

Informe Trabajo Práctico N° 3

Grupo 5:

Aragón, Julián - Escape, Martín - Giacosa Ballejo, Iara Nail - Rodriguez Oliva, Franco Nicolas - Salera, Denise Ana- Sommer, Elias - Utrera, Francisco

A partir de las consignas dadas en clase (realizar un diagrama de bloques de la parte física del robot, crear un diagrama de flujo del algoritmo de funcionamiento del mismo y realizar un gemelo digital) y la observación del Robot N6-Max de RobotGroup, realizamos un diagrama de bloques de la estructura del mismo.

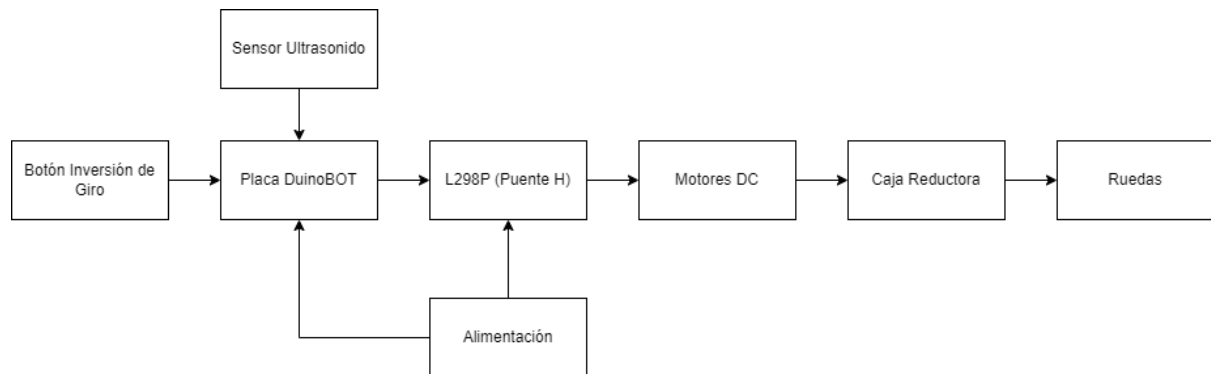


Diagrama de Bloques del N6 MAX.

Para crearlo, analizamos cada uno de los diagramas realizados por las y los integrantes del grupo para el trabajo individual, con el objetivo de acordar cuál sería el que mejor se adapte a la realidad del robot. De manera similar, con el diagrama de flujo.

Es importante aclarar que las versiones realizadas de cada diagrama en un principio no fueron las finales, sino que a medida que íbamos avanzando con el Trabajo Práctico fuimos realizando las distintas modificaciones de cada uno.

Para una mejor visualización, se puede acceder a los diagramas desde el siguiente [enlace](#), el cual direccionará a la correspondiente carpeta en el repositorio diseñado por el grupo para la cátedra.

Informe Trabajo Práctico N° 3

Grupo 5:

Aragón, Julián - Escape, Martín - Giacosa Ballejo, Iara Nail - Rodriguez Oliva, Franco Nicolas - Salera, Denise Ana- Sommer, Elias - Utrera, Francisco

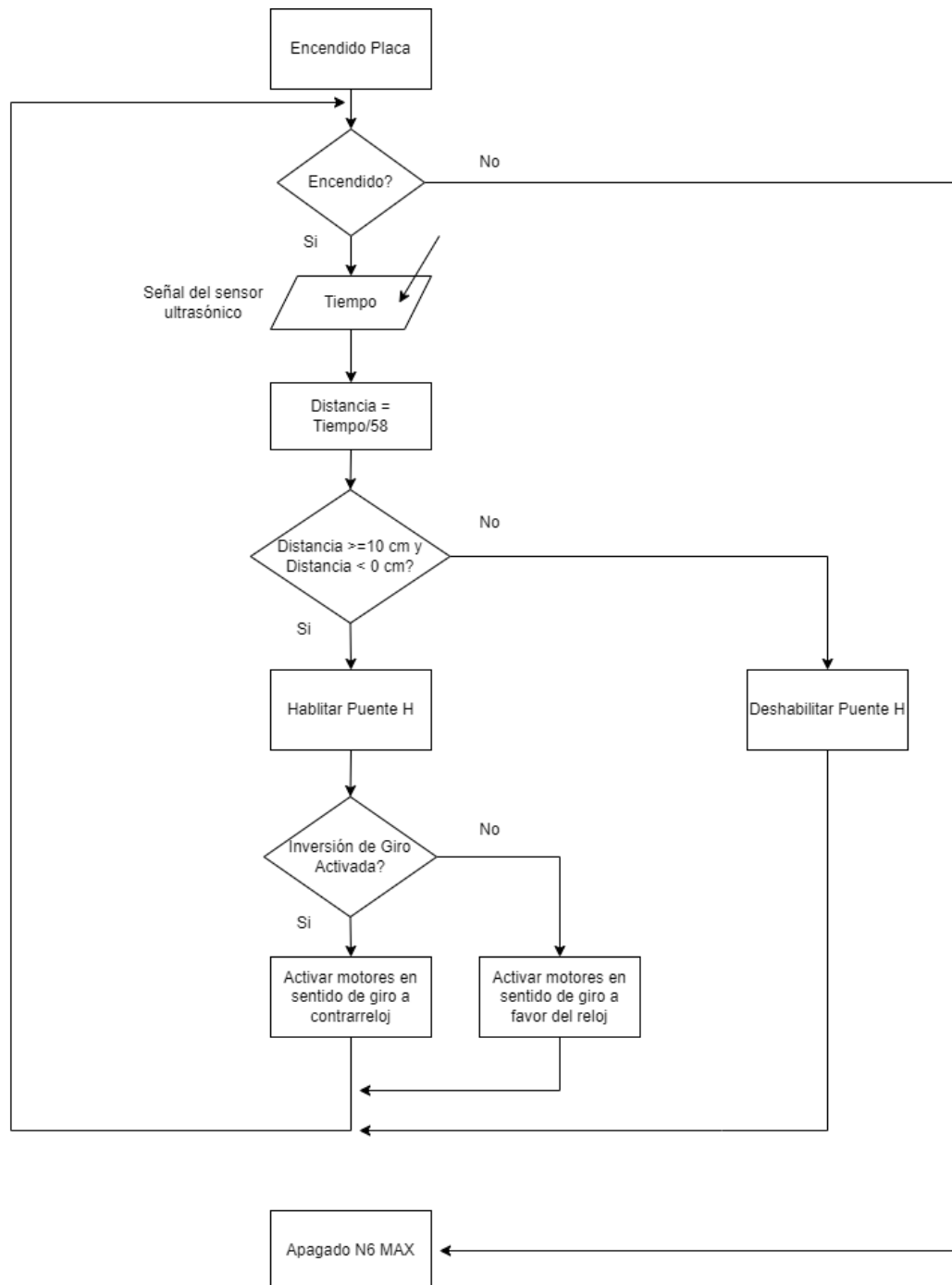


Diagrama de Flujo del N6 MAX.

Además de los diagramas antes expuestos, se agregó el diseño de un *gemelo digital*, una representación virtual del objeto en cuestión que abarca su ciclo de vida, debe actualizarse a partir de datos en tiempo real y utilizar simulaciones para facilitar la toma de decisiones.

Para esto, comenzamos a buscar información que nos pueda ayudar con la construcción en el software SimulIDE. Dado que el modelo fue discontinuado por la empresa, se tornó difícil encontrar datos relevantes, decidimos comunicarnos directamente con los desarrolladores, pero es entendible que el tiempo de respuesta termine cayendo en un estado de incertidumbre considerando que la

empresa no funciona más como tal, por lo que comenzamos a aproximar un modelo con la información que disponemos.

En la página [Archive.org](https://archive.org) se buscaron versiones de la página de robotgroup, la empresa que fabricaba al robot. De esto y la investigación previa que ya se había realizado la semana anterior cuando vimos por primera vez al robot, pudimos extraer información acerca de la constitución de este (se puede acceder a la documentación encontrada en el siguiente [link](#) En la carpeta Documentación se encuentran varios PDFs que encontramos para poder realizar el gemelo digital).

En primer lugar, en SimulIDE se realizó un trabajo de reconocimiento de componentes, para saber que podíamos usar. El modelo N6 MAX usa una placa DuinoBOT, placa que era fabricada por la empresa RobotGroup. Esta placa en particular no se encuentra en SimulIDE, pero sí otras placas Arduino, más específicamente el Arduino Uno, Arduino Mega, Arduino Nano y Arduino Duemilanove. De aquí la pregunta era qué placa usar. Como no se obtuvo una respuesta por parte de los antiguos desarrolladores de RobotGroup se procedió a investigar qué placa era parecida a la DuinoBOT. Las especificaciones de la placa DuinoBOT son las siguientes:

Microcontrolador Atmel AVR ATmega1284P hasta ~20 MIPS @ 20 MHz.

128 KBytes de memoria Flash (programa) automodificable. 16 KBytes de memoria SRAM. 4 KByte de EEPROM (datos no volátiles).

Figura 1: Especificaciones de la DuinoBOT.

Se encontró que el Arduino Uno es bastante parecido y por lo tanto se decidió utilizar éste. Para programar el Arduino Uno se decidió usar el Arduino IDE que provee Arduino en su página web (el DuinoBOT puede programarse de muchas maneras ya que se puede programar en entornos visuales como [miniBlog](#) o con lenguajes de programación).

Se realizó una primera versión del gemelo digital donde éste solamente sensa a qué distancia está el obstáculo más cercano mediante el sensor ultrasónico HC-SR04. Para simular los obstáculos se le conecta al sensor una fuente de voltaje donde a mayor voltaje, mayor distancia hay al objeto. Se conectó el sensor al Arduino Uno y, en Arduino IDE se programó el algoritmo detector de obstáculo que luego se le cargó a la placa. El programa lo que hace es sensar el tiempo de llegada de la onda de sonido (ya que el sensor envía una señal de sonido que rebota en el objeto y se toma el tiempo que tarda en volver (ver figura 2)) y a partir de eso calcular la distancia en centímetros al objeto con la fórmula proveída por el fabricante. Además, muestra en pantalla la distancia calculada y, si la distancia es menor a 10 cm (el obstáculo está cerca) informa esto con un mensaje.

Es importante aclarar que en el código, además de verificar si el obstáculo está a menos de 10 cm, verificamos también que esté a más de 0 cm. Esto lo hacemos ya que en la vida real si el obstáculo está fuera de rango del sensor, el sensor devuelve una medida negativa. Es por esto que al verificar que la distancia es positiva nos aseguramos que efectivamente el obstáculo está cerca y no lejos.

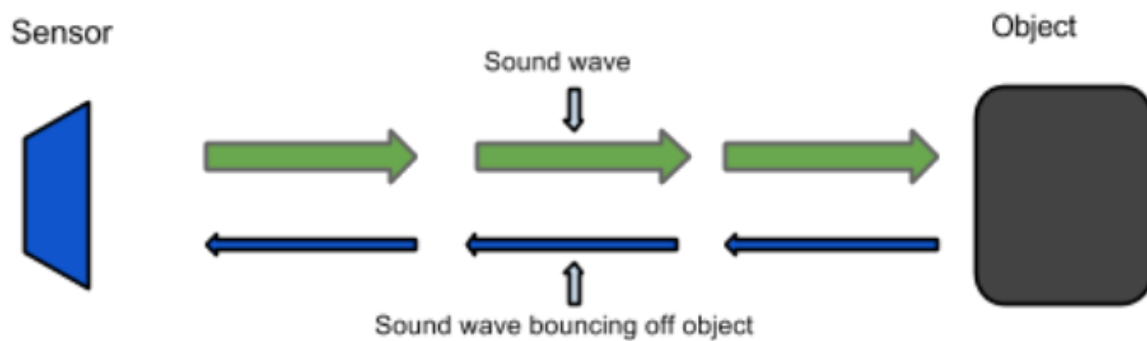


Figura 2: Funcionamiento del sensor ultrasónico.

Una vez realizado esto, se carga el programa a la placa y se puede correr la simulación. Para ver los resultados uno debe hacer click derecho en la placa y abrir el monitor Serial (ver figura 3) para poder ver las mediciones del sensor. Uno puede ir variando el voltaje que le llega al sensor para ver como va variando la distancia calculada.

Algo importante a aclarar es que en un principio vimos que no funcionaba el modelo y esto sucedía ya que uno de los pines del sensor fue conectado a uno de los pines PWM de la placa. Al modificar esto el problema fue solucionado.

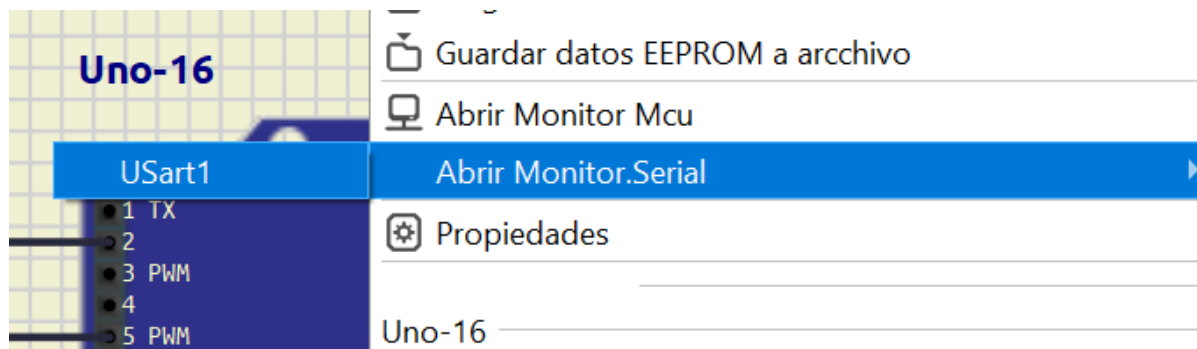


Figura 3: Manera de abrir el monitor Serial.

Una vez que se comprobó que este primer modelo que solo sensa obstáculos funciona, se procedió a agregar al modelo los motores. Para esto nos fijamos en las especificaciones del N6 MAX. Éste usa un puente H L298P para controlar los motores. Además se usan dos motores DC de 12 V. El N6 MAX utiliza una caja reductora para poder obtener 256 rpm, nosotros para simular esto fijamos la velocidad nominal de ambos motores a 256 rpm.

Para esta nueva versión del modelo se conectaron los pines 12 y 13 de la placa a los pines Input 1 e Input 2 (solo usaremos uno de los puentes ya que los motores funcionaran de manera conjunta, todo lo que hace un motor lo hace el otro) del L298P y se conectó el pin Out 1 al borne positivo de los motores y el pin Out 2 al borne negativo de los motores.

En el Arduino IDE se desarrolló la lógica de encendido y apagado de los motores. Se creó variable booleana que nos dice si hay o no un obstáculo cerca. En primer lugar se codeo la lógica para que ambos motores anden en una sola dirección. Cuando se hizo esto y se cargó el código a la placa, se vió que los motores no prendían. Tras volver a revisar la documentación del L298P se encontró que

Informe Trabajo Práctico N° 3

Grupo 5:

Aragón, Julián - Escape, Martín - Giacosa Ballejo, Iara Nail - Rodriguez Oliva, Franco Nicolas - Salera, Denise Ana- Sommer, Elias - Utrera, Francisco

esto pasaba ya que el pin EnableA es el que activa el primer puente H. Es así que se procedió a conectar el pin 5 a este pin y se modificó el código. Ahora en caso de que no haya ningún obstáculo, se habilita el primer puente mediante el pin EnableA y en caso de que sí haya un obstáculo se lo deshabilita al puente.

Cuando se le cargó este código a la placa, los motores se encendían si no había ningún obstáculo y se apagaban en el caso de que hubiera uno (recordar que para simular un obstáculo se usa la fuente de voltaje conectada al sensor ultrasónico). Para terminar este nuevo modelo se procedió a hacer la inversión de marcha. Se usó el pin de 5 V de la placa y un interruptor que indica cuando hay que invertir la marcha. En el código se implementó la lectura de este interruptor, donde si está cerrado se activan los motores en un sentido y si está abierto se activan en el otro sentido.

Al cargar este código a la placa se comprobó que efectivamente todo funcionaba bien. Si se detecta un obstáculo los motores paran y si se cierra el interruptor cambia el sentido de giro de los motores.

Otro de los desafíos fue el de realizar un repositorio donde poder organizar y disponer de manera más accesible todo el material generado por el grupo, para eso, decidimos utilizar GitHub, dado que se trata de una plataforma de código abierto muy popular.

Como una conclusión para nuestro trabajo, responderemos las preguntas dadas en clase:

¿Qué fue lo que más nos gustó?

Lo que más nos gustó fue el realizar un trabajo con distintos programas que ninguno del grupo había usado antes e incluso la mayoría no conocía. Esto nos permitió poder adentrarnos un poco en la rama de la electrónica a los que no la conocíamos y profundizar más en ella a los que ya teníamos varios conceptos de esta.

Hizo que varios integrantes del grupo conocieran la herramienta GitHub, que consideramos que para nuestra carrera es importante conocerla.

Además nos dio mucha satisfacción ver al gemelo digital en su versión final totalmente funcional, ya que a lo largo del proceso de la realización de este nos fuimos encontrando con distintos problemas que nos fueron frenando pero que pudimos superarlos.

¿Qué tenemos que seguir haciendo?

Cómo grupo humano, trabajamos de un modo fluido, y eso es valorable. Pero muchas veces terminamos sobrecargando más a algunos. Debido a que no contamos con conocimientos sobre los temas a trabajar.

La mayoría de integrantes no tenía conocimientos en electrónica, tuvimos que depender en gran parte del aporte de los compañeros que contaban con estos conocimientos previos. Por un lado, siempre es una buena experiencia aprender de otros, sucedió también con el aprendizaje de la utilización de GitHub por ejemplo, pero por otro, consideramos que necesitaríamos mayor orientación por parte de la cátedra, no por desconfiar de los saberes de los compañeros, sino, por encontrarnos en esta etapa tan temprana, donde la validación experta se torna tan importante para orientar nuestro avance.

¿Qué podemos hacer aún mejor?

Informe Trabajo Práctico N° 3

Grupo 5:

Aragón, Julián - Escape, Martín - Giacosa Ballejo, Iara Nail - Rodriguez Oliva, Franco Nicolas - Salera,
Denise Ana- Sommer, Elias - Utrera, Francisco

Tendríamos que seguir avanzando en la división de tareas, pero no tanto desde los conocimientos propios, sino, en una división productiva en subgrupos quizás, donde alguien que tenga mayor conocimiento, pueda trabajar con alguien que no, en un área desconocida para ambos, entendiendo que el principal obstáculo siempre es el tiempo.

Finalmente dejamos links de la página Archive.org que nos sirvieron a la hora de hacer el informe:

<https://web.archive.org/web/20190903131050/https://robotgroup.com.ar/>

<https://web.archive.org/web/20180222165700/http://robotgroup.com.ar/es/17-service/46-n6-max>