

GRADO EN INGENIERÍA EN TECNOLOGÍA INDUSTRIAL
TRABAJO FIN DE GRADO

***CONSTRUCCIÓN Y CONTROL DE UNA
MAQUETA FUNICULAR***

Alumna: De la Herrán Prada, Aida

Director: Gómez Garay, Vicente

Curso: 2017-2018

Fecha: 16/07/2018

Resumen Trilingüe

En este documento se expone la metodología seguida para la realización de la construcción y el control de una maqueta del funicular de Artxanda, utilizando para ello el kit Lego Mindstorms NXT. Es un proyecto realizado como Trabajo de fin de Grado, en colaboración con el Departamento de Automática y Control. A lo largo del mismo se exponen las diferentes etapas, como el estudio previo de viabilidad, el diseño de la maqueta, la construcción de esta o la programación del sistema de control. Se ha incluido también una lista detallada de las tareas realizadas y un presupuesto del proyecto.

Artxandako funikularraren eredua eraikitzeke eta kontrolatzeko metodologia aurkezten du dokumentu honek, LEGO Mindstorms NXT kit erabiliz. Lanbide amaierako lana da, Automatizazio eta Kontroleko Departamentuarekin lankidetzan. Aldi berean, fase desberdinak azaltzen dira, bideragarritasunaren aurreko azterketa, ereduaren diseinua, kontrol-sistemaren eraikuntza edo programazioa. Atazak eta proiektu aurrekontua ere barne hartu dituzte.

This document presents the methodology followed for the construction and control of a model of the Artxanda funicular, using the Lego Mindstorms NXT kit. It is a project carried out as End of Degree Work, in collaboration with the Department of Automation and Control. Throughout the same the different stages are exposed, like the previous study of viability, the design of the model, the construction of this or the programming of the control system. A detailed list of the tasks performed and a project budget have also been included.

Palabras clave: Lego Mindstorms NXT, programación, simulación, Funicular de Artxanda, maqueta.

Índice

Índice de Ilustraciones	4
Índice de Tablas	6
1. MEMORIA.....	7
1.1 Introducción.....	7
1.2 Objetivo y alcance de trabajo.....	7
1.3 Contexto.....	9
1.4 Beneficios que aporta el trabajo.....	13
1.5 Estado del Arte.....	14
1.6 Análisis de Alternativas.....	15
1.7 Descripción de la solución propuesta. Diseño básico.....	22
2. METODOLOGÍA SEGUIDA EN EL DESARROLLO DEL TRABAJO	25
2.1 Descripción de tareas y fases.....	25
2.1.1 Estudio de viabilidad.....	26
2.1.2 Diseño de la estructura	26
2.1.3 Construcción.....	31
2.1.4 Programación.....	43
2.2 Planificación del proyecto. Diagrama de Gantt	47
2.2.1 Listado de tareas.....	47
3. ASPECTOS ECONÓMICOS. PRESUPUESTO EJECUTADO	52
3.1 Gastos relacionados con el montaje.....	52
3.2 Gastos Hardware Lego Mindstorms.....	53
3.3 Mano de obra	53
3.4 Software utilizado.....	53
3.5 Presupuesto ejecutado.....	54
4. CONCLUSIONES	55
5. BIBLIOGRAFÍA	56
ANEXO I: Código en lenguaje NXC.....	57
ANEXO II: Listado de elementos	61

Índice de Ilustraciones

Ilustración 1- Antigua ilustración del trayecto del funicular.....	8
Ilustración 2-Funicular de Artxanda.....	9
Ilustración 3- Kit Lego Mindstorms.....	10
Ilustración 4- Ladrillo LEGO Mindstorms NXT.....	11
Ilustración 5- Servomotor LEGO Mindstorms NXT.....	12
Ilustración 6- Sensores LEGO Mindstorms NXT.....	12
Ilustración 7- Maqueta de los antiguos vagones del funicular de Artxanda.....	14
Ilustración 8- Maqueta del funicular de Zagreb con Lego[4].....	15
Ilustración 9- Poleas del funicular en la estación superior.....	16
Ilustración 10- Centro de control.....	17
Ilustración 11- Detalle del cable de acero trenzado.....	17
Ilustración 12- Detalle del entorno de programación BrixCC.....	25
Ilustración 13- Vistas de la base inclinada.....	26
Ilustración 14- Posición de las vías y sus dimensiones.....	27
Ilustración 15- Vistas de los vagones.....	28
Ilustración 16- Detalle de la bifurcación.....	28
Ilustración 17- Vistas de la estación superior.....	29
Ilustración 18- Vistas de la estación inferior.....	29
Ilustración 19- Localización de las estaciones y los elementos de control.....	30
Ilustración 20- Corte de la base con sierra de calar.....	31
Ilustración 21- Unión de las dos piezas de la base con tornillos y palomillas.....	32
Ilustración 22- Preparación de las piezas de unión de las dos partes de la base.....	32
Ilustración 23- Encolado de la bifurcación.....	33
Ilustración 24- Encolado de las vías.....	33
Ilustración 25- Lijado de la unión de vías para evitar desniveles.....	34
Ilustración 26- Unión de la vía de ambas piezas de la base.....	34
Ilustración 27- Detalle de las patas.....	35
Ilustración 28- Vista de la base inclinada con las vías.....	36
Ilustración 29- Taladrado y atornillado de las patas a la estructura.....	36
Ilustración 30- Bifurcación con las agujas ya colocadas.....	37
Ilustración 31- Taladrado del orificio donde se alojan las agujas.....	37
Ilustración 32- Preparación de las agujas.....	37
Ilustración 33- Vista de los vagones en la bifurcación.....	38
Ilustración 34- Vistas de la estación superior.....	38
Ilustración 35- Vistas de la estación inferior.....	39
Ilustración 36- Estación inferior colocada en la base.....	39
Ilustración 37- Elementos de sujeción de la estación superior en la base.....	40
Ilustración 38- Encolado de los elementos de sujeción del display sobre la base.....	40
Ilustración 39- Colocación del servomotor principal.....	41
Ilustración 40- Vista de los elementos de control sobre la base.....	41
Ilustración 41- Sensor de contacto en la estación superior.....	41

Ilustración 42- Unión del cable a los vagones	42
Ilustración 43- Vista del vagón con los cables	42
Ilustración 44- Detalle de la base con la tela verde y las vías pintadas.....	43
Ilustración 45- Diagrama de flujo del movimiento principal	44
Ilustración 46- Diagrama de flujo del control de la bifurcación.....	45
Ilustración 47- Relación temporal entre entradas y salidas	46
Ilustración 48- Diagrama de Gantt.....	51

Índice de Tablas

Tabla 1- Elección de la alternativa de construcción	21
Tabla 2- Elección de la alternativa del Lenguaje de Programación	22
Tabla 3- Presupuesto materiales construcción	52
Tabla 4- Presupuesto Gastos Hardware.	53
Tabla 5- Presupuesto Mano de obra.....	53
Tabla 6- Presupuesto software.....	53
Tabla 7- Presupuesto final.....	54

1. MEMORIA

1.1 Introducción

Este documento contiene la descripción de todas las etapas necesarias para la Construcción y Control de una maqueta del funicular que une la villa de Bilbao con la cima del monte Artxanda.

Se ha estructurado en tres partes. La primera parte se corresponde con la memoria, que recoge el estudio previo. En ella se han incluido la recopilación de información sobre el funicular y sobre el kit Lego que se ha utilizado, el estudio previo de viabilidad, la elección de las distintas alternativas posibles en base a distintos criterios y, por último, el diseño previo de la maqueta.

La segunda parte incluye una descripción de las tareas realizadas, el diseño detallado del sistema, las distintas fases de la construcción y la programación del sistema de control.

La última parte recoge un presupuesto detallado del proyecto.

También se han incorporado dos anexos, uno con el código de programación del sistema de control y el otro con una lista completa de los elementos que componen el sistema.

1.2 Objetivo y alcance de trabajo

El objetivo del trabajo de fin de grado, en adelante TFG, es replicar el diseño y funcionamiento del funicular de Artxanda, utilizando el kit Lego Mindstorms NXT y, si fuera necesario, otros materiales adicionales. Se trata de construir una maqueta lo más realista posible del funicular y crear un programa que el ladrillo inteligente Lego ejecute para simular el funcionamiento del funicular real.

Dada la imposibilidad de replicar con exactitud el funcionamiento real del funicular, la maqueta va a simular los siguientes aspectos:

- Movimiento de subida y bajada, siendo los vagones contrapeso el uno del otro, con un motor en la estación superior.

- Parada breve de los vagones al llegar a la estación, con la ayuda de un sensor de contacto, y reanudación del movimiento invirtiendo el sentido de giro del motor.
- Bifurcación de las vías para evitar la colisión cuando los vagones se encuentran en el mismo punto del recorrido.

Se van a hacer las simplificaciones siguientes:

- La pendiente es constante a lo largo del recorrido, emplearemos la pendiente correspondiente a la longitud total y el desnivel total.
- La trayectoria se considera rectilínea, por simplicidad se suprime la curva que tiene el recorrido en la parte inferior del mismo.



Ilustración 1- Antigua ilustración del trayecto del funicular

- El sistema se pone en funcionamiento estando cada vagón en una estación, es decir, estando el sensor de contacto pulsado. Esto evitará que los vagones se desplacen por el lado equivocado de la bifurcación. El vagón que sube va por la vía derecha y el que baja va por la izquierda.

1.3 Contexto

El TFG se lleva a cabo dentro del Departamento de Ingeniería de Sistemas y Automática de la Escuela Técnica Superior de Ingeniería de Bilbao. Este trabajo se realiza como conclusión del grado en Ingeniería en Tecnología Industrial.

Para presentar el contexto en el que se desarrolla el trabajo vamos a presentar brevemente, por una parte, el funicular de Artxanda y, por otra, el kit Lego que se va a emplear.

El funicular de Artxanda es un medio de transporte que comunica la villa de Bilbao con la cima del monte Artxanda, en Vizcaya (España), permitiendo una rápida ascensión del mismo.



Ilustración 2-Funicular de Artxanda

Fue construido por la empresa suiza Von Ron a iniciativa del donostiarra Evaristo San Martín.

Su primer viaje lo hizo el 7 de octubre de 1915, operado por una empresa privada. Al quebrar ésta, en 1939, el municipio se hizo cargo de él. Fue totalmente reformado en 1983 y actualmente está gestionado por el ayuntamiento de Bilbao.

Se construyó para poder acceder al casino que estaba situado en el monte Artxanda. En la guerra civil, durante el asedio de Bilbao, las vías y la estación superior fueron bombardeadas y se interrumpió el servicio hasta 1938. [1]

Por otro lado, Lego Mindstorms es una línea de juguetes de robótica para niños fabricado por la empresa LEGO, que posee elementos básicos de las teorías robóticas, como la unión de piezas y la programación de acciones en forma interactiva. Este producto de Lego fue comercializado por primera vez en septiembre de 1998. [2]



Ilustración 3- Kit Lego Mindstorms

Comercialmente se publicita como Robotic Invention System, en español Sistema de Invención Robotizada (RIS). También se vende como herramienta educativa, lo que originalmente se pensó en una colaboración entre LEGO y el MIT. La versión educativa se llama Lego Mindstorms for Schools y viene con un software de programación basado en la GUI de Robolab.

Lego Mindstorms puede ser usado para construir un modelo de sistema integrado con partes electromecánicas controladas por computador. Prácticamente todo puede ser representado con las piezas tal como en la vida real, como un elevador o robots industriales.

Hasta 2015 ha habido tres generaciones de Lego Mindstorms: el bloque RCX, el bloque NXT y el EV3.

Para este TFG vamos a usar kits Lego Mindstorms NXT, compuestos por displays (ladrillo inteligente), sensores y motores. Pueden conectarse varios ladrillos entre ellos y a ordenadores mediante conexión bluetooth y cable ethernet. [3]

El hardware de Lego Mindstorms se compone de tres tipos de piezas:

- **Ladrillo.** En el ladrillo reside todo el control del robot, es su “cerebro”. Soporta hasta cuatro sensores, conectados a los puertos de entrada del sistema, y controlar hasta tres servomotores, conectados a los puertos de salida. Las conexiones con los puertos se realizan mediante cable o bluetooth y permiten la ejecución del programa creado por el usuario. Para programar este dispositivo, existen diferentes lenguajes de programación de diferentes niveles.

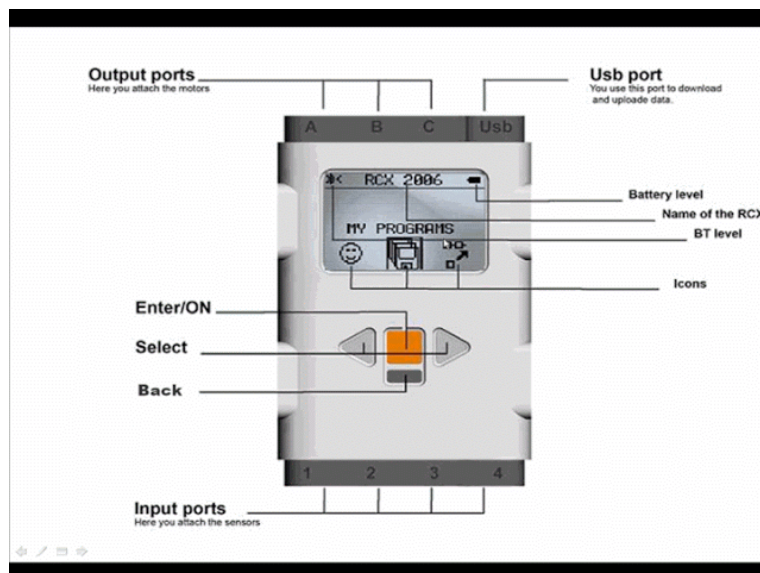


Ilustración 4- Ladrillo LEGO Mindstorms NXT

- **Servomotores.** Proporcionan al robot la capacidad de moverse. Disponen de un sensor de rotación integrado, lo que permite que los movimientos sean muy precisos.

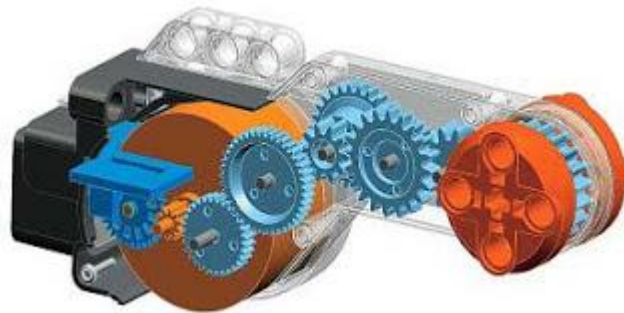


Ilustración 5- Servomotor Lego Mindstorms NXT

- **Sensores.** Perciben cambios en el entorno y envían señales eléctricas al microprocesador. Existen cuatro tipos de sensores: sensor de táctil (es un interruptor que puede presionarse y liberarse, permite saber si el robot ha colisionado con algún objeto), sensor acústico (detecta sonidos, es capaz de discriminar tonos), sensor fotosensible (permite al robot distinguir entre luminosidad y oscuridad, mide las variaciones del nivel de luz y de color) y, por último, el sensor ultrasónico (emite un sonido y mide el tiempo que tarda en ir volver, lo que le da una idea de la distancia a la que se encuentran los objetos).



Ilustración 6- Sensores Lego Mindstorms NXT

1.4 Beneficios que aporta el trabajo

Los beneficios que aporta el trabajo han sido separados en tres categorías distintas. La primera se corresponde con los beneficios técnicos, la segunda con los beneficios personales y la tercera con los beneficios sociales.

Técnicos: Han sido diversos los conocimientos técnicos adquiridos realizando el proyecto. En primer lugar, se ha utilizado un lenguaje de programación con el que no habíamos trabajado antes para crear un programa estructurado que controle la maqueta para, posteriormente ver el resultado obtenido. Por otra parte, ha sido necesario realizar un gran trabajo de construcción para realizar la maqueta, lo que permite adquirir otro tipo de conocimientos que no se han trabajado en el grado pero que, como se ha podido ver, se han combinado con el trabajo de programación para llegar al resultado final. También ha supuesto realizar un gran trabajo de investigación, con la ayuda de Internet, contrastando la información de diversas fuentes.

Personales: La realización del trabajo ha supuesto un reto a la hora de crear un proyecto desde el principio hasta el final, con todos los pasos intermedios como el estudio de viabilidad, el diseño y la construcción, cometiendo errores y aprendiendo de ellos. Supone un reto a la hora de combinar los conocimientos adquiridos de las diferentes asignaturas del grado, como Informática, Gráficos de Ingeniería, Mecánica o Fundamentos de Ciencia de Materiales, para crear un prototipo cuyo funcionamiento replique el del funicular real y además sea estéticamente correcto.

Sociales: El trabajo final podrá ser expuesto en diferentes eventos realizados por la Escuela, como las jornadas de puertas abiertas dirigidas al futuro alumnado, despertando el interés de este por la ingeniería y, más concretamente, por la rama de Automatización y Control. Además, también podrá ser de utilidad para Trabajos de Fin de grado de otros alumnos que quieran realizar proyectos similares o incluso mejor la maqueta del funicular y/o su programa de control.

1.5 Estado del Arte

Después de una búsqueda sobre la realización de maquetas del funicular de Artxanda se puede decir que:

- Solo se ha documentado la construcción de una maqueta del funicular de Artxanda, que se encuentra actualmente en la estación superior, no está construida con piezas de Lego si no con otros materiales y no se le ha dotado de movimiento, es estática. Representa los vagones antiguos del funicular, no su aspecto actual. Si se han hecho más maquetas a lo largo de la historia no se ha documentado su construcción.



Ilustración 5- Maqueta de los antiguos vagones del funicular de Artxanda

- En cambio, sí se han fabricado anteriormente maquetas de otros funiculares con piezas de Lego, algunas animadas con la ayuda de los kits Lego Mindstorms, como es el caso del funicular de Zagreb.



Ilustración 6- Maqueta del funicular de Zagreb con Lego[4]

Por tanto, se puede decir que es la primera vez que se crea una maqueta animada que represente el funicular de Artxanda y su movimiento

1.6 Análisis de Alternativas

El primer paso es realizar un análisis detallado del funicular real y sus características técnicas. Después hay que analizar los materiales disponibles para, finalmente, ver cuáles son las distintas alternativas de construcción de la maqueta y elegir la más adecuada, según distintos criterios de valoración, para la realización del trabajo.

Análisis técnico del Funicular de Artxanda

El objetivo de este análisis es conocer los datos geométricos (dimensiones y forma de las vías y los vagones), los datos mecánicos (materiales, tipo de motores, velocidades, etc.) para poder recrear el funcionamiento de la manera más real posible.

Según los datos proporcionados por el ayuntamiento de Bilbao [5], los datos técnicos más relevantes se describen a continuación:

- El recorrido mide 770.34m, con una curva intermedia que hace diferente este funicular frente a otros, se gana un desnivel de 226.49m con pendientes máximas del 44.98%. La longitud del tramo de cruce es de 105m.
- Se alcanza una velocidad de 5m/s (18km/h).
- La capacidad de los vagones es de 70 personas. Las dimensiones son de 10.5m de largo y 2.65m de ancho. El peso aproximado de cada vehículo en servicio es de 15850kg.
- El funcionamiento es en base a un cable de 4cm de diámetro formado por hilos de acero que, a través de unas poleas forradas con goma para evitar el deslizamiento del cable y su desgaste, están repartidas a lo largo de la vía. Ambos vagones están unidos a través de este cable y se utilizan como contrapeso mutuo, lo cual no impide el uso de un motor de 300kw de potencia que, a 1500rev/min, ataca a un reductor que las baja a 30rev/min y este a su vez lo hace sobre la gran polea motriz de 5m de diámetro que arrastra el cable.



Ilustración 7- Poleas del funicular en la estación superior

- Existen unos frenos: principal y operacional, junto con los de los vagones, y un automatismo PLC que gestiona todas las funciones del funicular y hace prácticamente imposible que se produzca un accidente.



Ilustración 8- Centro de control

- El cable de acero es chequeado anualmente por personal externo especializado para controlar su estado y recomendar su sustitución si fuera preciso, siendo la vida útil unos 15-25 años.



Ilustración 9- Detalle del cable de acero trenzado

- Todos los días se realiza un viaje de pruebas, sin viajeros, previo a su apertura.

Análisis del material disponible

En esta fase, se ejecutó un estudio sobre los recursos útiles para la construcción de la maqueta.

1. El kit de Lego Mindstorms NXT disponible en el Departamento incluye ladrillos NXT, servomotores NXT y sensores de contacto, sonido, luz y ultrasonidos, todos ellos descritos anteriormente.

El kit incluye también piezas estructurales de los llamados Lego Technic, compuestos por vigas, ejes, engranajes, ruedas, etc.

También hay disponibles piezas de Lego Classic, que pueden resultar útiles para la construcción de los vagones y las estaciones.

2. En cuanto a las herramientas disponibles, contamos con sierras de calar, taladros y brocas, destornilladores, cola blanca, limas y algún cúter.

Alternativas posibles

Existen diferentes alternativas de construcción de la maqueta y lenguaje de programación. Se exponen esas distintas alternativas, así como las ventajas y desventajas principales que supone su utilización.

- **Construcción de la maqueta funicular.**

1. **Utilizar solamente piezas Lego.** Esta opción se basa en utilizar solamente elementos incorporados en el kit Lego como ruedas, vigas, ladrillos y elementos de unión.

Ventajas: Piezas Lego disponibles en el Departamento de la Escuela (reducción de costes), manejo intuitivo, fácil unión y piezas ligeras.

Desventajas: Las piezas imponen las dimensiones y formas de la maqueta, lo que puede suponer un problema para obtener la escala deseada. Dificultad de crear una base inclinada de gran tamaño con piezas Lego.

2. **Maqueta entera de madera.** Crear la base, las vías, las estaciones y los vehículos solo con madera.

Ventajas: Mayor rigidez, mayor flexibilidad para adaptar la maqueta a las dimensiones y formas deseadas, mejor apariencia estética que con las piezas Lego.

Desventajas: Necesidad de adquirir más material adicional y herramientas para su tratamiento, la madera cambia sus propiedades con el tiempo y la humedad. Dificultad para incorporar ciertos elementos como pueden ser las ruedas de los vagones. Dificultad para adaptar los motores y sensores a la estructura de madera. Mayor peso.

3. **Combinar las piezas Lego con elementos de madera.** Utilizar la madera para crear la base inclinada y las vías y las piezas de Lego para crear los vagones y las estaciones

Ventajas: Posibilidad de crear la base de una sola pieza y adaptar el tamaño de las vías al tamaño de los vehículos.

Desventajas: Mayor peso. Necesidad de adquirir madera y herramientas para su tratamiento. Dificultad para adaptar los motores y sensores a la estructura de madera.

- **Elección de la escala**

1. **Usar la misma escala para toda la maqueta.** Elegir una escala y crear todos los elementos (base, vías, vagones y estaciones) con esa escala

Ventajas: Más realismo.

Desventajas: Si elegimos la escala pensando en el tamaño de la base (por ejemplo 1:500) los vagones y las estaciones tienen un tamaño muy pequeño, que no podemos construir con las piezas de Lego, además, visualmente no queda bien. Si adaptamos la escala al tamaño de vagones

deseado (por ejemplo 1:100) tendríamos que construir una base muy grande, lo cual no es viable.

2. ***Dos escalas distintas.*** Crear la base y las vías con una escala 1:500 y las estaciones y los vehículos con una escala 1:100.

Ventajas: Base de un tamaño suficientemente grande y que se puede transportar con relativa facilidad. Vagones y estaciones con unas dimensiones realizables con piezas de Lego. Mejora la estética de la maqueta.

Desventajas: Menor realismo.

- **Lenguaje de programación.**

1. ***ROBOT-C.*** Lenguaje textual basado en AnsiC. [6]

Ventajas: Los lenguajes textuales ocupan mucha menos memoria que los gráficos, muy importante debido a las limitaciones de memoria del ladrillo NXT. Programas son livianos y de rápida ejecución.

Desventajas: Conocimientos de programación necesarios.

2. ***NTX-G.*** Lenguaje de programación en modo gráfico basado en iconos, se trata de un software oficial de LEGO basado en Labview. Orientado a público sin conocimientos previos de programación. [7]

Ventajas: No es necesario tener conocimientos de programación. Fácil e intuitivo para crear programas sencillos. Incluido en el kit.

Desventajas: Para códigos más complejos son necesarios muchos iconos, ocupa mucha memoria y se ejecuta muy despacio.

3. ***NXC (Not Exactly C).*** Lenguaje de programación textual creado por la comunidad de usuarios de LEGO, similar al lenguaje C. [8]

Ventajas: Se puede encontrar como software libre. Fácil e intuitivo, no son necesarios conocimientos de programación. Programas muy ligeros y de rápida ejecución.

Desventajas: Los programas desarrollados en el lenguaje Robot-C y NXC no son compatibles dado que cada uno usa sus propias librerías.

Elección entre las alternativas posibles

- **Construcción de la maqueta funicular.**

A continuación, se incluye una tabla en la que se puntúa, del 1 al 10, cada alternativa según cinco criterios de selección distintos: la flexibilidad a la hora del montaje, la posibilidad de elegir las dimensiones, los costes, la estética y el deterioro de los componentes. A cada criterio se le ha asignado una ponderación. El resultado final indica el grado de cumplimiento de los criterios:

CRITERIOS	FLEXIBILIDAD	DIMENSIONES	COSTES	ESTÉTICA	DETERIORO	TOTAL
PONDERACIÓN	0,3	0,3	0,2	0,1	0,1	1
ALTERNATIVA 1	4	2	9	3	9	4,8
ALTERNATIVA 2	5	9	2	8	2	5,6
ALTERNATIVA 3	6	7	6	7	5	6,3

Tabla 1- Elección de la alternativa de construcción

En vista de los resultados obtenidos, se elige la alternativa 3: el uso combinado de las piezas de Lego con elementos de madera.

- **Elección de la escala**

La primera alternativa no resulta viable, utilizaremos por lo tanto la alternativa 2: emplear dos escalas diferentes.

- **Lenguaje de programación.**

Al igual que para la elección de los materiales de construcción, se ha realizado una tabla en la que se puntúan cuatro criterios: la facilidad de uso, es decir si son necesarios o no conocimientos previos, los costes, la memoria disponible necesaria y la rapidez de ejecución de los programas. También en este caso se obtienen los resultados en base a una ponderación de los criterios:

	FACILIDAD DE USO	COSTES	MEMORIA	RAPIDEZ	TOTAL
PONDERACIÓN	0,1	0,3	0,3	0,3	1
ALTERNATIVA 1	2	6	8	9	7,1
ALTERNATIVA 2	9	7	3	3	4,8
ALTERNATIVA 3	9	8	8	9	8,4

Tabla 2- Elección de la alternativa del Lenguaje de Programación.

En base a estos resultados, se ha elegido la opción 3: crear el código de control en el lenguaje NXC.

1.7 Descripción de la solución propuesta. Diseño básico

- **Maqueta funicular.**

Se compone de los siguientes elementos:

- **Base inclinada.** Construida en madera, con una patas que le dan la inclinación adecuada. Se divide en dos (parte superior y parte inferior) para facilitar su transporte. Sobre ella se encolan las vías, también de madera, y los elementos necesarios para fijar las estaciones y los elementos de control.

- **Vagones.** Construidos con piezas de Lego. Unidos entre sí mediante un hilo de acero. Gracias a unas ruedas circulan por el interior de las vías.
- **Mecanismo de desvío.** Para evitar que los vagones colisionen uno con otro al llegar al mismo punto, el funicular cuenta con una bifurcación o desvío. Con la ayuda de dos servomotores, los raíles móviles, llamados agujas, cambian de posición para dirigir cada vagón a un lado o al otro.
- **Estaciones.** Se crean con piezas de Lego. En la estación superior se coloca el eje que transmite, a través del cable de acero, el movimiento a los vagones. Para mayor realismo, la estación superior es de intemperie y la inferior subterránea.
- **Sistema de control.** Formado por componentes físicos (hardware) y lógicos (software de programación). Se utiliza un ladrillo inteligente Lego NXT, al que se conectan por cable los motores y sensores necesarios. El ladrillo se coloca en la zona media de la parte superior, para que los cables lleguen del mismo hasta el motor principal (controla el eje en la estación superior) y hasta los motores secundarios (controlan las agujas de la bifurcación).

Se emplean dos tipos de sensores: un sensor táctil y uno fotosensible. El sensor táctil, situado en la estación superior, detecta la llegada de un vagón, esto implica que el otro vagón se encuentra en la estación inferior. El sensor fotosensible, colocado en la bifurcación, detecta cuando los vagones están pasando por la misma y provoca el movimiento de las agujas. Utilizamos el software BricxCC para crear el algoritmo de control en el ordenador y lo cargamos en el ladrillo mediante un cable USB (también puede hacerse por bluetooth).

- **Lenguaje de programación.**

Utilizamos el lenguaje de programación estructurado NXC. Es un lenguaje de programación de alto nivel basado en C. Es un software de programación textual y de libre distribución (se puede descargar de <http://bricxcc.sourceforge.net/>). NXC significa Not eXactly C, aunque en realidad

las diferencias con C son mínimas dado que es una especie de lenguaje C adaptado a la programación de robots LEGO. Está construido sobre el compilador NBC (anterior a NXC, de tipo ensamblador), y es la evolución natural del anterior lenguaje de programación NQC (Not Quite C), usado en las primeras versiones de LEGO Mindstorms, los RCX. Su creador fue John C. Hansen, ingeniero de software y ampliamente conocido en los foros de Mindstorms. Para aprovechar al máximo la capacidad de NXC existe un firmware mejorado. [8]

Usamos el entorno Bricx Command Center (BricxCC) para programar en NXC, un entorno de programación desrrollado inicialmente por Mark Overmans. En su origen, se creó para programar en lenguaje NQC (Not Quite C) haciendo uso del anterior modelo de ladrillo, RXC. Durante los últimos años, se han realizado cambios en el entorno original con el objetivo de poder ser utilizado para el modelo NXT y el lenguaje NXC. Este entorno permite escribir los programas, descargarlo en el ladrillo NXT, iniciar su ejecución y detenerlos, así como también permite indagar acerca del estado de la memoria flash del NXT, convertir archivos de sonido para ser usados por el ladrillo y mucho más. BrixCC actúa más como un procesador de texto, pero con algunas características adicionales. [9]

Este entorno de programación se compone de:

- **Una barra de menú:** incluye las operaciones que se pueden realizar con los archivos, comandos de edición, compilación y de descarga de programas hacia el ladrillo NXT.
- **Una barra de herramientas:** incluye las utilidades de la barra de menú más usadas.
- **Un editor:** Separado en dos partes: un árbol donde se muestra una lista de elementos del lenguaje y la sección de edición. El editor provee al usuario de un soporte de alto nivel para desarrollar programas NXC.

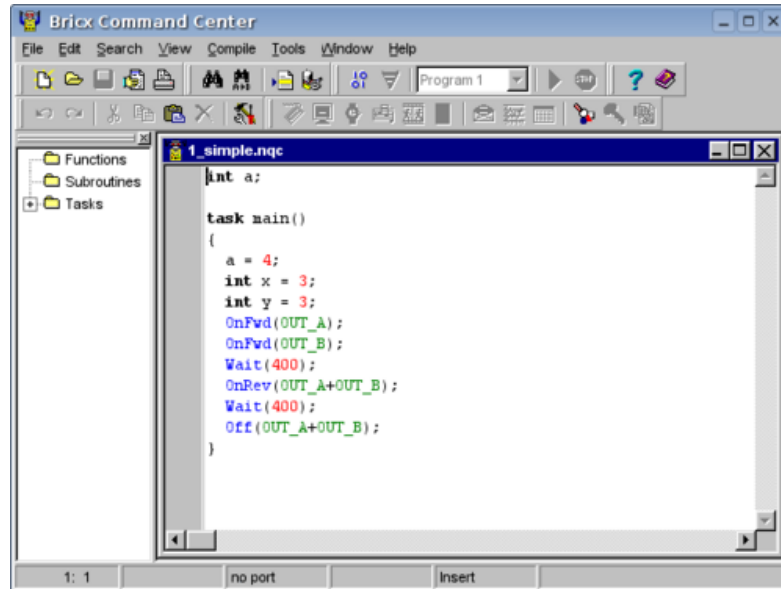


Ilustración 10- Detalle del entorno de programación BrixCC

2. METODOLOGÍA SEGUIDA EN EL DESARROLLO DEL TRABAJO

2.1 Descripción de tareas y fases

Las tareas que han permitido alcanzar los objetivos planteados anteriormente son las siguientes:

1. Estudio de viabilidad del proyecto. Recopilación de información sobre el funicular real.
2. Elección del diseño y de los materiales.
3. Construcción de la maqueta en base al diseño realizado.

4. Diseño y programación del sistema de control mediante BricxCC (NXG) para el control de los motores y sensores del kit de Lego Mindstorms.

2.1.1 Estudio de viabilidad

Esta fase se corresponde con el análisis de alternativas ya explicado anteriormente.

2.1.2 Diseño de la estructura

Una vez terminado el análisis previo, llega el momento de realizar un diseño más detallado. Para ello, se realiza manualmente el alzado, planta y perfil de cada uno de los elementos que componen el sistema. Todas las cotas están en milímetros.

- **Base inclinada.**

A continuación, se incluyen esquemas en los que se puede ver el alzado, la planta y el perfil de la base. El primer esquema muestra las dimensiones de la base, en el segundo se han incluido las vías y sus dimensiones. Los detalles de la bifurcación se explican más abajo.

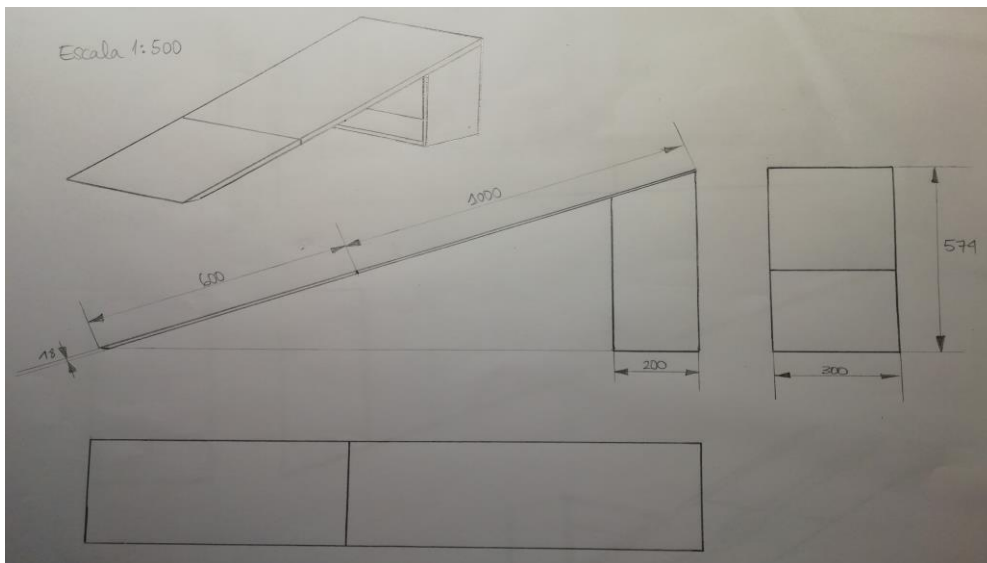


Ilustración 11- Vistas de la base inclinada

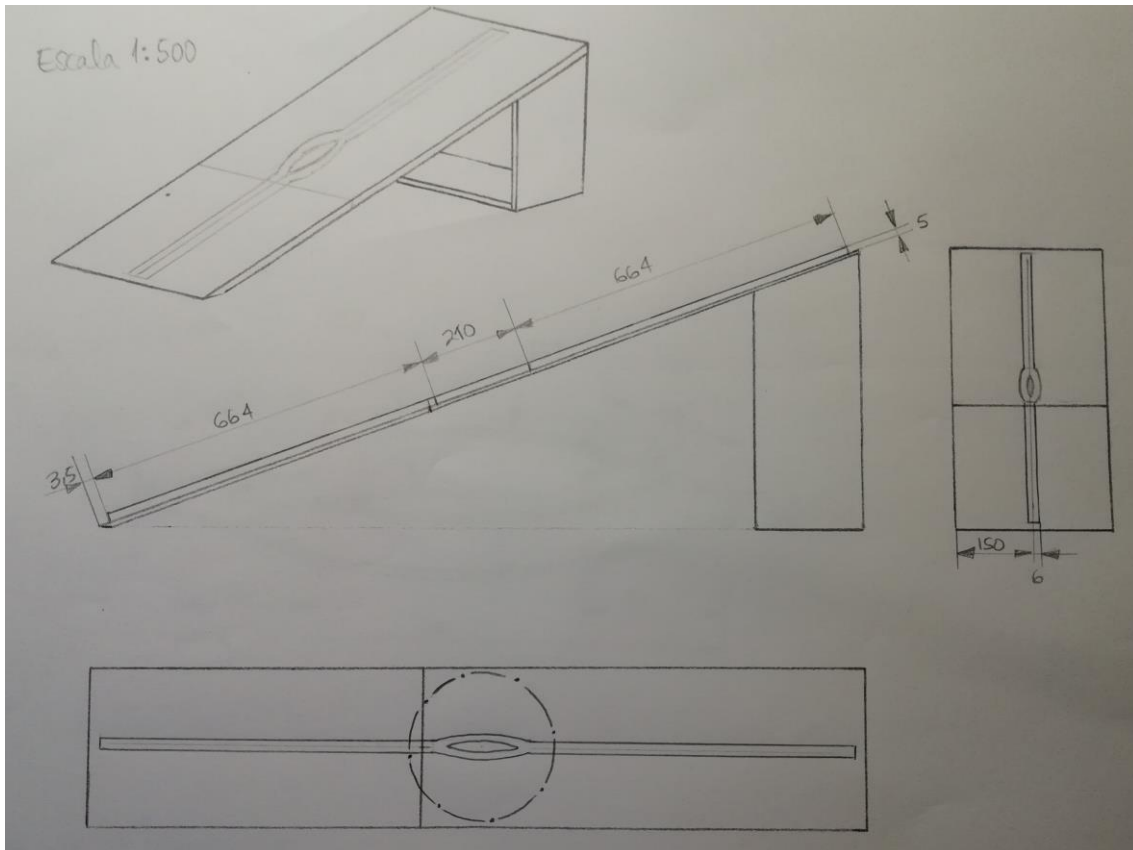


Ilustración 12- Posición de las vías y sus dimensiones

○ **Vagones.**

Para el caso de los vagones, las dimensiones se han adaptado al tamaño de las piezas de Lego y, por tanto, no son las mismas que las que se habían calculado en base a la escala 1:100, aunque son lo más parecidas posible.

Podemos ver las dimensiones finales en el siguiente esquema.

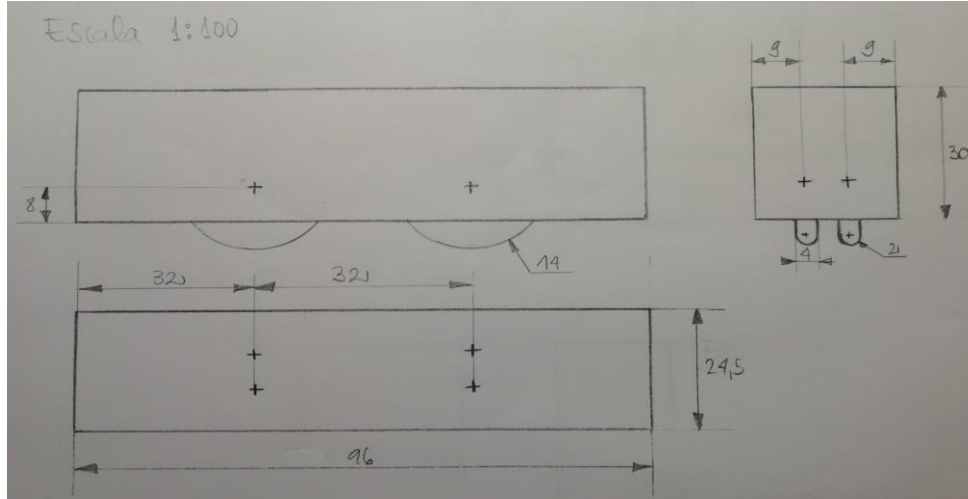


Ilustración 13- Vistas de los vagones

○ **Mecanismo de desvío.**

El siguiente esquema muestra las distintas dimensiones de la bifurcación en detalle. Como ya se ha expuesto en el diseño básico, las agujas son las partes móviles de las vías que pueden rotar alrededor de un eje, gracias a un servomotor, para dirigir cada vagón hacia una vía u otra, evitando así que colisionen entre ellos. Este servomotor se acciona mediante un sensor fotosensible que detecta el paso de los vagones por el punto medio de la bifurcación y, pasado un tiempo establecido con un temporizador, vuelve a su posición inicial.

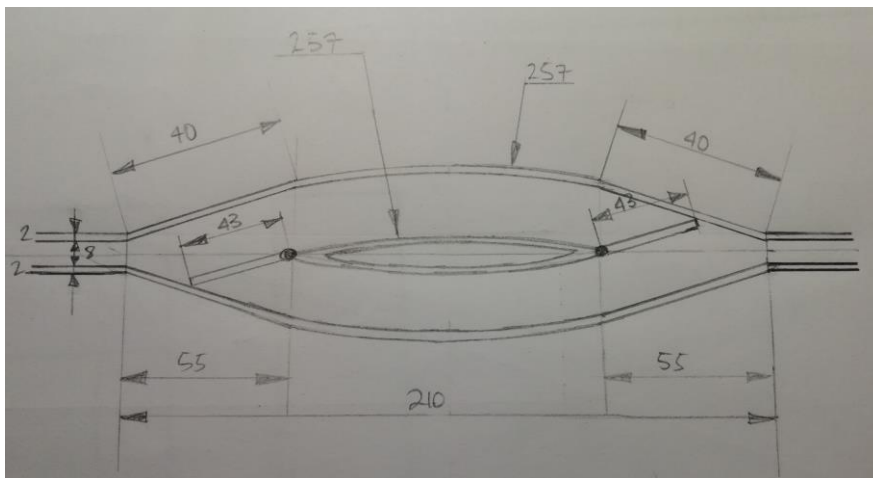


Ilustración 14- Detalle de la bifurcación

○ **Estaciones.**

Las estaciones se han dimensionado de tal forma que el resultado sea estéticamente bueno, no en base a su verdadero tamaño. La estación superior es de intemperie y la inferior subterránea (dentro de un edificio), al igual que en el caso real, aunque en este el último tramo de la vía es subterráneo.

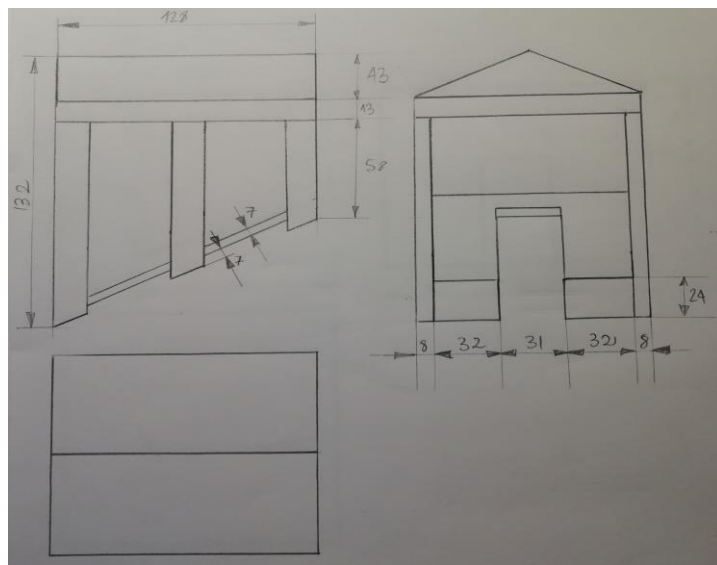


Ilustración 15- Vistas de la estación superior

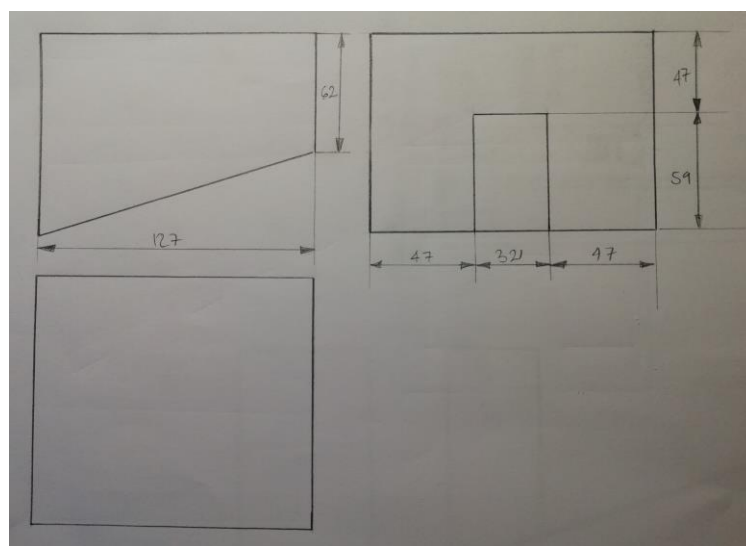


Ilustración 16- Vistas de la estación inferior

○ **Sistema de control.**

A continuación, se adjunta un esquema en el que puede verse la localización de los elementos de control sobre la maqueta. El display ha sido colocado de tal modo que los cables lleguen correctamente a todos los motores y sensores. También se ha incluido la localización de las estaciones.

El eje que mueve el cable se recubre con un trozo de cámara de bicicleta, sujeto con dos bridas, para evitar el deslizamiento.

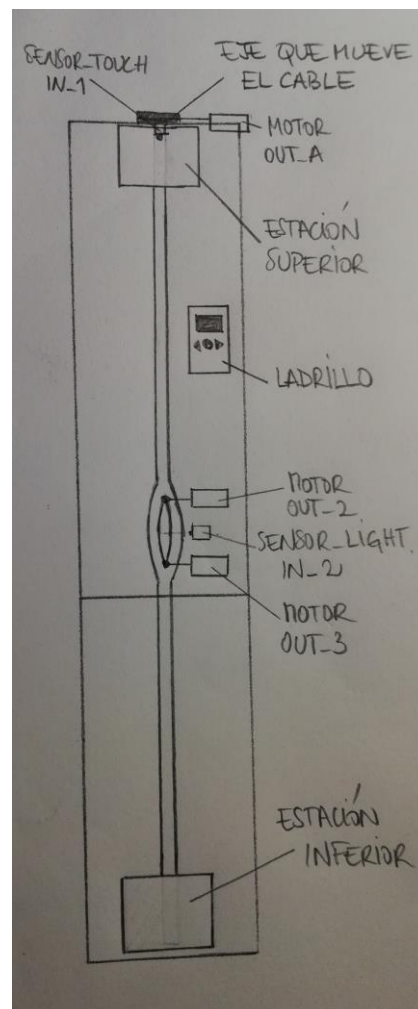


Ilustración 17- Localización de las estaciones y los elementos de control

2.1.3 Construcción

Una vez terminada la fase de diseño comienza la de construcción. A continuación, se detallan los pasos seguidos, con una breve descripción y con información de las herramientas y materiales empleados. También se añaden fotografías para describir gráficamente el proceso de construcción.

(i) Construcción de la base

Descripción: Recortar el tablón a las dimensiones deseadas, cortar en las dos partes (superior e inferior), preparar cuatro trozos de madera y taladrarlos para conseguir unir las dos piezas. Atornillarlo todo.

Herramientas: Sierra de calar, Taladro y brocas, Destornillador con punta de llave allen.

Piezas: Tablón de madera de pino, tornillos con cabeza de allen, palomillas.



Ilustración 18- Corte de la base con sierra de calar



Ilustración 20- Preparación de las piezas de unión de las dos partes de la base



Ilustración 19- Unión de las dos piezas de la base con tornillos y palomillas

(ii) Construcción de las vías

Descripción: Dibujar la posición de las vías, recortar los listones de madera de tilo y encolarlos en la posición deseada, con cuidado de que encajen las vías de los tramos superior e inferior, limar para mejorar la unión

Herramientas: Cúter, Cola blanca, Lima
Piezas: Listones de madera de tilo



Ilustración 22- Encolado de las vías



Ilustración 21- Encolado de la bifurcación



Ilustración 24- Unión de la vía de ambas piezas de la base

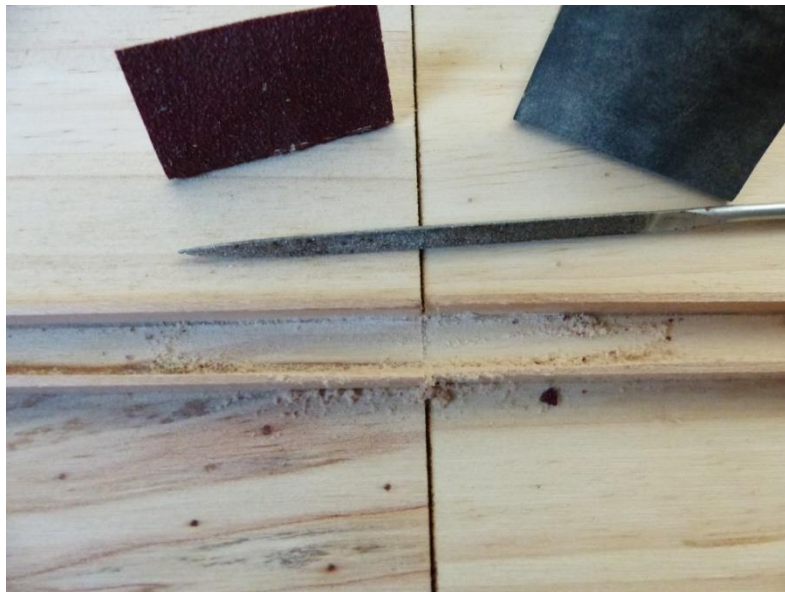


Ilustración 23- Lijado de la unión de vías para evitar desniveles

(iii) Construcción de las patas

Descripción: Recortar el tablón a las dimensiones deseadas, unir las patas a la base y entre ellas.

Herramientas: Sierra de calar, Taladro y brocas, Destornillador con punta de estrella.

Piezas: Tablón de madera de pino, tirafondos



Ilustración 25- Detalle de las patas



Ilustración 26-Vista de la base inclinada con las vías



Ilustración 27- Taladrado y atornillado de las patas a la estructura

(iv) Construcción de las agujas

Descripción: Recortar los listones al tamaño necesario, recortar la varilla de madera y pegarlo. Taladrar los agujeros donde se alojan las agujas

Herramientas: Cúter, Cola blanca, Taladro + broca

Piezas: Listones de madera de tilo, varilla de madera

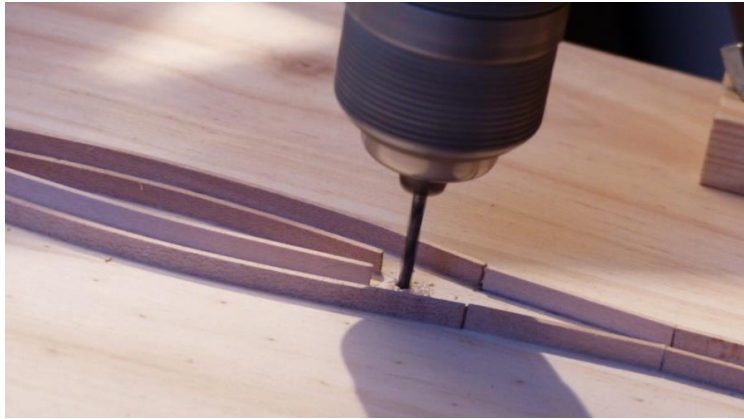


Ilustración 29- Taladrado del orificio donde se alojan las agujas



Ilustración 28- Preparación de las agujas

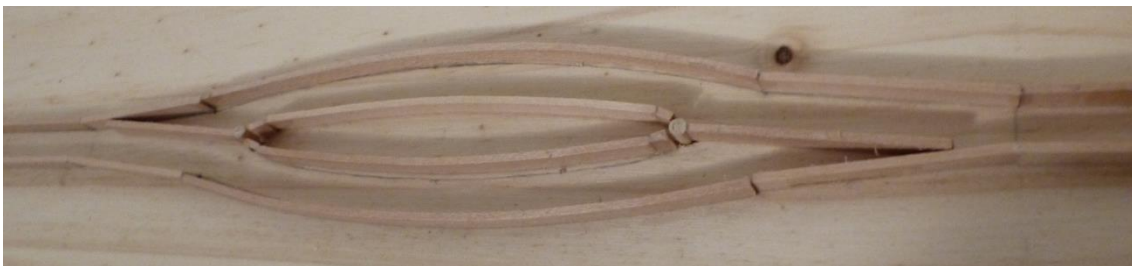


Ilustración 30- Bifurcación con las agujas ya colocadas

(v) Construcción de las estaciones y los vagones

Descripción: Construir las estaciones superior e inferior con piezas Lego y colocarlas en la base con la ayuda de la madera de balsa. Construir los vagones con piezas Lego.

Herramientas: Cúter, Cola blanca

Piezas: Piezas de Lego, Madera de balsa



Ilustración 31- Vista de los vagones en la bifurcación

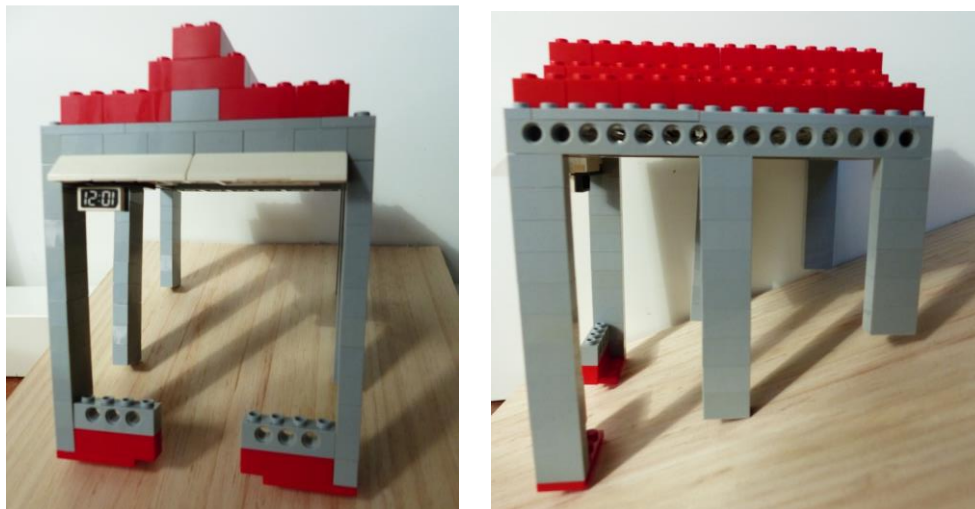


Ilustración 32- Vistas de la estación superior



Ilustración 33- Vistas de la estación inferior

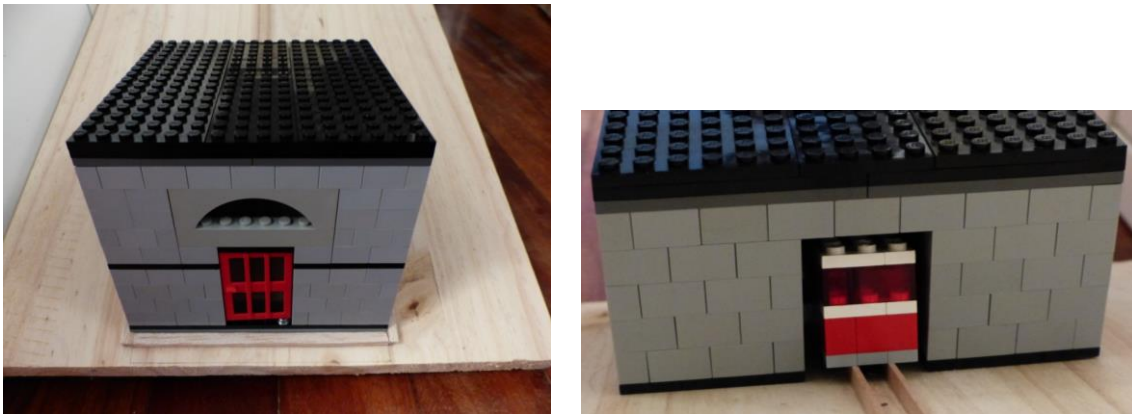


Ilustración 34- Estación inferior colocada en la base



Ilustración 35- Elementos de sujeción de la estación superior en la base

(vi) Colocación del sensor, los motores y el display

Descripción: Con la ayuda de la madera de balsa, colocar sensores, motores y display

Herramientas: Cúter, Cola blanca

Piezas: madera de balsa



Ilustración 36- Encolado de los elementos de sujeción del display sobre la base



Ilustración 39- Colocación del servomotor principal



Ilustración 38- Vista de los elementos de control sobre la base



Ilustración 37- Sensor de contacto en la estación superior

(vii) Incorporación del cable a los vagones

Descripción: Unir los vagones entre ellos con el cable de acero de la longitud necesaria, con la ayuda de los alicates aplastar las piezas cilíndricas para fijar el cable.

Herramientas: Alicates

Piezas: Cable de acero, piezas cilíndricas para fijar el cable

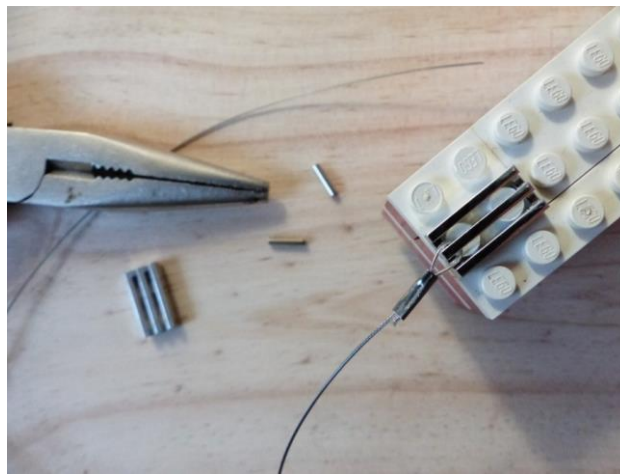


Ilustración 41- Unión del cable a los vagones

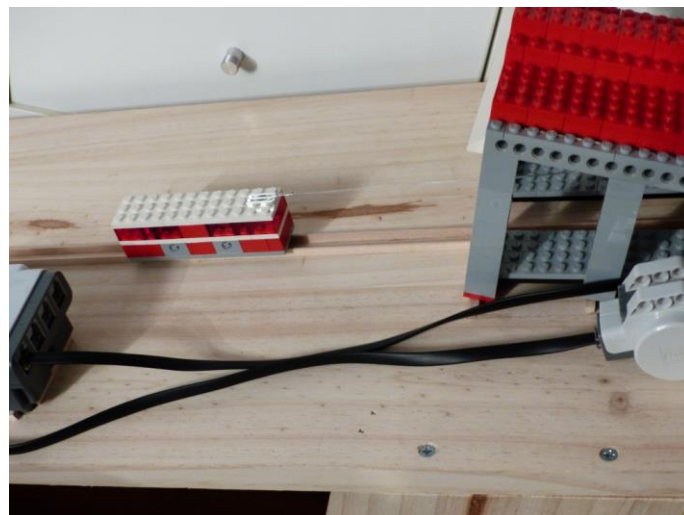


Ilustración 40- Vista del vagón con los cables

(viii) Mejora de la apariencia visual

Descripción: Recortar la tela con la forma adecuada para simular la hierba del monte y pegarla en su lugar, pintar las vías.

Herramientas: Cúter, Cola blanca, pincel

Piezas: pintura, tela de fieltro verde

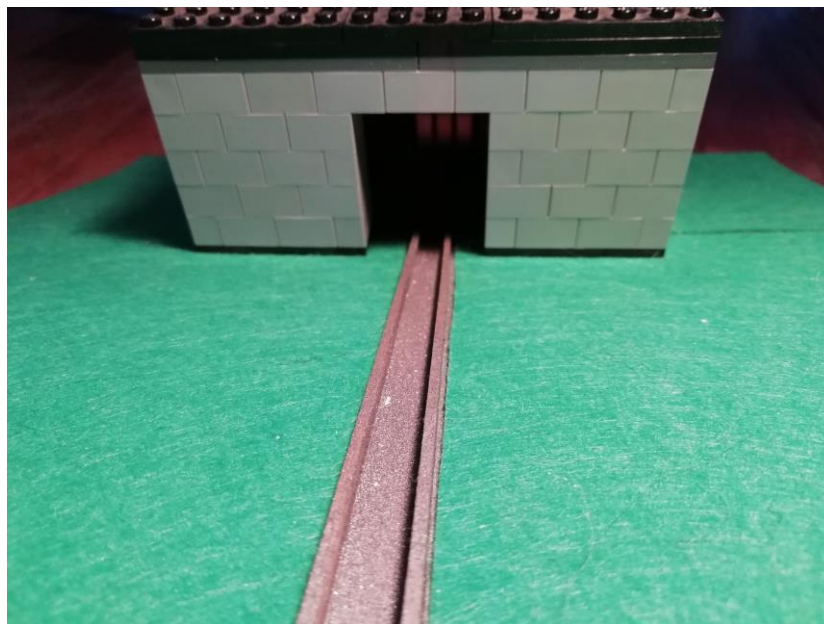


Ilustración 42- Detalle de la base con la tela verde y las vías pintadas

2.1.4 Programación

Acabada la construcción, llega la hora de programar en lenguaje NXC el control de movimiento de los vagones y el control de movimiento de las agujas. Para la redacción del código se ha utilizado el Tutorial de NXC para programar Robots Lego Mindstorms NXT, de Daniele Benedettelli [10]. El código de programa se encuentra detallado en el Anexo I.

2.1.4.1 Diseño del control de movimiento de los vagones

El objetivo principal de control es que un vagón suba y el otro baje, movidos por el motor principal (OUT_A), hasta llegar a las estaciones, en ese punto el vagón que se encuentra en la estación superior pulsa el sensor táctil (IN_1). Los vagones se detienen dos segundos, simulando las paradas de funicular real para que monten los usuarios de este, y reanudan la marcha en sentido inverso. También se puede leer la palabra “PARADA” en la pantalla del ladrillo mientras los vagones se encuentran en las estaciones.

A continuación, se incluye un diagrama de flujo que representa este movimiento.

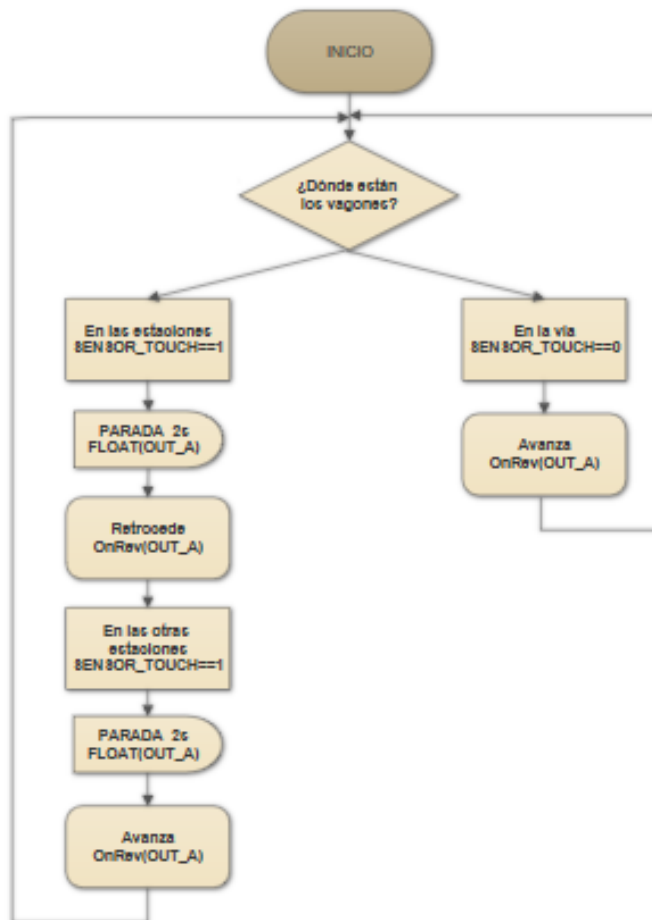


Ilustración 43- Diagrama de flujo del movimiento principal

2.1.4.2 Control de las agujas de la bifurcación

En su movimiento de subida y bajada, los vagones llegan a la bifurcación, donde las agujas los dirigen hacia el lado de la vía que corresponde. Una vez dentro, un sensor fotosensible detecta los vagones y manda la señal al display para que los motores secundarios (OUT_B y OUT_C) cambien la posición de las agujas, permitiendo así la salida de la bifurcación de cada vagón. También se configura el sistema para que, automáticamente, las agujas vuelvan a su posición inicial dos segundos y medio después.

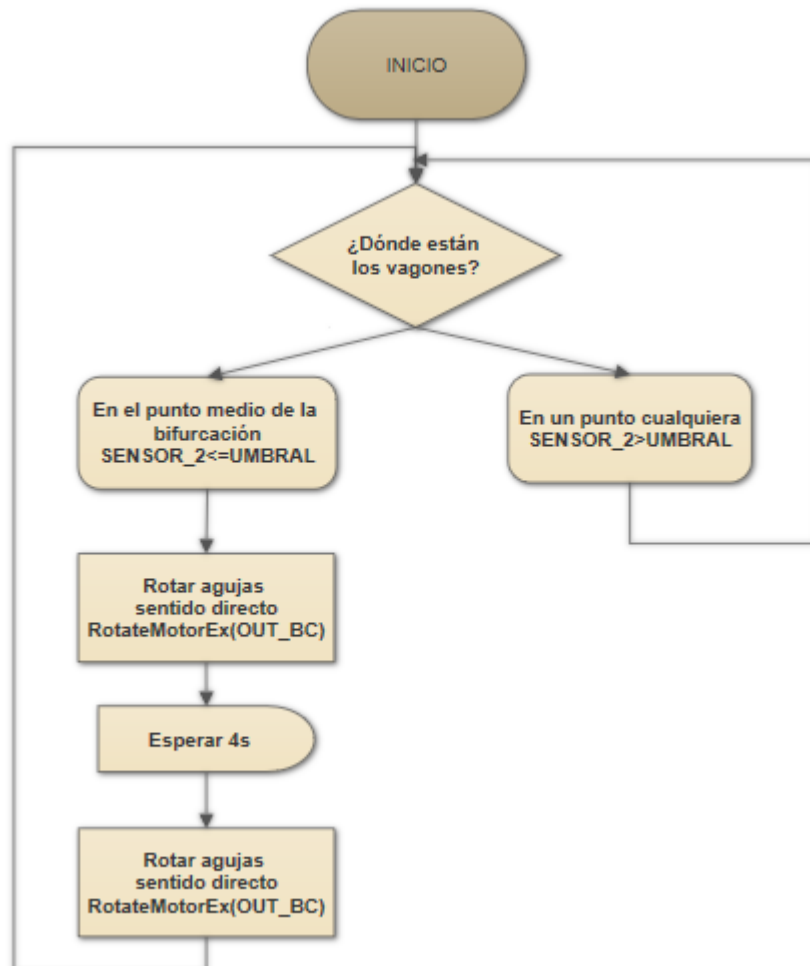


Ilustración 44- Diagrama de flujo del control de la bifurcación

La gráfica siguiente muestra la relación entre las variables de entrada y salida a lo largo del tiempo.

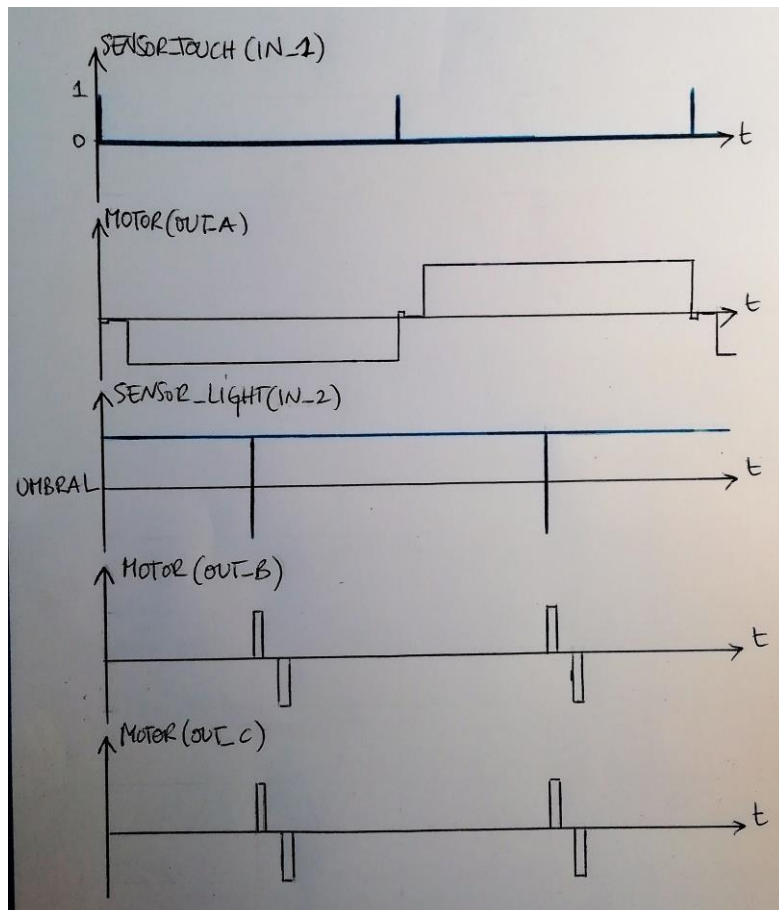


Ilustración 45- Relación temporal entre entradas y salidas

2.1.4.3 Programación de la calibración del sensor fotosensible.

El sensor de luz permite que el robot pueda percibir colores. Normalmente se emplea para distinguir negro y blanco.

El sensor informa al robot transmitiéndole un dato numérico cuyo valor está entre 0 y 100. 0 indicará que el sensor de luz está leyendo color negro -oscuro absoluto- y 100 blanco -luminosidad absoluta-.

Lo primero que hay que hacer cuando se van a utilizar sensores de luz es calibrarlos. Calibrar significa obtener los valores reales de luminosidad cuando el sensor o los sensores están enfocando a color negro y a color blanco [11].

Por ello se ha creado una función que permite calibrar el sensor cada vez que se va a ejecutar el programa. El objetivo de esta función es leer los actuales valores de negro y blanco y calcular el valor umbral como la media de ambos. Esto permite que el resultado sea óptimo independientemente del nivel de luminosidad que haya en el ambiente.

2.2 Planificación del proyecto. Diagrama de Gantt

A continuación, se presenta la planificación del proyecto desde el principio hasta el final. En primer lugar, se presenta una lista con las diferentes tareas realizadas a lo largo del proyecto y después el diagrama de Gantt correspondiente.

La fecha de inicio del proyecto es el 1 de febrero de 2018 y la fecha de finalización el 4 de septiembre de 2018, fecha de la defensa oral. El trabajo no ha sido continuo, ya que no se ha trabajado en el proyecto en los periodos de exámenes, ni cuando la carga laboral por parte de otras asignaturas del grado ha sido muy elevada. En el diagrama de Gantt se pueden ver estos periodos, en los que no se ha realizado ninguna tarea. Las fechas en cuestión son las siguientes:

- Del 21 de mayo de 2018 hasta el 8 de junio de 2018
- Del 25 de junio de 2018 hasta el 7 de julio de 2018

2.2.1 Listado de tareas

A continuación, podemos ver un listado de todas las tareas realizadas, y por realizar, en el proyecto, junto a una breve descripción y la duración total de la misma. Las tareas han sido realizadas por el alumno en colaboración con el director del proyecto.

1. Búsqueda de información

Descripción: Recopilación de información sobre el funcionamiento real del funicular y sobre el kit Lego Mindstorms.

Duración: 10 días

2. Estudio de viabilidad

Descripción: Estudio previo del proyecto en el que se definen los objetivos, el alcance, las distintas alternativas, los posibles problemas previstos, etc.

Duración: 15 días

3. Diseño

Descripción: Elección del diseño y de los materiales que van a emplearse.

Duración: 5 días

4. Adquisición de los materiales

Descripción: Periodo de recopilación de los distintos materiales y piezas necesarias para la construcción de la maqueta.

Duración: 7 días

5. Construcción de la maqueta

Esta fase ya se ha explicado anteriormente por lo que solo se especifican las duraciones de cada subtarea.

Duración total: 15 días

(i) Construcción de la base

Duración: 2 días

(ii) Construcción de las vías

Duración: 1 día

(iii) Construcción de las patas

Duración: 1 día

(iv) Construcción de las agujas

Duración: 1 día

(v) Construcción de las estaciones y los vagones

Duración: 2 días

(vi) Colocación del sensor, los motores y el display

Duración: 2 día

(vii) Incorporación del cable a los vagones

Duración: 1 día

(viii) Mejora de la apariencia visual

Duración: 5 días

6. Programación del sistema

Descripción: diseñar el programa de control del sistema, con la ayuda del programa BrixCC, compilarlo y cargarlo en el display.

Duración: 15 días

7. Pruebas experimentales y solución de imprevistos

Descripción: Probar el funcionamiento del sistema y adaptar los valores de velocidad y tiempo de funcionamiento hasta conseguir el resultado adecuado. Solucionar los problemas que han surgido. Por ejemplo, cambiar el hilo de acero por uno de algodón, más flexible.

Duración: 15 días

8. Preparación de la presentación oral

Descripción: Preparar el PowerPoint de apoyo a la presentación oral, que sea visual, una ayuda para la comprensión del público. Trabajar en lo que se va a decir y en cómo decirlo.

Duración: 15 días

9. Presentación oral

Descripción: Defensa oral del Trabajo de fin de grado. Fecha a asignar por la Escuela.

Duración: 1 día

A continuación, se presenta el diagrama de Gantt del proyecto. La línea roja indica la fecha actual, lo que nos permite ver que tareas han finalizado, cuales están en curso y cuales pendientes de realizar.

🔧	Título	Fecha de inicio	Fecha de venc...	Duración
1	▼ TFG	02/01/2018	09/04/2018	
2	Busqueda de información	02/01/2018	02/14/2018	10d
3	Estudio de viabilidad	02/15/2018	03/07/2018	15d
4	Diseño	03/08/2018	03/14/2018	5d
5	Adquisición de materiales	03/15/2018	03/23/2018	7d
6	▼ Construcción de la maqueta	03/26/2018	03/26/2018	1d
7	Construcción de la base	03/26/2018	03/26/2018	1d
8	Construcción de las vías	03/27/2018	03/27/2018	1d
9	Construcción de las patas	03/28/2018	03/28/2018	1d
10	Construcción de las agujas	03/29/2018	03/29/2018	1d
11	Construcción de estacione...	03/30/2018	04/02/2018	2d
12	Colocación sensores, moto...	04/03/2018	04/04/2018	2d
13	Incorporación del cable a l...	04/05/2018	04/05/2018	1d
14	Mejora de la apariencia vis...	04/06/2018	04/12/2018	5d
15	Programación del sistema	04/13/2018	05/03/2018	15d
16	Pruebas experimentales y sol...	05/04/2018	05/17/2018	10d
17	Preparación de la presentaci...	07/09/2018	07/27/2018	15d
18	Presentación oral	09/04/2018	09/04/2018	1d

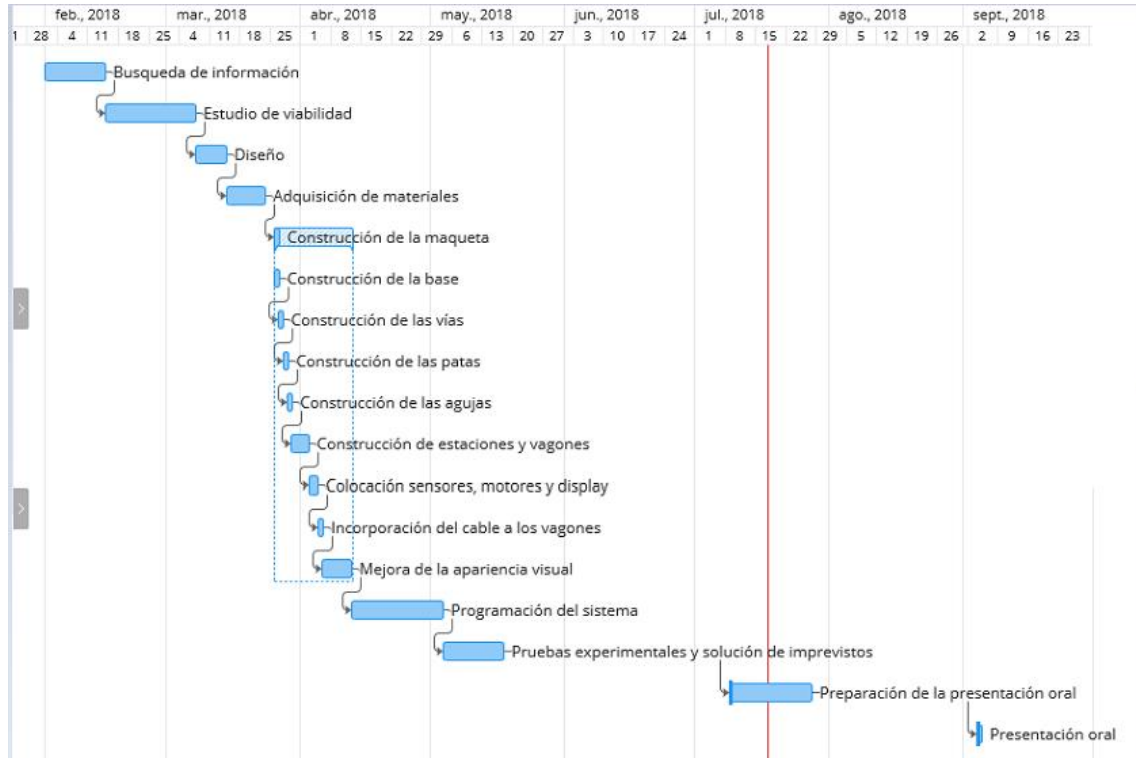


Ilustración 46- Diagrama de Gantt

3. ASPECTOS ECONÓMICOS. PRESUPUESTO EJECUTADO

En este apartado se recogen los diferentes gastos que ha implicado la realización del trabajo. Se ha dividido los gastos en cuatro apartados, por un lado, los gastos derivados del montaje, después los gastos relacionados con el hardware de Lego Mindstorms, por otro lado, los relacionados con el software empleado y por último los gastos de mano de obra. Todos los precios incluyen el IVA. La última tabla muestra el presupuesto final, al que se la ha añadido un 10% del presupuesto calculado para poder cubrir posibles imprevistos.

3.1 Gastos relacionados con el montaje

Producto	Precio (€)
Kit piezas Lego Classic	34,95
Tablero madera pino 2000x500x18	11,95
Pack 7 listones madera tilo 1000x5x2	3,00
Pack 20 palomillas M6	3,99
Pack 10 tornillos cabeza cilíndrica allen M6x60	2,05
Pack 15 tornillos tirafondo	1,59
Hilo acero trenzado Ø0.4mm 10m	2,59
Varilla de haya Ø3mm 1m	0,60
Cola blanca	1,03
Tela fieltro verde 1mx2m	1,50
Bote pintura plateada	3,00
	66,25

Tabla 3- Presupuesto materiales construcción

3.2 Gastos Hardware Lego Mindstorms

Producto	Precio (€)
Kit Lego Mindstorms NXT (ladrillo + sensores + servomotores + piezas technic)	242,75
	242,75

Tabla 4- Presupuesto Gastos Hardware.

3.3 Mano de obra

Concepto	Precio €/h	Horas	Precio (€)
Mano de obra	8,00	150	1.200,00
			1.200,00

Tabla 5- Presupuesto Mano de obra.

3.4 Software utilizado

Software Utilizado	Gasto (€)
Licencia Microsoft Project Standar	769,00
Licencia anual Microsoft Oficce 365 Hogar y Empresas	279,00
	1.048,00

Tabla 6- Presupuesto software

3.5 Presupuesto ejecutado

Presupuesto ejecutado (€)	
Materiales construcción	66,25
Hardware	242,75
Mano de obra	1.200,00
Software	1.048,00
Subtotal	2557,00
Imprevistos (10%)	255,70
Total	2812,70

Tabla 7- Presupuesto final.

4. CONCLUSIONES

Para terminar, se han incluido las principales conclusiones a las que se ha llegado mediante la realización del presente TFG.

Hemos visto que, combinando conocimientos de Gráficos de Ingeniería, Mecánica, Ciencia de Materiales, Programación, y otras muchas asignaturas, se puede planificar, diseñar, construir y programar una maqueta que simule el comportamiento del funicular de Artxanda.

También hemos visto las limitaciones que pueden surgir al realizar el proyecto. Algunas se pueden prever y otras aparecen a lo largo del proceso. Tienen distinta naturaleza, pueden ser técnicas, temporales y/o económicas, y distinto impacto, algunas tienen solución y otras no, lo que implica buscar alternativas.

En cuanto a la programación, existen numerosos lenguajes de programación y numerosas plataformas para crear los códigos del programa de control. Se han comparado las posibles opciones en base a distintos criterios de elección para elegir la más adecuada.

Todo esto nos lleva a la siguiente reflexión: ¿Es posible recrear a escala cualquier elemento? ¿Hasta qué punto puede llegar a parecerse el comportamiento de un robot al del objeto real que simula?

5. BIBLIOGRAFÍA

- [1] Breve historia del Funicular de Artxanda. Página web oficial del ayuntamiento de Bilbao: www.bilbao.eus.
- [2] Robótica educativa en la enseñanza de las matemáticas e integración transversal de asignaturas científicas y humanísticas. Iván Esteban Perez. Universidad de las Américas (Chile).
- [3] Lego Mindstorms NXT User's guide. The Lego group. 2009. Página oficial Lego. [Formato PDF].
- [4] Uspinjaca Funicular. Sven Franic. 21 de abril de 2017. Disponible online en: www.flickr.com.
- [5] Característica técnicas del funicular de Artxanda. Página web oficial del ayuntamiento de Bilbao: www.bilbao.eus.
- [6] Página web oficial de Robot-C: www.robotc.net.
- [7] NXT-G: the development environment supplied with Lego Mindstorms, NXT-G. Vanessa Mazzari. 26 de septiembre de 2015.
- [8] Uso conjunto de la plataforma Lego Mindstorms NXT y metodologías PBL en Informática Industrial. Isidro Calvo, Gorka Parianez. Departamento de Sistemas y Automática. Escuela Universitaria de Ingeniería de Vitoria-Gasteiz. UPV/EHU.
- [9] Construcción y programación de un grupo de robots móviles sobre la base del producto Lego Mindstorms NXT. Medardo Ángel Silva Amores. Julio de 2011. Escuela Politécnica Nacional. Quito.
- [10] Tutorial de NXC para programar Robots Lego Mindstorms NXT. Daniele Benedettelli, revisado por John Hansen. 2007. [Formato PDF],
- [11] Taller de robótica 2014. Sesión 5. El blog de tecnología del IES Sefarad (Toledo). Disponible online en: www.tecnosefarad.com

ANEXO I: Código en lenguaje NXC

Programa para el control de la velocidad angular del motor

```
int calibracion()      /*Función para la calibración del sensor fotosensible */
{
    bool pulsado=FALSE ;
    int negro;
    int blanco;
    int umbral1;
    while(pulsado==FALSE)
    {
        pulsado=ButtonPressed(BTNRIGHT,TRUE);
    }
    pulsado=FALSE;
    PlayTone(4000,500);      /*Mediante un pitido anuncia que está
preparado para calibrar*/
    Wait(500);
    negro=SENSOR_2;      /*Asignar el valor negro (0) al valor leído por el
sensor, que se ha colocado delante de un fondo negro antes de pulsar el botón
derecho*/
    NumOut(6,LCD_LINE3,negro);      /*Muestra el valor leído en la
pantalla*/
    PlayTone(5000,500);
    Wait(500);      /*Se repite el procedimiento para asignar el valor del color
blanco*/
    while(pulsado==FALSE)
    {
        pulsado=ButtonPressed(BTNRIGHT,true);
```

```

    }
    pulsado=FALSE;
    PlayTone(4000,500);
    Wait(500);
    blanco=SENSOR_2;
    NumOut(6,LCD_LINE3,blanco);
    PlayTone(5000,500);
    Wait(500);
    while(pulsado==FALSE)
    Wait(2000);
    umbral1=(negro+blanco)/2;      /*Calcula el valor umbral como media de
los valores blanco y negro medidos*/
    NumOut(6,LCD_LINE3,blanco);
    Wait(2000);
    ClearScreen();
    return umbral1;                /*Devuelve el valor calculado*/
}

int UMBRAL;      /* Inicializa la variable UMBRAL*/
task main()      /*Programa principal*/
{
    SetSensor(IN_1,SENSOR_TOUCH);      /*Asignar los sensores a sus
correspondientes entradas*/
    SetSensorLight(IN_2);      /*Sensor fotosensible en modo de luz activa*/
    UMBRAL=calibracion();      /*Asigna a la variable UMBRAL el valor
devuelto por la función calibración*/
    while (true)      /*Comienzo del ciclo infinito*/
    {

```

```
repeat(1)          /*Repetir el proceso*/
{
    OnFwd(OUT_A,30);      /*El motor comienza a girar*/
    if (SENSOR_2 < UMBRAL) /*Comprueba si los
vagones están en la bifurcación*/
    {
        Wait(500);
        RotateMotorEx(OUT_BC,20,14,0,true,true); /*Gira
los dos motores de manera sincronizada*/
        Wait(2500);
        RotateMotorEx(OUT_BC,20,-14,0,true,true);
    }
    until(SENSOR_1==1); /*Comprueba si los vagones están
en las estaciones*/
    OnRev(OUT_A,10); Wait(500); /*Invierte el sentido de
giro*/

    TextOut(30,LCD_LINE3,"PARADA",true);
    Float(OUT_A);      /*Detiene lentamente el motor*/
    Wait(2000);
    ClearScreen();    /*Limpia la pantalla*/
    OnRev(OUT_A,30);
    if (SENSOR_2 < UMBRAL)
    {
        Wait(500)
        RotateMotorEx(OUT_BC,20,14,0,true,true);
        Wait(2500);
        RotateMotorEx(OUT_BC,20,-14,0,true,true);
    }
    until(SENSOR_1==1);
```

```
OnFwd(OUT_A,10); Wait(500);  
TextOut(30,LCD_LINE3,"PARADA",true);  
Float(OUT_A);  
Wait(2000);  
ClearScreen();  
OnFwd(OUT_A,30);  
    }  
}  
}
```

ANEXO II: Listado de elementos

Elemento	Cantidad
LEGO Ladrillo Mindstorms NXT (9841)	1
LEGO Servomotor NXT (9842)	3
LEGO Sensor táctil NXT (9843)	1
LEGO Sensor fotosensible NXT (9844)	1
LEGO Cables conexión 37cm	3
LEGO Cables conexión 50cm	2
LEGO Technic Eje 95mm	1
LEGO Technic Eje 25mm	4
LEGO Technic Ruedas + Neumáticos (2815 + 4185)	4
LEGO Classic Parts	85
Listones madera de tilo 1000x5x2mm	3
Tablón madera de pino 200x500x18mm	1
Tornillos cabeza cilíndrica allen M6x60mm	6
Palomillas M6	6
Tornillos tirafondos 40mm	14
Tornillo tirafondos 25mm	1
Hilo de algodón	161cm
Cámara de rueda de bicicleta	1
Bridas	2
Tela fieltro verde 200x100mm	1
Pintura plateada	