



**Instituto Tecnológico de Estudios Superiores de Monterrey**

**TC1032**

**Modelación de sistemas mínimos y arquitecturas computacionales**

Soluciona un problema de Automotriz

Etapa 3.

Gpo 2

Integrantes:

Alejandro Armenta Arellano	A01734879
Jesús Jiménez Aguilar	A01735227
Daniela Berenice Hernandez de Vicente	A01735346

Profesor:

Dr. David Antonio-Torres

20 de Octubre de 2021

La muerte es inevitablemente una situación a la que nos enfrentamos diariamente de diversas maneras, ya sea por alguna de las enfermedades a las cuales somos propensos de contraer, por algún accidente e inclusive por factores de la situación política del lugar de residencia.

De acuerdo con la OMS, anualmente mueren alrededor de 1,3 millones de personas a consecuencia de accidentes de tránsito. Sumado a esto, entre 20 y 50 millones de personas sufren traumatismos causantes de discapacidad debido a lesiones causadas por el tránsito (OMS, 2021). Estos datos son alarmantes, ya que dejando de lado las cifras de muertes por enfermedades degenerativas y por COVID-19, los decesos por accidentes automovilísticos son altamente comunes, principalmente en México. El Instituto Nacional de Salud Pública informa que nuestro país desafortunadamente se encuentra en el séptimo lugar a nivel mundial y el tercero en la región de Latinoamérica en muertes a causa de siniestros viales, ya que a diario ocurren 22 fallecimientos de jóvenes de entre 15 y 29 años, y 24 mil decesos en promedio al año por esta causa, siendo así la primera causa de muerte en jóvenes entre 5 y 29 años de edad y la quinta entre la población general (INSP, 2020).

Investigando a profundidad esta causa de muertes, de acuerdo con la NHTSA y el INEGI en México el panorama demuestra que en promedio más del 90% de los accidentes viales, la causa fue un error humano (INEGI, 2015). De ese porcentaje se calcula que aproximadamente el 40% es por una mínima distracción del conductor, ya sea por buscar algo, mirar algo por la ventana, utilizar el celular se crean faltas de atención del conductor a su entorno, etc (Carrillo, 2021). Aunque para la industria automotriz no es posible lograr prevenir cualquier distracción, mantiene la obligación de mantener a los usuarios protegidos inclusive de sí mismos.

Para poder determinar las propuestas que la industria automotriz ofrecería para lograr evitar que las distracciones al volante sigan siendo una fuente de afectación, es muy importante conocer primero cuales son las variables involucradas en dichas distracciones y que estas puedan ser mensurables para que con el uso de sensores se logre prevenir estas incidencias ayudando así al usuario. Algunas variables involucradas en esta problemática se encuentran orientadas a la distancia que se conduce de otros usuarios con relación al control de la velocidad (incluyendo la aceleración y desaceleración del vehículo).

Por lo tanto, será necesario implementar variables como un sensor de distancia y otro sensor de velocidad; con la finalidad de detectar y alertar al software del automóvil para que este notifique de manera audiovisual con timbres y señales de advertencia en el panel de GPS al usuario en alguno de los siguientes casos:

- Está muy cerca de otro vehículo por lo que debe reducir la velocidad y aumentar la distancia entre él y el otro vehículo para evitar un choque.
- Va a alta velocidad por lo que debe reducirla para evitar algún accidente.

El sonido del timbre será molesto auditivamente para que el usuario siga las instrucciones al pie de la letra por su seguridad.

Una vez que ya tenemos planteada la solución para este reto es esencial investigar acerca de las herramientas que serán utilizadas para así conocer los microcontroladores que se usarán

pues estos son la pieza clave en el sistema, debido a que son circuitos programables que guardan en su memoria las instrucciones a seguir en forma de código.

En primer lugar, el microprocesador PIC18F4550 es uno de los recientes circuitos integrados programables más utilizados dentro del mercado, el cual cuenta con diversas características de alto rendimiento que lo hacen capaz de seguir instrucciones que se encuentren dentro de códigos con alta complejidad para después almacenar los datos obtenidos dentro de la memoria de este e integrarlos ya sea en un sistema ciber-físico o en el uso que se desee. Este microprocesador forma parte de la familia de microcontroladores PIC18 y de los microprocesadores PIC18F2455/2550/4455/4550.

Además, la memoria de programa para este tipo de microprocesadores se divide en diferentes instrucciones y tiene gran capacidad de almacenamiento para estas, ya que puede almacenar en su interior como máximo número de instrucciones hasta 16,384 bytes para solo una palabra, lo cual significa una precisión de gran amplitud. También, demuestra una gran capacidad para lecturas de posiciones pues cuenta con 32 KB (32768 bytes) para memoria Flash así como 48 HMz de frecuencia máxima de operación para un procesamiento de alta velocidad.

A partir de esto se puede decir que la dimensión máxima de palabra que emula este microprocesador es de 16 bits, lo cual es suficiente para los comandos que se requieran. Dentro de la creación de un sistema mínimo de cómputo lo anterior se expresa a través del siguiente Opcode y Address para la distribución de los 16 bits:



$$\text{Máximo de palabra} = \frac{32768}{16384} = 2 \text{ bytes} = 16 \text{ bits}$$

Otro aspecto que hace increíble a este microprocesador es su alto rendimiento, ya que su consumo de energía es bajo pero de muy buena calidad pues su promedio de consumo de energía va de 4.2V a 5.5V, así como con un voltaje de funcionamiento en un rango de 2.0V a 5.5V.

En las siguientes tablas comparativas de los diferentes microprocesadores de la familia PIC18F2455/2550/4455/4550 se pueden observar algunos de los datos que se mencionaron con anterioridad:

Device	Program Memory		Data Memory		I/O	10-Bit A/D (ch)	CCP/ECCP (PWM)	SPP	MSSP		EUSART	Comparators	Timers 8/16-Bit
	Flash (bytes)	# Single-Word Instructions	SRAM (bytes)	EEPROM (bytes)					SPI	Master I <sup>2</sup> C™			
PIC18F2455	24K	12288	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F2550	32K	16384	2048	256	24	10	2/0	No	Y	Y	1	2	1/3
PIC18F4455	24K	12288	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3
PIC18F4550	32K	16384	2048	256	35	13	1/1	Yes	Y	Y	1	2	1/3

TABLE 1-1: DEVICE FEATURES

Features	PIC18F2455	PIC18F2550	PIC18F4455	PIC18F4550
Operating Frequency	DC – 48 MHz	DC – 48 MHz	DC – 48 MHz	DC – 48 MHz
Program Memory (Bytes)	24576	32768	24576	32768
Program Memory (Instructions)	12288	16384	12288	16384
Data Memory (Bytes)	2048	2048	2048	2048
Data EEPROM Memory (Bytes)	256	256	256	256
Interrupt Sources	19	19	20	20
I/O Ports	Ports A, B, C, (E)	Ports A, B, C, (E)	Ports A, B, C, D, E	Ports A, B, C, D, E
Timers	4	4	4	4
Capture/Compare/PWM Modules	2	2	1	1
Enhanced Capture/Compare/PWM Modules	0	0	1	1
Serial Communications	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART	MSSP, Enhanced USART
Universal Serial Bus (USB) Module	1	1	1	1
Streaming Parallel Port (SPP)	No	No	Yes	Yes
10-Bit Analog-to-Digital Module	10 Input Channels	10 Input Channels	13 Input Channels	13 Input Channels
Comparators	2	2	2	2
Resets (and Delays)	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT	POR, BOR, RESET Instruction, Stack Full, Stack Underflow (PWRT, OST), MCLR (optional), WDT
Programmable Low-Voltage Detect	Yes	Yes	Yes	Yes
Programmable Brown-out Reset	Yes	Yes	Yes	Yes
Instruction Set	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled	75 Instructions; 83 with Extended Instruction Set enabled
Packages	28-pin PDIP 28-pin SOIC	28-pin PDIP 28-pin SOIC	40-pin PDIP 44-pin QFN 44-pin TQFP	40-pin PDIP 44-pin QFN 44-pin TQFP

Después de haber revisado los anteriores datos se puede hacer diferentes comparaciones con los otros microprocesadores utilizados dentro de la industria automotriz. El ECU o Engine Control Unit, como se les conoce comúnmente a los microprocesadores y microchips de diferentes tipos; es la computadora interna del automóvil que realiza la administración de múltiples tareas dentro de este, como determinar la cantidad de combustible, el punto de ignición y otros parámetros monitorizando el motor a través de distintos sensores y actuadores. Los cuales mediante sincronización con los circuitos y semiconductores logran almacenar en su memoria las instrucciones /tareas a realizar e indicarles a dónde deben dirigirse para su posible ejecución.

Es importante recalcar que debido a los microprocesadores utilizados para regular el sistema ciber-físico, significa que este tipo de chips son de sistemas digitales y no solamente mecánicos o electrónicos, Además, para realizar este tipo de tareas se necesita de la ciencia de datos y no solamente es un sistema exclusivo de internet ya que se utilizan para todo y están presentes en cualquier lado, incluso en el correcto funcionamiento del automóvil.

Actualmente, hay entre 25 y 35 microcontroladores en un automóvil típico y entre 60 y 100 en un automóvil de lujo. Uno de los microcontroladores más utilizados es el microprocesador MC9S12A512, el cual es usado para la regulación y correcto funcionamiento de las bombas eléctricas y sistemas eléctricos en diferentes marcas de automóviles. En la siguiente tabla comparativa se pueden observar las diferencias de algunos de los datos que se mencionaron con anterioridad entre el microprocesador PIC18F4550 y el MC9S12A512:

Feature	PIC18F4550	MC9S12A512
Program Memory (Single-Word Instructions)	16,384 Bytes	4 KB
Program Memory (Flash)	32 KB	512 KB
Timers	2 Bytes / 16 bit	128 Bytes / 1024 bits
Operating Frequency	48 MHz	50 MHz
Voltage Range	4.2V - 5.5V	2.5V - 5.0V

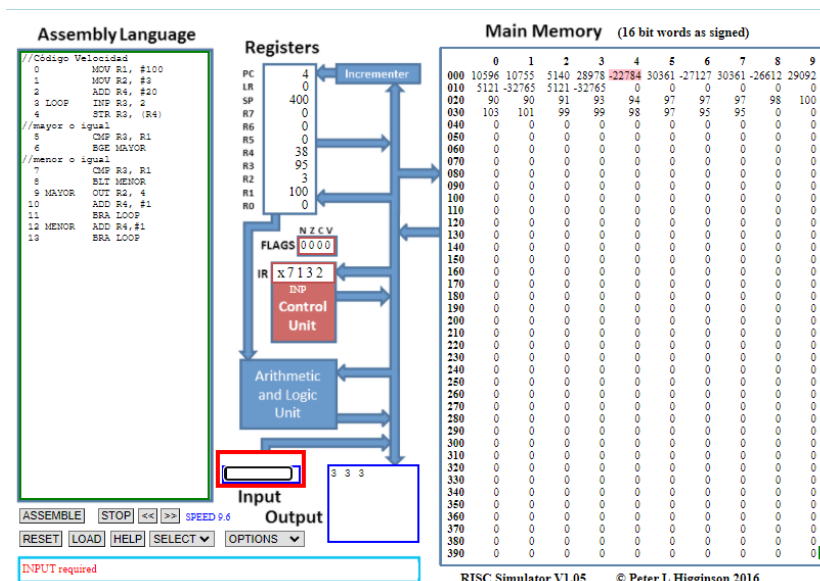
Para desarrollar el programa en lenguaje ensamblador se debe tomar como referencia dos sensores que entregan una lectura por medio de un bus paralelo que se encargan de almacenar la velocidad del vehículo y la distancia entre este y otros vehículos a su alrededor para que cuando se encuentre a alta velocidad o esté muy cerca de otro vehículo se muestre una alerta en pantalla que informe al usuario generando que este vaya debajo del límite de velocidad y mantenga su distancia de otros vehículos.

Aunque cada uno de los programas trabaja independientemente, ambos tienen el mismo esquema de trabajo. El cual consiste en recibir un valor, almacenarlo y si se cumple cierta condición (en el caso del sensor de velocidad si este es mayor que 120 y con el sensor de distancia si este es menor a 12) se lanza un mensaje de alerta al usuario para que este actúe según sea el caso para prevenir algún percance automovilístico que pueda poner en riesgo la vida del usuario y sus acompañantes.

Cada uno de los programas ocupará su propio espacio de memoria aunque considerando que cada programa cuenta con 14 líneas, es decir, 14 instrucciones y cada instrucción cuenta con su opcode y address correspondiente. Esto significa que cada instrucción almacena en binario 16 bits de memoria por lo que se puede determinar que el tamaño del programa para cada sensor es de 224 bits, lo cual equivale a que cada uno ocupa 28 Bytes de memoria respectivamente. Además, cabe recordar que estos sensores serán analógicos ya que la señal es continua y puede tomar valores infinitos.

Video demostrativo: <https://acortar.link/7Th9az>

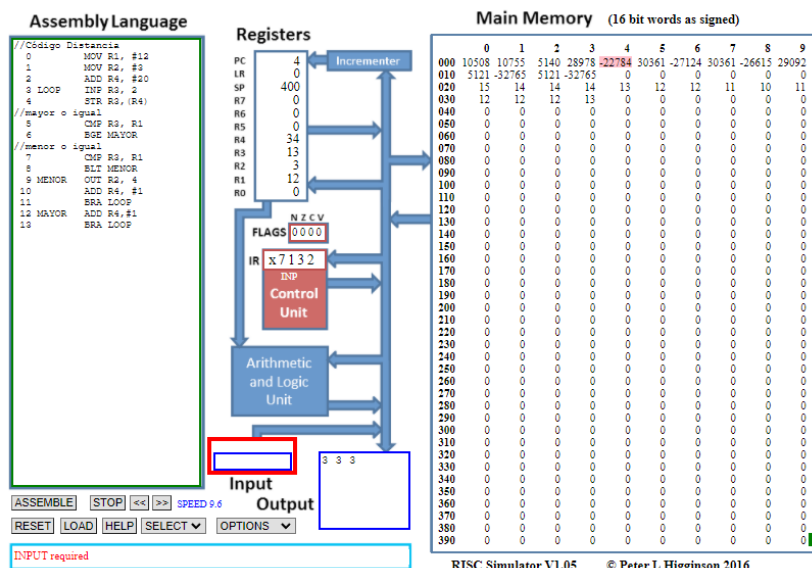
- Sensor de velocidad



```

//Codigo Velocidad
MOV R1, #100
MOV R2, #3
ADD R4, #20
LOOP INP R3, 2
STR R3, (R4)
//mayor o igual
CMP R3, R1
BGE MAYOR
//menor o igual
CMP R3, R1
BLT MENOR
MAYOR OUT R2, 4
ADD R4, #1
BRA LOOP
MENOR ADD R4, #1
BRA LOOP
  
```

- Sensor de distancia



```
//Código Distancia
MOV R1, #12
MOV R2, #3
ADD R4, #20
LOOP INP R3, 2
STR R3,(R4)
//mayor o igual
CMP R3, R1
BGE MAYOR
//menor o igual
CMP R3, R1
BLT MENOR
MENOR OUT R2, 4
ADD R4, #1
BRA LOOP
MAYOR ADD R4, #1
BRA LOOP
```

Comparando el procesador MC9S12A512 usado en la industria automotriz común con el utilizado en la solución de este reto podemos definir que ambos cuentan con pocas características en común y en cambio existen varias brechas de diferencias entre ambos por ejemplo, en el procesador que usamos la cantidad de bytes con los que cuenta es casi 4 veces mayor que el usado comúnmente en los automóviles. En cambio, este último tiene aproximadamente 16 veces más almacenamiento de memoria en bytes y por consiguiente una dimensión de palabra generada más grande que el implementado por nosotros. Por lo tanto, se concluye que el procesador MC9S12A512 cuenta con mayor espacio para su uso industrial a pesar de que su capacidad para el almacenamiento de instrucciones es menor y rústico, pero ambos procesadores tienen un consumo de energía promedio no mayor a los 5.5V, por lo tanto no se asume que estas diferencias deben ser tan relevantes para el reto sino de la versión misma la cual se utiliza del atributo dentro de cada microchip, el cual tendrá cambios mínimos. Así mismo, cabe resaltar situaciones como esta el poder de la ciencia de datos, ya que hoy en día gracias a la industria 4.0, el análisis y recopilación de los mismos está presente en todas partes y puede ser utilizada para el bien común de la sociedad, en este caso la seguridad de los usuarios al transportarse en automóviles.

- Reflexión Alejandro:

Conforme los años van pasando los automóviles y la tecnología van avanzando y evolucionando de la mano y era ilógico imaginar que estos dos nos se iban a combinar en algún momento , la mayoría de los autos de la actualidad traen más circuitos y sensores que varias computadoras y teléfonos. Esto se debe principalmente a que la mayoría de accidentes son por distracciones humanas y mediante la implementación de sensores hemos podido reducir la cantidad de accidentes gracias a la información extra que nos brindan estos sensores a la hora de manejar.

- Reflexión Daniela Berenice:

Antes de resolver el reto planteado realmente el conocimiento que tenía sobre estas situaciones era casi nulo, pero conforme los avances fueron realizados, pude comprender como funciona el lenguaje ensamblador, y es que antes de llevar esta materia, pensaba que la programación de un coche o del sensor de un coche, sería parecida a la programación que se realiza en arduino o en otro tipo de aplicaciones de ese índole.

De igual manera gracias a los avances tecnológicos que se han tenido en la industria automotriz se han podido incluso prevenir grandes accidentes, y se han hecho de las calles un lugar más seguro, realmente considero que en algún punto esto nos llevará a un automóvil sin conductor.

- Reflexión Jesús:

A través de los años la transformación digital dentro de la industria automotriz ha incrementado el análisis e implementación de sensores en dicha industria, lo cual facilita la solución de problemas como el planteado en esta situación. Todo esto permite que se logren avances que hace un par de años eran considerados imposibles trayendo consigo beneficios al momento de hablar de transporte y seguridad vial, ya que se pueden detectar algún fallo o problema a tiempo reduciendo así cualquier posibilidad de algún accidente automovilístico.

#### Referencias Bibliográficas:

Carrillo, M. (2021, 18 marzo). 6 principales causas de accidentes viales en carretera. Recuperado el 28 de septiembre de 2021, de Fleet Complete México. Sitio web: <https://blog.fleetcomplete.mx/6-principales-causas-de-accidentes-viales-en-carretera>

Freescall Semiconductor, Inc. (2001). MC9S12DP512 Device Guide V01.25. Recuperado el 15 de octubre de 2021, de: <https://bit.ly/3AKg8uT>

INEGI. (2015). Accidentes de tránsito terrestre en zonas urbanas y suburbanas. Recuperado el 28 de septiembre de 2021, de Información de 1997 a 2016. Sitio web: <http://www.inegi.org.mx/est/contenidos/proyectos/registros/economicas/accidentes/descripciones.aspx>

INSP. (2020). México, séptimo lugar mundial en siniestros viales. Recuperado el 28 de septiembre de 2021, de Gobierno de México. Sitio web: <https://www.insp.mx/avisos/4761-seguridad-vial-accidentes-transito.html>

López, D; (2019). Funcionamiento básico de la Unidad de Control de Motor (ECU). Recuperado el 15 de octubre de 2021, de: <https://bit.ly/3IIYmUD>

Microchip Technology Inc. (2009). 28/40/44-Pin, low-Power, High-Performance, Enhanced Flash, USB Microcontrollers with nanoWatt Technology. Recuperado el 15 de octubre de 2021, de: <https://bit.ly/3aLaI8t>

OMS. (2021, 21 junio). Traumatismos causados por el tránsito. Recuperado el 28 de septiembre de 2021, de Organización Mundial de la Salud. Sitio web: <https://www.who.int/es/news-room/fact-sheets/detail/road-traffic-injuries>