ES_0 ; bucle para cuando echo es 1
18 BCF      INTCON,5 ; Detengo interrupciones por timer 0
19 CALL     MUESTRA_DISPLAY ; voy a mostrar en el display la distancia medida
20 MOVF     DISTANCIA_BLOQUEO, 0
21 BTFSC    STATUS, Z
22 CALL     RESTAURAR_MOVIMIENTO
23 GOTO     START ; vuelvo a iniciar una medicion

```

Listing 3: Medición de distancia con sensor ultrasónico

```

1  LECTURA_ADC
2  BANKSEL  ADCON0
3  BSF      ADCON0, GO ; Iniciar conversion
4  ADC_WAIT
5  BTFSC    ADCON0, GO
6  GOTO     ADC_WAIT ; Esperar hasta que termine
7  MOVF     ADRESH, W
8  MOVWF    DISTANCIA_UMBRAL
9  RETURN

```

Listing 4: Lectura del valor del potenciómetro mediante el ADC

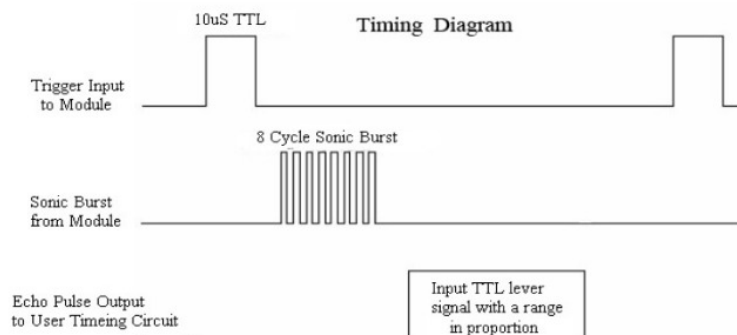


Figura 1: Encendido del TRIGGER durante  $10\mu s$

### 3.4. Displays 7 segmentos

La distancia umbral configurada por el usuario se visualiza en tiempo real en dos displays de 7 segmentos conectados al puerto PORTD, controlados mediante multiplexado utilizando los pines RC2 y RC3. Se desarrolló una rutina cíclica que actualiza cada display con su correspondiente valor BCD convertido y codificado, permitiendo una visualización clara y estable.

```

1      LOOP
2      BSF    PORTC,3 ; rutina para multiplexar displays
3      MOVF   DISPLAY1,0
4      MOVWF  PORTD
5      CALL   DELAY5MS
6      BCF    PORTC,3
7      BSF    PORTC,2
8      MOVF   DISPLAY2,0
9      MOVWF  PORTD
10     CALL   DELAY5MS
11     BCF    PORTC,2
12     DECFSZ CONTADOR3
13     GOTO   LOOP

```

Listing 5: Multiplexado de displays

### 3.5. Pulsador e interruptores adicionales

Adicionalmente, se incorporaron tres botones conectados a los pines RB2, RB3 y RB5, los cuales permiten ejecutar funciones adicionales sin interferir con la comunicación Bluetooth. El botón en RB2 actúa como freno de emergencia y detiene al vehículo de inmediato, mientras que los botones en RB3 y RB5 permiten girar hacia la izquierda y derecha, respectivamente, sin necesidad de enviar un comando desde el celular.

```

1      BOTON_POLL
2      CALL   DELAY5MS ; Antirrebote
3      CALL   DELAY5MS
4      CALL   DELAY5MS
5      CALL   DELAY5MS
6      BTFSS  PORTB, 2 ; Verifico si sigue presionado
7
8      GOTO   BOTON_ACCION ; Si, ejecuto accion
9      BTFSS  PORTB, 3
10     GOTO   BOTON_IZQ
11     BTFSS  PORTB, 5
12     GOTO   BOTON_DER
13     GOTO   CONTINUAR ; No, fue un rebote
14
15     BOTON_IZQ
16     CALL   IZQUIERDA ; Izquierda
17     MOVLW  'I'
18     MOVWF  ESTADO_MOVIMIENTO
19     MOVWF  DATO_RX
20     GOTO   CONTINUAR
21
22     BOTON_DER
23     CALL   DERECHA ; Derecha
24     MOVLW  'D'
25     MOVWF  ESTADO_MOVIMIENTO
26     MOVWF  DATO_RX
27     GOTO   CONTINUAR
28
29     BOTON_ACCION
30     CALL   STOP ; Detengo motores
31     MOVLW  'S'
32     MOVWF  ESTADO_MOVIMIENTO
33     MOVWF  DATO_RX
34
35     CONTINUAR
36     RETURN

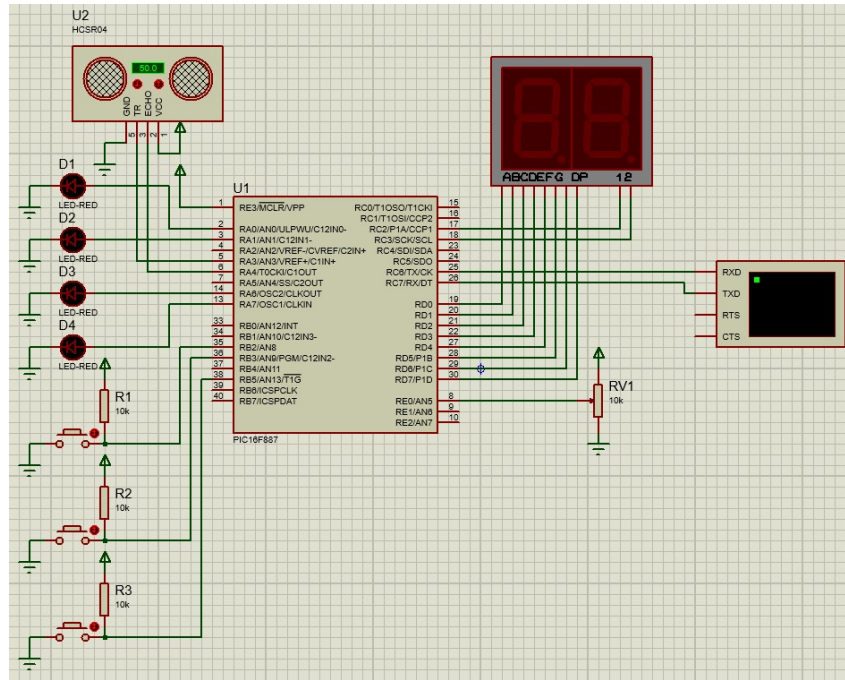
```

Listing 6: Check de pulsador e interruptores

El sistema se encuentra diseñado para funcionar de forma robusta, con rutinas antirrebote por software, manejo eficiente de interrupciones por recepción serial y temporización precisa para la medición ultrasónica. Todo el comportamiento del sistema está gestionado mediante estados definidos en registros internos, que permiten mantener el último comando recibido y restaurarlo cuando sea necesario.

## 4. Diagrama circuital

En la sección a continuación se detalla un diagrama circuital obviando las resistencias y los transistores utilizados para simplificar el entendimiento del mismo. Por otro lado, las salidas de control de los motores están representadas como LEDS.



## 5.1. Inicialización de periféricos

```

1  CONF
2  BANKSEL ANSEL
3  CLRF ANSEL ; seleccion puertos digitales y analogicos
4  CLRF ANSELH
5
6  BANKSEL OSCCON
7  movlw b'01100000' ;Oscilador interno a 4 MHz
8  movwf OSCCON
9
10 ; Configurar RE0 como entrada
11 BANKSEL TRISE
12 BSF TRISE, 0 ; RE0 como entrada
13
14 ; Habilitar AN5 (RE0) como canal analogico
15 BANKSEL ANSEL
16 BSF ANSEL, 5 ; AN5 habilitado
17
18 BANKSEL PORTE
19 CLRF PORTE
20
21 BANKSEL TRISA
22 MOVWL b'10110000' ; RC7/RX entrada
23 MOVWF TRISC ; RC6/TX salida
24 ; Y PARA MULTIPLEXAR DISPLAYS
25 MOVWL b'00100100' ; configuracion USART
26 MOVWF TXSTA ; y activacion de transmision
27 MOVWL .25 ; 9600 Baudios
28 MOVWF SPBRG
29 BSF PIE1,RCIE ; Habilita interrupcion en recepcion
30 CLRF TRISA
31 BCF TRISA,3 ; TRIGGER = SALIDA
32 BSF TRISA,4 ; ECHO = ENTRADA
33 CLRF TRISD ; PUERTO D DISPLAY
34 CLRF TRISB
35 BSF TRISB, 2 ; RB2 --> INPUT
36 BSF TRISB, 3 ; RB3 --> INPUT
37 BSF TRISB, 5 ; RB5 --> INPUT
38
39 BANKSEL RCSTA
40 MOVWL b'10010000' ; configuracion del USART para recepcion continua
41 MOVWF RCSTA ; puesta en ON
42
43 BANKSEL OPTION_REG ; configuracion timer 0
44 MOVWL B'00000000'
45 MOVWF OPTION_REG
46
47 BANKSEL ADCON0
48 MOVWL b'00010101' ; Canal AN5, ADC encendido
49 MOVWF ADCON0
50
51 BANKSEL ADCON1
52 MOVWL b'00000000'
53 MOVWF ADCON1
54
55 MOVWL b'11000000' ; Habilitacion de las interrupciones en general
56 MOVWF INTCON
57
58 BANKSEL PORTC ; limpio registros que voy a utilizar
59 BCF TRIGGER ; APAGO TRIGGER
60 CLRF PORTA
61 CLRF PORTB
62 CLRF PORTC
63 CLRF PORTD
64 CLRF W
65 CLRF CONTO
66 CLRF ESTADO_MOVIMIENTO
67 CLRF DATO_RX
68 MOVWL 'S'
69 MOVWF ESTADO_MOVIMIENTO

```

Listing 7: Configuración de puertos, ADC, USART, variables



Este bloque configura los puertos analógicos, la comunicación serie (USART), el ADC, y activa las interrupciones necesarias.

## 5.2. Subrutina de interrupción

```
1      ISR ; rutina de interrupcion
2      BTFSC    PIR1,RCIF    ; Interrupcion por recepcion?
3      GOTO     RECEPCION    ; Si --> voy a decodificar dato recibido
4      BTFSS    INTCON,2     ; Interrupcion por tmr0?
5      GOTO     FINALIZAR    ; No --> termina ISR
6      MOVLW    TMRO_CARGA58 ; mientras el echo sea 0 las interrupciones estaran
7      MOVWF    TMRO        ; habilitadas y se repetira la recarga de tmr0 hasta
8      INCF     VAR_DISTANCIA ; que se termine la medicion
9      BCF      INTCON,2
10     RETFIE
```

Listing 8: Subrutina de interrupción

## 5.3. Control de motores

```
1      ADELANTE
2      BCF     PORTA,0
3      BCF     PORTA,1
4      BCF     PORTA,6
5      BCF     PORTA,7
6      BSF     PORTA,1
7      BSF     PORTA,7
8      RETURN
```

Listing 9: Rutina para avanzar

El control del movimiento se realiza fijando los niveles adecuados en los pines de los motores. Rutinas para retroceder, girar y detenerse son análogas a la mostrada.

## 5.4. Lectura de distancia con sensor ultrasónico

```
1      START
2      BCF     PIR1, RCIF
3      CLRF    VAR_DISTANCIA
4      BSF     TRIGGER
5      CALL    DELAY_10_MICROS
6      BCF     TRIGGER
```

Listing 10: Inicio de medición ultrasónica

Se genera un pulso de 10 microsegundos en el pin TRIGGER y se mide el tiempo hasta recibir el eco haciendo uso del TIMER0.

## 5.5. Lectura del umbral desde potenciómetro

```
1      LECTURA_ADC
2      BANKSEL ADCON0
3      BSF     ADCON0, GO
4      ADC_WAIT
5      BTFSC    ADCON0, GO
6      GOTO     ADC_WAIT
7      MOVF     ADRESH, W
8      MOVWF    DISTANCIA_UMBRAL
9      RETURN
```

Listing 11: Lectura del valor de frenado

El valor leído del ADC representa la distancia mínima antes de frenar automáticamente, el valor se guarda en la variable DISTANCIA UMBRAL.

## 5.6. Visualización en displays

```
1      LOOP
2      BSF      PORTC ,3
3      MOVF     DISPLAY1 ,0
4      MOVWF    PORTD
5      CALL     DELAY5MS
6      BCF      PORTC ,3
7
8      BSF      PORTC ,2
9      MOVF     DISPLAY2 ,0
10     MOVWF    PORTD
11     CALL     DELAY5MS
12     BCF      PORTC ,2
13
14     DECFSZ    CONTADOR3
15     GOTO     LOOP
```

Listing 12: Multiplexado de displays de 7 segmentos

Este fragmento actualiza los valores visualizados en los displays de forma alternada para generar el efecto de multiplexado.

## 5.7. Frenado por botón físico

```
1      BOTON_ACCION
2      CALL     STOP
3      MOVLW    'S'
4      MOVWF    ESTADO_MOVIMIENTO
5      MOVWF    DATO_RX
```

Listing 13: Frenado de emergencia

Ante la activación de un botón físico conectado a RB2, el sistema detiene los motores y guarda el nuevo estado como "S" (Stop).

# 6. Pruebas, problemas y resultados

Para validar el correcto funcionamiento del sistema desarrollado, se llevaron a cabo una serie de pruebas funcionales sobre cada uno de los subsistemas implementados. Estas pruebas se realizaron sobre el prototipo final, alimentado con una batería de 9V.

## 6.1. Pruebas de movimiento por Bluetooth

Se utilizaron comandos enviados desde la aplicación **LightBlue** para verificar los desplazamientos del vehículo en las cuatro direcciones posibles. Se comprobó que el sistema responde correctamente a los comandos 'A' (adelante), 'R' (retroceso), 'I' (izquierda) y 'D' (derecha), enviando las señales apropiadas a los pines de control de los motores. Se observó una respuesta inmediata al recibir cada comando, y la dirección se mantiene hasta recibir una nueva orden o activarse alguna condición de frenado.

## 6.2. Frenado automático por proximidad

Con el sensor ultrasónico ubicado en la parte delantera del vehículo, se midió la distancia a obstáculos colocados en su trayectoria. Se utilizó un potenciómetro para configurar la distancia mínima de frenado, la cual se visualiza en tiempo real en los displays. Cuando el obstáculo se encuentra a una distancia menor que la configurada, el sistema detiene automáticamente el movimiento del vehículo, y se mantiene detenido hasta que el obstáculo se retira. Al tener únicamente 2 displays se tomó como máxima distancia los 99 cm, pero de acuerdo al datasheet la distancia máxima de medición es mayor (400 cm).

### 6.3. Visualización en displays

Los displays de 7 segmentos muestran el valor del umbral de frenado leído por el ADC. Se comprobó que el valor cambia en tiempo real al girar el potenciómetro. La multiplexación garantiza una visualización estable, con leves parpadeos debido a las múltiples mediciones.

### 6.4. Interacción con pulsador e interruptores físicos

Se probaron tres botones conectados a los pines RB2, RB3 y RB5. El botón en RB2 cumple la función de freno de emergencia, deteniendo el movimiento del vehículo y sobrescribiendo el estado anterior. Los botones en RB3 y RB5 permiten girar a la izquierda o a la derecha sin necesidad de utilizar el celular. Todos los botones cuentan con una rutina antirrebote por software que evita activaciones no deseadas.

### 6.5. Resumen de resultados

El sistema respondió correctamente a prácticamente todo lo esperado, y se comprobó que es capaz de:

- Ejecutar órdenes de movimiento de forma remota con mínima latencia.
- Frenar automáticamente al detectar obstáculos, de acuerdo con un umbral configurable.
- Detenerse con un pulsador de emergencia.
- Visualizar correctamente la distancia de frenado en los displays.
- Realizar giros laterales utilizando botones físicos.

Se concluye que el comportamiento del sistema es robusto y cumple con los objetivos propuestos.

### 6.6. Problemas

Durante el desarrollo e implementación del proyecto se presentaron distintos inconvenientes tanto a nivel de hardware como de software. A continuación se detallan los problemas de mayor relevancia, junto con las estrategias adoptadas para su resolución.

#### 6.6.1. 1. Consumo de energía y comportamiento errático de los motores

Uno de los primeros problemas observados fue el comportamiento inestable del vehículo durante las pruebas prolongadas. Los motores no respondían de forma adecuada, deteniéndose, no arrancando, apagando todos los componentes o mostrando pérdida de potencia. Este comportamiento fue atribuido al rápido desgaste de las baterías utilizadas, debido a los picos de corriente producidos por los motores al momento de arrancar y frenar.

**Solución:** Se sustituyó la batería agotada por una batería de pruebas durante el desarrollo, y se reservó una batería nueva y completamente cargada exclusivamente para la presentación final. Esta medida permitió asegurar el funcionamiento estable del sistema en todo momento.

#### 6.6.2. 2. Fallas en la lectura de botones

Inicialmente se proyectó la utilización de un teclado matricial como método de entrada adicional para controlar funciones del vehículo sin necesidad de Bluetooth. Sin embargo, su implementación presentó múltiples dificultades:

- El teclado interfería con las pruebas de movilidad, al dificultar el acceso físico al prototipo en movimiento.
- Las interrupciones por cambio de estado en el puerto PORTB no permitían el funcionamiento del teclado matricial.

- Se observaron comportamientos erráticos incluso con interrupciones por flanco en el pin RB0.

Se realizaron numerosas pruebas para intentar resolver el problema, entre ellas:

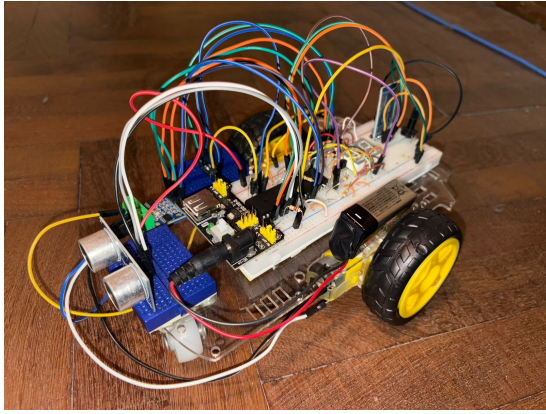
- Reemplazo del teclado por botones individuales y/o interruptores.
- Variación del tipo de interrupción: cambio de estado, flanco de bajada y flanco de subida.
- Utilización de distintos pines del puerto PORTB (desde RB0 hasta RB7).
- Soldado de cables directamente a los pines del pulsador.
- Verificación del correcto funcionamiento de los botones con un multímetro (continuidad y caída de tensión).
- Pruebas con resistencias de pull-up internas del microcontrolador, y con resistencias externas conectadas manualmente.

A pesar de todos los intentos, no se logró un funcionamiento estable utilizando interrupciones. Las resistencias de pull-up internas del PIC 16F887 no funcionaron adecuadamente con los botones implementados, lo cual agravaba la situación. También se realizó una simulación en Proteus, cuyo resultado está detallado en un vídeo en la sección `.Anexo`.

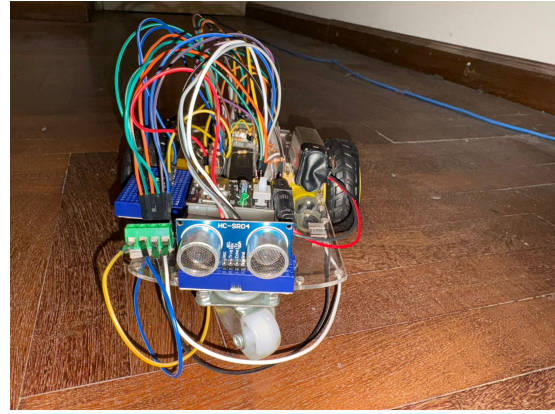
**Solución:** Finalmente se descartó el uso de interrupciones para la lectura de botones, y se optó por una verificación por **polling** durante el ciclo principal del programa. Esta técnica permitió una detección robusta del estado de los botones, acompañada de una rutina antirrebote por software (antirrebote realizado a partir de la rutina de delay ya configurada) para evitar falsas activaciones.

## 7. Montaje

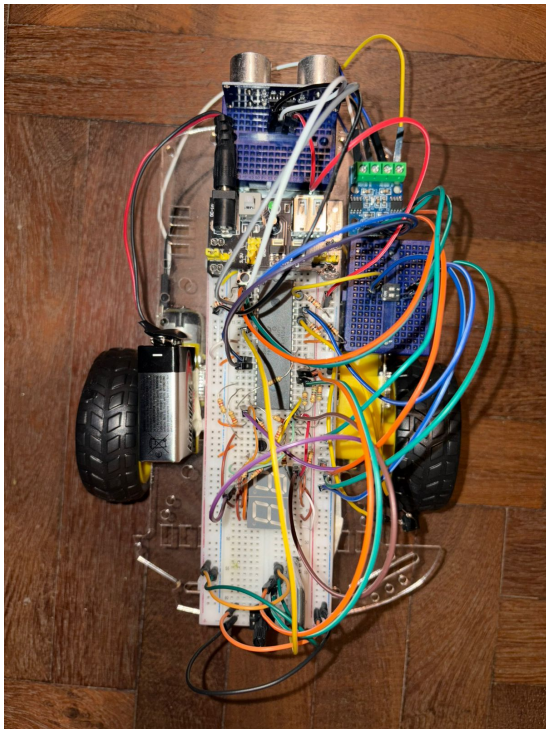
En la sección a continuación se muestran diferentes imágenes donde se puede observar desde diferentes perspectivas el montaje del auto:



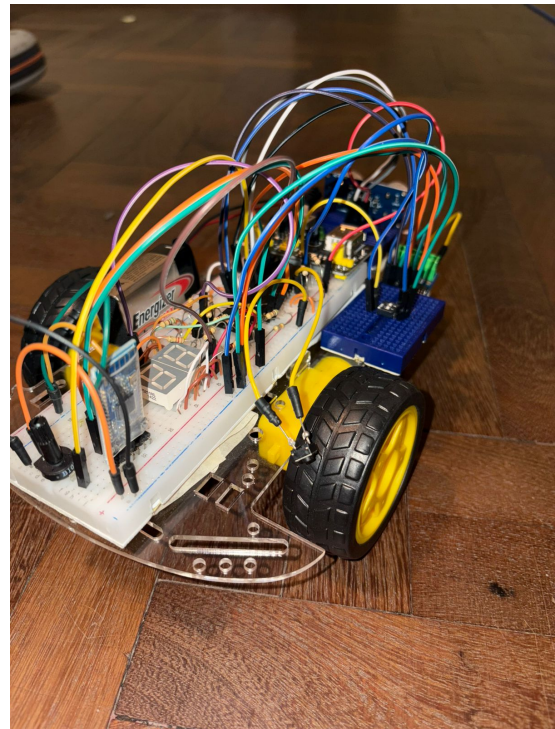
(a) Montaje del circuito - vista 1



(b) Montaje del circuito - vista 2



(c) Montaje del circuito - vista 3



(d) Montaje del circuito - vista 4

Figura 3: Montaje físico del sistema en distintos ángulos.

## 8. Conclusiones

El desarrollo del presente proyecto permitió integrar diversos conocimientos adquiridos a lo largo de la asignatura, aplicándolos en un sistema funcional. El vehículo **Terreneitor** se mostró de amplia utilidad para la experimentación y uso de las diferentes funcionalidades del microcontrolador PIC16F887, tales como el módulo ADC, TIMER0, USART, interrupciones y técnicas de multiplexado.

Se logró implementar un sistema que opera por un lado, mediante comunicación Bluetooth desde un celular, y por otro, mediante botones físicos que permiten la operación directa del vehículo.

La inclusión del sensor ultrasónico y del potenciómetro como entrada al ADC permitió simular una funcionalidad de “frenado automático” configurable por el usuario, lo cual añade una capa de complejidad y automatización al comportamiento del robot.

Asimismo, el uso de displays de 7 segmentos controlados por multiplexado permitió una visualización en tiempo real del umbral de frenado, haciendo el sistema más comprensible para el usuario.

A pesar de los desafíos encontrados, particularmente en la implementación del botón, las soluciones adoptadas permitieron alcanzar un sistema estable y funcional. En definitiva, este proyecto permitió consolidar habilidades en diseño digital, programación de un microcontrolador PIC16F887, análisis de hardware, y resolución de problemas de integración electrónica.



## A. Anexo - Repositorio y documentación técnica

### A.1. Repositorio del proyecto

El código fuente completo del trabajo práctico final, así como imágenes y documentación adicional (vídeo de simulación interrupción con botón), se encuentran disponibles en el siguiente repositorio (de mi propiedad) público de GitHub:

- [github.com/EsqueletoX/Trabajo\\_Final\\_Digital\\_2](https://github.com/EsqueletoX/Trabajo_Final_Digital_2)

### A.2. Hojas de datos de componentes

A continuación, se listan las hojas de datos (datasheets) correspondientes a los principales componentes utilizados en el desarrollo del proyecto:

- Sensor ultrasónico HC-SR04: [alldatasheet.com/datasheet/HC-SR04](https://www.alldatasheet.com/datasheet/HC-SR04)
- Módulo Bluetooth BLE (basado en CC2541): [alldatasheet.com/datasheet/CC2541](https://www.alldatasheet.com/datasheet/CC2541)
- Regulador de tensión AMS1117: [alldatasheet.com/datasheet/AMS1117](https://www.alldatasheet.com/datasheet/AMS1117)
- Transistor BC547: [alldatasheet.com/datasheet/BC547](https://www.alldatasheet.com/datasheet/BC547)
- Microcontrolador PIC16F887: [alldatasheet.com/datasheet/PIC16F887](https://www.alldatasheet.com/datasheet/PIC16F887)
- Display 7 segmentos: [alldatasheet.com/datasheet/7SEG](https://www.alldatasheet.com/datasheet/7SEG)