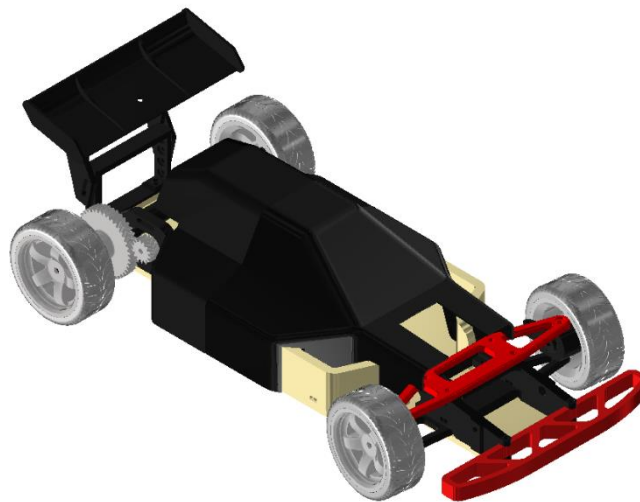




CONTROL 2 (EL73)
TRABAJO FINAL
Ciclo 2021-2



Profesor: Dante Anael Vargas Machuca Acevedo
Sección: EL248

Apellidos y Nombres	Código
Bartolo Palomino, Joel	U201822396
Cruzado Vásquez, Anderson	U201822491
De la Torre Velarde, Elvis Joel	U201822520
Huallanca Escalera, Harold	U201822201
Montalvo Cuadros, Luis Angel	U201822203
Perez Quispe, Jesús Carlos	U201822409
Reyes Shishco, Merly	U201823320
Salcedo Rodríguez, Joan	U201822603
Vasquez Frias, Jean Carlos	U201822202

Tabla de contenido

1. Introducción:	4
2. Objetivo general:	4
3. Objetivos específicos:	4
4. Desarrollo:	4
4.1. Modelado de espacio de estados de la velocidad	4
Función de transferencia de la posición del servomotor DC	8
4.2. Diseño mecánico	10
4.3. Diseño electrónico	11
4.3.1. DISEÑO EN SIMULINK PARA EL CONTROL DEL MOTOR 222050 DC	11
4.3.2. RESULTADOS EN SIMULINK Y MATLAB	12
4.3.2.1. Corriente Simulink	12
4.3.2.2. Corriente Matlab	12
4.3.2.3. Ley de control Simulink	13
4.3.2.4. Ley de control Matlab	13
4.3.2.5. Set point vs velocidad (out put) Simulink	14
4.3.2.6. Set point vs velocidad (out put) Matlab	14
4.3.2.7. Resultados en matlab	15
4.4. Diseño en Proteus	15
4.5. Selección de la etapa de potencia	18
4.5.1. Transistor TIP122:	18
4.5.2. Relay G2R-2:	19
4.6. Sensores a utilizar	20
4.6.1. Sensor de velocidad: LM331	20
4.6.2. Sensor de corriente: ACS712ELCTR-05B-T	23
4.7. Conexión via celular	25
4.7.1. Conexión Serial	25
4.7.2. Diseño de interfez gráfica	26
4.8. Motores utilizados	27
4.8.1. Motor 222050	27
4.8.2. Servomotor	27
5. Conclusiones:	28
6. Bibliografía	29
7. Anexos	30
7.1. Anexo 1: Código de MATLAB de la velocidad optimal	30

Indice de imágenes:

Imagen 1 Modelamiento de espacio de estados del motor DC modelo 222050	4
Imagen 2 Diagrama de bloques del controlador proporcional	9
Imagen 3 Plano de explosión del diseño mecánico del carrito	10
Imagen 4 Simulación en Simulink del controlador de velocidad con ganancia optimal.....	11
Imagen 5 Gráfica de la corriente en Simulink	12
Imagen 6 Gráfica de la corriente en MATLAB	12
Imagen 7 Gráfica de la Ley de control en Simulink	13
Imagen 8 Gráfica de la Ley de Control en MATLAB	13
Imagen 9 Gráfica de Set point vs velocidad en Simulink	14
Imagen 10 Gráfica de set point vs velocidad en MATLAB.....	14
Imagen 11 Resultados de la velocidad, corriente y el voltaje en MATLAB	15
Imagen 12 Control proporcional para la posición.....	15
Imagen 13 Comunicación con el microcontrolador	16
Imagen 14 Resultados	16
Imagen 15 Control velocidad por espacio de estados	17
Imagen 16 Descripción del transistor TIP122	18
Imagen 17 Dimensiones del transistor TIP122	18
Imagen 18 Descripción del relay OMRON.....	19
Imagen 19 Dimensiones del relay OMRON.....	19
Imagen 20 Diagrama de la etapa de potencia en Proteus.....	20
Imagen 21 Diagrama dado en el datasheet de la conexión del LM331	21
Imagen 22 Diagrama realizado de la conexión del LM331 en Proteus.....	22
Imagen 23 Diagrama de la ganancia de voltaje	22
Imagen 24 Diagrama completo para el sensor de voltaje	23
Imagen 25 Gráfica de voltaje vs corriente	24
Imagen 26 Etapa de acondicionamiento para el sensor de corriente	24
Imagen 27 Etapa de acondicionamiento para el sensor de corriente	25
Imagen 28 Interfaz Gráfica y sus partes.....	26
Imagen 29 Motor 222050	27
Imagen 30 Servomotor HG995	28

1. Introducción:

A lo largo del curso de Ingeniería de Control 2 nos dedicamos a estudiar distintas herramientas y métodos para controlar un sistema como: modelo de espacio de estados, análisis de estabilidad, ecuaciones de estados, análisis de controlabilidad y observabilidad, diseño de controladores y observadores con y sin integrador, control optimal y uso del filtro de Kalman. Cada una de estas nos sirvieron para poder realizar este trabajo final del curso. Tal y como se irá presentando a lo largo del informe, hemos usado la mayoría de conocimientos adquiridos durante el curso para lograr el control de un carrito velocista. Asimismo, se presenta el diseño mecánico en 3D del carro en Inventor, uso del hardware de desarrollo Arduino, control clásico con controlador proporcional y diseño de interfaz gráfica para el control con lenguaje de programación Python.

2. Objetivo general:

Diseño del circuito de un carrito velocista de cuatro ruedas con inversión de giro con un motor DC modelo 222050 y un servomotor MG995.

3. Objetivos específicos:

- ✓ Realizar el modelado de espacios para el control de la velocidad.
- ✓ Analizar los resultados obtenidos de las simulación en MATLAB y Simulink.
- ✓ Simular el circuito en Proteus.
- ✓ Elaborar el diseño mecánico del carrito en el software Inventor.
- ✓ Seleccionar la etapa de potencia.
- ✓ Diseñar la interfaz gráfica para controlar el carrito desde el celular.

4. Desarrollo:

4.1. Modelado de espacio de estados de la velocidad

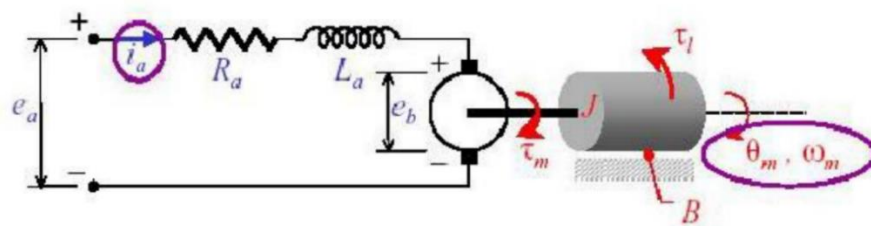


Imagen 1 Modelamiento de espacio de estados del motor DC modelo 222050

Modelo Matemático

Se definen las variables y las constantes mostradas en la figura

$e_b(t)$: *Voltahe contraelectromotriz*

$e_a(t)$: *Voltaje de entrada (V)*

$\theta_m(t)$: *Posición*

$\omega(t)$: *Velocidad*

$i_a(t)$: *Corriente*

R_a : *Resistencia terminal*

L_a : *Inductancia*

k_b : *Constante de fuerza contraelectromotriz del motor*

τ_m : *Torque del motor*

τ_l : *Torque de carga*

B : *Coficiente de friccion viscosa*

Datos del motor DC modelo 222050

$$R_a = 4.05\Omega$$

$$L_a = 0.2 \times 10^{-3}H$$

$$k_b = \frac{30}{557 \times \pi} \frac{V.s}{rad}$$

$$k_T = 17.2 \times \frac{10^{-3}Nm}{A}$$

$$\tau_m = 12.4 \times 10^{-3}Nm$$

$$B = 0,01 \times \frac{12 \times 10^{-3} N.ms}{6650 \times \frac{\pi}{30} rad}$$

$$\omega_N = 6650 \times \frac{\frac{\pi}{30} rad}{s}$$

$$J = 4,28 \times 10^{-3} \times 10^{-4} gcm^2$$

Ecuaciones diferenciales

Por la Ley de voltaje de Kirchhoffen la armadura:

$$e_a(t) = R_a \times i_a \times L_a \times \frac{di_a(t)}{dt} + e_b(t) \dots \dots \dots (1)$$

Voltaje contraelectromotriz:

$$e_b(t) = K_b \times \omega_m(t) \dots \dots \dots (2)$$

Por torque del motor y pérdidas:

$$\tau_m(t) - \tau_l(t) - B \times \omega_m(t) = J \times \frac{d\omega_m(t)}{dt} \dots \dots \dots (3)$$

Torque del motor:

$$\tau_m(t) = K_t \times i_a(t) \dots \dots \dots (4)$$

Definición de las variables de estados

$$x_1(t) = \omega_m(t)$$

$$x_2(t) = i_a(t)$$

Variables de entrada

$$e_a(t)$$

$$\tau_l(t): \text{Perturbación}$$

A partir de las ecuaciones diferenciales y las variables de estados definidas se procede a formar las ecuaciones que determinarán el modelo solicitado.

Despejando la velocidad del motor en la ecuación (3) y reemplazando (4)

$$\frac{d\omega_m(t)}{dt} = \frac{K_t \times i_a(t)}{J} - \frac{\tau_l(t)}{J} - \frac{B \times \omega_m(t)}{J}$$

Despejando la corriente en la ecuación (1) y reemplazando (2)

$$\frac{di_a(t)}{dt} = -\frac{R_a \times i_a(t)}{L_a} + \frac{e_a(t)}{L_a} - \frac{K_b \times \omega_m(t)}{L_a}$$

Reemplazando las variables de estados en las dos últimas ecuaciones obtenidas

$$\dot{x}_1(t) = -\frac{B \times x_1(t)}{J} + \frac{K_t \times x_2(t)}{J} - \frac{\tau_l(t)}{J}$$

$$\dot{x}_2(t) = -\frac{K_b \times x_1(t)}{L_a} - \frac{R_a \times x_2(t)}{L_a} + \frac{e_a(t)}{L_a}$$

Por lo tanto, el modelo de espacios de estados de la velocidad está dada por la siguiente expresión

$$\begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \end{bmatrix} = \begin{bmatrix} -\frac{B}{J} & \frac{K_t}{J} \\ -\frac{K_b}{L_a} & -\frac{R_a}{L_a} \end{bmatrix} \times \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L_a} \end{bmatrix} \times e_a(t) + \begin{bmatrix} -\frac{1}{J} \\ 0 \end{bmatrix} \times \tau_l(t)$$

$$y = [1 \quad 0] \times \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

De donde:

$$AV = \begin{bmatrix} -\frac{B}{J} & \frac{K_t}{J} \\ -\frac{K_b}{L_a} & -\frac{R_a}{L_a} \end{bmatrix} \quad ; \quad BV = \begin{bmatrix} 0 \\ \frac{1}{L_a} \end{bmatrix} \quad ; \quad EV = \begin{bmatrix} -\frac{1}{J} \\ 0 \end{bmatrix}$$

$$CV = [1 \quad 0] \quad ; \quad DV = 0$$

Función de transferencia de la velocidad

$$G(s) = C(sI - A)^{-1}B + D$$

$$G_\omega(s) = [1 \quad 0] \times \left[s \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} -\frac{B}{J} & \frac{K_t}{J} \\ -\frac{K_b}{L_a} & -\frac{R_a}{L_a} \end{bmatrix} \right]^{-1} \times \begin{bmatrix} 0 \\ \frac{1}{L_a} \end{bmatrix} + 0$$

$$G_\omega(s) = [1 \quad 0] \times \begin{bmatrix} s + \frac{B}{J} & -\frac{K_t}{J} \\ \frac{K_b}{L_a} & s + \frac{R_a}{L_a} \end{bmatrix}^{-1} \times \begin{bmatrix} 0 \\ \frac{1}{L_a} \end{bmatrix}$$

$$G_\omega(s) = [1 \quad 0] \times \begin{bmatrix} \frac{s + \frac{R_a}{L_a}}{s^2 + \left(\frac{B}{J} + \frac{R_a}{L_a}\right)s + \frac{B}{J} \times \frac{R_a}{L_a} + \left(\frac{K_t}{J} \times \frac{K_b}{L_a}\right)} & \frac{-\frac{K_t}{J}}{s^2 + \left(\frac{B}{J} + \frac{R_a}{L_a}\right)s + \left(\frac{K_t}{J} \times \frac{K_b}{L_a}\right)} \\ \frac{\frac{K_b}{L_a}}{s^2 + \left(\frac{B}{J} + \frac{R_a}{L_a}\right)s + \frac{B}{J} \times \frac{R_a}{L_a} + \left(\frac{K_t}{J} \times \frac{K_b}{L_a}\right)} & \frac{s + \frac{B}{J}}{s^2 + \left(\frac{B}{J} + \frac{R_a}{L_a}\right)s + \left(\frac{K_t}{J} \times \frac{K_b}{L_a}\right)} \end{bmatrix}^{-1} \times \begin{bmatrix} 0 \\ \frac{1}{L_a} \end{bmatrix}$$

$$G_{\omega}(s) = - \frac{\left(-\frac{K_t}{J}\right)s}{s^3 + \left(\frac{B}{J} + \frac{R_a}{L_a}\right)s^2 + \left(\frac{B}{J}\left(\frac{R_a}{L_a}\right) - \left(-\frac{K_t}{J}\right)\frac{K_b}{L_a}\right)s} \times \frac{1}{L_a}$$

$$\therefore G_{\omega}(s) = \frac{\frac{K_t}{J \times L_a}}{s^2 + \left(\frac{B \times L_a + R_a \times J}{J \times L_a}\right)s + \left(\frac{B \times R_a + K_t \times K_b}{J \times L_a}\right)}$$

Función de transferencia de la posición del servomotor DC

A partir de las ecuaciones diferenciales y las variables de estados definidas previamente se procede a formar las ecuaciones que determinarán el modelo solicitado.

Reordenando las ecuaciones (5) y (6)

$$J \times \frac{d\omega_m(t)}{dt} + B \times \omega_m(t) = K_t \times i_a(t) - \tau_l(t)$$

$$L_a \times \frac{di_a(t)}{dt} + R_a \times i_a(t) = e_a(t) - K_b \times \omega_m(t)$$

Despreciando la perturbación y aplicando la Transformada de Fourier a ambas ecuaciones

$$s(J \times s + B) \times W(s) = K_t \times I_a(s)$$

$$(L_a \times s + R_a) \times I_a(s) = E_a(s) - K_b \times s \times W_m(s)$$

Eliminando $I(s)$ de las ecuaciones se obtiene la función de transferencia de la velocidad de salida respecto al voltaje de entrada

$$\frac{W_m(s)}{E_a} = \frac{K_t}{(J \times s + B)((L_a \times s + R_a) + K_b^2)}$$

Para determinar la función de transferencia de la posición solo se agrega un integrador (1/s) a la función de la velocidad.

$$\frac{\theta_m(s)}{E_a} = \frac{K_t}{s \times [(Js + B)((L_a s + R_a) + K_t K_b)]}$$

$$\therefore G(s) = \frac{K_t}{JL_a s^3 + (J \times R_a + L_a \times B)s^2 + (K_t K_b + R_a B)s}$$

Diagrama de bloques del servo motor

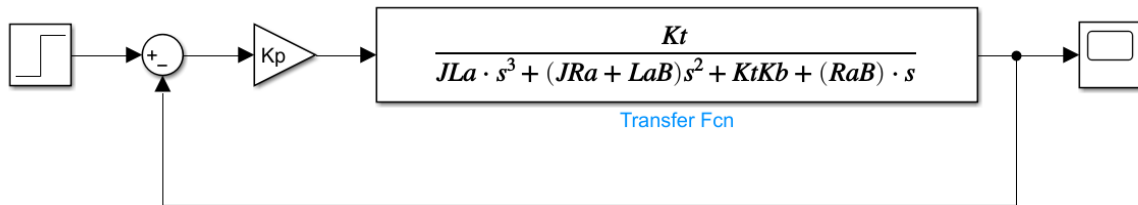


Imagen 2 Diagrama de bloques del controlador proporcional

Debido a que los datos técnicos encontrados del servomotor son insuficientes para ser reemplazados en la función de transferencia hallada se optó por diseñar un controlador proporcional a partir de las pruebas realizadas con el servomotor en Proteus.

4.2. Diseño mecánico

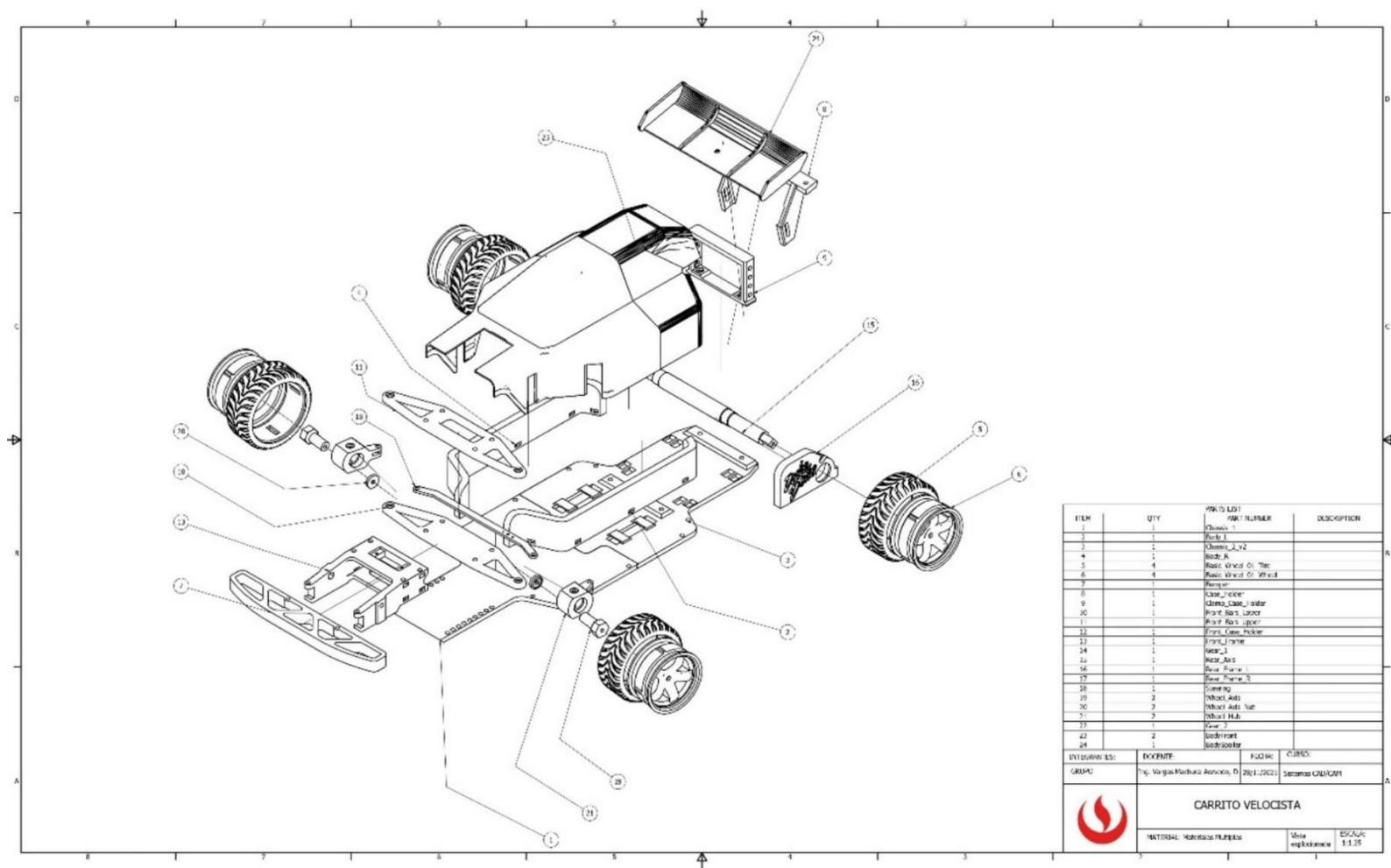


Imagen 3 Plano de explosión del diseño mecánico del carrito

4.3. Diseño electrónico

4.3.1. DISEÑO EN SIMULINK PARA EL CONTROL DEL MOTOR 222050 DC

CONTROLADOR DE VELOCIDAD CON GANANCIA OPTIMAL MÁS INTEGRADOR MOTOR 222050

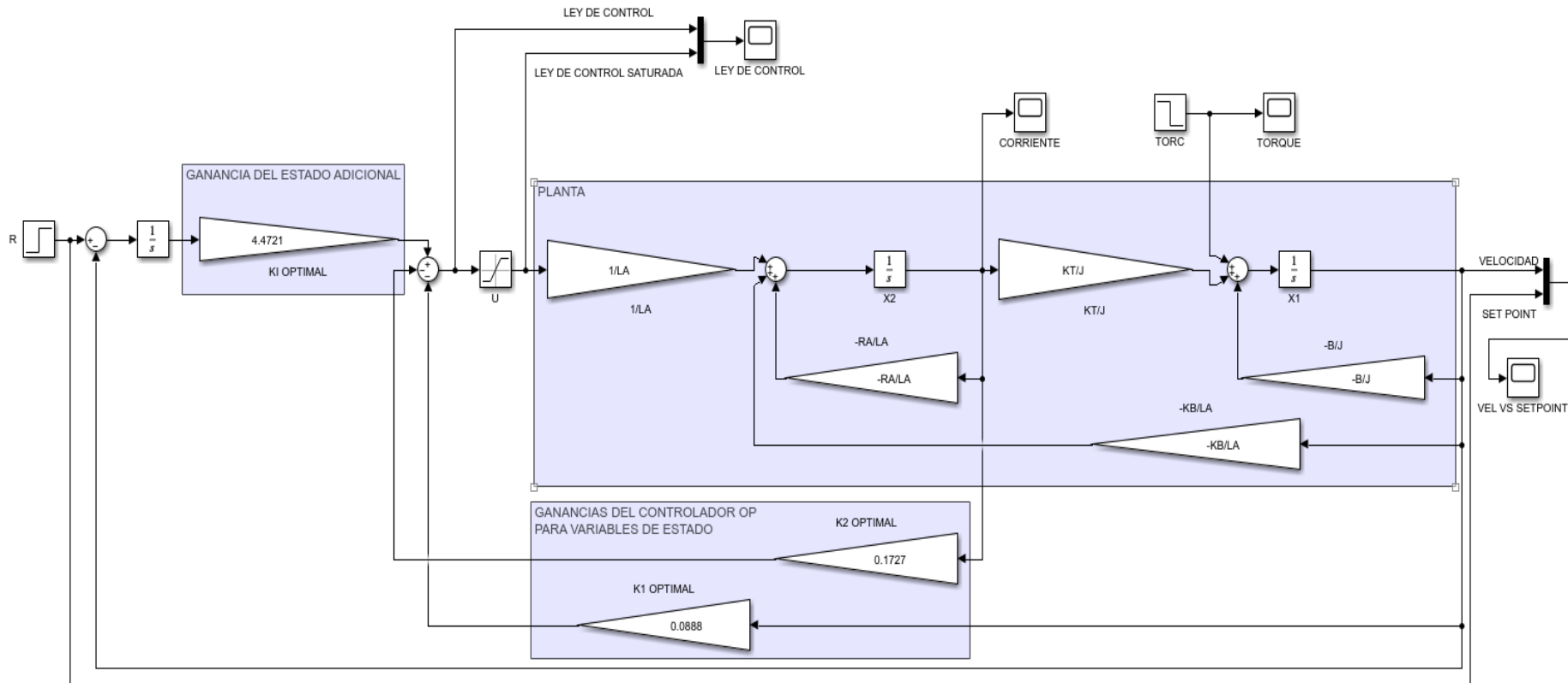


Imagen 4 Simulación en Simulink del controlador de velocidad con ganancia optimal

4.3.2. RESULTADOS EN SIMULINK Y MATLAB

4.3.2.1. Corriente Simulink

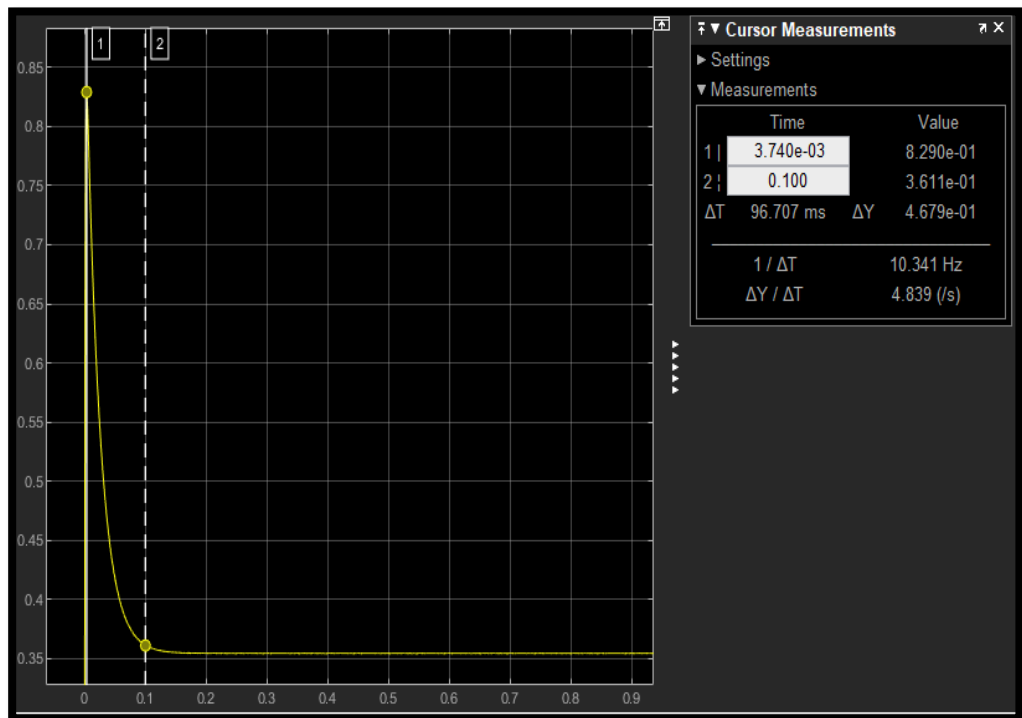


Imagen 5 Gráfica de la corriente en Simulink

4.3.2.2. Corriente Matlab

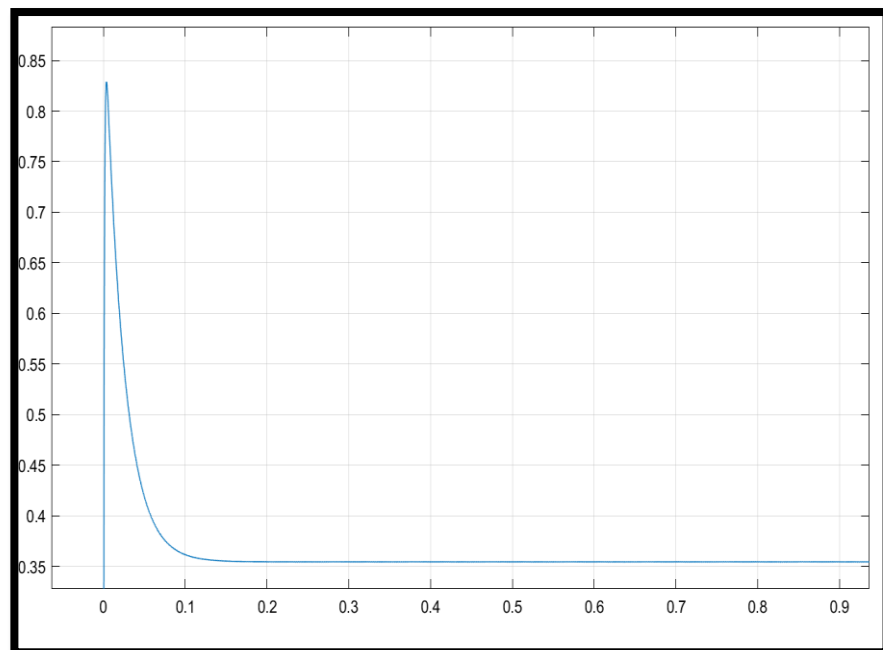


Imagen 6 Gráfica de la corriente en MATLAB

4.3.2.3. Ley de control Simulink

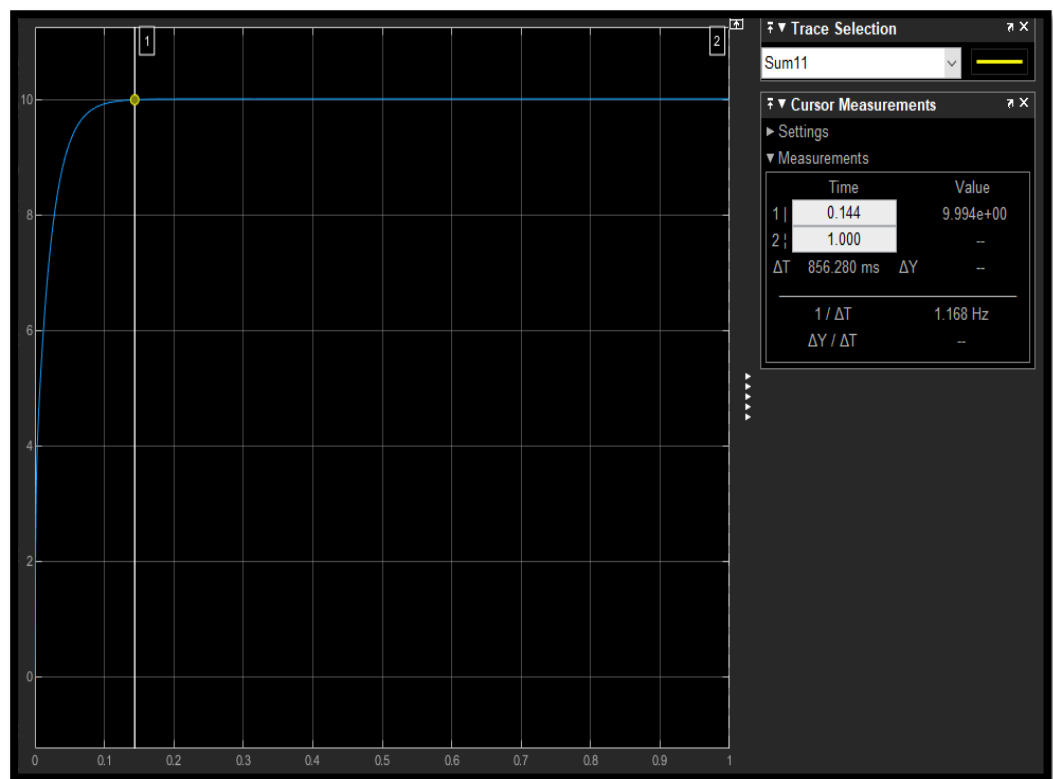


Imagen 7 Gráfica de la Ley de control en Simulink

4.3.2.4. Ley de control Matlab

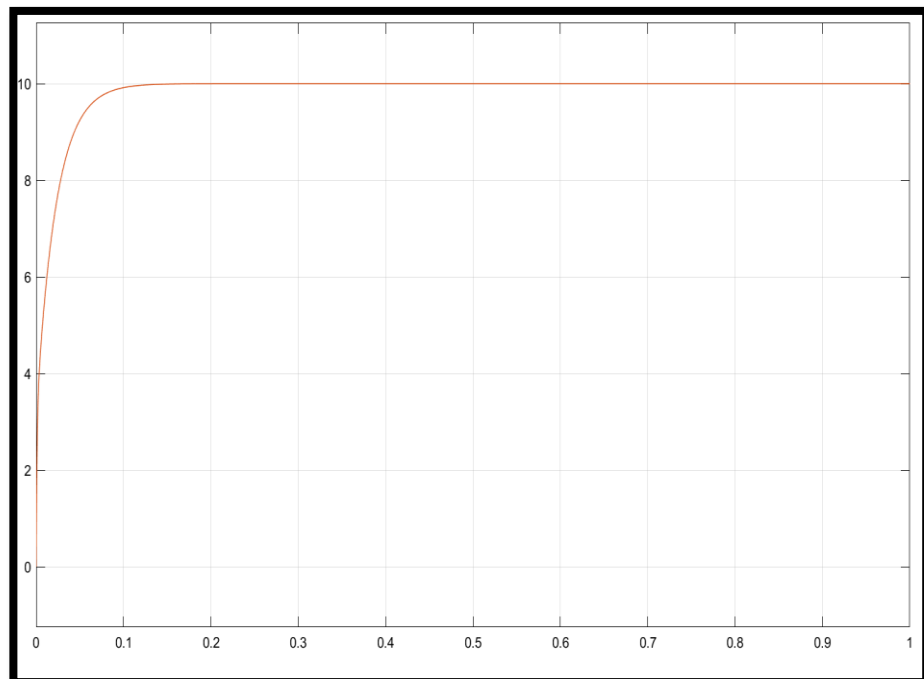


Imagen 8 Gráfica de la Ley de Control en MATLAB

4.3.2.5. Set point vs velocidad (out put) Simulink

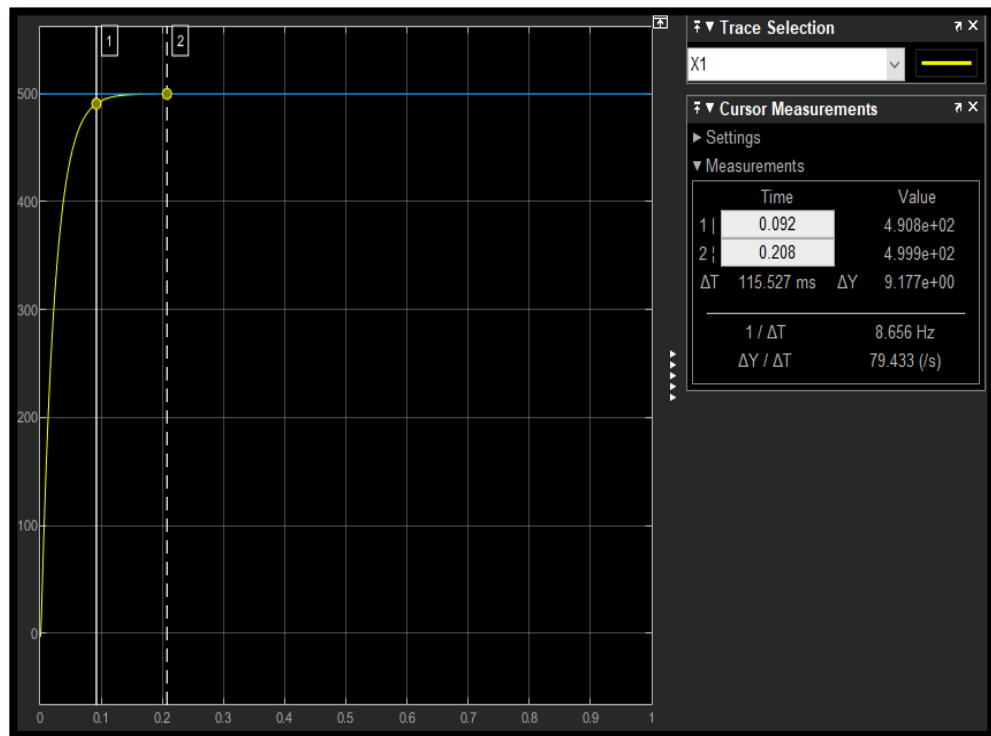


Imagen 9 Gráfica de Set point vs velocidad en Simulink

4.3.2.6. Set point vs velocidad (out put) Matlab

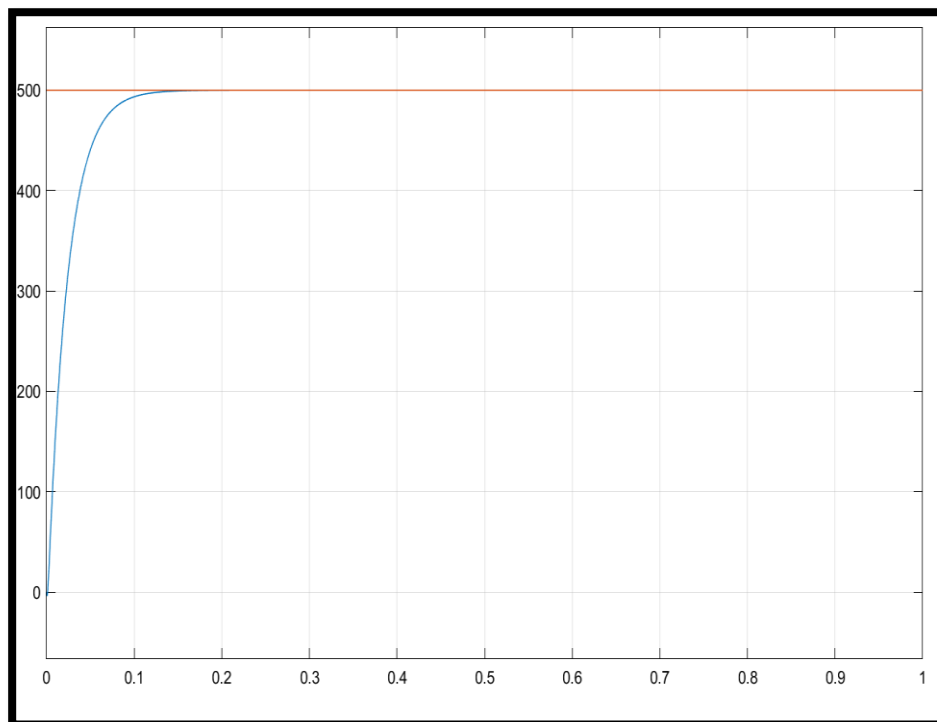


Imagen 10 Gráfica de set point vs velocidad en MATLAB

4.3.2.7. Resultados en matlab

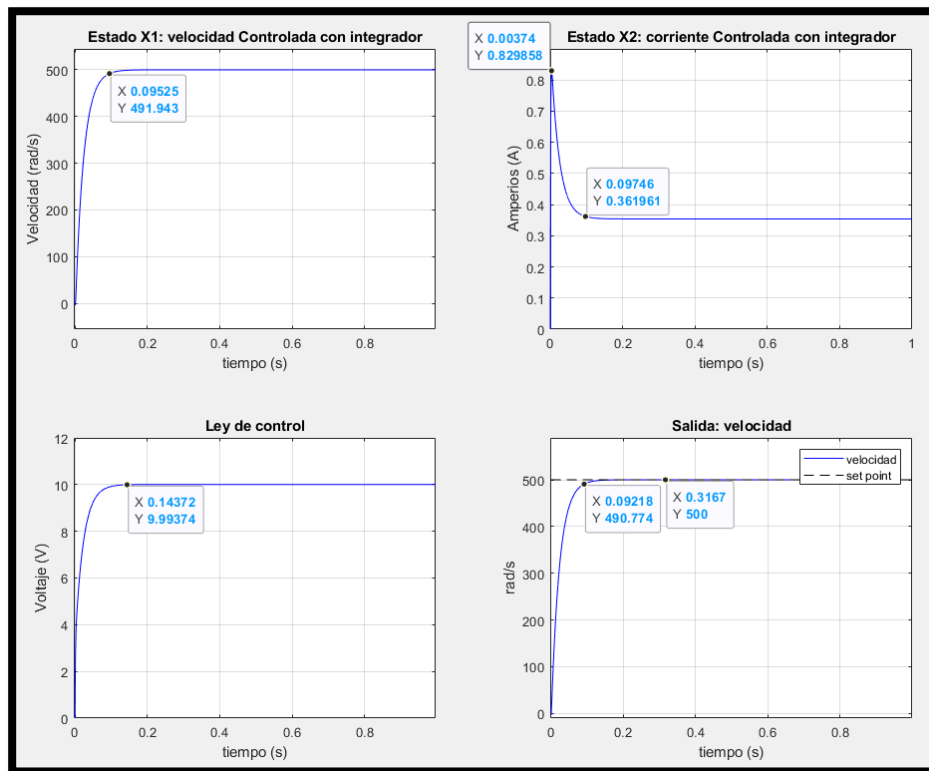


Imagen 11 Resultados de la velocidad, corriente y el voltaje en MATLAB

4.4. Diseño en Proteus

CONTROL PROPORCIONAL PARA LA POSICION

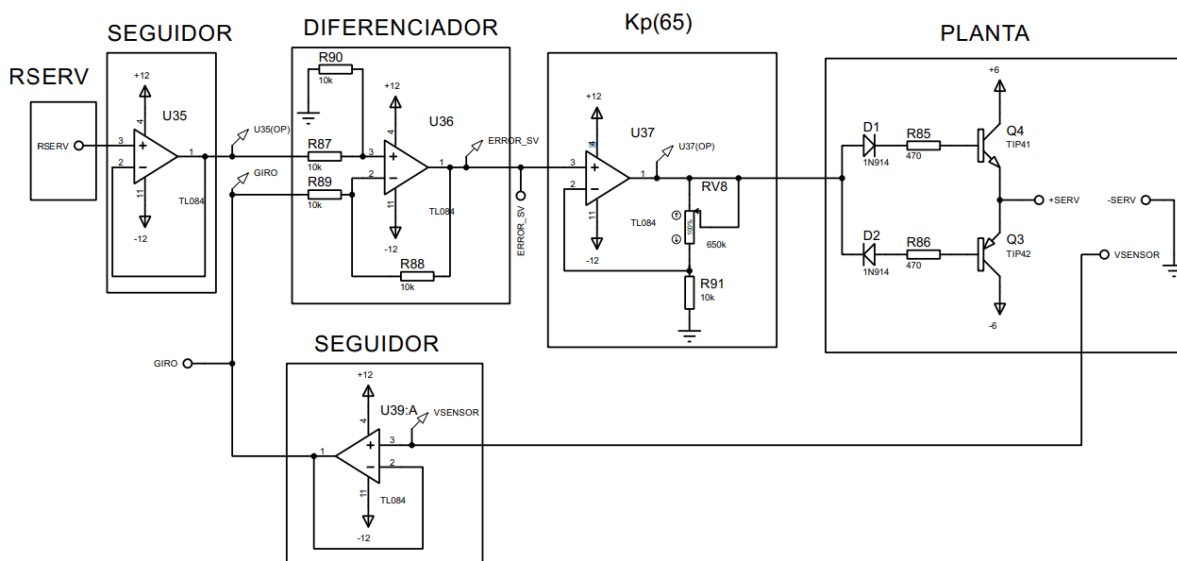


Imagen 12 Control proporcional para la posición

COMUNICACIÓN CON MICROCONTROLADOR

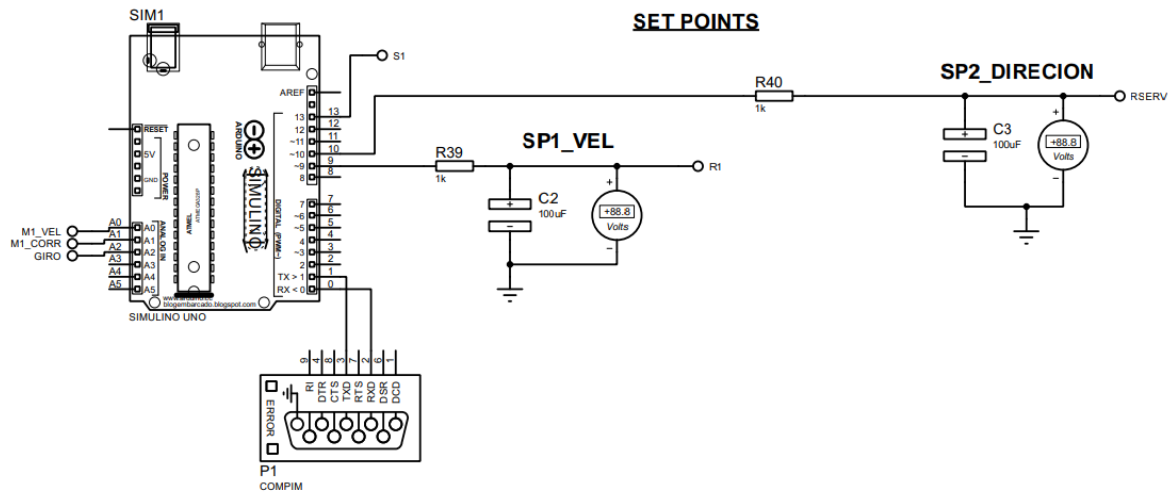
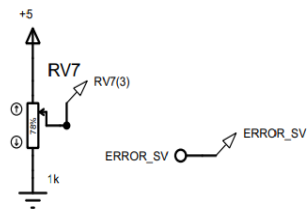


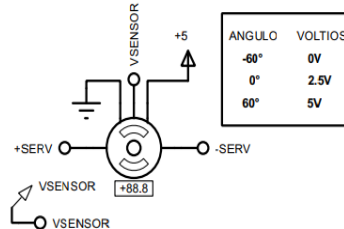
Imagen 13 Comunicación con el microcontrolador

RESULTADOS

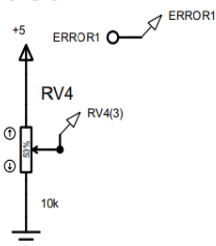
RSERV



SERVOMOTOR



R1



MOTOR

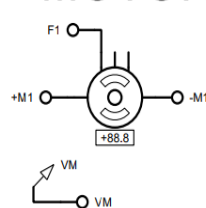


Imagen 14 Resultados

CONTROL DE VELOCIDAD POR ESPACIO DE ESTADOS

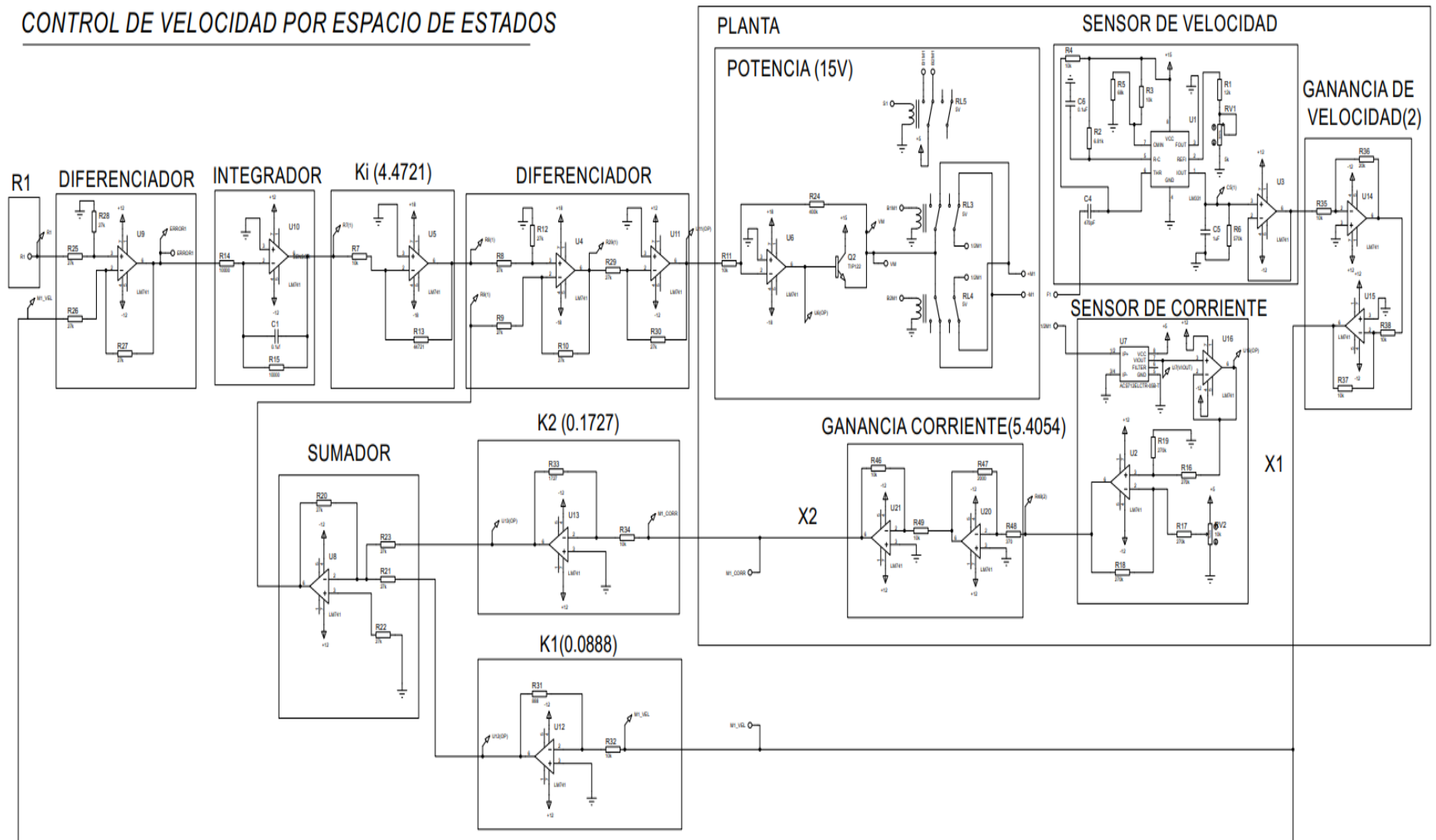


Imagen 15 Control velocidad por espacio de estados

4.5. Selección de la etapa de potencia

4.5.1. Transistor TIP122:

Este transistor es generalmente utilizado en la parte de potencia de un circuito para amplificar la corriente del circuito de control y obtener su equivalente para alta potencia. Esta amplificación se logra debido a su configuración Darlington, esta es una conexión entre dos transistores BJT con el emisor del primer transistor y la base del segundo transistor BJT. Con esto se logra obtener una ganancia equivalente a la multiplicación de las dos ganancias de los dos transistores internos.

Inchange Semiconductor	Product Specification
Silicon NPN Darlington Power Transistors	TIP120/121/122

DESCRIPTION

- With TO-220C package
- DARLINGTON
- High DC current gain
- Low collector saturation voltage
- Complement to type TIP125/126/127

APPLICATIONS

- Designed for general-purpose amplifier and low-speed switching applications.

PINNING

PIN	DESCRIPTION
1	Base
2	Collector, connected to mounting base
3	Emitter

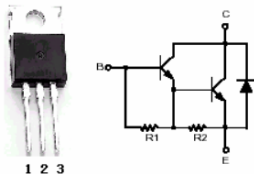


Fig.1 simplified outline (TO-220C) and symbol

Imagen 16 Descripción del transistor TIP122

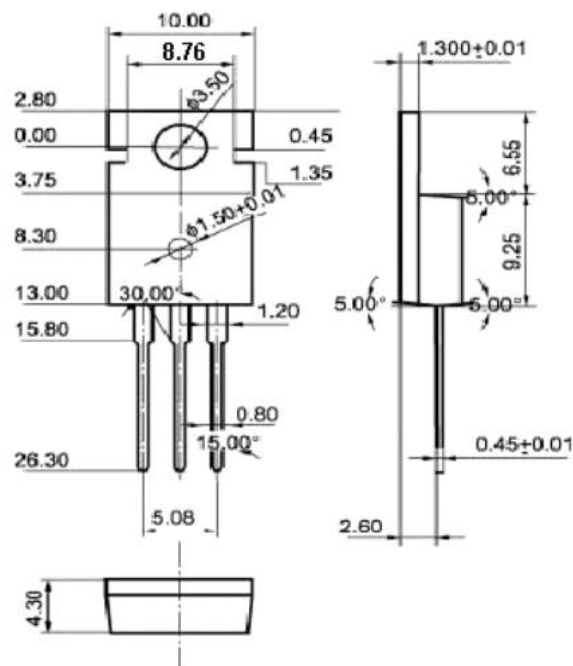


Imagen 17 Dimensiones del transistor TIP122

4.5.2. Relay G2R-2:

Este dispositivo se utilizó para poder conmutar la fuente que va hacia el motor y así obtener las velocidades negativas respectivas al set point, Se escogió este modelo debido a sus características, los cuales cumplen con nuestros requerimientos tales como un par de juego de contactos por relay, el voltaje de alimentación de este, 5v, y su velocidad de energización de bobina para obtener una respuesta rápida el cual es 7ms.

OMRON

PCB Relay

G2R

A Power Relay for a Variety of Purposes with Various Models

- Conforms to VDE0435 (VDE approval: C250 insulation grade), UL508, CSA22.2, SEV, SEMKO.
- Meets VDE0700 requirements for household products according to VDE0110.
- Clearance and creepage distance: 8 mm/8 m.
- Models with CTI250 material available.
- High-sensitivity (360 mW) and high-capacity (16 A) types available.
- Double-winding latching type available.
- Plug-in with test button and quick-connect terminals available.
- Highly functional socket available.

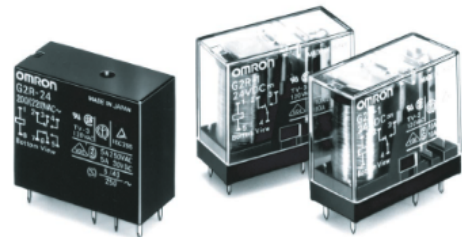


Imagen 18 Descripción del relay OMRON

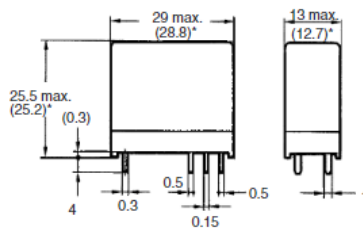
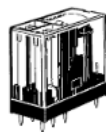
G2R

OMRON

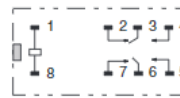
G2R

Relays with PCB Terminals

DPDT Relays
G2R-2, G2R-2-H



Terminal Arrangement/ Internal Connections (Bottom View)



Mounting Holes (Bottom View)

Tolerance: ± 0.1

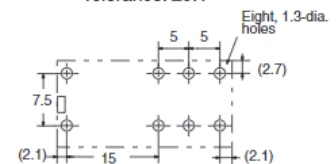


Imagen 19 Dimensiones del relay OMRON

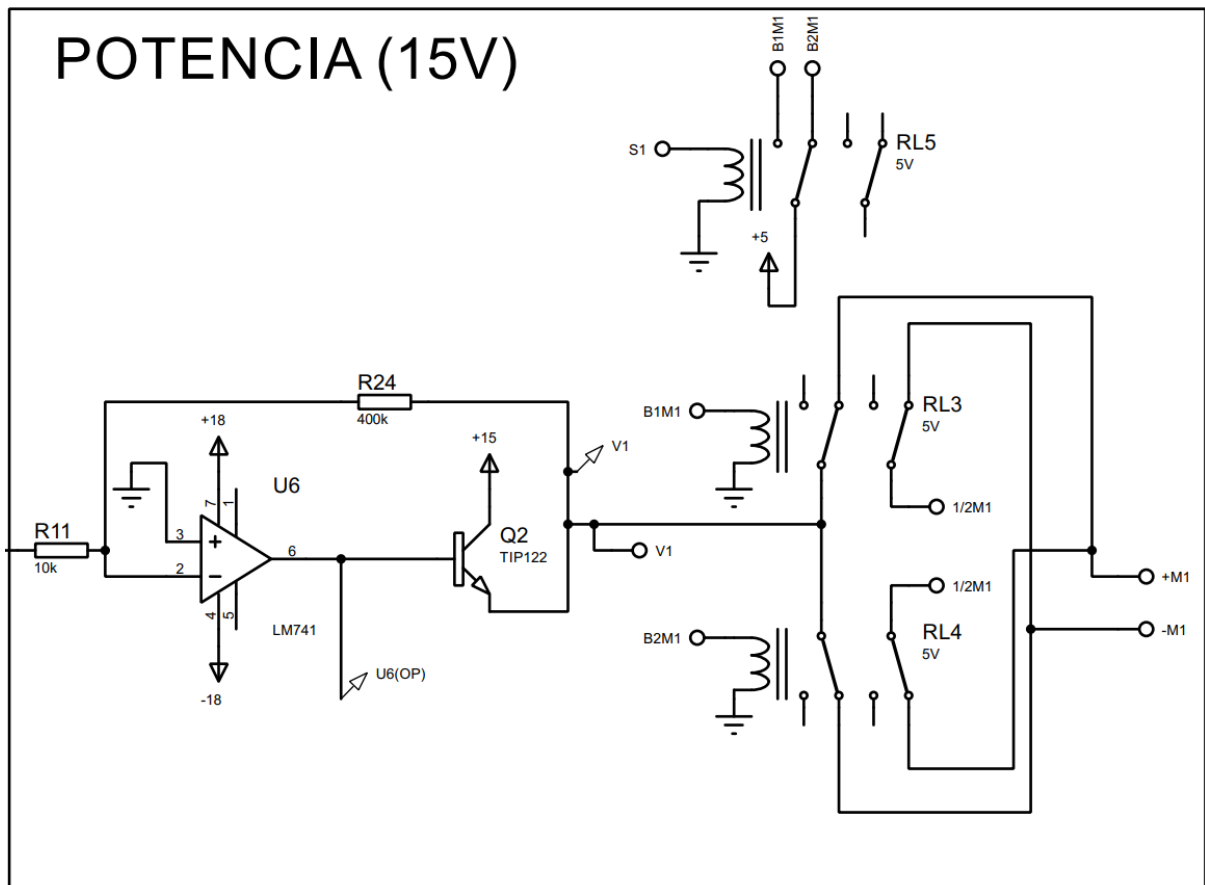


Imagen 20 Diagrama de la etapa de potencia en Proteus

4.6. Sensores a utilizar

4.6.1. Sensor de velocidad: LM331

Este sensor será utilizado para obtener la velocidad del motor. Para ello, se usará la configuración mostrada en la siguiente figura, la cual convierte la frecuencia de entrada (fin) en una medida de voltaje (Vout).

Datos técnicos:

- Temperatura operacional: 0 a 70 °C
- Voltaje de suministro: máximo 40 V
- Frecuencia de escala completa: 1 Hz a 100 kHz

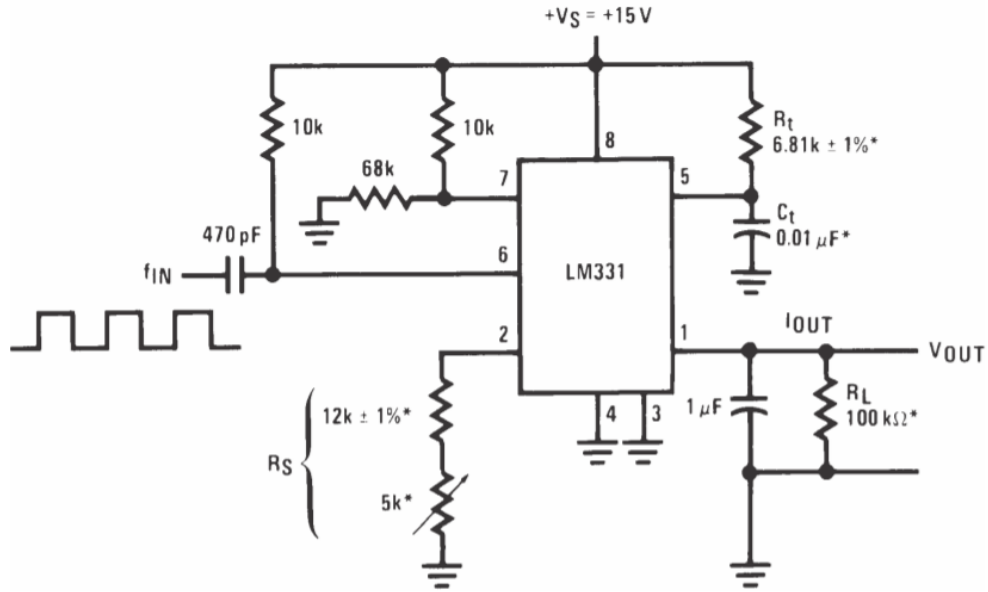


Imagen 21 Diagrama dado en el datasheet de la conexión del LM331

La relación de los parámetros mencionados se rige según la ecuación:

$$V_{OUT} = f_{IN} \times 2.09 \times \frac{R_L}{R_S} \times R_t C_t$$

Variables utilizadas:

$$R_L = 570 \text{ k}\Omega$$

$$R_S = 12\text{k} + 1 \times 5 \text{ k} = 17 \text{ k}\Omega$$

$$R_t = 6.81\text{k}\Omega$$

$$C_t = 0.1 \text{ }\mu\text{F}$$

Por lo tanto:

$$V_{OUT} = f_{IN} \times 2.09 \times \frac{570 \text{ k}\Omega}{17 \text{ k}\Omega} \times (6.81 \text{ k}\Omega)(0.1 \text{ }\mu\text{F})$$

$$\therefore V_{OUT} = (0.047722)f_{IN}$$

Circuito simulado en Proteus

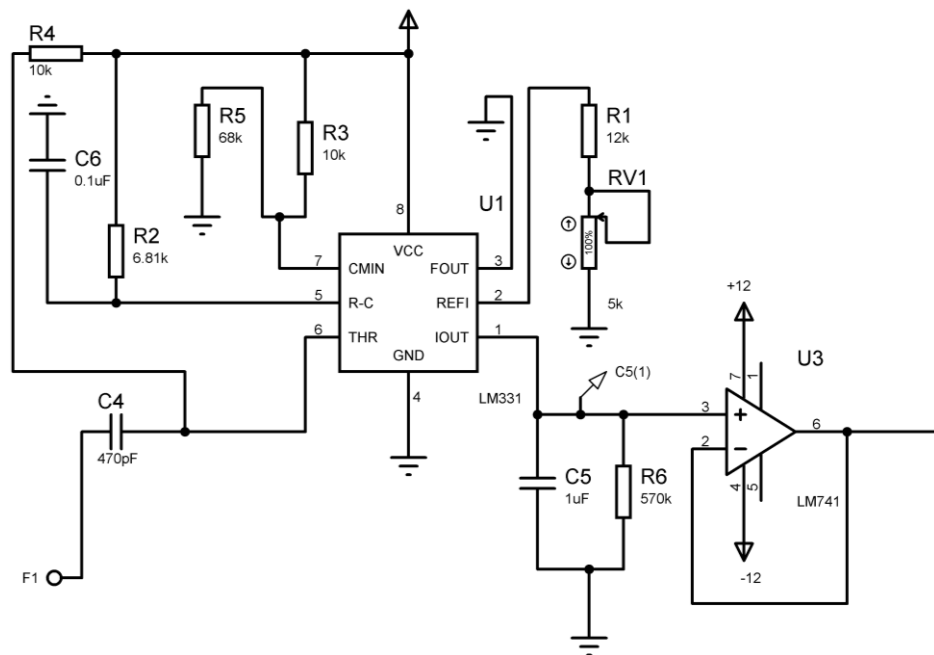


Imagen 22 Diagrama realizado de la conexión del LM331 en Proteus

Se añadirá un seguidor de voltaje a la salida del sensor con el objetivo de pasar de una etapa de alta impedancia a una de baja impedancia. Asimismo, de acuerdo al voltaje obtenido del sensor se agregó un amplificador con una ganancia de 2 seguido de un amplificador inversor para obtener el valor correspondiente al ingresado al motor.

$$G_{vel} = 2$$

GANANCIA DE VELOCIDAD

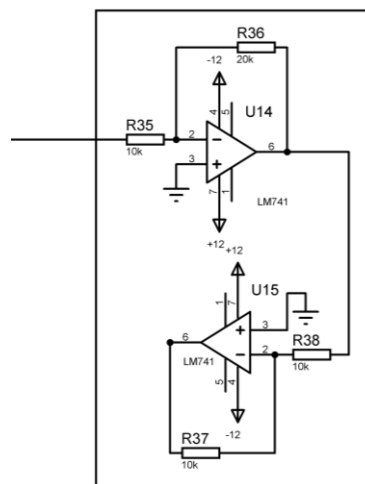


Imagen 23 Diagrama de la ganancia de voltaje

Finalmente, el circuito obtenido es el siguiente:

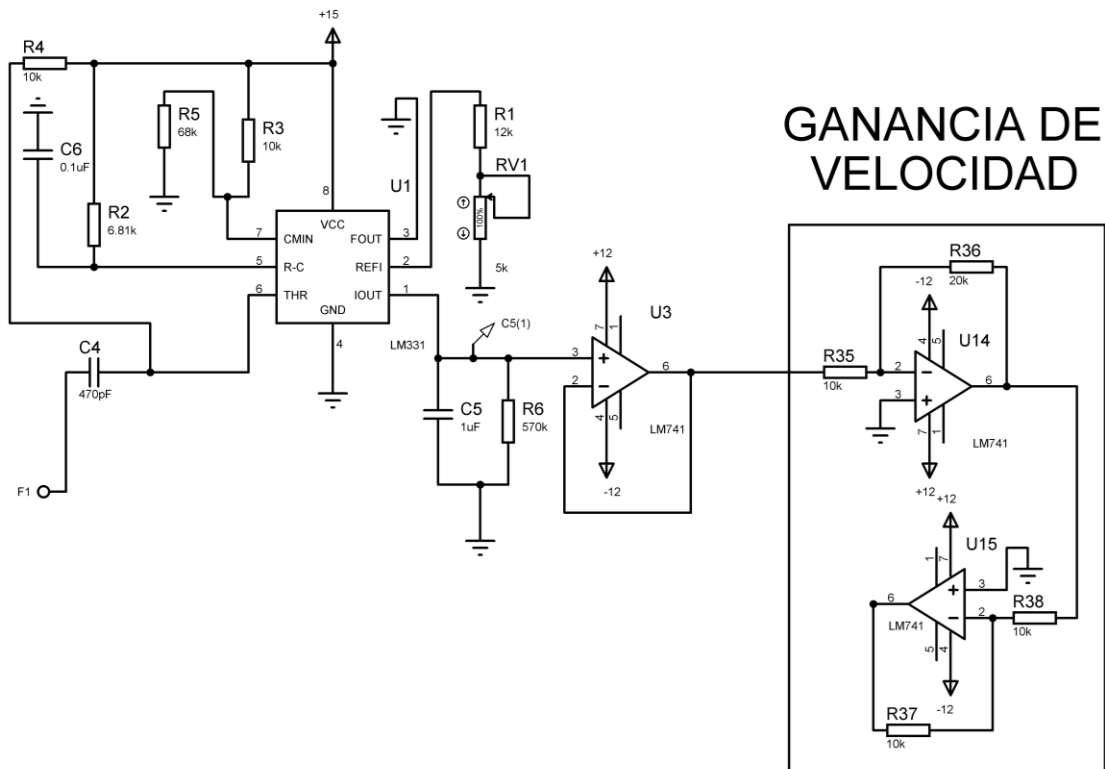


Imagen 24 Diagrama completo para el sensor de voltaje

4.6.2. Sensor de corriente: ACS712ELCTR-05B-T

Este sensor, convertirá la corriente que salga del motor y la convertirá en voltaje.

Datos técnicos:

- Temperatura operacional: -40 a 85 °C
- Sensibilidad: 185 mV/A
- Rango optimizado: ± 5 A
- Desfasaje

Output Voltage versus Sensed Current

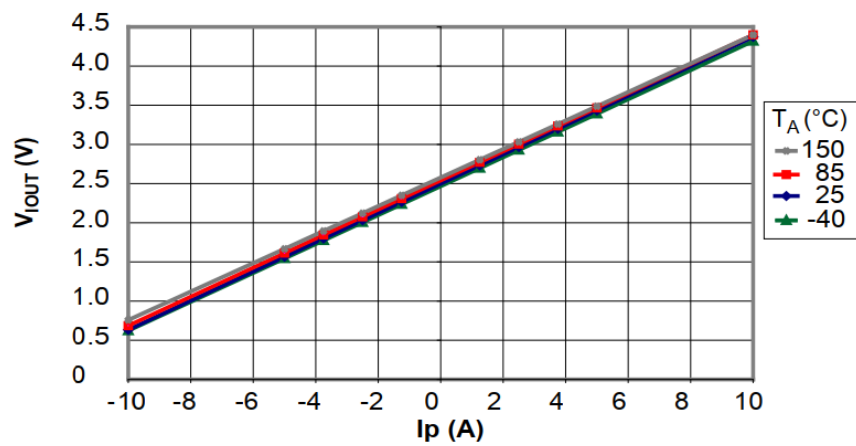


Imagen 25 Gráfica de voltaje vs corriente

De acuerdo a la gráfica, se confirma la linealidad del sensor, y además se observa que su salida presenta un desfase de 2,5 V. Es decir, para una corriente de entrada de 0 A la salida genera un voltaje de 2,5 V. Por ello, se agregará una etapa de acondicionamiento para compensar el desfase, la cual consiste en un amplificador diferenciador.

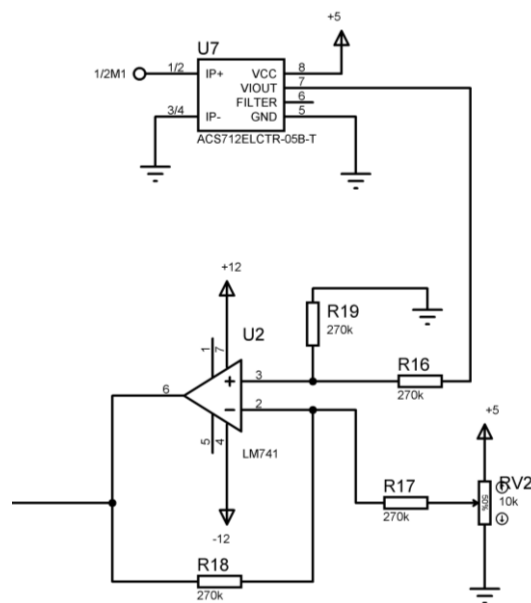


Imagen 26 Etapa de acondicionamiento para el sensor de corriente

Dado que el sensor añade una ganancia a su salida de 0,185 debido a su sensibilidad de acuerdo a su datasheet, se procede a agregar una ganancia inversa a este valor para convertir el valor medido al sistema internacional.

$$G_{corr} = \frac{2000}{370} = \frac{1}{0.185} = 5,4054$$

Circuito final de la medición de corriente

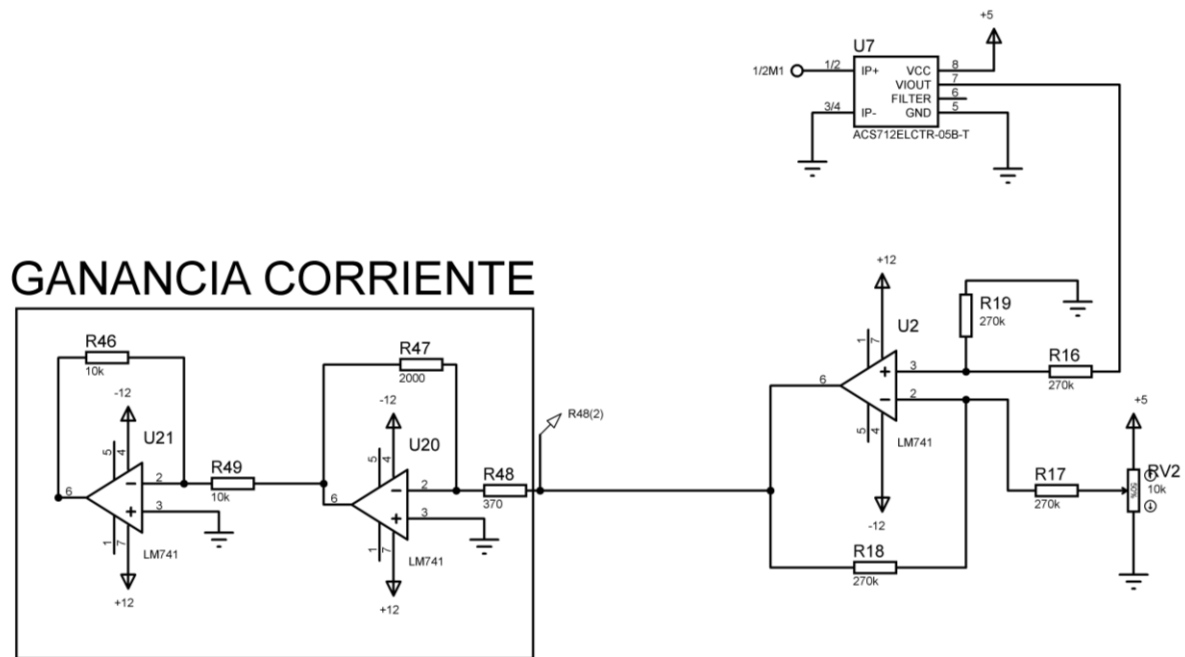


Imagen 27 Etapa de acondicionamiento para el sensor de corriente

4.7. Conexión via celular

Para realizar esto se realizaron los siguientes puntos:

4.7.1. Conexión Serial

La comunicación serial se refiere a un puerto utilizado para conectar dos dispositivos digitales. Se utilizan pines RX y TX. Por ejemplo, dependiendo del tipo de tarjeta, puede haber uno o más puertos. Estos puertos constan de pines llamados RX y Tx. En general los datos transmitidos se cifran mediante caracteres.

Normalmente, en la comunicación en serie con Arduino, se envían caracteres codificados en ASCII. Estos caracteres representan la mayoría de los caracteres utilizados en idiomas que utilizan caracteres occidentales. Por ejemplo, este conjunto de caracteres está indicado por el estándar UTF-8.

4.7.2. Diseño de interfaz gráfica

Se diseñó una interfaz gráfica con el lenguaje de programación Phyton para poder controlar los movimientos del carrito.

Sus partes son:



Imagen 28 Interfaz Gráfica y sus partes

4.8. Motores utilizados

4.8.1. Motor 222050



Imagen 29 Motor 222050

Datos del motor DC modelo 222050

$$R_a = 4.05\Omega$$

$$L_a = 0.2 \times 10^{-3}H$$

$$k_b = \frac{30}{557 \times \pi} V \cdot \frac{s}{rad}$$

$$k_T = 17.2 \times \frac{10^{-3}Nm}{A}$$

$$\tau_m = 12.4 \times 10^{-3}Nm$$

$$B = 0.01 \times \frac{12 \times 10^{-3}}{6650 \times \frac{\pi}{30}} \frac{N.ms}{rad}$$

$$\omega_N = 6650 \times \frac{\frac{\pi}{30}rad}{s}$$

$$J = 4.28 \times 10^{-3} \times 10^{-4}gcm^2$$

4.8.2. Servomotor

Se usa el modelo High Speed Metal Gear Dual Ball Brearing Servo MG9995

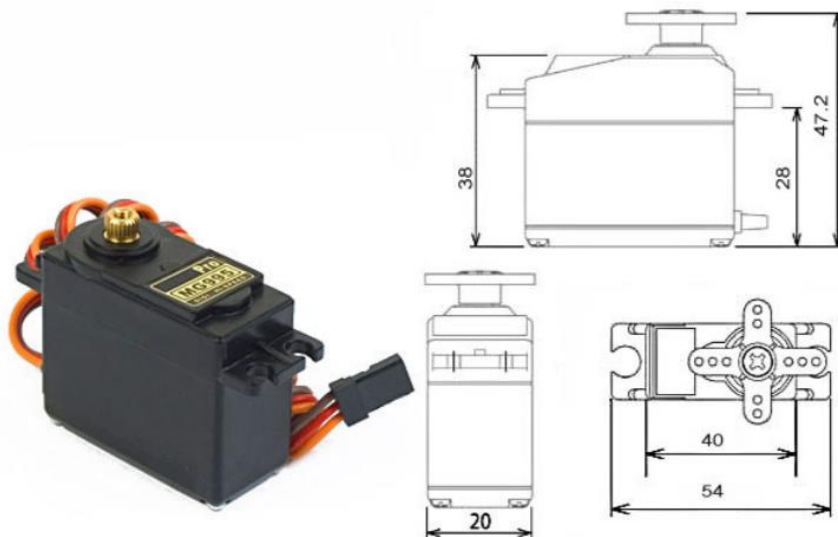


Imagen 30 Servomotor HG995

Este servomotor tiene las siguientes especificaciones:

Peso	55g
Par de parada	$8.5 \text{ kgf} * \text{cm} (4.8 \text{ V})$, $10 \text{ kgf} * \text{cm} (6 \text{ V})$
Velocidad de operación	$0.2 \frac{\text{s}}{60^\circ} (4.8 \text{ V})$, $0.16 \frac{\text{s}}{60} (6 \text{ V})$
Voltaje de operación	4.8V a 7.2V
Ancho de banda muerto	$5 \mu\text{s}$
Rango de temperatura	$0^\circ\text{C} - 55^\circ\text{C}$

5. Conclusiones:

Para el control de velocidad de un motor DC es conveniente implementarlo con un integrador adicional en el modelo de espacio de estados, puesto que si el torque u perturbación en la planta es considerable, la señal de salida, contralada sin integrador, en el estado estable resultará diferente al set point entre más torque u perturbación haya.

Para poder girar a la derecha y a la izquierda se utilizó un servomotor con un controlador proporcional, debido a que se controla la posición y basta con un proporcional para poder obtener error cercano a 0.

Se usó el controlador Arduino, ya que facilita la realización del proyecto. Además, porque tenemos mucha más bibliografía con este lenguaje de programación.

Para el diseño de la interfaz gráfica se utilizó el lenguaje de programación, ya que es un lenguaje que tiene muchas más librerías que nos facilitan el trabajo.

Se implementaron relés para el control del giro del motor, en vez de transistores, ya que estos últimos consumían tanta corriente que no era posible alimentar adecuadamente al motor.

Al inicio se consideró un motor de voltaje nominal de 42V, pero este se cambió por uno de voltaje nominal de 15V, puesto que los OPAMPS se saturaban antes de llegar a 42V, por lo que no era posible llegar a esa cantidad de voltaje con amplificadores comunes.

6. Bibliografía

Allegro. (s. f.). Fully Integrated, Hall Effect-Based Linear Current Sensor with 2.1 kVRMS Voltage Isolation and a Low-Resistance Current Conductor. Recuperado de <https://pdf1.alldatasheet.com/datasheet-pdf/view/168327/ALLEGRO/ACS712ELCTR-05B-T.html> [Consulta: 14 de noviembre de 2021].

Callejas, M., Chávez, C., Díaz, L. & Hernández, M. (2019). Control de Posición de un Motor DC por Asignación de Polos. Recuperado de <http://200.79.179.163/reia/descargables/2019/20-23.pdf> [Consulta: 27 de noviembre de 2021].

Carnegie Mellon (s.f.) Control Tutorials for Matlab. Recuperado de: <https://www3.diism.unisi.it/~control/ctm/examples/motor2/PID2.html> pdf [Consulta: 27 de noviembre de 2021].

Daniel Melchor (24 de mayo del 2021) CONTROL DE POSICIÓN DE UN MOTOR DE DC || CONTROL I || PROYECTO FINAL. [Archivo de video] Youtube. https://www.youtube.com/watch?v=wUfBg_5iXAs

Inchange Semiconductor Company Limited. TIP122 Datasheet. Recuperado de: <https://pdf1.alldatasheet.com/datasheet-pdf/view/441688/ISC/TIP122.html>

Omron Electronics LLC. GR2-1 Datasheet. Recuperado de: <https://pdf1.alldatasheet.com/datasheet-pdf/view/230619/OMRON/G2R-1A.html>

Servomotor HG995 (s.f) High Speed Metal Gear Dual Ball Bearing Servo. Recuperado de: https://www.electronicoscaldas.com/datasheet/MG995_Tower-Pro.pdf

Texas Instruments (s. f.). LM231A/LM231/LM331A/LM331 Precision Voltage-to-Frequency Converters. Recuperado de <http://pdf.datasheetcatalog.net/datasheet/texasinstruments2/lm331.pdf> [Consulta: 14 de noviembre de 2021].

7. Anexos

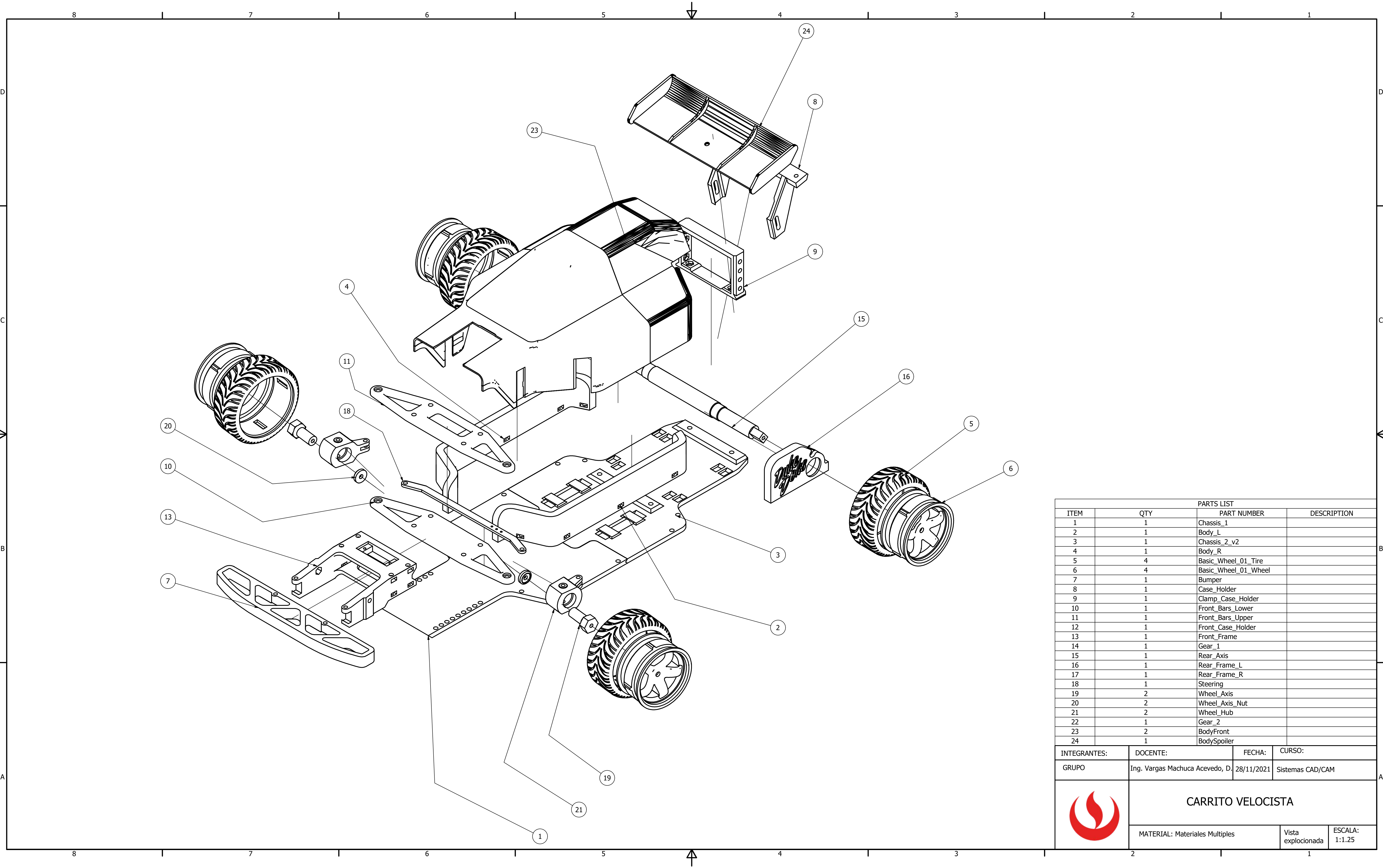
7.1. Anexo 1: Código de MATLAB de la velocidad optimal

```
1. close all; clear; clc
2.
3. %%Modelo del motor 222050
4. %Parametros del motor
5.
6. LA = 0.2 * 10^(-3); %Inductancia (Unidad : henrios H)
7. RA = 4.05; %Resistencia (Unidad : ohms)
8. KT = 17.2 * 10^(-3); %Torque constant (Unidad : N - m/A)
9. KB = 1/((557) * (pi)/30); %Speed constant (Unidad : V.s/rad)
10. J = 4.28 * (10^(-7)); %Inercia del rotor (Unidad : Kg.m^2)
11. TN = 12.4 * 10^(-3); %Torque nominal (Unidad : N - m)WN = (5850 * (pi))/30;
12. WN = (6650 * (pi))/30; %Velocidad angular(rad/s) (Unidad : rad/s)
13. B = (0.01 * TN)/WN; %Friccion del sistema (Unidad : N - ms/rad)
14.
15.
16. %%Modelo de estados
17. AV = [-B/J KT/J
18.      -KB/LA -RA/LA];
19.
20. BV = [0; 1/LA];
21.
22. CV = [1 0];
23.
24. DV = 0;
25.
26. %%Calcular K óptimo
27.
28. %Matrices del Sistema con Integrador:
29. Ai = [ AV zeros(2,1); -CV 0 ];
30. Bi = [ BV; 0 ];
31. Ci = [ CV 0 ];
32. Di = 0;
33. Wi = [ zeros(2,1); 1 ];
34.
35. %Cálculo del Controlador.
36. Q = diag([10 1 20000])/1000;
37. R = 1;
38. Km = lqr(Ai,Bi,Q,R);
39.
40. disp('Los K óptimos son: ')
41. K = Km(1,1:2) %#ok
42. Ki = -Km(1,3) %#ok
43.
44. ti = 0; dt = 0.00001; tf = 1;
45. %Condiciones Iniciales.
46. x = [0 0 0]';
47. u = 0;
48. r = 500;
49. TORC = 6 * 10^ - 3;
50.
51. %Simulación:
52. k = 1;
53. for t = ti:dt:tf
54.     TI(k,1) = t;
```

```

55. X1(k,1) = x(1,1);
56. X2(k,1) = x(2,1);
57. X3(k,1) = x(3,1);
58. U(k,1) = u;
59. xp = Ai * x + Bi * u + Wi * r + [-1/J; 0; 0] * TORC;
60. y = Ci * x + Di * u;
61. % ley de control
62. if (u > 15)
63.     u = 15;
64. end
65.
66. if (u < -15)
67.     u = -15;
68.     end
69. u = -K * x(1:2,1) + Ki * x(3,1);
70. Y(k,1) = y;
71. x = x + xp * dt;
72. k = k + 1;
73. end
74.
75. subplot(2,2,1)
76. plot(TI,X1,'b')
77. title('Estado X1: velocidad Controlada con integrador'); grid on;
78. xlabel('tiempo (s)')
79. ylabel('Velocidad (rad/s)')
80.
81. subplot(2,2,2)
82. plot(TI,X2,'b')
83. title('Estado X2: corriente Controlada con integrador'); grid on;
84. xlabel('tiempo (s)')
85. ylabel('Amperios (A)')
86.
87. subplot(2,2,3)
88. plot(TI,U,'b')
89. title('Ley de control'); grid on;
90. xlabel('tiempo (s)')
91. ylabel('Voltaje (V)')
92.
93. subplot(2,2,4)
94. plot(TI,Y,'b')
95. title('Salida: velocidad'); grid on;
96. xlabel('tiempo (s)')
97. ylabel('rad/s')
98.
99. hold on
100. plot(TI,r * ones(size(Y)), 'k--')
101. hold off
102. legend('velocidad', 'set point')

```

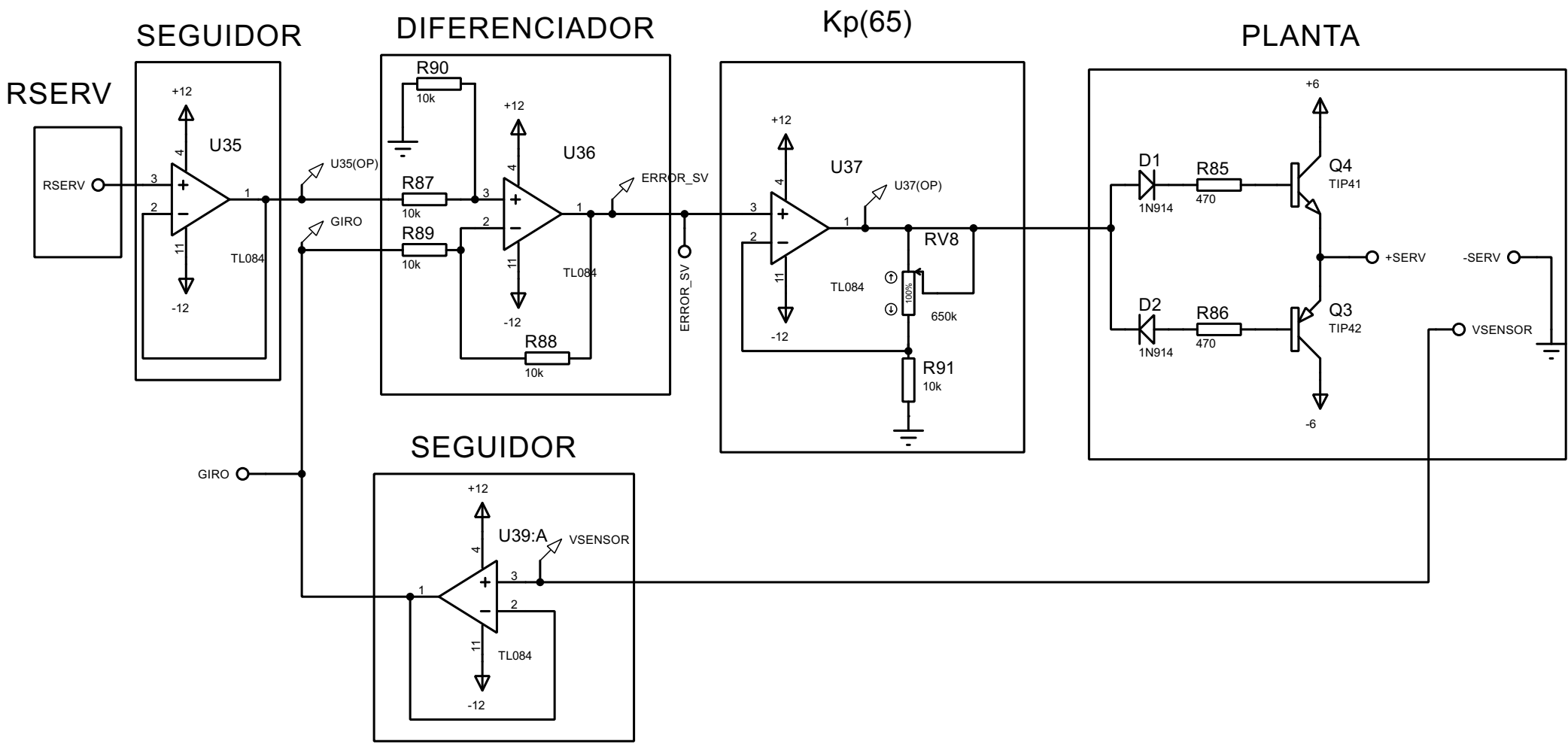
PARTS LIST			
ITEM	QTY	PART NUMBER	DESCRIPTION
1	1	Chassis_1	
2	1	Body_L	
3	1	Chassis_2_v2	
4	1	Body_R	
5	4	Basic_Wheel_01_Tire	
6	4	Basic_Wheel_01_Wheel	
7	1	Bumper	
8	1	Case_Holder	
9	1	Clamp_Case_Holder	
10	1	Front_Bars_Lower	
11	1	Front_Bars_Upper	
12	1	Front_Case_Holder	
13	1	Front_Frame	
14	1	Gear_1	
15	1	Rear_Axis	
16	1	Rear_Frame_L	
17	1	Rear_Frame_R	
18	1	Steering	
19	2	Wheel_Axis	
20	2	Wheel_Axis_Nut	
21	2	Wheel_Hub	
22	1	Gear_2	
23	2	BodyFront	
24	1	BodySpoiler	

INTEGRANTES:	DOCENTE:	FECHA:	CURSO:
GRUPO	Ing. Vargas Machuca Acevedo, D.	28/11/2021	Sistemas CAD/CAM
	CARRITO VELOCISTA		
	MATERIAL: Materiales Multiples	Vista explosionada	ESCALA: 1:1.25

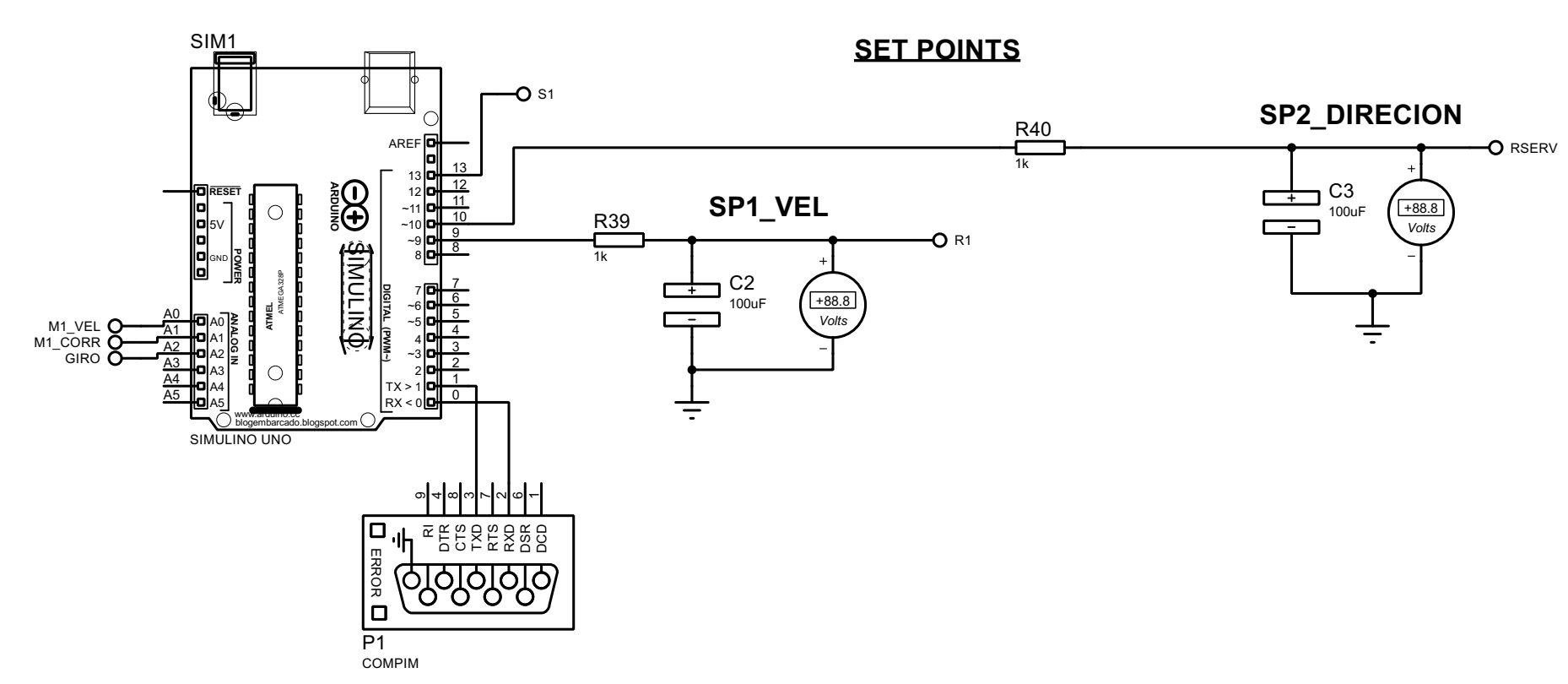
SISTEMA DE CONTROL DEL VEHICULO



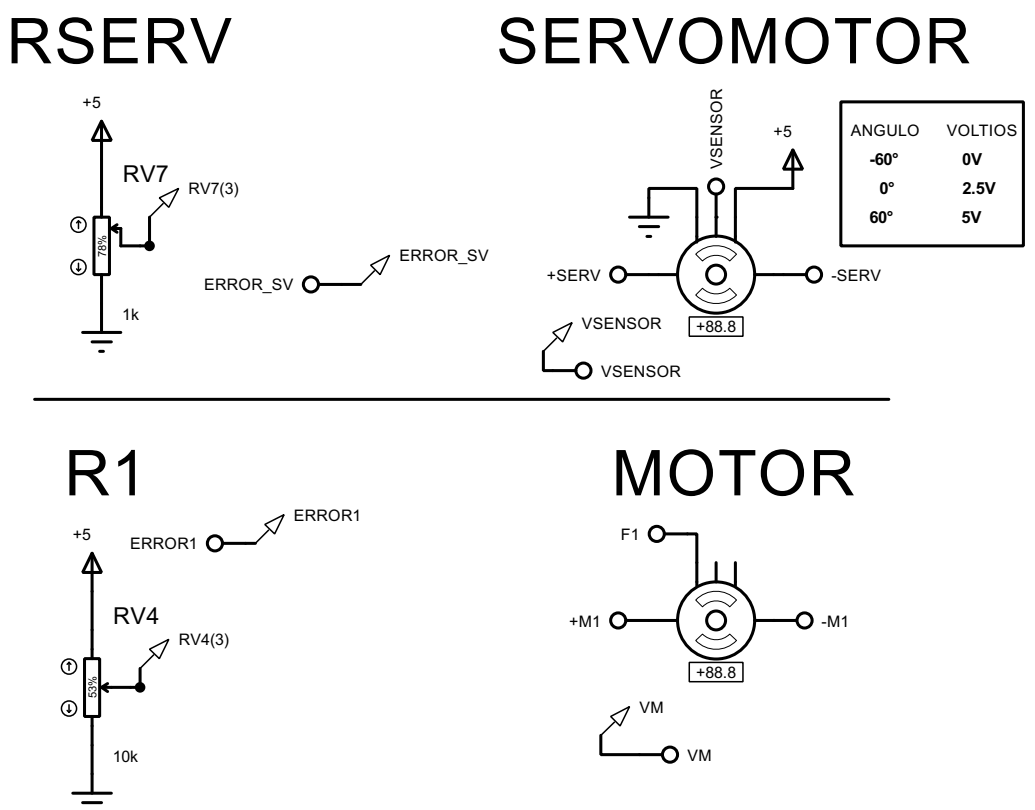
CONTROL PROPORCIONAL PARA LA POSICION



COMUNICACIÓN CON MICROCONTROLADOR



RESULTADOS



CONTROL DE VELOCIDAD POR ESPACIO DE ESTADOS

