

# STRD – Detección de distracciones al volante

Javier Alonso Silva  
Alfonso Díez Ramírez  
Sara Moreno Prieto  
Mihai Octavian Stănescu

2021

## Resumen

Se desarrolla un sistema de detección de distracciones al volante el cual se espera ayude a evitar los posibles accidentes derivados de la casuística anterior.

El desarrollo consiste en una evaluación de los requisitos, modelado del sistema mediante diagramas SysML hasta una implementación final en dos nodos diferenciados los cuales se comunican entre sí usando la tecnología CANBus.

El primer nodo (*nodo 1*) tendrá una carga balanceada entre la lectura de dispositivos así como la intervención en elementos físicos del vehículo, como son los frenos; y a su vez será el encargado de una transmisión continua de mensajes hacia el segundo nodo. El *nodo 2* leerá información sobre el estado psico-físico del conductor y, junto con la información recibida del *nodo 1*, alertará al mismo sobre distintos factores que se han visto peligrosos para que pueda reconducir su comportamiento. Finalmente, se ofrece al conductor un método para evitar ser distraído por el propio sistema pudiendo decidir entre tres niveles de avisos: completo, parcial e inactivo.

---

# Índice

<b>1. Introducción</b>	<b>1</b>
1.1. Nodo 1 . . . . .	2
1.2. Nodo 2 . . . . .	4
<b>2. Implementación</b>	<b>5</b>
2.1. Nodo 1 . . . . .	5
2.2. Nodo 2 . . . . .	5
<b>3. Diseño final</b>	<b>5</b>
<b>4. Aclaraciones</b>	<b>5</b>
<b>5. Glosario</b>	<b>5</b>
<b>A. Código fuente nodo 1</b>	<b>5</b>
<b>B. Código fuente nodo 2</b>	<b>5</b>

## 1. Introducción

Una de las mayores causas de accidentes son las distracciones de los conductores al volante, o bien por el uso de dispositivos electrónicos, somnolencia u otras acciones que llevan a la persona a no prestar atención a la carretera y su entorno.

A raíz de ese problema, los mecanismos de regulación internacionales han invertido tiempo, dinero y desarrollo en los sistemas ADAS (*Advanced Driving Assistance Systems*), con el fin de mitigar las situaciones anteriores y realizar una prevención activa sobre los accidentes de tráfico. Sin embargo, dichos sistemas no cuentan con una penetración significativa en el mercado, por lo que interesa agilizar su implantación y que pasen a ser un elemento de seguridad “por defecto” en los nuevos vehículos.

En este contexto, se ha pedido realizar una implementación distribuida, que cumpla con unos requisitos de tiempo real, en dos nodos que interactúan entre sí para actuar como un organismo conjunto sobre un vehículo como sistema ADAS.

El sistema a desarrollar contará con múltiples sensores:

- Giroscopio, para detectar en los ejes  $X$  y  $Y$  la inclinación de la cabeza del conductor y predecir una posible somnolencia.
- Giro del volante, para detectar si el conductor está pegando volantazos o está realizando “mini-correcciones”, características de un estado de somnolencia o de atender al móvil.
- Agarre del volante, donde se indicará si el conductor está agarrando el volante o no.
- Velocímetro, con un rango de valores comprendido entre los  $[0, 200] \text{ km/h}$ . Se usará para comprobar que se cumple la distancia de seguridad.
- Sensor de distancia, capaz de realizar lecturas en el rango  $[5, 200] \text{ m}$  y que le indicará al conductor si está cumpliendo o no la distancia de seguridad, según la velocidad a la que circule.

y múltiples actuadores:

- Luces de aviso, las cuales se usarán para emitir señales luminosas al conductor indicando cierto nivel de riesgo que se está produciendo.
- *Display*, usado para visualizar los datos que obtiene el sistema.
- Alarma sonora, emitiendo un sonido con 3 niveles de intensidad.
- Luz de aviso/freno automático, donde ante un peligro de colisión inminente el sistema podrá activar el freno con hasta 3 niveles de intensidad.

Cada uno de los sensores/actuadores estarán controlados y monitorizados por una o varias tareas las cuales registran los datos en objetos protegidos. Dichas tareas vienen definidas con sus periodos y *deadlines* en el cuadro 1:

Tareas/objetos protegidos	Tipo	$T_i$	$D_i$	WCET	Síntomas 1	Síntomas 2	Modo
Inclinación cabeza	C	600	400	?	$x_1$		
Detección de volantazos	C	400	400	?	$x_1$		
Cálculo distancia	C	300	300	?		$y_1$	
Relax al volante	C	500	200	?	$x_1$		
Emergencias	C	300	300	?	$x_2$	$y_2$	$z_2$
Mostrar información	C	2000	2000	?	$x_2$	$y_2$	
Detección pulsador	S	-	100	?			$z_1$
Síntomas 1	P	-	-	$x_1, x_2$			
Síntomas 2	P	-	-	$y_1, y_2$			
Modo	P	-	-	$z_1, z_2$			

Cuadro 1: Listado de tareas y objetos protegidos junto con sus tiempos.

Como hay multitud de tareas y se cuenta con dos nodos, el sistema a implementar irá distribuido entre ambos y viene representado por la figura 1:

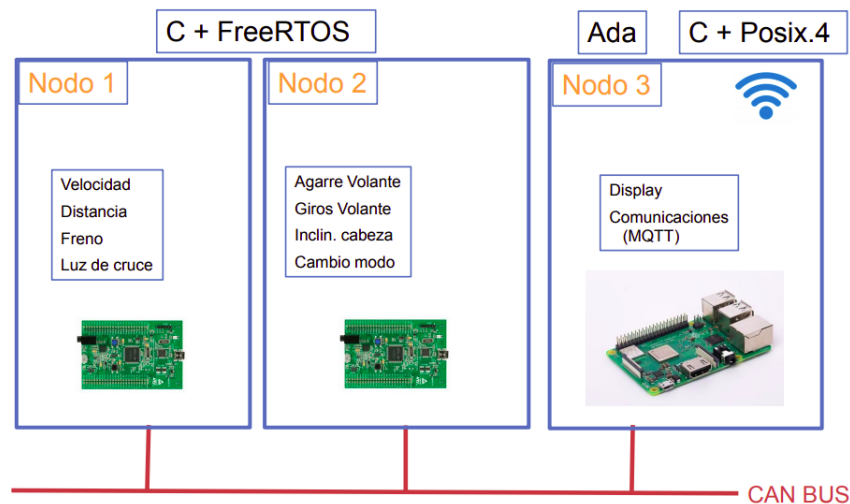


Figura 1: Modelo completo del sistema a implementar. Las tareas van distribuidas entre los dos nodos principales y se comunican entre ellos mediante CANBus.

### 1.1. Nodo 1

El primer nodo será el encargado principalmente de la actuación sobre distintos elementos del sistema, a saber: el freno, las luces de cruce e indirectamente sobre la alarma. Esto lo hará recogiendo datos de distintos sensores como son el velocímetro, el sensor de distancia y el sensor de luminosidad para adecuar su comportamiento a las circunstancias del entorno.

Este sistema contará con cuatro tareas en tiempo real y usará dos objetos protegidos: el primero de ellos para conservar el valor de la velocidad actual; y el segundo para guardar tanto el valor de la distancia con el vehículo precedente como la intensidad del freno que se ha de

aplicar en caso de peligro de colisión. Por su parte, las tareas en cuestión son:

1. Cálculo velocidad – cada 250 ms, realizará una lectura del sensor en cuestión mediante el ADC y actualizará el valor del objeto protegido  $V_{\text{actual}}$ .
2. Cálculo distancia – cada 300 ms, el sistema obtendrá la distancia con el vehículo precedente usando el sensor de ultrasonidos y actualizará el valor del objeto protegido  $D_{\text{actual}}$ . Además, leerá el valor de  $V_{\text{actual}}$  y computará lo que sería la distancia de seguridad mínima que hay que respetar, descrita por la ecuación 1:

$$d_{\min} = \left( \frac{V}{10} \right)^2, \begin{cases} d_{\min} & : \text{distancia mínima que hay que mantener.} \\ V & : \text{velocidad actual del vehículo.} \end{cases} \quad (1)$$

En caso de que la distancia de seguridad no se cumpla (y según el valor relativo con que no se cumple), la tarea indicará en  $\text{Intens\_Frenada}$  con qué intensidad se ha de aplicar el freno para evitar una colisión. Finalmente, activará la tarea esporádica Freno para que realice su ejecución.

3. Freno – cada 150 ms como mucho, realizará la activación progresiva del freno cada 100 ms hasta alcanzar la intensidad apropiada. Al ser una tarea esporádica, depende directamente de la activación desde Cálculo distancia, lo cual añadirá un *jitter* al tiempo de respuesta global de la tarea.
4. Luces de cruce – cada 1 000 ms, el sistema realizará una valoración de la luminosidad del entorno y procederá a encender o apagar automáticamente las luces de cruce. Se establece que las luces se activarán si la intensidad lumínica está por debajo de 100.

Todo este sistema viene modelado por la figura 2:

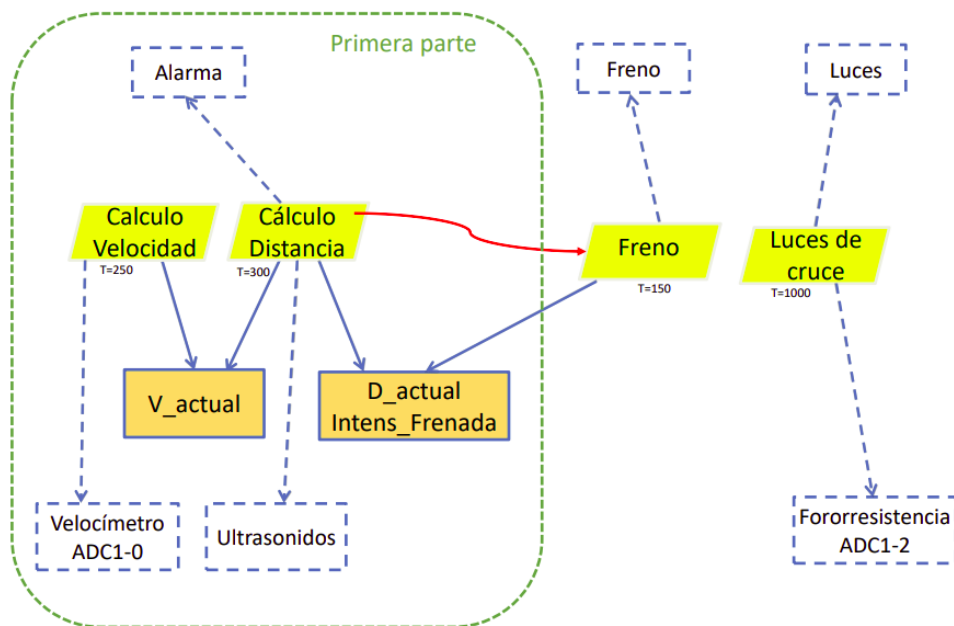


Figura 2: Modelado del nodo 1 junto con sus tareas, objetos protegidos, sensores y actuadores.

## 1.2. Nodo 2

El segundo nodo se encargará directamente de notificar al conductor cuando algún comportamiento es errático o peligroso. Entre otras tareas, este nodo se encarga de monitorizar el estado del conductor (y detectar posibles signos de somnolencia) y emitir avisos luminosos y sonoros cuando se produzcan situaciones de riesgo.

Este sistema cuenta con cinco tareas en tiempo real y tres objetos protegidos: el primero recoge datos sobre síntomas como son la inclinación de la cabeza o el giro del volante; el segundo, recoge información sobre si el conductor está sujetando o no el volante; y el tercero establecerá el modo de funcionamiento de los avisos del sistema. Con respecto a las tareas, se tiene:

1. Inclinación cabeza – cada 600 ms, leerá el valor del giroscopio integrado para actualizar los datos de las posiciones  $X$  e  $Y$ , en el objeto protegido Síntomas 1.
2. Detección volantazos – cada 400 ms el sistema leerá el valor de la posición del volante y actualizará el dato recogido en Síntomas 1.
3. Relax al volante – cada 500 ms, el sistema actualizará en Síntomas 2 si el conductor está sujetando o no el volante.
4. Detección pulsador – tarea esporádica que será activada desde la rutina de tratamiento de interrupciones *hardware* que establecerá cíclicamente el modo de funcionamiento del sistema en el objeto protegido Modo.
5. Riesgos – cada 300 ms, el sistema evaluará los datos recogidos en los objetos protegidos Síntomas 1, Síntomas 2 y Modo y establecerá el nivel de alarma para con el conductor. Dicha detección de riesgos viene definida por la siguiente secuencia:
  - Si el conductor presenta una inclinación de la cabeza en los ejes  $X, Y$  de más de  $20^\circ$  y no tiene sujeto el volante se considera que está manipulando el móvil u otro aparato. Se activa la luz amarilla y se emite un pitido nivel 1.
  - Si la inclinación de la cabeza es  $X > 20^\circ | Y > 20^\circ$ , el volante está agarrado y la velocidad es mayor de  $70^{km/h}$  se interpreta que el conductor no está prestando atención a la carretera y se encenderá la luz amarilla.
  - Si se detecta una inclinación en el eje  $X$  de más de  $30^\circ$  y el conductor está dando volantazos se interpreta como síntoma de somnolencia. Se encenderá la luz amarilla y se emitirá un pitido nivel 2.
  - Si se dan simultáneamente dos de los riesgos anteriores se pasa a estar en **NIVEL 2** de alerta y se encenderá la luz roja y emitirá un pitido nivel 2.
  - Si se produce un riesgo **NIVEL 2** y la distancia con el vehículo precedente es menor al 50 % de la distancia de seguridad recomendada, se estará ante una situación de **EMERGENCIA** y se activará el freno, junto con todo lo anterior.

## **2. Implementación**

### **2.1. Nodo 1**

### **2.2. Nodo 2**

## **3. Diseño final**

## **4. Aclaraciones**

## **5. Glosario**

### **A. Código fuente nodo 1**

### **B. Código fuente nodo 2**