

Proyecto Final  
Robot recolector de residuos  
Diseño, implementación y construcción física

Guillermo Campelo  
Juan Ignacio Goñi  
Diego Nul

23 de junio de 2010

**Resumen**

# Índice

<b>1. Requerimientos</b>	<b>5</b>
<b>2. Ideas de implementación</b>	<b>5</b>
2.1. Locomoción . . . . .	5
2.2. Sensado del entorno . . . . .	5
2.3. Controlador . . . . .	5
2.4. Método de recolección . . . . .	5
<b>3. Actuadores</b>	<b>5</b>
3.1. Motores de continua . . . . .	5
3.1.1. Características . . . . .	5
3.1.2. Circuito de control . . . . .	6
3.1.3. Rutinas de control . . . . .	7
3.2. Servo motores . . . . .	8
3.2.1. Circuito de control . . . . .	8
3.2.2. Rutinas de control . . . . .	9
<b>4. Sensado</b>	<b>9</b>
4.1. Telémetros infrarrojos . . . . .	9
4.1.1. Características . . . . .	9
4.1.2. Circuito de control . . . . .	9
4.1.3. Rutinas de control . . . . .	11
4.2. Sensor de distancia por ultrasonido . . . . .	11
4.2.1. Características . . . . .	11
4.2.2. Circuito de control . . . . .	11
4.2.3. Rutinas de control . . . . .	12
4.3. Sensor reflectivo de piso . . . . .	13
4.3.1. Características . . . . .	13
4.3.2. Circuito de control . . . . .	13
4.3.3. Diagrama de conexión . . . . .	13
4.3.4. Rutinas de control . . . . .	13
4.4. Encoders . . . . .	13
4.4.1. Características . . . . .	13
4.4.2. Circuito de control . . . . .	13
4.4.3. Diagrama de conexión . . . . .	13
4.4.4. Rutinas de control . . . . .	13
4.5. Sensado de la batería . . . . .	13
4.5.1. Características . . . . .	14
4.5.2. Circuito de control . . . . .	14
4.5.3. Rutinas de control . . . . .	14
4.6. Consumo del motor . . . . .	14
4.6.1. Características . . . . .	14
4.6.2. Circuito de control . . . . .	14
4.6.3. Rutinas de control . . . . .	14
4.6.4. Pulsador u otro dispositivo disparador . . . . .	14
4.6.5. Características . . . . .	14
4.6.6. Circuito de control . . . . .	14
4.6.7. Rutinas de control . . . . .	14

<b>5. Controladores</b>	<b>14</b>
5.1. Netbook . . . . .	14
5.2. Microcontrolador . . . . .	15
5.2.1. Características . . . . .	15
5.2.2. Diagrama del microcontrolador . . . . .	15
5.2.3. Módulos internos . . . . .	15
5.2.4. Programación del firmware . . . . .	15
<b>6. Comunicación</b>	<b>15</b>
6.1. Conectividad entre módulos . . . . .	15
6.2. Protocolo de comunicación . . . . .	15
6.2.1. Características básicas . . . . .	15
6.2.2. Comandos comunes . . . . .	15
6.2.3. Comandos específicos . . . . .	15
6.2.4. Estadísticas . . . . .	15
<b>7. Placas controladoras</b>	<b>16</b>
7.1. Placa genérica . . . . .	16
7.1.1. Características principales . . . . .	16
7.1.2. Módulo de comunicación . . . . .	16
7.1.3. Alimentación de la placa . . . . .	16
7.1.4. Configuración . . . . .	16
7.1.5. Esquemático . . . . .	16
7.1.6. Circuito . . . . .	16
7.1.7. Código básico . . . . .	16
7.1.8. Posibles extensiones . . . . .	16
7.2. Placa controladora de motores DC . . . . .	16
7.2.1. Características principales . . . . .	17
7.2.2. Módulo de comunicación . . . . .	17
7.2.3. Alimentación de la placa . . . . .	17
7.2.4. Configuración . . . . .	17
7.2.5. Esquemático . . . . .	17
7.2.6. Circuito . . . . .	17
7.2.7. Código básico . . . . .	17
7.2.8. Posibles extensiones . . . . .	17
7.3. Placas de sensado . . . . .	17
7.3.1. Características principales . . . . .	17
7.3.2. Módulo de comunicación . . . . .	18
7.3.3. Alimentación de la placa . . . . .	18
7.3.4. Configuración . . . . .	18
7.3.5. Esquemático . . . . .	18
7.3.6. Circuito . . . . .	18
7.3.7. Código básico . . . . .	18
7.3.8. Posibles extensiones . . . . .	18
7.4. Placa controladora de servo motores . . . . .	18
7.4.1. Características principales . . . . .	18
7.4.2. Módulo de comunicación . . . . .	18
7.4.3. Alimentación de la placa . . . . .	18
7.4.4. Configuración . . . . .	19
7.4.5. Esquemático . . . . .	19

7.4.6.	Circuito . . . . .	19
7.4.7.	Código básico . . . . .	19
7.4.8.	Posibles extensiones . . . . .	19
<b>8.</b>	<b>Armado del prototipo</b>	<b>19</b>
8.1.	Diseño . . . . .	19
8.2.	Características . . . . .	19
8.3.	Desarme . . . . .	19
8.4.	Costo y proveedores . . . . .	19
<b>A.</b>	<b>Primer apéndice Hardware</b>	<b>20</b>
<b>B.</b>	<b>Segundo apéndice Hardware</b>	<b>20</b>

## 1. Requerimientos

## 2. Ideas de implementación

### 2.1. Locomoción

distintos tipos de locomoción que tuvimos en cuenta y porque elegimos este

### 2.2. Sensado del entorno

distintos tipos de sensores disponibles y porque elegimos estos

### 2.3. Controlador

distintas formas de diagramar la forma de control, que tipo de controladores necesitamos, en cuales pensamos, con cuales nos quedamos

### 2.4. Método de recolección

distintos metodos que se nos ocurrieron

## 3. Actuadores

En nuestro caso, los motores son la principal forma en que el robot puede interactuar activamente con el ambiente que lo rodea. Cada una de las tareas que debíamos realizar requería de actuadores acordes.

Estas cuestiones son las que analizamos en este apartado.

### 3.1. Motores de continua

Para la tracción principal de las ruedas necesitabamos motores que tuvieran el torque necesario para mover el robot, pero que pudieramos medir y controlar la velocidad era la principal necesidad. Para esta tarea utilizamos motores de continua con caja reductora y encoder. Con dos de estos motores logramos poder garantizar una velocidad determinada en las ruedas, controlar de la cantidad de movimiento en forma independiente en cada rueda y entre otras cosas conocer la cantidad de las vueltas dadas por cada una de las ruedas.

#### 3.1.1. Características

Los motores que elegimos son de la marca Ignis <sup>1</sup> modelo *MR-2FA* con características expresadas en la tabla 1, están provistos de una caja reductora, poseen un encoder de 4 estados por vuelta en el eje del motor y un sensor de efecto de campo para determinar una vuelta en la salida de la caja reductora.

La caja reductora provee una relación de 94 vueltas del motor por cada 1 vuelta del eje de salida de la caja.

En la figura 1 mostramos las dimensiones exteriores del motor.

---

<sup>1</sup><http://www.ignis.com.ar>

Característica	Unidad	Mínimo	Nominal	Máximo
Tensión	V	8	9	12
Corriente	A	0.6	1.2	2.4
Velocidad	RPM	1	60	60
Aceleración	$1/s^2$	0.1	0.1	0.5
Torque	kgf*cm	0	1.2	6.4

Cuadro 1: Características del motor Ignis MR-2FA.

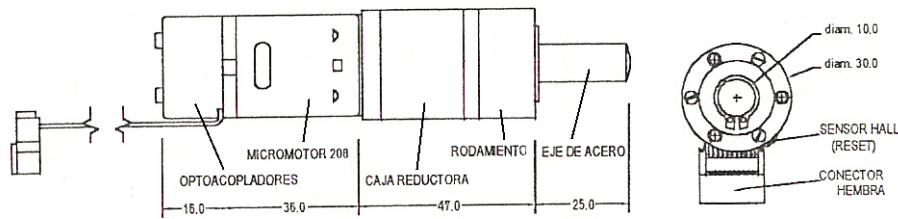


Figura 1: Vista lateral y frontal del motor Ignis MR-2FA. Escala 1 : 2.

Una ventaja que encontramos en este modelo es que ya trae el encoder integrado aunque su resolución podría haber sido mayor. Los encoders los explicamos más en detalle en la sección 4.4.

### 3.1.2. Circuito de control

Para alimentar y poder controlar los motores elegimos el driver *L298* de la marca ST <sup>2</sup>. Internamente tiene dos puentes H puentesables y puede soportar hasta 4A. Ideal para estos motores u otros que se elijan en el futuro.

La principal función del driver era proveer de la corriente y voltaje necesarios para el funcionamiento del motor, pero la configuración del puente H nos dio la posibilidad de, con una lógica simple, determinar el sentido de la corriente y potencia que recibía el motor.

Este integrado admite el puenteo de la salida aumentando así la corriente que pueda circular por él. Para hacer esto, conectamos las salidas *Out1* y *Out4* por un lado y por el otro las salidas *Out2* y *Out3*. De igual forma los pines de habilitación *EnA* y *EnB*, luego la entrada *In1* con la *In4* por un lado y por el otro la *In2* con la *In3*.

De esta forma, se controla con sólo 3 cables, uno de habilitación y otros 2 de *Input* que determinan la polarización de los transistores internos y por ende, el sentido de giro del motor. En la tabla 2 mostramos la tabla de verdad para los pines de control. En la figura 2 mostramos el diagrama interno del integrado.

Para determinar la potencia que recibiría el motor usamos el módulo de *PWM* del microcontrolador, que explicamos más en detalle en la sección 5.2.3. Variando el ancho del pulso sobre el pin de habilitación del driver determinamos la cantidad de tiempo que el motor recibe tensión, lo cual se traduce en la

<sup>2</sup><http://www.st.com>

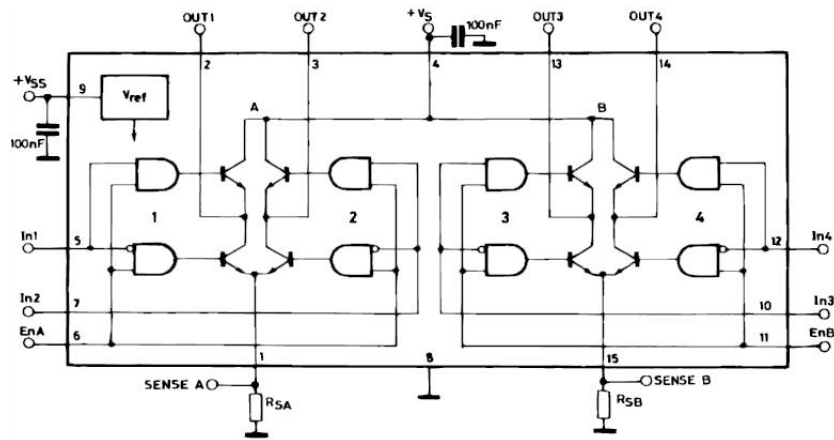


Figura 2: Diagrama interno del driver L298.

Enable	Input 1	Input 2	Función
H	H	L	Sentido horario
H	L	H	Sentido anti-horario
H	L	L	Motor frenado
H	H	H	Motor frenado
L	X	X	Motor libre

Cuadro 2: Tabla de verdad para el control del driver *L298*. Donde H es estado alto, L estado bajo y X cualquier estado.

potencia que este tiene para realizar el movimiento. Cuanto mayor es el tiempo en estado alto del pulso, mayor la potencia.

Para contrarrestar la corriente negativa en las salidas del driver usamos los diodos *FR304* que cumplen con las especificaciones del driver con 150ns de tiempo de recuperación y una corriente de 3A.

El consumo del motor lo medimos mediante los pines de sensado en el driver, conectado a masa por una resistencia y al módulo de *ADC* del microcontrolador, que explicamos en la sección 5.2.3. Conociendo el valor de la resistencia y el valor leído por el módulo de *ADC*, pudimos determinar cuanta corriente que circulaba por la resistencia y por ende la corriente consumida por el motor.

Para controlar la velocidad del motor usamos 1 de los 2 encoders que trae. Conectados como entrada para el módulo de *Timer* del microcontrolador, en configuración de contador, que explicamos más en detalle en la sección 5.2.3. Usando otro *Timer* para tener una base de tiempo fija y con el valor del contador pudimos determinar y controlar la velocidad de las ruedas.

En la sección 7.2 explicamos el desarrollo de la placa controladora de estos motores.

### 3.1.3. Rutinas de control

Desde el punto de vista del código, tuvimos que desarrollar las rutinas necesarias para el manejo de los motores según las instrucciones del controlador prin-

Característica	Unidad	Valor
Torque	kg	6,5
Velocidad	segundos/grado	$\frac{0,16}{60}$
Voltaje	V	4,8 a 6
Delay máximo	$\mu s$	4
Dimensiones	mm	40x20x38
Peso	g	39

Cuadro 3: Características del servo HX5010.

cial.

Configuramos a uno de los timers internos del microcontrolador para que genere una interrupción cada 6,25ms, la cual usamos para realizar chequeos y ejercer control sobre el motor. Verificamos el consumo del motor para evitar sobrecargar el circuito y los motores ante un posible atasco de las ruedas. También actualizamos el acumulado de vueltas realizadas por el motor para la odometría.

Cada 200ms, o sea 32 interrupciones, tomamos la cantidad de cuentas del encoder, corregimos la velocidad de giro del motor ajustando el ancho del pulso generado por el PWM. Luego borramos el contador y esperamos otros 200ms.

### 3.2. Servo motores

Para el movimiento de las partes del módulo de recolección, una cámara con paneo y giro o un sensor de ultrasonido colocado en la parte superior haciendo las veces de radar, pensamos en el uso de servo motores. La principal característica de estos actuadores es que nosotros sólo debemos indicar el ángulo al que queremos que este el eje del motor y este se coloca automáticamente. El ángulo de trabajo va desde 0° a 180° y algunos llegan hasta los 200°.

Aunque no fue implementado el ningún mecanismo que requiriera el uso de estos motores, explicamos en este apartado el trabajo realizado en torno a este tipo de actuadores. En la sección 7.4 explicamos el diseño de las placas que los controlan.

El servo de prueba que utilizamos para el desarrollo es el modelo *HX5010* de la marca Hextronik <sup>3</sup> con características que expresamos en la tabla 3

#### 3.2.1. Circuito de control

La alimentación y consumo depende del modelo específico, variando también el torque que posee el servo.

No es necesario el uso de un driver para manejarlos, simplemente con la alimentación y una señal con el ángulo es suficiente. La forma de comunicar el ángulo varía entre los distintos servos y fabricantes. Hay servos analógicos y servos digitales. En los primeros la posición se determina mediante un voltaje que varía según cierto rango y si es digital, se setea mediante el ancho de un pulso que tiene un tiempo mínimo y máximo para mapear los ángulos mínimo y máximo respectivamente.

<sup>3</sup>[www.hextronik.com/](http://www.hextronik.com/)



Dentro del modo de uso, podemos hacer que queden sueltos o que se queden fijos en cierta posición indicando, de forma continua, el valor del ángulo requerido. La frecuencia a la que se debe setear la posición depende el modelo.

### 3.2.2. Rutinas de control

Debido a que pensamos usar servos digitales y por lo menos íbamos a necesitar 3 servos, necesitábamos contar con varios módulos de PWM. Como sólo disponíamos de 1, decidimos realizar la misma función pero por software.

Usamos el timer de 16bits del microcontrolador configurado con el clock interno como medida del tiempo para crear 5 salidas con pulsos que varían el ancho en forma independiente cada una. Definimos un ancho mínimo y máximo, pudiendo configurar pasos intermedios de  $1^\circ$  (aproximadamente  $69,4\mu s$ ).

## 4. Sensado

En este apartado explicamos detalladamente cada uno de los sensores que utilizamos para realizar tanto las mediciones externas como las internas al robot. Analizamos las ventajas de cada uno, problemas que encontramos y sus soluciones.

En la sección 7.3 explicamos el diseño y construcción de las placas que controlan todos los sensores del robot.

### 4.1. Telémetros infrarrojos

El principio de funcionamiento de estos sensores es mediante un haz de luz infrarroja que es emitido hacia el objetivo, el cual es reflejado y captado a través de un lente por un sensor de posición relativa en el interior del sensor. En base a esta medición se calcula la distancia entre el sensor y el objeto reflectivo que se encuentra frente a él.

#### 4.1.1. Características

medidas, tiempo de muestreo, tipo de salida, rangos de distancia, rangos de voltaje, distancia vs voltaje

Los telémetros infrarrojos que elegimos son de la marca Sharp <sup>4</sup>, modelo *GP2D120*. En la tabla 4 detallamos los valores característicos del modelo.

Este tipo de sensores tienen un retardo de aproximadamente  $43,1ms$  durante el cual la lectura que se realiza no es confiable y luego las nuevas lecturas se hacen en ventanas de aproximadamente el mismo tiempo. En la figura 3 mostramos el diagrama de tiempos.

#### 4.1.2. Circuito de control

alimentación, conmutación (transistor, estado de habilitación: 0), conexión con el micro, salida del sensor, módulo ADC, muestreo, capacitor para alimentación, circuito mínimo (diagrama)

---

<sup>4</sup><http://sharp-world.com/products/device>

Característica	Unidad	Valor
Rango máximo	cm	30
Rango mínimo	cm	4
Tensión para la máxima distancia	V	1.95
Tensión para la mínima distancia	V	2.55
Tensión de alimentación	V	5
Consumo máximo	mA	50

Cuadro 4: Características del sensor de distancia por ultrasonido SRF05.

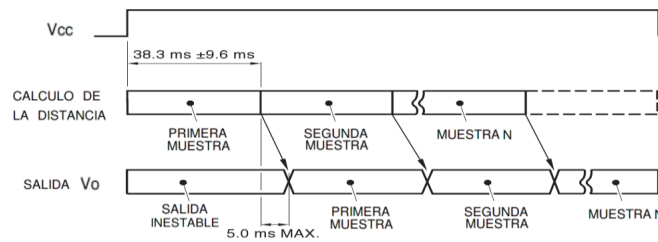


Figura 3: Diagrama de tiempos del sensor GP2D120.

No necesitábamos un driver para manejar los telémetros, pero decidimos usar un transistor para poder habilitarlos o no, de otra manera el sensor estaba tomando mediciones continuamente provocando un consumo de batería innecesario. De esta forma sólo se enciende cuando se va a utilizar.

El transistor que utilizamos es un conmutador y amplificador de uso general, el *BC327* y por ser de tipo PNP se exita con un estado bajo, por lo que la lógica de conmutación está negada.

El tipo de salida de este modelo de telémetros es analógica y están conectadas al módulo *ADC* del microcontrolador. En la figura 4 mostramos la tabla de conversión entre voltaje de salida y distancia al objeto, y en la figura 5 mostramos el ángulo de apertura de la zona de detección según la distancia al objetivo.

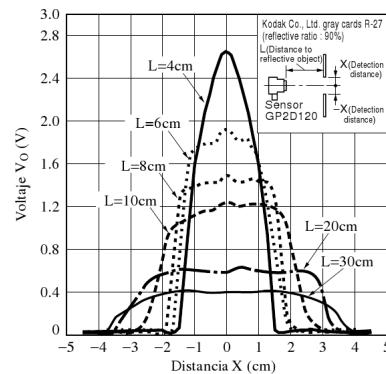
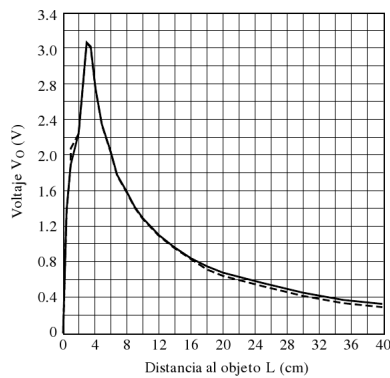


Figura 4: Voltaje de salida según la distancia al objeto del telémetro GP2D120. Figura 5: Ángulo de apertura según la distancia del telémetro GP2D120.

#### 4.1.3. Rutinas de control

Controlar los telémetros es relativamente sencillo. Usamos el timer de 16bits del microcontrolador configurado con el clock interno para determinar el tiempo en el cual debíamos tomar las muestras con el ADC y simplemente realizamos un promedio entre ellas para obtener la distancia al objetivo.

Para hacer esto desarrollamos una pequeña máquina de estados que controla y maneja los tiempos para tomar las muestras que explicamos más en detalle en la sección 7.3.7.

### 4.2. Sensor de distancia por ultrasonido

Estos sensores de distancia se basan en la velocidad del sonido para calcular la distancia al objetivo. Genera un tren de 8 pulsos ultrasónicos y luego se espera como respuesta, el mismo tren de pulsos que debería haber rebotado contra el objetivo. En base a la diferencia de tiempo entre la emisión del tren de pulsos y la respuesta, se calcula la distancia a la que se encuentra el objetivo.

#### 4.2.1. Características

El sensor de distancia por ultrasonido que elegimos es el modelo *SRF05* de la marca Devantech Ltd <sup>5</sup>. Esta versión mejorada del modelo *SRF04*, aumenta el rango de detección y mejora el modo de control y lectura de los datos, permitiendo hacerlo mediante un único pin.

La distancia medida mediante el tren de pulsos es codificada linealmente en el ancho de un pulso que varía de  $100\mu s$  a  $25ms$ . Si dentro del rango de detección no se encuentra ningún objeto, el pulso tendrá un ancho de  $30ms$ .

En la tabla 5 detallamos las características del sensor *SRF05* y en la figura 6 mostramos el haz ultrasónico del sensor.

Característica	Unidad	Valor
Tensión de alimentación	<i>V</i>	5
Corriente	<i>mA</i>	4
Frecuencia de trabajo	<i>KHz</i>	40
Rango máximo	<i>cm</i>	400
Rango mínimo	<i>cm</i>	1.7
Duración mínima del pulso de disparo	$\mu s$	10
Duración del pulso eco de salida	$\mu s$	100 - 25000
Tiempo mínimo de espera entre mediciones	<i>ms</i>	50
Dimensiones	<i>mm</i>	43x23x40

Cuadro 5: Características del sensor SRF05.

#### 4.2.2. Circuito de control

No necesitamos de un driver para manejar al sensor ya que lo conectamos directo a los 5v de la placa. El pin de *Mode* lo dejamos en estado bajo para

<sup>5</sup><http://www.robot-electronics.co.uk/>

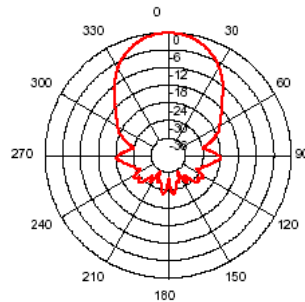


Figura 6: Haz ultrasónico del sensor SRF05.

indicar que debe funcionar bajo el nuevo modo y no en compatibilidad con el *SRF04*.

En el pin de *TRIGGER* sólo generamos un pulso de al menos  $10\mu s$  para desencadenar en la lectura de la distancia al objetivo. El sensor nos asegura que no generará el pulso de respuesta hasta pasados los  $700\mu s$  desde pasado el pulso de trigger. En la figura 7 mostramos el diagrama de tiempos.

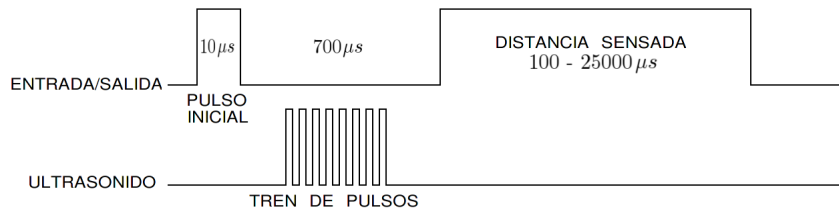


Figura 7: Diagrama de tiempos del sensor SRF05.

Para evitar que el rebote de otros sensores o sensados anteriores influya en la lectura, se debe esperar un mínimo de  $50ms$  antes de generar otro medición.

#### 4.2.3. Rutinas de control

Usamos uno de los pines con interrupción externa en el que conectamos el pin de *TRIGGER* y el timer de 16bits del microcontrolador configurado con el clock interno.

Para realizar la medición, generamos un pulso de  $15\mu s$  para asegurarnos el disparo del sensor y cambiamos el modo del pin a entrada con interrupción ante un flanco ascendente. Cuando salta la interrupción significa que comienza al pulso con la distancia codificada en su ancho, por lo que tomamos una muestra del timer y configuramos al pin para que genere ahora una interrupción ante un flanco descendente. Cuando salte la próxima vez la interrupción, será porque termino el pulso con la medición, por lo que sólo debemos hacer la resta entre el valor actual del timer y la muestra que tomamos al principio para conocer la distancia a la que se encuentra el objetivo.

Si el tiempo tomado mayor a los  $25ms$  indica que no se detectó ningún objeto dentro del rango de detección.

### **4.3. Sensor reflectivo de piso**

principio de funcionamiento

#### **4.3.1. Características**

modelo, marca, medidas, alimentacion, consumo, tiempo/frecuencia de muestreo, tipo de salida, distancia optima, rangos de voltaje, distancia vs voltaje

#### **4.3.2. Circuito de control**

alimentacion, resistencias elegidas, conmutacion (transistor, estado de habilitacion: 0), conexion con el micro, salida del sensor, modulo ADC, muestreo, capacitor para alimentacion, circuito minimo (diagrama)

#### **4.3.3. Diagrama de conexión**

asignacion de pines

#### **4.3.4. Rutinas de control**

codigo de lectura de nivel de reflexion (explicacion)

### **4.4. Encoders**

principio de funcionamiento

estamos usando solo uno de los sensores del encoder

#### **4.4.1. Características**

tipo de encoders, cuentas x vuelta de eje de motor, velocidad maxima y minima recomendable

relacion de caja 94:1 max recomendable: 300 cuentas x segundo min recomendable: 60 - 70 cuentas x segundo

#### **4.4.2. Circuito de control**

alimentacion, conexionado, circuito, resistencias pull-up, swtch selector, timer/counter

#### **4.4.3. Diagrama de conexión**

asignacion de pines

#### **4.4.4. Rutinas de control**

codigo de lectura y correccion de la velocidad del motor (explicacion)

### **4.5. Sensado de la bateria**

principio de funcionamiento

#### **4.5.1. Características**

grafico/tabla de voltaje bateria vs salida

#### **4.5.2. Circuito de control**

conexionado, circuito, modulo ADC, muestreo

#### **4.5.3. Rutinas de control**

codigo de lectura de nivel de tension en la bateria (explicacion)

### **4.6. Consumo del motor**

principio de funcionamiento

#### **4.6.1. Características**

grafico/tabla de corriente consumida vs voltaje, características del puente H, valores maximos y minimos, mensajes de consumo alto

#### **4.6.2. Circuito de control**

valor de la resistencia, circuito, modulo ADC, muestreo, vref en el micro

#### **4.6.3. Rutinas de control**

codigo de lectura de nivel de tension en la bateria (explicacion)

#### **4.6.4. Pulsador u otro dispositivo disparador**

posibilidad de poner un pulsador o cualquier otro dispositivo que genere un cambio de estado y lo detecte como trigger

#### **4.6.5. Características**

como deberia ser el boton o algun otro dispositivo que vayamos a poner ahi

#### **4.6.6. Circuito de control**

alimentacion, consumo, circuito, interrupciones

#### **4.6.7. Rutinas de control**

codigo de lectura de cambio de estado en el pin de trigger (explicacion)

## **5. Controladores**

### **5.1. Netbook**

modelo, marca, características, para que se usa, sistema operativo y lenguaje de programacion

## **5.2. Microcontrolador**

para que vamos a usar el micro y sus funciones principales

### **5.2.1. Características**

modelo, marca, familia, memorias, etc

### **5.2.2. Diagrama del microcontrolador**

grafico y asignacion de pines x modulo

### **5.2.3. Módulos internos**

listado de modulos que tiene y características de cada uno

### **5.2.4. Programación del firmware**

pines de programacion, icd2, IDE, lenguaje, version

## **6. Comunicación**

porque necesitamos comunicar los modulos, que necesidades hay, nivel de uso

### **6.1. Conectividad entre módulos**

daisy chain, diagrama, montado sobre rs232, control de errores

### **6.2. Protocolo de comunicación**

características necesarias en el protocolo, porque es importante, cosas que tuvimos en cuenta y decisiones, control de errores

#### **6.2.1. Características básicas**

formado por paquetes, formato basico del paquete (header), control de errores

#### **6.2.2. Comandos comunes**

contelo o listado de comandos comunes (en detalle o se van a un apendice)  
- son pocos.

#### **6.2.3. Comandos específicos**

contelo o listado de comandos especificos segun el tipo de placa (referencia a un apendice con cada uno explicado)

#### **6.2.4. Estadísticas**

analisis de paquetes por segundo, bytes de datos vs bytes de header, retransmisiones, etc

## **7. Placas controladoras**

porque tuvimos que diseñar nuestras propias placas, cosas que tuvimos en cuenta y decisiones tomadas, códigos fuente a los apéndices

### **7.1. Placa genérica**

función de una placa genérica, porque fue armada, para que sirva

#### **7.1.1. Características principales**

testeo de nuevos módulos, testeo de la programación, sniffear la comunicación, futuras expansiones

#### **7.1.2. Módulo de comunicación**

explicación de la comunicación, igual en todas, switch de configuración, pines, fichas, nodos en la cadena, cables pc-placa y placa-placa, max232

#### **7.1.3. Alimentación de la placa**

tensión para la alimentación, circuito de la fuente, consumo máximo, voltaje mínimo de alimentación, alimentación de 5V directos

#### **7.1.4. Configuración**

configuración mínima de la placa, leds, comunicación, header de programación

#### **7.1.5. Esquemático**

esquemas de la placa

#### **7.1.6. Circuito**

circuito de la placa

#### **7.1.7. Código básico**

explicación de lo mínimo que debería tener para ser parte de la cadena de comunicación

#### **7.1.8. Posibles extensiones**

posibles extensiones a futuro de la placa - nuevos módulos de testeo o control o lectura muy básica de señales, pasar a montaje superficial los componentes, hacerla más chica

### **7.2. Placa controladora de motores DC**

función de una placa controladora de motor DC, porque fue armada, para que sirva, porque hay 2, porque no está en una sola



#### **7.2.1. Características principales**

principio de funcionamiento, como logra controlar la velocidad, como logra ser parte de la cadena, como logra sensar el consumo, controlar el motor, puente H, diodos, leds, VREF

#### **7.2.2. Módulo de comunicación**

se explico en el modulo generico, se agregan los comandos especificos y se puede explicar como se obtiene la informacion para dar las respuestas

#### **7.2.3. Alimentación de la placa**

se explico en el modulo generico, tension para la alimentacion para los motores, necesidad de masa unica como referencia, consumo aproximado de los motores

#### **7.2.4. Configuración**

configuracion de la placa, leds, comunicacion, header de programacion, switch de seleccion de encoder

#### **7.2.5. Esquemático**

esquematicos de la placa

#### **7.2.6. Circuito**

circuito de la placa

#### **7.2.7. Código básico**

explicacion de lo minimo que deberia tener para ser parte de la cadena de comunicacion, sensado y control de la velocidad de los motores

#### **7.2.8. Posibles extensiones**

unificar en una placa el control de mas de un motor, pasar a montaje superficial los componentes, hacerla mas chica

### **7.3. Placas de sensado**

funcion de una placa de sensado, porque fue armada, para que sirve, que tipo de sensores puedo conectar, cuales son las posibles configuraciones, diferencias, sensado de la bateria

#### **7.3.1. Características principales**

principio de funcionamiento, como logra tomar las muestras de los sensores, como logra ser parte de la cadena, seteo de los tipos de sensores

### **7.3.2. Módulo de comunicación**

se explico en el modulo generico, se agregan los comandos especificos y se puede explicar como se obtiene la informacion para dar las respuestas

### **7.3.3. Alimentación de la placa**

se explico en el modulo generico

### **7.3.4. Configuración**

configuracion de la placa, comunicacion, header de programacion

### **7.3.5. Esquemático**

esquematicos de la placa

### **7.3.6. Circuito**

circuito de la placa

### **7.3.7. Código básico**

explicacion de lo minimo que deberia tener para ser parte de la cadena de comunicacion y sensado de los distintos perifericos

### **7.3.8. Posibles extensiones**

uso de componentes como resistencias variables para regular la alimentacion de los sensores de piso y resistencias pull-up, pasar a montaje superficial los componentes, hacerla mas chica

## **7.4. Placa controladora de servo motores**

funcion de una placa controladora de servos, porque no fue armada, para que se penso, alguna otra opcion de conexion, pines libres

### **7.4.1. Características principales**

principio de funcionamiento, como logra generar varios pwm por software, como logra ser parte de la cadena

### **7.4.2. Módulo de comunicación**

se explico en el modulo generico, se agregan los comandos especificos y se puede explicar como se obtiene la informacion para dar las respuestas

### **7.4.3. Alimentación de la placa**

se explico en el modulo generico, con modificaciones que permiten que circule una mayor cantidad de corriente para alimentar a los servos.

#### **7.4.4. Configuración**

configuracion de la placa, comunicacion, header de programacion

#### **7.4.5. Esquemático**

esquematicos de la placa

#### **7.4.6. Circuito**

circuito de la placa

#### **7.4.7. Código básico**

explicacion de lo minimo que deberia tener para ser parte de la cadena de comunicacion y control de los servos

#### **7.4.8. Posibles extensiones**

uso de componentes como resistencias variables para regular la alimentacion de los sensores de piso y resistencias pull-up, pasar a montaje superficial los componentes, hacerla mas chica

### **8. Armado del prototipo**

#### **8.1. Diseño**

#### **8.2. Características**

con las ruedas de 10cm y teniendo en cuenta que el motor gira a unas 300 cuentas/segundo (usando un solo sensor en el encoder) llegamos a 50 cm/s de velocidad

#### **8.3. Desarme**

#### **8.4. Costo y proveedores**

## **A. Primer apéndice Hardware**

protocolo de comunicacion.  
conexionado y configuracion de la comunicacion.  
circuitos de las placas.  
codigo fuente de las controladoras.  
costo del prototipo.

## **B. Segundo apéndice Hardware**