

Trabajo Fin de Grado  
Grado en Ingeniería de las Tecnologías de  
Telecomunicación

Diseño y desarrollo de un sistema de  
salpicadero para un monoplace de Formu-  
la Student

Autor: Rubén Tinoco de Soto

Tutor: Juan Antonio Sánchez Segura

**Dpto. Ingeniería Electrónica**  
**Escuela Técnica Superior de Ingeniería**  
**Universidad de Sevilla**

Sevilla, 2023





Trabajo Fin de Grado  
Grado en Ingeniería de las Tecnologías de Telecomunicación

# **Diseño y desarrollo de un sistema de salpicadero para un monoplaza de Formula Student**

Autor:

Rubén Tinoco de Soto

Tutor:

Juan Antonio Sánchez Segura

Profesor Colaborador

Dpto. Ingeniería Electrónica  
Escuela Técnica Superior de Ingeniería  
Universidad de Sevilla

Sevilla, 2023



Trabajo Fin de Grado:   Diseño y desarrollo de un sistema de salpicadero para un mono-plaza de Formula Student

Autor:           Rubén Tinoco de Soto  
Tutor:           Juan Antonio Sánchez Segura

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocal/es:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:



# Agradecimientos

---

A mi familia, abuelos, padres y hermana, por soportar todos los momentos de frustración y seguir apoyándome pasase lo que pasase.

A mis compañeros, ya amigos para toda la vida, de la puerta 7. En especial a Manolo y Pepe por todas las horas juntos, habéis hecho que levantarme por la mañana haya sido más sencillo.

A ti, Mer, por aguantar mis enfados, frustraciones y malos momentos, día tras día, pase lo que pase.

A Blanca, una de las principales culpables de que esté aquí, dando mis últimas pinceladas en el grado. WNW.

Al equipo ARUS, por enseñarme a vivir el momento, aprender de todas las situaciones y a formar una pequeña gran familia que siempre llevaré conmigo allá donde vaya.

Y por último, a todos mis profesores, por toda la ayuda que me han aportado durante estos años para conseguir superar los objetivos.

*Gracias a todos. Había que confiar en el plan.*

*Sevilla, 2023*



# Resumen

---

En el presente documento se detallará el proceso de diseño, fabricación y desarrollo del sistema de salpicadero de los monoplazas ART20C y ART20E.

El objetivo del sistema es aportar a los pilotos de los monoplazas información acerca del estado del vehículo, relevante para tanto la integridad de la mecánica como mejora de la conducción de los mismos.

Además, en el caso del vehículo eléctrico, el ART20E, será necesario aclimatar el sistema para el cumplimiento de la normativa exigida por las competiciones.



# Abstract

---

This document will explain the design, manufacturing and development of dashboard system for single seated cars ART20C and ART20E.

The system's main goal is to bring information to the drivers both relevant to the integrity of the vehicles and improvement of drivability.

In addition, in the electric vehicle, ART20E, will be necessary to adapt the system to fulfil the FSAE regulations in competitions.



# Índice

---

<i>Resumen</i>	III
<i>Abstract</i>	V
<i>Notación</i>	XI
<b>1 ARUS Andalucía Racing Team</b>	<b>1</b>
1.1 Formula Student	1
1.1.1 Technical Inspections	1
1.1.2 Pruebas dinámicas	1
Acceleration	2
Skidpad	2
Autocross	2
Endurance y Efficiency	2
1.1.3 Pruebas estáticas	2
Design	2
Cost	3
Business Plan	3
<b>2 Contexto del sistema</b>	<b>5</b>
2.1 Punto de partida	5
2.1.1 Esquema general	5
2.1.2 Dashboard Controller PCB ART19	5
Componentes principales	7
2.1.3 Indicators PCB	7
Componentes principales	7
2.2 Contexto interdepartamental	7
2.2.1 ART20C (Combustion)	8
Motor	8
Electrónica	8
Ergonomía	8
2.2.2 ART20E (Eléctrico)	8
Ergonomía	9
2.3 Herencia. Fallos encontrados	9
2.3.1 Fallos generales	9
2.3.2 Fallos en el sistema del ART19C	10
2.3.3 Fallos en el sistema del ART19E	10

<b>3</b>	<b>Diseño actual</b>	<b>11</b>
3.1	Opciones	11
3.1.1	Opciones comerciales	11
	AiM MXG 1.2 Strada	11
	AiM GS-Dash	12
	AEM Electronics CD-5	13
3.1.2	Opciones de diseño propio	14
	Diseño continuista	14
	Integración completa	14
	Integración moderada	15
3.1.3	Resumen	15
	Pantalla comercial	15
	Diseño propio: diseño continuista	15
	Diseño propio: integración completa	16
	Diseño propio: integración moderada	16
3.2	Diseño e implementación del sistema. Estudios previos	16
3.2.1	Diagrama de bloques general	16
3.2.2	Entorno del sistema. Fuentes de datos (CAN Bus)	17
3.2.3	Interfaz visual	18
	ART20C	18
	ART20E	19
3.3	Diseño e implementación del sistema. Hardware	20
3.3.1	Dashboard Controller PCB. Esquemático	20
	Unidad de alimentación	21
	Unidad del microcontrolador	21
	Módulo de CAN	21
	Módulo decoder	23
	Módulo auxiliar	23
	Módulo de interfaz de salida	23
3.3.2	Dashboard Controller PCB. Rutado	25
3.3.3	Dashboard Indicators PCB. Esquemático	25
3.3.4	Dashboard Indicators PCB. Rutado	27
3.4	Diseño e implementación del sistema. Software.	28
3.4.1	Código ART20C. Diagrama de flujo	30
3.4.2	Código ART20C. Fragmentos.	30
3.4.3	Código ART20E. Diagrama de flujo	33
3.4.4	Código ART20E. Fragmentos	33
<b>4</b>	<b>Rendimiento y conclusiones</b>	<b>35</b>
4.1	Decisiones tomadas y sus conclusiones	35
4.1.1	General	35
4.1.2	ART20C	36
4.1.3	ART20E	36
4.2	Estudio de consumo	36
4.2.1	Consumo en el sistema del ART20C	36
4.2.2	Consumo en el sistema del ART20E	37
4.3	Comparativa de precios	37
4.4	Mejoras futuras	38

---

Integración completa	38
Inclusión de pantalla TFT	38
<b>Apéndice A Esquemáticos del sistema</b>	<b>41</b>
<i>Índice de Figuras</i>	53
<i>Índice de Tablas</i>	55
<i>Índice de Códigos</i>	57
<i>Bibliografía</i>	59



# Notación

---

<i>PCB</i>	Printed Circuit Board
<i>LED</i>	Light Emitting Diode
<i>SPI</i>	Serial Peripheral Interface
<i>AMS</i>	Accumulator Management System
<i>IMD</i>	Insolation Monitoring Device
<i>CAN</i>	Controller Area Network
<i>RPM</i>	Revoluciones por minuto
<i>BCD</i>	Binary-Coded Decimal
<i>ECU</i>	Engine Control Unit
<i>MISO</i>	Master Input Slave Output
<i>MOSI</i>	Master Output Slave Input
<i>SCK</i>	Serial Clock
<i>GND</i>	Ground



# 1 ARUS Andalucía Racing Team

---

El equipo ARUS Andalucía Racing Team es el representante de la Universidad de Sevilla en las competiciones de Formula Student. ARUS diseñó, fabricó y desarrolló durante las temporadas 2019-20 y 2020-21 dos monoplazas de competición uno eléctrico, el ART20E y otro de combustión el ART20C. Estos vehículos de competición aparecieron en las siguientes citas de las competiciones Formula Student.

- ART20C
  - Formula Student Netherlands (Circuito de Assen)
  - Formula Student Austria (Red Bull Ring Circuit)
  - Formula Student Spain (Circuit de Barcelona-Catalunya)
  - Formula Student Germany (Hockenheimring)
- ART20E
  - Formula Student Netherlands (Circuito de Assen)
  - Formula Student Spain (Circuit de Barcelona-Catalunya)
  - Formula Student Germany (Hockenheimring)

## 1.1 Formula Student

El acceso a cualquier competición del contexto Formula Student está determinado por un examen grupal para todos los equipos candidatos en el que más de 600 equipos de todo el mundo participan y que tan solo unos pocos logran superar. Estos exámenes se realizan el último día del mes de enero y dan acceso a las competiciones que se llevan a cabo durante todo el verano. A continuación, se desglosarán las pruebas llevadas a cabo en todas las competiciones.

### 1.1.1 Technical Inspections

Antes de poder poner cada monoplaza en pista es necesario que los inspectores técnicos de la competición pasen las pruebas técnicas sobre los coches para verificar que cumplen la normativa. Entre las inspecciones se encuentran verificaciones mecánicas, eléctricas, electrónicas, de frenos, etc.

### 1.1.2 Pruebas dinámicas

Una vez superadas todas las verificaciones sobre los monoplazas, estos pueden comenzar la competición. Las pruebas dinámicas son las siguientes.

### Acceleration

Prueba en la que el monoplaza se enfrenta a una recta de 75 metros cuyo objetivo es completarla en el menor tiempo posible.

### Skidpad

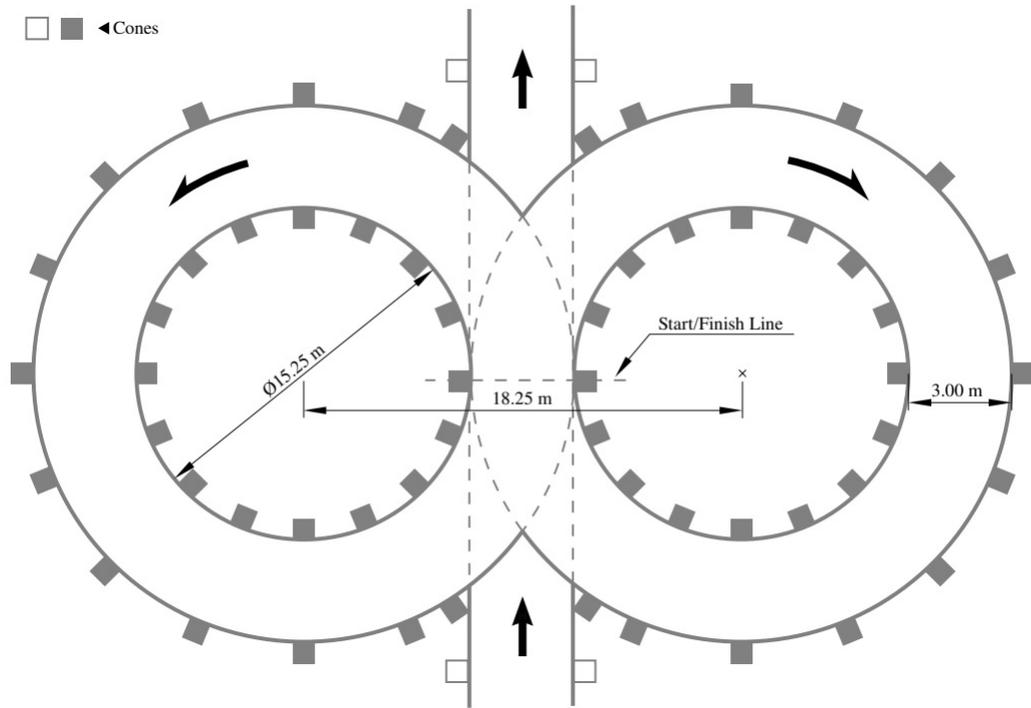


Figura 1.1 Pista para la prueba de skidpad.

En la Figura 1.1 se encuentra el trazado de esta prueba, en la que se tienen varios intentos para intentar conseguir el mejor tiempo posible.

### Autocross

En esta prueba cada competición pone su "Track layout" en la que cada equipo puede hacer 2 intentos con 2 pilotos y determina la posición de salida a pista en la prueba reina de la competición, la Endurance.

### Endurance y Efficiency

Con el mismo "Track layout" que en la prueba de Autocross, cada equipo tiene que completar 22 kilómetros en el menor tiempo posible con dos pilotos distintos. Esto va acompañado con la prueba "Efficiency" en la cual se mide el consumo de energía (ya sea eléctrica en el caso de las pruebas de vehículos eléctricos o gasolina en el caso de vehículos de competición) utilizada para completar la prueba completa.

#### 1.1.3 Pruebas estáticas

Se denominan pruebas estáticas debido a que no se pone el coche en pista para determinar la puntuación en la prueba. En los siguientes apartados se explicarán brevemente.

### Design

La prueba de Design consiste en exponer los sistemas más característicos del coche, así como los procesos de diseño y fabricación utilizados para mejorar el rendimiento del coche de un año a otro.

**Cost**

En esta prueba se trata de conseguir un desglose de precios detallado minuciosamente de cada materia prima y proceso de manera organizada. Además, se plantea el reto de planificar una fabricación en masa del sistema que desee cada competición.

**Business Plan**

Por último, la prueba estrella de ARUS, líderes indiscutibles mundialmente, en la que se trata de realizar un plan de negocios detallado con los coches como parte central de la presentación.



## 2 Contexto del sistema

---

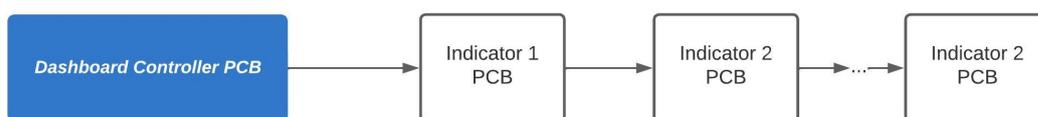
El sistema de salpicadero de los coches ART20C y ART20E no es nuevo en el equipo ARUS, ya que lleva siendo desarrollado desde el inicio de la vida del equipo. En este capítulo se explicará el contexto general del sistema, de donde procede, qué departamentos trabajan en él y las mejoras que se plantearon realizar para la temporada.

### 2.1 Punto de partida

Antes de dar paso a los requisitos del sistema para la nueva temporada y de los errores a corregir en el sistema, se hará un repaso de donde se partió para el diseño del nuevo sistema de salpicadero.

#### 2.1.1 Esquema general

En primer lugar, se desglosarán las partes que componen el sistema de salpicadero al completo. El sistema de salpicadero está formado por varias PCB, la PCB controladora o Dashboard Controller PCB y las PCB periféricas o Indicators PCBs. El esquema general se puede resumir en la Figura 2.1. De esta forma las PCB periféricas se conectan en serie y están ubicadas en la parte posterior de la



**Figura 2.1** Esquema general del sistema de salpicadero de los ART19.

ranura de cada indicador necesario, como se podrá ver en la Figura 2.4 y la Figura 2.5. Gracias a esta disposición se consigue dividir el problema en varias PCB, permitiendo así ahorrar tiempo a la hora de realizar trabajos en paralelo.

#### 2.1.2 Dashboard Controller PCB ART19

Se procede ahora a detallar el estado de cada integrante del sistema de salpicadero, comenzando con la PCB controladora. En la Figura 2.2 y la Figura 2.3 se puede ver un renderizado 3D de la PCB en la que se puede apreciar los componentes principales de la misma.

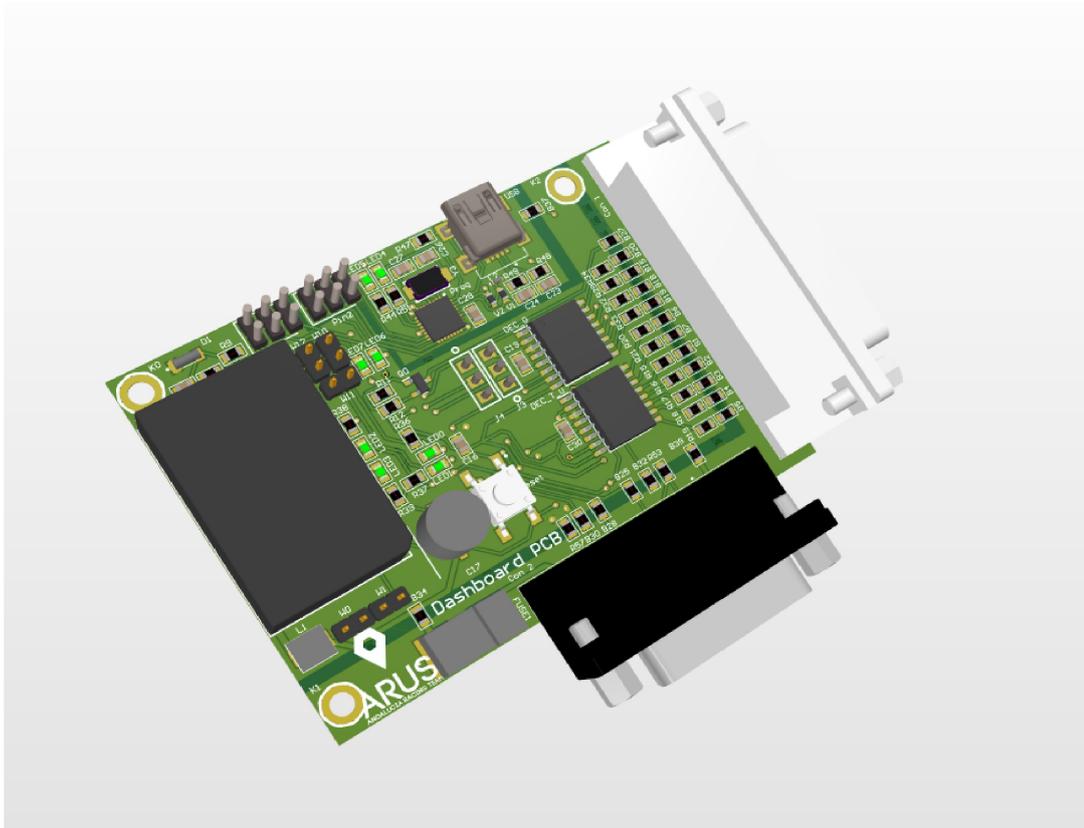


Figura 2.2 Renderizado 3D frontal de la PCB Dashboard Controller ART19.

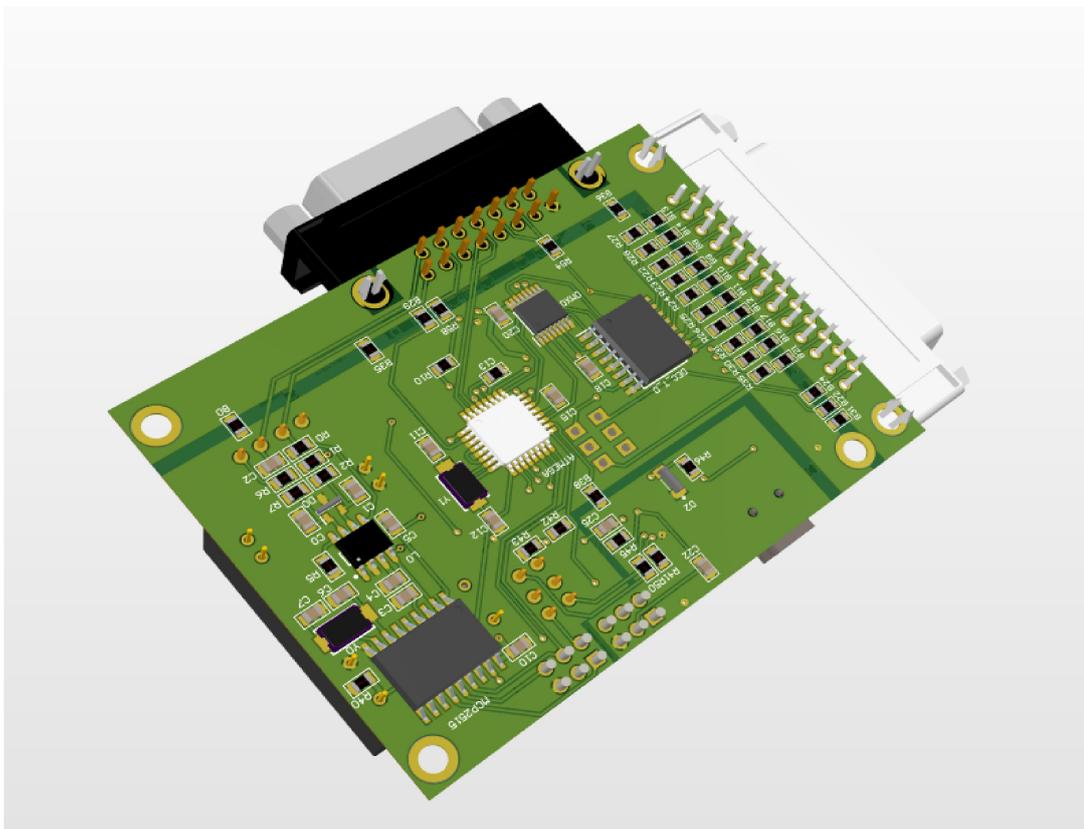


Figura 2.3 Renderizado 3D reverso de la PCB Dashboard Controller ART19.

### Componentes principales

- DCDC: Conversor de corriente continua a continua de 13V a 5V para la alimentación general del sistema. El modelo instalado es el TEN 6-1211N del fabricante TracoPower.
- Microcontrolador: encargado de procesar todas las señales provenientes de los sensores y sistemas electrónicos del vehículo y encender aquellos indicadores necesarios. El modelo elegido es el ATMega328P-AUR del fabricante ATMEL.
- Conversores BCD a 7 Segmentos: debido al limitado número de pines disponibles en el microcontrolador se necesita un conversor de código binario a 7 segmentos. De esta se pasa de 8 pines necesarios a tan solo 4 para encender los 7 segmentos.
- Demultiplexor: de nuevo, necesario para ahorrar pines para encender todos los displays de 7 segmentos instalados en las PCB periféricas.
- Módulo de CAN-Bus: formado por:
  - Controlador de CAN: encargado de generar las secuencias de bits necesarias para entender el protocolo CAN para el microcontrolador. Se escoge el modelo MCP2515 de MicroChip.
  - Transceptor de CAN: traduce el protocolo de CAN-Bus al mismo protocolo del controlador de CAN mencionado anteriormente, en este caso, ambos se comunican mediante UART.
- Módulo de programación: formado por el microcontrolador ATMega16U2, pensado para poder programar el microcontrolador principal de la controladora directamente vía USB. (No implementado)

#### 2.1.3 Indicators PCB

Como se ha indicado anteriormente, cada PCB periférica se instala detrás de la ranura mecánica impuesta por el departamento de ergonomía. Se verán los componentes principales.

### Componentes principales

- Para PCB de indicadores luminosos:
  - LEDs WS2812B: son LEDs programables, por lo que su conexión serie se realiza de tal forma que se envía por un solo pin del microcontrolador una trama de instrucciones. Esta trama contiene el color y brillo de cada uno de los LEDs conectados en la misma rama.
- Para PCB con indicadores numéricos:
  - Display 7 segmentos: se instalan tantos displays como sean necesarios en la misma placa. Estos son escogidos según disponibilidad de stock para unas medidas determinadas.
- Para PCB de mandos:
  - Mando rotativo: se instala un mando rotativo en conjunción con tantos divisores de tensión como posiciones tiene el mando. De esta forma el microcontrolador puede leer la posición que tiene seleccionada el piloto y ajustar sus parámetros internos.

Se puede encontrar un renderizado 3D de estas PCB periféricas en la Figura 2.4 y Figura 2.5

## 2.2 Contexto interdepartamental

Se detallarán las exigencias se imponen desde cada departamento y en cada coche.

### 2.2.1 ART20C (Combustion)

#### Motor

El departamento de motor no trabaja directamente sobre el sistema de salpicadero, sino que establece ciertos requisitos necesarios para mostrar al piloto determinada información.

En su caso desean mostrar los siguientes datos:

- Revoluciones del motor: esta información será necesaria para el piloto para determinar el punto de acelerador mínimo necesario para el cambio de marcha y para la salida en parado a máxima potencia.
- Temperatura del sistema de refrigeración: deberá ser mostrada mediante dos indicadores, en los que se muestra la temperatura del agua del radiador y la temperatura del aire de entrada al motor. Se utilizará para que el piloto pueda detectar fallos en el sistema de refrigeración y así poder parar el coche antes de que ocurra una avería grave.
- Temperatura de aceite: mostrada mediante un solo indicador. Gracias a esta información el piloto podrá exigir mecánicamente más o menos al motor en función de lo que vea en ese indicador.
- Marcha actual: mostrada mediante un display siete segmentos. Se utilizará esta información para conocer si estamos en los límites inferiores o superiores de la caja de cambios.

Además, desde el departamento de motor, se pide la inclusión de un mando rotativo que permita al piloto cambiar el punto de mordida del embrague, para poder calibrar las salidas en parado.

#### Electrónica

- SoC de la batería: debida a la ausencia de alternador, es necesario saber cómo de cargada está la batería para su reemplazo por una completamente cargada.
- Watchdog: este indicador es más genérico que el resto, ya que le indica al piloto cuando hay algún sistema electrónico de control del coche se ha caído. De nuevo, el piloto podrá parar el coche en el caso de que este indicador se ilumine.
- Modulo de CAN-Bus: para la comunicación entre sistemas, el sistema de salpicadero ha de tener instalado un módulo de CAN-Bus con las especificaciones requeridas por el departamento. Estos requisitos de CAN se establecerán en el capítulo siguiente, cuando se explique el diseño elegido para la nueva temporada.

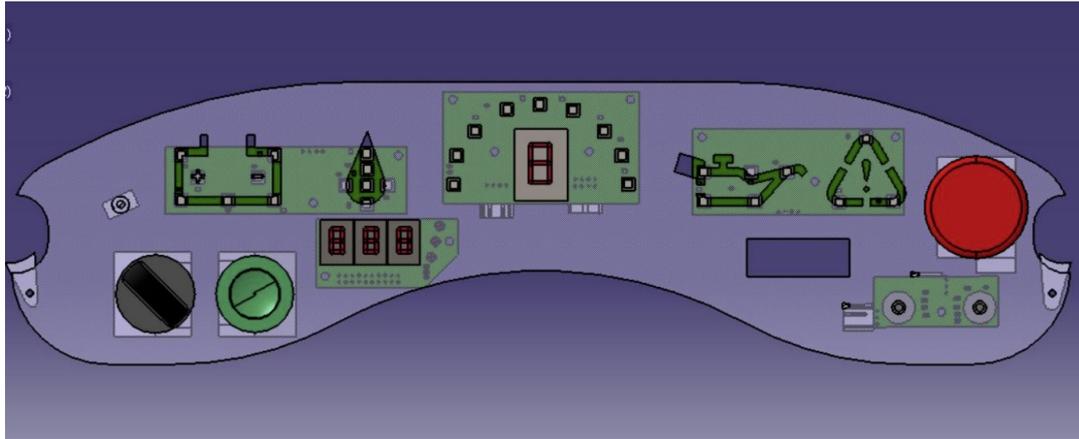
#### Ergonomía

El departamento de ergonomía, una vez se conocen todos los indicadores necesarios en el salpicadero, se encarga de disponer los indicadores de la mejor forma para que sean leídos por el piloto. En la Figura 2.4 se puede ver la disposición final de los indicadores diseñada por el departamento de ergonomía.

### 2.2.2 ART20E (Eléctrico)

En el caso del vehículo eléctrico, los requisitos sobre el sistema no vienen impuestos por el equipo, sino por la normativa que regula el acceso a las competiciones.

- IMD: indicador luminoso que indica al piloto si hay un fallo de aislamiento detectado por el sistema electrónico de seguridad IMD. En el caso de que este indicador se ilumine, se informa que hay alta tensión desde la batería al chasis del vehículo, por lo que el piloto deberá abandonar el vehículo sin tocar ninguna parte del chasis.

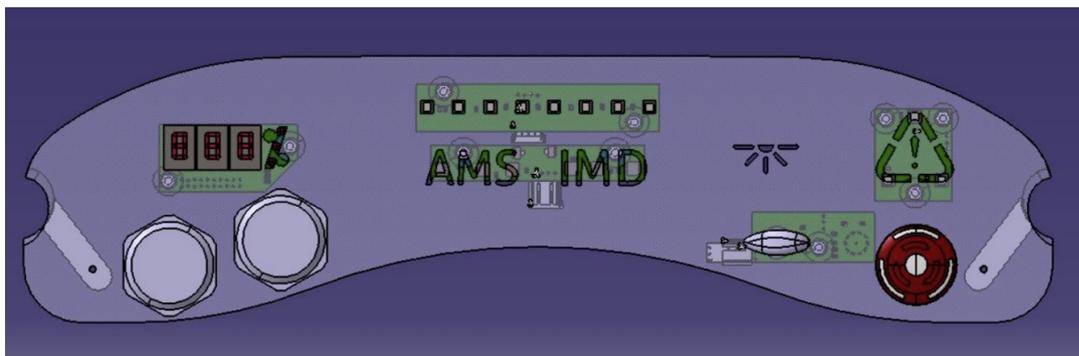


**Figura 2.4** Diseño del salpicadero del ART19C.

- AMS: indicador luminoso que indica al piloto si ha ocurrido algún fallo con la batería de alta tensión del vehículo.

### Ergonomía

Ocurre lo mismo en el coche eléctrico con el departamento de ergonomía, por lo que se adaptó el sistema electrónico al diseño 3D realizado por los compañeros de ergonomía. En la Figura 2.5 se puede ver el diseño realizado.



**Figura 2.5** Diseño del salpicadero del ART19E.

## 2.3 Herencia. Fallos encontrados

Como se indicó previamente, el sistema de salpicadero es heredado de la temporada 2018-19. Este sistema se divide en varias PCB distintas, la principal, denominada controladora, y las periféricas. De esta manera, la controladora es la misma para ambos coches, por lo que implementa las funcionalidades de estos, aumentando su complejidad. Como se aprecia en las figuras 2.1 y 2.2 las PCB periféricas se ubican en ranuras hechas con la forma del indicador deseado. Se pasa ahora a desglosar los fallos encontrados en el sistema de salpicadero, los cuales se verán corregidos en la nueva versión.

### 2.3.1 Fallos generales

A fallos generales se refiere a errores o condiciones no óptimas del sistema en general, tanto para el vehículo eléctrico como para el de combustión.

- Extrema complejidad en las PCB periféricas: estas están separadas las unas de las otras, por lo que la instalación mecánica de las mismas en cada ranura para cada indicador es extremadamente tediosa.
- Cableado interperiférico: al estar estas PCB separadas, y su conexión debe ser en serie, la única forma de conectarlas es mediante cables y conectores. Teniendo en cuenta el entorno en el que se encuentra el sistema, sometido a altas vibraciones y estrés mecánico constante, esto fue una fuente de fallos importante.
- Sobredimensionamiento de indicadores: los indicadores se realizaron de forma que tuviesen suficientes fuentes lumínicas como para que su visionado por el piloto fuese cómodo. Esta elección disparaba el consumo debido a los LEDs de cada PCB periférica.
- Elección de componentes sobredimensionados: en la PCB controladora se escogen ciertos componentes, en particular el DCDC de TracoPower, aumentando el gasto económico de manera innecesaria.
- Módulo interno de programación en controladora: la PCB controladora contaba con un módulo interno para poder cargar el programa al microcontrolador de una manera más sencilla. Con el paso de la temporada se vio que este módulo al completo era innecesario, ya que la programación de este tipo de microcontroladores se hace de una manera sencilla mediante el protocolo SPI.
- Serigrafía pobre: a la hora de hacer pruebas en laboratorio con las PCB resultó ser bastante difícil encontrar la señal deseada para su prueba.

### **2.3.2 Fallos en el sistema del ART19C**

Se pasa ahora con errores encontrados en el sistema del vehículo de combustión.

- Eliminación de display 7 segmentos innecesario: durante la temporada 2018-19 se implementaron 3 displays de 7 segmentos adicionales al central de la marcha para conocer temperatura exacta de aceite. Esto se vio que no era óptimo, sobre todo conociendo las limitaciones de la controladora a nivel de microcontrolador.

### **2.3.3 Fallos en el sistema del ART19E**

Por último se detallan los errores encontrados en el sistema del vehículo eléctrico relativos al sistema de salpicadero.

- Secuencia inicial correcta: de acuerdo con la normativa de la temporada 19-20 y 20-21, al iniciar el vehículo y, por ende, sus sistemas electrónicos de control y seguridad, los indicadores de AMS e IMD deben parpadear un número determinado de veces a una frecuencia concreta para su correcta verificación.
- Recepción de señales de AMS e IMD "rules compliant": el cambio en la normativa de una temporada a otra obligaba a detectar el fallo de "cable cortado" al llegar la señal al sistema de salpicadero, algo que no hacía la versión de la temporada 2018-19.

## 3 Diseño actual

---

Una vez se conoce el contexto del sistema y todos los cambios necesarios, se procede a detallar el proceso de diseño y toma de decisiones en el nuevo sistema de salpicadero para los ART20.

Al comienzo de la temporada se plantearon varias alternativas para el enfoque general del sistema, las cuales se explicarán detalladamente en el presente capítulo. Antes de comenzar, cabe destacar que **el contexto de la temporada 2019-20 se envuelve en la pandemia del COVID-19 y que obligó a la cancelación de las competiciones por parte de la organización de Formula Student, posponiendo las mismas al verano de 2021.**

### 3.1 Opciones

Antes de proceder con el diseño, se tuvieron en cuenta diversas opciones para la arquitectura del sistema. Opciones comerciales, diseño integrado o continuista fueron algunas de las opciones que se tuvieron en mente antes de comenzar.

En los siguientes apartados se verán todas las opciones consideradas, así como comparativas entre ellas y la opción final elegida.

#### 3.1.1 Opciones comerciales

En el mercado existen diferentes opciones para la implementación del sistema de salpicadero que pueden llegar a cumplir los requisitos mínimos del sistema. Evidentemente esto es monetariamente peor que hacer un diseño propio, pero esta opción fue barajada teniendo en cuenta el ahorro temporal que supondría para el equipo, algo que en un mundo como el deporte de competición es muy codiciado.

A continuación, se definirán las opciones más viables encontradas en el mercado, desglosando sus ventajas e inconvenientes, aunque antes de comenzar cabe destacar que todos los sistemas encontrados son orientados a vehículos de competición.

##### **AiM MXG 1.2 Strada**

- Pantalla TFT.
- 800x480 píxeles de resolución.
- 2 conexiones para CAN-Bus. CAN-Bus configurable en velocidad y formato (regular o extendido)
- 8 entradas analógicas para sensores a 1000Hz de velocidad de muestreo.
- 1 entrada digital, orientada solo a RPM.
- Consumo máximo de 400mA.

- Sensor de luz ambiente. Contraste y brillo personalizable según condiciones.
- Puertos disponibles para módulos externos: GPS, controladora Lambda, cámara on-board (solo para módulos del mismo fabricante).
- Protección IP65 (hermético al polvo y protegido contra agua proyectada, lluvia).
- 9 Indicadores luminosos programables, incluida una tira para revoluciones.
- Peso: 950g.

Precio: 1200€



Figura 3.1 AiM MXG 1.2 Strada.

### AiM GS-Dash

- Pantalla LCD.
- 268x128 píxeles de resolución.
- Conector sencillo de interfaz de CAN.
- Consumo máximo de 100mA.
- Protección IP65.
- 5 indicadores luminosos programables, incluido una tira de 5 LEDs para revoluciones.
- Color de fondo y brillo configurables.
- Peso: 460g

Precio: 500€



Figura 3.2 AiM GS-Dash.

**AEM Electronics CD-5**

- Pantalla TFT.
- 800x480 píxeles de resolución.
- Interfaz de CAN programable.
- Indicador luminoso de 7 LEDs para revoluciones.
- Pantalla paginada por indicadores.
- Protección IP66.
- Dos indicadores luminosos para alertas.
- Software de programación incluido, con interfaz gráfica de previsualización.
- Peso: 330g

Precio: bajo demanda.



**Figura 3.3** AEM Electronics CD-5.

Por último, en la Tabla 3.3 se encuentra una comparativa de las 3 opciones de sistemas en el mercado con sus características más determinantes para el equipo.

**Tabla 3.1** Comparativa entre pantallas.

	<b>MXG 1.2 Strada</b>	<b>GS-Dash</b>	<b>CD-5</b>
Pantalla	TFT	LCD	TFT
Resolución	800x480	268x128	800x480
Interfaz CAN	Si (configurable)	Si (fija)	Si (configurable)
Indicadores	9	4	2
Consumo	400mA	100mA	N/A
Protección	IP65	IP65	IP66
Peso	950g	460g	330g
Precio	1200€	500€	N/A

### 3.1.2 Opciones de diseño propio

Una vez estudiadas las opciones en el mercado, se procede a estudiar las diferentes arquitecturas posibles que se pueden aplicar al sistema. Cabe destacar que, ya de entrada, una solución completamente diseñada por el equipo nos da la ventaja de poder aplicar un diseño a cada problema que requiere el sistema, sin necesidad de ataduras a adquirir más características que no serían de interés para esta aplicación.

Se procede a describir las posibles arquitecturas de interés para la aplicación.

#### Diseño continuista

Una de las opciones barajadas fue la de mantener el diseño del sistema de salpicadero para los ART19, reparando los errores vistos durante esa temporada. Esto conllevaría un ahorro de tiempo importante, puesto que las PCB tendrían un diseño muy parecido al anterior.

Sin embargo, esto no arreglaría uno de los principales problemas que se tuvieron durante la temporada anterior, y es la fiabilidad del sistema. Como se comentó en el capítulo anterior, uno de los problemas principales fue la poca resistencia ante vibraciones que tenía el sistema de salpicadero, sobre todo en la interconexión de PCB de indicadores en serie, debido fundamentalmente al complejo problema de cableado y elección de conectores.

- Ventajas
  - Rapidez de diseño.
  - Muy bajo coste.
  - Bajo tiempo de fabricación e implementación.
- Inconvenientes
  - No se soluciona el problema de la fiabilidad.
  - Poca capacidad de desarrollo del sistema para el futuro.
  - Instalación compleja y tediosa.

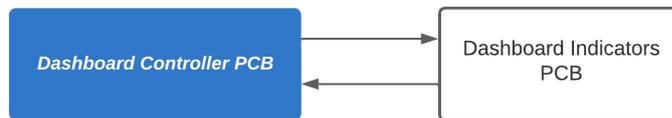
#### Integración completa

La siguiente opción fue recopilar todos los circuitos del sistema de salpicadero, es decir, todos los circuitos de la controladora y de las PCB de indicadores, e integrarlos en una única PCB. Este diseño se realizaría dejando en una sola cara de la PCB solo indicadores luminosos, cuadrando los mismos en cada ranura de indicador, y en la otra cara dejar la electrónica de control (actualmente la PCB controladora).

- Ventajas
  - Soluciona el problema de la fiabilidad en los indicadores.
  - Bajo coste.
  - Instalación sencilla y rápida, un solo conector hacia el exterior.
  - Posibilidad de desarrollo del sistema. Posible inclusión de pantalla TFT para los indicadores no críticos (todos menos los indicadores AMS e IMD en el vehículo eléctrico).
- Inconvenientes
  - Aumenta el tiempo de diseño y desarrollo.
  - Aumenta el esfuerzo mecánico sobre toda la electrónica, requiere de repuestos en caso de rotura o fallo.
  - Dificultad añadida de hacer coincidir los indicadores luminosos con su ranura.

### Integración moderada

Por último, se baraja la opción a medio camino entre las dos anteriores, integrando los indicadores en una sola PCB y dejando la controladora en otra PCB. Para hacernos una idea, en la Figura 3.4 se puede ver el esquema simplificado del sistema.



**Figura 3.4** Esquema simplificado de la integración moderada.

- Ventajas
  - Soluciona el problema de fiabilidad en los indicadores.
  - Permite un desarrollo en paralelo de controladora e indicadores.
  - Instalación rápida.
  - El esfuerzo mecánico es solo sobre una PCB pasiva.
  - Disminuye el tiempo de diseño.
  - Repuestos más baratos.
- Inconvenientes
  - No permite demasiado desarrollo del sistema.
  - Aumenta el coste del sistema.

### 3.1.3 Resumen

Se han visto todas las opciones que se barajan para la implementación del sistema de salpicadero, ahora se recopilarán todas las opciones y explicar los motivos de descartar o elegir la opción final.

#### Pantalla comercial

Se sabe que las opciones del mercado son más fáciles de implementar, más rápido e incluso más fiables. Sin embargo, estas ventajas se ven remitidas por el alto precio que se ha de pagar para cualquiera de las opciones de mercado. Para tener un orden de magnitud, el sistema de salpicadero de los ART19 tuvo un coste económico menor de 450€. Esto conlleva que la implementación comercial del sistema duplicaría el coste económico, algo que para un equipo como ARUS no es asumible.

**Por tanto, la opción de pantalla comercial se descarta por motivos económicos, se pasa a elegir alguna de las opciones de diseño propio**

#### Diseño propio: diseño continuista

Tal y como se ha visto en el apartado anterior, el diseño continuista, pese a ser más fácil de implementar y mucho más rápido que los otros dos, no soluciona el problema de la fiabilidad. Teniendo en cuenta que el contexto de la temporada 2019-20 se envuelve en que los coches tienen como objetivo mejorar la fiabilidad en general, **esta opción se ve descartada por este motivo.**

### Diseño propio: integración completa

La integración completa es una opción bastante atractiva en vistas a los objetivos de la temporada, aportando además la posibilidad de desarrollo del sistema para futuras temporadas. Sin embargo, la poca experiencia de los miembros del equipo en sistemas basados en PCB tan grandes y la necesidad de un proceso de diseño en un corto espacio de tiempo **hace que se descarte esta opción, al menos hasta que no se tenga un sistema de salpicadero fiable.**

### Diseño propio: integración moderada

Finalmente, se toma la decisión de implementar la opción de diseño propio basada en una integración moderada del sistema de salpicadero, por lo que en los próximos apartados se verá todo el proceso de diseño de esta opción.

## 3.2 Diseño e implementación del sistema. Estudios previos

En esta sección se explicará todo el proceso de diseño del nuevo sistema de salpicadero, comenzando por el diagrama de bloques del sistema, pasando por su integración en circuito y terminando por el ruteo de las PCB y del desarrollo del código del microcontrolador.

### 3.2.1 Diagrama de bloques general

En primer lugar, se introducen los diagramas de bloques del sistema de salpicadero del ART20C (Figura 3.5) y del ART20E (Figura 3.6). Como se puede apreciar, la intención es hacer solo

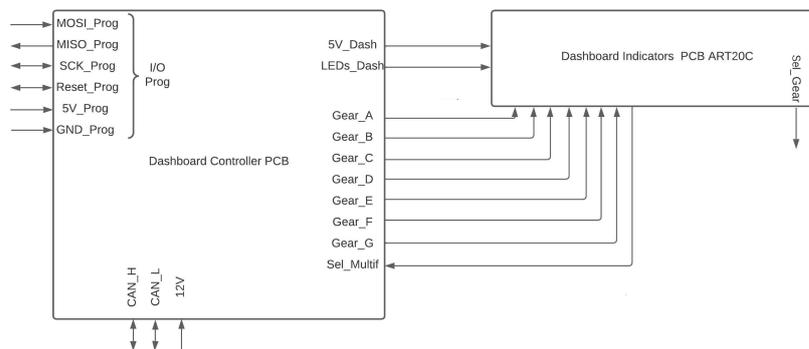


Figura 3.5 Diagrama de bloques del salpicadero para el ART20C.

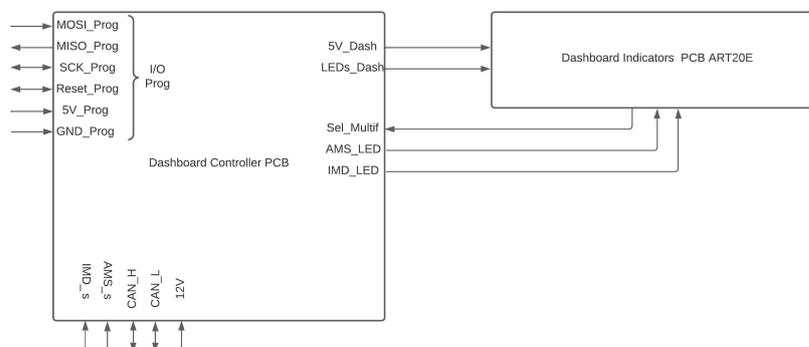


Figura 3.6 Diagrama de bloques del salpicadero para el ART20E.

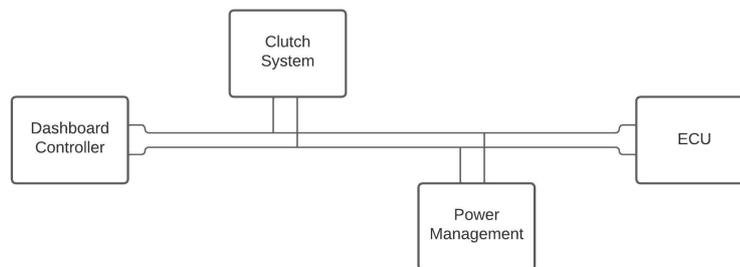
una controladora y variar en cada coche la PCB de indicadores. Esto aumenta ligeramente la complejidad de la controladora, en contraposición, las PCB de indicadores son mucho más sencillas. A continuación, se explicarán las señales de los diagramas de bloques.

- ART20C
  - Pines programación: MISO, MOSI, SCK, RESET, 5V y GND. Son pines del protocolo SPI, se utilizan para cargar el código para el microcontrolador del sistema.
  - Gear\_X: pines para el display 7 segmentos de la PCB de indicadores.
  - Sel\_Multif: entrada analógica del selector multifunción del salpicadero.
  - LEDs\_Dash: pin para la trama de programación de los LEDs de la PCB de indicadores.
  - CAN\_H y CAN\_L: pines del protocolo de CAN Bus.
  - 5V\_Dash: alimentación para la PCB de indicadores.
  - 12V: alimentación del sistema.
- ART20E (solo se explican los no repetidos)
  - IMD\_LED y AMS\_LED: señales de activación de indicadores de IMD y AMS.
  - IMD\_s y AMS\_s: señales de entrada provenientes de los sistemas IMD y AMS.

### 3.2.2 Entorno del sistema. Fuentes de datos (CAN Bus)

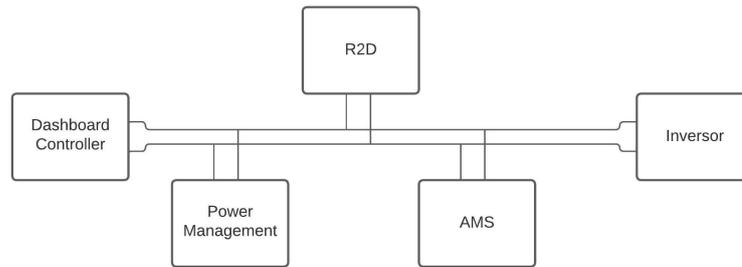
Como se vio en los capítulos y secciones anteriores, el sistema de salpicadero recoge información del resto de sistemas electrónicos del vehículo para mostrar la información relevante al piloto. Pero, ¿de dónde provienen esos datos? Todos los sistemas electrónicos del coche están conectados a una red de CAN-Bus, de ahí la necesidad de módulo CAN en la controladora.

En la Figura 3.7 y la Figura 3.8 se puede ver un esquema de los ART20 con los sistemas electrónicos a los que está conectado el sistema de salpicadero mediante CAN-Bus.



**Figura 3.7** Mapa de CAN del ART20C.

Ahora se desglosarán los datos a mostrar en cada coche para tener todo establecido antes de proceder con el diseño completo del sistema.

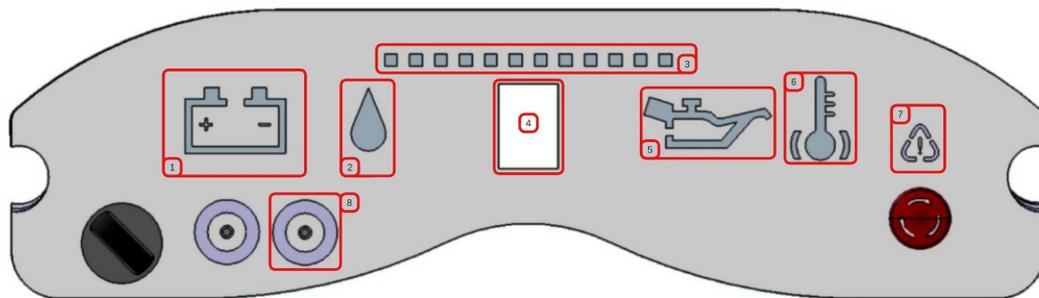


**Figura 3.8** Mapa de CAN del ART20E.

### 3.2.3 Interfaz visual

#### ART20C

Antes de ver los datos dispuestos para el piloto, se mostrarán los indicadores necesarios dados por el departamento de ergonomía.



**Figura 3.9** Salpicadero del ART20C.

En la Tabla 3.3 se puede observar la función que tiene cada elemento en el salpicadero del ART20C. El indicador más diferente es el denominado revoluciones/panel multifunción. Este indicador aprovecha las 10 posiciones del mando que lee la controladora del salpicadero para mostrarle al piloto un dato distinto según su posición. En este indicador puede haber datos incluso redundantes, aquellos de los que ya existe un indicador para ese dato en concreto. Sin embargo, si se selecciona su opción en el mando rotativo, el dato es más preciso, llegando incluso a tener una resolución de una unidad (por ejemplo, si se selecciona una temperatura, daría una resolución de 1°C)

El panel multifunción puede mostrar los siguientes datos:

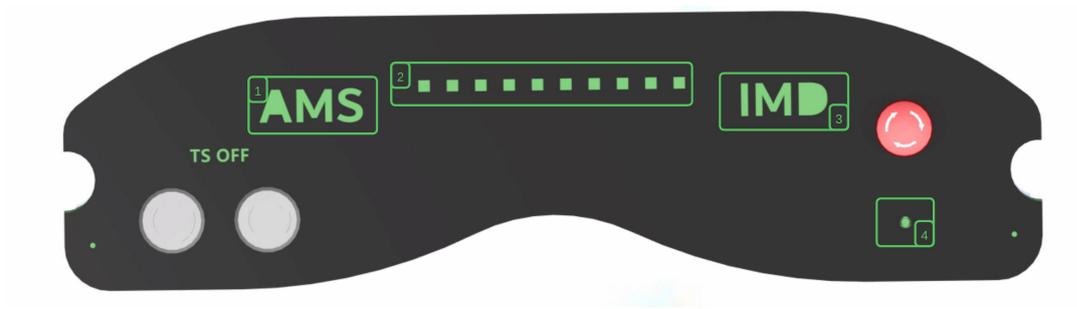
1. Revoluciones del motor
2. Temperatura de aceite
3. Temperatura de agua
4. Temperatura del cockpit
5. Lambda
6. Temperatura de aire
7. Presión de aceite
8. Punto de mordida

**Tabla 3.2** Elementos y funciones del salpicadero del ART20C.

Elemento	Funcion
1	Estado de la batería
2	Temperatura del líquido de refrigeración
3	Revoluciones/Panel multifunción
4	Marcha
5	Temperatura de aceite
6	Presión de aceite
7	Alerta general, orden al piloto de apagado del motor
8	Selector del panel multifunción

**ART20E**

Al igual que se ha hecho con el ART20E en la Figura 3.10 se muestra el salpicadero ART20E proporcionado por el departamento de ergonomía.



**Figura 3.10** Salpicadero del ART20E.

**Tabla 3.3** Elementos y funciones del salpicadero del ART20E.

Elemento	Funcion
1	Indicador AMS
2	Panel multifunción
3	Indicador IMD
4	Selector del panel multifunción

El panel multifunción puede mostrar los siguientes datos:

1. SoC de la batería
2. Revoluciones del motor
3. Velocidad del vehículo
4. Corriente demandada
5. Watchdog de sistemas electrónicos
6. Estado de los relés de aislamiento

- 7. Estado de Ready To Drive
- 8. Modo de conducción

### 3.3 Diseño e implementación del sistema. Hardware

Para comenzar con el proceso de diseño, se separa el sistema en 2 bloques, la PCB controladora (Dashboard Controller PCB) y la PCB de indicadores (Indicators PCB). En primer lugar se describirá el proceso de diseño hardware del sistema de salpicadero para los dos coches. El diseño hardware completo se realiza en el software Altium Designer®, tanto diseño de esquemático como ruteo de las PCB.

#### 3.3.1 Dashboard Controller PCB. Esquemático

La controladora, como se ha repetido varias veces a lo largo del presente documento, tiene que aportar las funcionalidades para los sistemas de salpicadero de los dos coches. En primer lugar, se describirá la PCB mediante bloques, agrupados en un esquemático denominado **Top.SchDoc**. Como se puede apreciar en la Figura 3.11, la controladora está dividida en 7 bloques distintos,

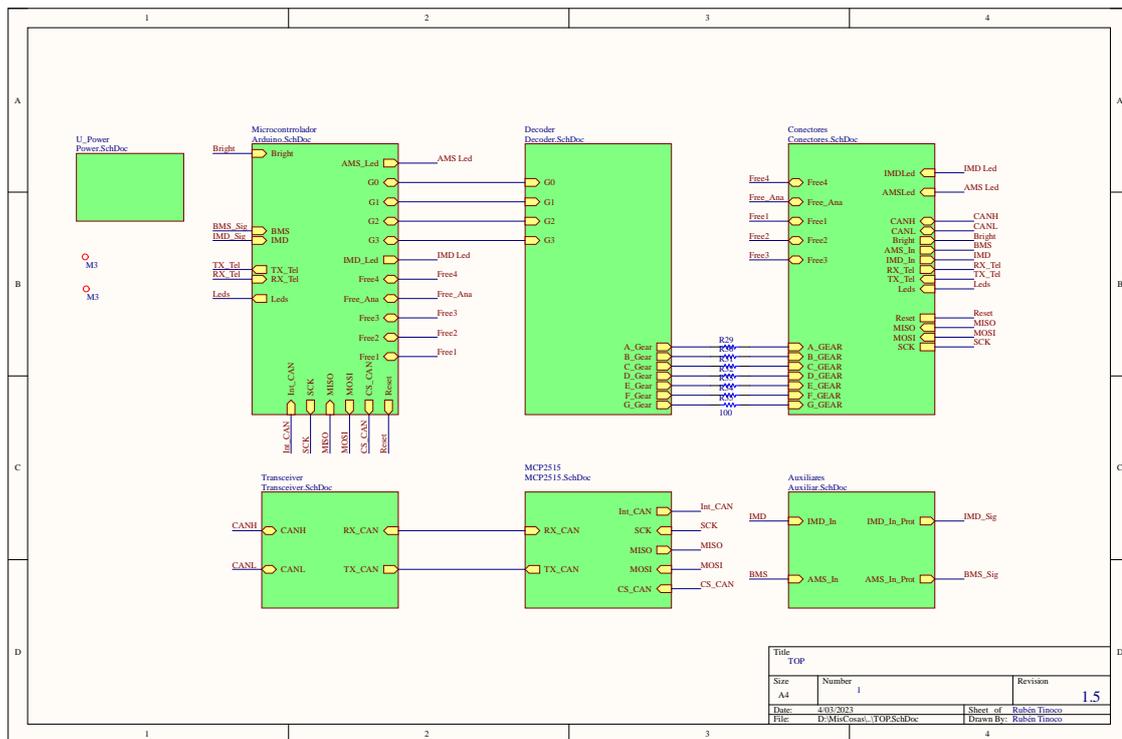
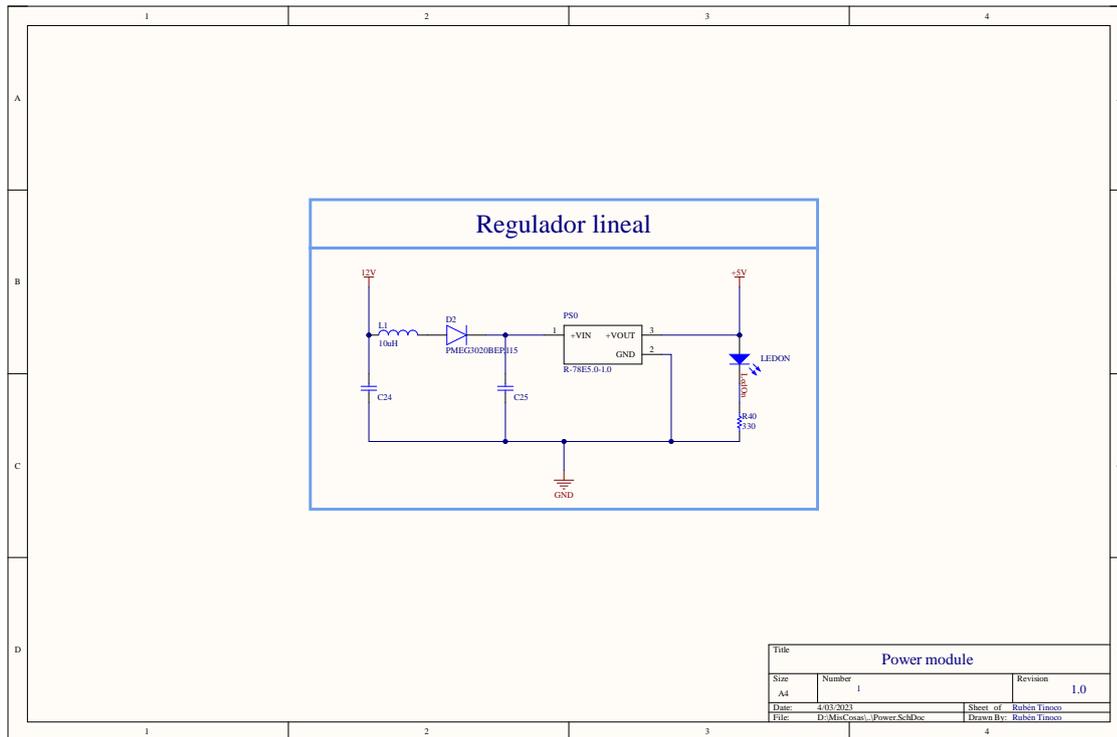


Figura 3.11 Nivel Top de Dashboard Controller PCB.

separando los circuitos de la PCB para una mejor comprensión y lectura de esquemático. Se procede ahora a explicar cada uno de los bloques y establecer cada una de las decisiones de diseño que se han tomado.

## Unidad de alimentación



**Figura 3.12** Unidad de alimentación de la PCB.

En contraposición con la elección de alimentación de la placa del año anterior, esta temporada se elige un regulador lineal que hará que la PCB disminuya en tamaño considerablemente. El regulador lineal elegido es el **R-78E5.0-1.0**, que aporta a su salida una tensión de 5V (tensión de alimentación del resto de circuitos integrados de la PCB) y una corriente máxima de 1A. El resto de componentes en este módulo han sido elegidos siguiendo la recomendación del fabricante, salvo el diodo de entrada, que se encarga de evitar que la corriente en momentos determinados salga de la PCB.

### Unidad del microcontrolador

Para mantener códigos desarrollados en la temporada anterior se escogió exactamente el mismo microcontrolador, el **ATMega328P-AUR**, ya que sigue cumpliendo con las especificaciones mínimas del sistema, tanto en velocidad de ejecución de código como en número de pines mínimos. Esto último, de hecho, para este año se ha visto reducido, ya que se necesitan menos señales de salida desde la controladora hacia el exterior.

Además del microcontrolador, esta unidad contiene un botón de reset manual, para, durante las pruebas de laboratorio, poder debuggear la PCB de una manera más sencilla. (Figura 3.13).

### Módulo de CAN

En este caso es necesario ver 2 bloques distintos del nivel Top, el bloque **MCP2515.SchDoc** (Figura 3.14) que incluye el driver de CAN y el bloque **Transceiver.SchDoc** (Figura 3.15) que acondiciona el transceptor de CAN a UART. Estos bloques en conjunto forman el módulo de CAN, necesario para la intercomunicación del sistema de salpicadero con el resto de sistemas electrónicos.

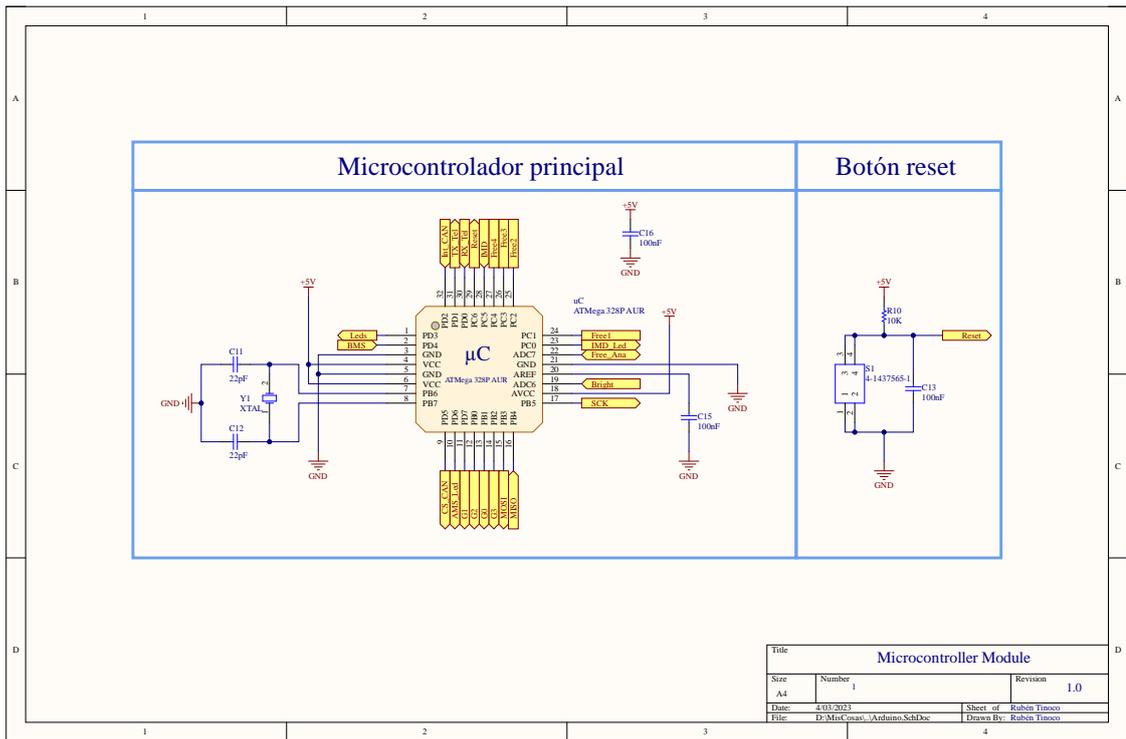


Figura 3.13 Unidad del microcontrolador de la PCB.

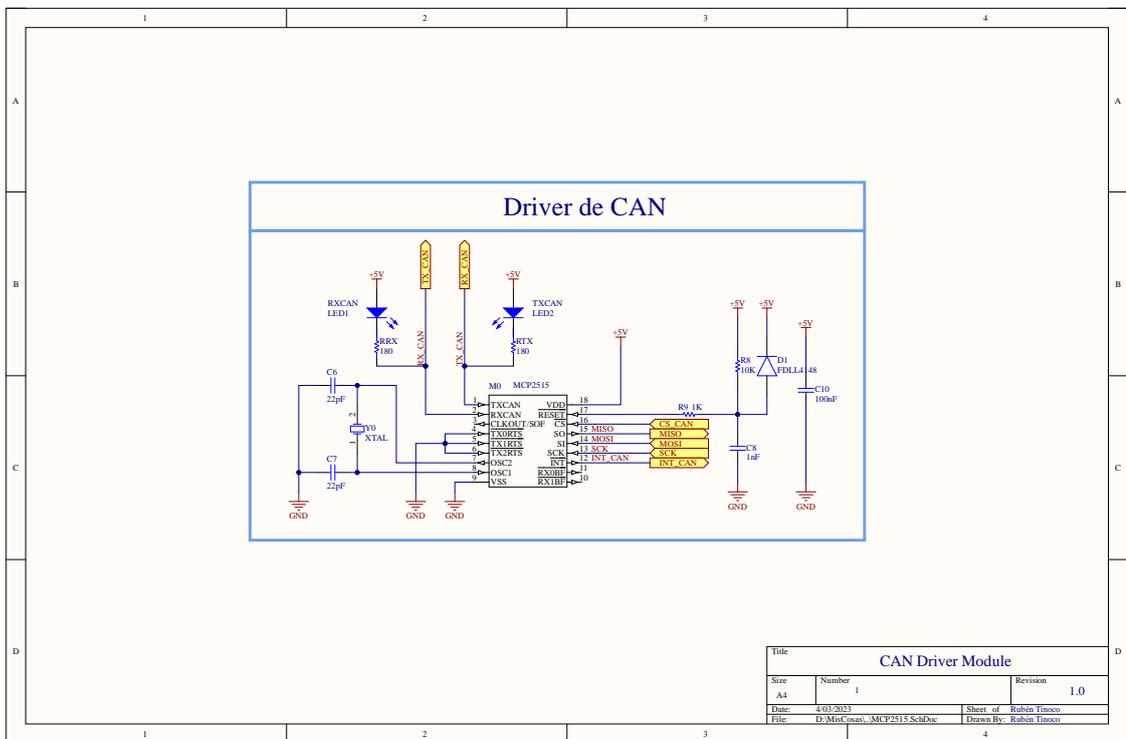


Figura 3.14 Unidad del driver de CAN.

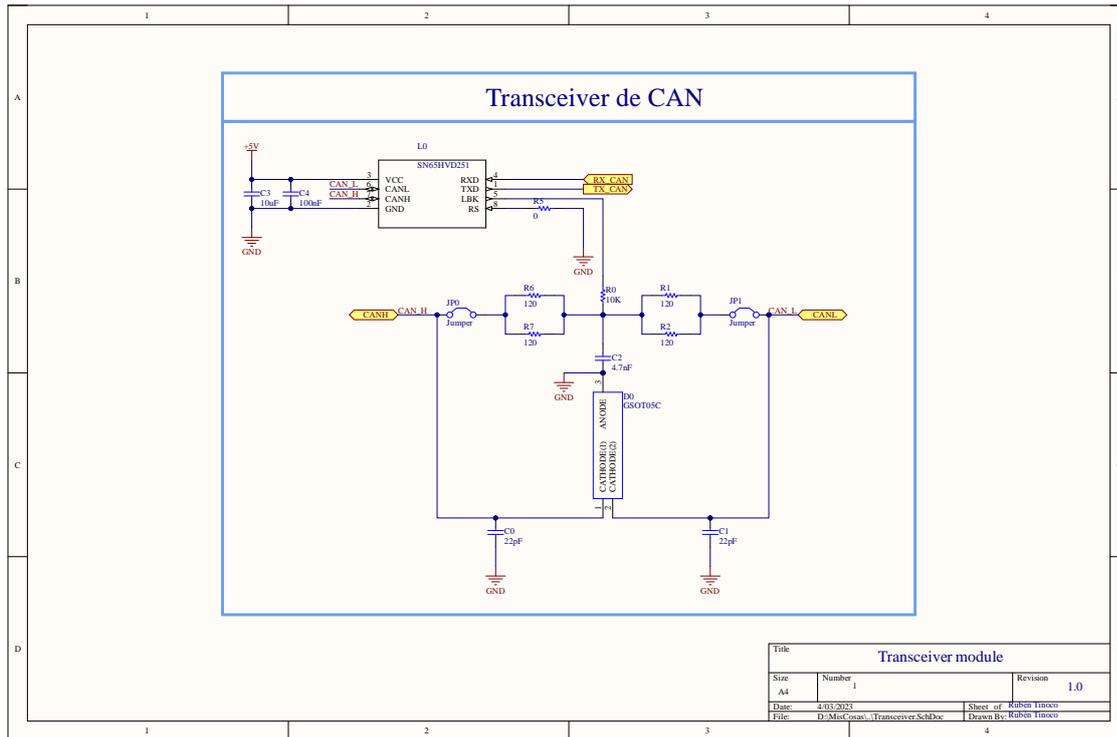


Figura 3.15 Unidad del transceiver de CAN a UART.

Por explicar un poco los esquemáticos, el módulo del driver de CAN se ha aclimatado según las recomendaciones del datasheet y el reloj se ha escogido el mismo que el elegido en el módulo de microcontrolador.

El transceiver también se ha aclimatado según recomendaciones del datasheet, teniendo en cuenta que el pin RS (configurador de slew-rate) se pone con una huella de resistencia 0805 a tierra para poder configurarlo a posteriori, durante las pruebas de laboratorio. El resto de elementos se instalan debido a la necesidad de resistencia terminadora del CAN Bus en el sistema de salpicadero.

**Módulo decoder**

Para reducir el número de pines usados por el microcontrolador y gestionar más fácilmente el display de 7 segmentos, se implementa un decoder de BCD a 7 segmentos. Tal y como se puede ver en la Figura 3.16 el decoder elegido es el **MC14511BDWR2G**, por razones de stock y de tensión de alimentación.

**Módulo auxiliar**

El módulo auxiliar (Figura 3.18) está compuesto por dos optoacopladores, los cuales protegen al sistema de posibles fallos de los sistemas AMS e IMD cuando pasan su señal para encendido de sus indicadores. Este módulo, una vez se tiene la PCB, no tiene por qué ser soldado cuando esta sea para el vehículo de combustión, ya que éste no tiene estos sistemas. Las señales de salida de estos optoacopladores se pasan directamente al microcontrolador para ser procesadas.

**Módulo de interfaz de salida**

Por último, el módulo de interfaz de salida (Figura 3.18), es un esquemático formado por un conector SUB-D de 25 pines para conectar la PCB al resto de sistemas. En este mismo esquemático, se puede observar cómo se han instalado ferritas para "limpiar" las señales de entrada y salida del sistema,

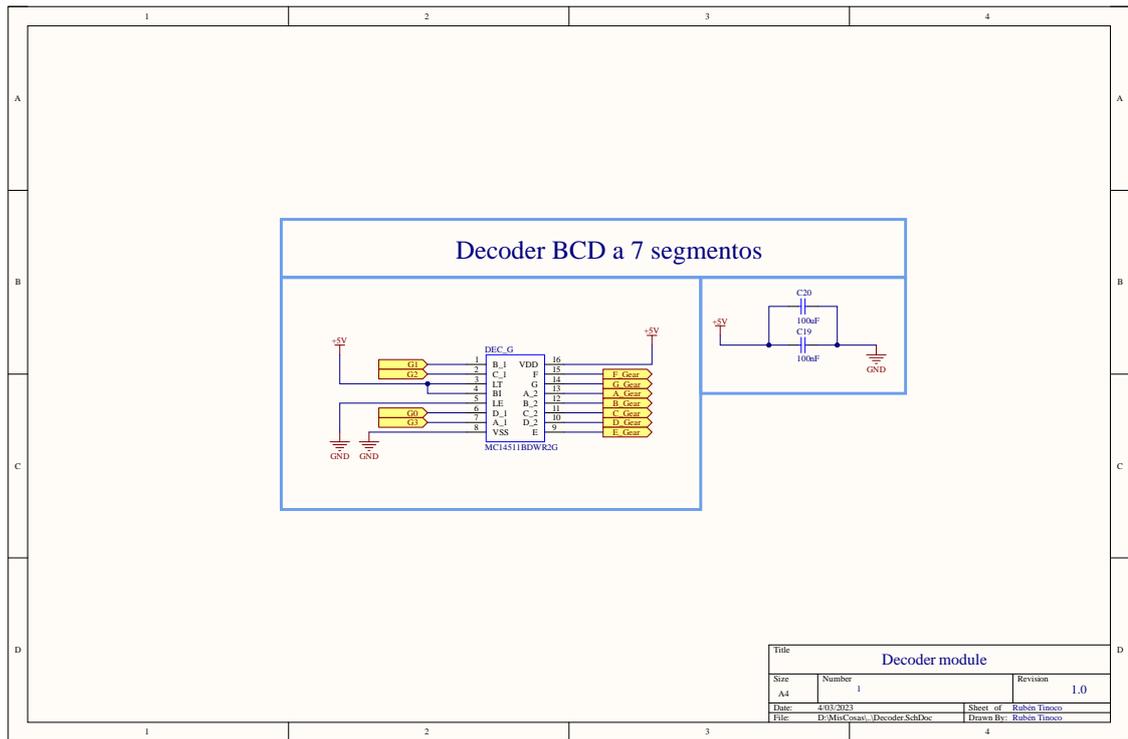


Figura 3.16 Unidad del decoder BCD - 7 Segmentos.

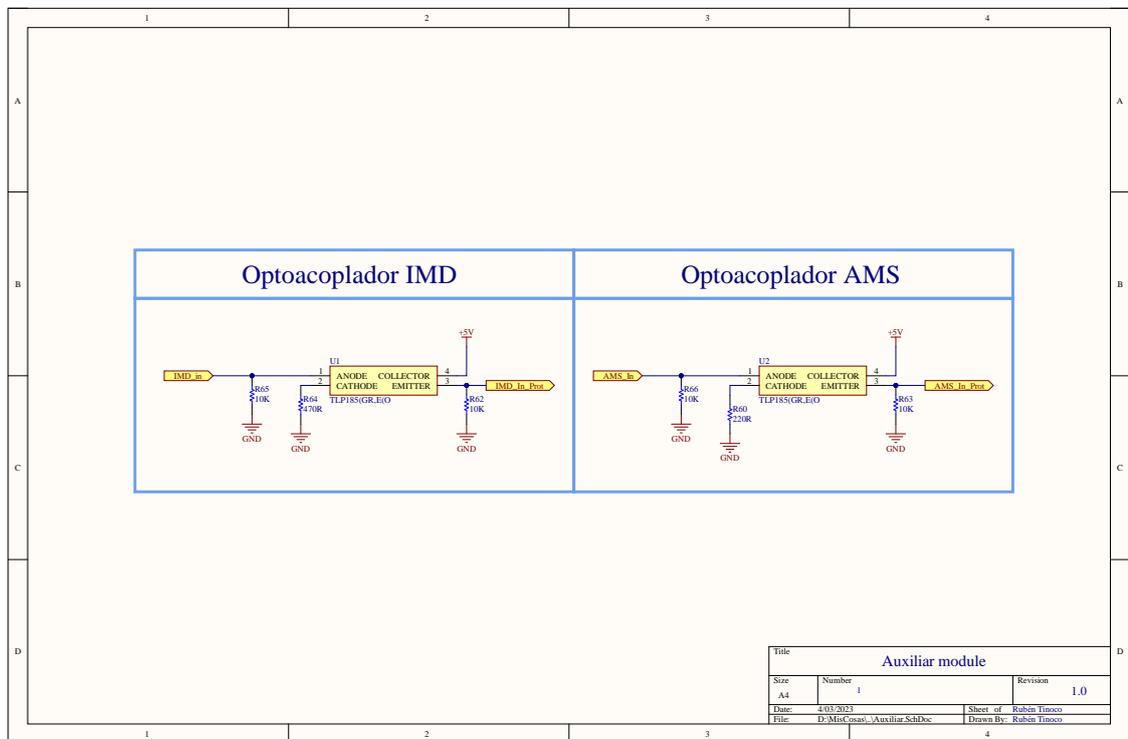


Figura 3.17 Unidad auxiliar.

salvo aquellas que sean de alta velocidad, como las de los protocolos de comunicación (CAN-Bus o UART para el puerto serie).

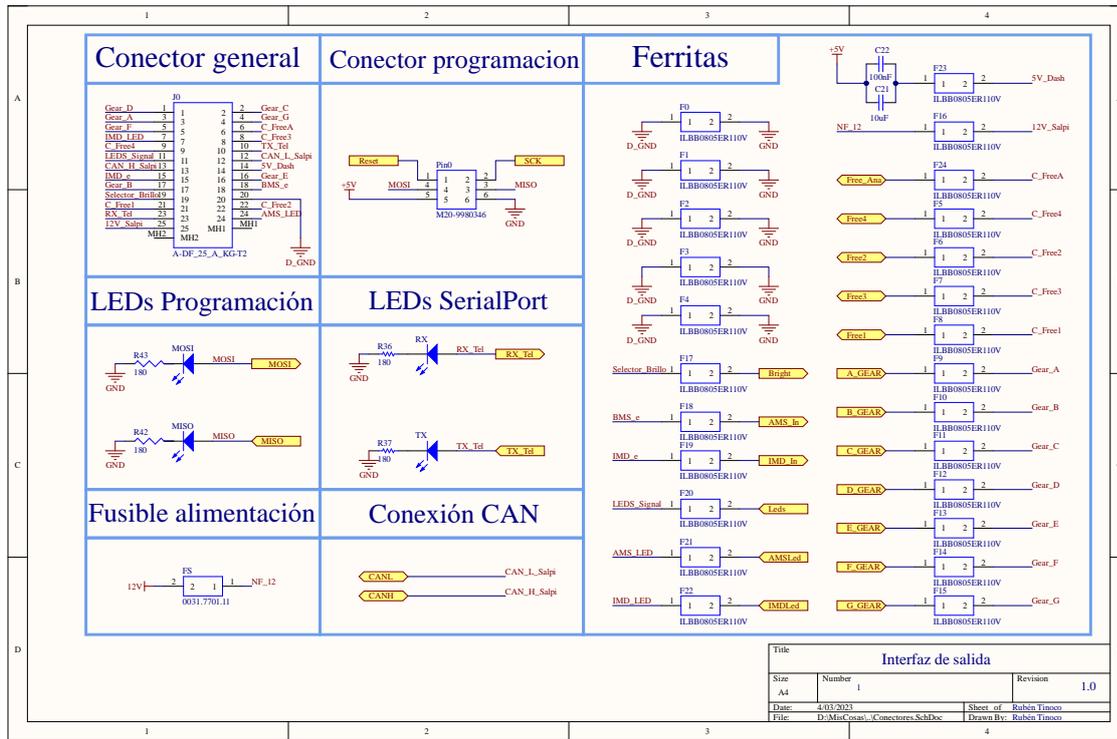


Figura 3.18 Interfaz de salida.

3.3.2 Dashboard Controller PCB. Rutado

En la Figura 3.19 (TOP) y en la Figura 3.20 (BOTTOM), se puede ver el ruteo de la PCB de salpicadero. Para una mayor limpieza de señal en general, se tomó la decisión de separar el plano de tierra en 2, uno "sucio" y otro "limpio", cuya diferencia es que el limpio está detrás de las 5 ferritas que se pueden ver en el esquemático de la Figura 3.18.

Por último, en la Figura 3.21 y la Figura 3.22 se puede encontrar el renderizado 3D de la PCB.

3.3.3 Dashboard Indicators PCB. Esquemático

Dado que los esquemáticos de las PCB de indicadores son muy sencillos (apenas son algunos LEDs y los selectores), se verán esquemáticos resúmenes sobre lo que se cree más importante.

En la Figura 3.23 se puede ver el ejemplo de implementación de un selector rotativo. En el caso del coche de combustión, el sistema de salpicadero tiene dos mandos, uno para el panel multifunción y otro para el punto de mordida del embrague. En el caso del vehículo eléctrico, solo tiene un selector, para el panel multifunción.

En la Figura 3.24 se puede ver como se ha implementado el indicador de AMS e IMD. Como se explicó anteriormente estos indicadores pertenecen a una red de señales críticas, de las cuales se tiene que detectar "cable cortado" en el caso en el que ocurra. De ahí que se haya utilizado un MOS-P, obligando a los sistemas AMS e IMD a mantener nivel alto a su salida en el caso de no fallo.

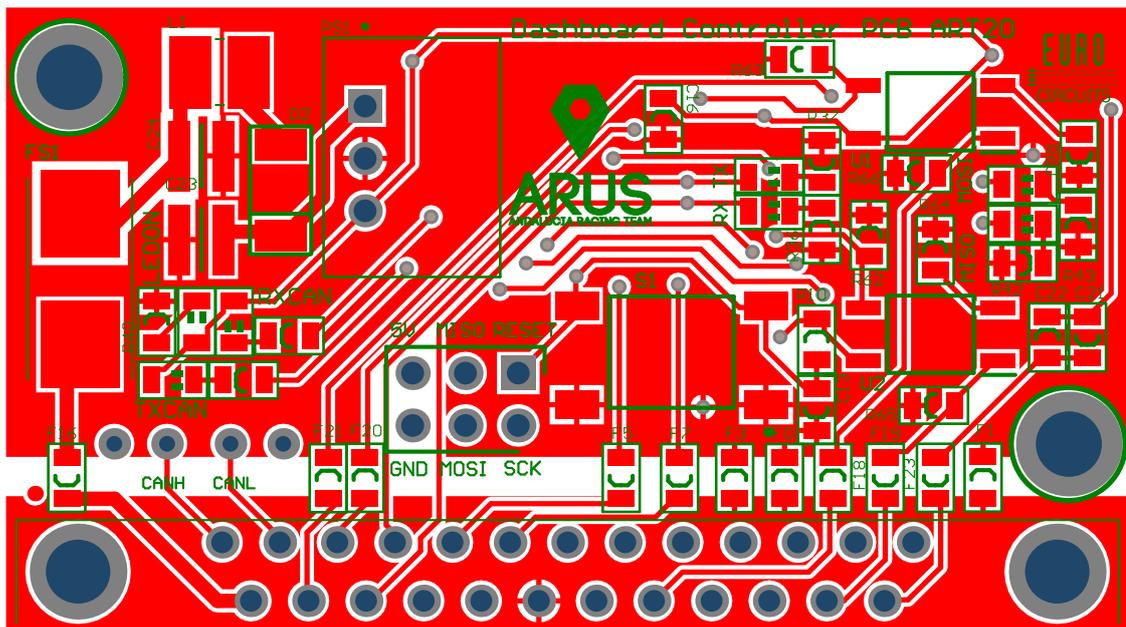


Figura 3.19 Rutado de la cara TOP de la PCB controladora.

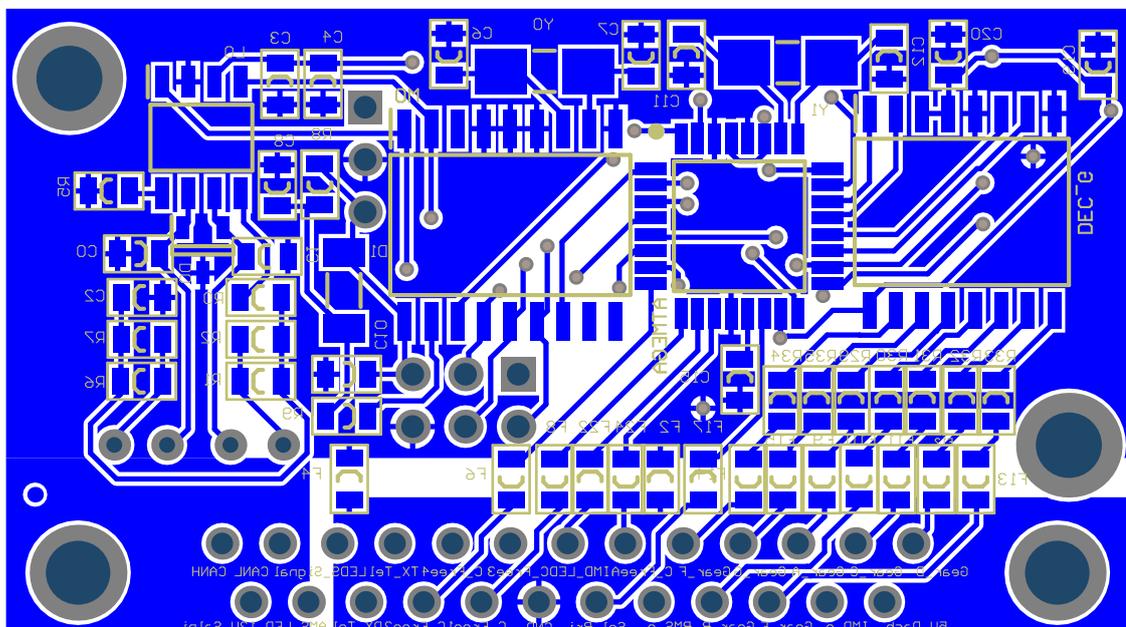


Figura 3.20 Rutado de la cara BOTTOM de la PCB controladora.

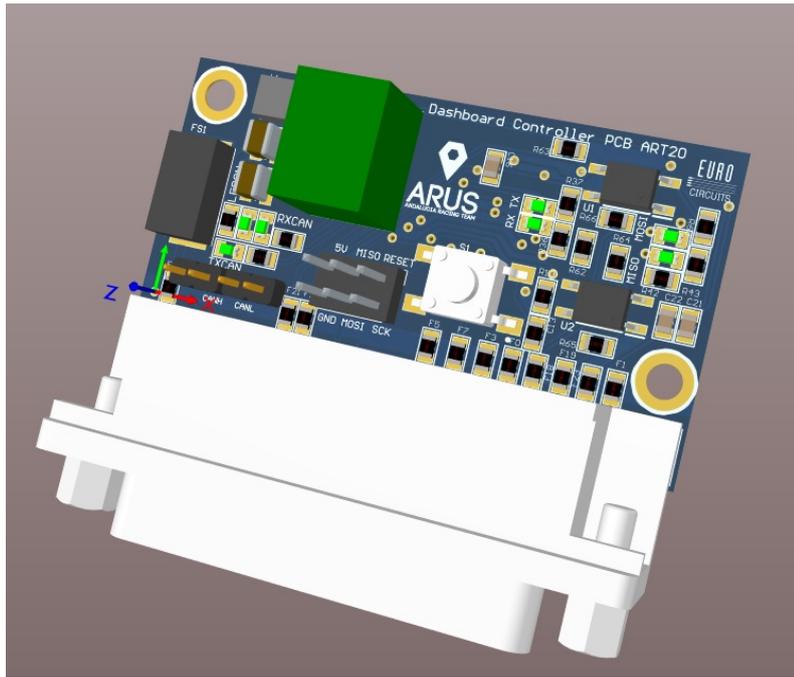


Figura 3.21 Parte frontal de la PCB controladora.

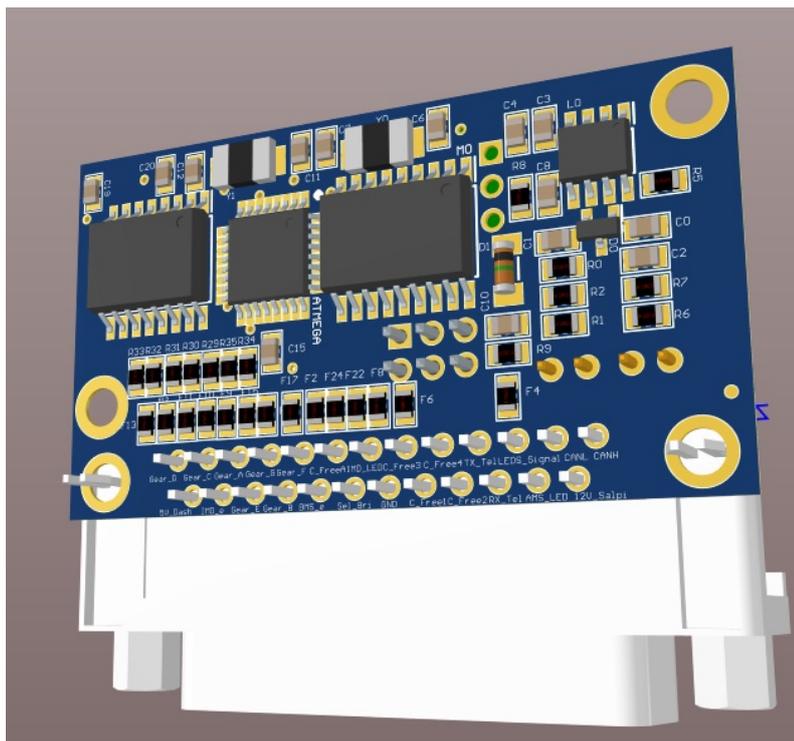


Figura 3.22 Parte trasera de la PCB controladora.

### 3.3.4 Dashboard Indicators PCB. Rutado

El proceso de rutado de las PCB de indicadores tiene una complicación añadida al resto de PCBs del monoplaça. El departamento de ergonomía proporciona el diseño de la parte mecánica del salpicadero, para luego hacer coincidir las perforaciones con las torretas. De esta forma, al fabricar la PCB, esta tendrá perforaciones que se usarán para anclar mecánicamente la PCB al chasis, sin

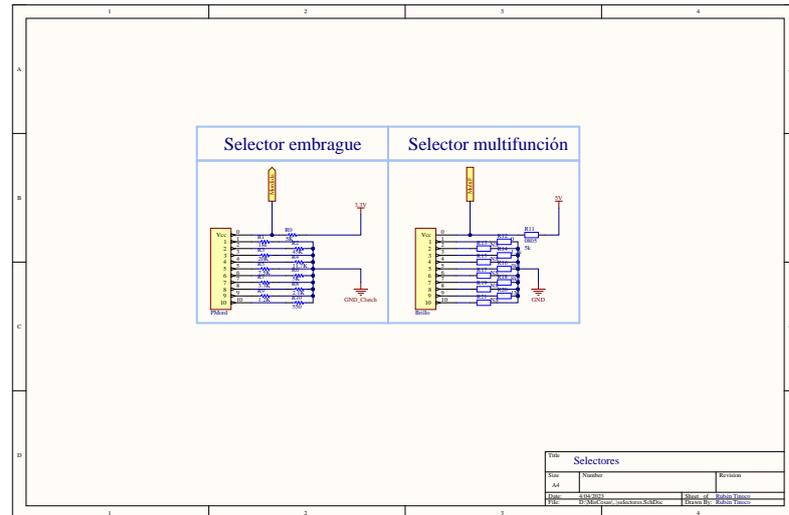


Figura 3.23 Esquemático de selectores rotativos.

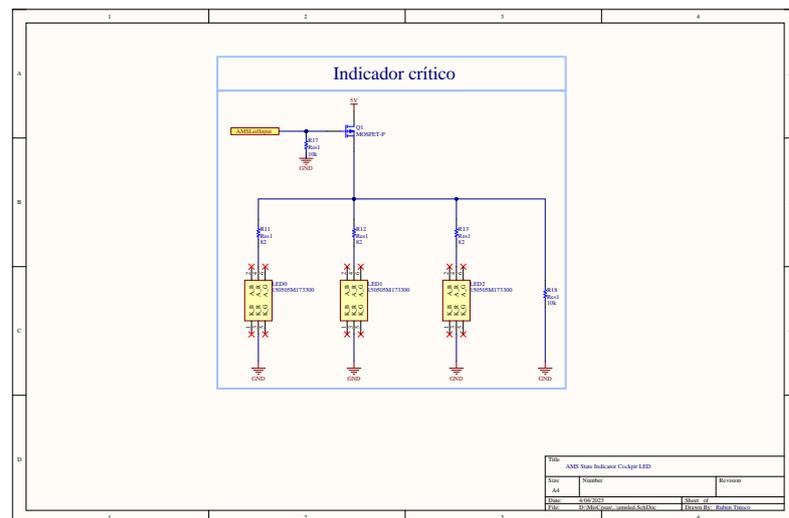


Figura 3.24 Esquemático del indicador crítico.

necesidad de sujeciones basadas en pegamento o cinta adhesiva.

En la Figura 3.25 se puede ver un explosionado del salpicadero del monoplaza de combustión, donde se puede ver gráficamente como se ha hecho coincidir la PCB con el resto de componentes mecánicos.

Por último, en la Figura 3.26 y la Figura 3.27 se pueden ver los renders 3D de las PCB de indicadores de los ART20C y ART20E respectivamente.

### 3.4 Diseño e implementación del sistema. Software.

El software del sistema de salpicadero ha sido desarrollado en el software Arduino® IDE. El motivo de desarrollar el software en un entorno tan básico es la facilidad que tiene para la instalación de librerías utilizadas en el código, así como la versatilidad del mismo a la hora de cargar el código en

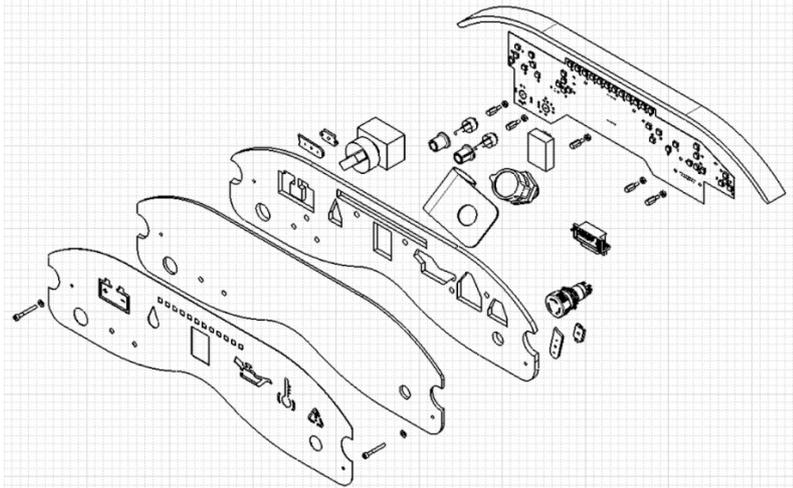


Figura 3.25 Explosión del salpicadero del ART20C.

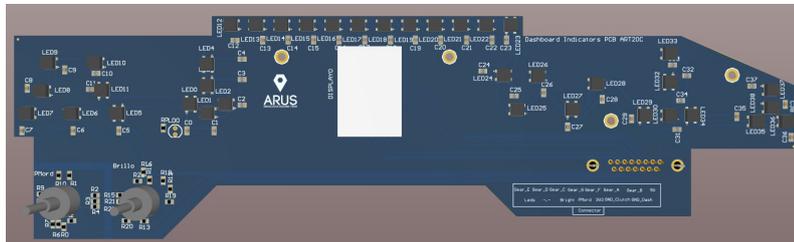


Figura 3.26 Render de la PCB de indicadores del ART20C.

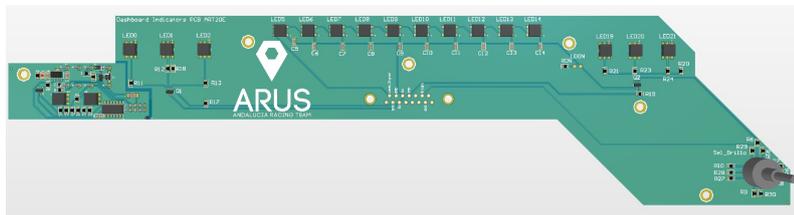


Figura 3.27 Render de la PCB de indicadores del ART20E.

la PCB controladora.

Tal y como se explicó en capítulos anteriores, la controladora no dispone de módulo de programación, por lo que para cargar el código en la placa es necesario un launchpad de Arduino® Uno. Este launchpad, a partir de ahora el programador, hace de intermediario entre el PC donde se encuentra almacenado el código y el microcontrolador objetivo. El motivo de escoger este programador es porque el microcontrolador instalado es el mismo que utiliza la PCB controladora, facilitando mucho la acción de carga de código.

Cuando se adquieren en el mercado los microcontroladores de la PCB controladora, los **ATMega328P-AUR**, su configuración por defecto hace la velocidad de reloj sea la interna del sistema. Sin embargo, tal y como se explicó en la sección anterior, la controladora tiene un cristal oscilador de 16MHz, por lo que antes de cargar cualquier código es necesario modificar los fuse bits internos del microcontrolador para que éste escoja como reloj principal el cristal externo instalado.

### 3.4.1 Código ART20C. Diagrama de flujo

El código en el monoplaza de combustión es sensiblemente más complejo dado que es necesario mostrarle al piloto más información que en el caso del monoplaza eléctrico.

En el diagrama de flujo mostrado en la Figura 3.28, se puede ver como el código del salpicadero del ART20C tiene 4 funcionalidades distintas:

- Lectura del mando rotativo: el divisor resistivo formado por el mando rotativo se conecta directamente a un pin de lectura analógica del **ATMega328P-AUR**. Este pin se lee cada iteración del bucle principal.
- Lectura de CAN-Bus: en cada iteración se pregunta si hay mensaje disponible, en caso negativo se espera a que llegue alguno.
- Actualización de panel multifunción: según la lectura del mando rotativo, se actualiza la información que desea el piloto en el panel multifunción.
- Actualización de todos los indicadores: el resto de indicadores se actualizan de la misma manera que el panel multifunción, solo que estos indicadores tienen un solo dato relacionado.

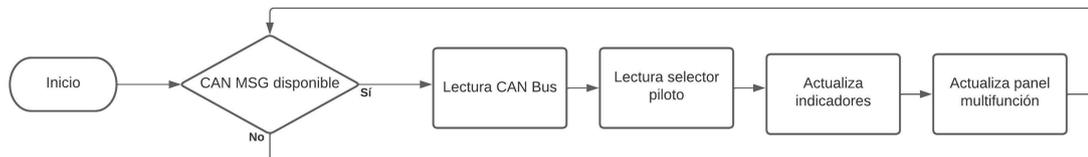


Figura 3.28 Diagrama de flujo del código para el ART20C.

### 3.4.2 Código ART20C. Fragmentos.

Para tener una idea de cómo se ha programado la controladora, en el código 3.2 se puede observar la configuración de pines y de CAN Bus así como el bucle principal del código del ART20C.

#### Código 3.1 Setup y bucle principal del código para el ART20C.

```

void setup()
{
  pinMode(A2, OUTPUT);
  digitalWrite(A2, LOW);
  pinMode(G0, OUTPUT);
  pinMode(G1, OUTPUT);
  pinMode(G2, OUTPUT);
  pinMode(G3, OUTPUT);
  Serial.begin(115200);
  FastLED.addLeds<WS2812B, LEDS_PIN, GRB>(leds, NUM_LEDS);
  FastLED.setBrightness(brillo);

  while (CAN_OK != CAN.begin(CAN_500KBPS, MCP_16MHz)) // init can
    bus : baudrate = 500k
  {
    Serial.println("CAN BUS Shield init fail");
    Serial.println(" Init CAN BUS Shield again");
  }
}

```

```
    delay(1000);
}

Serial.println("CAN BUS Shield init ok!");
}

void loop() {
    unsigned char len = 0;
    unsigned char rxBuf[8];
    if (!done) {
        sec_inicial();
        done = true;
        i = 0;
    }

    if (CAN_MSGAVAIL == CAN.checkReceive())    // Se espera a entrada de
        dato, no interrupción
    {
        CAN.readMsgBuf(&len, rxBuf);           // Lee dato y longitud de
        la trama
        unsigned int canId = CAN.getCanId();    // Lee ID de la trama de
        CAN
        unsigned char stmp[8] = {0, 0, 0, 0, 0, 0, 0, 0};

        switch (canId) {
            case ECU1:
                revs = rxBuf[0] * 100;
                tp = rxBuf[1] - 50;
                front_press = rxBuf[2];
                rear_press = rxBuf[3];
                stWheel = rxBuf[4] - 50;
                gear = rxBuf[5];
                lambda = rxBuf[6] * 256 + rxBuf[7];
                break;

            case ECU2:
                ect = rxBuf[0] - 50;
                t_radiador = rxBuf[1] - 50;
                t_cockpit = rxBuf[2] - 50;
                iat = rxBuf[3] - 50;
                t_aceite = rxBuf[4] - 50;
                t_ecu = rxBuf[5] - 50;
                p_aceite = rxBuf[6] * 256 + rxBuf[7];
                break;

            case ECU3:
                batt_lvl = rxBuf[0];
                f_table = rxBuf[1] + 50;
                tc_onoff = rxBuf[2]; //Será o 0xFF o 0x00
```

```

    slip = rxBuf[3] + 50;
    lambda_targ = rxBuf[4] * 256 + rxBuf[5];
    rpm_limit = rxBuf[6]; //Será o 0xFF o 0x00
    log_use = rxBuf[7] + 50;
    break;

case POTENCIA:
    i_servoPad = rxBuf[0];
    i_fan = rxBuf[1];
    i_elec = rxBuf[2];
    i_fuel = rxBuf[3];
    i_inyectors = rxBuf[4];
    temp_box = rxBuf[5];
    lv_bateria = rxBuf[6];
    break;

case CLUTCH:
    hall_r = rxBuf[0] + 50;
    pot_r = rxBuf[1] + 50;
    hall_l = rxBuf[2] + 50;
    pot_l = rxBuf[3] + 50;
    sel_rango = rxBuf[4];
    sel_clutch = sel_rango >> 4;
    misc = rxBuf[5];
    auto_start = bitRead(rxBuf[5],5);
    break;

case BRIGHT:
    revs2 = rxBuf[0] * 256;
    revs2 += rxBuf[1];
    neutral = rxBuf[2];
    break;

default:
    break;

}
}
driversel(); //Actualizamos tira de led con la
    selección del piloto
alertas(); //Actualizamos indicadores
    individuales despues de la seleccion del piloto, evitando solape
    de la tira a indicadores colindantes
FastLED.show(); //Mandamos el PWM completo a los
    LEDs
}

```

A continuación se describen brevemente las funciones utilizadas en este código:

- `CAN.checkReceive()`: comprueba si se ha recibido alguna trama de CAN con cualquier ID.

- `CAN.readMsgBuf()`: se lee la trama disponible en el driver.
- `CAN.getCanID()`: obtiene la ID de la trama de CAN recibida.
- `driverSel()`: lee el pin analógico del mando rotativo y actualiza la información en el panel multifunción.
- `alertas()`: prepara el resto de indicadores.
- `FastLED.show()`: Se actualizan todos los LEDs de la PCB de indicadores.

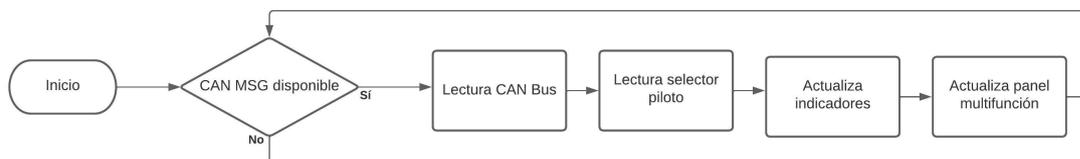
Como se puede apreciar, se utilizan varias funciones de librería. Las librerías utilizadas en el código son:

- FastLED: contiene las funciones driver de los LEDs `WS2812B`, que son usados en la PCB de indicadores.
- CAN: contiene las funciones para Arduino<sup>®</sup> Uno del driver de CAN MCP2515.

Estas librerías son incluidas manualmente en el software Arduino<sup>®</sup> IDE.

### 3.4.3 Código ART20E. Diagrama de flujo

El diagrama de flujo del código del sistema de salpicadero para el ART20E es exactamente igual que el del ART20C (Figura 3.29). Sin embargo, los procesos que se ejecutan en él son muy distintos. Con respecto a los procesos del ART20C las diferencias son las siguientes:



**Figura 3.29** Diagrama de flujo del código para el ART20E.

- Actualiza indicadores: en este caso, los LEDs de los indicadores del salpicadero del ART20E no son programables, sino LEDs usuales de color rojo. Estos LEDs se activan solo cuando se detecta un flanco de bajada en la señal de control del indicador correspondiente.
- Actualiza panel multifunción: el panel multifunción en este caso sigue un funcionamiento parecido al del ART20C, sin embargo, los datos, incluso la naturaleza de los mismos, son distintos.

### 3.4.4 Código ART20E. Fragmentos

Por parte del código del ART20E, no es interesante entrar en detalle en fragmentos de código del mismo, puesto que su funcionamiento es parecido con la salvedad del número de indicadores que gestiona. En el código 3.2, por explicar algo interesante del código, se puede ver como el sistema gestiona los indicadores críticos del salpicadero.

#### Código 3.2 Gestión de los indicadores críticos del ART20E.

```

void alertas(){
    if(digitalRead(IN_IMD)){

```

```
    digitalWrite(LED_IMD, HIGH);  
  
}else{  
  
    digitalWrite(LED_IMD, LOW);  
  
}  
  
if(digitalRead(IN_BMS)){  
  
    digitalWrite(LED_BMS, LOW);  
  
}else{  
  
    digitalWrite(LED_BMS, HIGH);  
  
}  
}
```

Se puede apreciar como la gestión se limita simplemente a pasar la información de entrada a los indicadores. Puede surgir la duda de por qué la necesidad de hacer pasar las señales de AMS e IMD por la controladora, pudiendo conectarlas directamente a la PCB de indicadores, esto se hace para poder llevar un monitoreo durante los tests de pretemporada y competición de cuáles son los sistemas que más fallos producen en el monoplaza para poder prevenirlos antes de salir a pista o, incluso, durante una prueba.

## 4 Rendimiento y conclusiones

---

Una vez explicado el proceso de diseño del sistema de salpicadero de los ART20, en el presente capítulo se explicarán los resultados de la implementación, así como las conclusiones llegadas al final de la temporada.

### 4.1 Decisiones tomadas y sus conclusiones

En primer lugar, se relacionarán las decisiones tomadas a principios de temporada con las conclusiones a las que se llegaron al volver de las competiciones, a finales del verano de 2021.

#### 4.1.1 General

Antes de comenzar con las conclusiones del sistema en general, cabe destacar que este terminó siendo como uno de los sistemas más fiables de los monoplasas, llegando incluso a debuggear otros sistemas de control en pista gracias al sistema de salpicadero. Por lo que se puede resumir el sistema en general como un éxito.

- Indicadores integrados: esto, sin lugar a duda, ha sido el cambio más importante en el sistema con grandes resultados. Durante las competiciones, el sistema de salpicadero funcionó sin ningún problema, incluso se aumentaron las capacidades durante las pruebas entre competiciones, introduciendo nuevas funcionalidades para las que en un principio no estaba preparado el sistema.
- Separación entre controladora y indicadores: una vez se tuvo funcionando el sistema de salpicadero, se reparó en que había tiempo suficiente como para haber integrado la electrónica de control del sistema de salpicadero en la misma placa de indicadores. Sin embargo, no ocasionó demasiado problema el no hacerlo, puesto que el desmontaje de la PCB controladora era incluso más sencillo que el desmontaje de la PCB de indicadores.
- Panel multifunción: fue el acierto más importante en este sistema. Gracias a estos indicadores se podía determinar sin necesidad de conexión a las ECUs todos los parámetros importantes del motor y del sistema de frenado del monoplasa, ahorrando mucho tiempo con el vehículo parado y permitiendo más tiempo en pista.
- Reducción de tamaño en la controladora: al disminuir el tamaño de la PCB controladora, se consiguió un espacio importante para el resto de sistemas de control de los monoplasas.

### 4.1.2 ART20C

Para el ART20C los objetivos fueron más agresivos que para el ART20E, ya que este último nunca había podido rodar en pista. El ART20C además de fiable tenía que mejorar en prestaciones, para ello, un sistema de mostrado de información era imprescindible.

- Fiabilidad en los datos: durante todas las horas de rodaje en pista del ART20C, no se reportó ningún fallo en el sistema.
- Temperaturas en rango: gracias a la precisión conseguida en el panel multifunción, el ART20C pudo ser testado siempre en rangos de temperaturas y presiones correctos.

### 4.1.3 ART20E

Como se explicó en el punto anterior, el ART20E tocó pista por primera vez en la temporada 2020-21. Esto ocasionaba unos requisitos distintos en el sistema para este monoplaça.

- Adaptación a la normativa: los indicadores de AMS e IMD y su secuencia de arranque pasaron sin problemas las inspecciones técnicas.
- SoC: gracias a la información proporcionada por el sistema AMS se pudo observar durante las competiciones el estado de carga de la batería del monoplaça, permitiendo calcular cuánto tiempo en pista le quedaba.

## 4.2 Estudio de consumo

Para poder tener una comparativa entre el consumo de las opciones comerciales de las que se hablaron en el Capítulo 3 se realizaron sendos análisis de consumo a los sistemas en ambos coches en diferentes situaciones.

### 4.2.1 Consumo en el sistema del ART20C

En la Tabla 4.1 se indican las situaciones en las que se realizaron los estudios de consumo y la corriente media demandada. Cabe destacar que estos estudios se realizaron en laboratorio, sin llegar nunca a realizarse pruebas en campo, sin embargo, estos se realizaron de la forma más fiable posible, con el sistema completo funcionando y simulando condiciones reales del mismo.

**Tabla 4.1** Consumo del sistema en el ART20C.

Condiciones	Consumo teórico	Consumo experimental
Indicadores apagados	33mA	37.22mA
Indicador batería encendido	93mA	90.15mA
Indicador batería y temperatura aceite	153mA	145.25mA
Condiciones R2D <sup>a</sup> diurno	115mA	107.25mA
Condiciones R2D nocturno	92mA	89.58mA

<sup>a</sup> En el ART20C las condiciones R2D se dan cuando el monoplaça tiene sus temperaturas y presiones en rangos óptimos para salir a pista

#### 4.2.2 Consumo en el sistema del ART20E

Por último, en la Tabla 4.2, se puede ver el estudio de consumo sobre el sistema de salpicadero del ART20E. Se puede apreciar como los consumos son sensiblemente inferiores a los consumos en el ART20C, ya que este dispone de más indicadores y, por tanto, mayor número de LEDs para encender.

Como aclaración, los cálculos de consumo teóricos se han realizado mediante datos incluidos en

**Tabla 4.2** Consumo en el sistema del ART20E.

Condiciones	Consumo teórico	Consumo experimental
Indicadores apagados	21mA	23.12mA
Panel multifunción completamente encendido	81mA	78.25mA
IMD, AMS y panel multifunción encendido	97mA	96.55mA
Condiciones R2D <sup>a</sup> diurno	78mA	76.5mA
Condiciones R2D nocturno	72mA	71.3mA

<sup>a</sup> En el ART20E las condiciones R2D se dan cuando el piloto acciona la alta tensión al motor y no existe ningún fallo en los sistemas electrónicos de seguridad (AMS e IMD)

los datasheets de los componentes utilizados en el sistema y las medidas experimentales se toman mediante el consumo indicado por la fuente de tensión utilizada en el laboratorio.

### 4.3 Comparativa de precios

Tal y como se comentó durante los primeros capítulos del presente documento, una de las razones por la que se descartó la opción comercial fue el desempeño económico de esta opción.

Una vez terminado el diseño, se resume en la Tabla 4.3 el gasto económico final del sistema completo, incluyendo fabricación de las PCB y componentes de las mismas.

**Tabla 4.3** Resumen de precios del sistema de salpicadero.

	PCB Controladora	Indicadores ART20C	Indicadores ART20E
Fabricación PCB	93.45€ (Eurocircuits <sup>®</sup> )	43.45€ (JLCPCB <sup>®</sup> )	39.85€ (JLCPCB <sup>®</sup> )
Componentes	84.25€	30.65€	25.55€

Finalmente, el sistema nos arroja un total de 317.20€. En la Tabla 4.4, vemos una comparativa de precios de las diferentes opciones del sistema de salpicadero.

**Tabla 4.4** Comparativa de precios final.

	MXG 1.2 Strada	GS-Dash	CD-5	Sistema 2019	Sistema 2020
Precio total	1200€	500€	N/A	450€	317.20€

Como se puede ver, se consigue una disminución de precio bastante importante, teniendo en cuenta que el sistema completo es para ambos coches, necesitando, en la opción comercial, desembolsar el doble de su precio para implementar en los dos coches.

## 4.4 Mejoras futuras

Sabiendo que en el futuro se partirá de un sistema ya fiable, se proponen ciertas mejoras en el diseño del sistema de salpicadero que podrían aumentar el rendimiento del mismo.

### **Integración completa**

Esta idea, como se explicó en apartados anteriores, se descartó debido al aumento de tiempo de diseño que requeriría, un riesgo excesivo teniendo en cuenta que no se tenía un sistema de salpicadero totalmente fiable como del que se dispone a partir de ahora.

Uno de los problemas detectados en competición, no a nivel de diseño del sistema sino a nivel de organización de los mismos, es que si el diseñador del sistema no puede estar en el sitio, resulta difícil realizar cambios propensos a implementar in situ, ya que el sistema de salpicadero de los ART20 son los únicos que implementan el microcontrolador ATmega328P. El resto de los sistemas electrónicos de los monoplazas implementan microcontroladores del fabricante Texas Instruments<sup>®</sup>, MSP430 o Tiva. Adherir el sistema de salpicadero a uno de estos microcontroladores aportaría homogeneidad a los sistemas electrónicos.

### **Inclusión de pantalla TFT**

Para un mostrado de datos más elegante, con mayor número de funcionalidades y más parecido a lo que se usa en competiciones automovilísticas como la Fórmula 1 o el WEC, se propone la inclusión de una pantalla en el sistema de salpicadero. Este cambio en la estructura del sistema provocaría también un cambio de microcontrolador, ya que el actual no tendría la potencia suficiente como para hacerse cargo del manejo de datos de lo que supondría incluir una pantalla

Como ejemplo, se propone el sistema basado en la familia de microprocesadores del fabricante STMicroelectronics<sup>®</sup> STM32. Dicha familia de microprocesadores posee una arquitectura ARM Cortex, lo que proveería al sistema de una potencia lo suficientemente grande como para poder procesar los datos de llegada y mostrarlos en una pantalla a una tasa de refresco suficiente. En esta familia de microprocesadores hay algunos modelos como el STM32L4 que disponen de interfaz de CAN directamente, sin necesidad de integrar en el diseño un módulo de CAN con driver y transceiver como se hace en el diseño actual, por lo que se implementa una de las funcionalidades más importantes del sistema ya directamente en el microcontrolador.

Por último, este ejemplo de diseño permitiría al futuro diseñador del sistema hacerse con un kit de desarrollo distribuido por el propio fabricante (como el que se puede ver en la Figura 4.1), por lo que el desarrollo del software se puede llegar a facilitar bastante.

Se deja a criterio de las generaciones venideras del equipo ARUS Andalucía Racing Team la mejora de este humilde diseño del sistema de salpicadero para las próximas temporadas. Suerte.



Figura 4.1 Kit de desarrollo de un STM32.



# Apéndice A

## Esquemáticos del sistema

Para recopilar los esquemáticos y se facilite la lectura de los mismos, en el presente apéndice se podrán encontrar todos los esquemáticos del sistemas, aquellos mostrados en todo el documento y aquellos que no.

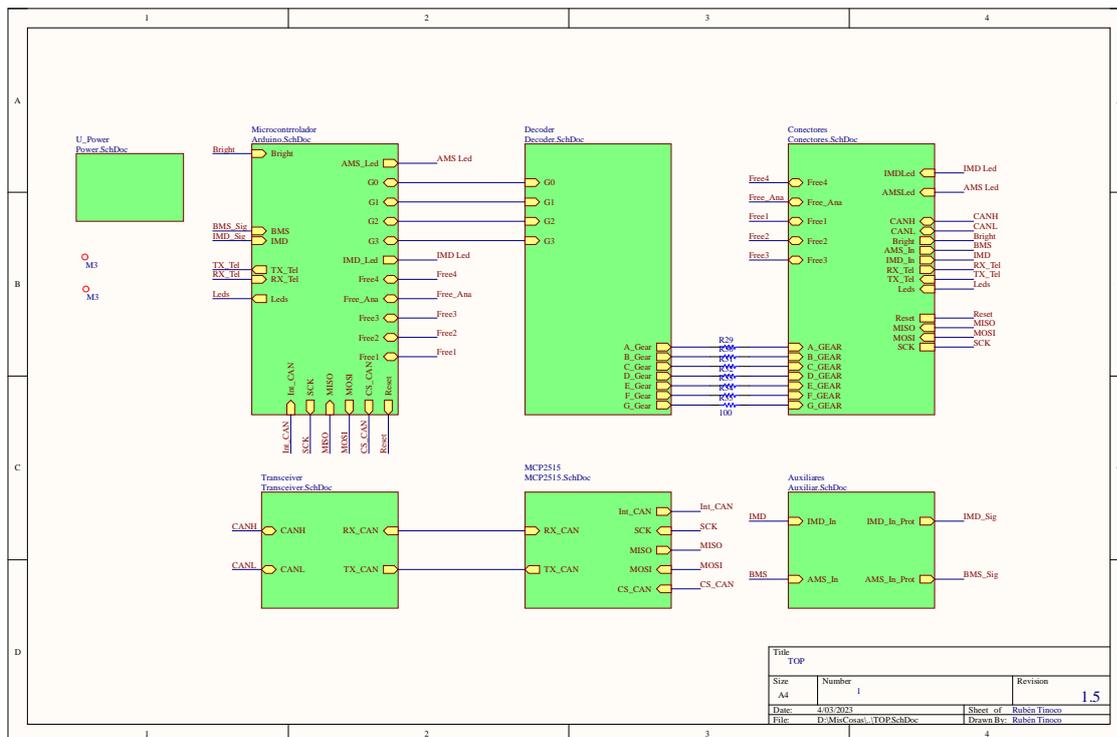


Figura A.1 Top.SchDoc controladora.

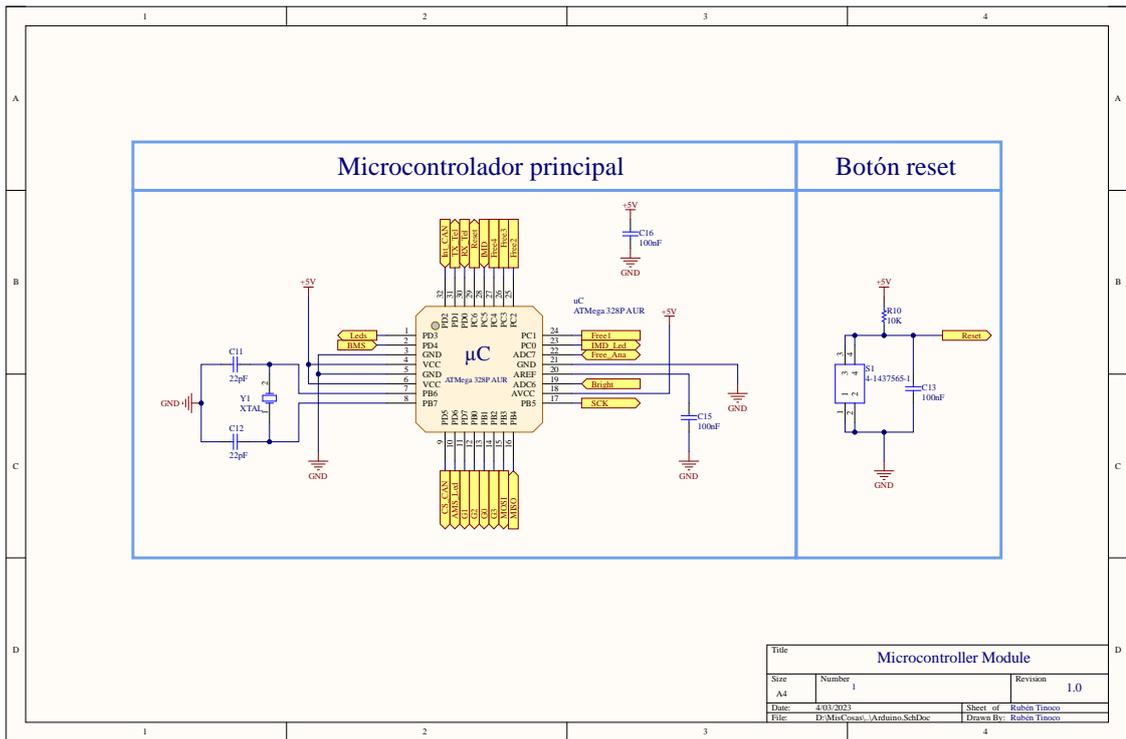


Figura A.2 Arduino.SchDoc.

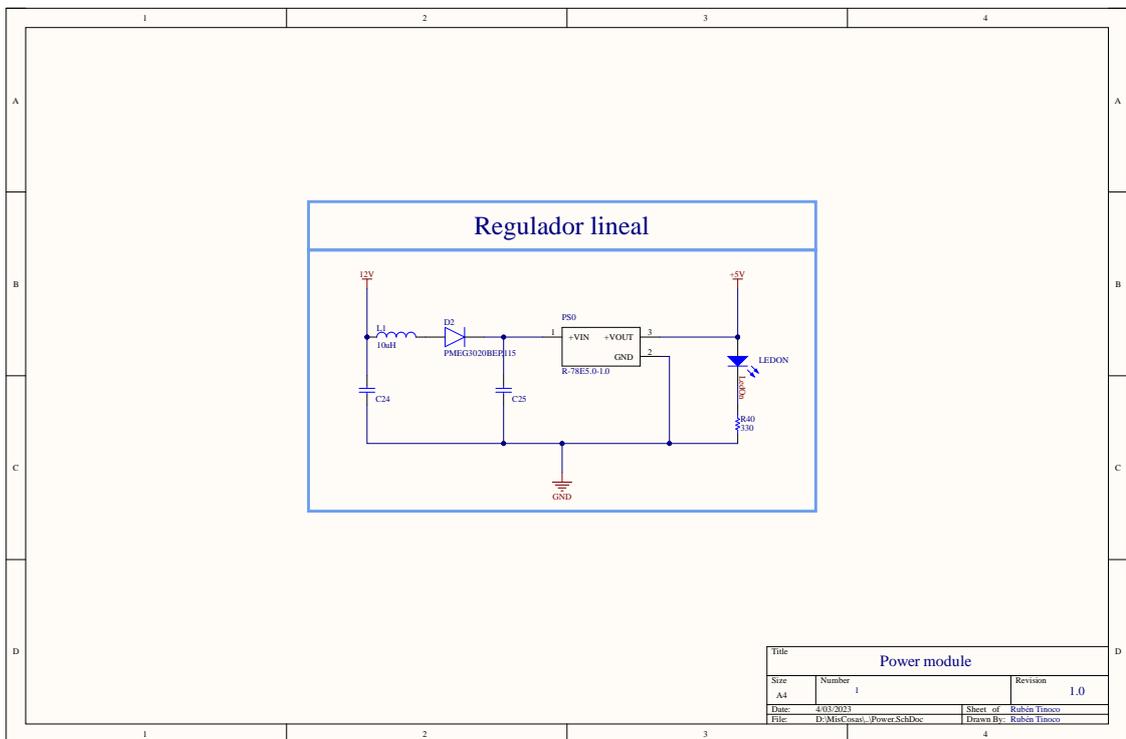


Figura A.3 Power.SchDoc.

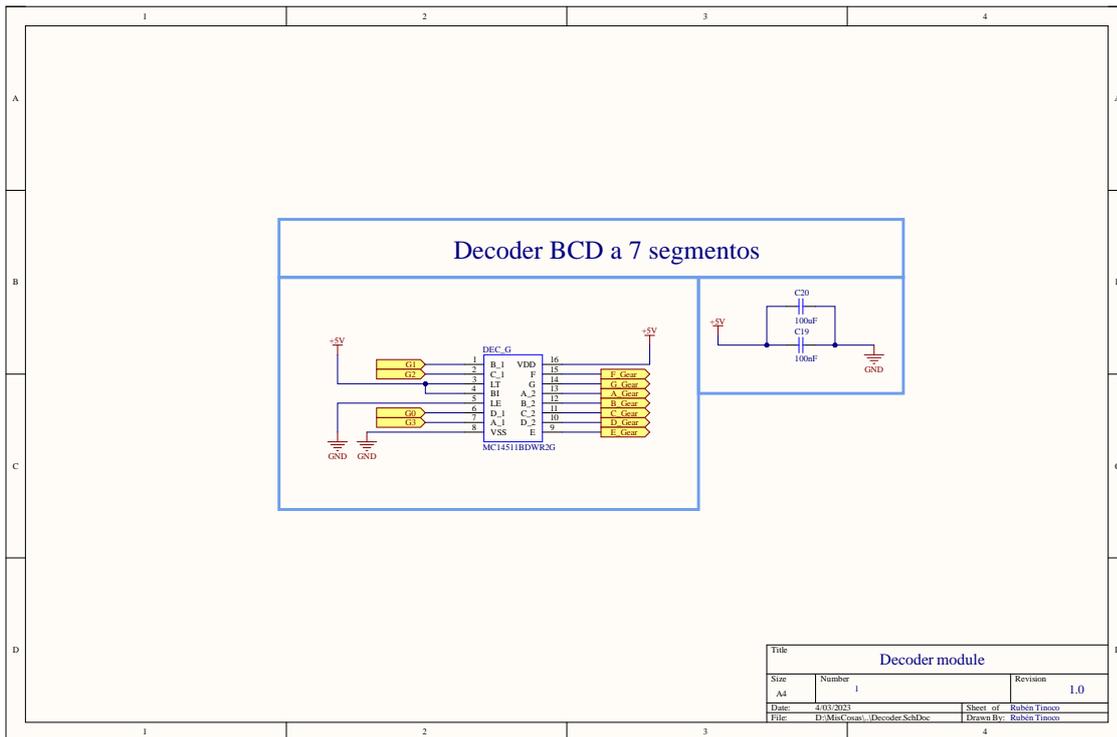


Figura A.4 Decoder .SchDoc.

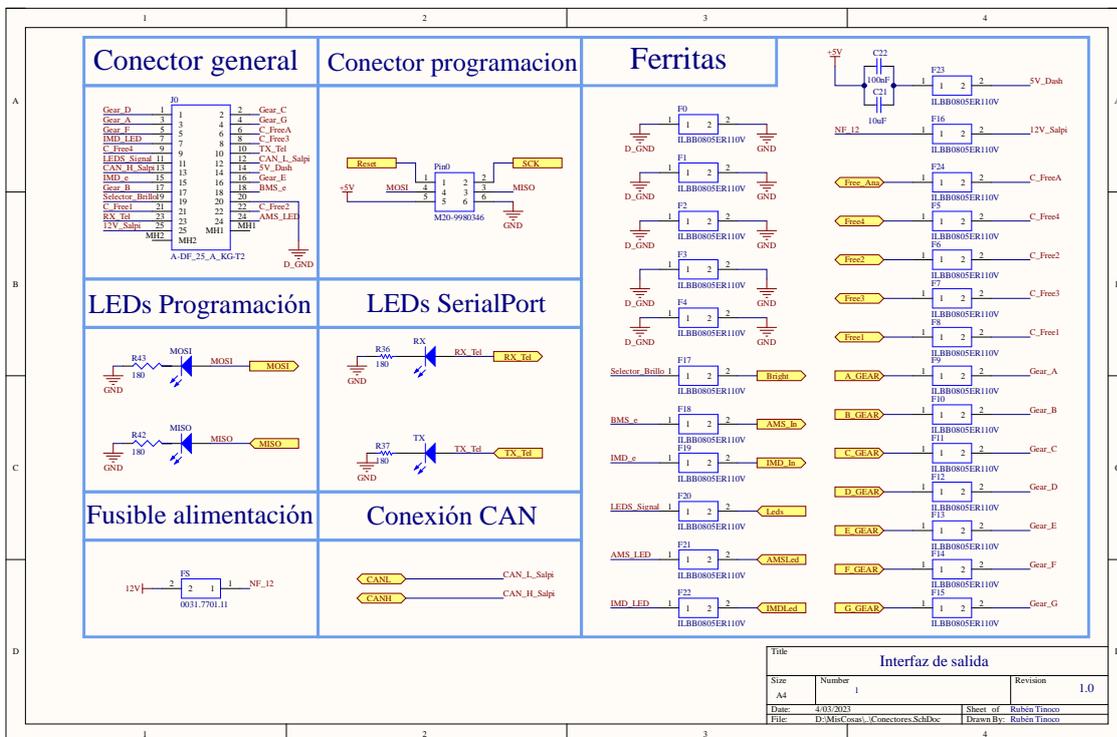


Figura A.5 Conectores .SchDoc.

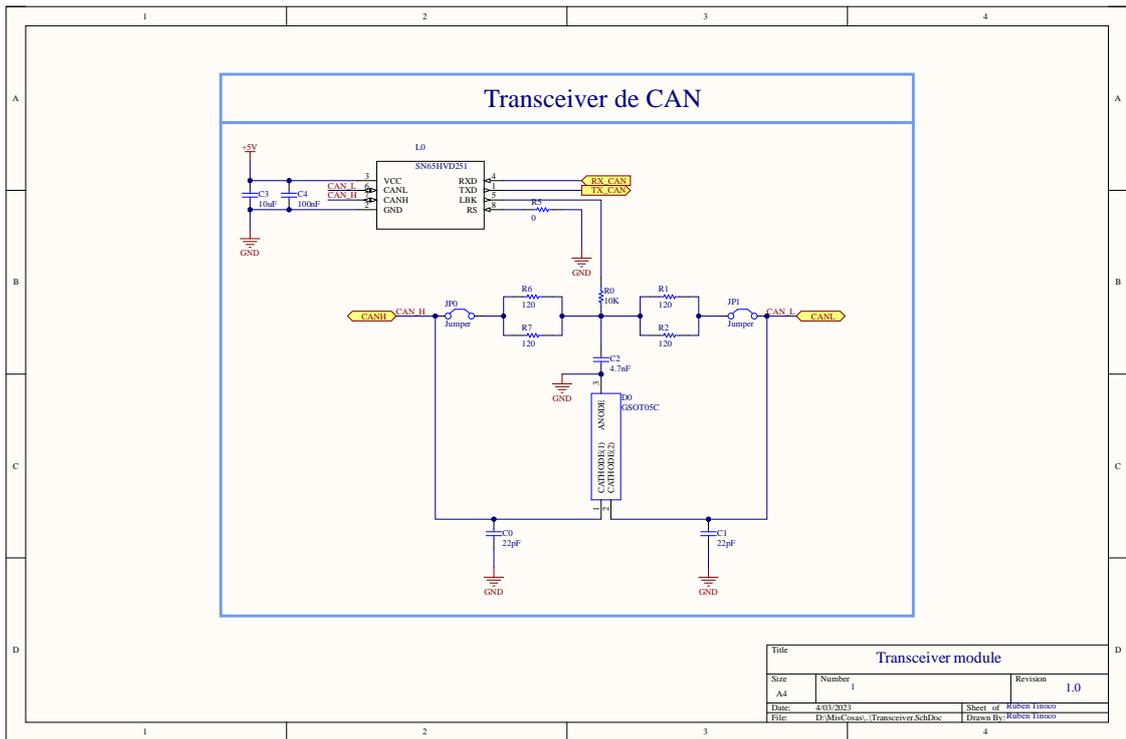


Figura A.6 Transceiver .SchDoc.

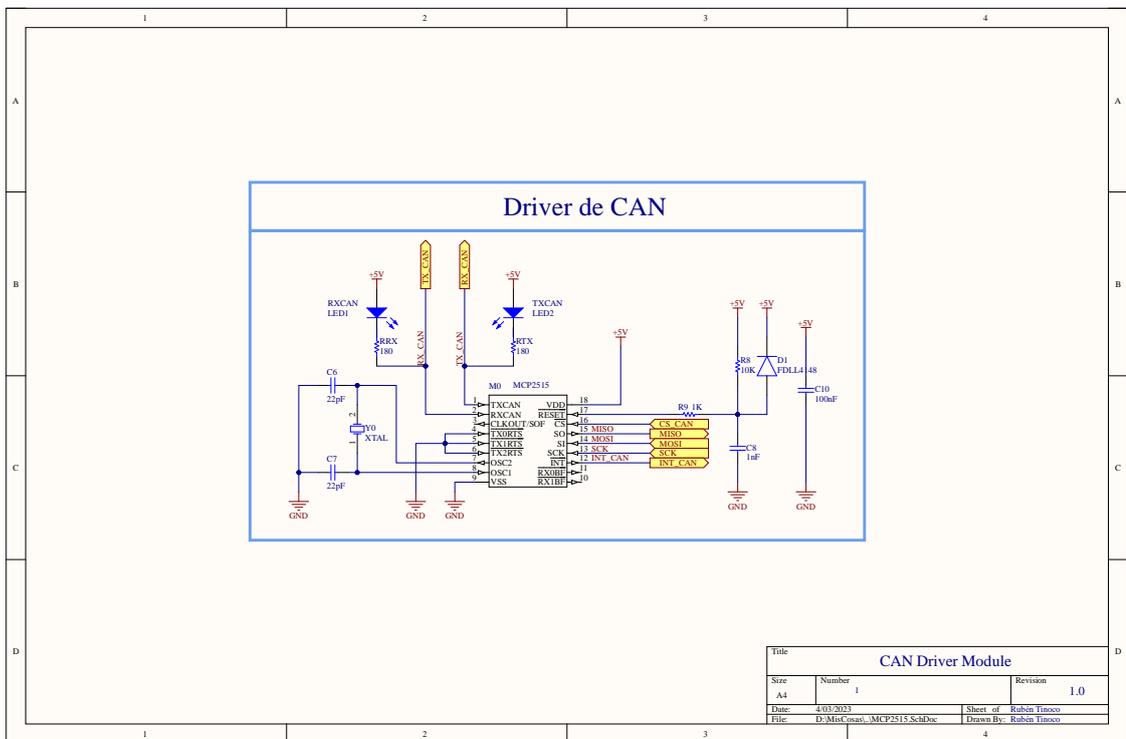


Figura A.7 MCP2515 .SchDoc.

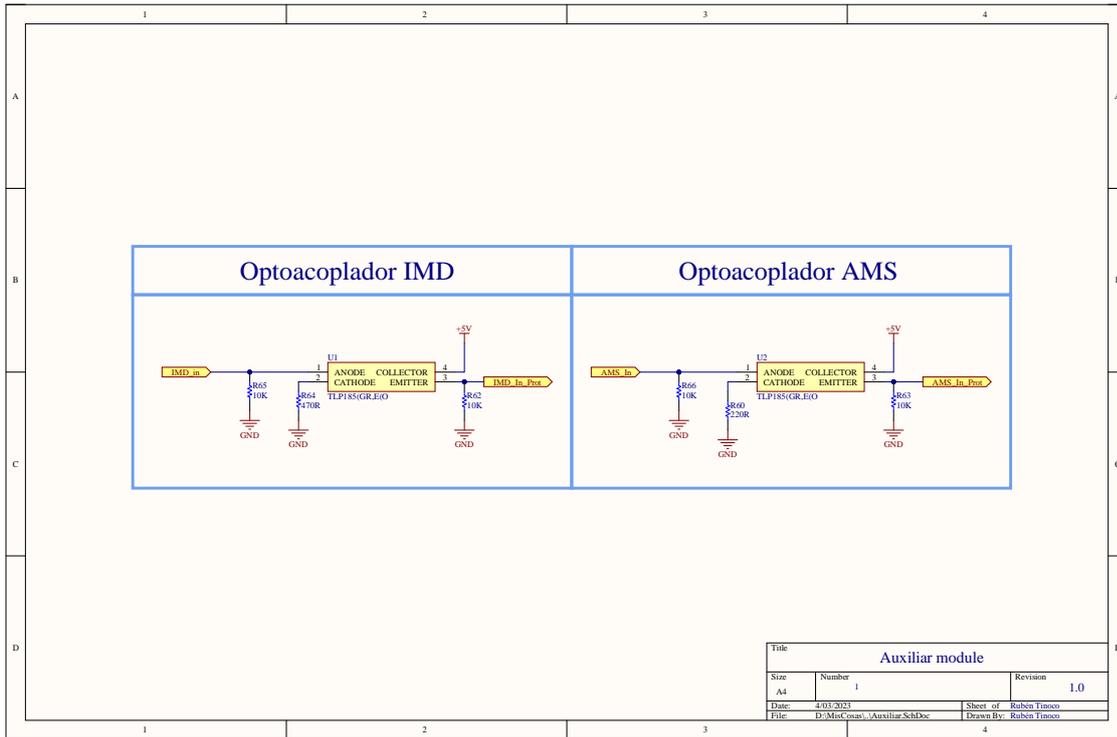


Figura A.8 Auxiliar.SchDoc.

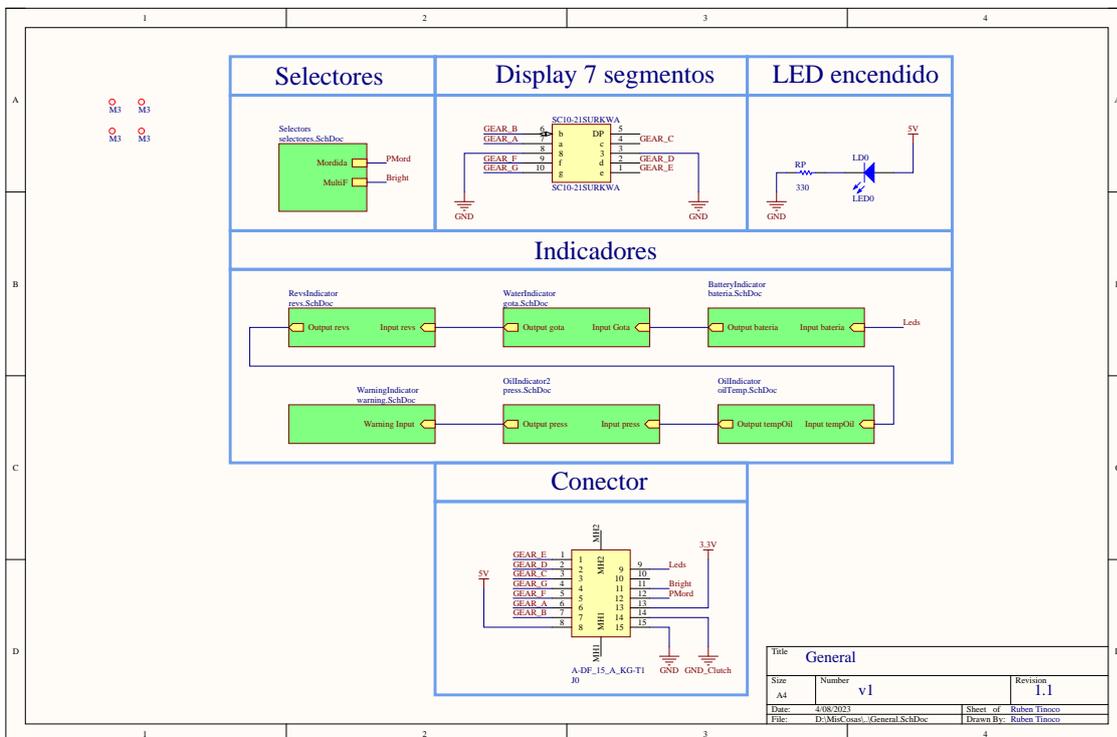


Figura A.9 General.SchDoc indicadores ART20C.

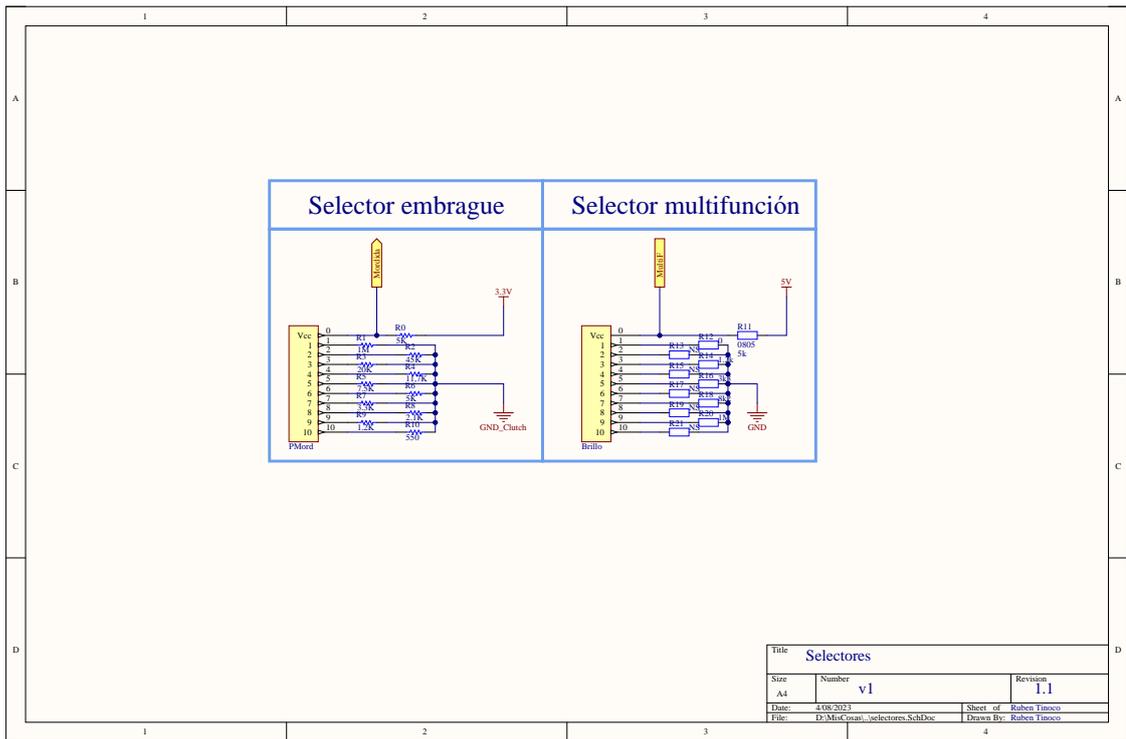


Figura A.10 Selectores.SchDoc indicadores ART20C.

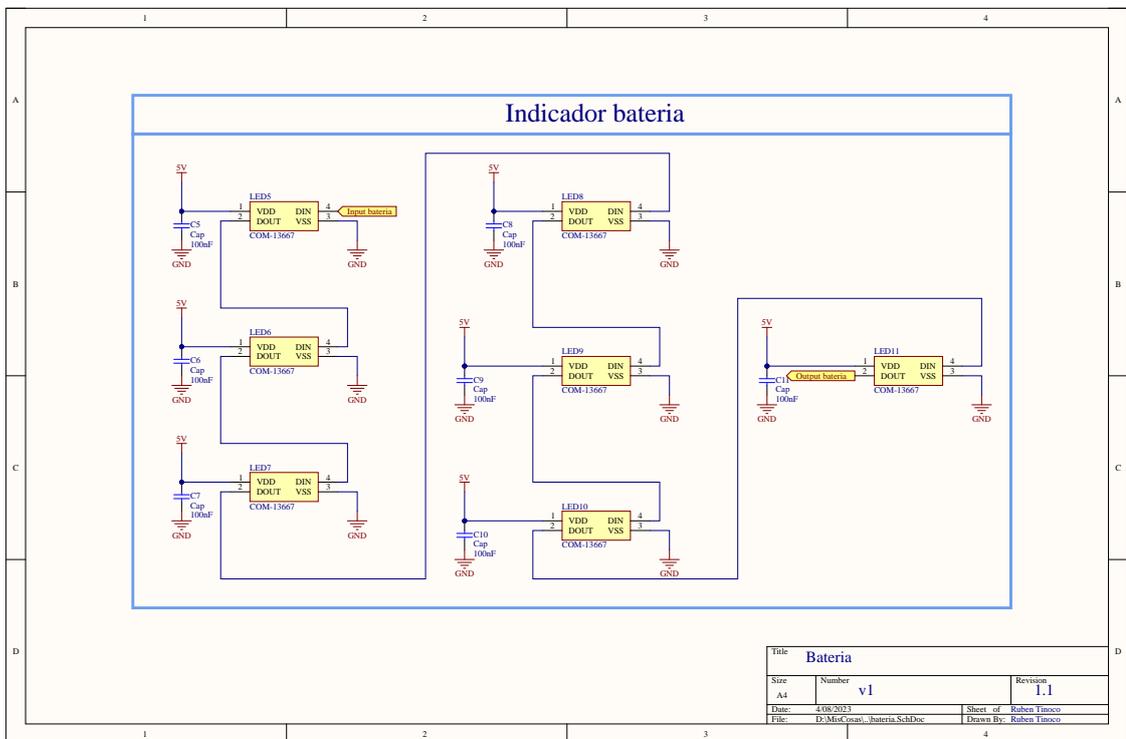


Figura A.11 Bateria.SchDoc indicadores ART20C.

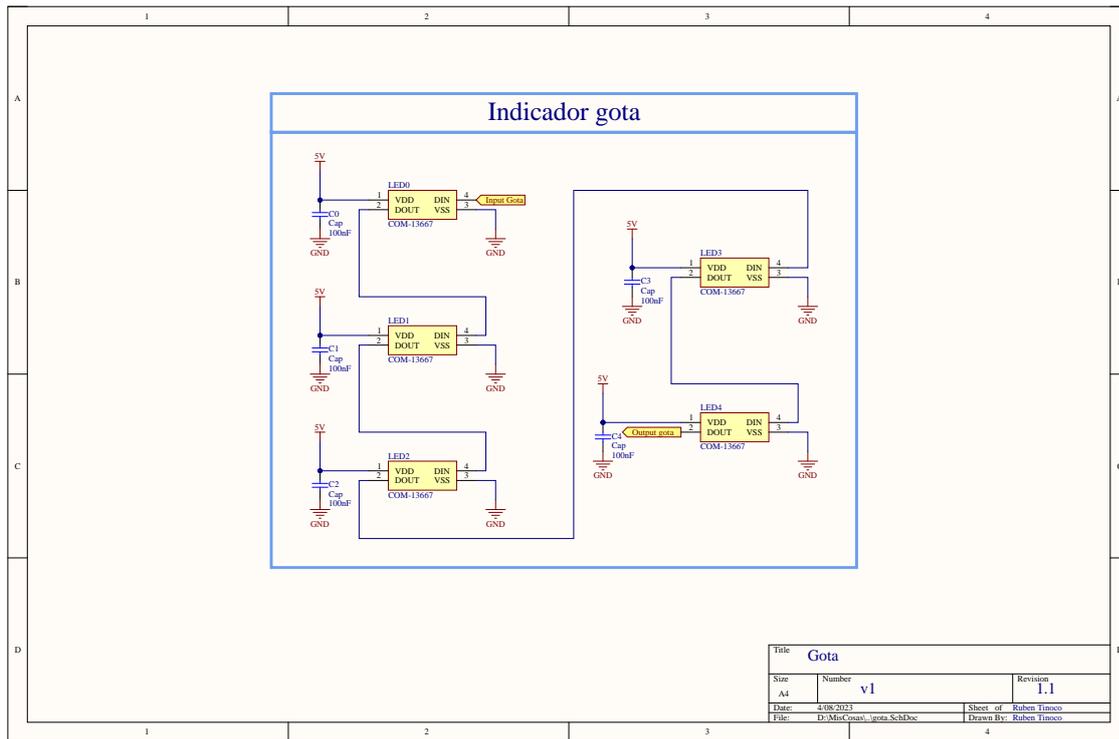


Figura A.12 Gota.SchDoc indicadores ART20C.

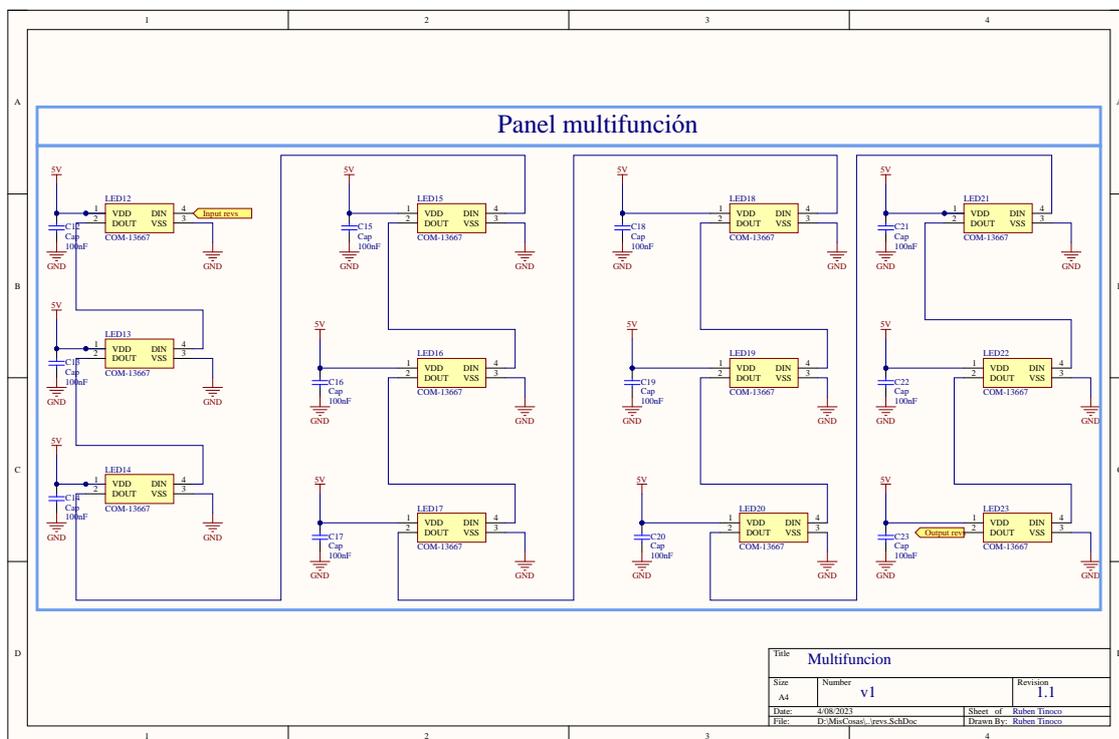


Figura A.13 Revs.SchDoc indicadores ART20C.

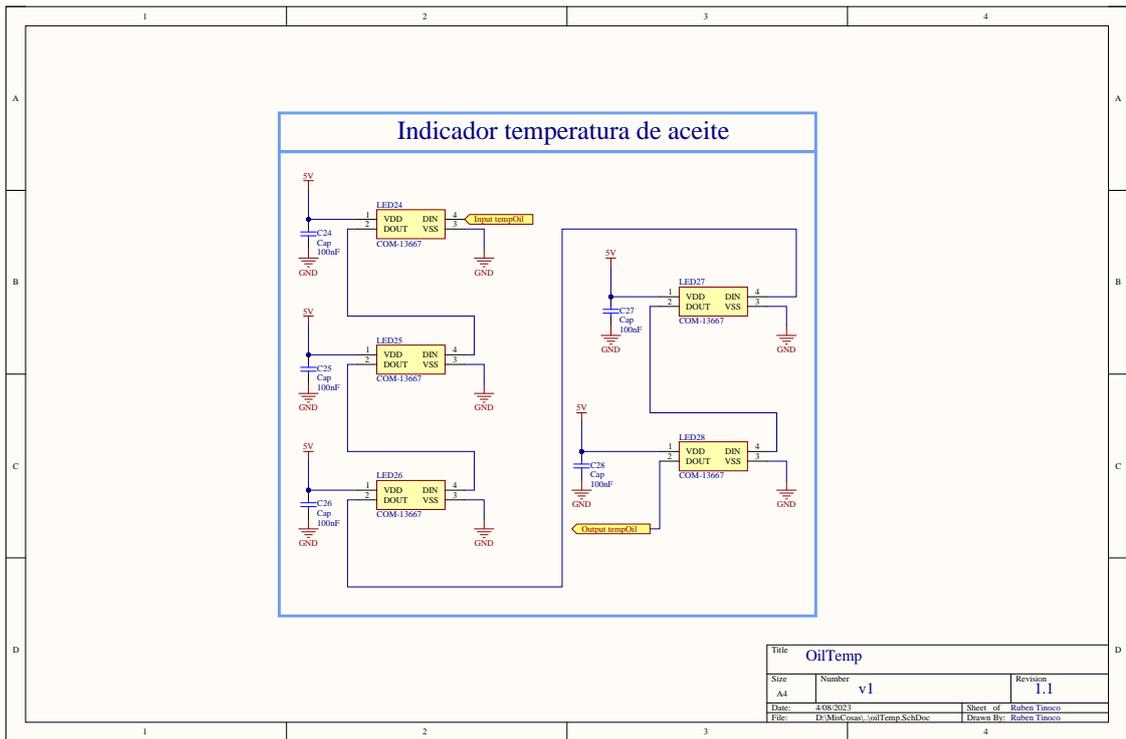


Figura A.14 OilTemp.SchDoc indicadores ART20C.

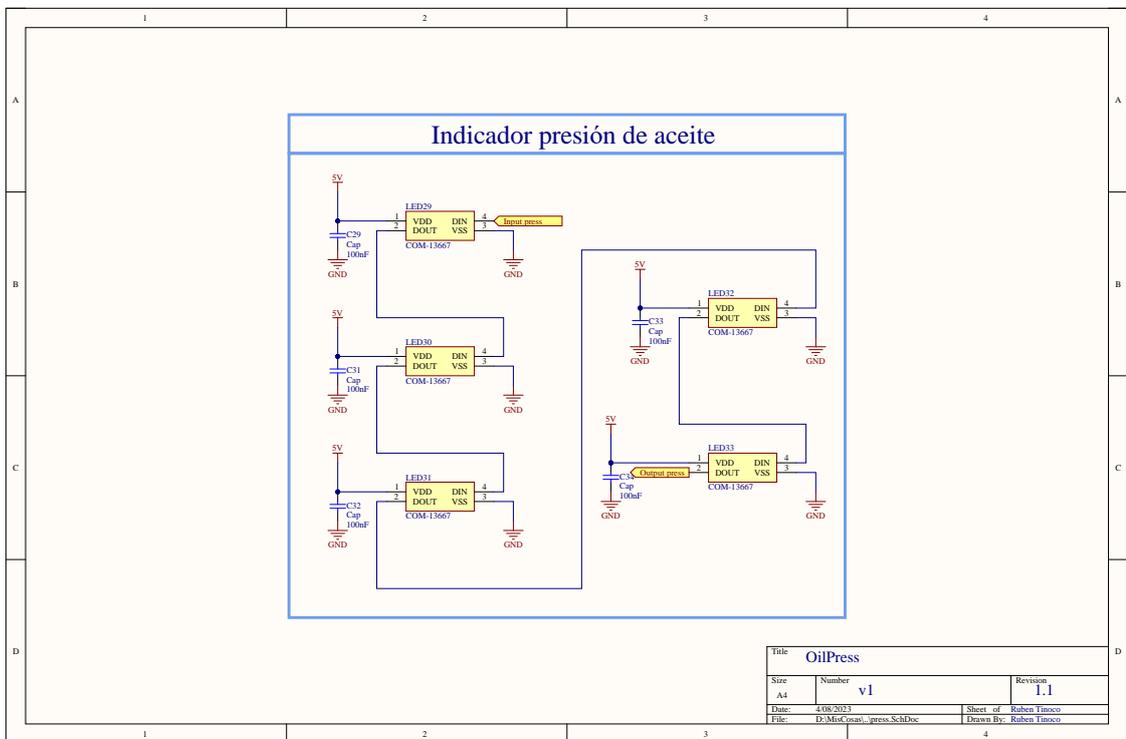


Figura A.15 Press.SchDoc indicadores ART20C.

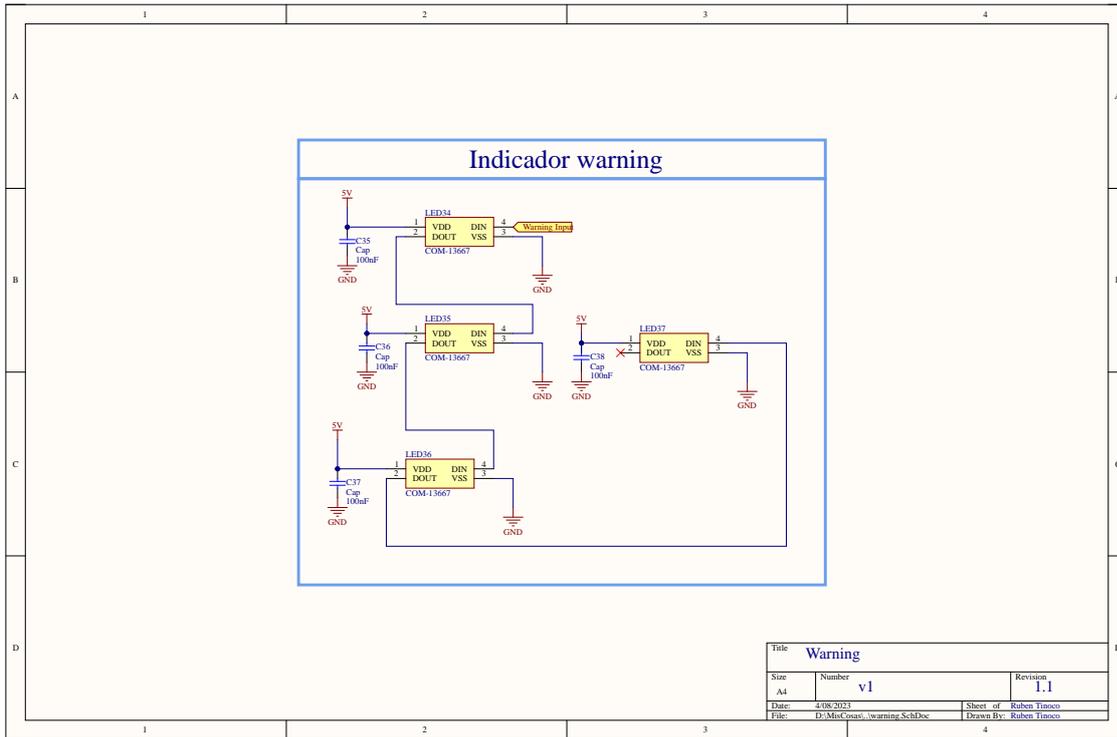


Figura A.16 Warning.SchDoc indicadores ART20C.

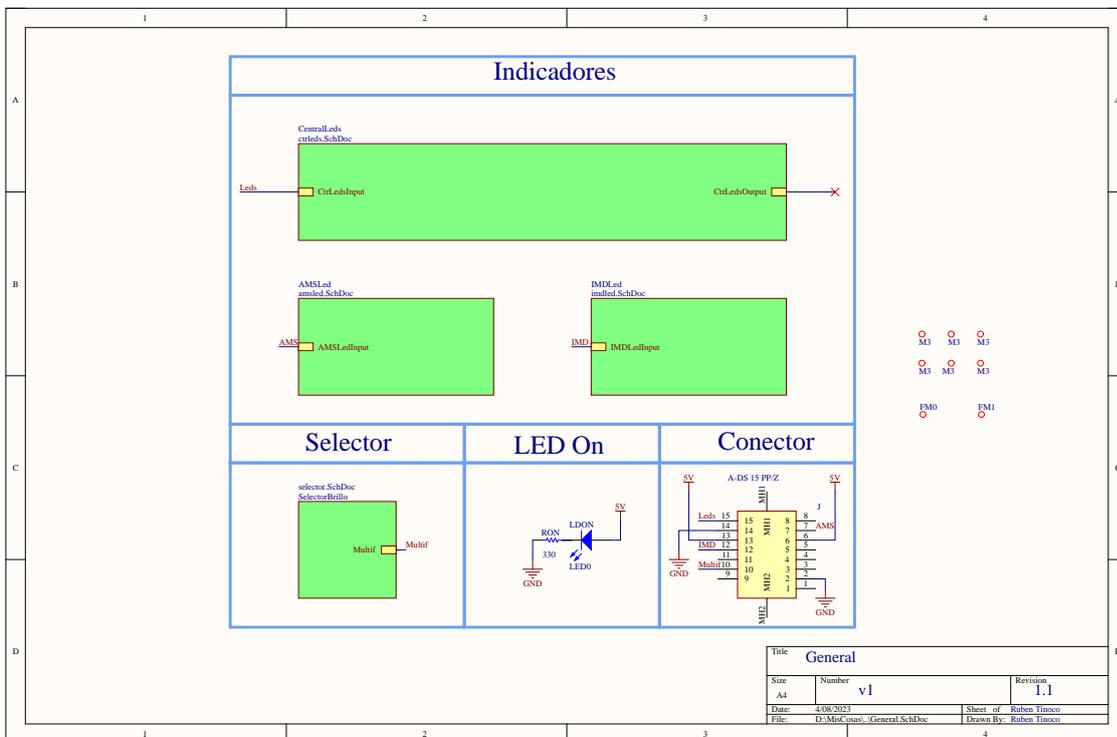


Figura A.17 General.SchDoc indicadores ART20E.

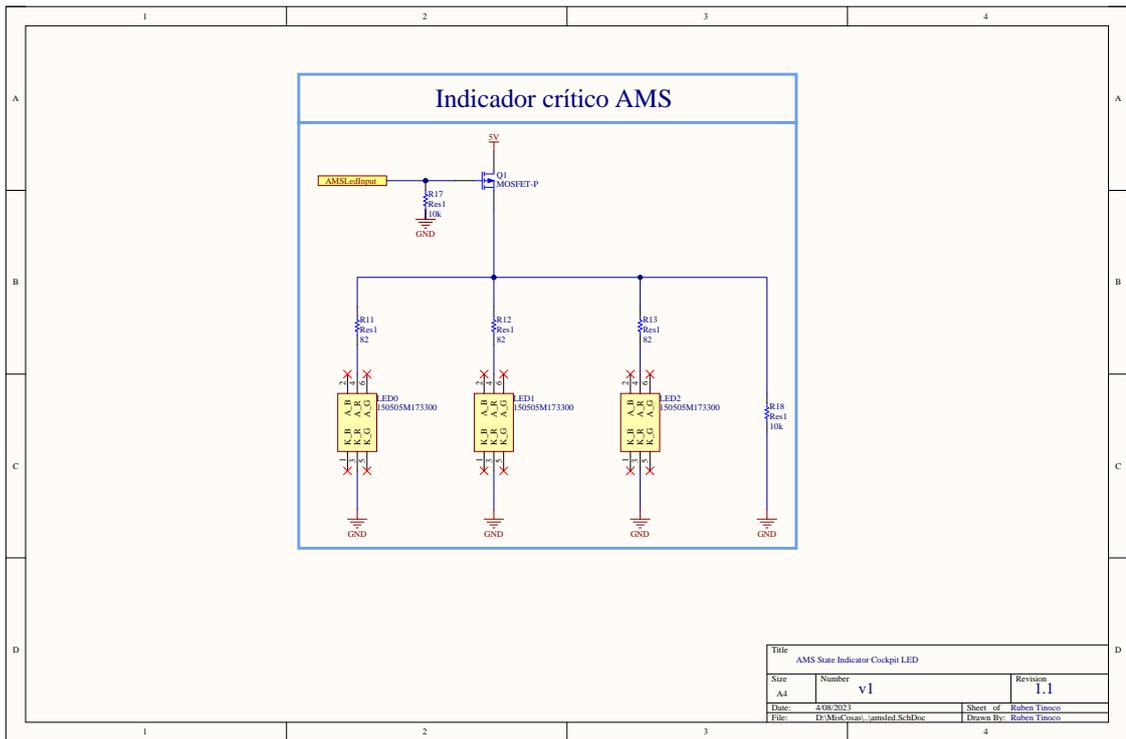


Figura A.18 AMSLed.SchDoc indicadores ART20E.

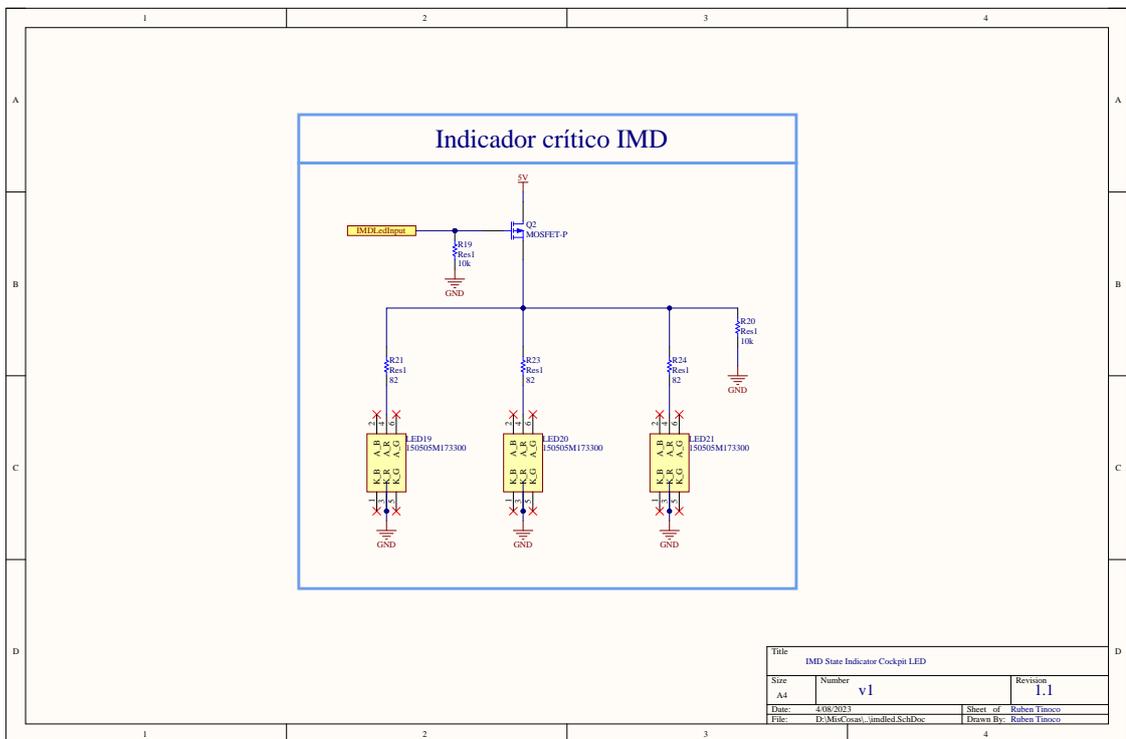


Figura A.19 IMDLed.SchDoc indicadores ART20E.

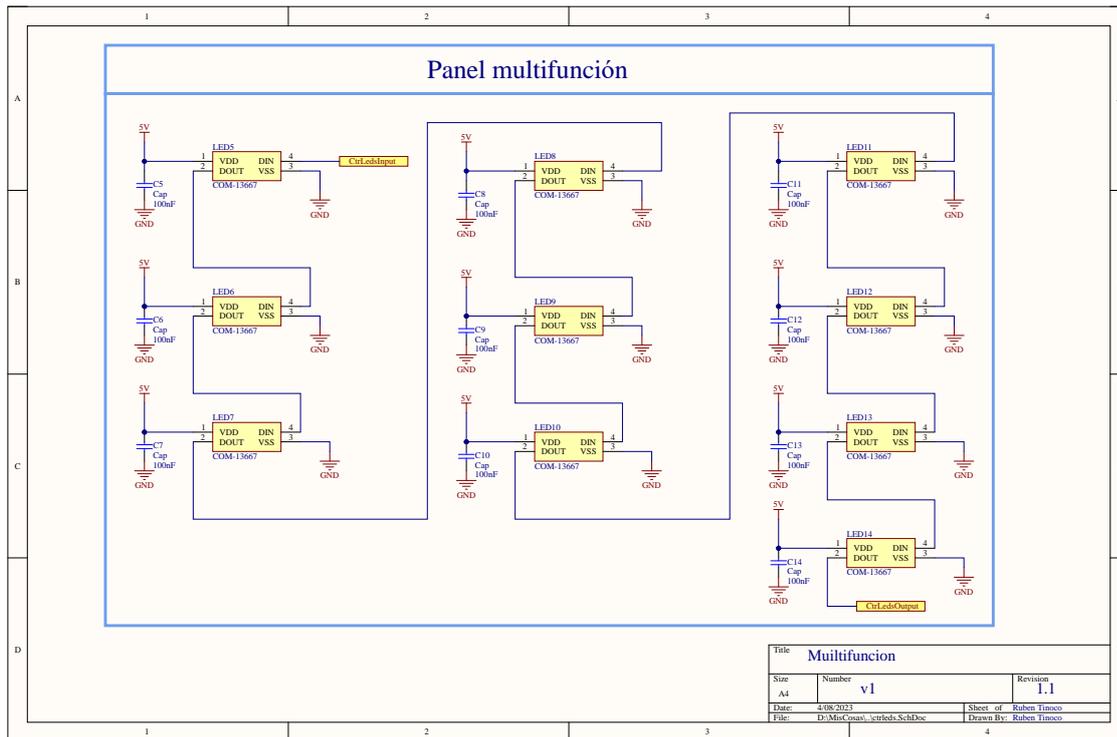


Figura A.20 multifuncion.SchDoc indicadores ART20E.

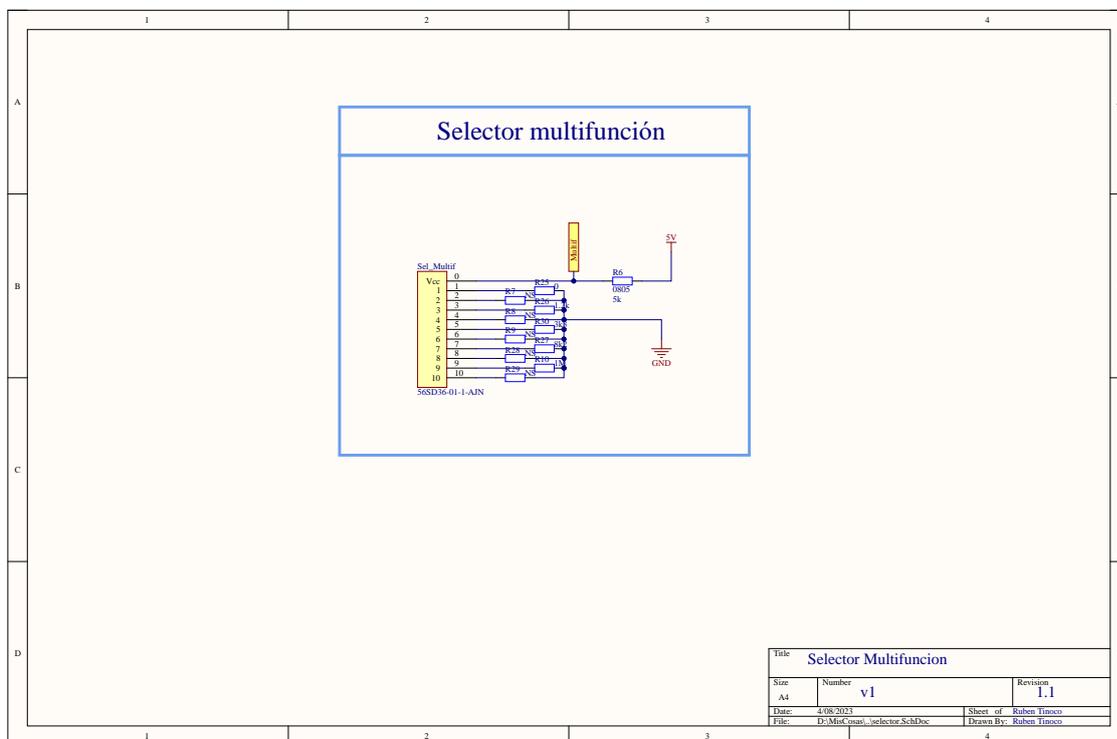


Figura A.21 selector.SchDoc indicadores ART20E.



# Índice de Figuras

---

1.1	Pista para la prueba de skidpad	2
2.1	Esquema general del sistema de salpicadero de los ART19	5
2.2	Renderizado 3D frontal de la PCB Dashboard Controller ART19	6
2.3	Renderizado 3D reverso de la PCB Dashboard Controller ART19	6
2.4	Diseño del salpicadero del ART19C	9
2.5	Diseño del salpicadero del ART19E	9
3.1	AiM MXG 1.2 Strada	12
3.2	AiM GS-Dash	12
3.3	AEM Electronics CD-5	13
3.4	Esquema simplificado de la integración moderada	15
3.5	Diagrama de bloques del salpicadero para el ART20C	16
3.6	Diagrama de bloques del salpicadero para el ART20E	16
3.7	Mapa de CAN del ART20C	17
3.8	Mapa de CAN del ART20E	18
3.9	Salpicadero del ART20C	18
3.10	Salpicadero del ART20E	19
3.11	Nivel Top de Dashboard Controller PCB	20
3.12	Unidad de alimentación de la PCB	21
3.13	Unidad del microcontrolador de la PCB	22
3.14	Unidad del driver de CAN	22
3.15	Unidad del transceiver de CAN a UART	23
3.16	Unidad del decoder BCD - 7 Segmentos	24
3.17	Unidad auxiliar	24
3.18	Interfaz de salida	25
3.19	Rutado de la cara TOP de la PCB controladora	26
3.20	Rutado de la cara BOTTOM de la PCB controladora	26
3.21	Parte frontal de la PCB controladora	27
3.22	Parte trasera de la PCB controladora	27
3.23	Esquemático de selectores rotativos	28
3.24	Esquemático del indicador crítico	28
3.25	Explosionado del salpicadero del ART20C	29
3.26	Render de la PCB de indicadores del ART20C	29
3.27	Render de la PCB de indicadores del ART20E	29
3.28	Diagrama de flujo del código para el ART20C	30

---

3.29	Diagrama de flujo del código para el ART20E	33
4.1	Kit de desarrollo de un STM32	39
A.1	Top.SchDoc controladora	41
A.2	Arduino.SchDoc	42
A.3	Power.SchDoc	42
A.4	Decoder.SchDoc	43
A.5	Conectores.SchDoc	43
A.6	Transceiver.SchDoc	44
A.7	MCP2515.SchDoc	44
A.8	Auxiliar.SchDoc	45
A.9	General.SchDoc indicadores ART20C	45
A.10	Selectores.SchDoc indicadores ART20C	46
A.11	Bateria.SchDoc indicadores ART20C	46
A.12	Gota.SchDoc indicadores ART20C	47
A.13	Revs.SchDoc indicadores ART20C	47
A.14	OilTemp.SchDoc indicadores ART20C	48
A.15	Press.SchDoc indicadores ART20C	48
A.16	Warning.SchDoc indicadores ART20C	49
A.17	General.SchDoc indicadores ART20E	49
A.18	AMSLed.SchDoc indicadores ART20E	50
A.19	IMDLed.SchDoc indicadores ART20E	50
A.20	multifuncion.SchDoc indicadores ART20E	51
A.21	selector.SchDoc indicadores ART20E	51

# Índice de Tablas

---

3.1	Comparativa entre pantallas	13
3.2	Elementos y funciones del salpicadero del ART20C	19
3.3	Elementos y funciones del salpicadero del ART20E	19
4.1	Consumo del sistema en el ART20C	36
4.2	Consumo en el sistema del ART20E	37
4.3	Resumen de precios del sistema de salpicadero	37
4.4	Comparativa de precios final	37



# Índice de Códigos

---

3.1	Setup y bucle principal del código para el ART20C	30
3.2	Gestión de los indicadores críticos del ART20E	33



# Bibliografía

---

- [1] Arduino, *Arduino uno r3*, 2023, <https://store.arduino.cc/products/arduino-uno-rev3>.
- [2] Enrique Manuel Carpintero Calderón, *Salpicadero de un monoplaza de competición*, 1 ed., Universidad de Sevilla, 2017.
- [3] AEM Electronics, *Aem electronics cd-5*, 2023, <https://www.aemelectronics.com/products/cd-digital-dash-displays-adapter-harnesses/cd-5-carbon-digital-racing-dash-displays>.
- [4] Formula Student Germany, *Formula student rules 2023*, 2023, <https://www.formulastudent.de/fileadmin/userupload/all/2023/rules/FSRules2023v1.1.pdf>.
- [5] AiM Racing, *Aim gs-dash*, 2023, <https://www.aim-racingstore.com/gb/displays/28-aim-gs-dash.html>.
- [6] ———, *Aim mxg strada*, 2023, <https://www.aim-racingstore.com/es/displays/97-aim-mxg-strada-12.html>.
- [7] Seeed Studio, *Can bus shield*, 2023, <https://wiki.seeedstudio.com/CAN-BUSShieldV2.0/>.
- [8] Wikipedia, *Bus can*, 2023, [https://es.wikipedia.org/wiki/Bus\\_CAN](https://es.wikipedia.org/wiki/Bus_CAN).