



# Tecnológico de Monterrey

*Reporte integrador*

*MR2004B.1 Implementación de sistemas mecatrónicos*

*Rodrigo Orta Guerra*

*Román Amezcua Castrejón*

*Miguel Angel Trujillo Jacobo*

*Martín Alonso Sinsel Duarte*

*Instituto Tecnológico de Estudios Superiores de Monterrey Zapopan, Jalisco, México*

Belén Muñoz A01636143

Oscar Alfredo Mercado Rico A01638228

María Ximena Zapata Magdaleno A01251683

Francisco Javier López Sánchez A01637518

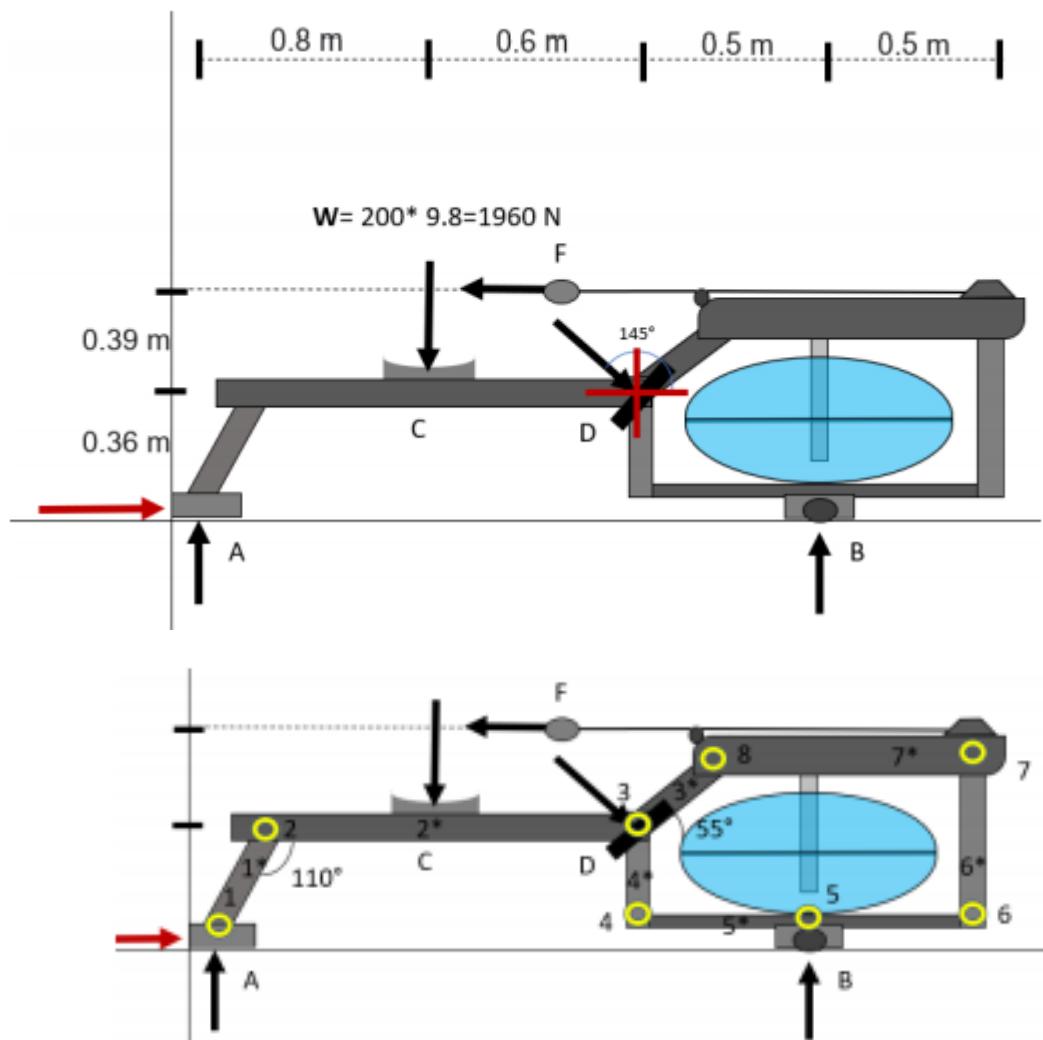
Sábado 12 de Junio 2021

## **Base y funcionamiento mecánico**

*Materiales utilizados:*

La remadora tiene un largo de 240 cm , un ancho de 61 cm y una altura del piso al asiento de 36 cm. El asiento tiene un ancho de 40 cm y un largo de 30 cm con un rango aproximado de desplazamiento horizontal de 100 cm.

Con base al análisis estático y esfuerzos de la remadora se seleccionaron los materiales adecuados para la estructura, considerando su accesibilidad, facilidad de manufactura, y que soporten las cargas presentes sin sufrir deformaciones permanentes ni fracturas.



**Figura 1.** DCL de la remadora (diseño inicial).

Cálculos de las cargas:

$$\Sigma F_x = 0 \quad \Sigma F_y = 0 \quad \Sigma M = 0$$

$$\Sigma F_x = -F + F_D \cos(35^\circ) = 0$$

$$F_D = \frac{F}{\cos(35^\circ)}$$

$$F_{Dy} = F * \tan(35^\circ)$$

$$\Sigma M_B = Ftan(35) * (0.5) + 1960 * (1.1) - A * (1.9) = 0$$

$$A = Ftan(35) \frac{0.5}{1.9} + \frac{2156}{1.9}$$

$$\Sigma M_A = -1960 * (0.8) - Ftan(35)(1.4) + B * (1.9) = 0$$

$$B = Ftan(35) \frac{1.4}{1.9} + \frac{1568}{1.9}$$

Resultados reemplazando F:

**14 kg**

$$F_D = 167.5 \text{ N}$$

$$A = 1160 \text{ N}$$

$$B = 908 \text{ N}$$

**18 kg**

$$F_D = 215.4 \text{ N}$$

$$A = 1167.3 \text{ N}$$

$$B = 916.3 \text{ N}$$

**Figura 2.** Análisis estático de la remadora

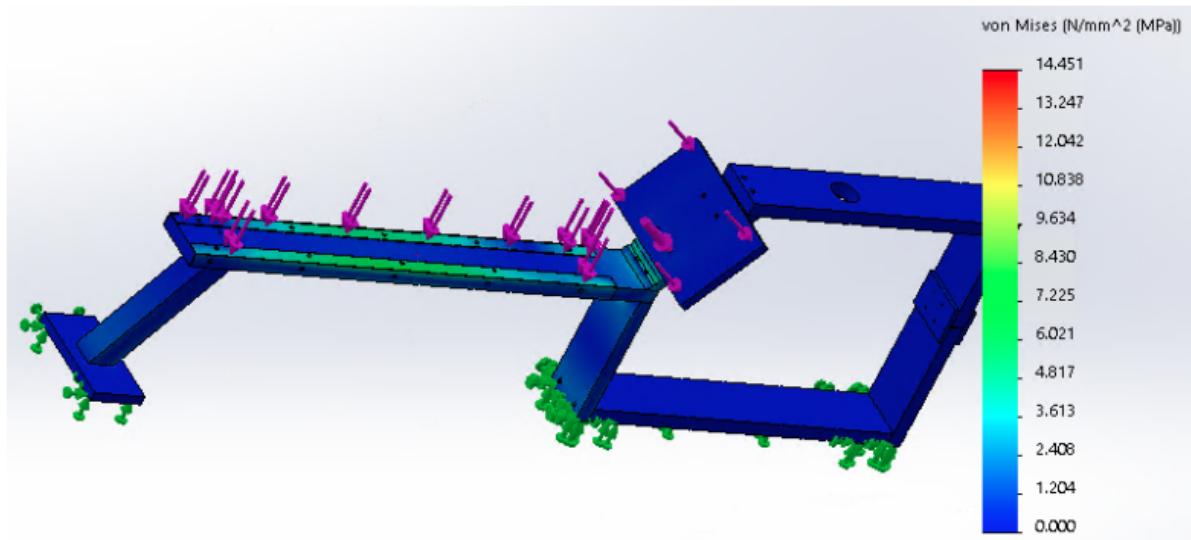
Una vez obtenidas las reacciones de la estructura se procedió al análisis de esfuerzos para la determinación de las áreas de secciones transversales mínimas para los tornillos y las tablas considerando un factor de seguridad de 1.5. Cabe mencionar que las partes más críticas de la estructura es la viga transversal (en donde va el carrito) que se encuentra a flexión y la viga vertical 6 que está a compresión.

Elemento	Área mínima a 14 kg ( $m^2$ )	Área mínima a 18 kg ( $m^2$ )
----------	-------------------------------	-------------------------------

Perno 1	$1.2 * 10^{-8}$	$1.2 * 10^{-8}$
Perno 2	$1.12 * 10^{-8}$	$1.12 * 10^{-8}$
Perno 3	$1.72 * 10^{-9}$	$2.22 * 10^{-9}$
Perno 4	$4.68 * 10^{-9}$	$4.72 * 10^{-9}$
Perno 5	$9.3 * 10^{-9}$	$9.44 * 10^{-9}$
Perno 6	$4.68 * 10^{-9}$	$4.72 * 10^{-9}$
Perno 7	$4.68 * 10^{-9}$	$4.72 * 10^{-9}$
Perno 8	$3.83 * 10^{-9}$	$3.87 * 10^{-9}$
Eslabón 1*	$4.2 * 10^{-8}$	$4.2 * 10^{-8}$
Eslabón 2*	$1 * 10^{-4}$	$1 * 10^{-4}$
Eslabón 3*	$1.43 * 10^{-8}$	$1.44 * 10^{-8}$
Eslabón 4*	$1.75 * 10^{-8}$	$1.76 * 10^{-8}$
Eslabón 5*	$1 * 10^{-4}$	$1 * 10^{-4}$
Eslabón 6*	$1.75 * 10^{-8}$	$1.76 * 10^{-8}$
Eslabón 7*	$8.2 * 10^{-9}$	$8.28 * 10^{-9}$

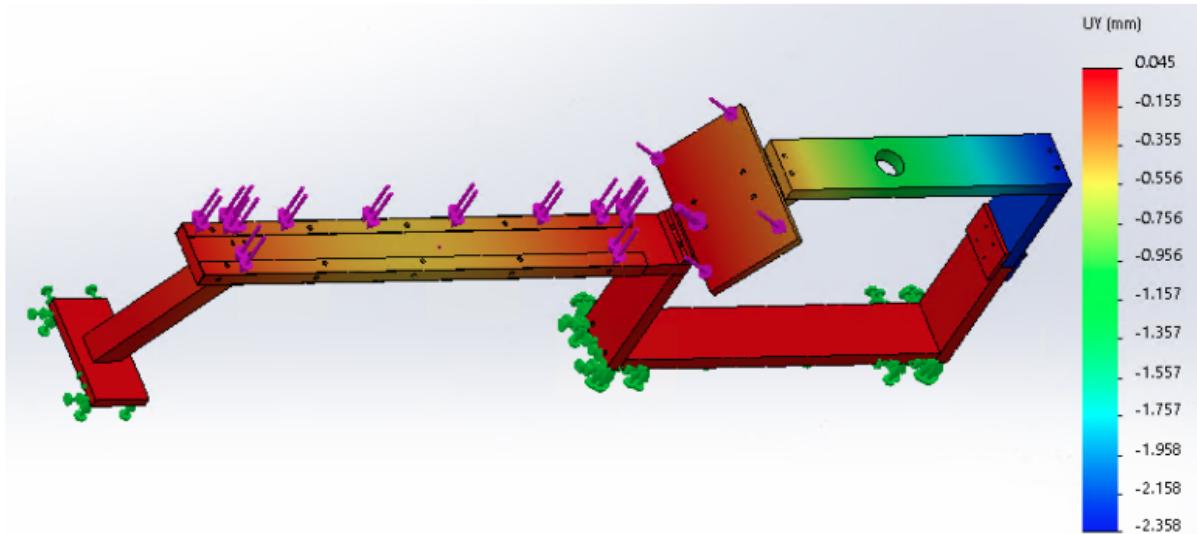
**Figura 3.** Áreas mínimas de elementos de la estructura.

## Stress



**Figura 4.** Análisis de esfuerzos de la remadora con elemento finito.

## Movimiento



**Figura 5.** Análisis de movimiento de la estructura de la remadora.

Con base al análisis anterior se seleccionó la madera de pino como material principal, ya que posee propiedades mecánicas aptas para nuestra aplicación como lo es su tenacidad y rigidez, además de que es más accesible y viable con respecto a su costo y manufactura a comparación de materiales como el acero.

Para los rieles se seleccionó aluminio 6061 ya que es el más popular en uso comercial y el más accesible. Con respecto a las piñas se seleccionó acero de medio carbón ya que es el más común y cumple con los requerimientos necesarios.

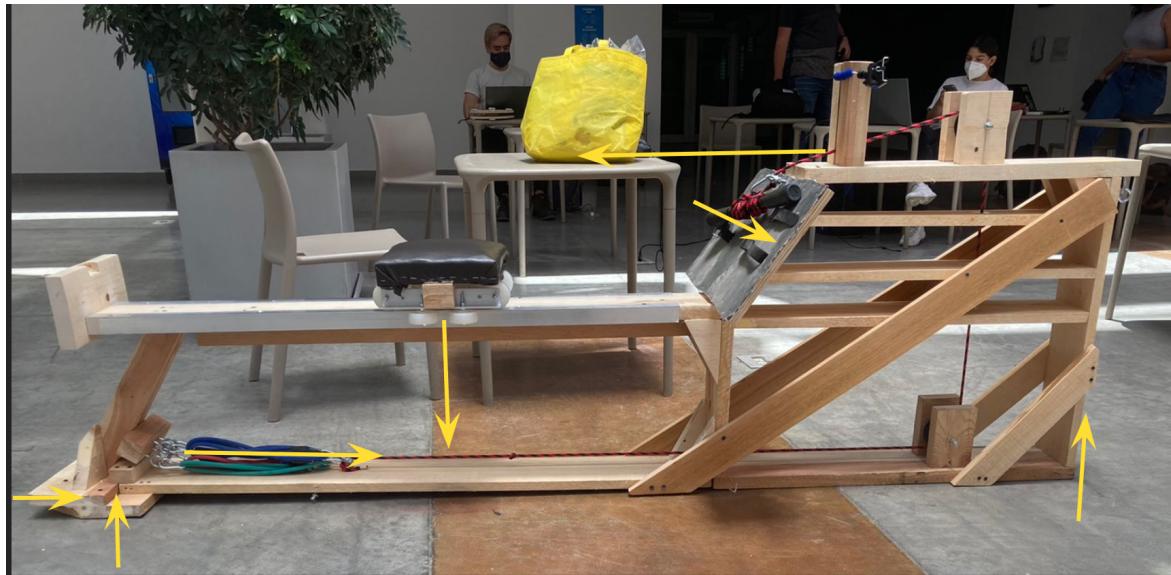
<b>Composition (summary)</b>				
Cellulose/Hemicellulose/Lignin/12%H2O				
<b>General properties</b>				
Density	(i)	440	-	600 kg/m <sup>3</sup>
Price	(i)	* 12	-	24 MXN/kg
Date first used	(i)	-10000		
<b>Mechanical properties</b>				
Young's modulus	(i)	8.4	-	10.3 GPa
Shear modulus	(i)	* 0.62	-	0.76 GPa
Bulk modulus	(i)	0.37	-	0.41 GPa
Poisson's ratio	(i)	* 0.35	-	0.4
Yield strength (elastic limit)	(i)	* 35	-	45 MPa
Tensile strength	(i)	* 60	-	100 MPa
Compressive strength	(i)	* 35	-	43 MPa
Elongation	(i)	* 1.99	-	2.43 % strain
Hardness - Vickers	(i)	* 3	-	4 HV
Fatigue strength at 10 <sup>7</sup> cycles	(i)	* 19	-	23 MPa
Fracture toughness	(i)	* 3.4	-	4.1 MPa.m <sup>0.5</sup>
Mechanical loss coefficient (tan delta)	(i)	* 0.007	-	0.01

Figura 6. Propiedades mecánicas de la madera de pino.

<b>General properties</b>				
Density	(i)	2.63e3	-	2.7e3 kg/m <sup>3</sup>
Price	(i)	* 45.6	-	49.3 MXN/kg
Date first used	(i)	1914		
<b>Mechanical properties</b>				
Young's modulus	(i)	69	-	74 GPa
Shear modulus	(i)	25	-	28 GPa
Bulk modulus	(i)	66	-	77 GPa
Poisson's ratio	(i)	0.32	-	0.34
Yield strength (elastic limit)	(i)	65.1	-	252 MPa
Tensile strength	(i)	151	-	323 MPa
Compressive strength	(i)	* 65.1	-	264 MPa
Elongation	(i)	4.3	-	23 % strain
Hardness - Vickers	(i)	37.8	-	98.3 HV
Fatigue strength at 10 <sup>7</sup> cycles	(i)	* 61.7	-	150 MPa
Fracture toughness	(i)	* 27	-	37 MPa.m <sup>0.5</sup>
Mechanical loss coefficient (tan delta)	(i)	0.0011		

Figura 7. Propiedades mecánicas del aluminio 6061.

## *Funcionamiento*



**Figura 8.** DCL del prototipo final

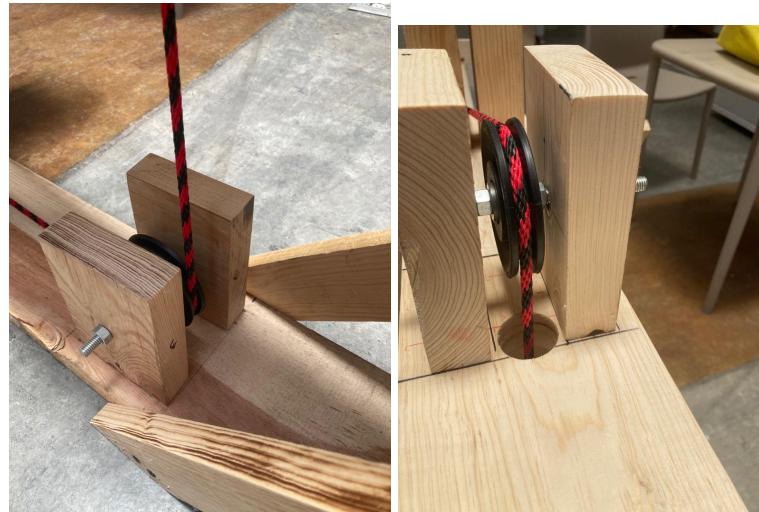
El funcionamiento (basado en la imagen anterior) consiste en que el usuario se sentará en el banco y se desplazará ejerciendo una fuerza sobre el reposapiés y jalando el manillar hacia su pecho, esto provocará que las ligas colocadas en la parte baja se tensionen generando una fuerza en sentido contrario al que la cual el usuario jala el manillar, dicha fuerza es la resistencia del mecanismo. El desplazamiento del usuario dependerá de su capacidad de resistencia, una vez que este ya no puede superar la fuerza de resistencia de las ligas o simplemente ya extendió al máximo sus piernas regresará a su posición inicial con las piernas flexionadas. Esta dinámica de fuerzas genera reacciones en los soportes que balancean la estructura.

## *Mecanismos de transmisión de fuerzas utilizados*

Se utilizaron 2 poleas de  $\frac{3}{8}$  de pulgada con el objetivo de permitir la transmisión de fuerza desde el manillar hasta las ligas y viceversa, permitiendo la correcta operación de la resistencia, además de que las poleas permiten direccionar la soga de tal manera que se pueda deslizar en ambos sentidos (cuando el usuario se impulsa y cuando el usuario regresa).



**Figura 9.** Polea de  $\frac{3}{8}$  de pulgada y eje.



**Figura 10.** Poleas de  $\frac{3}{8}$  de pulgada utilizadas en la remadora.

#### *Elemento de resistencias*

Originalmente se tenía planeado implementar una resistencia de agua utilizando un eje unido a una hélice, sin embargo por cuestiones de tiempo y adaptabilidad de la estructura se utilizaron ligas de gimnasio acopladas a la estructura por medio de argollas y ganchos, estas ligas se unen con la soga con seguros tipo alemán, ya la soga se desliza gracias a la implementación de poleas.

Las resistencias de las ligas son:

Liga	Resistencia
Amarilla	10 lbs
Roja	20 lbs
Verde	30 lbs
Azul	40 lbs
Negra	50 lbs

La implementación de este sistema de resistencia otorga la ventaja de una mejor facilidad de configuración, ya que si se quiere incrementar o disminuir simplemente se conectan o desconectan las ligas por medio de los ganchos, a diferencia de la resistencia de agua en la cual se tiene que cambiar el volumen de la tina o cambiar la hélice.

## Resumen de manufactura

- **Técnicas de manufactura:** De las técnicas de manufactura utilizadas en este proyecto fue el cepillado para darle un mejor acabado a lo que es la madera, también se hizo uso de la técnica de manufactura de taladrado para darle una guía a las piñas y así no abrir la madera al momento de ponerlas, se hizo uso de el aserrado para darle una longitud y un corte definido a la madera, el abrasivo para poder tener bien lijada cualquier superficie que haya tenido un corte, y por último se hizo uso de la técnica de desbaste la cual nos ayudó para quitar el excedente de madera que no necesitábamos.
- **Resumen de proceso de fabricación:**

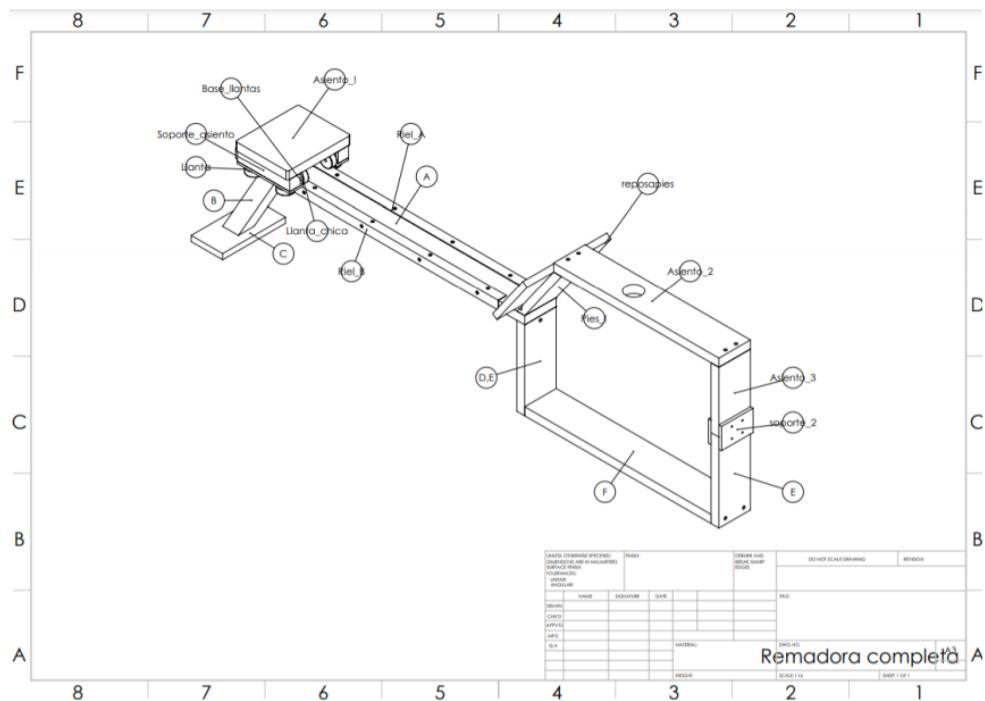


Figura 11. Planos de la remadora



Figura 12. Modelo anterior de la remadora

Como se puede observar, la imagen anterior es el diseño del plano en CAD de la remadora y posteriormente sale la remadora terminada según el CAD, pero al ponerla a prueba nos percatamos que existían varios inconvenientes por lo que se le hicieron varias modificaciones una de las modificaciones fue que le colocamos 3 travesaños más para reforzar el soporte de los pies ya que este al momento de realizar el ejercicio tenía un movimiento y con estos travesaños lo que hicimos fue darle un mejor apoyo al descansa pies para que al momento que hicieran el ejercicio y se impulsaran con una mayor confianza, también se le añadieron 4 barras en diagonal estas para evitar que al momento de jalar las cuerdas se levantará la parte frontal, ya que al hacer el ejercicio se generaba una inestabilidad en la remadora, pero al momento de poner estos refuerzos diagonales se corrigió un poco esto y también se le colocó una tabla de la parte trasera hasta la parte delantera para reforzar la estructura, esta tabla también tenía la función de evitar que el soporte trasero tuviera un movimiento al momento de hacer el ejercicio, y por último se elevo un poco la parte trasera de la remadora, esto fue para que poder evitar que la remadora no se levantara y funcionó, y con todos estos cambios no se asemeja la remadora actual al modelo inicial, se adjunta imagen del nuevo modelo de la remadora.



**Figura 13.** Modelo actual de la remadora

## Sistema mecatrónico

### Sensores utilizados y su integración

Para el desarrollo de la remadora se utilizaron los siguientes sensores:

- **Sensor ultrasónico:** Como su nombre lo indica, los sensores ultrasónicos miden la distancia mediante el uso de ondas ultrasónicas. Los sensores ultrasónicos miden la distancia al objeto contando el tiempo entre la emisión y la recepción.  
Específicamente se usó el sensor HC-SR04. Este está conformado por 4 pines:
  - GND: Tierra
  - VCC: Voltaje. Para su uso es necesario alimentarlo con 5 v.
  - Trigger: recibe un pulso de habilitación de parte del microcontrolador, mediante el cual se le indica al módulo que comience a realizar la medición de distancia.
  - Echo: el sensor devuelve al microcontrolador un pulso cuyo ancho es proporcional al tiempo que tarda el sonido en viajar del transductor al obstáculo y luego de vuelta al módulo.

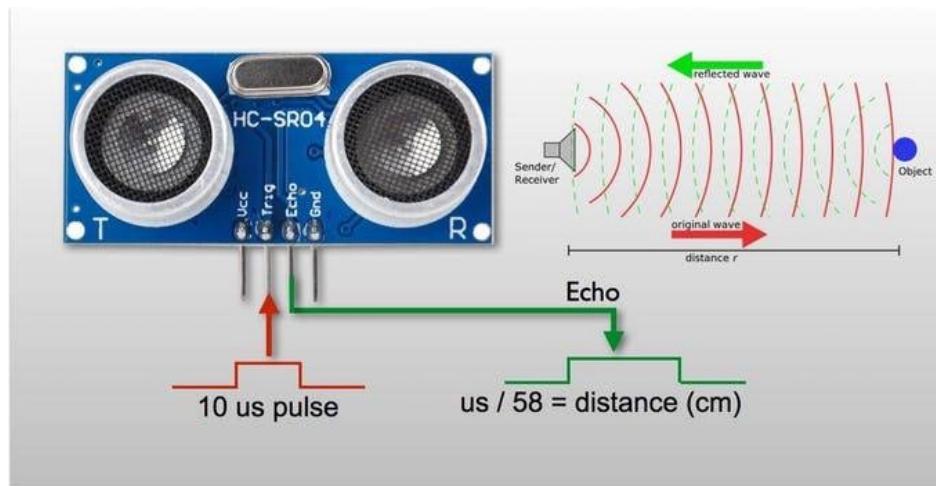


Figura 14. Sensor ultrasónico y diagrama del funcionamiento

- **Módulo Bluetooth HC-06:** A diferencia del HC-05, este solo trabaja de forma esclavo. En optimas condiciones tiene una señal de 10 metros. su voltaje de operación es de 3.6 VDC a 6 VDC, por lo que perfectamente se puede conectar al microcontrolador. Para este proyecto se utilizaron los pines:
  - Vcc: Voltaje de entrada
  - GND: Tierra
  - TXD: Transmite señal
  - RXD: Recibe señal

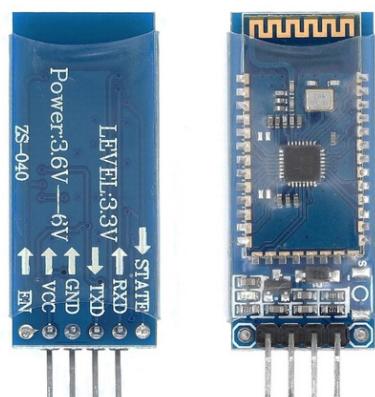
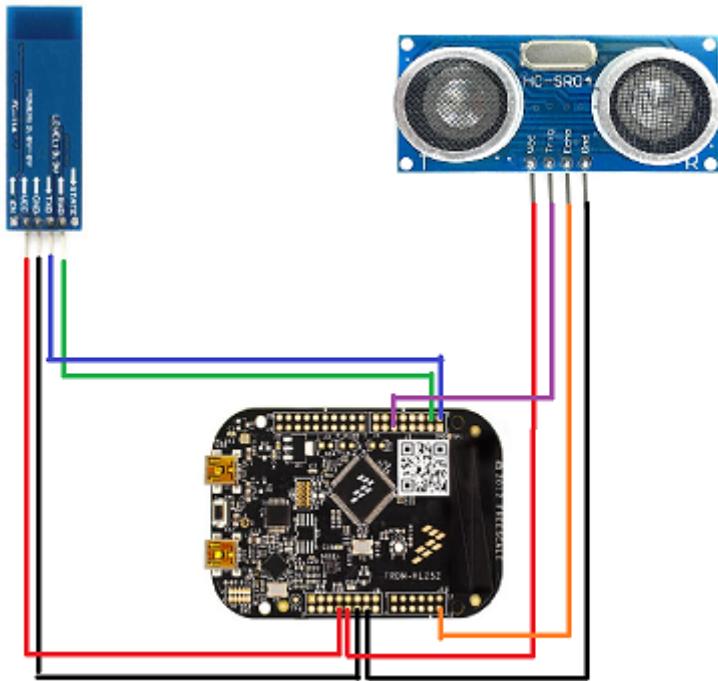


Figura 15. Módulo Bluetooth HC-06

*Diagrama del sistema:*



**Figura 16.** Diagrama general del sistema

Ambos sensores se alimentan desde el microcontrolador FRDM-KL25Z. En el caso del módulo de Bluetooth, se comunica usando el protocolo serial, su UART se transmite en la línea “TX” y recibe datos por la línea “RX”, por lo que se conecta a los pines PTA1 Y PTA2. Por otra parte el sensor ultrasónico está conectado, el TRIG en el pin PTC6 y el ECHO en el PTC1.

#### *Sistema de potencia:*

Para la alimentación del microcontrolador se decidió usar un Powerbank. Es una batería recargable de material de litio (ion o polímero), con puertos Universal Serial Bus. Para este proyecto se utilizó un Powerbank de 18000 mAh el cual tiene una salida 5v/1A/2A. Al solo usar dos sensores, no se consideró el uso de otra fuente, a parte de los pines del microcontrolador para su alimentación.

#### **Prototipo**

##### *Demostración de funcionamiento*

Para iniciar a utilizar la remadora debe colocarse el dispositivo móvil en el trípode instalado en la remadora y realizar los pasos que se mencionan más adelante para establecer la conexión Bluetooth de la remadora e inicializarla, después de esto puede comenzar a utilizarse. Para hacer un uso correcto de una remadora el usuario debe sentarse en el asiento y ajustar sus pies asegurandolos con los broches de velcro, una vez ajustado y encontrándose sentado con la espalda recta y las rodillas contraídas se debe tomar el manubrio con ambas

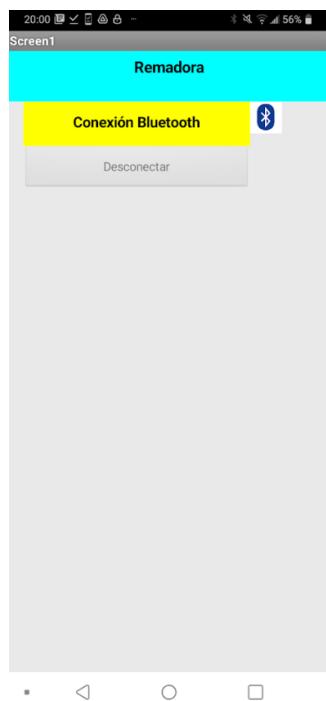
manos y tirar hacia atrás utilizando los músculos de los hombros, biceps y la parte superior de la espalda, al hacer esto el usuario terminará con las piernas extendidas y los brazos contraídos hacia el pecho, después de esto se deben doblar las rodillas lentamente y repetir el movimiento el número de veces que se desee. Es importante mencionar que para optimizar su desempeño el movimiento inicial donde se jala debe ser de alto impacto y el de regreso un poco más lento, también que debe cuidarse la postura de la espalda para evitar lesiones. Al terminar de utilizarla se finaliza la interfaz de usuario, se desabrochan los seguros de los pies y se coloca el manubrio en su posición inicial con cuidado.



**Figuras 17 y 18.** Demostración del funcionamiento

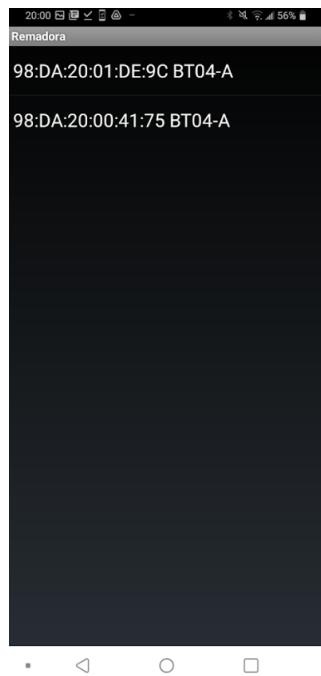
### *Interfaces de usuario*

Se decidió utilizar MIT App Inventor para crear una aplicación móvil que sirviera como interfaz de usuario para visualizar el tiempo transcurrido, la distancia recorrida por el usuario y las remadas realizadas por el usuario. Esta aplicación recibe la información proporcionada por el microcontrolador vía Bluetooth por lo que en la primera pantalla (figura 19) podemos ver la interfaz con la opción de hacer una conexión Bluetooth.



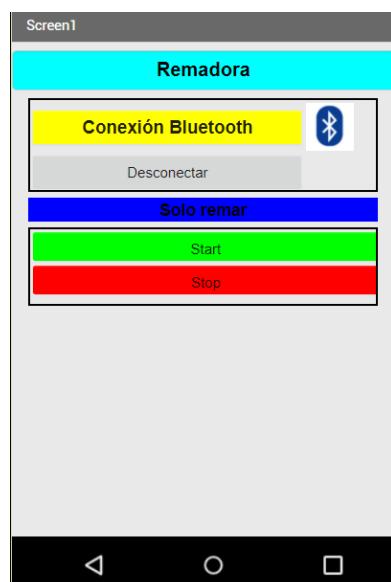
**Figura 19.** Pantalla inicial de interfaz de usuario

En la siguiente pantalla (figura 20) se observan las opciones de conexión que deben ser seleccionadas para comenzar a utilizar el dispositivo.



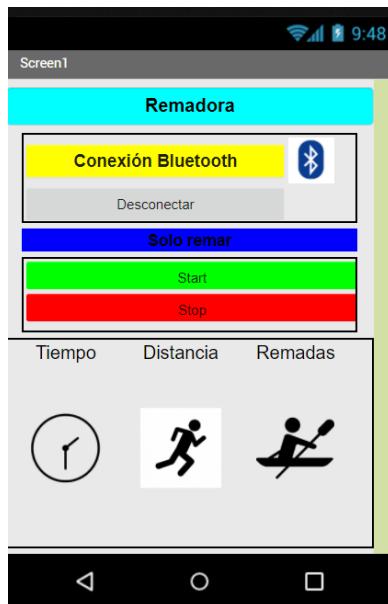
**Figura 20.** Conexión del dispositivo

En la siguiente pantalla (figura 21) se observa, una vez realizada la conexión Bluetooth, 2 botones, uno de inicio (start) y uno de fin (stop), el botón de inicio debe ser seleccionado para comenzar a remar, en esta interfaz el botón de fin se encuentra deshabilitado ya que no hay un ciclo de entrenamiento inicializado.



**Figura 21.** Menú de inicio

En la última interfaz se visualizan los valores de tiempo transcurrido, distancia recorrida y remadas realizadas.



**Figura 22.** Interfaz visual de los datos del usuario

La información enviada por el microcontrolador será desplegada debajo de Tiempo, Distancia y Remadas respectivamente, para finalizar su uso se presiona el botón de Stop y la interfaz regresa la forma que se encontraba en la figura 21, para desconectar el dispositivo se presiona el botón de desconectar, al hacer esto se pierde la conexión Bluetooth y la app regresa a la pantalla inicial.

#### *Variables censadas y mediciones obtenidas*

La principal variable sensada fue la distancia recorrida por parte del usuario, el algoritmo implementado en C que se le cargó al microcontrolador, utiliza:

- TPM: Captura los impulsos dados por el sensor ultrasónico cada determinado tiempo, además de que en su interrupción se transforma el dato recibido a cm (siendo la distancia entre el banco y el sensor), dicho dato en conjunto con condicionales se interpreta para saber en qué sentido se desplaza el usuario, detectar los cambios de sentido y conforme a ello calcular la distancia recorrida hasta ese punto , este dato se almacena en un arreglo que permite transformar el dato de decimal a ASCII.

- LPTM: Se utiliza para controlar la frecuencia con la que el TPM captura los pulsos arrojados por el sensor ultrasónico y la velocidad con la que se transmiten los datos a través del UART0, cabe mencionar que esta en LPO y en bypass, por lo que se cuenta en ms, la frecuencia se puede ajustar cambiando la referencia CMR.
- UART0: Permite la transmisión del arreglo que tiene el dato de la distancia remada al celular por medio del módulo de bluetooth, cabe mencionar que se transmitirá el dato cada vez que se habilite la interrupción del UART0 es decir cada vez que el registro THR este vacío (bandera THRE encendida). Para el modulo de bluetooth se usó el pin PTA1 y PTA2.
- NVIC: Se utiliza para habilitar la ejecución de todas las interrupciones de los módulos anteriores.

Cabe mencionar también que se habilitaron los relojes de cada módulo y se configuraron los pines correspondientes para su operación (véase anexo de código).

#### ***Viabilidad del prototipo***

- **Análisis de costo y tiempo de fabricación:** Del costo que se tenía contemplado para la remadora fue entre 3,500.00 y 4,500.00 pesos y pues al hacer cuentas se hizo un gasto total de 3,575.00 pesos en lo que fue todo el material empleado, dicho material consta de madera, ruedas, perfiles, pijas, seguros y armellas, ligas, punta cruz, cuerda, poleas, tornillo y tuercas, soporte de teléfono, esponja, vinil, y brocas.

Se anexa tabla de gastos:

<b><i>Material</i></b>	<b><i>Precio (MXN)</i></b>
Maderas	\$800.00
Ruedas	\$550.00
Perfiles	\$230.00
Pijas	\$300.00
Seguros y armellas	\$600.00
Ligas	\$230.00
Punta cruz	\$15.00
Cuerda	\$50.00
Poleas	\$480.00
Tornillo y tuercas	\$100.00

Soporte de teléfono	\$50.00
Esponja	\$40.00
Vinil	\$30.00
Brocas	\$100.00
<b>COSTO TOTAL:</b>	<b>\$3,575.00</b>

- **Tecnologías de manufacturas aplicables en producción en masa:**
- Vigas Diagonales: Las vigas diagonales tienen una estructura base muy sencilla, por lo que se utilizará únicamente una sierra inglete para los cortes de 45° y una sierra circular de mesa para los cortes longitudinales.
- Ángulos de aluminio: Para los ángulos de aluminio y del carrito se haría uso de una fresadora CNC, y para realizar los orificios para las piñas también se haría uso de una fresadora CNC.

### *Conclusiones*

*Oscar :*

En este bloque se concluyó el desarrollo de la remadora, agregando la parte del mecanismo y el sistema sensorial, además se nos dió la introducción al desarrollo de sistemas embebidos con la programación de microcontroladores en C y la implementación de sensores en sistemas. La verdad este bloque estuvo lleno de retos y situaciones adversas que nos exigieron idear diversas soluciones; iniciando con el mecanismo, la implementación de una tina e helice con eje resultaba un tanto compleja, debido a los materiales requeridos, la manera de conseguirlos y el posicionamiento de sensores asociados al mecanismo, pero principalmente por la desventaja de que este tipo de mecanismo ofrecía muy poca personalización de resistencia al usuario, con esto dicho se ideó el uso de bandas o ligas de gimnasio de diversas resistencias enganchadas a la estructura de la remadora en conjunto con poleas para direccionar dicha fuerza de resistencia hasta el manillar, ofreciendo así un mecanismo más sencillo de instalar, menos costoso y más adaptable a las necesidades del usuario; otro inconveniente que se presentó fue el hecho de que una vez instalado el mecanismo de resistencia al momento de jalar el manillar y desplazarse con el carro , la parte de enfrente de la estructura sufría un momento agresivo que hacía que se levantara , esto en

parte a que no se tenía un contrapesa en dicho extremo (ya que no se contaba con una tina de agua) que balanceara el momento generado por la fuerza del manillar y el impulso del usuario, aunque si se le colocaba un contrapeso en la parte de enfrente el soporte trasero se contraía por la fuerza de tensión de las ligas, estos dos problemas estructurales fueron un dolor de cabeza por 2 semanas, pero ante ello se reforzó la estructura con tablones de madera transversales que unen el soporte trasero con el resto de la estructura y la viga diagonal del reposapiés con la viga vertical, además de utilizar soportes diagonales, con estas modificaciones el momento se redujo a casi nulo; otro inconveniente fue la configuración del programa y el sensor, se seleccionó el sensor ultrasónico para calcular la distancia recorrida y la cantidad de remadas, sin embargo el sensor ultrasónico mide distancias en un solo sentido por lo que al estarse acercando al sensor estaría midiendo distancias cada vez menores y esto si no se interpreta bien puede afectar los cálculos relacionados, en el programa se realizó un algoritmo con condicionales dentro de la interrupción del TPM, estas condicionales permiten con el dato dado por el sensor ya convertido en cm (distancia del banco con respecto al sensor) interpretar el sentido en el que se desplaza el banco, almacenar los cambios de sentido en una variable y con base a ello calcular la distancia recorrida por el usuario almacenando el dato en una variable que se está sumando constantemente y que se transmite al celular en cada interrupción del UART0, sin embargo el programa cuenta con un error de lógica que lamentablemente no se pudo detectar. Al final el prototipo entregado estaba en un 85% completado, faltando ajustes en la parte de la aplicación y el programa del microcontrolador. El bloque en general me gustó ya que fue mi primera aproximación a lo que es el desarrollo de sistemas mecatrónicos aplicados a un producto específico , aunque fue un reto estresante y pesado, debido a las situaciones comentadas anteriormente y la falta de experiencia en programación en C y desarrollo de aplicaciones, pero a pesar de ello me voy con un sentido de adaptabilidad mayor ya que se trabajó con un proyecto el cual yo no inicié sin embargo se analizó, se comprendió y se mejoró entregando un producto final, esto en el campo profesional y laboral me servirá para adaptarme ante cualquier situación y entregar lo solicitado , con esto puedo concluir que este proyecto con todo y sus adversidades atribuye a mi desarrollo profesional y personal.

*Francisco:*

Este periodo fue la presentación final y conclusión de la remadora, y en verdad que fue un reto, ya que se tuvo que reforzar el mecanismo como se menciona en el documento, la

remadora original tenía varios detalles, pero a lo largo del curso se le fueron arreglando esos pequeños detalles, que fue ya el conjunto de todos los bloques, pero lo que más se aplicó de este bloque para la remadora fue la parte de microcontroladores, ver lo de los sensores que se iban a utilizar para que nos pudieran ayudar a medir distancias, y también se vio parte de mecanismos. En sí en este bloque existieron muchas complicaciones, estas complicaciones eran más enfocadas a los tiempos ya que en ocasiones no podíamos trabajar más tiempo en la remadora, y también que se le tenían que hacer varios cambios a la remadora, otro tipo de complicación fue el constante cambio de equipos, que en ese punto nos atrasaba mucho debido a que se tenía que continuar el trabajo de un equipo anterior y a veces le hacía falta hacerle más cosas, otra de las complicaciones como se menciona en el documento fue por parte de la remadora, ya que al momento de realizar el ejercicio existía la complicación de que se levantaba y se movía el soporte trasero, pero esto se pudo solucionar colocando más soportes, y una tabla que bloqueara el movimiento del soporte trasero, pero la mayor complicación fue que no se pudo completar con éxito el código pero sí nos mostraba. Este bloque lo sentí más pesado de lo normal, pero en general me gustó porque fue el primer reto que hacemos después de mucho tiempo y me gustó, lo que sentí me sentí perdido fue en la parte del código ya que lo poco que había visto de programación en C++ en la preparatoria pues no me acordaba de ello y siento que íbamos muy rápido y en ocasiones no entendía al cien la clase y tenía que repasarla. Pero en sí me gustó mucho este proyecto ya que pudimos ver cosas más a fondo de la carrera y pudimos trabajar en algo físico, aunque eso de lo de los cambios de equipo no me gusto pues es cierto lo que nos dijeron que nos servirá para en el campo laboral si llegamos a un proyecto ya avanzado adaptarse y continuar con el proyecto.

*Belen:* Este bloque supuso un gran reto debido a que concluimos de armar las remadoras. En mi caso fue un reto el estar adaptándose a distintos equipos a lo largo del semestre. En este bloque me tocó trabajar con una remadora distinta a la que había estado trabajando. A pesar de esto, pienso que pude lograr el aprendizaje esperado. Por parte del mecanismo surgió la idea de cambiar el concepto pensado al ver la remadora vista por los maestros. Por otra parte, el uso del microcontrolador con los sensores fue una parte importante para el desarrollo del proyecto. Aunque no se logró terminar con éxito, pudimos apreciar mejor la lógica de funcionamiento de los diferentes pines y de la programación del microcontrolador. Esta fue la parte más difícil ya que no teníamos experiencia con su uso, esperamos en un futuro tener mejores resultados con su uso.

*Ximena:* Considero que este bloque fue bastante retador en distintos aspectos, principalmente porque llegamos a la culminación del reto de la remadora, durante el semestre atravesamos distintas etapas relacionadas con su desarrollo, desde su diseño, manufactura hasta la implementación del mecanismo, el uso del microcontrolador la electrónica lo relacionado al desarrollo y administración de proyectos. Inicialmente, comenzamos el trabajo de este bloque arreglando y optimizando detalles que quedaron pendientes en la etapa anterior, esto nos tomó un poco de tiempo adicional al que teníamos contemplado dedicar al trabajo en esta etapa y, por lo tanto, nos retrasó un poco. En cuanto al mecanismo, desde etapas anteriores se tenía contemplado que la remadora fuera de aire y no de agua, ya que se diseñó bajo este concepto, es por esto que optamos por uso de ligas de resistencia que fueron el principal elemento de resistencia y poleas utilizadas para direccionar la fuerza. En cuanto a la parte electrónica optamos por utilizar un sensor ultrasónico para sensar la distancia recorrida por el usuario y así poder calcular las medidas deseadas y asimismo enviarlas a la interfaz visual, esta parte supuso un gran reto para nosotros debido a nuestra poca experiencia con la programación de microcontroladores en C. En general, considero que mi experiencia de aprendizaje fue buena ya que pudimos aplicar los conceptos y competencias aprendidas en durante los módulos a la solución de nuestro reto, en mi caso, estuve trabajando en esta remadora desde el primer bloque, con distintos equipos y bastantes situaciones inesperadas y complicaciones, a pesar de todo eso considero que el resultado fue satisfactorio para mí ya que entregamos una remadora funcional, apegada al diseño y bastante innovadora, sin embargo, espero en un futuro poder obtener más conocimientos que me permitan mejorar los actuales y trabajar en las áreas de oportunidad que surgieron en este reto.

### ***Anexo de código:***

\* main implementation: use this 'C' sample to create your own application

\*

\*/

```

#include "derivative.h" /* include peripheral declarations */

unsigned char mensaje[]={"      \r\n"};

unsigned char i=0;

unsigned short t_rising;

unsigned char rising_edge=0;

unsigned short d_cm;

unsigned short distancia_actual;

unsigned short distancia_previa = 300;

unsigned short med = 0;

unsigned short cs = 0;

unsigned short distancia_ma;

unsigned short distancia_re;

unsigned short distancia_rv;

unsigned short distanciatotal_p;

unsigned short distanciatotal_n;

void delay_us (unsigned short t_us)

{

    PIT_LDVAL0=4*t_us;          //el factor de 4, es porque el reloj está 4 MHz (250 ns)

    PIT_TCTRL0=1;                //hab timer

    do {}while (PIT_TFLG0==0); //mientras que no llegue a 0 el contador

    PIT_TFLG0=1;                //apagar la bandera

    PIT_TCTRL0=0;                //deshabilita conteo del PIT0

}

```

```

void LPTimer_IRQHandler()

{
    LPTMR0_CSR=(1<<7)+(1<<6)+1;

    GPIOC_PDOR= (1<<6);      //PORTC_6 es trig del sensor ultrasonico

    delay_us(12);

    GPIOC_PDOR= 0;

}

void UART0_IRQHandler()

{
    UART0_D=mensaje[i++];//Se transmite distancia recorrida en ASCII al celular

    if (mensaje[i]==0)

    {
        i=0;
        NVIC_ICER=(1<<12);

    }

}

void FTM0_IRQHandler()

{
    unsigned short t_us;

    if ((TPM0_C0SC & (1<<7))!=0)

    {
        TPM0_C0SC=(1<<6)+(1<<7); //Apagar bandera y mantener configuración

        if (rising_edge==0)

        {

```

```

t_rising=TPM0_C0V;

rising_edge=1;

TPM0_C0SC=(2<<2)+(1<<6); //input capture/falling edge, hab intr

}

else

{

t_us=TPM0_C0V-t_rising; //TPM0_C0V tiene el tiempo cuando detectó falling edge

d_cm=t_us/59;

distanzia_actual = d_cm; //Se almacena el dato dado por el sensor en cm que es la distancia actual de esa iteración

if (distanzia_actual < distanza_previa) //sentido negativo (carrito regresa y se acerca al sensor)

{

if (med == 0) //Primer dato del sentido negativo

{

distanzia_ma = distanza_actual;//se define la distancia mayor = la que el usuario inició a
remar o una nueva remada

med = 1;//se cambia el valor de la bandera de medición

}

else

{



if (cs == 0) // No ha habido un cambio de sentido, por lo que no se ha completado una remada completa

{



distanzia_re = distanza_ma - distanza_actual;//Distancia recorrida

}

else //Ya ha habido un cambio de sentido, por lo que ya se llevan varias remadas

{



distanzia_rv = distanciatotal_p;//Distancia total recorrida en el sentido anterior

distanzia_re = distanca_rv + (distanzia_ma - distanza_actual);//Distancia recorrida

```

```

    }

    distancia_previa = distancia_actual;//Se define la distancia previa

    distanciatotal_n = distancia_re;//Distancia total de dicho sentido

}

else if (distancia_actual > distancia_previa)//Sentido positivo (carrito avanza)

{

    if (med == 1) //Primer dato del sentido positivo

    {

        cs = cs + 1;//Cambio de sentido

        med = 0;//Se cambia el valor de la bandera de medición

    }

    else

    {

        distancia_rv = distanciatotal_n;//Distancia total recorrida en el sentido anterior

        distancia_re = distancia_rv + distancia_actual;//Distancia recorrida

        distancia_previa = distancia_actual;//Se define distancia previa

        distanciatotal_p = distancia_re;//Distancia total de dicho sentido

    }

    else

    {

        mensaje[3]=distancia_re%10+0x30; //aislo las unidades y le sumo 0x30 por la tabla ASCII

        distancia_re=distancia_re/10;

        mensaje[2]=distancia_re%10+0x30; //aislar las decenas

        distancia_re=distancia_re/10;

        mensaje[1]=distancia_re%10+0x30; //aislar las centenas
    }
}

```

```

distanzia_re=distanzia_re/10;

mensaje[0]=distanzia_re%10+0x30;

NVIC_ISER=(1<<12);

rising_edge=0;

TPM0_C0SC=(1<<2)+(1<<6); //input capture/rising edge, hab intr

}

}

void clk_init (void)

{

// FIRC = 4 MHz. BusClk = 4 MHz

// UART0: FIRC. UART1: BusClk. UART2: BusClk. TPM: FIRC. IIC: BusClk


MCG_C1|=(1<<6) + (1<<1); //MCGOUTCLK : IRCLK. CG: Clock gate, MCGIRCLK enable pag 116

MCG_C2|=1; //Mux IRCLK : FIRC (4 MHz) pag 116

MCG_SC=0; //Preescaler FIRC 1:1 pag 116


SIM_CLKDIV1=0; //OUTDIV4=OUTDIV1= 1:1 pag 116. Busclk 4 MHz

SIM_SOPT2|=15<<24; //Seleccion MCGIRCLK tanto para UART0 como para TPM

}

```

```

int main(void)

{
    clk_init();

    //TPM0_C0 PORTC_1 ALT4

    SIM_SCGC5|=(1<<11);      //clk interfaz PORTC

    PORTC_PCR1=(4<<8);      // TPM0_C0

    SIM_SCGC6|=(1<<24);      //clk interfaz TPM0

    TPM0_SC=(1<<3)+2;        //clk interno (4 MHz), preescalador 4 (4MHz/4 = 1 MHz (1 us))

    TPM0_C0SC=(1<<2)+(1<<6); //input capture/rising edge, hab intr

    NVIC_ISER=(1<<17);       //Hab intr TPM0

    //LPTMR0

    SIM_SCGC5|=1;             //clk interfaz LPTMR0

    LPTMR0_PSR=5;              //Se selecciona LPO y bypass (ms)

    LPTMR0_CMR=250;            //Se configura la referencia a 250ms

    LPTMR0_CSR=(1<<6)+1;     //Se habilita la intr de LPTMR0 y el contador CNR

    //PIT

    SIM_SCGC6|=(1<<23)+(1<<24); //PIT

    PIT_MCR=0;                 //Hab PITs

    PORTC_PCR6=(1<<8);        //GPIO

    GPIOC_PDOR=0;               //Valor inicial = 0

```

```
GPIOC_PDDR=(1<<6);      //Output

//UART0

SIM_SCGC5|=(1<<9);      //reloj interfaz PORTA

PORTA_PCR1=(2<<8);      //funcion alternativa 2: UART0_RX

PORTA_PCR2=(2<<8);      //funcion alternativa 2: UART0_TX


SIM_SCGC4|=(1<<10);     //reloj interfaz UART0

UART0_BDH=0;

UART0_BDL=26;              //ecuacion de SBR para generar baud rate (9600 bps)

UART0_C2 = (1<<2) + (1<<3) + (1<<5) +(1<<7); //Se habilita la rx, tx y sus interrupciones

NVIC_ISER=(1<<28); //Se habilita interrupcion de uart0 en NVIC

for(;;) {

}

return 0;
```