



# **Universidad Nacional Autónoma de México**

---

Facultad de Estudios Superiores

Plantel Aragón

## **Sistema de adquisición de datos para un robot móvil.**

### **TESIS**

Para obtener el título de:

**INGENIERO MECÁNICO**

**P R E S E N T A**

Aja Olmos Luis Fabián

**DIRECTOR DE TESIS**

M. en I. Humberto Mancilla Alonso

Ciudad Nezahualcóyotl, Edo. De México, 2021.





Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

## ***Agradecimientos***

A mis padres, Lucia y Javier, por haberme regalado la vida y por su apoyo incondicional a lo largo de ella. Sin su apoyo jamás lo habría logrado.

A mi asesor, maestro y amigo Humberto Mancilla Alonso, quien con su sabiduría y conocimientos me guió a través de cada una de las etapas de este proyecto y por todos y cada uno de los semestres que cursé en la facultad.

A la Universidad Nacional Autónoma de México por brindarme la oportunidad de ser mi segundo hogar por más de 7 años.

Al la Facultad de Estudios Superiores Aragón y a todos y cada uno de los profesores de ingeniería mecánica por darme los conocimientos necesarios para la vida profesional.

Al Club de Mecatrónica por haber sido una familia más para mi durante mi estancia en la facultad.

A mis compañeros y amigos: Macbeth Bali, Néstor Arteaga, Diego Jiménez, Jesús Colín, Andrea Neftalí, Gandhi Peña y Víctor Robles que, con nuestra amistad y apoyo incondicional, logramos terminar este camino.

A mi amiga, Fernanda Ramírez, que nunca me dejo solo y siempre me motivó a seguir adelante.

Muchas gracias a todos.

# ÍNDICE

CAPÍTULO 1 .....	1
Introducción .....	1
Antecedentes .....	3
Estado del arte .....	3
Configuraciones cinemáticas de los Robots Móviles con Ruedas .....	7
CAPÍTULO 2 .....	9
Planteamiento del problema .....	9
2.1. Levantamiento .....	9
2.2. Hipótesis .....	9
2.3. Objetivo general .....	9
2.4. Objetivos específicos .....	10
CAPÍTULO 3 .....	11
Marco teórico .....	11
CAPÍTULO 4 .....	13
Desarrollo .....	13
4.1. Elección de controlador .....	14
4.2. Interfaz visual .....	16
4.2.1. Instalación y configuración de software .....	17
4.2.2. Uso de Virtual Network Computing Connect como acceso remoto .....	19
4.3. Acondicionamiento de señales obtenidas .....	24
4.4. Conversión analógico - digital .....	36
4.4.1 ADS 1115 .....	37
4.5. Programación .....	40
4.5.1. Programa de captura de fotografías .....	41
4.5.2. Programa de captura de video .....	44
4.5.3. Programa de lectura de sensores .....	47
CAPÍTULO 5 .....	50
Pruebas y resultados .....	50
5.1. Prueba de captura de imagen .....	50
5.2. Prueba de captura de videos .....	56
5.3. Prueba de lectura de sensores .....	59
CAPÍTULO 6 .....	61

Discusión.....	61
Conclusiones .....	62
Referencias.....	63
APÉNDICES.....	64
a) Diagramas de conexión del sistema de adquisición de datos .....	64
b) Sensor MQ-2 .....	68
c) Sensor MQ-9 .....	71
d) ADS 1115.....	74
e) Raspberry Pi 4 model B .....	78

# CAPÍTULO 1

## **Introducción**

La instrumentación electrónica tiene su campo de aplicación en numerosas actividades relacionadas con la ciencia y la tecnología donde la electrónica se incorpora de forma masiva. En el laboratorio y en la industria se realiza un uso intensivo de la instrumentación electrónica porque en estos entornos, como en otros muchos, es necesario conocer de manera exacta cualquier tipo de variable física.

Dentro de la instrumentación electrónica, los sistemas de adquisición de datos toman especial importancia. Estos cumplen la función de convertir datos del mundo real en información. Al integrar sensores y acondicionadores de señal, son capaces de procesar, exhibir y almacenar datos en una computadora.

Existen sistemas de adquisición de datos de línea industrial que son desarrollados por empresas multinacionales. Es aquí donde surge la oportunidad de desarrollar un sistema de adquisición de datos sobre una plataforma de bajo costo (comparado con sistemas de línea industrial).

En este trabajo se desarrolla la integración de un sistema de adquisición de datos sobre una plataforma de bajo costo, partiendo de la necesidad de conocer el entorno que rodea a un robot móvil.

En el capítulo 1, se analiza el antecedente histórico de la robótica móvil y hasta donde los sistemas de adquisición de datos han sido integrados en los mismos.

Después, en el capítulo 2, se realiza el planteamiento del problema y se delimita el alcance del proyecto estableciendo los objetivos de este.

En el tercer capítulo se crea un marco teórico, definiendo conceptos claves que ayudarán a tener una mejor comprensión del tema.

Durante el capítulo 4, se presenta como se llevó a cabo la instrumentación electrónica y programación del proyecto.

Posteriormente, en el capítulo 5, se realizan pruebas y se muestran los resultados de la instrumentación y los programas hechos en el capítulo 4.

Finalmente, en el capítulo 6, se realiza la discusión y conclusiones del tema enfatizando el uso de una plataforma de bajo costo. Además, se agregan los anexos al final del documento.

## **Antecedentes**

### **Estado del arte**

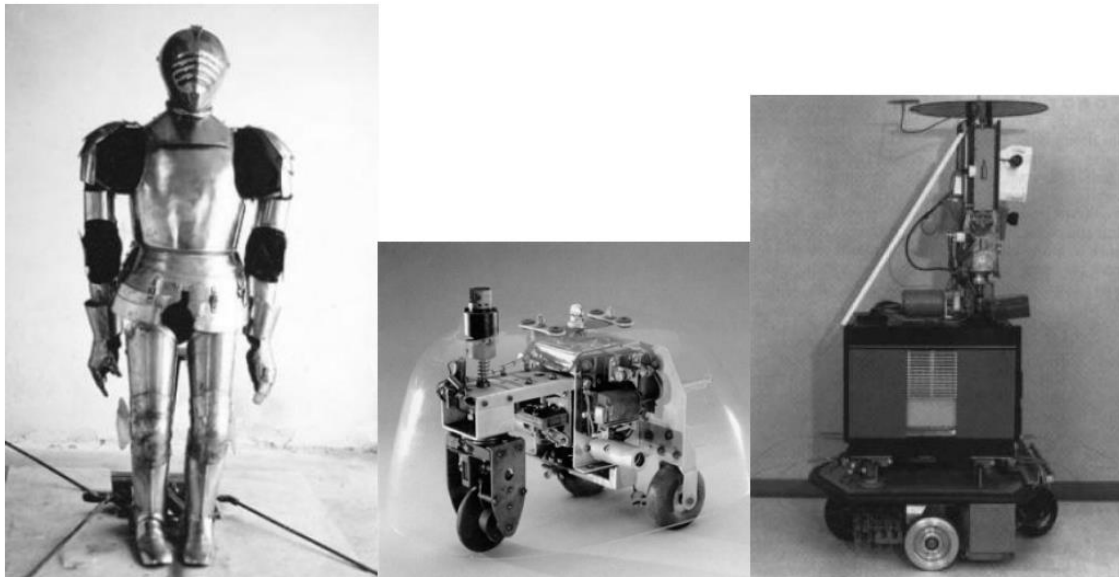
A principios de los años sesenta se introducen en la industria, de modo significativo, los robots manipuladores como un elemento más del proceso productivo. Esta proliferación, motivada por la amplia gama de posibilidades que ofrecía, suscitó el interés de los investigadores para lograr manipuladores más rápidos, precisos y fáciles de programar. La consecuencia directa de este avance originó un nuevo paso en la automatización industrial, que flexibilizó la producción con el nacimiento de la noción de célula de fabricación robotizada. Los trabajos desarrollados por los robots manipuladores consistían frecuentemente en tareas repetitivas, como la alimentación de las distintas máquinas componentes de la célula de fabricación robotizada. Ello exigía ubicarlas en el interior de un área accesible para el manipulador, caracterizada por la máxima extensión de sus articulaciones, lo cual podría resultar imposible a medida que la célula sufría progresivas ampliaciones. Una solución a este problema se logra al desarrollar un vehículo móvil sobre rieles para proporcionar un transporte eficaz de los materiales entre las distintas zonas de la cadena de producción. De esta forma, aparecen en los años ochenta los primeros vehículos guiados automáticamente (AGV's). Una mejora con respecto a su concepción inicial estriba en la sustitución de los rieles como referencia de guiado en la navegación por cables enterrados, reduciéndose, con ello, los costes de instalación.

La posibilidad de estructurar el entorno industrial permite la navegación de vehículos con una capacidad sensorial y de razonamiento mínimas. De este modo, la tarea se ordena en una secuencia de acciones en la que a su término el vehículo supone que ha alcanzado el objetivo para el que está programado. Ante cualquier cambio inesperado en el área de trabajo que afecte el desarrollo normal de la navegación, el sistema de navegación del vehículo se encontrará imposibilitado para ejecutar acciones alternativas que le permitan reanudar su labor. Sin embargo, por sus potenciales aplicaciones fuera del ámbito industrial, donde resulta costoso o imposible estructurar el entorno, se les dotó, en la búsqueda de un vehículo de propósito general apto para



desenvolverse en cualquier clase de ambiente, de un mayor grado de inteligencia y percepción. Así en los años noventa surge el **robot móvil** [1].

Una definición correcta de robot móvil plantea la capacidad de movimiento sobre entornos no estructurados, de los que se posee un conocimiento incierto, mediante la interpretación de la información suministrada a través de sus sensores y del estado actual del vehículo. En la figura 1.1 podemos observar tres robots que sirvieron de inspiración:



*Figura 1.1. Tres robots que sirvieron de inspiración.*

Los robots móviles son dispositivos de transporte automático, es decir, una plataforma mecánica dotada de un sistema de locomoción capaz de navegar a través de un determinado ambiente de trabajo, dotado de cierto nivel de autonomía para su desplazamiento portando cargas. Sus aplicaciones pueden ser muy variadas y siempre están relacionadas con tareas que normalmente son riesgosas o nocivas para la salud humana, en áreas como la agricultura, en el transporte de cargas peligrosas o en tareas de exploración solitarias o cooperativas junto a otros vehículos no tripulados. Ejemplos clásicos son el traslado y acopio de materiales, las tareas de mantenimiento en reactores nucleares, la manipulación de materiales explosivos, la exploración subterránea, etc. En la figura 1.2, 1.3, y 1.4 podemos observar algunos ejemplos:



*Figura 1.2. Inspector Bot de Eventronic ES.*



*Figura 1.3. Sistema de transporte de Material Automatizado (AMTS por sus siglas en inglés).*



*Figura 1.4. Robot militar Warrior X700 de iRobot.*

Es necesario aclarar que el concepto de autonomía no sólo se relaciona con cuestiones energéticas, sino que también se refiere a la capacidad de percibir, modelar, planificar y actuar para alcanzar determinados objetivos, sin la intervención (o con una intervención muy pequeña) del operador humano ya que el robot se puede desenvolver en ambientes estructurados o no estructurados, total o parcialmente conocidos. El papel de este debe ser desempeñado por el propio sistema de control del vehículo, al que debe suplir con la inteligencia necesaria para mover al robot correctamente. La denominación de robot móvil hace referencia a esa capacidad para alcanzar uno o varios objetivos con una intervención muy pequeña de supervisores humanos. Por otro lado, la denominación de vehículo autoguiado está referida a las estructuras móviles que sólo se limitan a seguir caminos preestablecidos (líneas pintadas en el suelo, bandas magnéticas, bandas reflectoras) [2].

Los robots móviles brindan la posibilidad de navegar en distintos terrenos y tienen aplicaciones como: exploración minera, exploración planetaria, misiones de búsqueda y rescate de personas, limpieza de desechos peligrosos, automatización de procesos, vigilancia, reconocimiento de terreno, y también son utilizados como plataformas móviles que incorporan un brazo manipulador.

Los robots móviles se pueden clasificar por el tipo de locomoción utilizado, en general, los tres medios de movimiento son: por ruedas, por patas y orugas. Cabe señalar que, aunque la locomoción por patas y orugas han sido

ampliamente estudiadas, el mayor desarrollo se presenta en los Robots Móviles con Ruedas (RMR) [3]. Dentro de los atributos más relevantes de los RMR, destacan su eficiencia en cuanto a energía en superficies lisas y firmes, a la vez que no causan desgaste en la superficie donde se mueven y requieren un número menor de partes y menos complejas, en comparación con los robots de patas y de orugas, lo que permite que su construcción sea más sencilla. Son precisamente estos argumentos los que motivan el análisis de este tipo de robots, y surge la necesidad, en primera instancia, de tener una definición que satisfaga el contexto de los RMR. De esta manera, se puede definir un robot móvil de ruedas como un sistema electromecánico controlado, que utiliza como locomoción ruedas de algún tipo, y que es capaz de trasladarse de forma autónoma a una meta preestablecida en un determinado espacio de trabajo.

### Configuraciones cinemáticas de los Robots Móviles con Ruedas

Existen diferentes configuraciones cinemáticas para los RMR, estas dependen principalmente de la aplicación hacia dónde va enfocado, no obstante, de manera general se tienen las siguientes configuraciones: Ackerman, triciclo clásico, tracción diferencial, skid steer, síncrona y tracción omnidireccional (véase Figura 1.5).



Figura 1.5. Configuraciones de los RMR.

Dependiendo de la configuración cinemática que lo conforme, los RMR utilizan cuatro tipos de ruedas para su locomoción, estas son: convencionales, tipo castor, ruedas de bolas y omnidireccionales. En el marco de las configuraciones cinemáticas posibles y las ruedas que estas utilizan, los RMR documentados en

la literatura utilizan comúnmente la configuración de tracción diferencial, (véase Figura 1.5. c), donde se utilizan ruedas convencionales (Figura 1.6. a), como ruedas motrices y una o dos ruedas tipo castor (Figura 1.6. b), de bola (Figura 1.6. c), u omnidireccionales (Figura 1.6. d) para proveer de estabilidad al móvil.



a) Convencionales



b) Castor



c) De bolas



d) Omnidireccionales

*Figura 1.6. Tipos de locomoción.*

# CAPÍTULO 2

## Planteamiento del problema

### 2.1. Levantamiento

- *Problema:*

El Omnimóvil [4] no tiene la capacidad de reconocer las variables físicas de su entorno y mostrarlas al usuario.

- *Necesidad:* Conocer las áreas caóticas para ayudar a que el usuario esté seguro.
- *Usuario:* Personal del departamento de bomberos de Cuauhtémoc “Comandante Eulalio Mújica Pérez”.
- *Cliente:* Departamento de bomberos
- *Contacto:* Comandante Jesús Guerrero.
- *Entorno:* El robot deberá ser capaz de estar en entornos caóticos.
- *Proceso:* El sistema implementado en el robot leerá las variables físicas en su entorno para después acondicionar la señal y ser mostrada en una interfaz visual.

### 2.2. Hipótesis

Es posible integrar un sistema de adquisición de datos sobre una plataforma de bajo costo (comparado con tarjetas de adquisición de datos de línea industrial), capaz de adquirir información del entorno en el que se encuentre un robot móvil, procesarla y mostrarla al operador del sistema.

### 2.3. Objetivo general

Implementar un sistema de adquisición de datos en un robot móvil de manera que identifique diversas variables físicas en su entorno y sean mostradas al usuario, tales como: sustancias nocivas para la salud, imagen y video del entorno mismo.

## **2.4. Objetivos específicos**

- Implementar la instrumentación electrónica necesaria para la adquisición de datos.
- Hacer el procesamiento de las señales obtenidas.
- Diseñar una HMI adecuada para el despliegue de información esencial para el operador.

# CAPÍTULO 3

## Marco teórico

Tabla 1

### Glosario de términos

Concepto	Descripción	Autor
Sensor	“Es un dispositivo que, a partir de la energía del medio donde se mide, da una señal de salida transducible que es función de la variable medida”.	(Ramón, Pallás, Areny, 2003, p. 3.) [5]
Transductor	“Se denomina transductor, en general, a todo dispositivo que convierte una señal de una forma física en una señal correspondiente, pero de otra forma física distinta”.	(Ramón, Pallás, Areny, 2003, p. 2.) [5]
Acondicionadores de señal	“Son circuitos que convierten los parámetros eléctricos de salida de los transductores en una señal eléctrica que se puede medir fácilmente”.	(Mercedes, Granda, Miguel, 2015, p. 6.) [6]
Ancho de banda	“Es la diferencia, medida en Hertz, entre la máxima y la mínima frecuencia que puede pasar por un canal de comunicaciones”.	(Marcelo, Semeria, 2015, p. 5.) [6]
Frecuencia de Nyquist	“Toda señal limitada en banda (debido a que se encuentra acotada por el AB del canal) se puede reconstruir completamente a partir de las muestras tomadas de misma, siempre que la velocidad del muestreo se realice como	(Marcelo, Semeria, 2015, p. 10.) [7]



	mínimo al doble de la máxima frecuencia de la señal”.	
HMI	“Un HMI (Human Machina Interface), es un sistema computarizado para dar acceso a los parámetros de control de un proceso”.	(Castro, Bazúa, Aaron, 2012, p. 358.) [8]
Protocolo de comunicación	“Se define como el proceso mediante el cual la información se transfiere desde un lugar fuente a un destino, donde interviene los siguientes elementos: transductor, emisor o transmisor, medio o canal de transmisión, receptor”.	(Sanchis, Enrique, 2004, p. 13.) [9].
Microcontrolador	“Es un circuito integrado, el cual está formado por un microprocesador y una memoria interna”.	(Santamaria, Eduardo, 1993, p. 15.) [10].
Adquisición de datos	“La adquisición de datos es un proceso mediante el cual fenómenos físicos del mundo real son transformados en señales eléctricas. Estas señales son medidas y convertidas en formato digital para su procesamiento y manipulación en una computadora”.	(Di Paolo, Maurizio, 2013, p. 11.) [11].
Instrumentación	"La instrumentación comprende todas las técnicas, equipos y metodologías relacionadas con el diseño, la construcción y la aplicación de dispositivos físicos para mejorar, completar y aumentar la eficiencia de los mecanismos de percepción del ser humano".	(Ferrero, Javier, 2004, p.1.) [12].

# CAPÍTULO 4

## Desarrollo

El desarrollo del proyecto comenzó con la definición de las variables físicas que se van a adquirir del entorno en que se encuentre el robot. Para esta etapa se requirió de una consulta con el que será el usuario del sistema. En este caso el comandante Jesús Guerrero del departamento de bomberos de Cuauhtémoc “Comandante Eulalio Mújica Pérez” ubicado en la delegación Cuauhtémoc, CDMX.

Después de conocer las necesidades del usuario se determinó lo siguiente:

- El sistema debe ser capaz de mostrar imagen y video desde la perspectiva del robot.
- Se necesita saber si en el entorno del robot hay presencia de sustancias nocivas para la salud.
- La interfaz del sistema debe ser sencilla de usar.

Para saber qué sustancias eran necesarias detectar se hizo uso de la guía de respuesta en caso de emergencia proporcionada por la Secretaría de Comunicaciones y Transportes del gobierno de México y con base en la consulta que se realizó al usuario, se determinó que el Monóxido de Carbono y el Gas LP serán las variables por adquirir.

El diseño del sistema se clasificó en cuatro categorías:

- Elección del controlador
- Uso de la interfaz visual
- Acondicionamiento de las señales obtenidas
- Programación

## 4.1. Elección de controlador

Para la elección del controlador se tomaron en cuenta los siguientes aspectos:

- Rendimiento
- Compatibilidad de periféricos
- Tipo de software
- Costo

Una vez determinados estos parámetros, se llegó a la elección de utilizar el Raspberry Pi 4 model B (Figura 4.1) como controlador para nuestro sistema de adquisición de datos.

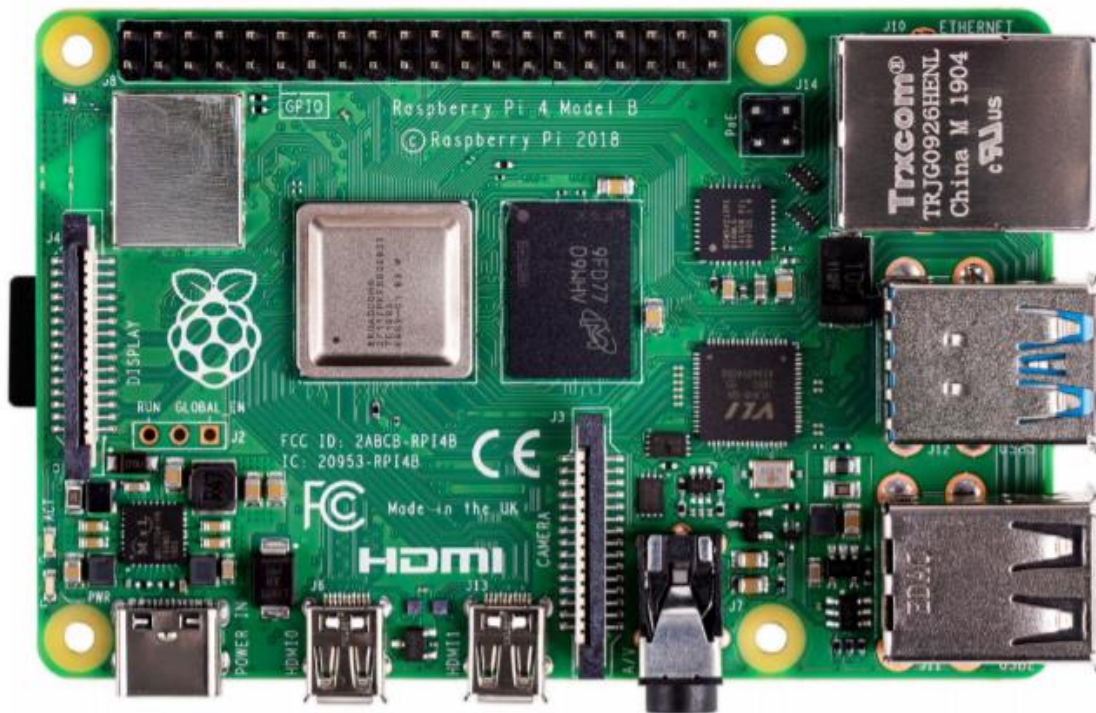


Figura 4.1. Raspberry Pi 4 model B.

El Raspberry Pi es un miniordenador de bajo costo desarrollado por Raspberry Pi Foundation. Las principales ventajas encontradas en este controlador son su diversa compatibilidad de periféricos y su alto rendimiento para el procesamiento de tareas aun siendo utilizado de manera remota.

A continuación, en la figura 4.2, se muestran las características de las distintas versiones de hardware en los diferentes modelos de Raspberry Pi:

	RASPERRY PI 4 MODEL B	RASPERRY PI 3 MODEL B+	RASPERRY PI 3 MODEL B
PROCESADOR	Broadcom BCM2711B0, quad-core Cortex-A72	Broadcom BCM2837B0, Cortex-A53 (ARMv8) 64-bit SoC	Broadcom BCM2837, Cortex-A53 (ARMv8) 64-bit SoC
FRECUENCIA DE RELOJ	1,5 GHz	1,4 GHz	1,2 GHz
GPU	VideoCore VI 500 MHz	VideoCore IV 400 MHz	VideoCore IV 400 MHz
MEMORIA	1/2/4 GB LPDDR4-3200	1GB LPDDR2 SDRAM	1GB LPDDR2 SDRAM
CONECTIVIDAD INALÁMBRICA	Wi-Fi 2,4GHz / 5GHz IEEE 802.11.b/g/n/ac Bluetooth 5.0, BLE	Wi-Fi 2,4GHz / 5GHz IEEE 802.11.b/g/n/ac Bluetooth 4.2, BLE	Wi-Fi 2,4GHz IEEE 802.11.b/g/n Bluetooth 4.1
CONECTIVIDAD DE RED	Gigabit Ethernet	Gigabit Ethernet over USB 2.0 (300 Mbps de máximo teórico)	Fast Ethernet 10/100 Gbps
PUERTOS	GPIO 40 pines 2 x Micro HDMI 2 x USB 2.0 2 x USB 3.0 CSI (cámara Raspberry Pi) DSI (pantalla tácil) Toma auriculares / video compuesto Micro SD Micro USB (alimentación) USB-C (alimentación) Power-over-Ethernet (PoE)	GPIO 40 pines HDMI 4 x USB 2.0 CSI (cámara Raspberry Pi) DSI (pantalla tácil) Toma auriculares / video compuesto Micro SD Micro USB (alimentación) Power-over-Ethernet (PoE)	GPIO 40 pines HDMI 4 x USB 2.0 CSI (cámara Raspberry Pi) DSI (pantalla tácil) Toma auriculares / video compuesto Micro SD Micro USB (alimentación)

*Figura 4.2. Características de Raspberry Pi*

Debido a la necesidad de transmitir información de manera remota y a su mayor capacidad de procesamiento, se eligió para este proyecto utilizar el modelo 4 B siendo la versión con 2 GB de memoria RAM la adecuada para el propósito de este proyecto. La Raspberry Pi se alimenta a 5V y 3 Amperes por medio de un cable USB tipo C, lo cual facilita su implementación en el Omnimóvil [8].

Otra de las características más relevantes del Raspberry Pi, son sus 40 pines de propósito general (GPIO), entre los cuales podemos encontrar pines dedicados a la alimentación (5 y 3.3 Volts), pines dedicados a protocolos de comunicación (UART, I2C y SPI) y el resto de los pines que pueden ser configurados como I/O. Esta distribución la podemos observar en la figura 4.3:



Figura 4.3. Distribución de pines GPIO de Raspberry Pi.

Para poder configurar y utilizar de manera óptima el controlador, fue necesario adquirir los siguientes materiales:

- Tarjeta microSD con capacidad de 16 GB
- Cable tipo micro HDMI - HDMI
- Cable USB tipo C
- Case para Raspberry Pi
- Ventilador para procesador Raspberry Pi

Toda la lista de materiales, sumado al precio del Raspberry Pi 4 model B con 2 GB de RAM, nos generó un costo de \$1,300.00 MXN (2021).

## 4.2. Interfaz visual

Una de las ventajas de utilizar el Raspberry Pi como controlador, es la variedad de software y la libertad de configuraciones que se pueden utilizar.

Para utilizar la Raspberry Pi, se puede elegir alguno de los siguientes softwares:

- Raspberry Pi OS with desktop and recommended software
- Raspberry Pi OS with desktop
- Raspberry Pi OS Lite
- Debian Buster with Raspberry Pi Desktop
- Ubuntu Desktop
- Ubuntu Server

Dadas las características de cada sistema operativo, y la necesidad de tener una interfaz gráfica que sea sencilla de utilizar para el usuario y pueda

funcionar de manera remota, se optó por instalar el Raspberry Pi OS with desktop and recommended software.

Ya seleccionado el controlador y el software, iniciamos el proceso de configuración de este.

#### **4.2.1. Instalación y configuración de software**

Para poder configurar nuestro controlador es necesario descargar e instalar nuestro sistema operativo en nuestra memoria SD. En este caso se instaló Raspberry Pi OS with desktop and recommended software con un peso de 2,868 MB. Posteriormente insertamos nuestra memoria microSD en el puerto para la tarjeta de nuestro Raspberry.

En la configuración inicial de la Raspberry es necesario utilizar los siguientes periféricos:

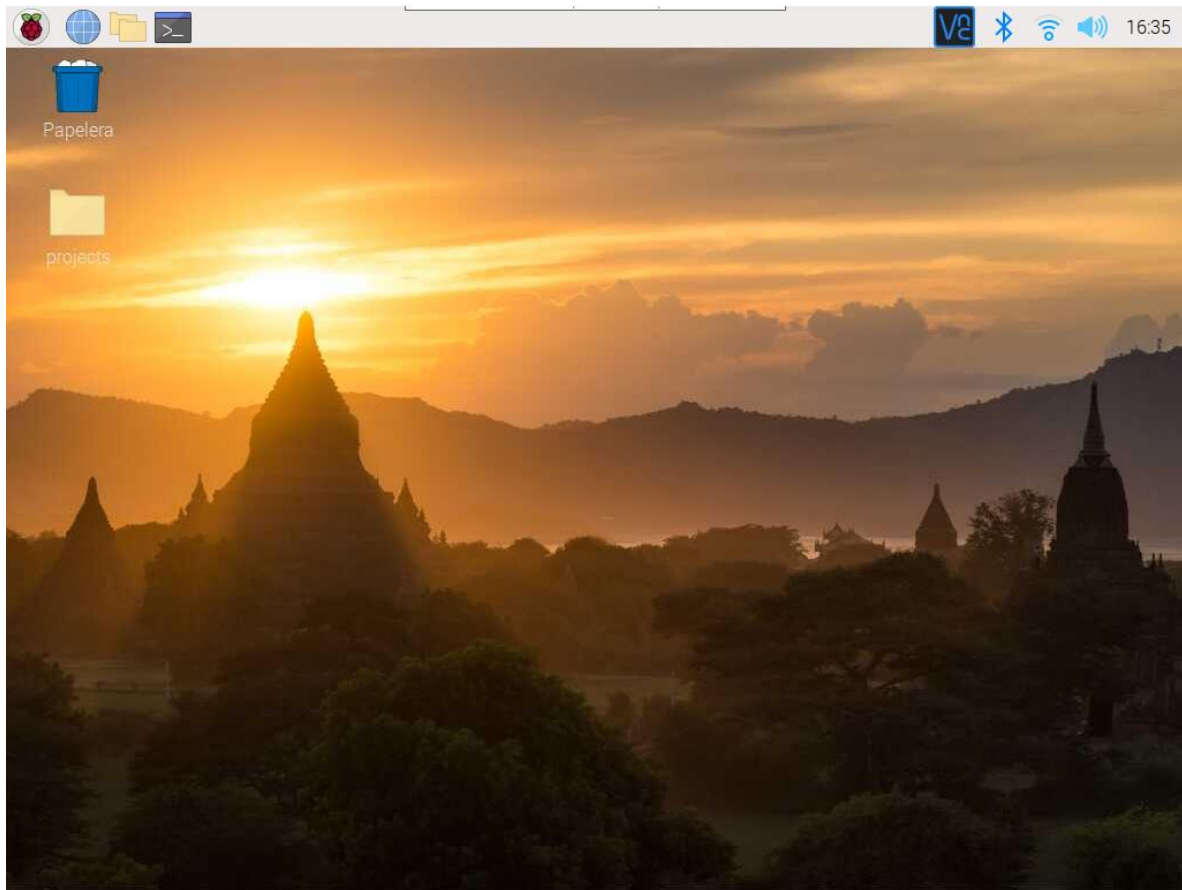
- Teclado
- Mouse
- Monitor/TV

Una vez conectados estos periféricos, conectaremos un cable ethernet para poder tener acceso a internet y nuestro cable de alimentación USB C (véase Figura 4.4) para poder utilizar la Raspberry.



*Figura 4.4 Configuración inicial de Raspberry Pi.*

Ya con el sistema configurado, rescataremos nuestra dirección IP para poder acceder a nuestro Raspberry de manera remota a través de VNC viewer (Figura 4.5).



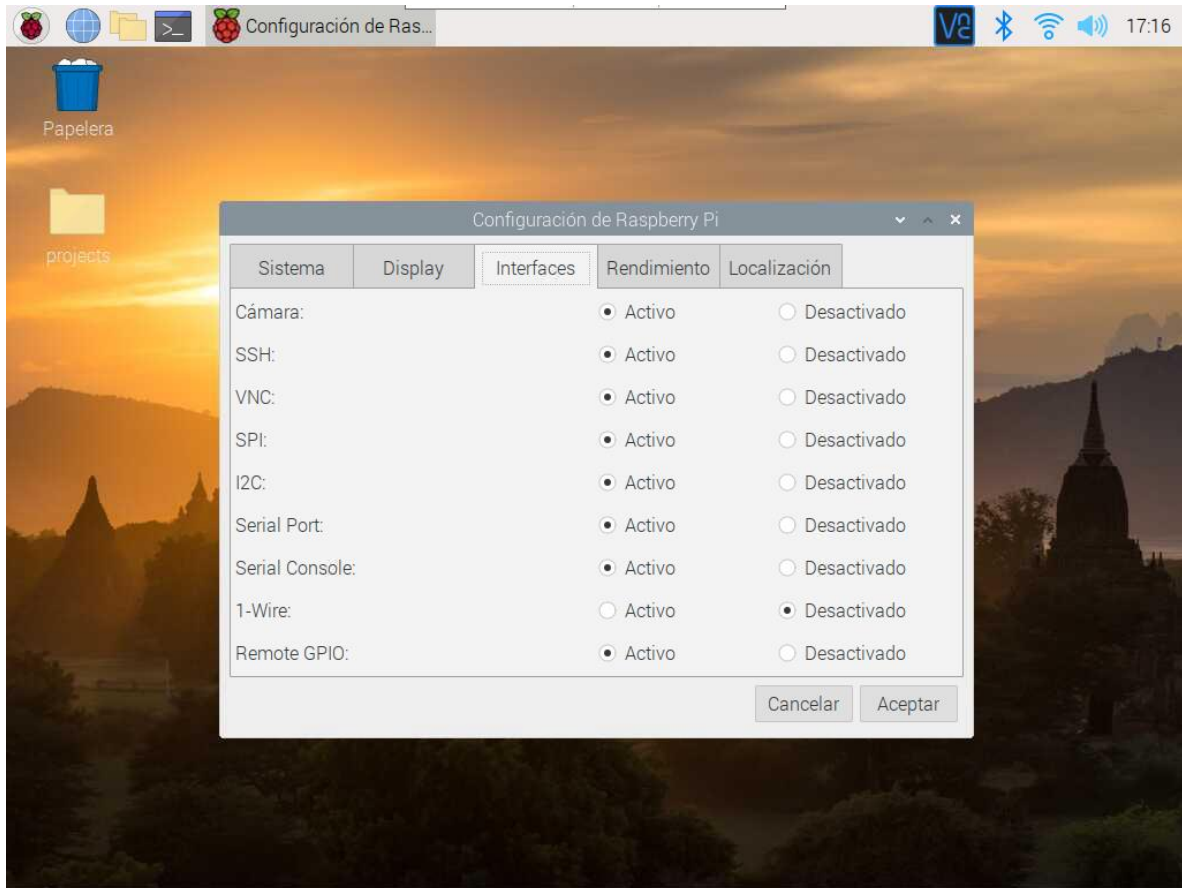
*Figura 4.5. Interfaz gráfica de Raspberry Pi OS with desktop.*

#### **4.2.2. Uso de Virtual Network Computing Connect como acceso remoto**

VNC Connect es un software de pantalla compartida que permite establecer una conexión con un equipo remoto situado en cualquier parte del mundo. Una vez instalado VNC en el dispositivo, es posible acceder a él desde cualquier parte del mundo a través de Internet y desde cualquier dispositivo, como ordenadores o smartphones.

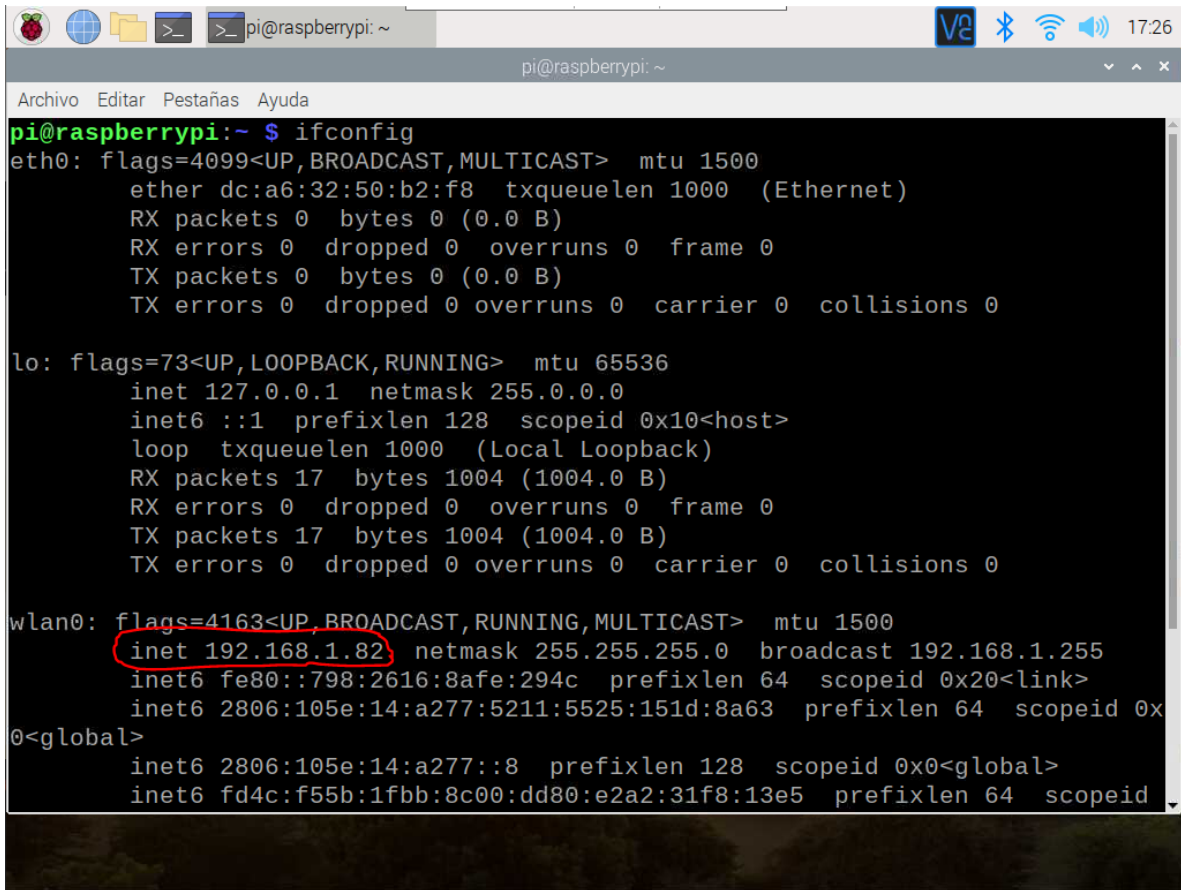
Para poder realizar una conexión remota es necesario habilitar el apartado de conexiones remotas en la configuración de la Raspberry Pi (véase Figura 4.6):





*Figura 4.6. Habilitación de conexión remota.*

Para realizar la conexión remota es indispensable conocer la dirección IP de nuestra Raspberry Pi, la cual podemos consultar por medio del monitor serial como se muestra en la figura 4.7:



```
pi@raspberrypi: ~  
Archivo Editar Pestañas Ayuda  
pi@raspberrypi:~ $ ifconfig  
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500  
    ether dc:a6:32:50:b2:f8 txqueuelen 1000 (Ethernet)  
    RX packets 0 bytes 0 (0.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 0 bytes 0 (0.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 1000 (Local Loopback)  
    RX packets 17 bytes 1004 (1004.0 B)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 17 bytes 1004 (1004.0 B)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 192.168.1.82 netmask 255.255.255.0 broadcast 192.168.1.255  
    inet6 fe80::798:2616:8afe:294c prefixlen 64 scopeid 0x20<link>  
    inet6 2806:105e:14:a277:5211:5525:151d:8a63 prefixlen 64 scopeid 0x  
0<global>  
    inet6 2806:105e:14:a277::8 prefixlen 128 scopeid 0x0<global>  
    inet6 fd4c:f55b:1fbb:8c00:dd80:e2a2:31f8:13e5 prefixlen 64 scopeid
```

Figura 4.7. Dirección IP del controlador.

La dirección IP de la Raspberry será como vamos a identificar a nuestro controlador en la aplicación de VNC viewer. Una vez que instalamos el software de VNC viewer en el dispositivo desde el cual vamos a utilizar de manera remota nuestro controlador, se añadió el dispositivo con las direcciones necesarias para poder iniciar comunicación (Figura 4.8):

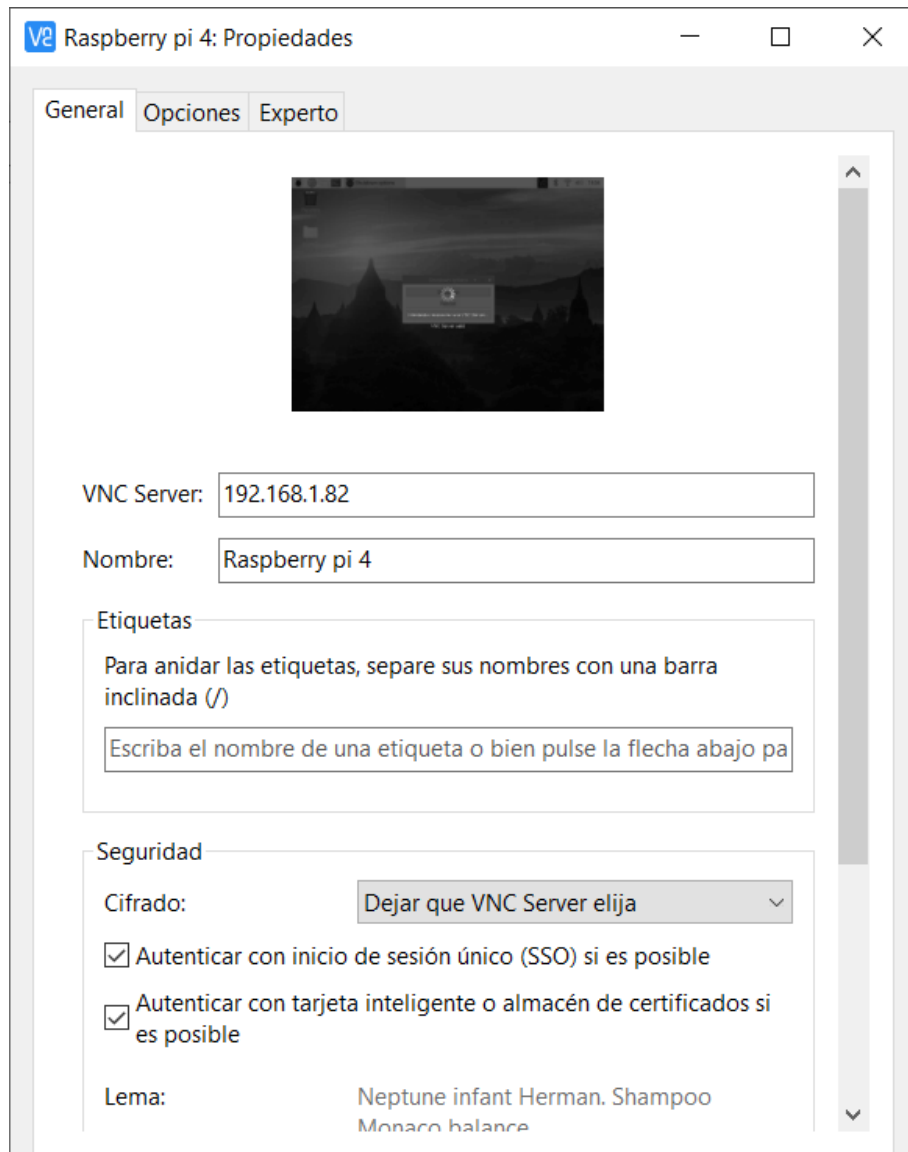
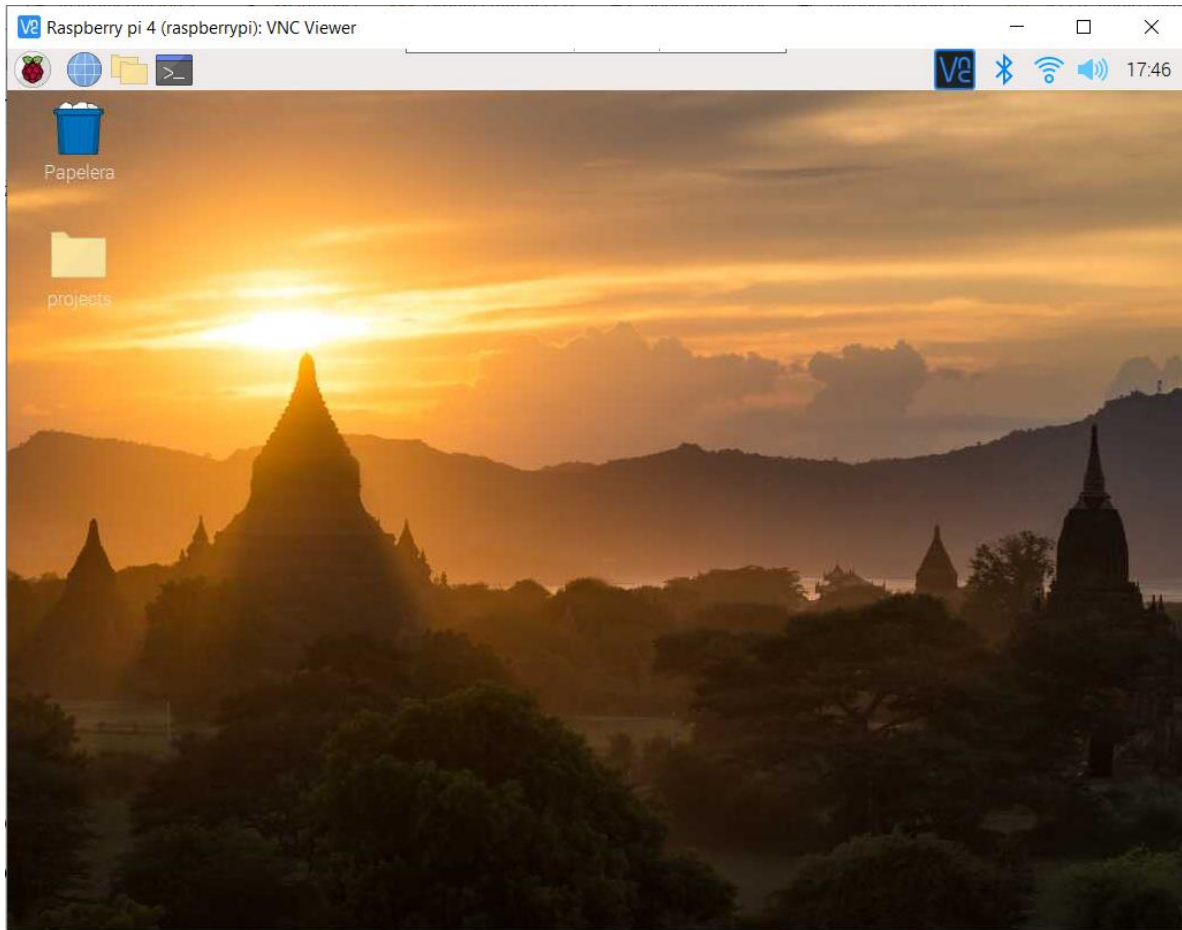


Figura 4.8. Configuración en VNC viewer.

Realizada la configuración y una vez que se establece conexión es posible utilizar la Raspberry de manera remota (véase Figura 4.9).



*Figura 4.9. Uso remoto de nuestro controlador.*

### **4.3. Acondicionamiento de señales obtenidas**

La señal de salida de un sistema de medición en general se debe procesar de una forma adecuada para la siguiente etapa de la operación. La señal puede ser por ejemplo demasiado pequeña, y sería necesario amplificarla; podría contener interferencias que eliminar; ser no lineal y requerir su linealización; ser análoga y requerir su digitalización; ser digital y convertirla en análoga; ser un cambio de voltaje y convertirla a un cambio de corriente de magnitud adecuada, etc. A todas estas modificaciones se les designa por lo general con el término “Acondicionamiento de Señales”.

La parte fundamental de la adquisición de datos es el acondicionamiento de las señales que se quieren obtener. Los acondicionadores de señal “Son circuitos que convierten los parámetros eléctricos de salida de los transductores en una señal eléctrica que se puede medir fácilmente” (Mercedes, Granda, Miguel, 2015, p. 6.). [6].

Existen distintos tipos de acondicionamiento de señal que como se mencionó anteriormente, dependerá directamente de cómo sea la señal que se quiere adquirir. Entre ellos están:

#### *Aislamiento*

Las señales de voltaje fuera del rango del digitalizador pueden dañar el sistema de medición y ser peligrosas para el operador. Por esta razón, normalmente es preciso tener el aislamiento y la atenuación para proteger al sistema y al usuario de voltajes de alta tensión o picos. También se puede necesitar aislamiento si el sensor está en un plano de tierra diferente al del sensor de medición (como un termopar montado en una máquina).

#### *Amplificación*

Cuando los niveles de voltaje que va a medir son muy pequeños, la amplificación se usa para maximizar la efectividad de su digitalizador. Al amplificar la señal de entrada, la señal acondicionada usa más efectivamente el rango del convertidor analógico-digital (ADC) y mejora la precisión y resolución de la medición. Algunos sensores que típicamente requieren de amplificación son los termopares y galgas extensiométricas.

### *Atenuación*

La atenuación es lo opuesto a la amplificación. Es necesario cuando el voltaje que se va a digitalizar es mayor al rango de entrada del digitalizador. Esta forma de acondicionamiento de señal disminuye la amplitud de la señal de entrada, para que la señal acondicionada esté dentro del rango del ADC. La atenuación es necesaria para medir voltajes altos.

### *Multiplexado*

Típicamente, el digitalizador es la parte más costosa del sistema de adquisición de datos. Al multiplexar, usted puede rutear secuencialmente un cierto número de señales a un solo digitalizador, logrando así un sistema de bajo costo y extendiendo el número de conteo de señales de su sistema. El multiplexeo es necesario para cualquier aplicación de alto conteo de canales.

### *Filtrado*

Los filtros son necesarios para remover cualquier componente de frecuencia no deseada en una señal, principalmente para prevenir aliasing y reducir la señal de ruido. Algunas mediciones de termopares generalmente requieren de filtros pasa bajos para remover el ruido de las líneas de poder. Las mediciones de vibración normalmente requieren de filtros antialiasing para remover componentes de señales más allá del rango de frecuencias del sistema de adquisición de datos.

### *Linealización*

Algunos tipos de sensores producen señales de voltaje que no son lineales en relación con la cantidad física que están midiendo. La linealización, el proceso de interpretar la señal del sensor como una medición física, puede realizarse a través de acondicionamiento de señal o software. Los termopares son un ejemplo típico de un sensor que requiere linealización.

### *Conversión analógico - digital (ADC)*

La mayoría de los sensores que encontramos en el mundo real son analógicos, es decir, modifican algún parámetro constructivo en función de la magnitud física que tratamos de medir. Si bien la mayoría de los sensores son analógicos, todos los microprocesadores, con los que trabajamos y que son los

que deben procesar la información suministrada por el sensor, son digitales, es decir, solo pueden procesar señales binarias. Un conversor analógico a digital nos permite convertir una señal de origen analógico en una señal digital que si puede ser procesada por un microcontrolador.

#### *Conversión digital - analógica (DAC)*

Un convertidor digital a analógico (DAC) es un dispositivo que convierte los datos digitales en una señal analógica que es una tensión, corriente o carga eléctrica. Los datos digitales generalmente son una secuencia de impulsos de tiempo finitos que se procesan y convierten en una señal analógica física continua.

Para determinar qué tipo de acondicionamiento de señal se tiene que realizar, primero, se revisaron nuestros sensores y el tipo de señal que estos nos dan.

Dadas las variables físicas que queremos conocer (Monóxido de Carbono, Gas LPG, imagen y video de nuestro entorno) se seleccionaron los siguientes sensores y cámara para poder adquirir las variables físicas deseadas:

- Sensor MQ-2
- Sensor MQ-9
- Cámara OV5647 con leds Infrarrojos para Raspberry PI.

#### *Cámara OV5647*

La cámara OV5647 (Figura 4.10) con leds Infrarrojos está basada en el sensor OV5647 de 5 megapíxeles diseñada por Raspberry Pi Foundation. Incluye dos focos LED infrarrojos para permitir la visión nocturna. No tiene un filtro infrarrojo (NoIR) porque está diseñada para visión nocturna por lo que, cuando se usa con luz de día los colores se verán un tanto extraños.



*Figura 4.10. Cámara OV5647 con leds Infrarrojos.*

*Características:*

- 5 megapíxeles
- Módulo fotosensible: OV5647
- Apertura (f): 1.8
- Distancia focal: 3.6mm (ajustable)
- Máxima resolución: 1080p
- Alimentación externa: 3.3V
- Dimensiones: 25mm x 24mm x 6mm
- Cable conector 15 PIN FFC

La principal ventaja de utilizar este dispositivo diseñado por Raspberry Pi Foundation es que no es necesario realizar algún tipo de acondicionamiento de señal para poder adquirir la variable física (Imagen y video en este caso) y que el usuario pueda ver la imagen tal cual la captura la cámara. En la figura 4.11 se observa el puerto en nuestro controlador especialmente dedicado a la cámara:



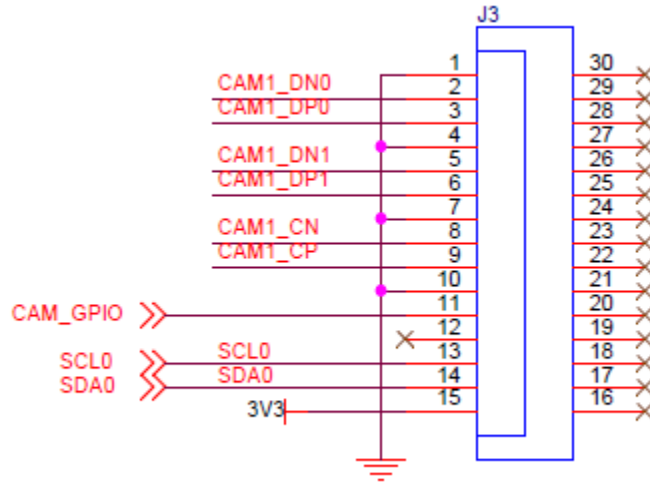


Figura 4.11. Diagrama de conexión del puerto para la cámara.

La cámara se comunica con el microcontrolador a través de un conector FFC (Figura 4.12) que es un cable flexible, lo que permite colocar la cámara en la posición que se desee.

La única configuración necesaria para utilizar la cámara es activar el protocolo de comunicación con la cámara desde el apartado de interfaces (véase Figura 4.6.).



Figura 4.12. Cámara OV5647 con leds infrarrojos conectada a la Raspberry Pi.

## Sensores MQ

Los sensores MQ son una serie de sensores electroquímicos que, a partir de entrar en contacto con determinada sustancia, generan una diferencia de voltaje.

Dentro de la serie de sensores MQ, existen diferentes modelos que son sensibles a determinados compuestos, en la figura 4.13 se muestran los distintos modelos:

Modelo	Sustancias detectadas	Calentador
MQ-2	Metano, butano, GLP, humo	5V
MQ-3	Alcohol, Etanol, humo	5V
MQ-303A	Alcohol, etanol, humo	0.9V
MQ-4	Metano, gas natural comprimido (GNP)	5V
MQ-5	Gas natural, GLP	5V
MQ-6	Butano, GLP	5V
MQ-306A	Butano, GLP	0.9V
MQ-7	Monóxido de carbono	Alternado 5V y 1.4V
MQ-307A	Monóxido de carbono	Alternado 0.2 y 0.9V
MQ-8	Hidrógeno	5V
MQ-9	Monóxido de carbono, gases inflamables	Alternado 5V y 1.5V

Figura 4.13. Serie de sensores MQ.

Para el objetivo de este proyecto, es necesario conocer cuando exista presencia de Monóxido de Carbono y gas LPG (GLP), por lo tanto, los sensores seleccionados son el MQ-2 y MQ-9 (véase Figura 4.14).

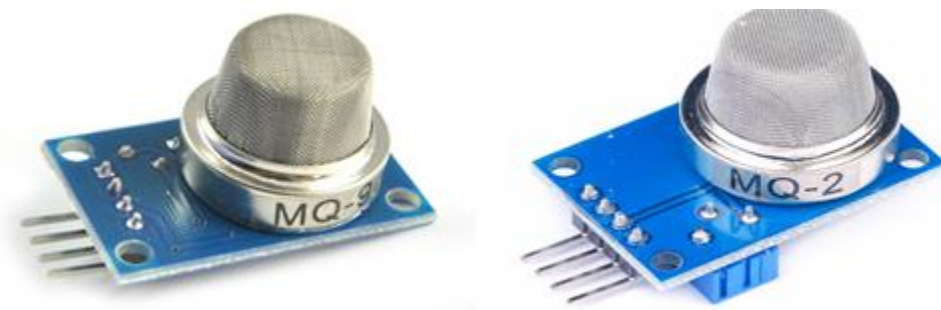


Figura 4.14. Sensores MQ-9 y MQ-2.

Ambos sensores funcionan bajo el mismo principio de funcionamiento (véase Figura 4.15). Están compuestos por dos bobinas de platino, ambas encapsuladas en un material cerámico de alúmina. Uno de estos encapsulados está cubierto de un material catalizador -normalmente de paladio- que causa y acelera la oxidación del elemento (esta parte es conocida como elemento detector) mientras que el otro encapsulado no tiene ese material para la oxidación del gas (esta parte se conoce como elemento de referencia), por lo que es inerte.

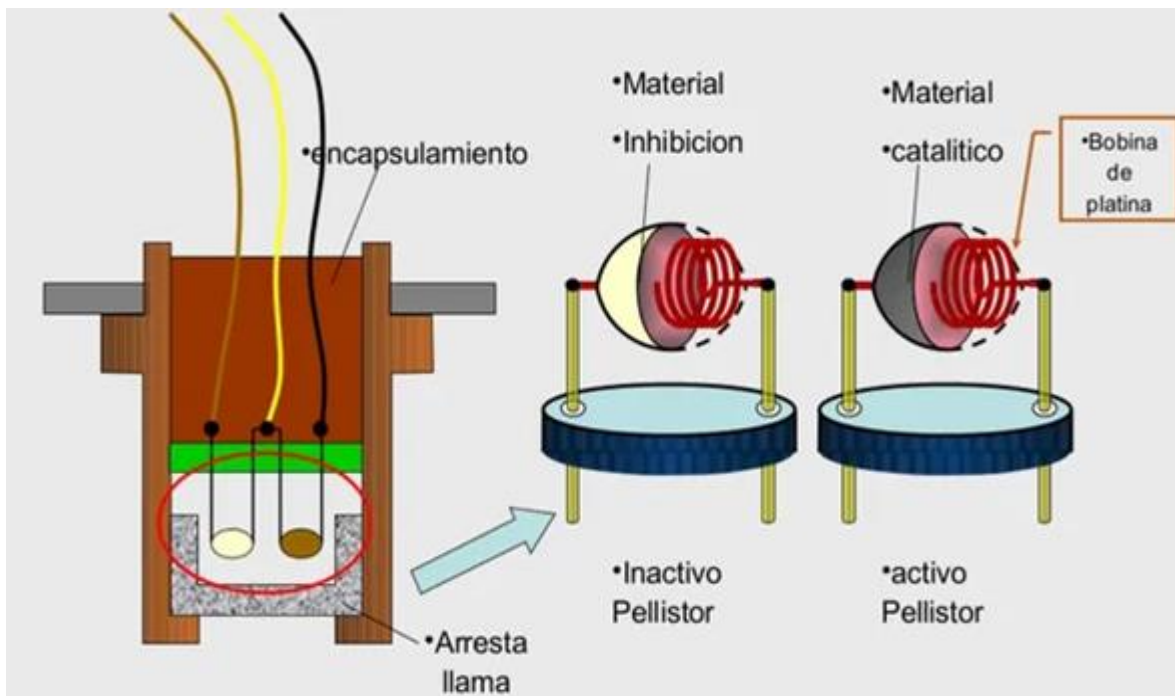


Figura 4.15. Principio de funcionamiento de sensores MQ.

La diferencia entre el encapsulado de uno y otro sensor es lo que determina sus características de sensibilidad. En la figura 4.16 y 4.17 podemos observar las características de sensibilidad de los sensores que se utilizarán:

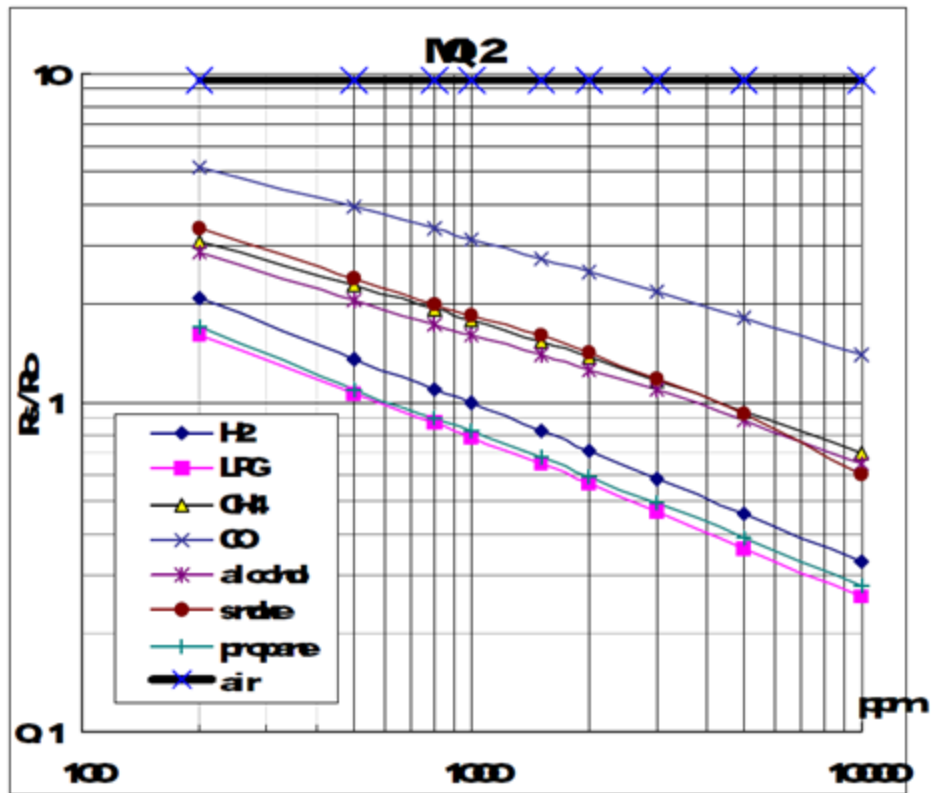


Figura 4.16. Característica de sensibilidad del sensor MQ-2.

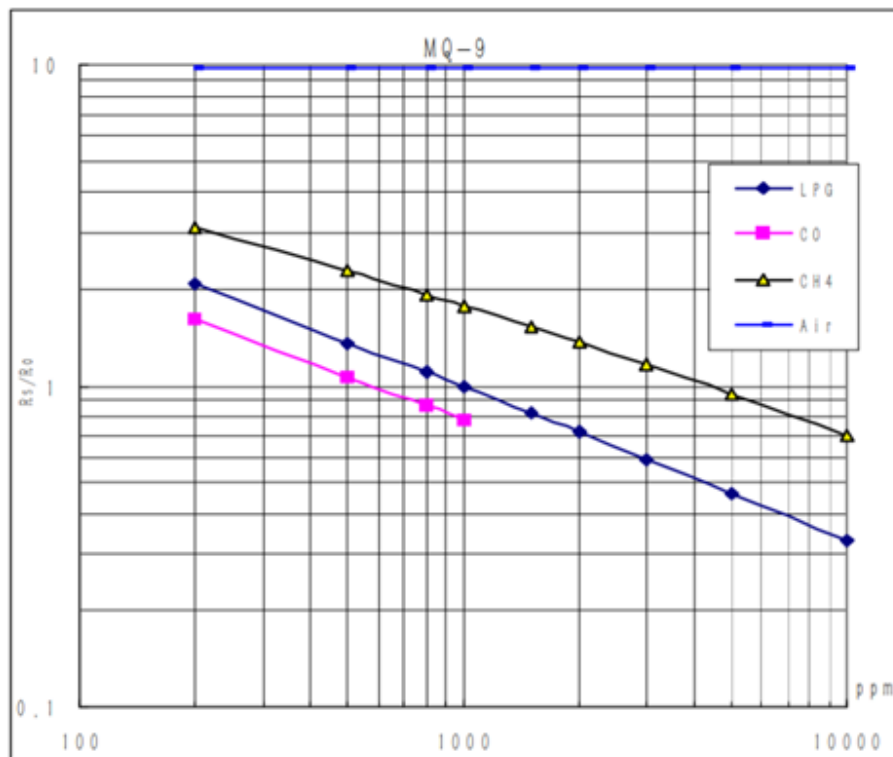
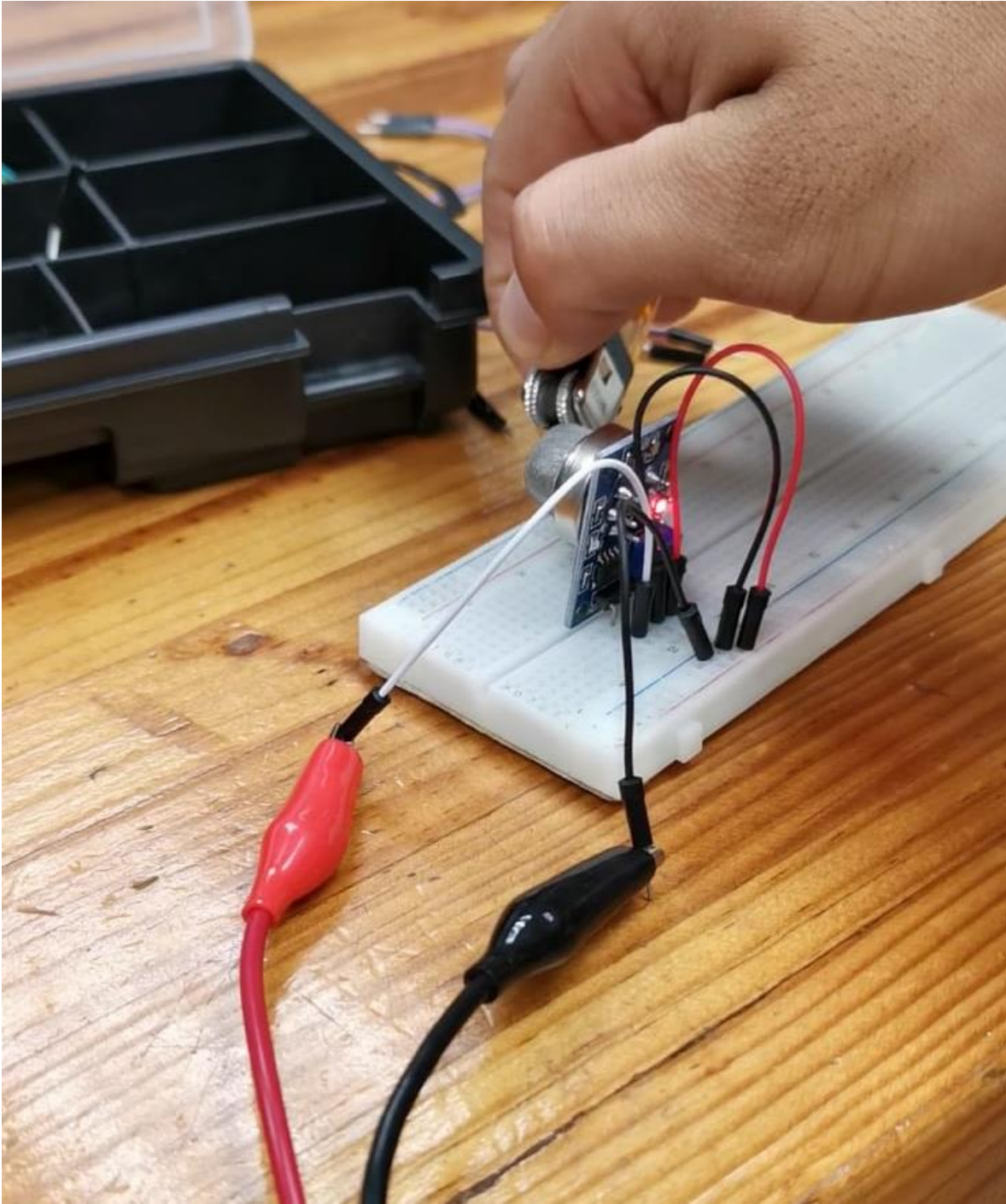


Figura 4.17. Características de sensibilidad del sensor MQ-9.

Para poder determinar el tipo de acondicionamiento de señal necesario para nuestros sensores, se tuvo que analizar la señal entregada por los mismos en un osciloscopio en el laboratorio L3 de la Fes Aragón (véase Figura 4.18).



*Figura 4.18. Experimento realizado en el laboratorio L3.*

Para este experimento se sometió a ambos sensores a la presencia de gas butano (ambos sensores son sensibles a este gas) y se analizó la señal con y sin la presencia del gas como se muestra en las figuras 4.19 y 4.20:



*Figura 4.19. Señal del sensor sin perturbación.*

Una vez que ajustamos los parámetros de osciloscopio, colocamos el sensor en presencia del gas:



*Figura 4.20. Señal del sensor con una perturbación.*

Del anterior experimento, logramos determinar lo siguiente acerca de la señal de nuestros sensores:

- La señal presenta una variación de 0.1 hasta 1.4 volts; Al no estar en el rango de los milivolts, no requiere una amplificación.
- La señal no presenta ruido; No es necesario aplicar un filtrado.
- La salida de señal es lineal; Por lo cual, nuestro sensor no requiere una linealización.
- El voltaje de salida está dentro del rango que acepta nuestro controlador; No se necesita hacer una atenuación de la señal.
- La señal es analógica; Por lo cual, habrá que digitalizarla.



## 4.4. Conversión analógico - digital

Un ADC (Conversor analógico a digital) es un sistema que transforma señales analógicas en señales digitales.

El objetivo de estos sistemas es traducir los valores de una señal analógica en una señal binaria para poder ser interpretada en un sistema digital.

El ADC realiza una muestra de la tensión analógica en un instante, y luego determina cuál sería el valor en binario en el lado de salida del ADC. La cantidad de muestras que el dispositivo toma cada segundo será mencionada en su documentación. La frecuencia de muestreo debe ser por lo menos dos veces mayor que la frecuencia más alta que se debe transmitir. Esto se conoce como la tasa de Nyquist.

### *Teorema de Nyquist*

Nyquist, en 1924, demostró que no es necesario enviar todo un ciclo de una señal (una sucesión de infinitos puntos que caracterizan a toda señal analógica) para que del lado del receptor pueda ser interpretada, sino que basta con solo dos muestras por ciclo para que aún se pueda recuperar la señal original.

En forma más precisa: *Toda señal limitada en banda (debido a que se encuentra acotada por el AB del canal) se puede reconstruir completamente a partir de las muestras tomadas de la misma, siempre que la velocidad del muestreo se realice como mínimo al doble de la máxima frecuencia de la señal. A esta velocidad de muestreo se la denomina "Frecuencia de Nyquist" [6].*

**Ancho de Banda:** es la diferencia, medida en Hertz, entre la máxima y la mínima frecuencia que puede pasar por un canal de comunicaciones.

$$AB = F_{max} - F_{min}$$

donde  $F_{max}$  nos indica el valor de la máxima frecuencia que puede pasar por el canal mientras que  $F_{min}$  nos dará el límite mínimo de la misma (véase Figura 4.21).

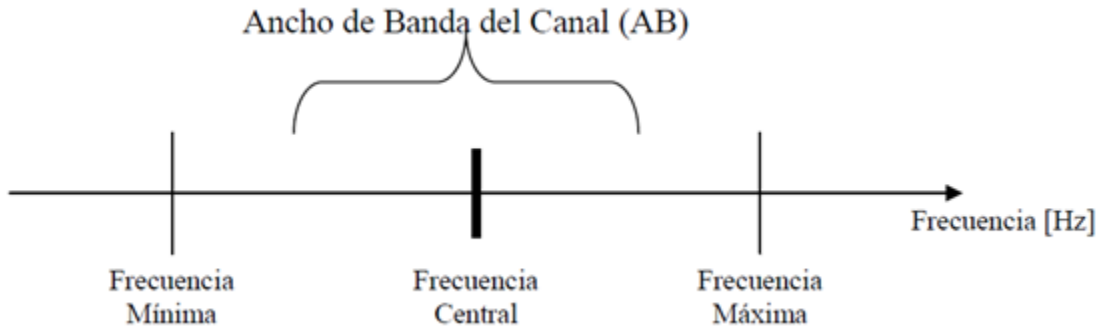
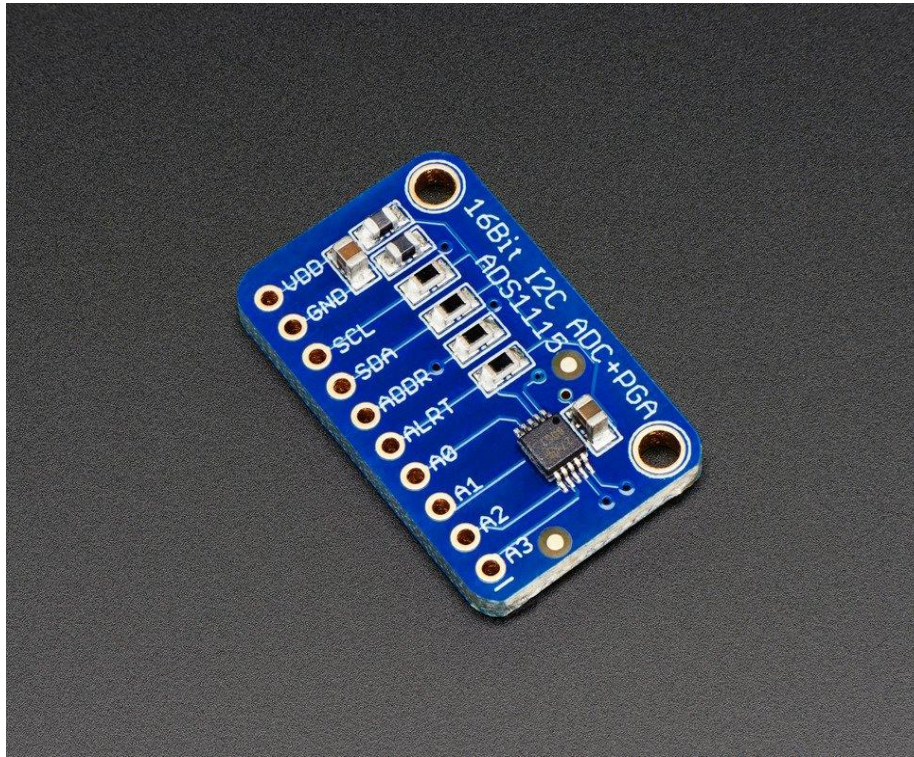


Figura 4.21. Ancho de banda.

Tomando en cuenta el teorema de Nyquist, se seleccionó un ADC que cumple con esta condición para que la reconstrucción de la señal sea confiable.

#### 4.4.1 ADS 1115

Para este sistema se utilizó el conversor analógico - digital ADS1115 (Figura 4.22). Este conversor nos proporciona una precisión de 16 bits a 860 muestras / segundo y se puede alimentar con un voltaje de 2 a 5V. La comunicación serial se realiza mediante el protocolo de comunicación I2C, el cual es compatible con Raspberry Pi.



*Figura 4.22. ADS1115.*

El ADS1115 se comunica con nuestro controlador por medio de sus pines SCL y SDA. Ambos van a los pines SCL y SDA de la Raspberry Pi respectivamente.

Para la alimentación del ADS se utilizó la salida de voltaje de 5V de la Raspberry (pin 2) y la conexión a tierra con el mismo controlador (pin 39).

Ambos sensores se alimentarán a 5V (pin 2) y la conexión a tierra será común (pin 39). La conexión con nuestros sensores se realizó a través de las entradas analógicas A0 (MQ-2) y A1 (MQ-9) como se muestra en la figura 4.23:

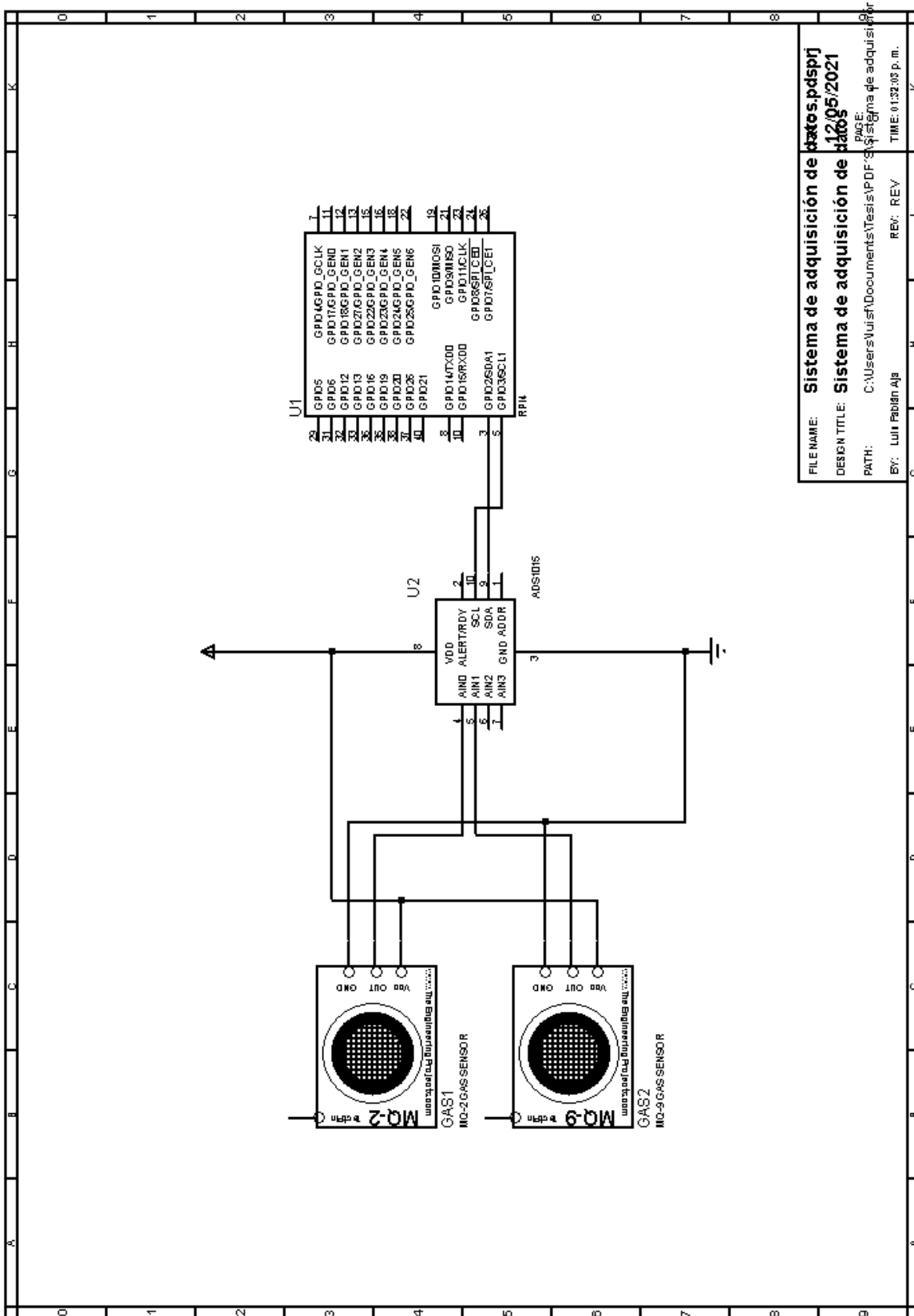


Figura 4.23. Diagrama de conexión del sistema de adquisición de datos.

## 4.5. Programación

Dadas las necesidades del proyecto, la programación de nuestro proyecto se realizó en tres distintos programas:

- Programa para capturar fotografías
- Programa para capturar videos
- Programa para monitorear sensores

Estos se pensaron a partir de las necesidades que se plantearon al inicio del proyecto.

Todos los programas se realizaron en un IDE de Python que viene incluido en el software que instalamos en nuestra Raspberry llamado Thonny Python (Figura 4.24). Esto con la finalidad de que el entorno de desarrollo sea más sencillo de utilizar para el usuario.

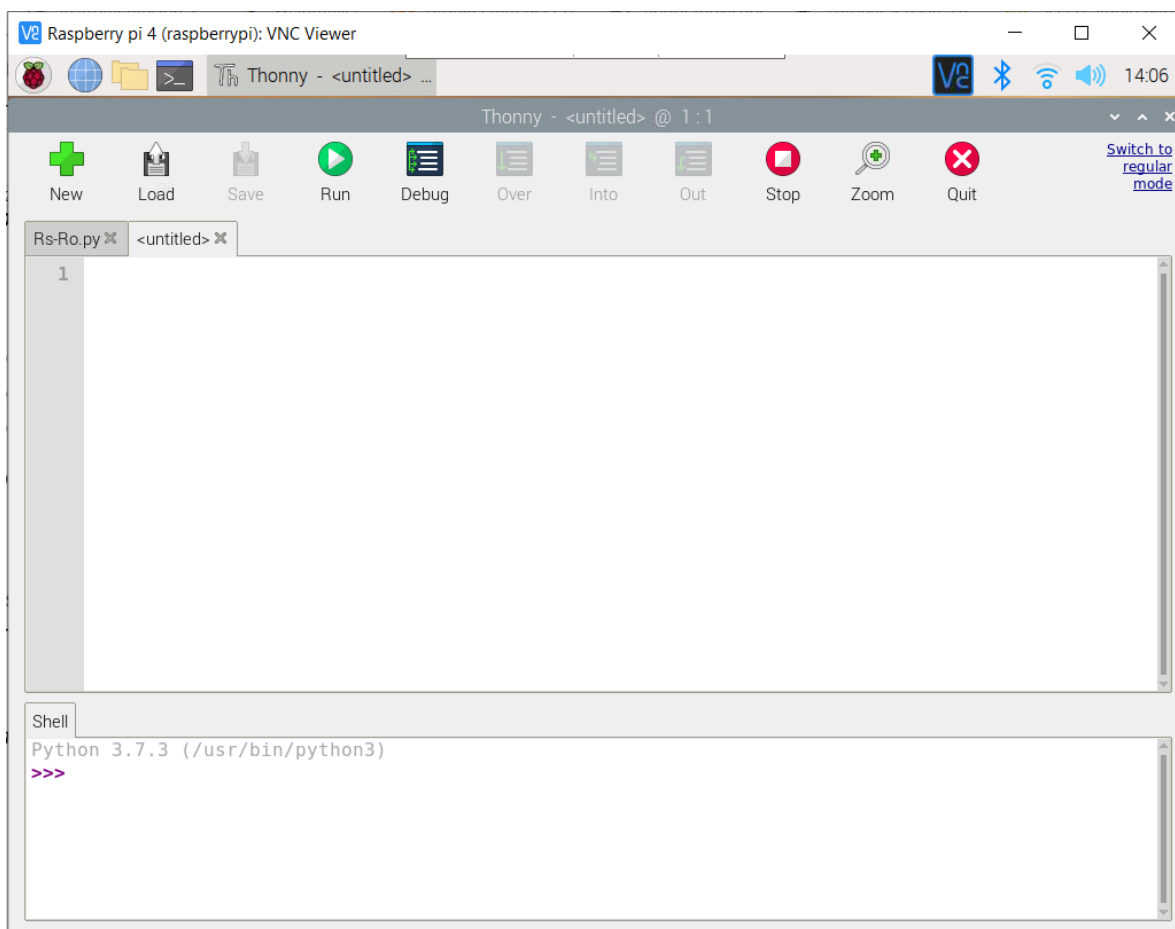


Figura 4.24. Thonny Python IDE en Raspberry Pi.

Una de las necesidades del usuario es conocer el entorno que rodea al robot. Para esto se pensó en dos programas: Uno para tomar “X” cantidad de fotografías (las determinará el usuario) y video con una duración de “X” segundos (los determinará el usuario).

#### **4.5.1. Programa de captura de fotografías**

Para este programa se tomó en cuenta la necesidad de realizar una captura y que esta sea de fácil acceso para el usuario. Por lo tanto, nuestro algoritmo quedó de la siguiente manera:

- *Algoritmo:*

El programa llamará a las librerías OS, Datetime, PiCamera y time para utilizar la cámara. Crea un objeto llamado “destino” que será la ruta para guardar las capturas y “cámara” que lo asocia a la PiCamera. Se crea una variable “i” y una variable “n” para generar un ciclo “for” en el que se ejecutarán las capturas de imagen “n” veces con un retraso de 1 segundo y guardar la captura con la fecha exacta en que se tomó en el destino previo.

- *Diagrama de flujo:*

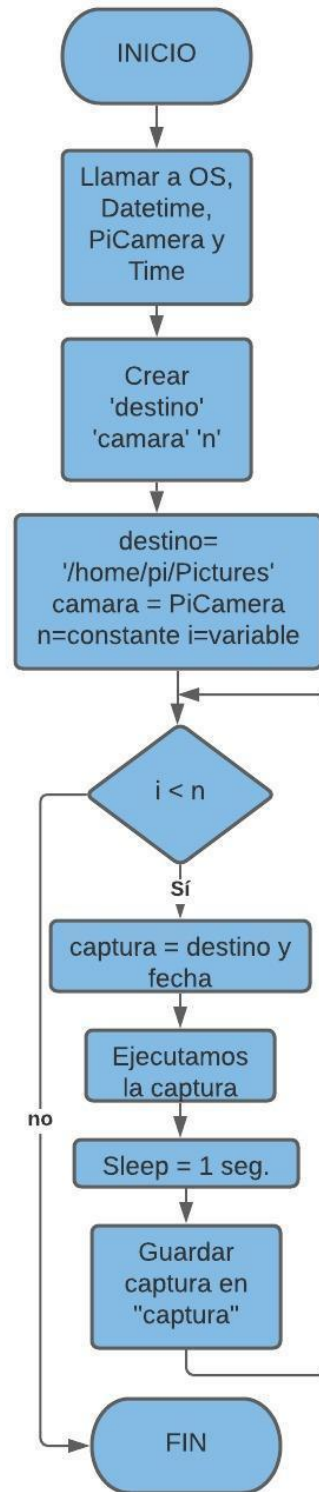
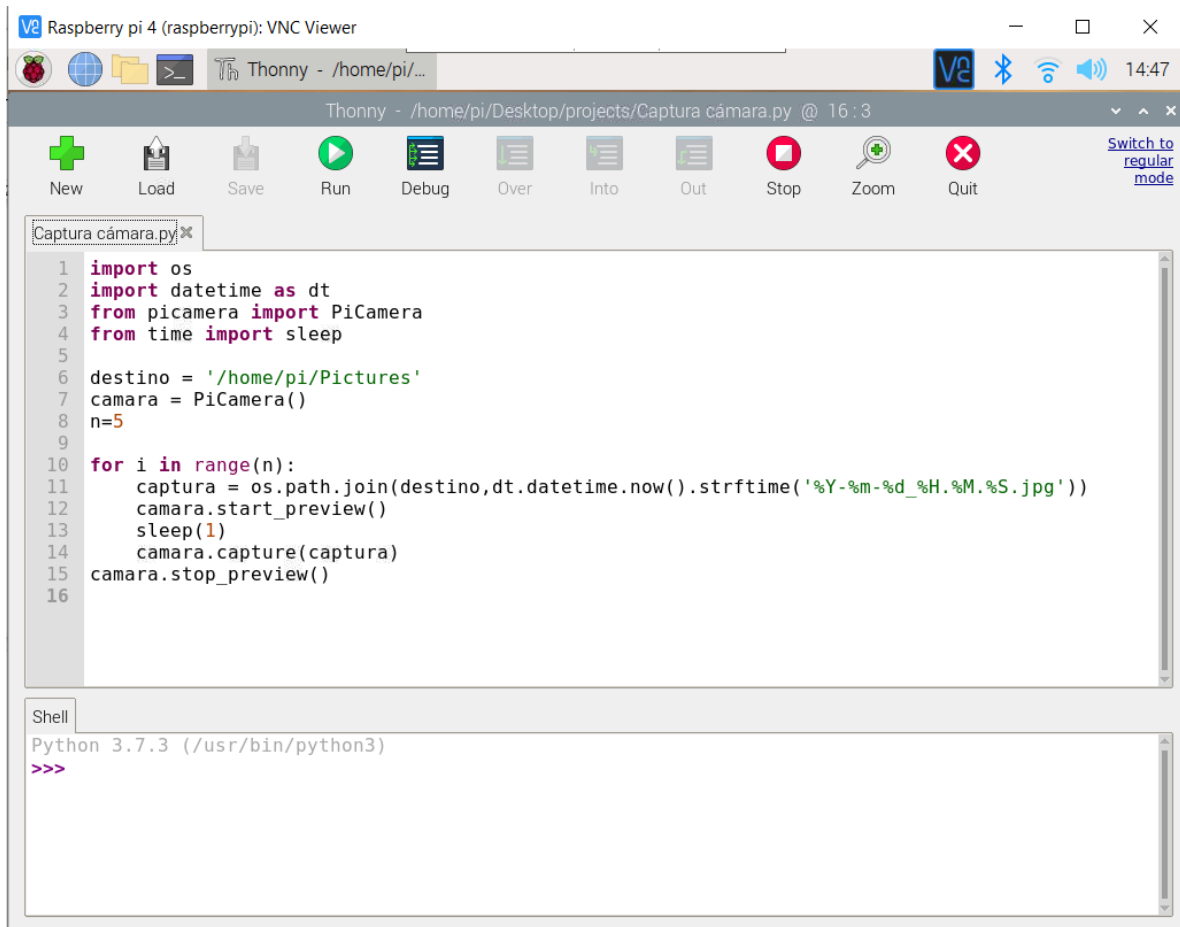


Figura 4.25. Diagrama de flujo de captura de imagen.

- **Código:**



```
1 import os
2 import datetime as dt
3 from picamera import PiCamera
4 from time import sleep
5
6 destino = '/home/pi/Pictures'
7 camara = PiCamera()
8 n=5
9
10 for i in range(n):
11     captura = os.path.join(destino,dt.datetime.now().strftime('%Y-%m-%d_%H.%M.%S.jpg'))
12     camara.start_preview()
13     sleep(1)
14     camara.capture(captura)
15     camara.stop_preview()
16
```

Shell

```
Python 3.7.3 (/usr/bin/python3)
>>>
```

Figura 4.26. Código del programa de captura de imagen.



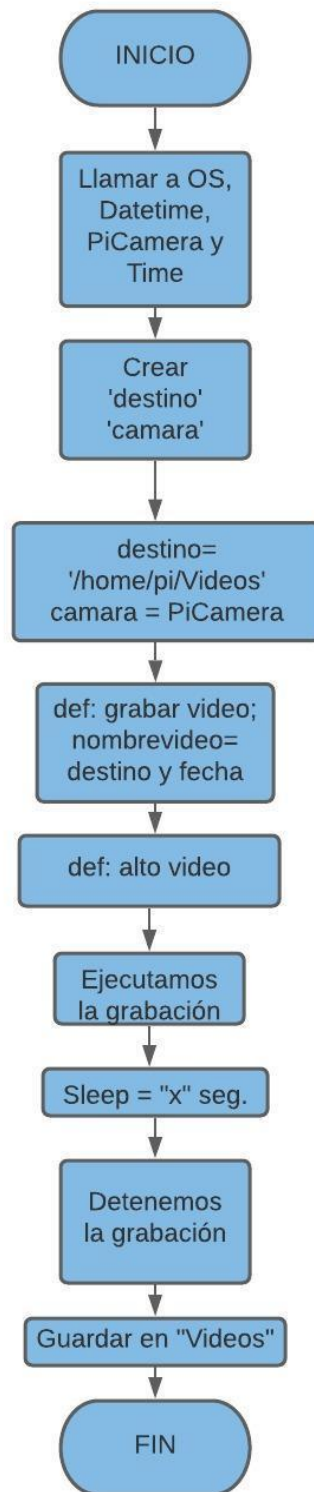
## 4.5.2. Programa de captura de video

Para el caso en el que el robot esté en movimiento, se realizó un programa de captura de video en el que se configuró el tiempo de sleep del programa para que equivalga en segundos, la duración del video se puede modificar de acuerdo con la necesidad del usuario.

- *Algoritmo:*

El programa llamará a las librerías OS, Datetime, PiCamera y time para utilizar la cámara. Crea un objeto llamado “destino” que será la ruta para guardar el video y “cámara” que lo asocia a la PiCamera. Se define una función para iniciar la grabación del video en el que se guardará el video en la ruta definida y con la fecha exacta en que se tomó. Se define una nueva función para detener la grabación del video para posteriormente ejecutar la primera función, dar un sleep deseado (que será la duración del video) y ejecutar la segunda función para detener la grabación.

- *Diagrama de flujo:*



*Figura 4.27. Diagrama de flujo de captura de video.*

- **Código:**

```
1 import os
2 import datetime as dt
3 from picamera import PiCamera
4 from time import sleep
5 from signal import pause
6
7 destino = '/home/pi/Videos'
8 camara = PiCamera()
9
10 def grabar_video():
11     nombrevideo = os.path.join(destino,dt.datetime.now().strftime('%Y-%m-%d_%H.%M.%S.h264'))
12     camara.start_preview()
13     camara.start_recording(nombrevideo)
14
15 def alto_video():
16     camara.stop_recording()
17     camara.stop_preview()
18
19 grabar_video()
20 sleep(10)
21 alto_video()
```

Shell  
Python 3.7.3 (/usr/bin/python3)  
>>>

Figura 4.28. Código del programa de captura de video.

### 4.5.3. Programa de lectura de sensores

El robot puede ingresar en ambientes contaminados con gases nocivos para la salud, por lo tanto, es necesario monitorear de manera constante la lectura que estén arrojando.

- *Algoritmo:*

El programa llamará a las librerías `time`, `board`, `busio`, `adafruit_ads1x15.ads1115` para utilizar el ADS 1115. Crea un bus de comunicación con el protocolo I2C para después crear el objeto del ADS y lo asocia al ADS mediante el bus I2C. Creamos dos objetos para cada canal a utilizar del ADS y los asociamos al puerto correspondiente a usar en el ADS. Se utiliza el voltaje obtenido de cada canal y se opera para obtener el valor  $R_s$  de cada sensor para después operar la relación  $R_s/R_o$  y mostrarla en el monitor serial mediante un ciclo `while true` con un `sleep` de 2 segundos.

- *Diagrama de flujo:*

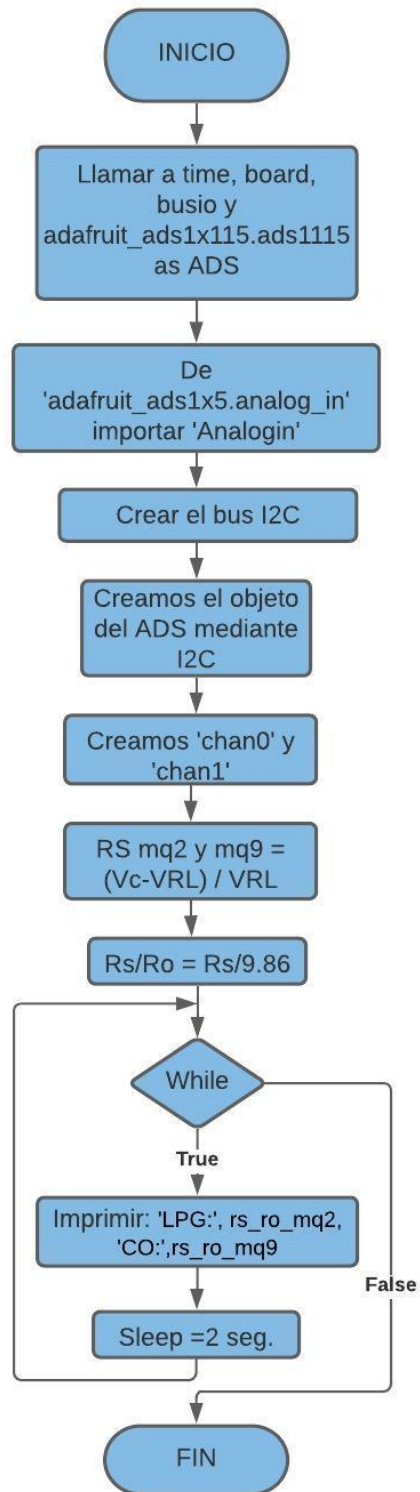


Figura 4.29. Diagrama de flujo de lectura de sensores.

- **Código:**

```
4 import adafruit_ads1x15.ads1115 as ADS
5 from adafruit_ads1x15.analog_in import AnalogIn
6
7 # Creamos el bus I2C
8 i2c = busio.I2C(board.SCL, board.SDA)
9
10 # Creamos el objeto del ADS mediante el bus i2c
11 ads = ADS.ADS1115(i2c)
12
13 # Creamos objetos para cada canal de comunicación
14 chan = AnalogIn(ads, ADS.P0)
15 chan2 = AnalogIn(ads, ADS.P1)
16
17 #Operamos el voltaje para obtener la característica de sensibilidad
18 rs_mq2 = ((5.15-chan2.voltage)/chan2.voltage)
19 rs_mq9 = ((5.15-chan.voltage)/chan.voltage)
20
21 #Obtenemos Rs/Ro
22 rs_ro_mq2 = (rs_mq2/9.86)
23 rs_ro_mq9 = (rs_mq9/9.86)
24
25 while True:|
26     print('LPG:', rs_ro_mq2, 'CO:',rs_ro_mq9)
27     time.sleep(2)
```

Shell  
Python 3.7.3 (/usr/bin/python3)  
>>>

Figura 4.30. Código del programa de lectura de sensores.

# CAPÍTULO 5

## Pruebas y resultados

Se realizaron pruebas para poder verificar el correcto funcionamiento del sistema de adquisición de datos una vez que se integraron todas las etapas del desarrollo de este. Se probaron los tres programas para adquirir las variables físicas deseadas a través de la conexión remota con nuestro controlador.

### 5.1. Prueba de captura de imagen

Para realizar las capturas, se determinó probar la cámara en ambientes con y sin iluminación.

Ambas pruebas se realizaron con la cámara colocada en la misma posición para evitar variaciones en la profundidad del campo y poder verificar la capacidad de la cámara en ambos ambientes de manera correcta (Figura 5.2 y Figura 5.5).

Para la prueba con iluminación, el programa se configuró para tomar cinco capturas (véase Figura 5.1) y que fueran guardadas en la carpeta de imágenes con la hora y fecha exactas en que fueron tomadas (Figura 5.3).

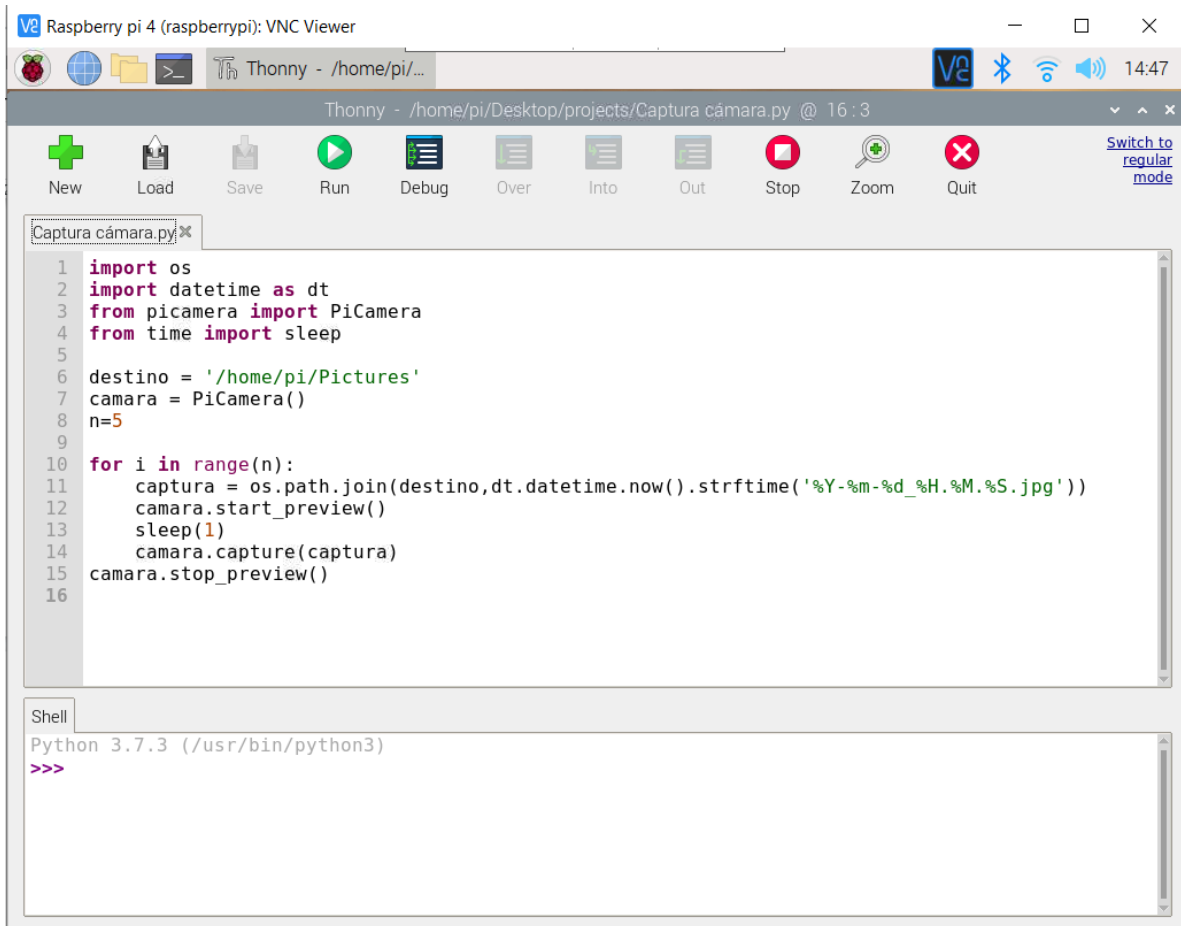
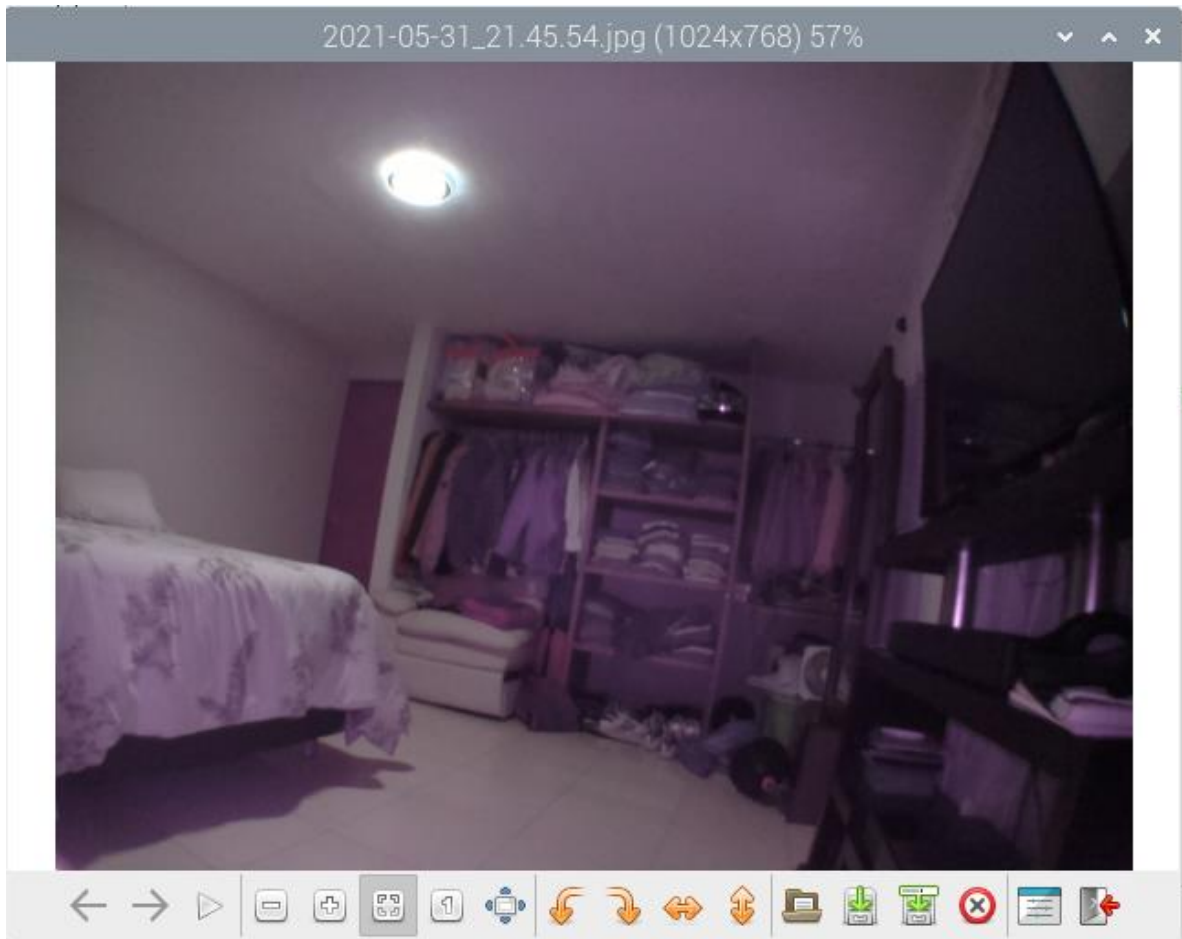


Figura 5.1. Configuración del programa.





*Figura 5.2. Captura con iluminación.*

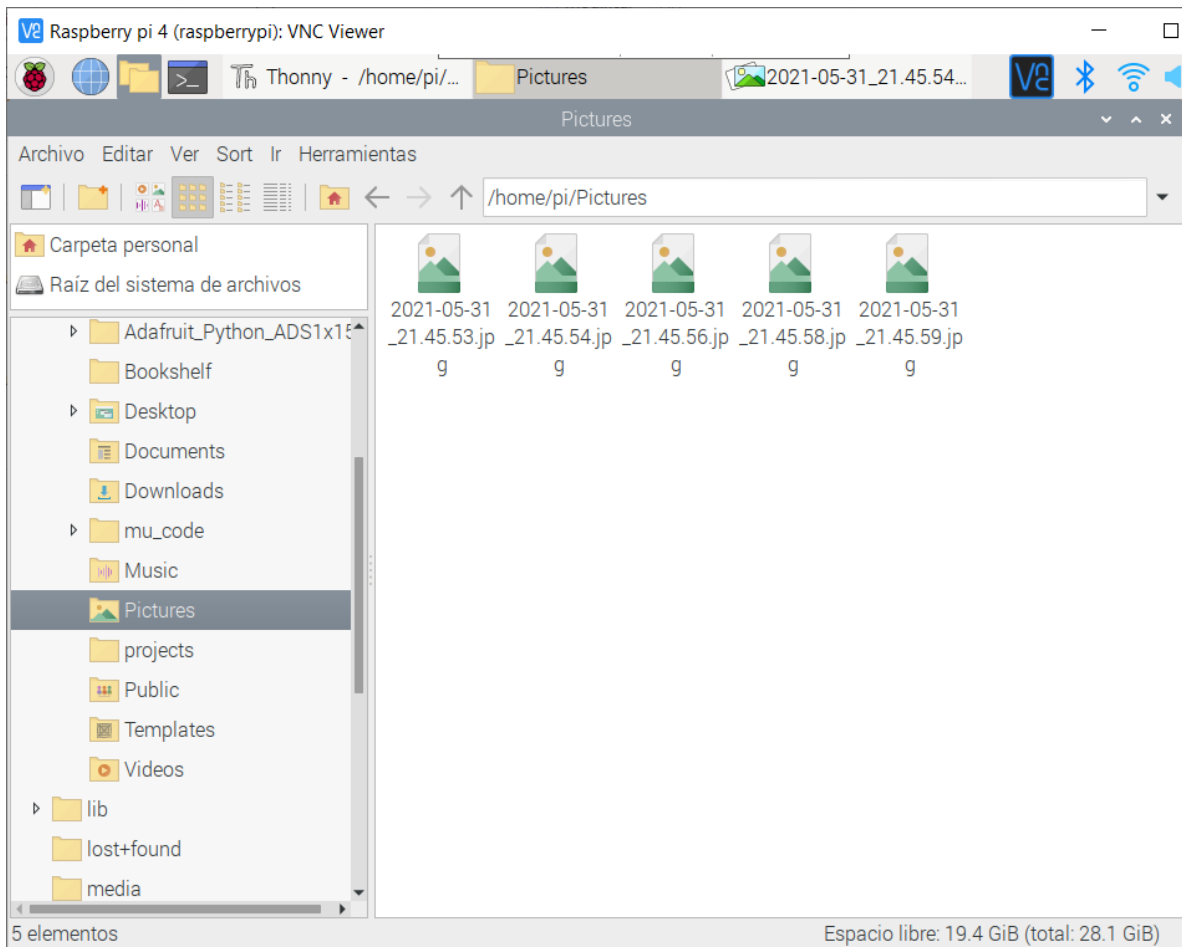


Figura 5.3. Archivo de capturas.

Para la prueba sin iluminación se configuró el programa a una captura como se muestra en la figura 5.4:

```
1 import os
2 import datetime as dt
3 from picamera import PiCamera
4 from time import sleep
5
6 destino = '/home/pi/Pictures'
7 camara = PiCamera()
8 n=1
9
10 for i in range(n):
11     captura = os.path.join(destino,dt.datetime.now().strftime('%Y-%m-%d_%H.%M.%S.jpg'))
12     camara.start_preview()
13     sleep(1)
14     camara.capture(captura)
15     camara.stop_preview()
16
```

Figura 5.4. Configuración del programa.

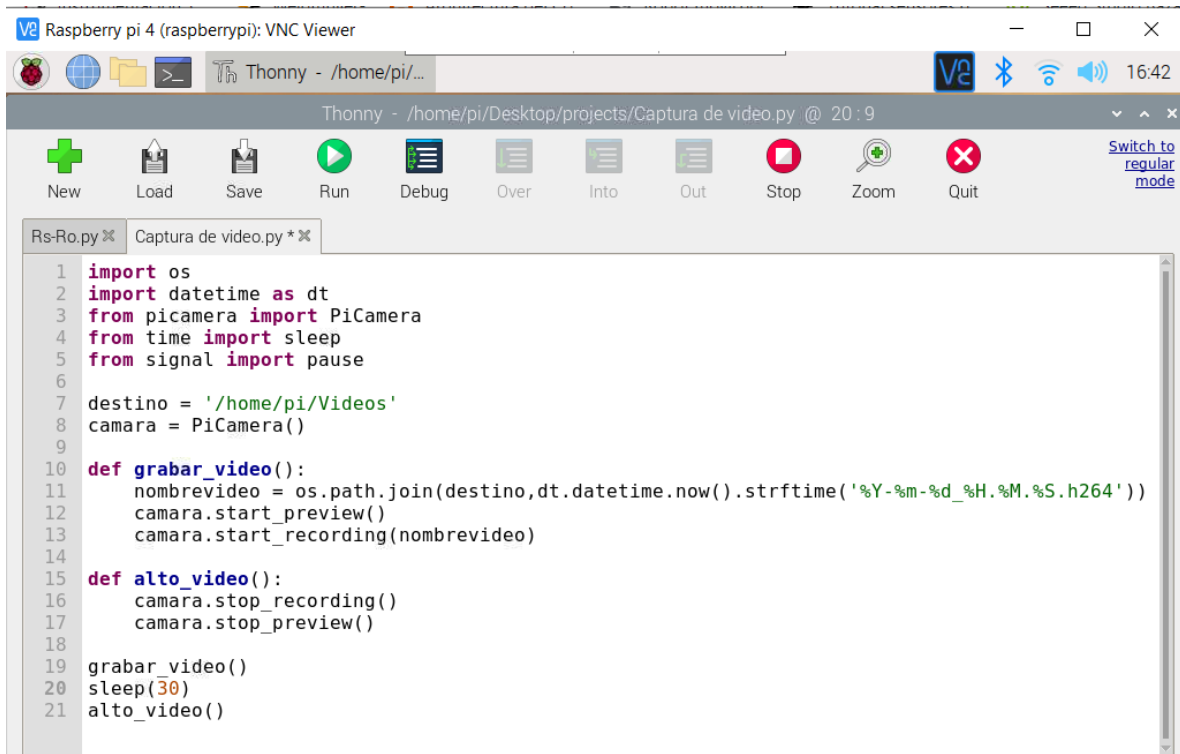


*Figura 5.5. Captura sin iluminación.*

Podemos observar que la nitidez de la imagen es buena con una profundidad de campo de 5 metros en un ambiente sin iluminación.

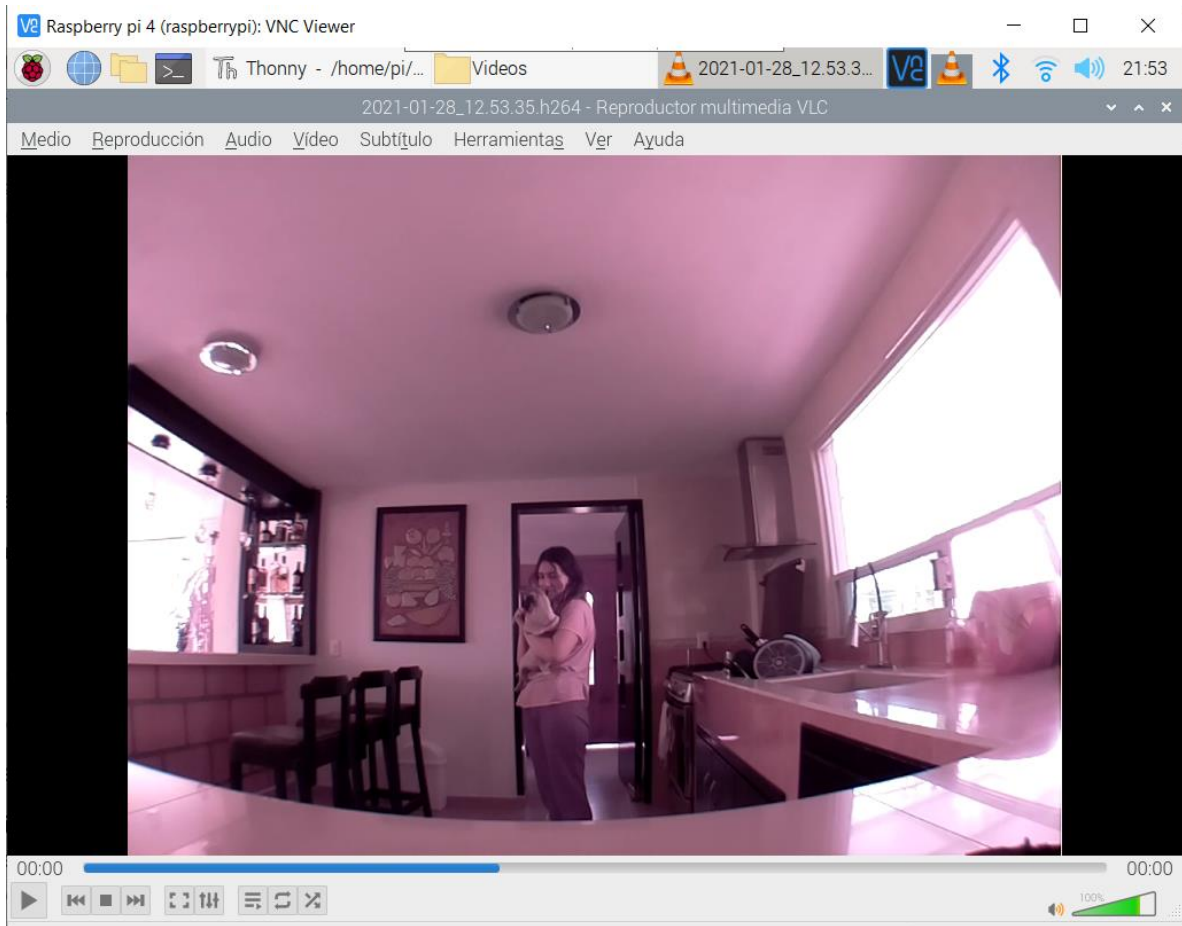
## 5.2. Prueba de captura de videos

Para la prueba de captura de video (Figura 5.7), se configuró el programa para que se capturara un video con duración de 30 seg. (véase Figura 5.6). Este se guardó en el archivo de video con la fecha y hora exacta en que se capturó (Figura 5.8).



```
1 import os
2 import datetime as dt
3 from picamera import PiCamera
4 from time import sleep
5 from signal import pause
6
7 destino = '/home/pi/Videos'
8 camara = PiCamera()
9
10 def grabar_video():
11     nombrevideo = os.path.join(destino, dt.datetime.now().strftime('%Y-%m-%d_%H.%M.%S.h264'))
12     camara.start_preview()
13     camara.start_recording(nombrevideo)
14
15 def alto_video():
16     camara.stop_recording()
17     camara.stop_preview()
18
19 grabar_video()
20 sleep(30)
21 alto_video()
```

Figura 5.6. Configuración del programa de captura de video.



*Figura 5.7. Reproductor de video.*

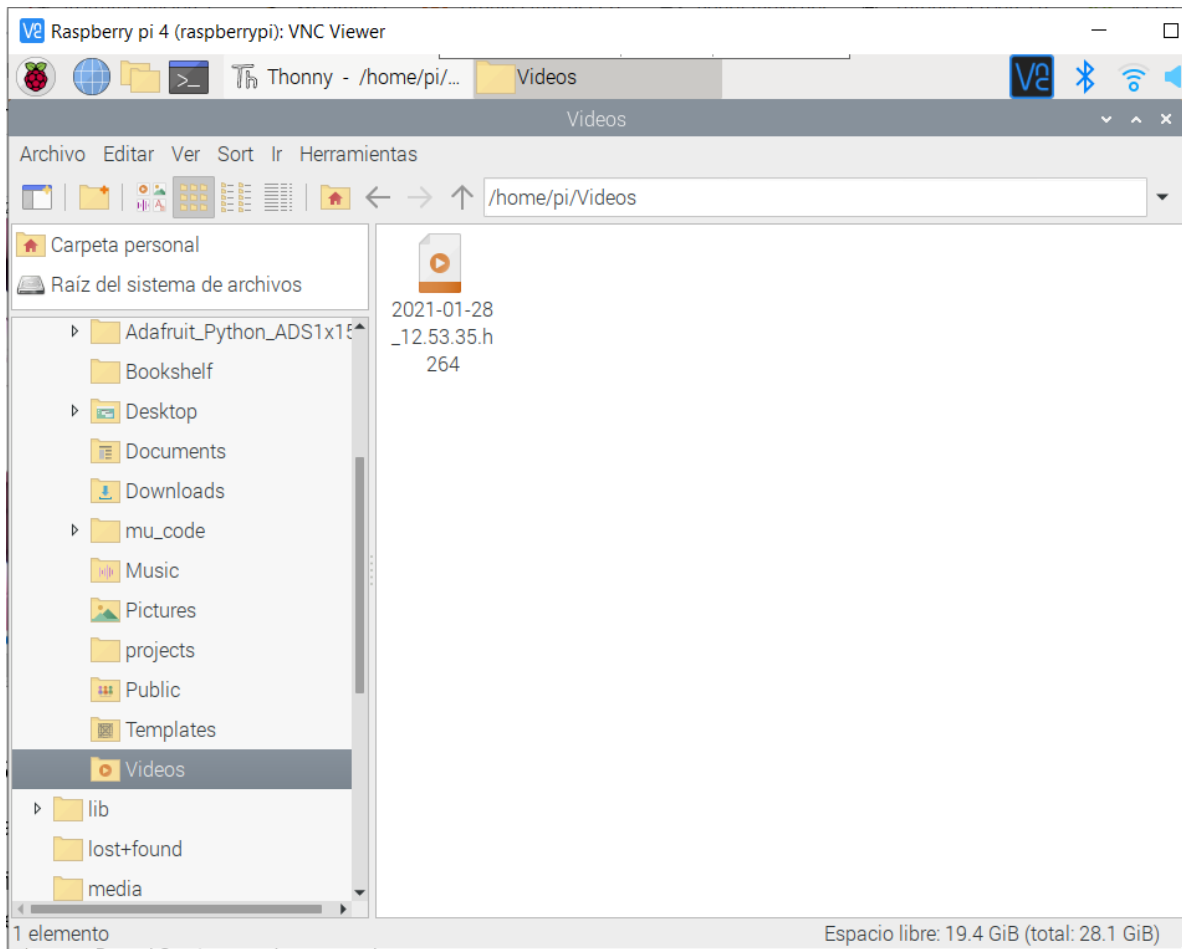


Figura 5.8. Archivo de video.

### 5.3. Prueba de lectura de sensores

Para verificar el correcto funcionamiento de los sensores y del ADS, primero se comparó el voltaje de salida de los sensores realizando una lectura con un multímetro y realizando otra lectura mediante el monitor serial de nuestro IDE como se muestra en la figura 5.9:

```
Raspberry pi 4 (raspberrypi): VNC Viewer
Thonny - /home/pi/...
Thonny - /home/pi/Desktop/projects/Ppm.py @ 18:37
New Load Save Run Debug Over Into Out Stop Zoom Quit
Rs-Ro.py x Lectura de gas.py x Ppm.py x
1 import time
2 import board
3 import busio
4 import adafruit_ads1x15.ads1115 as ADS
5 from adafruit_ads1x15.analog_in import AnalogIn
6
7 # Creamos el bus I2C
8 i2c = busio.I2C(board.SCL, board.SDA)
9
10 # Creamos el onjetos del ADS mediante el bus i2c
11 ads = ADS.ADS1115(i2c)
12
13 # Creamos variables para cada canal de comunicación
14 chan = AnalogIn(ads, ADS.P0)
15 chan2 = AnalogIn(ads, ADS.P1)
16
17 while True:
18     print('MQ2:', chan2.voltage, 'V', 'MQ9:', chan.voltage, 'V')
Shell
Python 3.7.3 (/usr/bin/python3)
>>> %Run Ppm.py
MQ2: 1.6620507217627492 V MQ9: 1.1120339365825374 V
MQ2: 1.6620507217627492 V MQ9: 1.1040336924344616 V
MQ2: 1.6500503555406356 V MQ9: 1.1120339365825374 V
MQ2: 1.6420501113925596 V MQ9: 1.1020336313974426 V
MQ2: 1.6400500503555409 V MQ9: 1.086033143101291 V
MQ2: 1.6320498062074649 V MQ9: 1.0900332651753288 V
```

Figura 5.9. Voltaje de salida en los sensores.

Verificando el valor del voltaje de salida de ambos sensores con el multímetro, comprobamos que el ADS está reconstruyendo la señal analógica en digital de manera correcta.

Dado que la calibración de los sensores se tiene que realizar con base en las condiciones atmosféricas y que el uso del sistema de adquisición de datos se puede realizar en condiciones diversas, el programa se configuró para que una vez que inicie, el valor que se muestre en el monitor serial sea constante (véase Figura 5.10), y que, cuando exista presencia de gas, el valor comience a variar.



```
10 # Creamos el objeto del ADS mediante el bus i2c
11 ads = ADS.ADS1115(i2c)
12
13 # Creamos objetos para cada canal de comunicación
14 chan = AnalogIn(ads, ADS.P0)
15 chan2 = AnalogIn(ads, ADS.P1)
16
17 #Operamos el voltaje para obtener la caracteristica de sensibilidad
18 rs_mq2 = ((5.15-chan2.voltage)/chan2.voltage)
19 rs_mq9 = ((5.15-chan.voltage)/chan.voltage)
20
21 #Obtenemos Rs/Ro
22 rs_ro_mq2 = (rs_mq2/9.86)
23 rs_ro_mq9 = (rs_mq9/9.86)
24
25 while True:
26     print('LPG:', rs_ro_mq2, 'CO:',rs_ro_mq9)
27     time.sleep(2)
```

```
Python 3.7.3 (/usr/bin/python3)
>>> %Run Rs-Ro.py

LPG: 0.2667172931258421 CO: 0.4398895966378845
LPG: 0.2667172931258421 CO: 0.4398895966378845
LPG: 0.2667172931258421 CO: 0.4398895966378845
LPG: 0.2667172931258421 CO: 0.4398895966378845
LPG: 0.2667172931258421 CO: 0.4398895966378845
```

Figura 5.10. Lectura de sensores.

# CAPÍTULO 6

## **Discusión**

Partiendo del antecedente histórico de la robótica móvil, los sistemas de adquisición de datos permiten aportar capacidades que los robots pueden llegar a necesitar.

El sistema desarrollado en el presente proyecto es una buena opción por implementar dentro de la robótica móvil ya que se logró desarrollar en una plataforma de bajo costo y es capaz de adquirir, procesar y mostrar información del entorno en el que se encuentra el robot a el usuario de manera remota.

La interfaz del sistema nos brinda la capacidad de operar el sistema de manera sencilla, sin embargo, le da la libertad al usuario de modificar las configuraciones necesarias para que el sistema opere de manera correcta.

Respecto a la adquisición de imagen, no se logró integrar el uso de periféricos de mayor capacidad que se plantearon al inicio del proyecto (Kinect 2.0) esto debido a la incompatibilidad de software que el fabricante restringe para su uso particular.

El uso de sensores de la serie MQ, si bien permiten conocer si existe la presencia de gases nocivos para la salud, no permiten saber con exactitud la cantidad que pueda estar concentrada en el entorno del robot, esto deja abierta la opción de poder integrar futuros sensores optimizados al sistema de adquisición de datos.

## **Conclusiones**

La instrumentación electrónica comprende la medición de cualquier tipo de magnitud física, la conversión de la misma a magnitudes eléctricas y su tratamiento para proporcionar la información adecuada a un sistema de control, a un operador humano o a ambos.

El sistema de adquisición de datos desarrollado en este proyecto es un ejemplo de instrumentación electrónica, ya que es capaz de adquirir las variables físicas necesarias para que el usuario pueda determinar si se encuentra en un entorno seguro o nocivo para la salud.

Se instaló una interfaz visual ergonómica en el microcontrolador que permite al usuario interactuar de manera sencilla con el sistema y poder conocer cuál es el entorno en el que se encuentra.

Además, el acondicionamiento realizado a las señales provenientes de los sensores entrega la información adecuada a nuestro sistema de control. Lo que se verificó durante las pruebas realizadas en el desarrollo del proyecto.

Dadas las condiciones sanitarias durante el desarrollo de este proyecto (2020 - 2021) no fue posible integrar el sistema de adquisición de datos a un robot móvil, por lo que será necesario el diseño de una base o mecanismo para su integración.

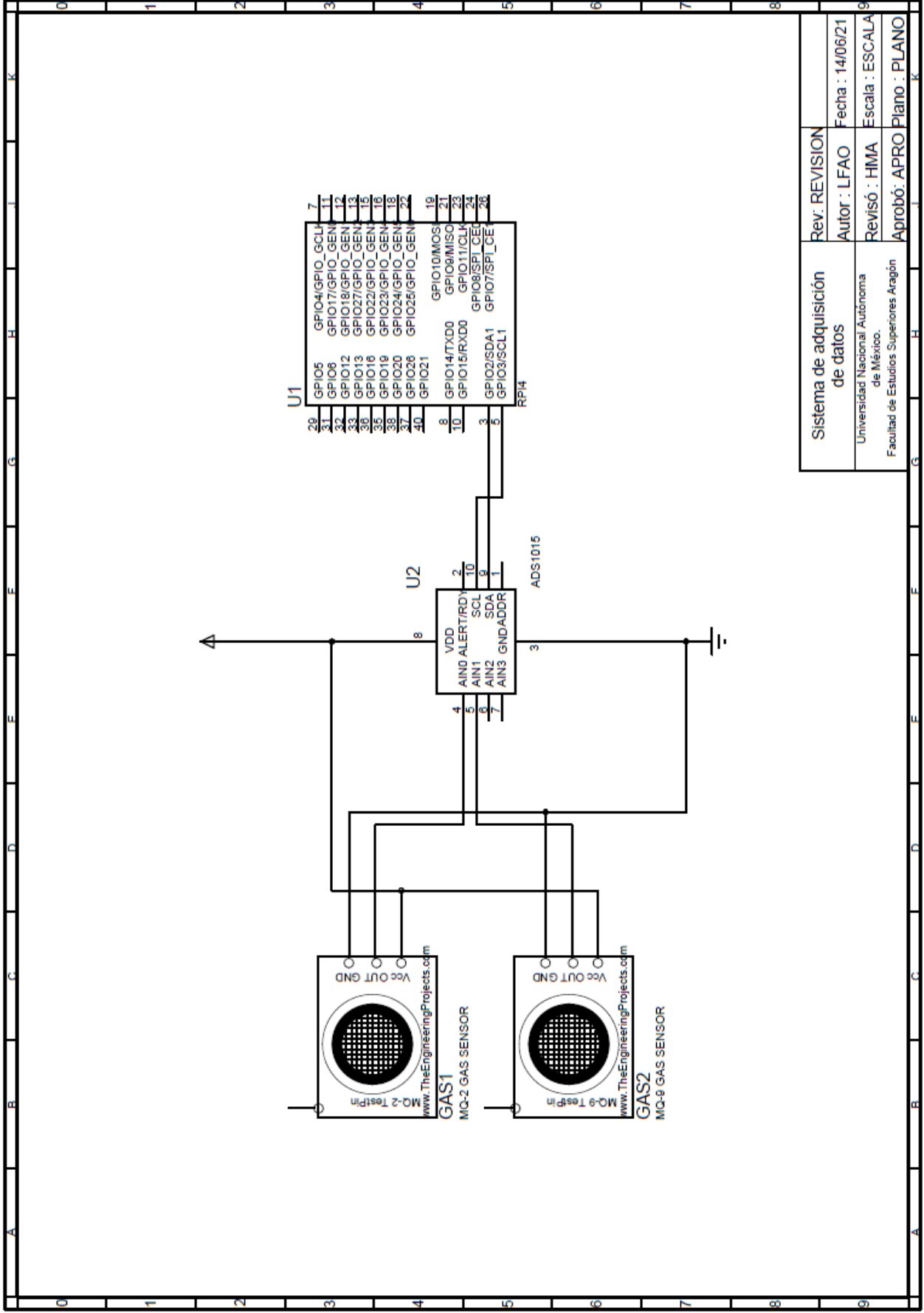
Cabe señalar que el sistema integra todas las etapas de instrumentación electrónica sobre una plataforma de bajo costo (comparado con tarjetas de adquisición de datos de línea industrial).

## Referencias

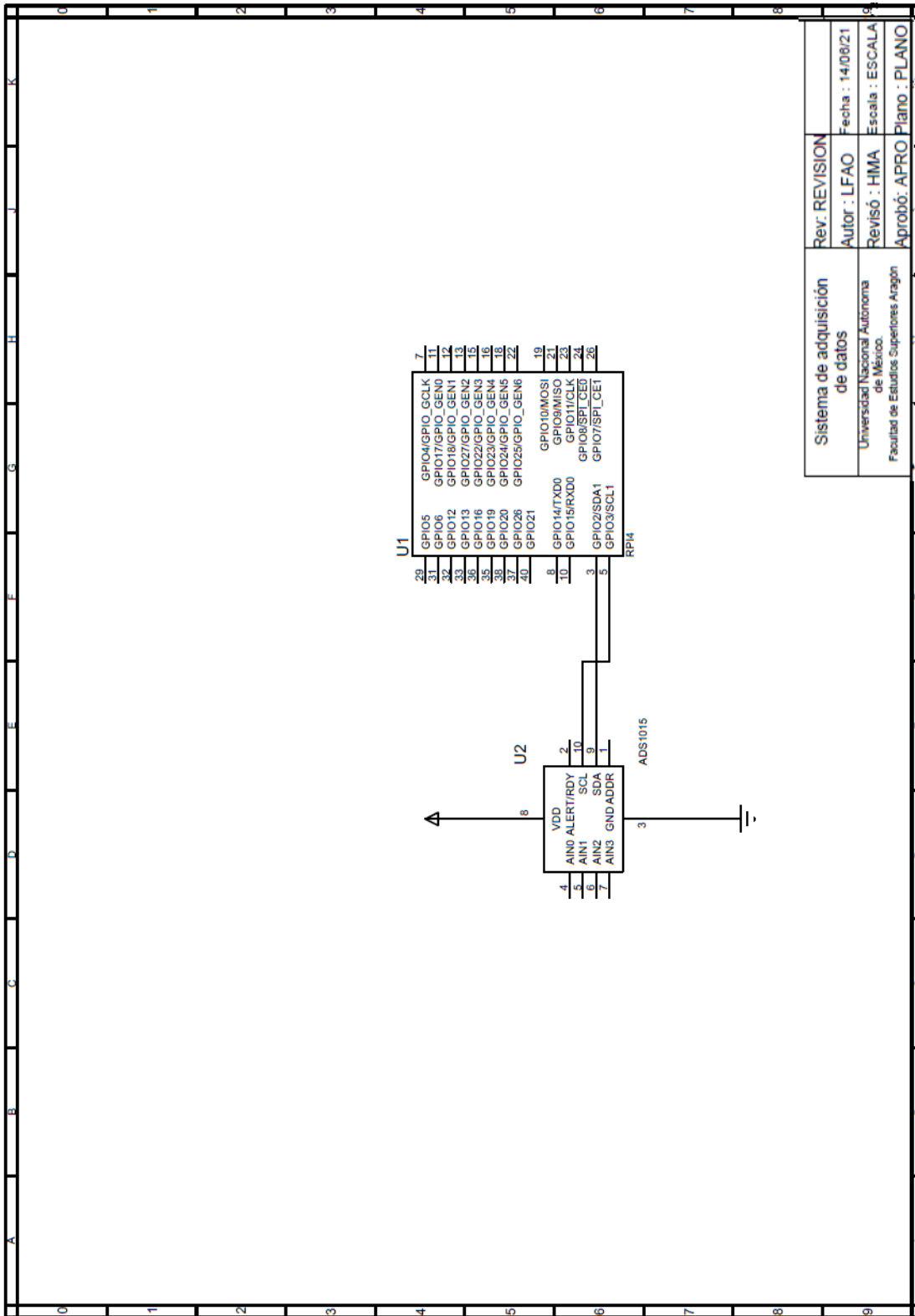
- [1] Il, B. (2008). Una introducción a los robots móviles. Recuperado el 19 de junio de 2020 de [https://www.aadeca.org/pdf/CP\\_monografias/monografia\\_robot\\_movil.pdf](https://www.aadeca.org/pdf/CP_monografias/monografia_robot_movil.pdf)
- [2] Ollero, A. (2001). *Robótica. Manipuladores y robots móviles*. Barcelona, España: Marcombo.
- [3] García, L. *Modelado Cinemático y Control de Robots Móviles con Ruedas*. Tesis doctoral. Universidad Politécnica de Valencia.
- [4] Raya, V. (2014). *Diseño de un robot terrestre omnimóvil*. Tesis de licenciatura. Universidad Nacional Autónoma de México.
- [5] Pallás, R. (2003). *Sensores y acondicionadores de señal*. (4° ed.). Barcelona, España: Marcombo.
- [6] Mercedes, M. & E. (2015). *Instrumentación electrónica: Transductores y acondicionadores de señal*. Santander, España: Editorial Universidad Cantabria.
- [7] Semeria, M. (2015). *Los Tres Teoremas. Fourier - Nyquist – Shannon*. Argentina: Universidad del CEMA.
- [8] Castro, A. (2012). *C# para automatización electrónica e industrial*. México: Editor independiente.
- [9] Sanchís, E. (2004). *Fundamentos y electrónica de las comunicaciones*. Valencia, España: Universitat de València.
- [10] Santamaria, E. (1993). *Electrónica digital y Microprocesadores*. Madrid, España: Univ Pontifica Comillas.
- [11] Di Paolo, M. (2013). "Data Acquisition Systems: From Fundamentals to Applied Design". New York: Springer.
- [12] Pérez, M. (2004). "Instrumentación Electrónica". Asturias, España: Thomson.

# APÉNDICES

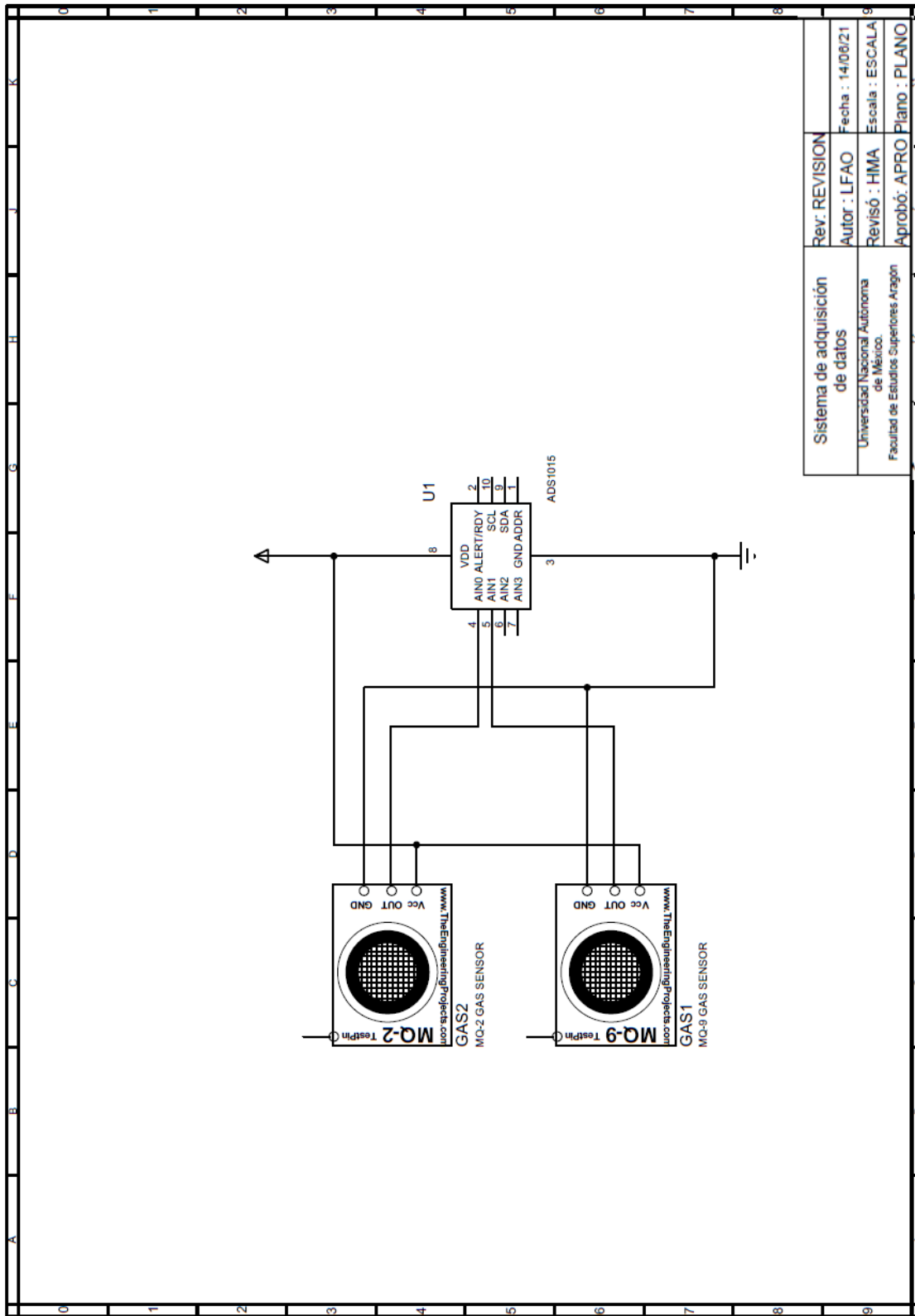
## a) Diagramas de conexión del sistema de adquisición de datos



<b>Sistema de adquisición de datos</b>	<b>Rev: REVISION</b>
Universidad Nacional Autónoma de México.	<b>Autor : LFAO</b> Fecha : 14/06/21
Facultad de Estudios Superiores Aragón	<b>Revisó : HMA</b> Escala : ESCALA
	<b>Aprobó: APRO</b> Plano : PLANO



Sistema de adquisición de datos	Rev: REVISION
Universidad Nacional Autónoma de México	Autor : LFAO
Facultad de Estudios Superiores Aragon	Revisó : HMA
	Aprobó: APRO
	Fecha : 14/06/21
	Escala : ESCALA
	Plano : PLANO



Sistema de adquisición de datos	Rev. REVISION
Universidad Nacional Autónoma de México. Facultad de Estudios Superiores Aragón	Autor : LFAO Fecha : 14/06/21
	Revisó : HMA Escala : ESCALA
	Aprobó: APRO Plano : PLANO



## **b) Sensor MQ-2**

**TECHNICAL DATA****MQ-2 GAS SENSOR****FEATURES**

Wide detecting scope  
Stable and long life

Fast response and High sensitivity  
Simple drive circuit

**APPLICATION**

They are used in gas leakage detecting equipments in family and industry, are suitable for detecting of LPG, i-butane, propane, methane ,alcohol, Hydrogen, smoke.

**SPECIFICATIONS****A. Standard work condition**

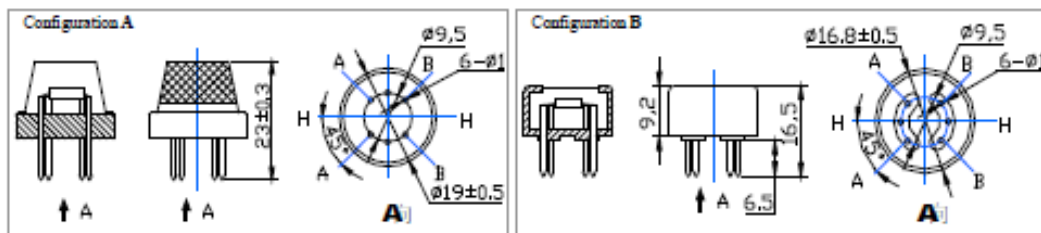
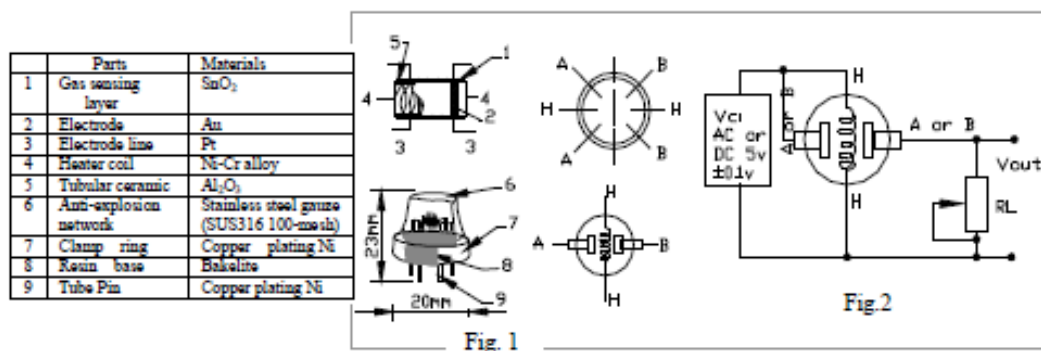
Symbol	Parameter name	Technical condition	Remarks
Vc	Circuit voltage	5V±0.1	AC OR DC
V <sub>H</sub>	Heating voltage	5V±0.1	ACOR DC
R <sub>L</sub>	Load resistance	can adjust	
R <sub>H</sub>	Heater resistance	33 Ω ±5%	Room Tem
P <sub>H</sub>	Heating consumption	less than 800mw	

**B. Environment condition**

Symbol	Parameter name	Technical condition	Remarks
Tao	Using Tem	-20℃-50℃	
Tas	Storage Tem	-20℃-70℃	
R <sub>H</sub>	Related humidity	less than 95%Rh	
O <sub>2</sub>	Oxygen concentration	21%(standard condition)Oxygen concentration can affect sensitivity	minimum value is over 2%

**C. Sensitivity characteristic**

Symbol	Parameter name	Technical parameter	Remarks
R <sub>s</sub>	Sensing Resistance	3KΩ-30KΩ (1000ppm iso-butane)	Detecting concentration scope: 200ppm-5000ppm LPG and propane 300ppm-5000ppm butane
α (3000/1000) isobutane	Concentration Slope rate	≤0.6	5000ppm-20000ppm methane 300ppm-5000ppm H <sub>2</sub> 100ppm-2000ppm Alcohol
Standard Detecting Condition	Temp: 20℃±2℃ Humidity: 65%±5%	Vc:5V±0.1 Vh: 5V±0.1	
Preheat time	Over 24 hour		

**D. Structure and configuration, basic measuring circuit**

Structure and configuration of MQ-2 gas sensor is shown as Fig. 1 (Configuration A or B), sensor composed by micro Al<sub>2</sub>O<sub>3</sub> ceramic tube, Tin Dioxide (SnO<sub>2</sub>) sensitive layer, measuring electrode and heater are fixed into a

crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-2 have 6 pin ,4 of them are used to fetch signals, and other 2 are used for providing heating current.

Electric parameter measurement circuit is shown as Fig.2

E. Sensitivity characteristic curve

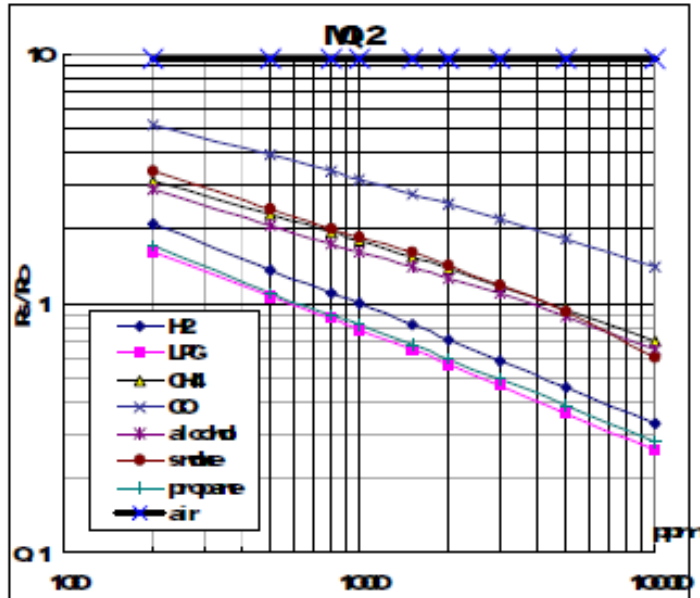


Fig.2 sensitivity characteristics of the MQ-2

Fig.3 is shows the typical sensitivity characteristics of the MQ-2 for several gases.

in their: Temp: 20 °C ,

Humidity: 65% ,

O<sub>2</sub> concentration 21%

RL=5k Ω

R<sub>0</sub>: sensor resistance at 1000ppm of H<sub>2</sub> in the clean air.

R<sub>s</sub>: sensor resistance at various concentrations of gases.

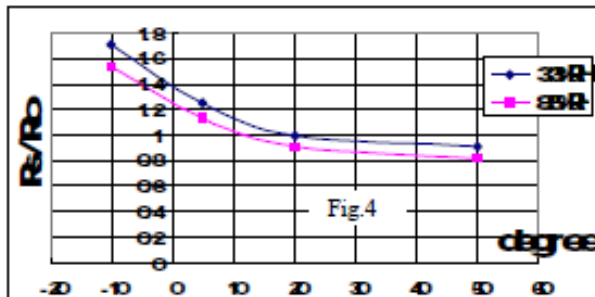


Fig.4 is shows the typical dependence of the MQ-2 on temperature and humidity.

R<sub>0</sub>: sensor resistance at 1000ppm of H<sub>2</sub> in air at 33%RH and 20 degree.

R<sub>s</sub>: sensor resistance at 1000ppm of H<sub>2</sub> at different temperatures and humidities.

### SENSITIVITY ADJUSTMENT

Resistance value of MQ-2 is difference to various kinds and various concentration gases. So, When using this components, sensitivity adjustment is very necessary. we recommend that you calibrate the detector for 1000ppm liquified petroleum gas<LPG>,or 1000ppm iso-butane<i-C<sub>4</sub>H<sub>10</sub>>concentration in air and use value of Load resistance that( R<sub>L</sub>) about 20 K Ω (5K Ω to 47 K Ω).

When accurately measuring, the proper alarm point for the gas detector should be determined after considering the temperature and humidity influence.

## **c) Sensor MQ-9**

## MQ-9 Semiconductor Sensor for CO/Combustible Gas

Sensitive material of MQ-9 gas sensor is SnO<sub>2</sub>, which with lower conductivity in clean air. It make detection by method of cycle high and low temperature, and detect CO when low temperature (heated by 1.5V). The sensor's conductivity is more higher along with the gas concentration rising. When high temperature (heated by 5.0V), it detects Methane, Propane etc combustible gas and cleans the other gases adsorbed under low temperature. Please use simple electrocircuit, Convert change of conductivity to correspond output signal of gas concentration.

MQ-9 gas sensor has high sensitivity to Carbon Monoxide, Methane and LPG. The sensor could be used to detect different gases contains CO and combustible gases, it is with low cost and suitable for different application.

### Character

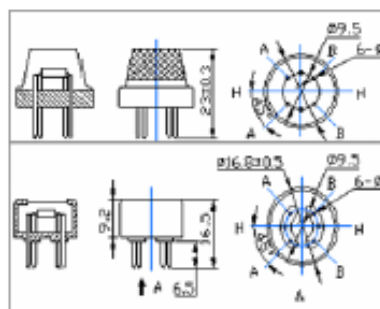
- \* Good sensitivity to CO/Combustible gas
- \* High sensitivity to Methane, Propane and CO
- \* Long life and low cost
- \* Simple drive circuit

### Application

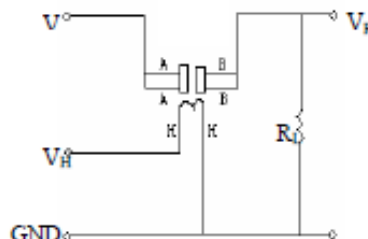
- \* Domestic gas leakage detector
- \* Industrial gas detector
- \* Portable gas detector

### Technical Data

### Configuration



### Basic test loop



The above is basic test circuit of the sensor. The sensor need to be put 2 voltage, heater voltage (V<sub>H</sub>) and test voltage (V<sub>C</sub>). V<sub>H</sub> used to supply certified working temperature to the sensor, while V<sub>C</sub> used to detect voltage (V<sub>R</sub>) on load resistance (R<sub>L</sub>) whom is in series with sensor. The sensor has light polarity, V<sub>c</sub> need DC power. V<sub>C</sub> and V<sub>H</sub> could use same power circuit with precondition to assure performance of sensor. In order to make the sensor with better

Model No.		MQ-9	
Sensor Type		Semiconductor	
Standard Encapsulation		Bakelite	
Detection Gas		CO and combustible gas	
Concentration		10-1000ppm CO 100-10000ppm combustible gas	
Circuit	Loop Voltage	V <sub>c</sub>	≤10V DC
	Heater Voltage	V <sub>H</sub>	5.0V±0.2V AC or DC (High) 1.5V±0.1V AC or DC (Low)
	Heater Time	T <sub>L</sub>	60±1S (High) 90±1S (Low)
	Load Resistance	R <sub>L</sub>	Adjustable
Character	Heater Resistance	R <sub>H</sub>	31Ω±3Ω (Room Tem.)
	Heater consumption	P <sub>H</sub>	≤350mW
	Sensing Resistance	R <sub>s</sub>	2KΩ-20KΩ(in 100ppm CO)
	Sensitivity	S	R <sub>s</sub> (in air)/R <sub>s</sub> (100ppm CO)≥5
	Slope	α	≤0.8(R <sub>300ppm</sub> /R <sub>100ppm</sub> CO)
Condition	Tem. Humidity	20℃±2℃; 65%±5%RH	
	Standard test circuit	V <sub>c</sub> :5.0V±0.1V; V <sub>H</sub> (High) : 5.0V±0.1V; V <sub>H</sub> (Low) : 1.5V±0.1V	
	Preheat time	Over 48 hours	

performance, suitable R<sub>L</sub> value is needed:

$$\text{Power of Sensitivity body}(P_s) : P_s = V_c^2 \times R_s / (R_s + R_L)^2$$

Tel: 86-371-87169070/80 Fax: 86-371-87169090

Email: [sales@hwsensor.com](mailto:sales@hwsensor.com) 3 | 72

Resistance of sensor( $R_s$ ):  $R_s=(V_o/V_{RL}-1)\times R_L$

**Sensitivity Characteristics**

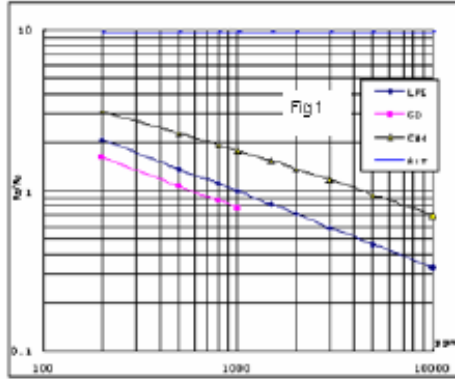


Fig.1 shows the typical sensitivity characteristics of the MQ-9, ordinate means resistance ratio of the sensor ( $R_s/R_o$ ), abscissa is concentration of gases.  $R_s$  means resistance in different gases,  $R_o$  means resistance of sensor in 1000ppm LPG. All test are under standard test conditions.

**Influence of Temperature/Humidity**

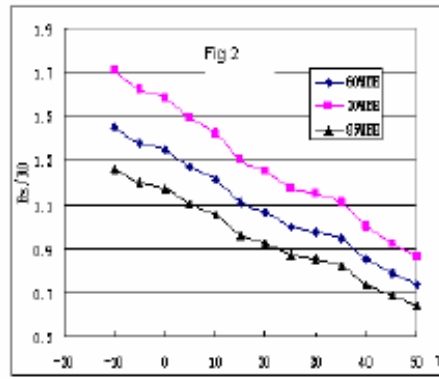
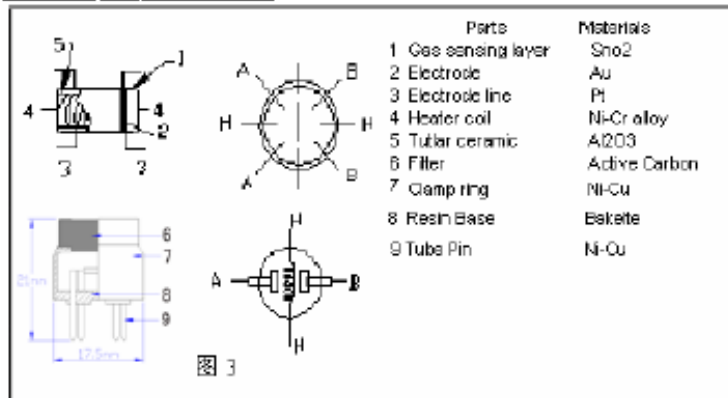


Fig.2 shows the typical temperature and humidity characteristics. Ordinate means resistance ratio of the sensor ( $R_s/R_o$ ),  $R_s$  means resistance of sensor in 1000ppm Propane under different tem. and humidity.  $R_o$  means resistance of the sensor in environment of 1000ppm Propane, 20°C/65%RH

**Structure and configuration**



Structure and configuration of MQ-9 gas sensor is shown as Fig. 3, sensor composed by micro Al<sub>2</sub>O<sub>3</sub> ceramic tube, Tin Dioxide (SnO<sub>2</sub>) sensitive layer, measuring electrode and heater are fixed into a crust made by plastic and stainless steel net. The heater provides necessary work conditions for work of sensitive components. The enveloped MQ-7 have 6 pin, 4 of them are used to fetch signals, and other 2 are used for providing heating current.

## **d) ADS 1115**



This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

### ORDERING INFORMATION

For the most current package and ordering information, see the Package Option Addendum at the end of this document, or see the TI web site at [www.ti.com](http://www.ti.com).

### ABSOLUTE MAXIMUM RATINGS<sup>(1)</sup>

	ADS1113, ADS1114, ADS1115	UNIT
VDD to GND	-0.3 to +5.5	V
Analog input current	100, momentary	mA
Analog input current	10, continuous	mA
Analog input voltage to GND	-0.3 to VDD + 0.3	V
SDA, SCL, ADDR, ALERT/RDY voltage to GND	-0.5 to +5.5	V
Maximum junction temperature	+150	°C
Storage temperature range	-60 to +150	°C

(1) Stresses above those listed under *Absolute Maximum Ratings* may cause permanent damage to the device. Exposure to absolute maximum conditions for extended periods may affect device reliability.

### PRODUCT FAMILY

DEVICE	PACKAGE DESIGNATOR MSOP/QFN	RESOLUTION (Bits)	MAXIMUM SAMPLE RATE (SPS)	COMPARATOR	PGA	INPUT CHANNELS (Differential/Single-Ended)
ADS1113	BROI/N6J	16	860	No	No	1/1
ADS1114	BRNI/N5J	16	860	Yes	Yes	1/1
ADS1115	BOGI/N4J	16	860	Yes	Yes	2/4
ADS1013	BRMI/N8J	12	3300	No	No	1/1
ADS1014	BRQI/N8J	12	3300	Yes	Yes	1/1
ADS1015	BRPI/N7J	12	3300	Yes	Yes	2/4



## ELECTRICAL CHARACTERISTICS

All specifications at  $-40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$ ,  $V_{\text{DD}} = 3.3\text{V}$ , and Full-Scale (FS) =  $\pm 2.048\text{V}$ , unless otherwise noted. Typical values are at  $+25^{\circ}\text{C}$ .

PARAMETER	TEST CONDITIONS	ADS1113, ADS1114, ADS1115			UNIT
		MIN	TYP	MAX	
<b>ANALOG INPUT</b>					
Full-scale input voltage <sup>(1)</sup>	$V_{\text{IN}} = (\text{AIN}_P) - (\text{AIN}_N)$		$\pm 4.096/\text{PGA}$		V
Analog input voltage	$\text{AIN}_P$ or $\text{AIN}_N$ to GND	GND		$V_{\text{DD}}$	V
Differential input impedance			See Table 2		
Common-mode input impedance	$\text{FS} = \pm 6.144\text{V}^{(1)}$		10		$\text{M}\Omega$
	$\text{FS} = \pm 4.096\text{V}^{(1)}$ , $\pm 2.048\text{V}$		6		$\text{M}\Omega$
	$\text{FS} = \pm 1.024\text{V}$		3		$\text{M}\Omega$
	$\text{FS} = \pm 0.512\text{V}$ , $\pm 0.256\text{V}$		100		$\text{M}\Omega$
<b>SYSTEM PERFORMANCE</b>					
Resolution	No missing codes	16			Bits
Data rate (DR)			8, 16, 32, 64, 128, 250, 475, 860		SPS
Data rate variation	All data rates	-10		10	%
Output noise		See Typical Characteristics			
Integral nonlinearity	$\text{DR} = 8\text{SPS}$ , $\text{FS} = \pm 2.048\text{V}$ , best fit <sup>(2)</sup>			1	LSB
Offset error	$\text{FS} = \pm 2.048\text{V}$ , differential inputs		$\pm 1$	$\pm 3$	LSB
	$\text{FS} = \pm 2.048\text{V}$ , single-ended inputs		$\pm 3$		LSB
Offset drift	$\text{FS} = \pm 2.048\text{V}$		0.005		$\text{LSB}/^{\circ}\text{C}$
Offset power-supply rejection	$\text{FS} = \pm 2.048\text{V}$		1		$\text{LSB}/\text{V}$
Gain error <sup>(3)</sup>	$\text{FS} = \pm 2.048\text{V}$ at $25^{\circ}\text{C}$		0.01	0.15	%
Gain drift <sup>(3)</sup>	$\text{FS} = \pm 0.256\text{V}$		7		$\text{ppm}/^{\circ}\text{C}$
	$\text{FS} = \pm 2.048\text{V}$		5	40	$\text{ppm}/^{\circ}\text{C}$
	$\text{FS} = \pm 6.144\text{V}^{(1)}$		5		$\text{ppm}/^{\circ}\text{C}$
Gain power-supply rejection			80		$\text{ppm}/\text{V}$
PGA gain match <sup>(3)</sup>	Match between any two PGA gains		0.02	0.1	%
Gain match	Match between any two inputs		0.05	0.1	%
Offset match	Match between any two inputs		3		LSB
Common-mode rejection	At dc and $\text{FS} = \pm 0.256\text{V}$		105		dB
	At dc and $\text{FS} = \pm 2.048\text{V}$		100		dB
	At dc and $\text{FS} = \pm 6.144\text{V}^{(1)}$		90		dB
	$f_{\text{CM}} = 60\text{Hz}$ , $\text{DR} = 8\text{SPS}$		105		dB
	$f_{\text{CM}} = 50\text{Hz}$ , $\text{DR} = 8\text{SPS}$		105		dB
<b>DIGITAL INPUT/OUTPUT</b>					
Logic level					
$V_{\text{IH}}$		$0.7V_{\text{DD}}$		5.5	V
$V_{\text{IL}}$		$\text{GND} - 0.5$		$0.3V_{\text{DD}}$	V
$V_{\text{OCL}}$	$I_{\text{OL}} = 3\text{mA}$	$\text{GND}$	0.15	0.4	V
Input leakage					
$I_{\text{H}}$	$V_{\text{IH}} = 5.5\text{V}$			10	$\mu\text{A}$
$I_{\text{L}}$	$V_{\text{IL}} = \text{GND}$	10			$\mu\text{A}$

(1) This parameter expresses the full-scale range of the ADC scaling. In no event should more than  $V_{\text{DD}} + 0.3\text{V}$  be applied to this device.

(2) 99% of full-scale.

(3) Includes all errors from onboard PGA and reference.

### ELECTRICAL CHARACTERISTICS (continued)

All specifications at  $-40^{\circ}\text{C}$  to  $+125^{\circ}\text{C}$ ,  $V_{\text{DD}} = 3.3\text{V}$ , and Full-Scale (FS) =  $\pm 2.048\text{V}$ , unless otherwise noted. Typical values are at  $+25^{\circ}\text{C}$ .

PARAMETER	TEST CONDITIONS	ADS1113, ADS1114, ADS1115			UNIT
		MIN	TYP	MAX	
<b>POWER-SUPPLY REQUIREMENTS</b>					
Power-supply voltage		2		5.5	V
Supply current	Power-down current at $25^{\circ}\text{C}$		0.5	2	$\mu\text{A}$
	Power-down current up to $125^{\circ}\text{C}$			5	$\mu\text{A}$
	Operating current at $25^{\circ}\text{C}$		150	200	$\mu\text{A}$
	Operating current up to $125^{\circ}\text{C}$			300	$\mu\text{A}$
Power dissipation	$V_{\text{DD}} = 5.0\text{V}$		0.9		mW
	$V_{\text{DD}} = 3.3\text{V}$		0.5		mW
	$V_{\text{DD}} = 2.0\text{V}$		0.3		mW
<b>TEMPERATURE</b>					
Storage temperature		$-60$		$+150$	$^{\circ}\text{C}$
Specified temperature		$-40$		$+125$	$^{\circ}\text{C}$

### PIN CONFIGURATIONS



### PIN DESCRIPTIONS

PIN #	DEVICE			ANALOG/ DIGITAL INPUT/ OUTPUT	DESCRIPTION
	ADS1113	ADS1114	ADS1115		
1	ADDR	ADDR	ADDR	Digital Input	$I^2\text{C}$ slave address select
2	NC <sup>(1)</sup>	ALERT/RDY	ALERT/RDY	Digital Output	Digital comparator output or conversion ready (NC for ADS1113)
3	GND	GND	GND	Analog	Ground
4	AIN0	AIN0	AIN0	Analog Input	Differential channel 1: Positive input or single-ended channel 1 input
5	AIN1	AIN1	AIN1	Analog Input	Differential channel 1: Negative input or single-ended channel 2 input
6	NC	NC	AIN2	Analog Input	Differential channel 2: Positive input or single-ended channel 3 input (NC for ADS1113/4)
7	NC	NC	AIN3	Analog Input	Differential channel 2: Negative input or single-ended channel 4 input (NC for ADS1113/4)
8	VDD	VDD	VDD	Analog	Power supply: 2.0V to 5.5V
9	SDA	SDA	SDA	Digital I/O	Serial data: Transmits and receives data
10	SCL	SCL	SCL	Digital Input	Serial clock input: Clocks data on SDA

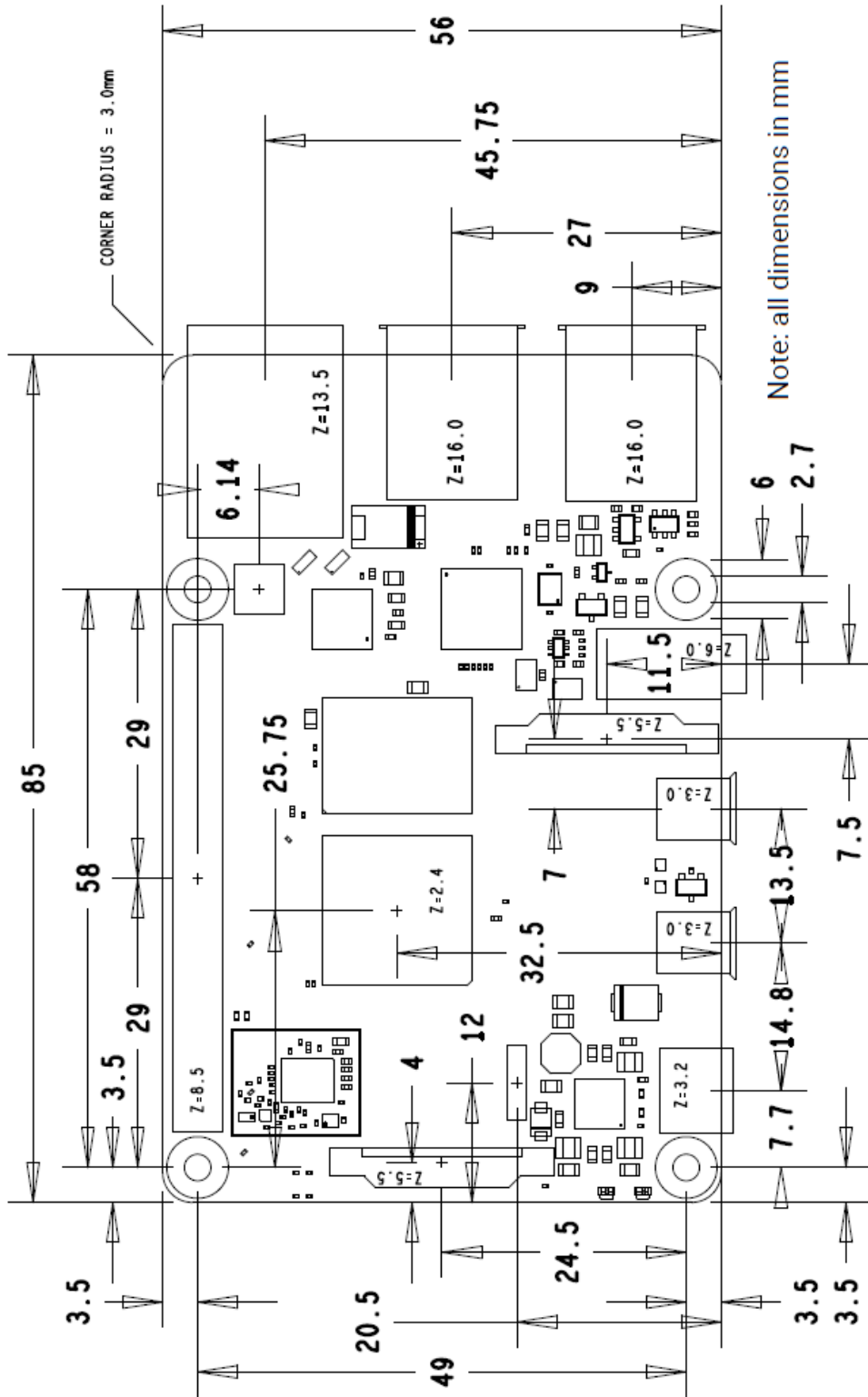
(1) NC pins may be left floating or tied to ground.

## **e) Raspberry Pi 4 model B**

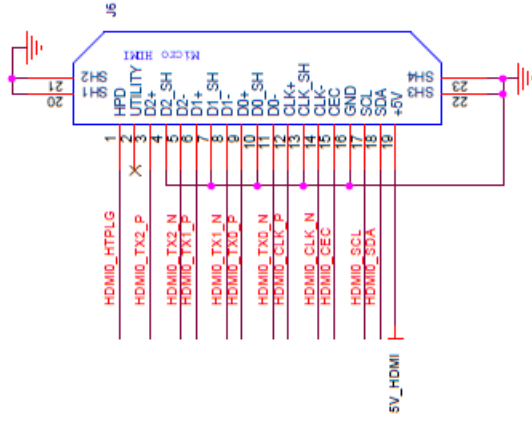
# Specification

<b>Processor:</b>	Broadcom BCM2711, quad-core Cortex-A72 (ARM v8) 64-bit SoC @ 1.5GHz
<b>Memory:</b>	1GB, 2GB, 4GB or 8GB LPDDR4 (depending on model) with on-die ECC
<b>Connectivity:</b>	2.4 GHz and 5.0 GHz IEEE 802.11b/g/n/ac wireless LAN, Bluetooth 5.0, BLE Gigabit Ethernet 2 × USB 3.0 ports 2 × USB 2.0 ports.
<b>GPIO:</b>	Standard 40-pin GPIO header (fully backwards-compatible with previous boards)
<b>Video &amp; sound:</b>	2 × micro HDMI ports (up to 4Kp60 supported) 2-lane MIPI DSI display port 2-lane MIPI CSI camera port 4-pole stereo audio and composite video port
<b>Multimedia:</b>	H.265 (4Kp60 decode); H.264 (1080p60 decode, 1080p30 encode); OpenGL ES, 3.0 graphics
<b>SD card support:</b>	Micro SD card slot for loading operating system and data storage
<b>Input power:</b>	5V DC via USB-C connector (minimum 3A*) 5V DC via GPIO header (minimum 3A*) Power over Ethernet (PoE)-enabled (requires separate PoE HAT)
<b>Environment:</b>	Operating temperature 0–50°C
<b>Compliance:</b>	For a full list of local and regional product approvals, please visit <a href="https://www.raspberrypi.org/documentation/hardware/raspberrypi/conformity.md">https://www.raspberrypi.org/documentation/hardware/raspberrypi/conformity.md</a>
<b>Production lifetime:</b>	The Raspberry Pi 4 Model B will remain in production until at least January 2026.

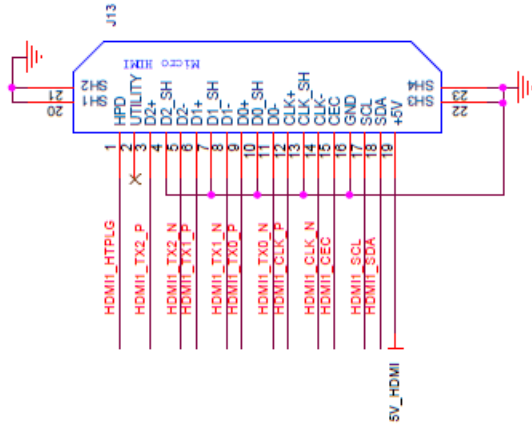
# Physical Specifications



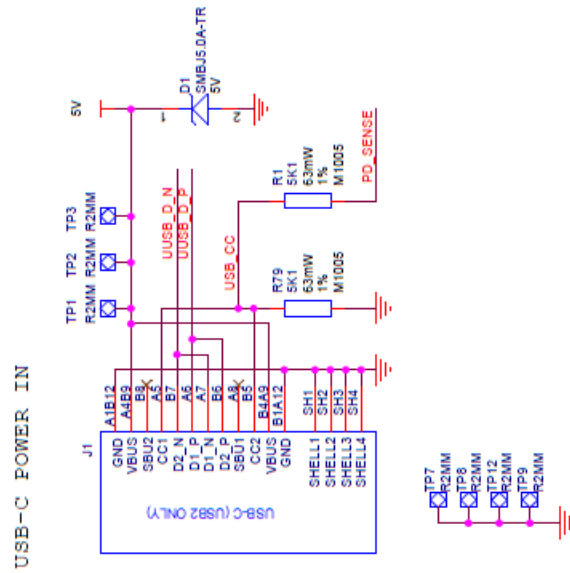
Note: all dimensions in mm

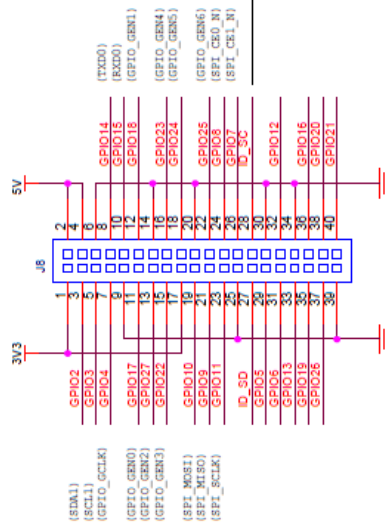


MICRO HDMI 0



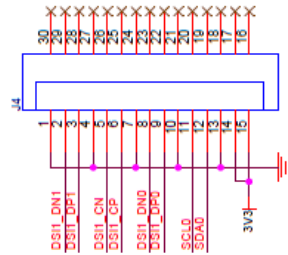
MICRO HDMI 1



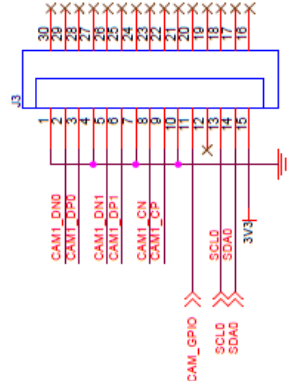


GPIO EXPANSION

**ID\_SD and ID\_SC PINS:**  
 These pins are reserved for BAT ID EEPROM. At boot time this I2C interface will be interrogated to look for an EEPROM that identifies the attached board and allows automatic setup of the GPIOs (and optionally, blink drivers).  
**DO NOT USE these pins for anything other than the ID EEPROM. The ID EEPROM must be connected if ID EEPROM not required.**

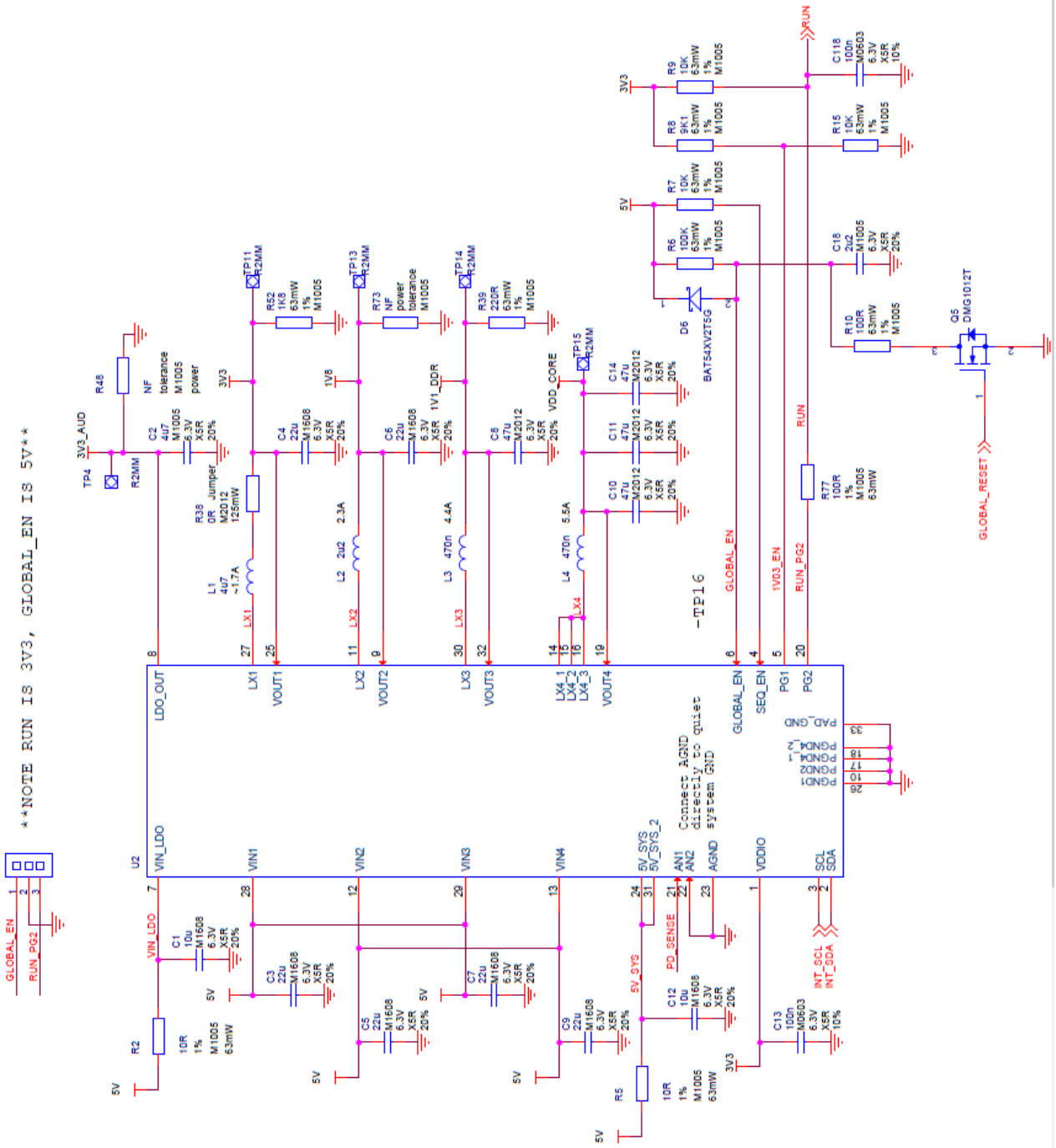


DISPLAY



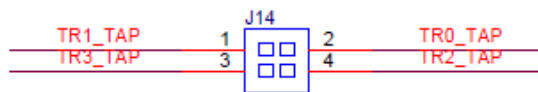
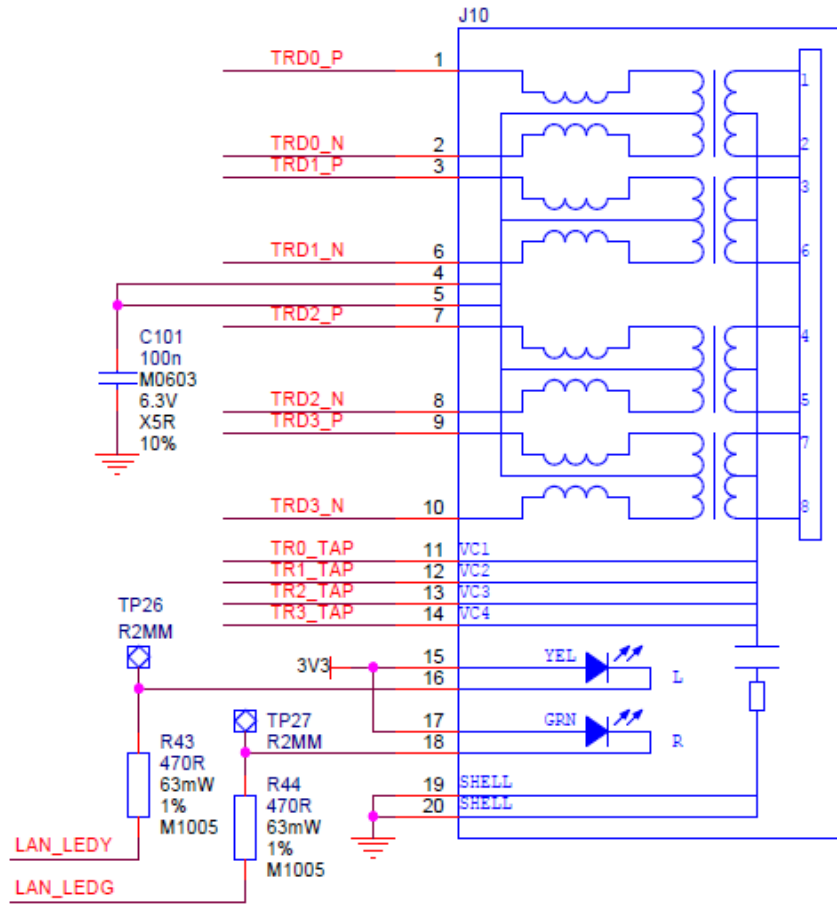
CAMERA

\*\*\*NOTE RUN IS 3V3, GLOBAL\_EN IS 5V\*\*\*





# GIGABIT ETHERNET



PoE HEADER