ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Diseño e Implementación de un Sistema de Control Automático para Oxigenoterapia Respiratoria No Invasiva de Alto Flujo

TRABAJO DE TITULACIÓN

Previo a la obtención del Título de:

MAGÍSTER EN INGENIERÍA BIOMÉDICA

Presentado por:

Iván Javier Bravo Lalama

GUAYAQUIL - ECUADOR

Año: 2022

DEDICATORIA

El presente trabajo se lo dedico a mi mamá, Roma Lalama; a mi papá, Iván Bravo; a mi novia, Cindy Vanessa; a mis hermanos, Javier, Andrés, Karla, Vero, Jaime y Lili; a mis tíos, Nacha, Carolina y Jaime; y a mi abuelita Piedad.

AGRADECIMIENTOS

A mi tutor, Geovanny Arguello, por su valiosa ayuda en el desarrollo de este proyecto, sus conocimientos fueron fundamentales.

DECLARACIÓN EXPRESA

“Los derechos de titularidad y explotación, me corresponde conforme al reglamento de propiedad intelectual de la institución; yo, *Iván Javier Bravo Lalama,* doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"

Iván Bravo Lalama

TRIBUNAL DE EVALUACIÓN

Ph.D. María Antonieta Álvarez Villanueva
SUBDECANA DE LA FIEC

Msc. Geovanny Argüello Andrade
DIRECTOR DEL TRABAJO DE TITULACIÓN

Ph.D. César Martín Moreno
MIEMBRO PRINCIPAL DEL TRIBUNAL

RESUMEN

En oxigenoterapia, según el estado del paciente, cuadro clínico, y dependiendo de las causas de la hipoxemia, se determina el método de administración de oxígeno; el mismo que siempre dependerá de la asistencia de un profesional especializado. Con este antecedente parte la idea de realizar un sistema de control automático que pueda emplearse en ambientes no hospitalarios, cuya función sea la de ajustar la fracción de inspiración de oxígeno, FiO2, dependiendo la condición del paciente. Para esto se deben adquirir parámetros fisiológicos, ya que son indicadores del funcionamiento de órganos y sistemas de un ser vivo, tales como la frecuencia cardiaca y saturación de oxigeno (SpO2), que se obtienen mediante oximetría de pulso, cuyo principio es la fotopletismografía, y conocer su relación con la FiO2 requerida. Para esto se diseñó el sistema de control automático mediante válvulas neumáticas, las mismas que son controladas por modulación de ancho de pulso, PWM. El módulo de control empleado es una Raspberry Pi 3 B+ que recibe los datos obtenidos por el oxímetro de pulso y por la celda de oxígeno y a su vez, genera señales PWM, indicadores y alarma.

Los resultados obtenidos, con la programación realizada, dan a conocer que la FiO2 generada por las válvulas, se ajustan correctamente a lo requerido por un paciente, ya que la lectura de la celda de oxígeno coincide, con un error menor al 4%, a los parámetros configurados. Este dispositivo podría ser utilizado en ambientes no hospitalarios, siempre y cuando haya sido probado en pacientes patológicos.

Los parámetros fisiológicos SpO2, ritmo cardiaco y la FiO2 entregada, mediante la plataforma Thingsboard, pueden ser transmitidos a un médico y/o centro de salud para que puedan ser monitorizados remotamente, es decir, Telemonitorización.

Palabras Clave: Control automático, Fracción de inspiración de oxígeno, Oxigenoterapia de alto flujo, Saturación de oxígeno.

ABSTRACT

In oxygen therapy, depending on the patient's condition, symptoms, and depending on the causes of hypoxemia, the method of oxygen administration is determined; the same one that will always depend on the assistance of a specialized professional. With this background, the idea of making an automatic control system that can be used in non-hospital environments, whose function is to adjust the fraction of inspiration of oxygen, FiO2, depending on the patient's condition, starts. For this, physiological parameters must be acquired, since they are indicators of the functioning of organs and systems of a living being, such as heart rate and oxygen saturation (SpO2), which are obtained by pulse oximetry, whose principle is photoplethysmography, and know its relationship with the required FiO2. For this, the automatic control system was designed using pneumatic pumps, which are controlled by pulse width modulation, PWM. The control module used is a Raspberry Pi 3 B + that receives the data obtained by the pulse oximeter and the oxygen cell and, in turn, generates PWM signals, indicators and alarm.

The results obtained, with the programming carried out, reveal that the FiO2 generated by the valves are correctly adjusted to what is required by a patient, since the reading of the oxygen cell coincides, with an error of less than 4%, at the configured parameters. This device could be used in non-hospital environments, as long as it has been tested on pathological patients.

Physiological parameters SpO2, heart rate and FiO2 delivered, through the Thingsboard platform, can be transmitted to a doctor and/or health center so that they can be monitored remotely, that is, Telemonitoring.

Keywords: Automatic Control, Fraction of Inspiratory Oxygen, High Flow Oxygen Therapy, Oxygen Saturation

ÍNDICE GENERAL

ABREVIATURAS

API	Interfaz de programación de aplicaciones
CO2Hb	Carbaminohemoglobina
COHb	Carboxihemoglobina
DC	Corriente directa
CO2	Dióxido de carbono
EPOC	Enfermedad pulmonar obstructiva crónica
ESPOL	Escuela Superior Politécnica del Litoral
FiO2	Fracción de inspiración de oxígeno
BPM	Latidos por minuto
HB	Hemoglobina
MetHb	Metahemoglobina
O2	Oxígeno
PaO2	Presión parcial de oxígeno
SpO2	Saturación de oxígeno medida por oximetría de pulso
IoT	Internet de las cosas, por sus siglas en inglés

SIMBOLOGÍA

Mg	Miligramo
Ph	Potencial de Hidrógeno
GPH	Galones por hora
GHz	GigaHertz
°C	Grados Celsius
g	Gramo
L	Litro
l/min	Litros por minuto
mA	Miliamperio
mm	Milímetro
mmHg	Milímetro de mercurio
mV	Milivoltio
um	Micrómetro
nm	Nanómetro
uA	Microamperio
V	Voltio
W	Vatio

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

CAPÍTULO 1

1. INTRODUCCIÓN

1.1 Identificación del problema

Los pacientes que requieren oxigenoterapia dependen siempre de asistencia profesional debido a que una baja concentración de oxígeno o Fracción de Inspiración de Oxígeno (FiO2) puede provocar desde un desmayo, hasta llegar a hipoxia (falta de oxígeno en tejidos) cuyos síntomas iniciales son un aumento en la frecuencia respiratoria y en el ritmo cardiaco; además de ansiedad, fatiga y dolor de cabeza. Si la hipoxia persiste, los síntomas se agudizan y el funcionamiento cerebral puede comprometerse. El corazón acelera su ritmo cardiaco para tratar de suministrar mayor cantidad de oxígeno, ocasionando taquicardia. Si la hipoxia no es tratada, la presión arterial disminuirá, causando la muerte. [1] [2]

Por el contrario, si la FiO2 es muy elevada, puede provocar proceso de envejecimiento prematuro, hospitalización mayor a la requerida, infecciones, daño cerebral, carcinogénesis, daño a nivel genético, displasia broncopulmonar, [3] [2]

Es necesario un sistema automático que permita ajustar la FiO2 dependiendo de la condición del paciente y evitar complicaciones en un paciente que ya se encuentra delicado de salud.

1.2 Justificación del problema

En la actualidad, la medicina y la ingeniería convergen en diferentes disciplinas, principalmente en la electrónica. Los equipos médicos electrónicos son indispensables para dar un diagnóstico más completo y confiable. Debido a esto los objetivos que se alcanzan actualmente son mayores y permiten mejores prestaciones de los servicios, incluso que un paciente pueda realizar, desde un ambiente domiciliario, el control de diversos parámetros fisiológicos, como presión sanguínea, ritmo cardíaco, saturación de oxígeno, glucosa, etc. [4]

La saturación de oxígeno, porcentaje de hemoglobina libre, ritmo cardiaco, se pueden obtener de manera no invasiva gracias a la fotopletismografía. [5] [4] [6]..

La fotopletismografía, entre otras definiciones, es una técnica que se basa en medir la cantidad de luz, visible e IR, que es absorbida por la hemoglobina, oxihemoglobina, carbaminohemoglobina, etc.

El oxígeno es el tratamiento primario para pacientes con insuficiencia respiratoria aguda. La oxigenoterapia de alto flujo [7] , mayor a 6 Lpm, necesita mascarillas especiales para controlar el FiO2 adecuado , pero con el sistema propuesto, se necesitará una mascarilla normal o cánula, y el FiO2 será regulado por el sistema de control. Esto permitirá un mejor control del FiO2 y tendrá retroalimentación para ajustar de forma automática, dependiendo de los parámetros fisiológicos del paciente.

Un sistema de oxigenoterapia automático eficiente permitirá mayor seguridad y mejores resultados en cuanto a la recuperación del paciente se refiere. [8]

Para la oportuna y eficiente transmisión de los parámetros que indican el estado del paciente, es necesario el uso de la Telemedicina, particularmente, servicios como Telemonitorización.

Debido a que es software libre, la Telemonitorización se basará en la plataforma Thingsboard. Mediante el uso de esta herramienta, se pueden observar datos enviados por un módulo de control, para que puedan ser observados en una locación remota. [9]

Con el software Docker, se puede visualizar el entorno gráfico de un dominio web para poder Telemonitorizar los parámetros fisiológicos. [10]

Esto permite que el o los profesionales de salud, puedan evaluar a un paciente en ambientes no hospitalarios o incluso, rurales, facilitando que la atención sea oportuna, y en casos críticos, aumentar las probabilidades de salvar la vida de un paciente en riesgo. [11]

1.3 Objetivos

1.3.1 Objetivo General

Diseñar un sistema de control para oxigenoterapia respiratoria no invasiva de alto flujo, cuya implementación se realizará mediante el uso complementario de componentes electrónicos y mecánicos, para asistir a pacientes disneicos.

1.3.2 Objetivos Específicos

1. Analizar mediante revisión de documentos existentes, curvas de absorbancia de Hemoglobina (Hb), Oxihemoglobina (HbO2), Carbaminohemoglobina (CO2Hb). Esto permitirá conocer las longitudes de onda adecuadas en fotopletismografía para cada uno de estos parámetros fisiológicos.
2. Obtener datos de parámetros fisiológicos de Saturación de Oxígeno (SpO2), ritmo cardiaco, con base en el sistema propuesto.
3. Analizar la relación entre el FiO2 requerido y parámetros fisiológicos como SpO2.
4. Diseñar un sistema de control automático mediante electroválvulas neumáticas dependiente de los parámetros fisiológicos obtenidos, para el suministro exacto de FiO2.
5. Implementar el sistema de control propuesto utilizando fotopletismografía.
6. Evaluar aspectos técnicos como confiabilidad, rendimiento, facilidad de mantenimiento, seguridad; y aspectos médicos, con sujeto de control sano, el desempeño del sistema de control propuesto.

1.4 Metodología

Se diseñará un sistema de oxigenoterapia, utilizando un módulo comercial basado en microcontroladores, el mismo que permitirá mediante programación, la comunicación y acoplamiento de los diferentes sensores, para obtener los parámetros fisiológicos requeridos y posteriormente analizar mediante software, las curvas de absorbancia y en caso de requerirse, capnometría. Este circuito de adquisición basado en fotopletismografía, proporcionará las señales de entrada necesarias para modificar la Fracción de Inspiración de Oxígeno -FiO2-. El control será realizado por una tarjeta electrónica programable, que recibirá de forma alámbrica, las señales del circuito de adquisición y modulará sus salidas para controlar la apertura de dos electroválvulas, las cuales suministrarán oxígeno y aire respectivamente. La proporción del oxígeno en la mezcla, es la FiO2. La mezcla se entregará al paciente con mascarilla o cánula y la FiO2 se ajustará automáticamente según se modifiquen los parámetros fisiológicos en el paciente durante el suministro de oxígeno. La tarjeta de control tendrá la capacidad de alertar, de forma audible y visible, mediante leds y bocina, cuando no se detecten señales de entrada o cuando los parámetros fisiológicos decaigan.

Para la evaluación del sistema propuesto se utilizará un sujeto de control sano, el cual dará su consentimiento escrito y posterior a las pruebas, su impresión del procedimiento mediante encuesta.

Como aporte adicional, se implementará, mediante la plataforma Thingsboard, un sistema de telemonitorización de los parámetros fisiológicos y FiO2 entregado por el sistema.

1.5 Alcance

El sistema a diseñar, el cuál será un prototipo completo, permitirá el suministro exacto y automáticamente ajustable de FiO2, a un paciente que requiera oxigenoterapia, dependiendo de sus parámetros fisiológicos. El sistema deberá ser confiable para garantizar la seguridad y óptima recuperación del paciente. Para esto se realizará:

- Evaluación de aspectos técnicos, como confiabilidad, rendimiento, facilidad de mantenimiento, seguridad.
- Con un sujeto de control sano, será evaluado por uno o más profesionales de salud para determinar su usabilidad, confiabilidad, efectividad, utilidad y aceptación.

CAPÍTULO 2

2. ESTADO DEL ARTE

2.1 Conceptos

2.1.1 Oxígeno y dióxido de carbono

2.1.1.1 Oxígeno

El oxígeno es uno de los elementos presentes en la naturaleza, de representado por el símbolo “O” y con número atómico “8”. Su nombre proviene del griego “Oxys”, en la tabla periódica forma parte de los anfígenos y es un agente oxidante. Con presión y temperatura normal, el oxígeno, al igual que la mayoría de gases, con excepción de los gases nobles, está libre en el ambiente formando una molécula de dos átomos de oxígeno (O2), la cual es indispensable para la vida en la tierra. En la atmósfera, el O2 constituye el 20.8% del volumen total.

Dado que la mayor parte del agua es oxígeno, este último es también aporta la mayoría de masa en los seres vivos, forma parte de lípidos, proteínas, hormonas, enzimas, carbohidratos, huesos, dientes. [12]

Es producido por las plantas, algas y bacterias fotosintéticas a través de la fotosíntesis, proceso que se describe a continuación:

$$6\ CO_2 + 6\ H_2O + \text{fotones} \rightarrow C_6H_{12}O_6 + 6\ O_2 \qquad (2.1)$$

dióxido de carbono + agua + luz solar → glucosa + Oxígeno gaseoso

El oxígeno es indispensable para la respiración celular en todos los organismos aerobios. Las mitocondrias, organelo celular e ncargada del metabolismo celular, utilizan el oxígeno para producir ATP (Adenosín Trifosfato) durante la respiración aeróbica.

$$C_6H_{12}O_6 + 6\ O_2 \rightarrow 6\ CO_2 + 6\ H_2O + 2880\ kJ{\bullet}mol^{-1} \qquad (2.2)$$

En los vertebrados, el O_2 es entregado al cuerpo por un intercambio gaseoso que ocurre en los alveolos pulmonares. Los glóbulos rojos o eritrocitos, son las células encargados de transportar el O_2. [13]

2.1.1.2 Dióxido de carbono

El Dióxido de carbono o anhídrido carbónico (CO_2), es una molécula formada por un átomo de carbono y dos de oxígeno. Su concentración en la atmósfera es 0.04%. Es producido naturalmente por volcanes, que emiten cerca de 0,3 mil millones de toneladas anualmente, aguas termales y es liberado por rocas carbonatadas al diluirse en agua. La actividad humana produce cerca de 30 mil millones de toneladas cada año.

El CO2 es regulado en la naturaleza por organismos fotosintéticos, como plantas, algas, bacterias, que lo utilizan, en conjunto con agua, para sintetizar sacáridos, produciendo O2 como residuo.

Es producto de la respiración de todos los organismos aerobios y también de la combustión de materia orgánica. [14]

$$C_6H_{12}O_6 + 6\ O_2 \rightarrow 6\ CO_2 + 6\ H_2O + 2880\ \text{kJ}\cdot\text{mol}^{-1} \qquad (2.3)$$

Este gas molecular es el mayor responsable del efecto invernadero.

2.1.2 La sangre

Es un tejido conectivo que recorre el cuerpo de los vertebrados a través del sistema circulatorio.

Su color rojizo se debe a que está constituido principalmente por glóbulos rojos, los cuales portan hemoglobina, proteína de este color.

La sangre es producida en la médula ósea, presente en el tejido esponjoso de los huesos planos y en el interior de los huesos largos, presentes en las extremidades. Aporta cerca del 7% de la masa corporal y su pH se encuentra en 7.33 y 7.44. [15]

2.1.2.1 Funciones de la sangre

Dentro de las funciones de la sangre se encuentra:

- Protección del cuerpo contra infecciones.
- Transporte de nutrientes a los diferentes tejidos
- Transporte de oxígeno
- Transporte de desechos del organismo, como CO2
- Transporta calor y participa en la termorregulación del cuerpo

- Transporte de sustancias reguladoras, como enzimas y hormonas
- Encargada de la coagulación y cicatrización inicial. [15] [16]

2.1.2.2 Composición de la sangre

La sangre se compone de plasma o suero y células sanguíneas. [17]

1. Plasma sanguíneo: Líquido levemente amarillento que presenta el componente extracelular en el que se suspenden las células sanguíneas, conforma el 55% de la sangre y está formado cerca el 90% de agua. Se encarga del transporte de nutrientes, proteínas, electrolitos, sustancias reguladoras como hormonas, enzimas, desechos metabólicos y además aporta en la coagulación. Es un medio isotónico para las células, las cuales requieren concentración de 0.9% para realizar sus procesos vitales.

2. Hematocritos: Conforman el 45% de la sangre y se dividen en:
    a. Glóbulos rojos, hematíes o eritrocitos: Conforman el 95% de los hematocritos. Estas células tienen forma bicóncava y carecen de núcleo y organelos. Su citoplasma está constituido por hemoglobina, proteína encargada de transportar oxígeno y dióxido de carbono. Diámetro aproximado de 8 [um].

    b. Glóbulos blancos: También conocidos como leucocitos, su función principal es la protección ante organismos patógenos y células anormales, es decir, conforman el sistema inmunitario. Son capaces de producir anticuerpos. Tamaño aproximado de [10 um]. Se dividen en:
    - Agranulocitos:
        - Linfocitos: Actúan en infecciones virales y cáncer.
        - Monocitos: Actúan contra virus, parásitos, tumores y leucemias.
    - Granulocitos:
        - Basófilos: Contribuyen con el proceso antiinflamatorio.
        - Eosinófilos: Actúan contra parásitos, alergias y asma.
        - Neutrófilos: Actúan en la fagocitosis.

c. Plaquetas: Son células sin núcleo y sin color. Son capaces de formar trombina, sustancia que inicia la coagulación. Tamaño aproximado de 3 [um].

2.1.3 Hemoglobina

Se encuentra dentro de los glóbulos rojos, es una proteína de estructura cuaternaria, cada una de ellas posee un átomo de hierro, el cual tiene la capacidad de ligarse, de forma reversible, a una molécula de oxígeno. Un gramo de hemoglobina transporta 1.34 [mL] de oxígeno.

Cuando la hemoglobina se combina con el O_2, se conoce como oxihemoglobina o HbO2, y se conoce como carbaminohemoglobina o HbCO2, cuando se combina con el CO_2. La hemoglobina no oxigenada también es llamada desoxihemoglobina, hemoglobina reducida o RHb. Existen también dos moléculas adicionales, carboxihemoglobina COHb y metahemoglobina MetHb, pero se encuentran en mínimas concentraciones, excepto en alguna condición patológica relacionada, por ejemplo, asfixia por inhalación de monóxido de carbono.

La oxihemoglobina, carbaminohemoglobina y la desoxihemoglobina, tienen diferentes curvas de absorbancia, la cuales se describirán posteriormente. [15]

2.1.4 Intercambio y transporte de oxígeno y dióxido de carbono

El transporte de oxígeno y dióxido de carbono, como se mencionó en el punto anterior, se produce cuando cada uno de ellos se liga a la hemoglobina presente en el interior de los glóbulos rojos.

El aire inspirado llega a los pulmones luego de atravesar todo el sistema respiratorio, nariz, cavidad nasal, laringe, tráquea bronquios, bronquiolos y alveolos pulmonares. El intercambio de estas moléculas gaseosas se produce en el sistema respiratorio, específicamente, en los alveolos pulmonares, los cuales son estructuras microscópicas, con forma de saco con aire. Debido a que sus paredes celulares son muy finas, permiten el intercambio gaseoso entre la sangre y el aire inspirado. El CO_2 transportado por la sangre es liberado y expulsado del organismo mediante la expiración y el O_2 inspirado se liga a la molécula de hemoglobina. [18]

La frecuencia normal respiratoria en un adulto sano es entre 12 y 18 respiraciones por minuto y en cada una de ellas, el volumen inspirado es de aproximadamente 500 [cc]. El volumen inspirado se conoce como volumen tidal. La eficiencia del intercambio gaseoso está dada por el tiempo de contacto de los eritrocitos con la superficie alveolar y por la concentración de eritrocitos en la sangre. [19]

2.2 Fundamentos físicos

2.2.1 Fracción de inspiración de oxígeno

La fracción de inspiración de oxígeno o FiO_2 es la proporción de oxígeno en el aire ambiente que se mide. En condiciones normales, corresponde a la concentración de oxígeno en el aire, es decir 21%, o FiO2 de 0.21. [20]

2.2.2 Presión parcial de oxígeno

La presión parcial de oxígeno o PaO2, es la medición de las partículas de oxígeno disueltas en la sangre, las unidades utilizadas son mmHg o kPA. Este valor se mantiene estable en todo el árbol arterial, debido a que no hay consumo de oxígeno. Se entiende por lo tanto que esta medición denota el grado de oxemia. Hace referencia a la presión que ejerce el oxígeno que está disuelto en el plasma sanguíneo. [20]

Tabla 2.1 Relación Nivel de Oxígeno y PaO2 [20]

Nivel de oxígeno en la sangre	PaO2 (Presión parcial de oxígeno en la sangre) [mmHg]
Normal	80 - 100
Hipoxemia leve	71 - 80
Hipoxemia moderada	61 - 70
Hipoxemia grave	40 - 60
Daño cerebral y miocárdico	< 40
Riesgo de muerte inminente	< 20

Al respirar aire ambiente y al nivel del mar, debe ser siempre superior a 90 [mmHg].

Altos niveles de PaO2 se considera hiperoxemia, la cual puede ser tóxica ya que se producen radicales libres de oxígeno. [21]

2.2.3 Ley de Beer-Lambert

La absorción de la luz está descrita por la ley de Beer-Lambert, ésta establece que cada sustancia absorbe luz, dependiendo de la longitud de onda, en distintas cantidades. El modelo matemático relaciona la absorción de la luz con la concentración de la sustancia. Toda sustancia obedece esta ley, incluyendo la sangre y el oxígeno dentro de ella, por lo que es posible conocer la SpO_2. [22]

La absorbancia de un material dependerá de lo siguiente:

1. La concentración molar del material
2. La distancia que la luz debe atravesar dentro del material, desde su incidencia hasta que sale de él.
3. Coeficiente de extinción o absortividad: La probabilidad de que un fotón sea absorbido completamente por el material.

La relación matemática de la absorbancia del material es la siguiente:

$$A = - \varepsilon c d \tag{2.4}$$

Dónde:

A: Absorbancia o densidad óptica

ε: Coeficiente molar de extinción

d: Distancia que debe atravesar la luz dentro del material

c: Concentración molar del material

El porcentaje de un haz de luz que atraviesa un material se conoce como Transmitancia. La transmitancia se define matemáticamente como:

$$T = Io / Ii \times 100\% \tag{2.5}$$

Dónde

T: Transmitancia

Io: Intensidad de luz que atraviesa el material.

Ii: Intensidad de luz

Adicionalmente, podemos relacionar la transmitancia con la absorbancia mediante la siguiente ecuación:

$$A = \text{Log}\,(1/T) \qquad (2.6)$$

2.2.4 Cálculo de la saturación de oxígeno

La saturación de oxígeno se puede conocer mediante el uso de la oximetría de pulso. El principio de funcionamiento de este dispositivo es la ley de Beer-Lambert, permitiendo un análisis espectrofotométrico, es decir, las porciones de luz transmitida y absorbida por la hemoglobina.

El oxímetro de pulso, dispositivo electrónico útil para determinar la saturación de oxígeno o SpO2, emite dos haces de luz, roja, perteneciente al espectro visible, e infrarroja, la cual no es visible para el ojo humano, con 660 [nm] y 940 [nm] respectivamente. La luz roja es absorbida principalmente por la desoxihemoglobina y la luz infrarroja se absorbe principalmente por la oxihemoglobina.

Cuando se utiliza un oxímetro de pulso, los haces de luz deben atravesar diferentes tejidos, como piel, músculo, hueso, uña, y adicionalmente, sangre, que puede ser venosa o arterial.

Solo la sangre arterial pulsátil presenta variaciones de absorbancia en el tiempo. A este componente se le conoce como AC.

Se conoce como componente DC a la proporción de luz absorbida por los diferentes tejidos en los que no hay diferencias en el tiempo. [23] [24]

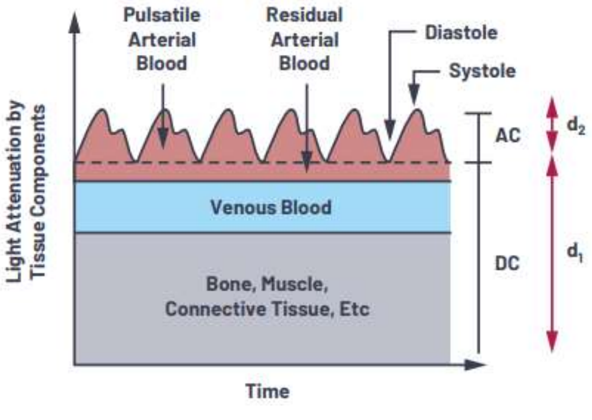


Figura 2.1 Atenuación de la luz a través del tejido. [23]

La relación de absorbancias realizada por un dispositivo de oximetría de pulso se puede realizar de la siguiente manera:

$$\frac{(AC_{660}) / (DC_{660})}{(AC_{940}) / (DC_{940})} \qquad (2.7)$$

En esta relación, una proporción de 1, equivale a SpO2 de 85%; una proporción de 0.4, equivale a SpO2 de 100%; y una proporción de 3.4, equivale a una SpO2 de 0%.

Otra forma de calcular la saturación de oxígeno se basa en calcular únicamente con el componente AC. La relación es la siguiente:

$$SpO2 = R \times 100 \qquad (2.8)$$

$$R = \log(I_{ac1}) / \log(I_{ac2}) \qquad (2.9)$$

Dónde:
Iac1, corresponde a la intensidad de la luz roja recibida
Iac2, corresponde a la intensidad de la luz infrarroja recibida

En la siguiente gráfica se representa una señal típica de oximetría de pulso. La señal representa la absorción de la sangre arterial.

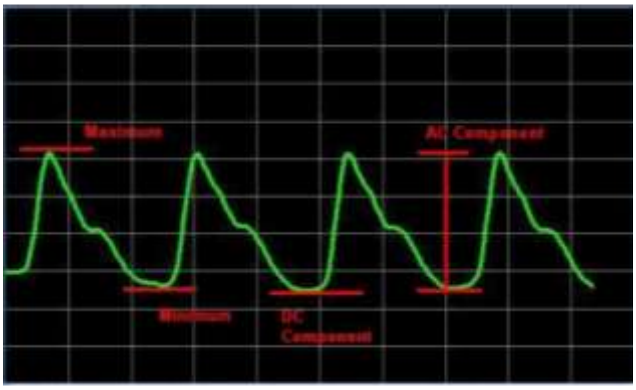


Figura 2.2 Componente AC de la señal capturada por oximetría de pulso. [25]

En la siguiente figura se muestra un diagrama de bloques básico de un oxímetro de pulso.

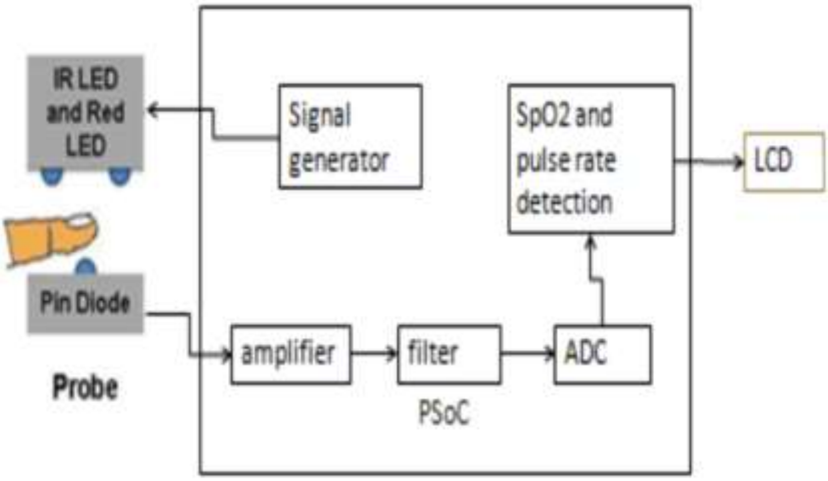


Figura 2.3 Diagrama de bloques general de un oxímetro de pulso [26]

2.2.5 Curvas de absorbancia de hemoglobina, oxihemoglobina, carboxihemoglobina, y metahemoglobina

Con el uso de la fotopletismografía podemos conocer la cantidad de luz absorbida por el cuerpo y, por tanto, la luz que lo atraviesa. La luz absorbida dependerá de la longitud de onda y de las características del medio, el cual puede ser, desoxihemoglobina o RHb, oxihemoglobina o HbO2, carboxihemoglobina o HbCO y metahemoglobina o HbMet. [23]

Las curvas de absorbancia de RHb, HbO2, HbCO y HbMet, se presentan en la siguiente imagen.

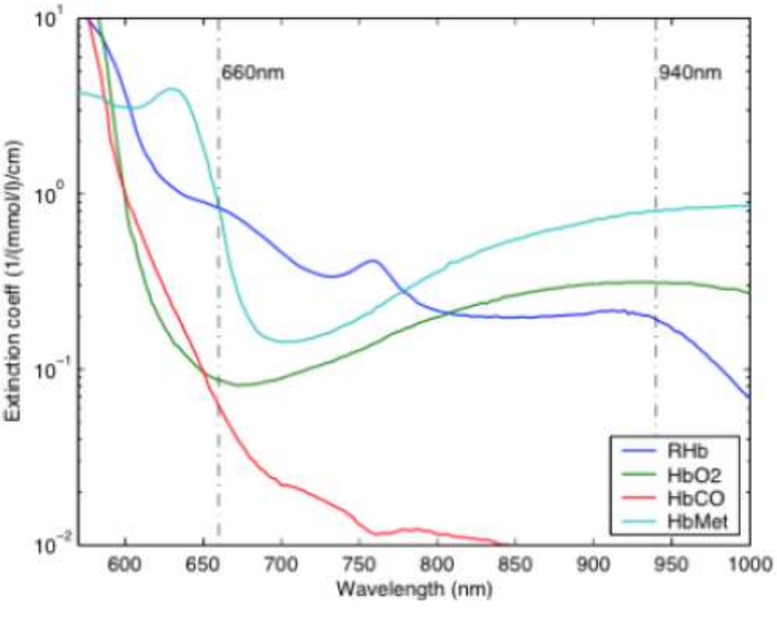


Figura 2.4 Curvas de absorbancia de Hemoglobina reducida, Oxihemoglobina, Carboxihemoglobina y Metahemoglobina [27]

2.3 Oxigenoterapia de alto flujo

2.3.1 Conceptos

2.3.1.1 Oxigenoterapia

Procedimiento terapéutico que tiene como objetivo, aumentar el contenido de oxígeno en las células de los diferentes tejidos en el organismo, aumentando la concentración de O2 en el gas inspirado y de esta manera, tratar o evitar la hipoxemia y sus efectos. La presión parcial de O2 en la sangre es proporcional a la FiO2, por lo tanto, una concentración mayor de O2 en el aire inspirado, restituirá la oxihemoglobina en los eritrocitos y permitirá mantener una presión parcial de oxígeno (PaO2) superior a 80 [mmHg] o una saturación de oxígeno igual (SpO2) o mayor a 92 %. El intercambio gaseoso ocurre en los alveolos y mayor concentración de O2, eleva la tensión de este gas en ellos, permitiendo absorber mayor cantidad de O2 a través de la membrana alveolo-capilar.

La concentración de oxígeno en el aire inspirado, normalmente es regulada adaptando el flujo de este gas mediante el uso de válvulas manuales. El O2 es almacenado en tanques o puede existir una toma de oxígeno medicinal y por lo general se administra a presión atmosférica.

La oxigenoterapia puede ser de bajo flujo y de alto flujo. La diferencia entre ellas consiste en que la primera no alcanza el requerimiento de volumen máximo del paciente, por lo tanto, la diferencia se complementa con aire ambiental, reduciendo la efectividad de este método.

La oxigenoterapia de alto flujo, por el contrario, supera la demanda de gases del paciente, incluso llegando al 100% de O2, debido a esto, el O2 debe ser humidificado para no causar efectos adversos innecesarios. Dado que el paciente recibirá únicamente la mezcla de gases del módulo de oxigenoterapia de alto flujo, se puede controlar con mayor exactitud la proporción de O2 y aire, es decir, FiO2. [28] [29]

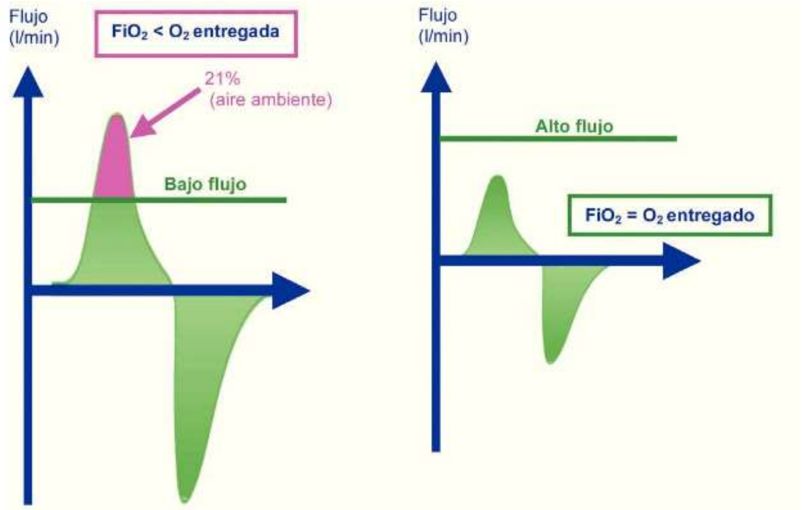


Figura 2.5 Comparación entre bajo flujo (izquierda) y alto flujo (derecha). Con bajo flujo, el paciente completa su pico de inspiración, con aire ambiental. Con alto flujo, la totalidad de la mezcla aire oxígeno, es suministrada por el dispositivo. [29]

2.3.2 Elementos en la oxigenoterapia de alto flujo

Los elementos utilizados en la oxigenoterapia de alto flujo son:

- Fuente o suministro de oxígeno
  - Cilindro de oxígeno
  - Red central de oxígeno
- Manómetro y válvula reguladora de presión: El primero de ellos mide la presión al interior del tanque de oxígeno. La válvula permite regular la presión de oxígeno
- Flujómetro o caudalímetro: Permite controlar la cantidad de litros por minuto que salen de la fuente de oxígeno
- Humidificador: El oxígeno entregado por la fuente de oxígeno es seco y frío, por lo tanto, debe ser humedecido para no secar las vías respiratorias.
- Dispositivo de entrega o interfaz, para entrega de gases. Existen diferentes tipos, como mascarilla de alto flujo, cánula nasal de alto flujo, mascarilla con reservorio, entre otros. En el presente trabajo utilizaremos cánula de alto flujo, debido a que no restringe ciertas actividades del paciente, como comer o hablar. [28] [30]

2.3.3 Relación FiO2 y flujo en oxigenoterapia

La relación entre la FiO2 y el flujo, medido en L/min, depende del interfaz a emplear. El dispositivo estará dirigido para emplear cánula nasal de alto flujo, con este interfaz, se obtiene una alta FiO2, sin emplear flujo elevado.

Tabla 1.2 Relación entre L/min y FiO2 [31]

Flujo [L / min]	FiO2 [%]
0 (aire ambiente)	21
1	24
2	28
3	32
4	36
5	40
6	44
7	48
8	52
9	56
10	60

2.3.4 Relación entre FiO2 y parámetros fisiológicos SpO2 y PaO2

Debido a que depende de la condición de cada paciente, no existe una fórmula entre la FiO2 y los parámetros fisiológicos como SpO2 y PaO2, sin embargo, podemos indicar que existe una relación directamente proporcional entre la FiO2 y parámetros SpO2 y PaO2, siempre y cuando no se trate de pacientes con enfermedad pulmonar obstructiva crónica, EPOC, ya que, en ellos, la relación podría ser inversamente proporcional.

Podemos también relacionar la saturación de oxígeno con la presión parcial de oxígeno. La relación se muestra en la siguiente gráfica. [32]

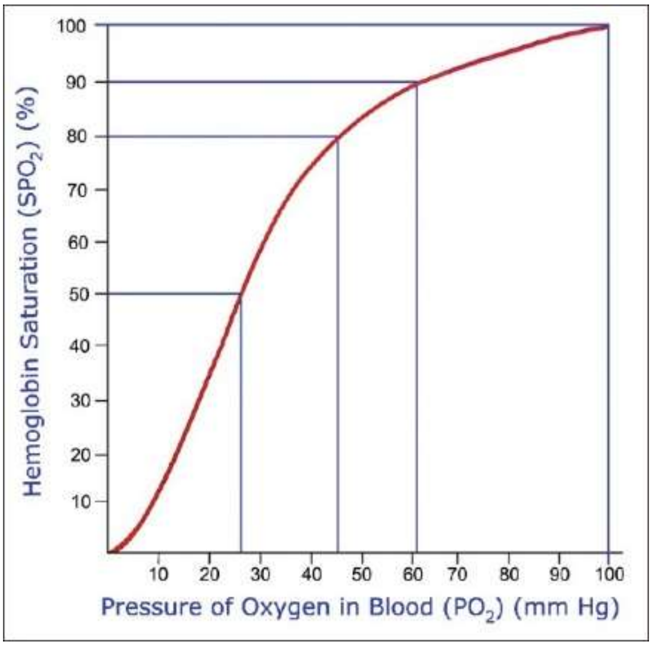


Figura 2.6 Relación entre la presión parcial de oxígeno y la saturación de oxígeno [32]

Dado que los oxímetros de pulso entregan valores de SpO2, requerimos convertir estos valores a PaO2. Con base en las siguientes relaciones, aproximaremos los valores requeridos.

Tabla 2.3 Cálculo de PaO2 desde la SpO2 [32]

SpO2 [%]	**Cálculo de PaO2**	**Rango resultante de PaO2 [mmHg]**
100 - 90	PaO2 decrece en 4 [mmHg] por cada punto de porcentaje de SpO2 reducido	100 - 60
90 - 80	PaO2 decrece en 1.5 [mmHg] por cada punto de porcentaje de SpO2 reducido	60 - 45
Menor a 60	PaO2 es aproximadamente SpO2 / 2	< 40

Tabla 2.2 Relación entre Sp02 y PaO2

SpO2	PaO2	SpO2	PaO2	SpO2	PaO2	SpO2	PaO2	SpO2	PaO2
100	100	80	45	60	30	40	20	20	10
99	96	79	44	59	29,5	39	19,5	19	9,5
98	92	78	43	58	29	38	19	18	9
97	88	77	42	57	28,5	37	18,5	17	8,5
96	84	76	41	56	28	36	18	16	8
95	80	75	40	55	27,5	35	17,5	15	7,5
94	76	74	39,5	54	27	34	17	14	7
93	72	73	39	53	26,5	33	16,5	13	6,5
92	68	72	38	52	26	32	16	12	6
91	64	71	37	51	25,5	31	15,5	11	5,5
90	60	70	36	50	25	30	15	10	5
89	58,5	69	35	49	24,5	29	14,5	9	4,5
88	57	68	34	48	24	28	14	8	4
87	55,5	67	33,5	47	23,5	27	13,5	7	3,5
86	54	66	33	46	23	26	13	6	3
85	52,5	65	32,5	45	22,5	25	12,5	5	2,5
84	51	64	32	44	22	24	12	4	2
83	49,5	63	31,5	43	21,5	23	11,5	3	1,5
82	48	62	31	42	21	22	11	2	1
81	46,5	61	30,5	41	20,5	21	10,5	1	0,5

2.4 Telemonitorización

Este servicio, parte de la telemedicina, será implementado como aporte adicional al presente proyecto. Permitirá el acceso, en tiempo real, a la información de los parámetros fisiológicos del paciente, y también a la FiO2 que está siendo suministrada. Como plataforma se empleará Thingsboard.

El rápido crecimiento en la cantidad de dispositivos de imágenes, monitoreo de la salud, datos biométricos, información clínica, ha generado una gran cantidad de información, y debido a su naturaleza diversa y compleja, la visualización de datos está jugando un papel muy importante.

La visualización de datos es el proceso de convertir datos numéricos, en datos no numéricos, sin afectar la entropía de los datos originales. Es decir, la visualización de datos es primordial en la creación de imágenes, y datos en general.

Hoy en día existen herramientas mediante las cuales el personal de salud puede medir y monitorear de manera remota, los parámetros fisiológicos de los pacientes, como la frecuencia cardíaca o la saturación de oxígeno, entre otros.
La capacidad de visualizar esta información, mejora el acceso a los servicios de salud para personas que se encuentran en zonas rurales, de tercera edad o personas con dificultad para movilizarse. [11]

La plataforma IoT, o internet de las cosas, por sus siglas en inglés, es la responsable de realizar funciones principales, como interconectar dispositivos, recopilar, monitorear, administrar y analizar datos. Existen muchas plataformas comerciales y de código abierto para satisfacer los requerimientos de los desarrolladores e investigadores, y ellos a su vez, cumplir con las necesidades del personal de salud. En particular, Thingsboard, plataforma de código abierto, es considerada una plataforma eficaz y eficiente, con características de recopilación, procesamiento, visualización y gestión de dispositivos de datos. [9]

Existen algunas limitaciones básicas al emplear plataformas de IoT:
Primero, la necesidad de representar datos en tiempo real no ha sido completamente satisfecha. Esto es un requisito muy importante en el seguimiento pacientes, en particular en áreas rurales o pacientes ancianos o discapacitados.
Segundo, actualmente solo brindan funciones de medición, transfieren datos de salud del paciente a través de métodos de comunicación tradicionales.
En tercer lugar, el usuario final, el cual es un personal de salud, no suele ser experto en el uso de tecnologías de información y las comunicaciones, por tanto, es necesario implementar soluciones con interfaces fáciles de usar y que permitan obtener los parámetros fisiológicos del paciente. [33]

..

El sistema propuesto tiene los siguientes requisitos básicos:

- Capacidad de monitorear en tiempo real
- Capacidad de integrarse con múltiples dispositivos de visualización, diferentes sensores de adquisición de parámetros, y plataformas.
- Interfaz amigable.
- Se adapta a diferentes de servicios.

2.4.1 Thingsboard

Es una plataforma de código abierto de IoT que permite un rápido desarrollo, escalado, gestión de proyectos, e integración de distintos dispositivos y sistemas modernos, e incluso sistemas que podrían considerarse obsoletos. [9] [30]
Los protocolos empleados en Thingsboard son: MQTT (Message Queuing Telemetry Transport), CoAP (Protocolo de aplicación restringida) y HTTP (Protocolo de transferencia de hipertexto). Entre sus características están las siguientes:

- Recopilación: Thingsboard admite la recopilación y almacenamiento de datos remotos de forma fiable. Se puede acceder a los datos mediante el uso de sitios web personalizados.
- Visualización: Thingsboard proporciona utilidades para mostrar gráficos en tiempo real, de los datos recopilados.
- Gestión de dispositivos: Thingsboard permite registrar y administrar los dispositivos IoT. Permite monitorear las propiedades del dispositivo del lado del cliente y el aprovisionamiento del lado del servidor.
- Tablero: se utiliza para mostrar datos y controlar dispositivos de forma remota en tiempo real.
- Administrar alertas: Thingsboard presenta una herramienta para crear, administrar y monitorizar alarmas relacionadas con las entidades del sistema.

Thingsboard emplea los siguientes protocolos básicos:

- Protocolo MQTT: Protocolo de mensajería de publicación-suscripción. Por su bajo requerimiento de ancho de banda, es ideal para aplicaciones de IoT. En un sistema que usa el protocolo MQTT, clientes se conectan a un servidor, los cuales se registrarán, en un proceso denominado como "suscripción", y visualizar o agregar información, proceso denominado como "publicación". La información publicada puede ser visualizada por los otros clientes.
- Protocolo CoAP: CoAP, o protocolo de aplicación restringida, es un protocolo de capa de servicio diseñado para su uso en dispositivos de Internet con recursos restringidos, como los nodos de red de sensores

inalámbricos. Está diseñado para conectar dispositivos en la misma red enlazada, o entre dispositivos y nodos comunes en Internet, o entre dispositivos en diferentes redes limitadas que se conectan por Internet. Diseñado para convertirse fácilmente a HTTP con el fin de integrarse con aplicaciones web y cumplir requisitos específicos, como compatibilidad con multidifusión, una sobrecarga muy baja y simplicidad. Por lo tanto, se puede implementar fácilmente en dispositivos con poca capacidad de memoria y bajo consumo, especialmente aplicaciones en IoT, e incluso con sensores inalámbricos. [33]

2.4.2 Docker

Docker es un proyecto de código abierto que automatiza la implementación de aplicaciones dentro de los contenedores de Linux y brinda la capacidad de empaquetar una aplicación con sus dependencias de tiempo de ejecución en un contenedor. Proporciona una herramienta de línea de comandos para la gestión del ciclo de vida de contenedores basados en imágenes. Los contenedores de Linux permiten una implementación rápida de aplicaciones, pruebas, mantenimiento y solución de problemas más simples, al tiempo que mejoran la seguridad. El uso de Docker permite aumentar la eficiencia, facilitando su uso y habilitando un entorno de desarrollo más ágil, y optimizar recursos. [34]

Docker trabaja con los siguientes componentes fundamentales:

- Contenedor: Un entorno limitado de aplicaciones. Cada contenedor se basa en una imagen que contiene los datos de configuración necesarios. Cuando se lanza un contenedor desde una imagen, se agrega una capa de escritura encima de esta imagen. Cada vez que confirma un contenedor, se agrega una nueva capa de imagen para almacenar sus cambios.
- Imagen: Una captura de la configuración de los contenedores. La imagen es una capa de solo lectura que nunca se modifica, todos los cambios se realizan en la capa de escritura superior y solo se pueden guardar creando una nueva imagen.
- Imagen de plataforma: Las imágenes de la plataforma definen el entorno de tiempo de ejecución, los paquetes y las utilidades necesarias para que

se ejecute la aplicación en contenedores. La imagen de la plataforma es de solo lectura.

- Registro: Un repositorio de imágenes. Los registros son repositorios públicos o privados que contienen imágenes disponibles para descargar. Algunos registros permiten que los usuarios carguen imágenes para ponerlas a disposición de otros.
- Dockerfile: Un archivo de configuración con instrucciones de compilación para las imágenes de Docker. Proporcionan una forma de automatizar, reutilizar y compartir procedimientos de compilación.

Docker trae una API, interfaz de programación de aplicaciones, para la gestión de contenedores, un formato de imagen y la posibilidad de usar un registro remoto para compartir contenedores. Este esquema beneficia tanto a los desarrolladores como a los administradores de sistemas con ventajas como:

- Implementación rápida de aplicaciones: los contenedores incluyen los requisitos mínimos de tiempo de ejecución de la aplicación, lo que reduce su tamaño y permite que se implementen rápidamente.
- Portabilidad entre máquinas: una aplicación y todas sus dependencias se pueden agrupar en un solo contenedor que es independiente de la versión host del kernel de Linux, la distribución de la plataforma o el modelo de implementación. Este contenedor puede transferirse a otra máquina que ejecute Docker y ejecutarse allí sin problemas de compatibilidad.
- Control de versiones y reutilización de componentes: puede realizar un seguimiento de las versiones sucesivas de un contenedor, inspeccionar las diferencias o retroceder a versiones anteriores. Los contenedores reutilizan componentes de capas anteriores, lo que los hace notablemente livianos.
- Compartir: puede usar un repositorio remoto para compartir su contenedor con otros.
- Huella ligera y gastos generales mínimos: las imágenes de Docker suelen ser muy pequeñas, lo que facilita la entrega rápida y reduce el tiempo para implementar nuevos contenedores de aplicaciones.
- Mantenimiento simplificado: Docker reduce el esfuerzo y el riesgo de problemas con las dependencias de la aplicación.

CAPÍTULO 3

3. DISEÑO Y COMPONENTES DEL SISTEMA

El sistema tendrá el siguiente diagrama de bloques.

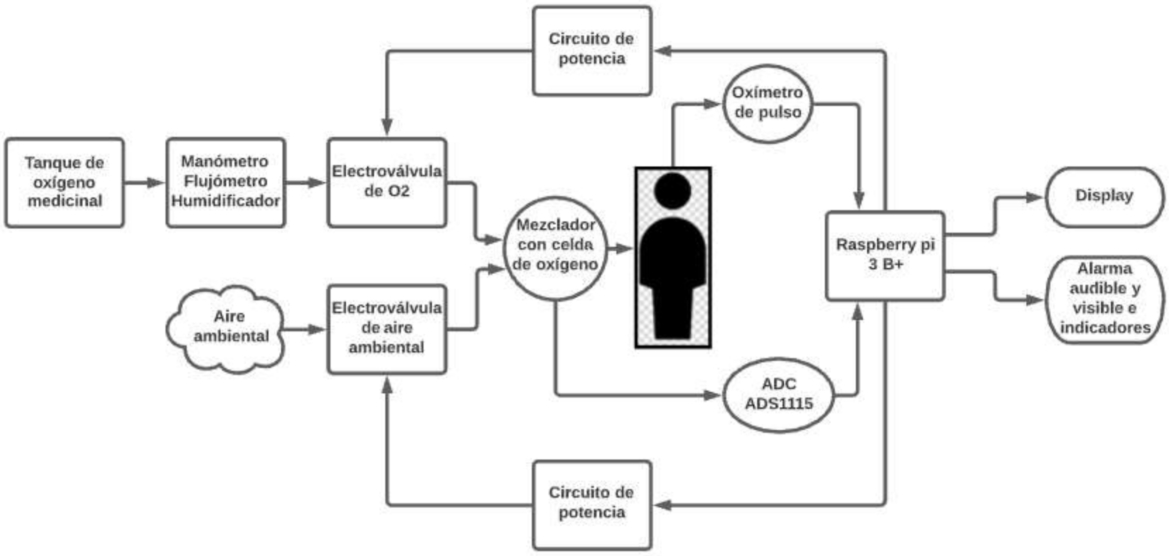


Figura 3.1 Diagrama de bloques del sistema

3.1 Sistema electrónico

El sistema electrónico propuesto estará compuesto por dos etapas principales, control y potencia. La etapa de control estará compuesta por una Raspberry Pi 3 B+, sensor de oximetría de pulso, y adicionalmente se agregará un sensor o celda de oxígeno. La señal entregada por la celda de oxígeno debe ser digitalizada.

La etapa de potencia recibirá señales de la etapa de control para ajustar la apertura de las válvulas.

3.1.1 Sensores

3.1.1.1 Sensor de oximetría de pulso MAX 30100

La oximetría de pulso se utiliza para medir la saturación de oxígeno o SpO2. Adicionalmente se puede medir la frecuencia o ritmo cardiaco. El MAX30100 integra oximetría de pulso y monitorización de ritmo cardíaco y su comunicación es I2C. Posee dos LEDs y un fotodetector, con procesamiento de señales analógicas de bajo ruido, para detectar las señales antes descritas.

Opera de 1.8 [V] - 3.3 [V] y mediante software se puede configurar en modo de espera, optimizando recursos y permitiendo su conexión en todo momento.

Debido al bajo requerimiento de energía, este sensor puede emplearse en aplicaciones de dispositivos portátiles, dispositivos de monitorización de parámetros fisiológicos, y dispositivos asistentes de actividad física, optimizando la duración de la batería. [35]

Características adicionales:

- Dimensiones: 5,6 [mm] x 2,8 [mm] x 1,2 [mm]
- Tasa de muestreo programable y corriente de LED para ahorro de energía
- Corriente de modo de espera ultra baja, 0,7 [µA]
- La alta SNR proporciona una robusta resiliencia contra artefactos por movimiento
- Cancelación de luz ambiental integrada
- Capacidad de alta frecuencia de muestreo
- Capacidad de rápida salida de datos

Figura 3.2 Oxímetro de pulso MAX30100 [35]

Las especificaciones del fabricante del módulo MAX30100 se muestran en el anexo A.

3.1.1.2 Sensor o celda de Oxígeno

El dispositivo a utilizar corresponde al modelo PSR-11-75-ke7. Este sensor cumplirá la función de verificación de la FiO2 que recibirá el paciente. Si el valor medido por el sensor difiere del valor calculado, se generará una alarma y un mensaje, el cual indicará que el equipo y su funcionamiento requieren ser revisados. La salida de este dispositivo es analógica, razón por la cuál, se requerirá ser acondicionada. En el subcapítulo denominado como “Acondicionamiento de señales” se profundizará sobre la salida de este dispositivo.

Figura 3.3 Celda de oxígeno PSR-11-75-ke7 [36]

Las especificaciones técnicas se presentan en la siguiente tabla:

Tabla 3.1 Especificaciones técnicas del sensor de oxígeno PSR-75-ke7 [36]

Rango de medición de concentración de oxígeno	0 - 100%
Salida en aire ambiental	10 – 14 [mV]
Interfaz eléctrica	Entrada DC
Exactitud	Cumple requerimientos ISO 80601-2-55
Reproducibilidad	Menor al 1%, con presión y temperatura constante
Exactitud	Dentro del 2%
Tiempo de respuesta	Menos de 13 segundos
Temperatura de operación	0 - 45 [°C]
Humedad de operación	0 - 99 % No condensado
Variación de salida a largo plazo	Menor al 1% por mes, menor al 15% durante su vida útil
Temperatura de almacenamiento	0 [°C] - 25 [°C]
Tiempo de calentamiento	Menor a 30 minutos, luego de su instalación
Peso	Aproximadamente 28 [g]

Las especificaciones completas de esta celda de oxígeno se muestran en el anexo B.

3.1.2 Acondicionamiento de señales

La etapa de acondicionamiento permite captar las señales de los sensores, tratarla, y, finalmente, transmitirla. Tratar la señal incorpora diferentes posibilidades, tales como: amplificar, filtrar, muestrear, conversión analógico digital, conversión digital analógico, multiplexar, etc.

El oxímetro de pulso o sensor de SpO2 MAX30100, tiene comunicación I2C, que puede ser conectada directamente a los pines correspondientes a SDA y SCL del módulo de control

El sensor o celda de oxígeno tiene salida analógica, la cual está en el orden de los milivoltios, por lo tanto, esta señal debe ser convertida a señal digital.

Para digitalizar el voltaje entregado por la celda de oxígeno, se utilizará el módulo integrado de 16 bits de resolución, ADS1115. Este módulo convertidor analógico digital tiene oscilador integrado. [37]

Figura 3.4 Convertidor analógico digital ADS1115 [37]

A continuación, una tabla con las especificaciones generales de este convertidor.

Tabla 3.2 Especificaciones técnicas del convertidor analógico digital. [37]

Presentación ultra pequeña	2 [mm] x 1.5 [mm] x 0,4 [mm]
Amplio rango de alimentación	2 [V] - 5 [V]
Bajo consumo de corriente	Modo continuo: 150 [uA] Modo mono disparo: Auto-apagado
Tasa de datos programable	8 SPS A 860 SPS
Baja desviación de voltaje de referencia	
Oscilador interno	
PGA Interna	
Interfaz I2C	
Comparador programable	
Cuatro pines de entrada	

Las especificaciones del fabricante de este módulo se mostrarán en el anexo C.

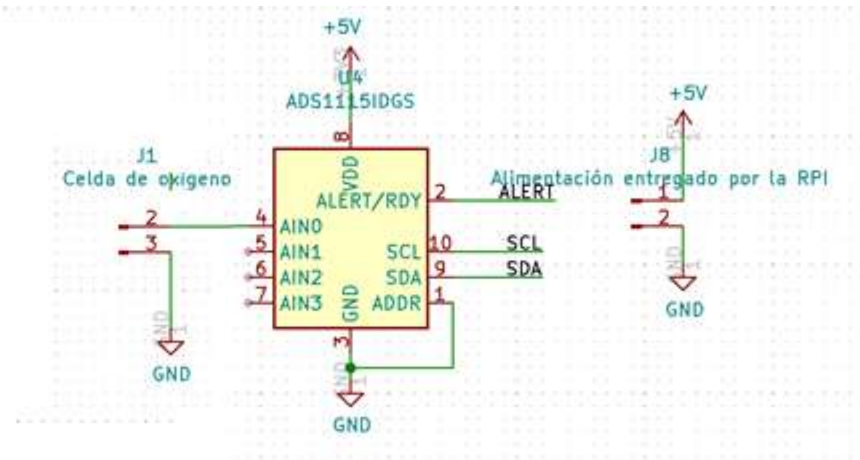


Figura 3.5 Esquemático del circuito celda de oxígeno y convertidor ADS1115

3.1.3 Circuito de Potencia

La Raspberry pi 3B+ se encargará, entre otras funciones, de controlar la FiO2 que recibirá el paciente. Este control se realizará regulando el ciclo de trabajo de cada una de las válvulas de aire y oxígeno. Dado que los actuadores requieren 24 [V] para su funcionamiento, es imposible que el módulo de control las alimente directamente. Un circuito de potencia es necesario. Para este circuito se utilizan transistores BJT, TIP 41C. Considerando que la amplitud de las señales PWM 1 y PWM 2 entregadas por la Raspberry pi 3 B+, es 3.3 [V], se considera que el valor de la resistencia de base adecuado es 330 [Ω]. En paralelo a cada una de las válvulas, se utiliza un diodo de protección, el cuál protegerá al transistor de corrientes inversas al momento de la conmutación.

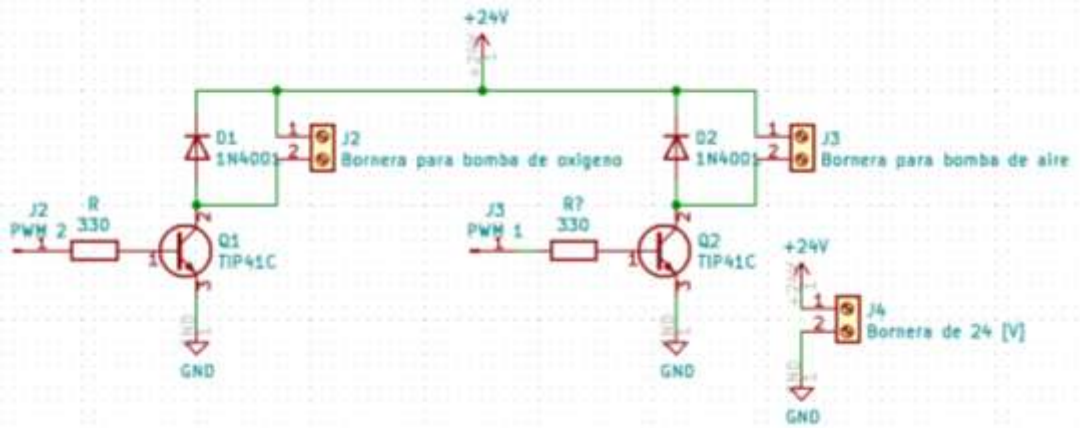


Figura 3.6 Esquemático del circuito de potencia

3.1.4 Electroválvulas

Para el sistema propuesto, se utilizan dos válvulas de 300 [GPH], es decir, cada una de ellas es capaz de suministrar la totalidad del volumen tidal requerido por un paciente adulto.

Estas válvulas serán controladas por la raspberry Pi 3 B+ mediante PWM, a través de un circuito del circuito de potencia. El voltaje DC requerido es 24 [V]. La potencia nominal es 18 [W].

Figura 3.7 Electroválvula de 300 [GPH]

3.1.5 Mezclador

El Mezclador es un dispositivo que tendrá la función de recibir el aire y oxígeno medicinal entregado por las válvulas. En él se colocará la celda de oxígeno, permitiendo conocer la FiO2 real que recibirá el paciente. Adicionalmente, desde este dispositivo saldrá la conexión para colocar el circuito paciente. Dado que el volumen tidal de un paciente promedio es de 0.5 [L], se consideró que es adecuado duplicarlo, diseñando un mezclador de 1 [L].

Figura 3.8 Mezclador desarrollado con acrílico

3.1.6 Tarjeta de control y algoritmo

3.1.6.1 Raspberry pi 3B+

La tarjeta de control utilizada fue una Raspberry Pi 3B+. Este módulo cumple todos los requerimientos de entradas y salidas que son necesarios para la conexión de los diferentes sensores y actuadores utilizados en este sistema.

La Raspberry Pi 3B+ tiene procesador de cuatro núcleos de 64 bits, que funciona a 1 [GHz], conexión inalámbrica, conexión ethernet, cuatro puertos USB, comunicación I2C y UART. [38]

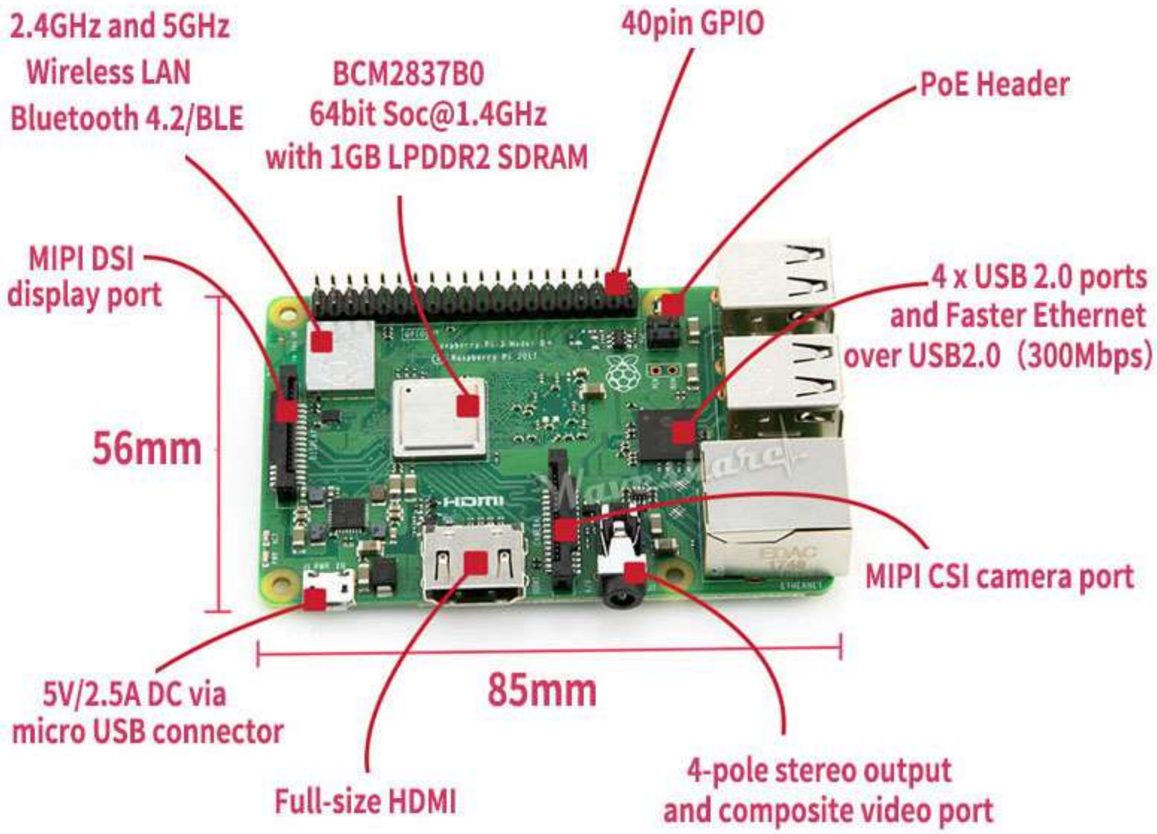


Figura 3.9 Configuración de entradas y salidas de la Raspberry pi 3B+. **[39]**

Tabla 3.3 Especificaciones técnicas de la Raspberry Pi 3B+. [38]

Procesador	Broadcom BCM2837B0, Cortex A53 de 64 Bits - 1,4 [GHz]
Memora	1 [GB] LPDDR2 SDRAM
Conectividad	2,4 [GHz] y 5 [GHz] IEEE 802.11 b/g/n/ac lan inalámbrico
	Bluetooth 4.2, BLE
	Gigabit ethernet - 300 Mbps
	4 puertos USB 2.0
Acceso	GPIO extendido de 40 pines
Video y sonido	1 puerto HDMI
	Puerto de display
	Puerto de cámara
	Salida de audio y video
Soporta tarjeta SD	Para almacenamiento de datos y carga del sistema operativo
Entrada	5 [V] / 2.5 [A] por conector micro USB
	5 [V] por el conector GPIO
Temperatura de operación	0 [°C] - 50 [°C]

Las especificaciones dadas por el fabricante se mostrarán en el anexo D.

3.1.6.2 Consideraciones generales para la programación

El oxímetro de pulso se conecta mediante I2C a la Raspberry Pi 3B+; la celda de oxígeno se conecta directamente al convertidor analógico digital, el cual se encargará de transmitir, también mediante comunicación I2C, los valores digitalizados a la Raspberry Pi 3B+.

Adicionalmente, el módulo se encargará de controlar las válvulas. Este control se realizará mediante modulación por ancho de pulso, PWM, por sus siglas en inglés. La frecuencia para las dos válvulas será fija, pero el ciclo de trabajo variará en función de los parámetros fisiológicos, SpO2 y ritmo cardiaco, entregados por el oxímetro de pulso. La relación entre estas variables se dará según la siguiente tabla:

Tabla 3.4 Relación entre SpO2, FiO2 y frecuencia cardiaca

SpO2 [%]	FiO2 Inicial	FiO2 luego de 10 minutos si FC normal y estable	FiO2 luego de 10 minutos si FC alta y no hay mejoría
88 - 92	28 % - 2 L/min	28 % - 2 L/min	32% - 3 L/min
85 - 88	32 % - 3 L/min	32 % - 3 L/min	36% - 4 L/min
83 - 85	40 % - 5 L/min	40 % - 5 L/min	52 % - 8 L/min
80 - 83	60 % - 10L/min	60 % - 10L/min	Ventilación mecánica

El aire tiene 21% de oxígeno, por lo tanto, se debe considerar ese porcentaje al momento de calcular la apertura o ciclo de trabajo de las válvulas. La relación es la siguiente:

$$B = (\text{FiO2} - 0.21)/0.79 \qquad (3.1)$$

Dónde: B corresponde a la proporción de oxígeno en decimal; $0 < B < 1$.
FiO2: Fracción de inspiración de oxígeno en decimal

Para programar la Raspberry Pi 3B+ se hace uso del programa Pycharm, la cual, mediante Putty y Python ya ha sido preconfigurada.

El algoritmo se divide en tres partes principales. La primera de ellas se encargará de la comunicación con el oxímetro de pulso mediante I2C. La segunda etapa corresponde a la programación para comunicarse, mediante I2C, con el convertidor analógico digital ADS1115. La tercera etapa, se encargará del control de las válvulas.

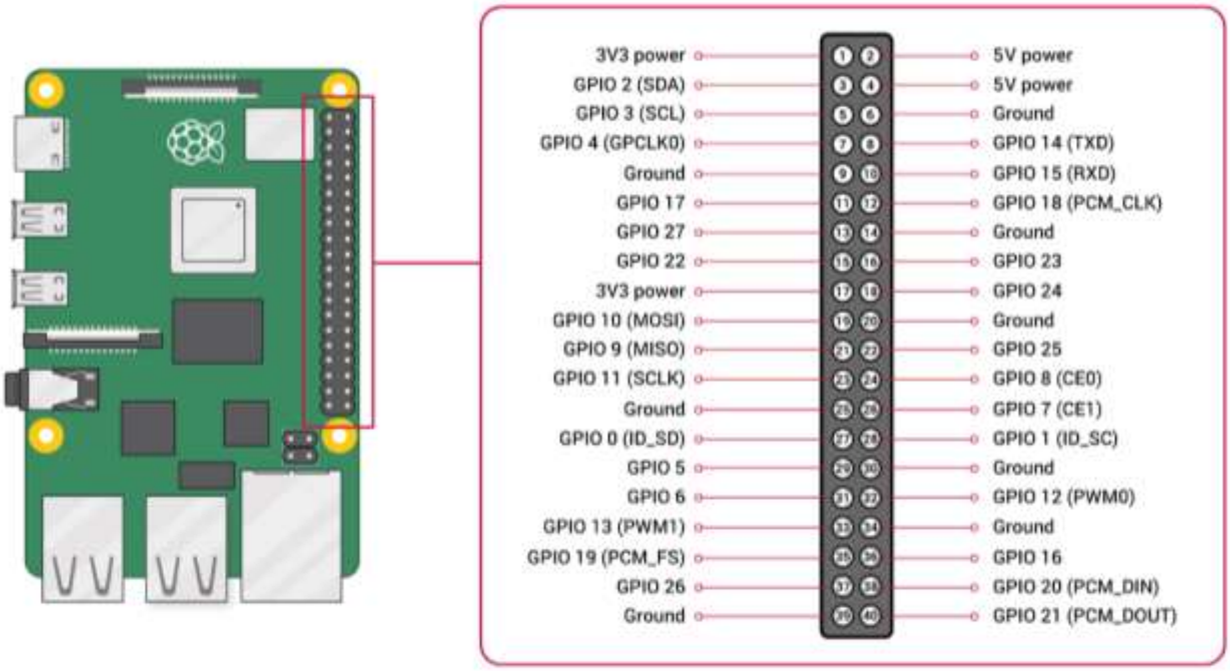


Figura 3.10 Configuración de pines de entrada y salida. [39]

3.1.6.3 Consideraciones para el algoritmo de la celda de oxígeno

Se crea una variable denominada OOM, la cual permitirá el registro del cálculo de FiO2 basado en el voltaje entregado por la celda de oxígeno. Este voltaje, según las especificaciones del fabricante, se encuentra entre 10 [mV] y 14 [mV], y experimentalmente, se obtuvo 12 [mV] en aire ambiental, es decir, FiO2 21%. Por un modelo matemático simple, se puede obtener la relación entre la FiO2 y el voltaje.

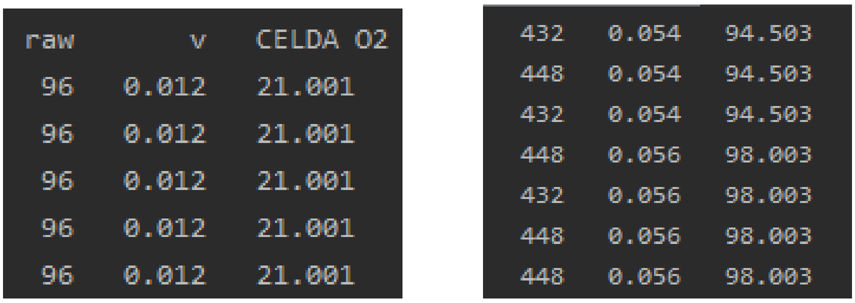
$$FiO2 = 20.9 * (Vi/Vo) \qquad (3.2)$$

Dónde:

FiO2: fracción de inspiración de oxígeno

Vi: Voltaje entregado en tiempo real por la celda de oxígeno

Vo: Voltaje de referencia en aire ambiental 12 [mV]



```
raw        v    CELDA O2
 96    0.012    21.001
 96    0.012    21.001
 96    0.012    21.001
 96    0.012    21.001
 96    0.012    21.001
```

```
432   0.054   94.503
448   0.054   94.503
432   0.054   94.503
448   0.056   98.003
432   0.056   98.003
448   0.056   98.003
448   0.056   98.003
```

Figura 3.11 A la izquierda, imagen de la ejecución del programa con la celda de oxígeno en aire ambiental y, a la derecha, con casi 100% de oxígeno medicinal.

3.1.6.4 Consideraciones para el algoritmo del control de válvulas

Experimentalmente se obtiene que la frecuencia adecuada es 0.25 [Hz] y el ciclo de trabajo sumado, de la válvula de aire y la válvula de oxígeno, debe sumar 50%. Debido a esto, la ecuación final para el cálculo del ciclo de trabajo de la válvula de oxígeno es la siguiente:

$$O2= 50*(FiO2-0.21) / 0.79 \qquad (3.3)$$

Esto último se debe a que el caudal, o volumen por minuto de cada válvula, es más del doble del requerido por un paciente promedio.

La variable FiO2, recibirá, en tiempo real, la fracción de inspiración de oxígeno requerida para el paciente, según la saturación de oxígeno instantánea.

3.1.6.5 Consideraciones sobre el algoritmo de comunicación con la plataforma Thingsboard

La configuración “THINGSBOARD_HOST” corresponde a la dirección IP de la Raspberry Pi 3B+, y el token de acceso debe coincidir en la plataforma Thingsboard.

3.1.6.6 Algoritmo completo de dispositivos y comportamiento del sistema

El algoritmo de cada uno de los componentes y etapas, así como el algoritmo completo del sistema de control automático de oxigenoterapia no invasiva de alto flujo, se muestra en el Anexo E.

3.1.7 Diseño de PCB

Empleando herramientas del programa KiCad, se realiza el diseño de la placa para el circuito del sistema. El esquemático del sistema se muestra en el Anexo F.

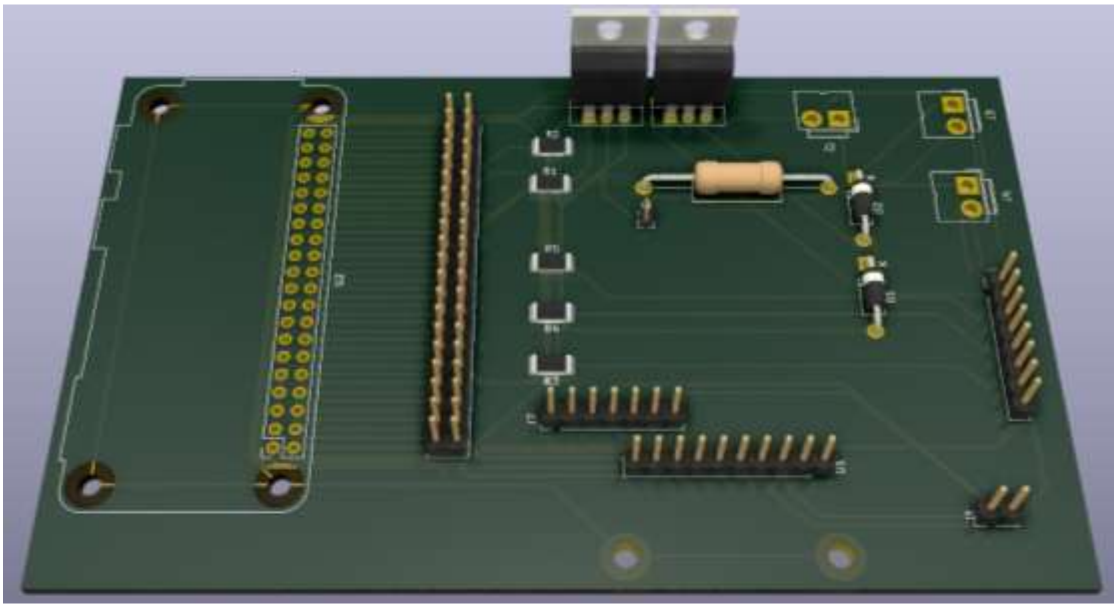


Figura 3.12 Diseño en 3D de placa para circuito impreso del sistema

3.1.8 Pantalla

La Raspberry pi 3B+ posee un puerto HDMI que, por su facilidad, será empleado para la visualización de la información requerida.

Dado que el sistema debe mostrar variables SpO2, ritmo cardiaco, fracción de inspiración FiO2, presión parcial de oxígeno PaO2, cualquier monitor es factible. Esta información podrá ser visualizada en todo momento.

3.2 Sistema neumático

3.2.1 Circuito paciente

El circuito paciente se compone de dos partes principales, el circuito propiamente dicho y la cánula nasal.

La cánula nasal utilizada permite suministrar hasta 100% de FiO2. La cánula es desechable y debe ser cambiada con cada paciente. La longitud de la cánula es 380 [mm]. El conector es de tipo 15F/22M, cumpliendo la normativa ISO 5356-1

Figura 3.13 Cánula nasal de alto flujo

El circuito requerido y utilizado, permite suministrar alto flujo, y, adicionalmente tiene sistema de calentamiento. El calefactor de la mezcla funciona con 12 [V] y es controlado por la misma señal PWM que controla la válvula de O2.

Figura 3.14 Circuito paciente de alto flujo, con sistema de calentamiento

3.2.2 Suministro de oxígeno y aire

El suministro de oxígeno puede ser un tanque de oxígeno medicinal, en diferentes capacidades, según sea el requerimiento del usuario. El suministro puede ser también una toma centralizada de O2. Según la fuente a utilizar, puede variar el tipo de conexión. Para el presente proyecto se utiliza tanque de oxígeno medicinal.

Figura 3.15 Tanque de oxígeno con flujómetro

3.2.3 Humidificador

La función del humidificador es proporcionar humedad a la mezcla de gases suministrada por el dispositivo. Cuando la FiO2 es alta, la mezcla carece de suficiente humedad y su temperatura es relativamente baja, produciendo irritación y resequedad del tracto oro faríngeo, por lo tanto, la mezcla debe ser humidificada y calentada. La humedad será generada mediante la evaporación de agua destilada y esterilizada.

Cuando el flujo no supera los 5 [L/min] o 40% de FiO2, no es necesario el uso del humidificador, ya que el otro 60% corresponde a aire ambiental.

Figura 3.16 Humidificador pasivo [40]

3.2.4 Flujómetro

El flujómetro, también llamado caudalímetro, permite dosificar el volumen de oxígeno por minuto que será suministrado al paciente. El control es manual y en condiciones normales, el caudal se fija previamente, según el requerimiento.

Para este caso, no será necesario un ajuste manual adicional al máximo requerido o el máximo que puede entregar, es decir 15 [L/min], debido a que la válvula de oxígeno se encargará de dosificar los L/min según la condición del paciente.

Figura 3.17 Flujómetro

3.2.5 Envolturas de componentes del sistema e interfaz del usuario

3.2.5.1 Sensor óptico

El sensor óptico corresponde únicamente al oxímetro de pulso mencionado previamente. Cuando se enciende el módulo, el sensor MAX30100 se enciende automáticamente y no requiere ajuste alguno.

Este dispositivo tiene recubrimiento de plástico y se coloca en uno de los dedos del paciente. Los datos adquiridos son transmitidos en tiempo real a la raspberry pi 3 B+.

Este es el único dispositivo que se encargará de la adquisición de datos del paciente, ritmo cardíaco y saturación de oxígeno.

3.2.5.2 Equipo general

El equipo es construido con láminas de acrílico de 3 [mm] de espesor. Cada una de ellas es fabricada de acuerdo a las entradas y salidas requeridas.

La lámina anterior, requiere el espacio de conexión para el cable del oxímetro de pulso, el espacio para el actuador de la alarma audible y para los leds de alarma visible. Esta lámina también tiene el espacio necesario para la conexión del circuito paciente y su calefactor.

La lámina posterior requiere espacio para conexión del cable de poder, interruptor de encendido, para la conexión del cable HDMI y también para las entradas de aire ambiente y oxígeno medicinal.

3.2.5.3 Manual descriptivo de componentes complementarias

En el dispositivo se emplean otros componentes, como la toma de aire, la cual no requiere ser conectada a ningún dispositivo, ya que la válvula lo tomará del medio ambiente; también la toma de oxígeno, que se debe conectar, mediante una manguera, a la salida del humidificador o directamente a la salida del flujómetro.

Ya que el equipo funciona con 115 [V] de corriente alterna, el cable de poder debe conectarse en la parte posterior y el interruptor debe colocarse en posición de encendido, para luego proceder con el encendido del oxímetro de pulso.

El circuito paciente se conecta a la salida de gases del equipo. En el circuito paciente, se debe colocar la cánula de alto flujo, la cual es el interfaz adecuado para el dispositivo diseñado. La cánula y el oxímetro de pulso serán los únicos dispositivos que tendrán contacto directo con el paciente.

El dispositivo cuenta con leds, los cuales serán indicadores según la evolución del paciente.

Tabla 3.5 Indicadores LED del sistema

LED verde	LED Rojo	LED azul	SIGNIFICADO
Encendido	Apagado	Apagado	Oxímetro de pulso conectado y monitorizando al paciente.
Apagado	Encendido	Apagado	Oxímetro de pulso desconectado y/o no detecta señales fisiológicas del paciente.
Encendido	Apagado	Encendido	Paciente no mejora luego de 10 minutos de recibir oxigenoterapia de alto flujo y presenta taquicardia, es decir, más de 110 latidos por minuto. Se aumentará FiO2 hasta un máximo de 60%.
Encendido	Encendido	Encendido	Paciente requiere ventilación mecánica

3.2.6 Análisis de costos

A continuación, se presenta de forma detallada, los componentes electrónicos, insumos adquiridos, junto a sus valores reales. Presupuesto del proyecto ajustado a los valores del mercado tecnológico del año 2021.

Tabla 3.63 Costo de los componentes del sistema

Componente	Cantidad	Costo unitario en dólares de EUA	Total
Oxímetro de pulso	1	$15,00	$15,00
Celda de oxígeno	1	$150.00	$150,00
Válvulas	2	$23,00	$46,00
ADS1115	1	$10,00	$10,00
Raspberry pi 3 B+	1	$62,99	$62,99
Trabajo en acrílico	1	$42,00	$42,00
Paquete de componentes electrónicos	1	$13,32	$13,32
Cable de poder	1	$2,00	$2,00
Placa PCB	1	$12,00	$12,00
Alquiler de tanque de O2, flujómetro y humidificador	1	$80,00	$80,00
Abrazaderas	6	$0,50	$3,00
Mangueras	2	$2,25	$4,50
Circuito paciente con calefactory (insumo)	1	$40.00	$40.00
			$480,81

CAPÍTULO 4

4. RESULTADOS EXPERIMENTALES Y EVALUACIÓN DEL SISTEMA

4.1 Configuración del sistema

La configuración realizada, permite que, al encender el equipo, automáticamente ingrese a modo de espera. Este modo finaliza de manera automática al recibir una señal fisiológica del oxímetro de pulso. Adicionalmente al ser encendido, durante 5 segundos mantiene encendidas las alarmas audibles y visibles, como verificación.

Durante todo el proceso de adquisición, el dispositivo realizará el procedimiento correspondiente, anteriormente descrito, incluso si el paciente se encuentra con SpO2 mayor o igual a 92%, es decir, se mantendrá operando, aunque la FiO2 entregada sea 21%. Esto se debe a que la SpO2 de un paciente que ha alcanzado valores normales, puede volver a descender.

El sistema puede ser detenido al apagarlo.

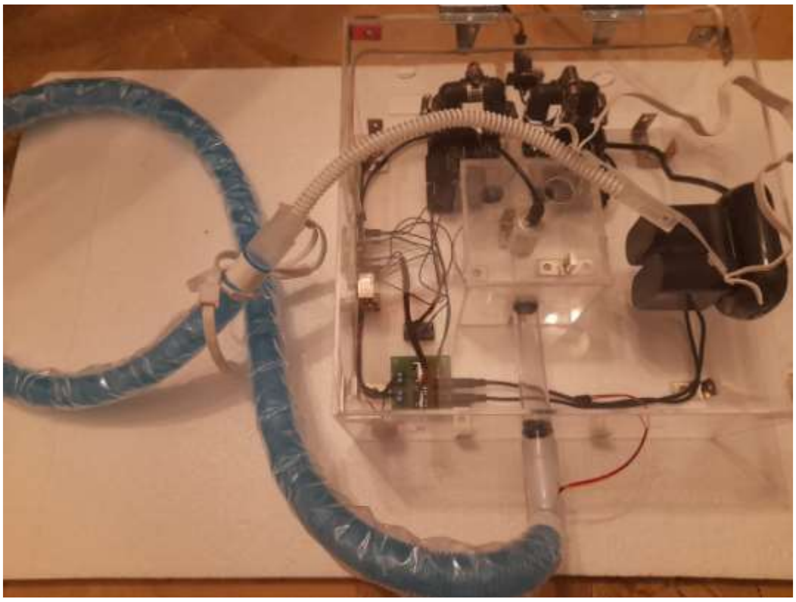


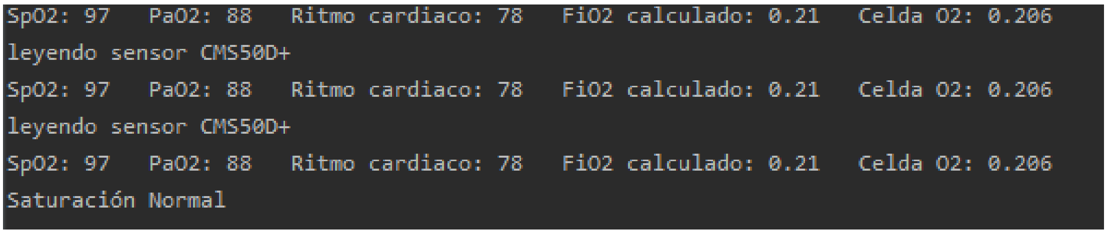
Figura 4.1 Hardware del sistema y circuito paciente

4.2 Pruebas experimentales

Considerando que las pruebas para el presente proyecto se realizan en un sujeto de control sano, no se puede determinar su correcto funcionamiento con los parámetros fisiológicos adquiridos por el oxímetro de pulso, ya que, para este caso, la FiO2 requerida es 21%, aire ambiental, por lo tanto, las pruebas experimentales se realizaron modificando la variable SpO2 por valores numéricos que podrán variar entre 80% y 98%. Para todos los casos, se asumirá que el paciente mantiene su estado inicial.

Se agrega una línea de código para fijar los valores de SpO2 y ritmo cardiaco.
`raw = [0, 0, 0, "RC", 0, "RC", "SpO2", 0, 0].`

1. Inicialmente, se configura el ritmo cardiaco en 78, correspondiente a la letra “N”, y la saturación de oxígeno en 97%, correspondiente a la letra “a”. Para este caso, el led verde, se mantiene encendido, mientras que el rojo y el azul, al igual que la alarma audible, se encuentran apagados.

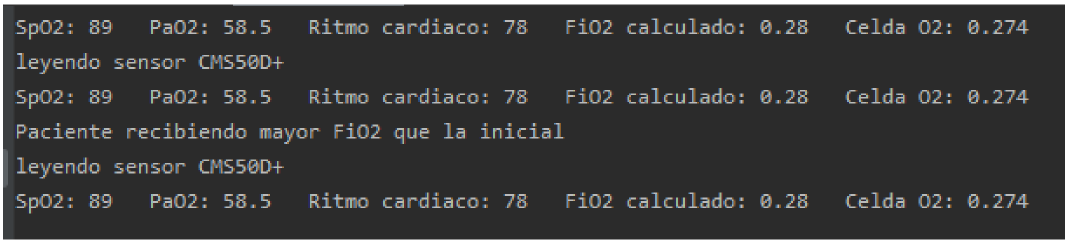


```
SpO2: 97   PaO2: 88   Ritmo cardiaco: 78   FiO2 calculado: 0.21   Celda O2: 0.206
leyendo sensor CMS50D+
SpO2: 97   PaO2: 88   Ritmo cardiaco: 78   FiO2 calculado: 0.21   Celda O2: 0.206
leyendo sensor CMS50D+
SpO2: 97   PaO2: 88   Ritmo cardiaco: 78   FiO2 calculado: 0.21   Celda O2: 0.206
Saturación Normal
```

Figura 4.2 Ejecución del programa con SpO2 98%, se requiere FiO2 21%

Figura 4.3 Led verde encendido. Este led se enciende cuando los parámetros fisiológicos del paciente están dentro del rango normal

2. Se configura la saturación de oxígeno en 89%, correspondiente a la letra “Y”. El ritmo cardiaco se mantiene fijo. Para este caso, el led verde se enciende al inicio, pero al no detectar mejoría, el dispositivo ajusta la FiO2 a 32 % y muestra el mensaje “Paciente recibiendo mayor FiO2 que la inicial” y encendiendo también el led de color azul. El led rojo y la alarma audible se mantienen apagados.

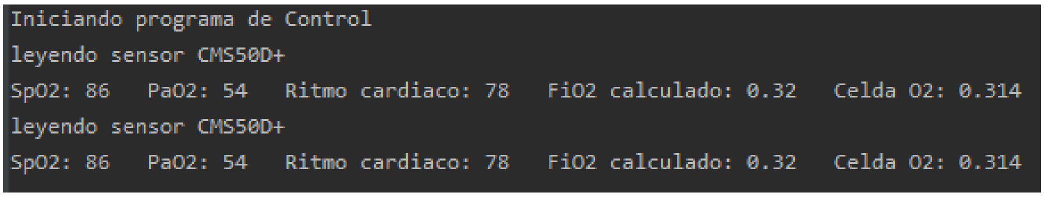


```
SpO2: 89   PaO2: 58.5   Ritmo cardiaco: 78   FiO2 calculado: 0.28   Celda O2: 0.274
leyendo sensor CMS50D+
SpO2: 89   PaO2: 58.5   Ritmo cardiaco: 78   FiO2 calculado: 0.28   Celda O2: 0.274
Paciente recibiendo mayor FiO2 que la inicial
leyendo sensor CMS50D+
SpO2: 89   PaO2: 58.5   Ritmo cardiaco: 78   FiO2 calculado: 0.28   Celda O2: 0.274
```

Figura 4.4 Ejecución del programa con SpO2 89%, inicialmente se requiere FiO2 28%

Figura 4.5 Leds verde y azul encendidos. Indica que el paciente está por debajo del rango normal

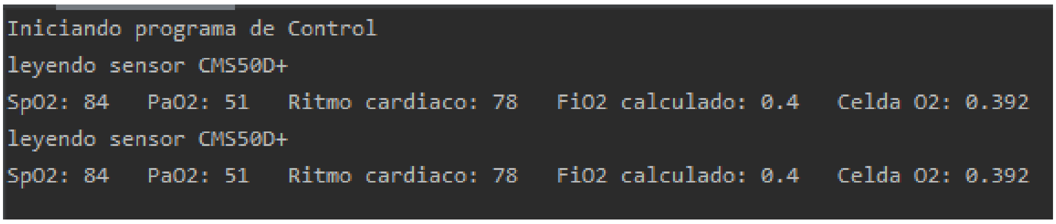
3. Se configura la saturación de oxígeno en 86%, correspondiente a la letra “V”. El ritmo cardiaco se mantiene fijo. El ritmo cardiaco se mantiene fijo. Para este caso, el led verde se enciende al inicio, pero al no detectar mejoría, el dispositivo ajusta la FiO2 a 36 % y muestra el mensaje “Paciente recibiendo mayor FiO2 que la inicial” y encendiendo también el led de color azul. El led rojo y la alarma audible se mantienen apagados.



```
Iniciando programa de Control
leyendo sensor CMS50D+
SpO2: 86   PaO2: 54   Ritmo cardiaco: 78   FiO2 calculado: 0.32   Celda O2: 0.314
leyendo sensor CMS50D+
SpO2: 86   PaO2: 54   Ritmo cardiaco: 78   FiO2 calculado: 0.32   Celda O2: 0.314
```

Figura 4.6 Ejecución del programa con SpO2 86%, inicialmente se requiere FiO2 32%

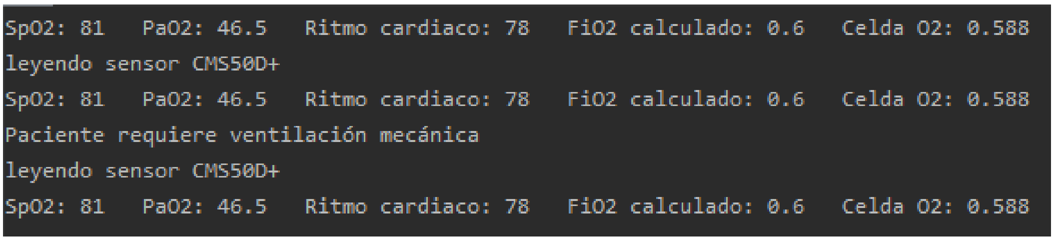
4. Se configura la saturación de oxígeno en 84%, correspondiente a la letra “T”. El ritmo cardiaco se mantiene fijo. El ritmo cardiaco se mantiene fijo. Para este caso, el led verde se enciende al inicio, pero al no detectar mejoría, el dispositivo ajusta la FiO2 a 52 % y muestra el mensaje “Paciente recibiendo mayor FiO2 que la inicial” y encendiendo también el led de color azul. El led rojo y la alarma audible se mantienen apagados.



```
Iniciando programa de Control
leyendo sensor CMS50D+
SpO2: 84   PaO2: 51   Ritmo cardiaco: 78   FiO2 calculado: 0.4   Celda O2: 0.392
leyendo sensor CMS50D+
SpO2: 84   PaO2: 51   Ritmo cardiaco: 78   FiO2 calculado: 0.4   Celda O2: 0.392
```

Figura 4.7 Ejecución del programa con SpO2 84%, inicialmente se requiere FiO2 40%

5. Se configura la saturación de oxígeno en 81%, correspondiente a la letra “Q”. El ritmo cardiaco se mantiene fijo. El ritmo cardiaco se mantiene fijo. Para este caso, el led verde se enciende al inicio, pero al no detectar mejoría, el dispositivo ajusta la FiO2 a 60 % y muestra el mensaje “Paciente requiere ventilación mecánica” y encendiendo también el led de color azul. El led rojo y la alarma audible se encienden también.



```
SpO2: 81   PaO2: 46.5   Ritmo cardiaco: 78   FiO2 calculado: 0.6   Celda O2: 0.588
leyendo sensor CMS50D+
SpO2: 81   PaO2: 46.5   Ritmo cardiaco: 78   FiO2 calculado: 0.6   Celda O2: 0.588
Paciente requiere ventilación mecánica
leyendo sensor CMS50D+
SpO2: 81   PaO2: 46.5   Ritmo cardiaco: 78   FiO2 calculado: 0.6   Celda O2: 0.588
```

Figura 4.8 Ejecución del programa con SpO2 81%, se requiere FiO2 40%

Figura 4.9 Leds verde, azul y rojo encendidos. Adicionalmente, se enciente la alarma audible

Adicionalmente, se realiza pruebas de recepción y visualización de la información en la plataforma Thingboard, información obtenida en tiempo real, con paciente de control sano.

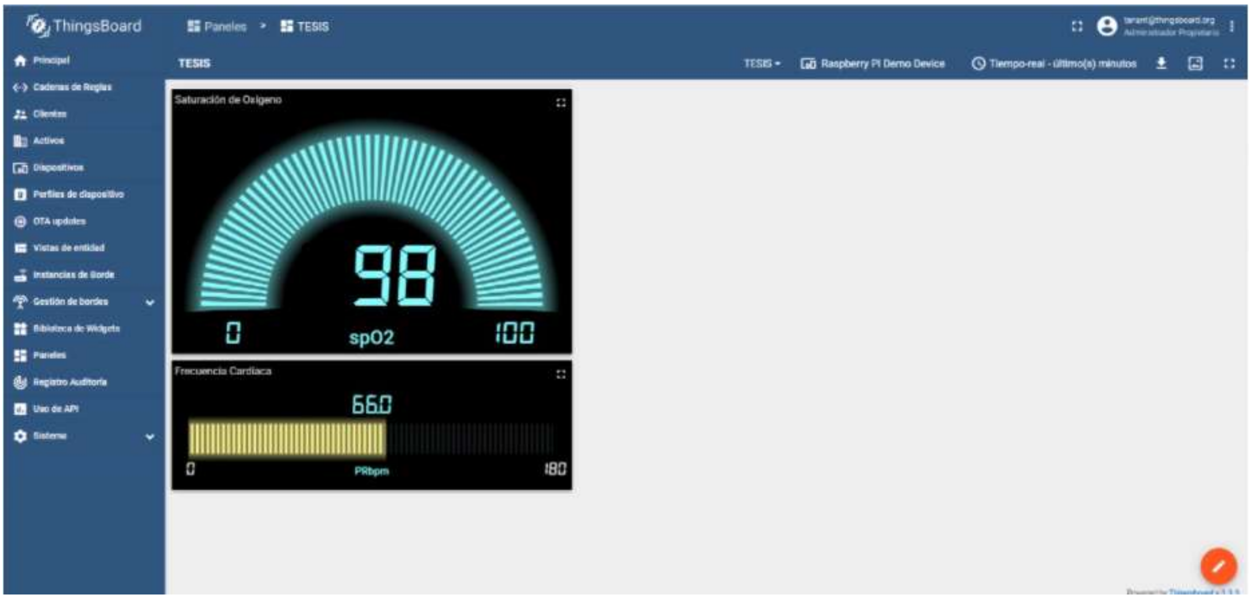


Figura 4.10 Captura de pantalla de las lecturas SpO2 y RC en la plataforma Thingsboard

4.3 Evaluación técnica y médica

4.3.1 Evaluación técnica

La evaluación técnica encierra el análisis de aspectos como:

- Fiabilidad: El prototipo fue probado técnicamente mediante pruebas funcionales, las cuales se describen a continuación:
    - Evaluación de voltajes entregados por la fuente de poder: El voltaje nominal es 24 [V], el voltaje medido es 23.12 [V].
    - Pruebas de estabilidad eléctrica en los dispositivos: El voltaje entregado por la fuente de voltaje es estable en modo de espera y a plena carga.
    - La medición de la FiO2 real, medida por la celda de oxígeno, presenta una variación o exactitud, menor al 4%.
    - Dado que el sistema es alámbrico, no presenta pérdidas de comunicación, siempre y cuando el oxímetro de pulso esté encendido.
- Rendimiento: El dispositivo es compacto y la potencia total nominal requerida es 50 [W]. El oxímetro de pulso empleado es de muy bajo consumo de potencia.
- Facilidad de mantenimiento: El sistema está compuesto por pocos elementos, compactos y de fácil acceso. El mantenimiento preventivo se limita a la limpieza y verificación de cada uno de ellos.
- Facilidad de operación: El sistema solo requiere ser encendido para entrar a modo de espera. Para funcionar requerirá que el oxímetro de pulso envíe los datos de los parámetros fisiológicos obtenidos.
- Seguridad: El dispositivo y cada uno de sus componentes, son seguros para el personal de salud y el usuario final. En la oxigenoterapia de alto flujo, la mezcla de aire y O2, debe ser humidificada y preferiblemente calefaccionada. El sistema cumple a través del humidificador y el circuito paciente, respectivamente.

4.3.2 Evaluación médica

La evaluación médica encierra el análisis de aspectos como usabilidad, confiabilidad, efectividad, utilidad y aceptación. Estos parámetros son obtenidos mediante encuesta, realizada a un profesional de la salud que realiza procedimientos de oxigenoterapia de alto flujo. Las respuestas son basadas en el comportamiento del sistema al configurar directamente la saturación de oxígeno, es decir, sin emplear los datos recibidos por el oxímetro de pulso, debido a que, en el sujeto de control sano, no se requiere FiO2 superior a la del aire. El resultado de la encuesta es mostrado en el Anexo G.

- Usabilidad:

¿Tuvo dificultad para aprender a utilizar el sistema de control automático de oxigenoterapia de alto flujo?

1 2 3 4 5

¿El rendimiento fue óptimo?

1 2 3 4 5

- Confiabilidad:

Indique el nivel de seguridad del dispositivo según su apreciación

1 2 3 4 5

¿El sistema le ha alertado correctamente cuando ha existido alguna complicación con el paciente?

1 2 3 4 5

- Efectividad:

¿El sistema de control automático de oxigenoterapia de alto flujo le permitió visualizar la información?

1 2 3 4 5

¿El sistema de control automático de oxigenoterapia de alto flujo le permitió visualizar y escuchar alarmas?

1 2 3 4 5

¿La FiO2 real entregada por el dispositivo se ajusta al rango empleado por usted al someter a un paciente a la oxigenoterapia de alto flujo convencional?

1 2 3 4 5

- Utilidad:

¿Considera usted que el sistema puede ser empleado en ambientes hospitalarios?

1 2 3 4 5

¿Considera usted que el sistema puede ser empleado en ambientes no hospitalarios?

1 2 3 4 5

¿Facilitaría el sistema la atención oportuna a un paciente que recibe oxigenoterapia de alto flujo?

1 2 3 4 5

- Aceptación:

¿En las condiciones actuales, sometería a sus pacientes al sistema desarrollado?

1 2 3 4 5

¿En las condiciones actuales, recomendaría el uso del sistema desarrollado?

1 2 3 4 5

4.4 Discusión

De acuerdo con el objetivo general, se procedió a realizar un sistema de control automático para ser usado en la oxigenoterapia respiratoria de alto flujo. Este sistema ajusta la Fracción de Inspiración de Oxigeno (FiO2), dependiendo de la condición del paciente, y evita complicaciones en el individuo que se encuentra en un estado de salud mermado. La oxigenoterapia es una herramienta fundamental para el tratamiento de la insuficiencia respiratoria, tanto aguda como crónica, es decir, el paciente respira el gas suministrado por el sistema, esto hace que la fracción de oxigeno inspirado administrado sea constante y predecible, logrando contribuir a prever factores de riesgos, brindándole atención médica de manera oportuna. Otra ventaja, es que, por ser un sistema no invasivo, evita efectos secundarios, como las infecciones, y, además, facilita su uso en centros de salud y ambientes no hospitalarios. Igualmente, el dispositivo de oxigenoterapia con control automático, permite mayor seguridad y mejor resultado en lo que se refiere a la pronta recuperación del paciente afectado. Por otro lado, se considera positivo que no es dependiente de la presencia permanente del personal especializado. Sin embargo, una de las debilidades que presenta este sistema es que no se puede aplicar a todos los pacientes en general, debido a que es si un paciente presenta una patología como el EPOC, disminuye la saturación de O2 de manera progresiva y sostenida en el tiempo no pueden ser sometidos a FiO2 porque en lugar de hacer un efecto positivo ocasionaría la depresión del centro Respiratorio. Siendo que el oxígeno, como muchos medicamentos, tiene rangos de dosis seguras, efectos adversos a la exposición prolongada y manifestaciones toxicas secundarias que se asocian con altas dosis y uso prolongado. Por lo que se debe tener presente que fracciones inspiradas de oxigeno (FiO2) mayores al 50% aumentan el riesgo de toxicidad.

En cuanto a lo referente a trabajos futuros relacionados con la presente investigación, se recomienda realizar mediciones con el sistema en un número mayor de participantes (pacientes) con diferentes cualidades tales como estado físico, edad, sexo, a fin de realizar un mayor análisis estadístico y así evaluar el desempeño del sistema en el total de los individuos.

CONCLUSIONES Y RECOMENDACIONES

CONCLUSIONES

1. Luego de las pruebas realizadas se logró construir un dispositivo basado en fotopletismografía, el cual permite reducir, pero no suprimir, el tiempo de supervisión del profesional de la salud. El sistema se encarga de suministrar la FiO2 requerida en todo momento, y en caso de no evolución, mediante alarmas, advierte, sin dejar de suministrar el FiO2 adecuado, que el paciente requiere mayor atención o cambio de dispositivo a ventilación mecánica.
2. Mediante el uso de un dispositivo basado en fotopletismografía, como el oxímetro de pulso, se pueden obtener parámetros fisiológicos de manera no invasiva, eliminando riesgos y efectos adversos inherentes al uso de procedimientos invasivos.
3. Aunque no existen modelos matemáticos para relacionar directa o indirectamente la fracción de inspiración de oxígeno, FiO2, con la saturación de oxígeno, SpO2, sí se puede estimar, con base en conocimientos de un profesional médico, el porcentaje de oxígeno o litros por minutos del mismo, que requerirá un paciente sometido a oxigenoterapia.
4. Mediante el uso de la celda de oxígeno, se pudo verificar que el sistema suministra una FiO2 con una diferencia menor al 4% del valor que debe ser entregado al paciente. Adicionalmente, el sistema cuenta con pocos componentes y, por tanto, presenta facilidad para su mantenimiento.

   Debido a esto, el sistema podría ser utilizado en ambientes no hospitalarios.

RECOMENDACIONES

1. Con el fin de tener una valoración completa, el sistema debe ser probado en sujetos con patologías respiratorias, siempre con la supervisión de un médico especialista.
2. Con el avance de la tecnología, ya se han desarrollado oxímetros de pulso capaces de medir también la frecuencia respiratoria, aunque sin comunicación USB o I2C. Este parámetro, al igual que la frecuencia cardiaca, es complementario a la saturación de oxígeno del paciente. Incorporar este dispositivo mejoraría las prestaciones de este sistema.
3. La celda de oxígeno varía su medición de oxígeno en función del tiempo de uso, por tanto, es recomendable que una vez al mes, el voltaje entregado en aire ambiental, es decir, FiO2 igual a 21%, sea medido para reconfigurar dicho parámetro en la programación.
4. Considerando que el sistema recibe oxígeno a través de un elemento externo, como un cilindro, este último debe ser observado regularmente, ya que el sistema no tiene sensores para verificar la cantidad de oxígeno restante.
5. Los parámetros fisiológicos del paciente, la FiO2 calculada y real, enviados a la nube, deben ser siempre monitorizados por profesionales de salud.

BIBLIOGRAFÍA

[1] Á. Castilla *et al.*, «Respuesta a la hipoxia: Un mecanismo sistémico basado en el control de la expresión génica», *Med. B. Aires*, abr. 2006, Accedido: abr. 26, 2020. [En línea]. Disponible en: https://www.scienceopen.com/document?vid=7a8b05bd-c99d-4706-8540-3903421c6dc7.

[2] «PROTOCOLO DE ADMINISTRACION DE OXIGENO», p. 10.

[3] «A_Sola_Oxigeno_y_sus_abusos.pdf». Accedido: abr. 26, 2020. [En línea]. Disponible en: https://www.seneo.es/images/site/publicaciones/congresos/2007/A_Sola_Oxigeno_y_sus_abusos.pdf.

[4] «Diseño e implementación de un pulsioxímetro». https://riunet.upv.es/handle/10251/91753 (accedido abr. 26, 2020).

[5] «Recomendaciones Para El Estudio de La Cooximetra (2010) (1)». https://dokumen.tips/documents/recomendaciones-para-el-estudio-de-la-cooximetria-2010-1.html (accedido abr. 26, 2020).

[6] S.-E. Carlson, U. Schnell, D. Schegg, y M. Liechti, «Increasing the performance of an optical pulsoximeter», US20050049468A1, mar. 03, 2005.

[7] F. J. Pilar Orive y Y. M. López Fernández, «Oxigenoterapia de alto flujo», *An. Pediatría Contin.*, vol. 12, n.º 1, pp. 25-29, ene. 2014, doi: 10.1016/S1696-2818(14)70163-5.

[8] D. A. Barker, M. D. Stewart, P. G. Hawkins, K. P. O'donnell, y R. W. Burgess, «Aparato de ayuda respiratoria», ES2720224T3, jul. 18, 2019.

[9] T. M. Kadarina and R. Priambodo, "Monitoring heart rate and SpO2 using Thingsboard IoT platform for mother and child preventive healthcare," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 453, p. 012028, Nov. 2018, doi: 10.1088/1757-899X/453/1/012028.

[10] D. Merkel, «Docker: lightweight Linux containers for consistent development and deployment», *undefined*, 2014, Accedido: 28 de marzo de 2022. [En línea]. Disponible en: https://www.semanticscholar.org/paper/Docker%3A-lightweight-Linux-containers-for-consistent-Merkel/875d90d4f66b07f90687b27ab304e04a3f666fc2

[11] D. M. MIHAI, “Distributed telemedicine system and method,” US10430552B2, Oct. 01, 2019.

[12] «Oxígeno». https://www.quimica.es/enciclopedia/Ox%C3%ADgeno.html (accedido 28 de marzo de 2022).

[13] Zschimmer, «Cómo se forma el oxígeno: así nació en el planeta Tierra», *ZS España*, 29 de julio de 2020. https://www.zschimmer-schwarz.es/noticias/como-se-forma-el-oxigeno-asi-nacio-en-el-planeta-tierra/ (accedido 28 de marzo de 2022).

[14] J. Martínez, A. Fernández Bremauntz, P. Osnaya, y Mexiko, Eds., *Cambio climático: una visión desde México*, Primera reimpresión. México, D.F: Secretaría de Medio Ambiente y Recursos Naturales, Instituto Nacional de Ecología, 2005.

[15] H. F. Bunn, M. H. May, W. F. Kocholaty, y C. E. Shields, «Hemoglobin function in stored blood», *J. Clin. Invest.*, vol. 48, n.º 2, pp. 311-321, feb. 1969, doi: 10.1172/JCI105987.

[16] M. E. Reid y N. Mohandas, «Red blood cell blood group antigens: structure and function», *Semin. Hematol.*, vol. 41, n.º 2, pp. 93-117, abr. 2004, doi: 10.1053/j.seminhematol.2004.01.001.

[17] H. A. Krebs, «Chemical Composition of Blood Plasma and Serum», *Annu. Rev. Biochem.*, vol. 19, n.º 1, pp. 409-430, 1950, doi: 10.1146/annurev.bi.19.070150.002205.

[18] H. Tazawa, «Oxygen and COi2nEtxhcehaAnvgiaenanEdmAbrcyido1-Base Regulation», p. 10.

[19] J. Petersson y R. W. Glenny, «Gas exchange and ventilation–perfusion relationships in the lung», *Eur. Respir. J.*, vol. 44, n.º 4, pp. 1023-1041, oct. 2014, doi: 10.1183/09031936.00037014.

[20] D. S. Karbing *et al.*, «Variation in the PaO2/FiO2 ratio with FiO2: mathematical and experimental description, and clinical relevance», *Crit. Care Lond. Engl.*, vol. 11, n.º 6, p. R118, 2007, doi: 10.1186/cc6174.

[21] J. Ameghino Bautista, J. Morales Corbacho, y M. Apolaya-Segura, «Correlación entre SO2/FiO2 y PaO2/FiO2 en pacientes con insuficiencia respiratoria en ventilación mecánica», *Rev. Cuba. Investig. Bioméd.*, vol. 37, n.º 3, pp. 1-9, sep. 2018.

[22] R. W. Ricci, M. Ditzler, y L. P. Nestor, «Discovering the Beer-Lambert Law», *J. Chem. Educ.*, vol. 71, n.º 11, p. 983, nov. 1994, doi: 10.1021/ed071p983.

[23] «How to Design a Better Pulse Oximeter | Analog Devices». https://www.analog.com/en/technical-articles/how-to-design-a-better-pulse-oximeter.html (accedido 28 de marzo de 2022).

[24] Y. Mendelson, «Pulse oximetry: theory and applications for noninvasive monitoring.», *Clin. Chem.*, 1992, doi: 10.1093/CLINCHEM/38.9.1601.

[25] «Figure 2.1: PPG Signal from Pulse Oximeter», *ResearchGate*. https://www.researchgate.net/figure/PPG-Signal-from-Pulse-Oximeter_fig1_266334216 (accedido 28 de marzo de 2022).

[26] «Fig. 1. Block diagram of Pulse Oximeter system [1]», *ResearchGate*. https://www.researchgate.net/figure/Block-diagram-of-Pulse-Oximeter-system-1_fig1_312325893 (accedido 28 de marzo de 2022).

[27] «Fig. 3. Absorción de HbO 2 y RHb para diferentes longitudes de onda», *ResearchGate*. https://www.researchgate.net/figure/Absorcion-de-HbO-2-y-RHb-para-diferentes-longitudes-de-onda_fig1_318227414 (accedido 28 de marzo de 2022).

[28] M. Nishimura, «High-flow nasal cannula oxygen therapy in adults», *J. Intensive Care*, vol. 3, n.º 1, p. 15, 2015, doi: 10.1186/s40560-015-0084-5.

[29] F. J. Pilar Orive y Y. M. López Fernández, «Oxigenoterapia de alto flujo», *An. Pediatría Contin.*, vol. 12, n.º 1, pp. 25-29, ene. 2014, doi: 10.1016/S1696-2818(14)70163-5.

[30] «The Effects of Flow on Airway Pressure During Nasal High-Flow Oxygen Therapy | Respiratory Care». http://rc.rcjournal.com/content/56/8/1151 (accedido 28 de marzo de 2022).

[31] «Actualización En Oxigenoterapia Para Enfermería. - EnfermeriaAPS». https://www.enfermeriaaps.com/portal/actualizacion-en-oxigenoterapia-para-enfermeria (accedido 28 de marzo de 2022).

[32] A. Madan, «Correlation between the levels of SpO2 and PaO2», *Lung India Off. Organ Indian Chest Soc.*, vol. 34, n.º 3, pp. 307-308, 2017, doi: 10.4103/lungindia.lungindia_106_17.

[33] T. Nguyen-Duc y T. Ngo, «Real-time IoT healthcare data visualisation», ene. 2021.

[34] «Chapter 7. Linux Containers with Docker Format Red Hat Enterprise Linux 7», *Red Hat Customer Portal*. https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/7.0_release_notes/chap-

red_hat_enterprise_linux-7.0_release_notes-linux_containers_with_docker_format (accedido 28 de marzo de 2022).

[35] «MAX30100 Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health | Maxim Integrated».

[36] «PSR-11-75-KE1-Data-Sheet.pdf». Accedido: 28 de marzo de 2022. [En línea]. Disponible en: https://docs.careoxmed.com/documents/PSR-11-75-KE1-Data-Sheet.pdf

[37] «ads1115.pdf». Accedido: 28 de marzo de 2022. [En línea]. Disponible en: https://cdn-shop.adafruit.com/datasheets/ads1115.pdf

[38] «Raspberry Pi 3 B+ Pinout with GPIO functions, Schematic and Specs in detail». https://www.etechnophiles.com/raspberry-pi-3-b-pinout-with-gpio-functions-schematic-and-specs-in-detail/ (accedido 28 de marzo de 2022).

[39] jecrespom, «Manejar GPIO Raspberry Pi», *Aprendiendo Arduino*, 4 de marzo de 2020. https://aprendiendoarduino.wordpress.com/2020/03/04/manejar-gpio-raspberry-pi/ (accedido 21 de abril de 2022).

[40] «7100R/7200R/7300R Series - Humidificador de burbujas by Genstar Technologies Company | MedicalExpo». https://www.medicalexpo.es/prod/genstar-technologies-company/product-81490-519108.html (accedido 28 de marzo de 2022).

ANEXOS

ANEXO A

MAX30100

Pulse Oximeter and Heart-Rate Sensor IC for Wearable Health

General Description

The MAX30100 is an integrated pulse oximetry and heart-rate monitor sensor solution. It combines two LEDs, a photodetector, optimized optics, and low-noise analog signal processing to detect pulse oximetry and heart-rate signals.

The MAX30100 operates from 1.8V and 3.3V power supplies and can be powered down through software with negligible standby current, permitting the power supply to remain connected at all times.

Applications

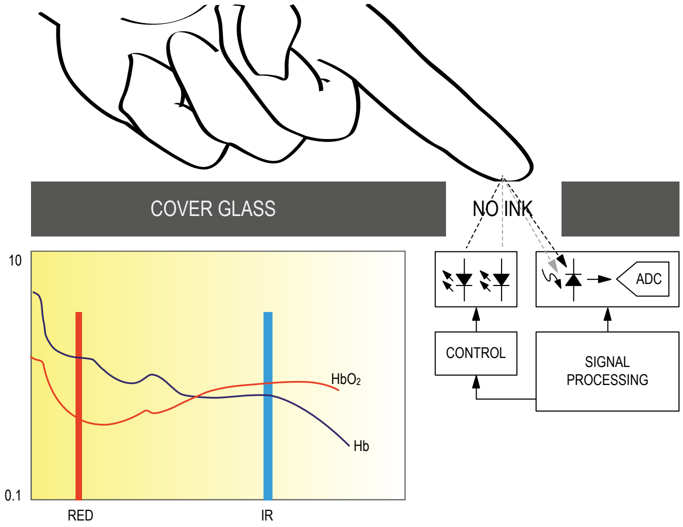
- Wearable Devices
- Fitness Assistant Devices
- Medical Monitoring Devices

Benefits and Features

- Complete Pulse Oximeter and Heart-Rate Sensor Solution Simplifies Design
  - Integrated LEDs, Photo Sensor, and High-Performance Analog Front -End
  - Tiny 5.6mm x 2.8mm x 1.2mm 14-Pin Optically Enhanced System-in-Package
- Ultra-Low-Power Operation Increases Battery Life for Wearable Devices
  - Programmable Sample Rate and LED Current for Power Savings
  - Ultra-Low Shutdown Current (0.7μA, typ)
- Advanced Functionality Improves Measurement Performance
  - High SNR Provides Robust Motion Artifact Resilience
  - Integrated Ambient Light Cancellation
  - High Sample Rate Capability
  - Fast Data Output Capability

Ordering Information ***appears at end of data sheet.***

System Block Diagram



Absolute Maximum Ratings

V_{DD} to GND	-0.3V to +2.2V
GND to PGND	-0.3V to +0.3V
x_DRV, x_LED+ to PGND	-0.3V to +6.0V
All Other Pins to GND	-0.3V to +6.0V
Output Short-Circuit Current Duration	Continuous
Continuous Input Current into Any Terminal	±20mA
Continuous Power Dissipation (T_A = +70°C)	
OESIP (derate 5.8mW/°C above +70°C)	464mW
Operating Temperature Range	-40°C to +85°C
Soldering Temperature (reflow)	+260°C
Storage Temperature Range	-40°C to +105°C

Package Thermal Characteristics (Note 1)

OESIP

Junction-to-Ambient Thermal Resistance (θ_{JA})	150°C/W
Junction-to-Case Thermal Resistance (θ_{JC})	170°C/W

Note 1: Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to **www.maximintegrated.com/thermal-tutorial**.

Electrical Characteristics

(V_{DD} = 1.8V, V_{IR_LED+} = V_{R_LED+} = 3.3V, T_A = +25°C, min/max are from T_A = -40°C to +85°C, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
POWER SUPPLY							
Power-Supply Voltage	V_{DD}	Guaranteed by RED and IR count tolerance		1.7	1.8	2.0	V
LED Supply Voltage (R_LED+ or IR_LED+ to PGND)	V_{LED+}	Guaranteed by PSRR of LED Driver		3.1	3.3	5.0	V
Supply Current	I_{DD}	SpO_2 and heart rate modes, PW = 200µs, 50sps			600	1200	µA
		Heart rate only mode, PW = 200µs, 50sps			600	1200	
Supply Current in Shutdown	I_{SHDN}	T_A = +25°C, MODE = 0x80			0.7	10	µA
SENSOR CHARACTERISTICS							
ADC Resolution					14		bits
Red ADC Count (Note 3)	RED_C	Propriety ATE setup RED_PA = 0x05, LED_PW = 0x00, SPO2_SR = 0x07, T_A = +25°C		23,000	26,000	29,000	Counts
IR ADC Count (Note 3)	IR_C	Propriety ATE setup IR_PA = 0x09, LED_PW = 0x00, SPO2_SR = 0x07, T_A = +25°C		23,000	26,000	29,000	Counts
Dark Current Count	DC_C	RED_PA = IR_PA = 0x00, LED_PW = 0x03, SPO2_SR = 0x01			0	3	Counts
DC Ambient Light Rejection (Note 4)	ALR	Number of ADC counts with finger on sensor under direct sunlight (100K lux) LED_PW = 0x03, SPO2_SR = 0x01	RED LED		0		Counts
			IR LED		0		

Electrical Characteristics (continued)

(V_{DD} = 1.8V, V_{IR_LED+} = V_{R_LED+} = 3.3V, T_A = +25°C, min/max are from T_A = -40°C to +85°C, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
IR ADC Count—PSRR (V_{DD})	$PSRR_{VDD}$	Propriety ATE setup 1.7V < V_{DD} < 2.0V, LED_PW = 0x03, SPO2_SR = 0x01, IR_PA = 0x09, IR_PA = 0x05, T_A = +25°C		0.25	2	%
		Frequency = DC to 100kHz, 100mV_{P-P}		10		LSB
RED/IR ADC Count—PSRR (X_LED+)	$PSRR_{LED}$	Propriety ATE setup 3.1V < X_LED+ < 5V, LED_PW = 0x03, SPO2_SR = 0x01, IR_PA = 0x09, IR_PA = 0x05, T_A = +25°C		0.05	2	%
		Frequency = DC to 100kHz, 100mV_{P-P}		10		LSB
ADC Integration Time	INT	LED_PW = 0x00		200		μs
		LED_PW = 0x03		1600		μs
IR LED CHARACTERISTICS (Note 4)						
LED Peak Wavelength	λ_P	I_{LED} = 20mA, T_A = +25°C	870	880	900	nm
Full Width at Half Max	Δλ	I_{LED} = 20mA, T_A = +25°C		30		nm
Forward Voltage	V_F	I_{LED} = 20mA, T_A = +25°C		1.4		V
Radiant Power	P_O	I_{LED} = 20mA, T_A = +25°C		6.5		mW
RED LED CHARACTERISTICS (Note 4)						
LED Peak Wavelength	λ_P	I_{LED} = 20mA, T_A = +25°C	650	660	670	nm
Full Width at Half Max	Δλ	I_{LED} = 20mA, T_A = +25°C		20		nm
Forward Voltage	V_F	I_{LED} = 20mA, T_A = +25°C		2.1		V
Radiant Power	P_O	I_{LED} = 20mA, T_A = +25°C		9.8		mW
TEMPERATURE SENSOR						
Temperature ADC Acquisition Time	T_T	T_A = +25°C		29		ms
Temperature Sensor Accuracy	T_A	T_A = +25°C		±1		°C
Temperature Sensor Minimum Range	T_{MIN}			-40		°C
Temperature Sensor Maximum Range	T_{MAX}			85		°C

Electrical Characteristics (continued)

(V_{DD} = 1.8V, V_{IR_LED+} = V_{R_LED+} = 3.3V, T_A = +25°C, min/max are from T_A = -40°C to +85°C, unless otherwise noted.) (Note 2)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
DIGITAL CHARACTERISTICS (SDA, SDA, $\overline{\text{INT}}$)						
Output Low Voltage SDA, $\overline{\text{INT}}$	V_{OL}	I_{SINK} = 6mA			0.4	V
I2C Input Voltage Low	V_{IL_I2C}	SDA, SCL			0.4	V
I2C Input Voltage High	V_{IH_I2C}	SDA, SCL	1.4			V
Input Hysteresis	V_{HYS}	SDA, SCL		200		mV
Input Capacitance	C_{IN}	SDA, SCL		10		pF
Input Leakage Current	I_{IN}	V_{IN} = 0V, T_A = +25°C (SDA, SCL, INT)		0.01	1	μA
		V_{IN} = 5.5V, T_A = +25°C (SDA, SCL, INT)		0.01	1	μA
I2C TIMING CHARACTERISTICS (SDA, SDA, $\overline{\text{INT}}$)						
I2C Write Address				AE		Hex
I2C Read Address				AF		Hex
Serial Clock Frequency	f_{SCL}		0		400	kHz
Bus Free Time Between STOP and START Conditions	t_{BUF}		1.3			μs
Hold Time (Repeated) START Condition	$t_{HD,START}$		0.6			μs
SCL Pulse-Width Low	t_{LOW}		1.3			μs
SCL Pulse-Width High	t_{HIGH}		0.6			μs
Setup Time for a Repeated START Condition	$t_{SU,START}$		0.6			μs
Data Hold Time	$t_{HD,DAT}$		0		900	ns
Data Setup Time	$t_{SU,DAT}$		100			ns
Setup Time for STOP Condition	$t_{SU,STOP}$		0.6			μs
Pulse Width of Suppressed Spike	t_{SP}		0		50	ns
Bus Capacitance	C_B				400	pF
SDA and SCL Receiving Rise Time	t_R		20 + 0.1C_B		300	ns
SDA and SCL Receiving Fall Time	t_{RF}		20 + 0.1C_B		300	ns
SDA Transmitting Fall Time	t_{TF}		20 + 0.1C_B		300	ns

Note 2: All devices are 100% production tested at T_A = +25°C. Specifications over temperature limits are guaranteed by Maxim Integrated's bench or proprietary automated test equipment (ATE) characterization.

Note 3: Specifications are guaranteed by Maxim Integrated's bench characterization and by 100% production test using proprietary ATE setup and conditions.

Note 4: For design guidance only. Not production tested.

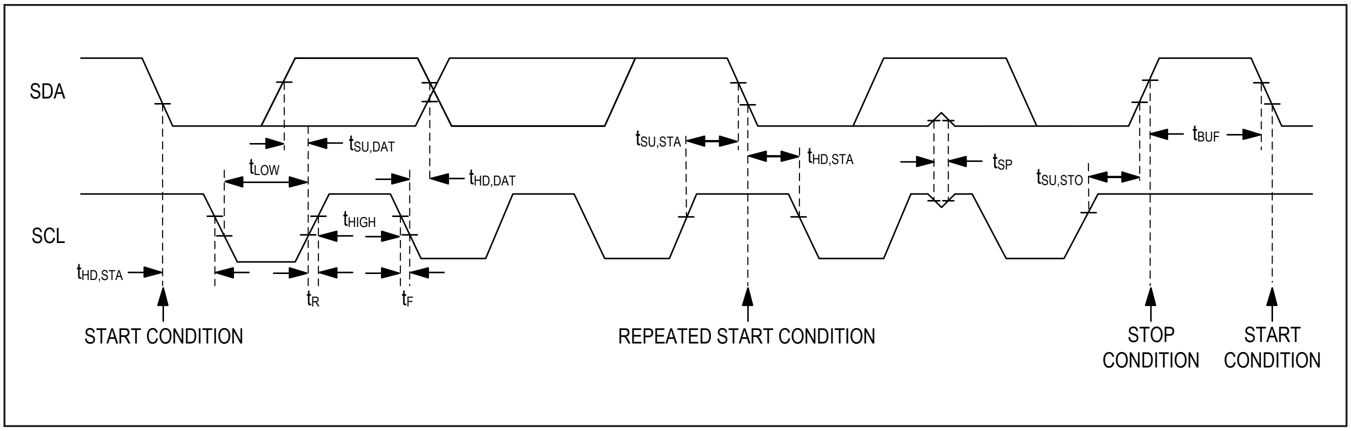
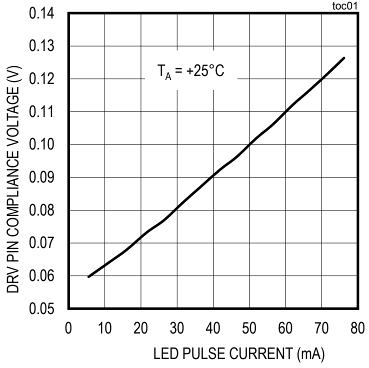


Figure 1. I²C-Compatible Interface Timing Diagram

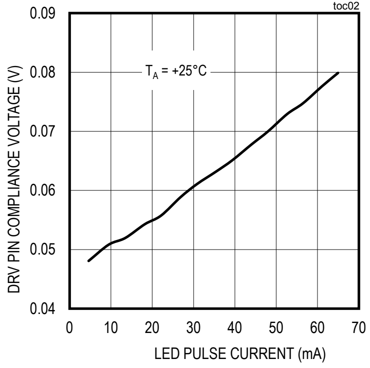
Typical Operating Characteristics

(V_{DD} = 1.8V, V_{IR_LED+} = V_{R_LED+} = 3.3V, T_A = +25°C, unless otherwise noted.)

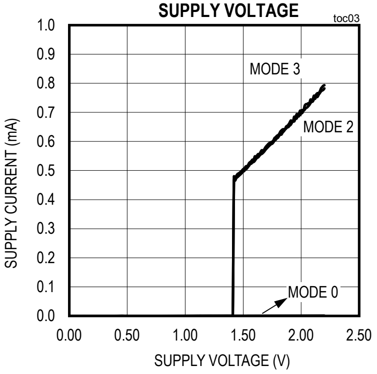
RED LED SUPPLY HEADROOM
(-10% CURRENT)



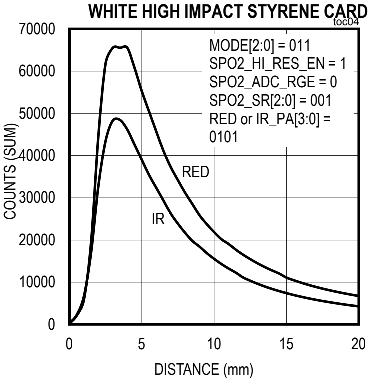
IR LED SUPPLY HEADROOM
(-10% CURRENT)



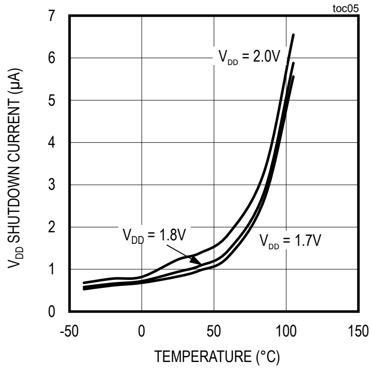
V_{DD} SUPPLY CURRENT vs.
SUPPLY VOLTAGE



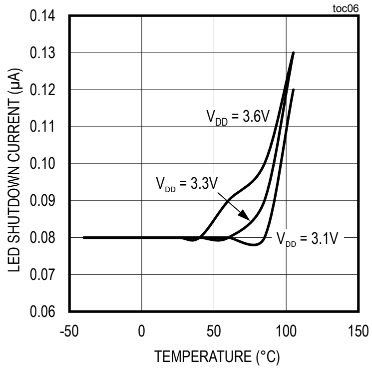
DC COUNTS vs. DISTANCE FOR
WHITE HIGH IMPACT STYRENE CARD



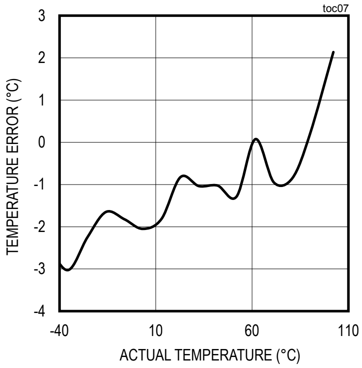
V_{DD} SHUTDOWN CURRENT
vs. TEMPERATURE



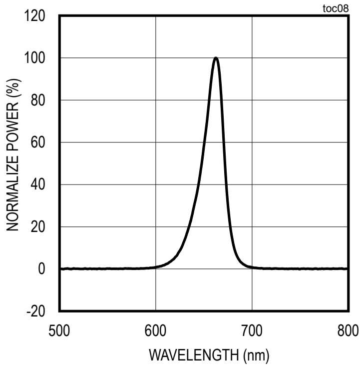
LED SHUTDOWN CURRENT
vs. TEMPERATURE



ON-BOARD TEMPERATURE vs. ERROR



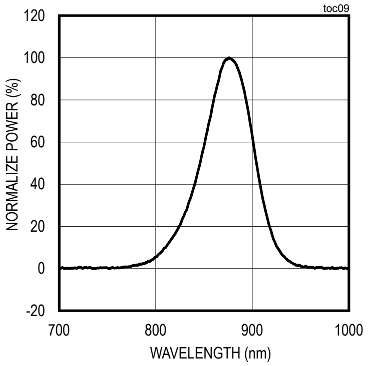
RED LED SPECTRA at +30°C



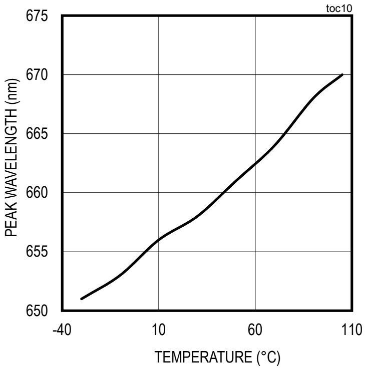
Typical Operating Characteristics (continued)

(V_{DD} = 1.8V, V_{IR_LED+} = V_{R_LED+} = 3.3V, T_A = +25°C, unless otherwise noted.)

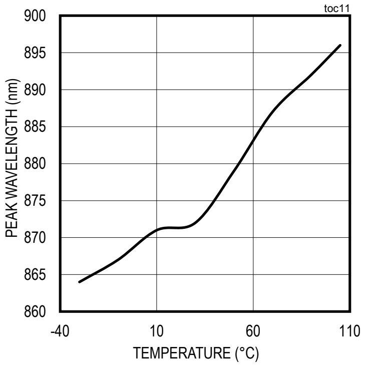
IR LED SPECTRA at +30°C



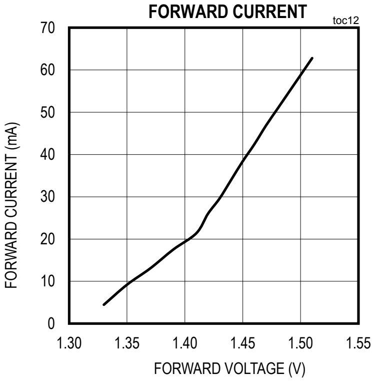
RED LED WAVELENGTH vs. TEMPERATURE AT LED CURRENT = 25mA



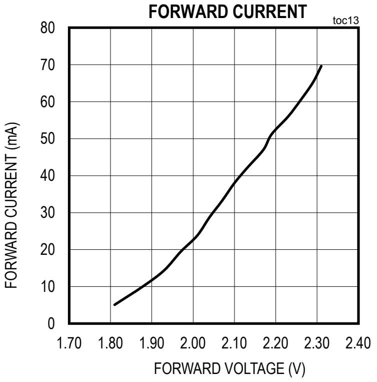
IR LED WAVELENGTH vs. TEMPERATURE AT LED CURRENT = 25mA



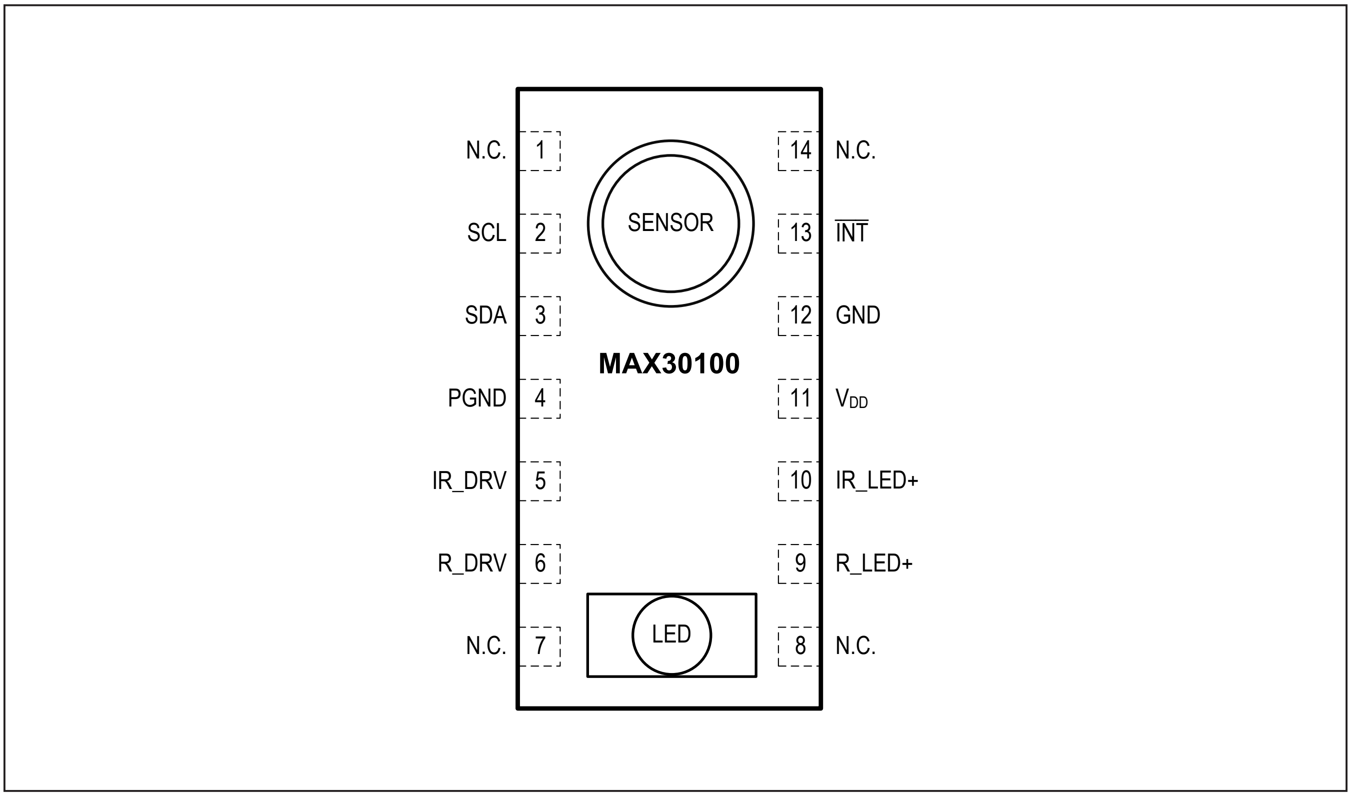
RED LED FORWARD VOLTAGE vs. FORWARD CURRENT



IR LED FORWARD VOLTAGE vs. FORWARD CURRENT



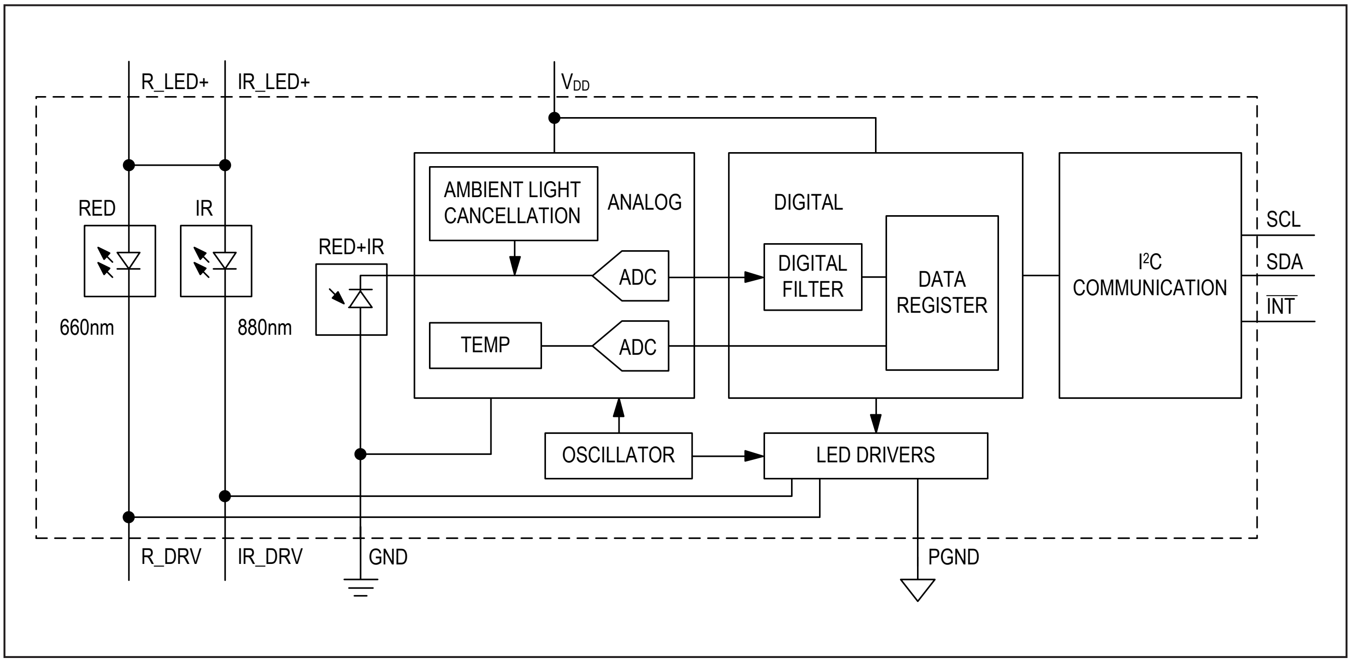
Pin Configuration



Pin Description

PIN	NAME	FUNCTION
1, 7, 8, 14	N.C.	No Connection. Connect to PCB Pad for Mechanical Stability.
2	SCL	I^2C Clock Input
3	SDA	I^2C Clock Data, Bidirectional (Open-Drain)
4	PGND	Power Ground of the LED Driver Blocks
5	IR_DRV	IR LED Cathode and LED Driver Connection Point. Leave floating in circuit.
6	R_DRV	Red LED Cathode and LED Driver Connection Point. Leave floating in circuit.
9	R_LED+	Power Supply (Anode Connection) for Red LED. Bypass to PGND for best performance. Connected to IR_LED+ internally.
10	IR_LED+	Power Supply (Anode Connection) for IR LED. Bypass to PGND for best performance. Connected to R_LED+ internally.
11	V_{DD}	Analog Power Supply Input. Bypass to GND for best performance.
12	GND	Analog Ground
13	$\overline{\text{INT}}$	Active-Low Interrupt (Open-Drain)

Functional Diagram



Detailed Description

The MAX30100 is a complete pulse oximetry and heart-rate sensor system solution designed for the demanding requirements of wearable devices. The MAX30100 provides very small total solution size without sacrificing optical or electrical performance. Minimal external hardware components are needed for integration into a wearable device.

The MAX30100 is fully configurable through software registers, and the digital output data is stored in a 16-deep FIFO within the device. The FIFO allows the MAX30100 to be connected to a microcontroller or microprocessor on a shared bus, where the data is not being read continuously from the device's registers.

SpO_2 Subsystem

The SpO_2 subsystem in the MAX30100 is composed of ambient light cancellation (ALC), 16-bit sigma delta ADC, and proprietary discrete time filter.

The SpO_2 ADC is a continuous time oversampling sigma delta converter with up to 16-bit resolution. The ADC output data rate can be programmed from 50Hz to 1kHz. The MAX30100 includes a proprietary discrete time filter to reject 50Hz/60Hz interference and low-frequency residual ambient noise.

Temperature Sensor

The MAX30100 has an on-chip temperature sensor for (optionally) calibrating the temperature dependence of the SpO_2 subsystem.

The SpO_2 algorithm is relatively insensitive to the wavelength of the IR LED, but the red LED's wavelength is critical to correct interpretation of the data. The temperature sensor data can be used to compensate the SpO_2 error with ambient temperature changes.

LED Driver

The MAX30100 integrates red and IR LED drivers to drive LED pulses for SpO_2 and HR measurements. The LED current can be programmed from 0mA to 50mA (typical only) with proper supply voltage. The LED pulse width can be programmed from 200µs to 1.6ms to optimize measurement accuracy and power consumption based on use cases.

Table 1. Register Maps and Descriptions

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
STATUS											
Interrupt Status	A_FULL	TEMP_RDY	HR_RDY	SPO2_RDY				PWR_RDY	0x00	0X00	R
Interrupt Enable	ENB_A_FULL	ENB_TEP_RDY	ENB_HR_RDY	ENB_SO2_RDY					0x01	0X00	R/W
FIFO											
FIFO Write Pointer					FIFO_WR_PTR[3:0]				0x02	0x00	R/W
Over Flow Counter					OVF_COUNTER[3:0]				0x03	0x00	R/W
FIFO Read Pointer					FIFO_RD_PTR[3:0]				0x04	0x00	R/W
FIFO Data Register	FIFO_DATA[7:0]								0x05	0x00	R/W
CONFIGURATION											
Mode Configuration	SHDN	RESET			TEMP_EN	MODE[2:0]			0x06	0x00	R/W
SPO2 Configuration		SPO2_HI_RES_EN	RESERVED	SPO2_SR[2:0]			LED_PW[1:0]		0x07	0x00	R/W
RESERVED									0x08	0x00	R/W
LED Configuration	RED_PA[3:0]				IR_PA[3:0]				0x09	0x00	R/W
RESERVED									0x0A – 0x15	0x00	R/W
TEMPERATURE											
Temp_Integer	TINT[7:0]								0x16	0x00	R/W
Temp_Fraction					TFRAC[3:0]				0x17	0x00	R/W
RESERVED									0x8D	0x00	R/W
PART ID											
Revision ID	REV_ID[7:0]								0xFE	0xXX*	R
Part ID	PART_ID[7]								0xFF	0x11	R/W

*XX denotes any 2-digit hexidecimal number (00 to FF). Contact Maxim Integrated for the Revision ID number assigned for your product.

Interrupt Status (0x00)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Interrupt Status	A_FULL	TEMP_ RDY	HR_RDY	SPO2_ RDY				PWR_ RDY	0x00	0X00	R

There are 5 interrupts and the functionality of each is exactly the same: pulling the active-low interrupt pin into its low state until the interrupt is cleared.

The interrupts are cleared whenever the interrupt status register is read, or when the register that triggered the interrupt is read. For example, if the SpO_2 sensor triggers an interrupt due to finishing a conversion, reading either the FIFO data register or the interrupt register clears the interrupt pin (which returns to its normal high state), and also clears all the bits in the interrupt status register to zero.

Bit 7: FIFO Almost Full Flag (A_FULL)

In SpO_2 and heart-rate modes, this interrupt triggers when the FIFO write pointer is the same as the FIFO read pointer minus one, which means that the FIFO has only one unwritten space left. If the FIFO is not read within the next conversion time, the FIFO becomes full and future data is lost.

Bit 6: Temperature Ready Flag (TEMP_RDY)

When an internal die temperature conversion is finished, this interrupt is triggered so the processor can read the temperature data registers.

Bit 5: Heart Rate Data Ready (HR_RDY)

In heart rate or SPO_2 mode, this interrupt triggers after every data sample is collected. A heart rate data sample consists of one IR data point only. This bit is automatically cleared when the FIFO data register is read.

Bit 4: SpO_2 Data Ready (SPO2_RDY)

In SpO_2 mode, this interrupt triggers after every data sample is collected. An SpO_2 data sample consists of one IR and one red data points. This bit is automatically cleared when the FIFO data register is read.

Bit 3: RESERVED

This bit should be ignored and always be zero in normal operation.

Bit 2: RESERVED

This bit should be ignored and always be zero in normal operation.

Bit 1: RESERVED

This bit should be ignored and always be zero in normal operation.

Bit 0: Power Ready Flag (PWR_RDY)

On power-up or after a brownout condition, when the supply voltage V_{DD} transitions from below the UVLO voltage to above the UVLO voltage, a power-ready interrupt is triggered to signal that the IC is powered up and ready to collect data.

Interrupt Enable (0x01)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Interrupt Enable	ENB_A_FULL	ENB_TEP_RDY	ENB_HR_RDY	ENB_SO2_RDY					0x01	0X00	R/W

Each source of hardware interrupt, with the exception of power ready, can be disabled in a software register within the MAX30100 IC. The power-ready interrupt cannot be disabled because the digital state of the MAX30100 is reset upon a brownout condition (low power-supply voltage), and the default state is that all the interrupts are disabled. It is important for the system to know that a brownout condition has occurred, and the data within the device is reset as a result.

When an interrupt enable bit is set to zero, the corresponding interrupt appears as 1 in the interrupt status register, but the $\overline{\text{INT}}$ pin is not pulled low.

The four unused bits (B3:B0) should always be set to zero (disabled) for normal operation.

FIFO (0x02–0x05)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
FIFO Write Pointer					FIFO_WR_PTR[3:0]				0x02	0x00	R/W
Over Flow Counter					OVF_COUNTER[3:0]				0x03	0x00	R/W
FIFO Read Pointer					FIFO_RD_PTR[3:0]				0x04	0x00	R/W
FIFO Data Register	FIFO_DATA[7:0]								0x05	0x00	R/W

FIFO Write Pointer

The FIFO write pointer points to the location where the MAX30100 writes the next sample. This pointer advances for each sample pushed on to the FIFO. It can also be changed through the I^2C interface when MODE[2:0] is nonzero.

FIFO Overflow Counter

When the FIFO is full, samples are not pushed on to the FIFO, samples are lost. OVF_COUNTER counts the number of samples lost. It saturates at 0xF. When a complete sample is popped from the FIFO (when the read pointer advances), OVF_COUNTER is reset to zero.

FIFO Read Pointer

The FIFO read pointer points to the location from where the processor gets the next sample from the FIFO via the I^2C interface. This advances each time a sample is popped from the FIFO. The processor can also write to this pointer after reading the samples, which would allow rereading samples from the FIFO if there is a data communication error.

FIFO Data

The circular FIFO depth is 16 and can hold up to 16 samples of SpO_2 channel data (Red and IR). The FIFO_DATA register in the I^2C register map points to the next sample to be read from the FIFO. FIFO_RD_PTR points to this sample. Reading FIFO_DATA register does not automatically increment the register address; burst reading this register reads the same address over and over. Each sample is 4 bytes of data, so this register has to be read 4 times to get one sample.

The above registers can all be written and read, but in practice, only the FIFO_RD_PTR register should be written to in operation. The others are automatically incremented or filled with data by the MAX30100. When starting a new SpO_2

or heart-rate conversion, it is recommended to first clear the FIFO_WR_PTR, OVF_COUNTER, and FIFO_RD_PTR registers to all zeros (0x00) to ensure the FIFO is empty and in a known state. When reading the MAX30100 registers in one burst-read I^{2}C transaction, the register address pointer typically increments so that the next byte of data sent is from the next register, etc. The exception to this is the FIFO data register, register 0x05. When reading this register, the address pointer does not increment, but the FIFO_RD_PTR does. So the next byte of data sent will represent the next byte of data available in the FIFO.

Reading from the FIFO

Normally, reading registers from the I^{2}C interface autoincrements the register address pointer, so that all the registers can be read in a burst read without an I^{2}C restart event. In the MAX30100, this holds true for all registers except for the FIFO_DATA register (0x05).

Reading the FIFO_DATA register does not automatically increment the register address; burst reading this register reads the same address over and over. Each sample is 4 bytes of data, so this register has to be read 4 times to get one sample.

The other exception is 0xFF, reading more bytes after the 0xFF register does not advance the address pointer back to 0x00, and the data read is not meaningful.

FIFO Data Structure

The data FIFO consists of a 16-sample memory bank that stores both IR and RED ADC data. Since each sample consists of one IR word and one RED word, there are 4 bytes of data for each sample, and therefore, 64 total bytes of data can be stored in the FIFO. Figure 2 shows the structure of the FIFO graphically.

The FIFO data is left-justified as shown in Table 1; i.e. the MSB bit is always in the bit 15 position regardless of ADC resolution.

Each data sample consists of an IR and a red data word (2 registers), so to read one sample requires 4 I^{2}C byte reads in a row. The FIFO read pointer is automatically incremented after each 4-byte sample is read.

In heart-rate only mode, the 3rd and 4th bytes of each sample return zeros, but the basic structure of the FIFO remains the same.

Write/Read Pointers

Table 2. FIFO Data

ADC RESOLUTION	IR [15]	IR [14]	IR [13]	IR [12]	IR [11]	IR [10]	IR [9]	IR [8]	IR [7]	IR [6]	IR [5]	IR [4]	IR [3]	IR [2]	IR [1]	IR [0]
16-bit	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
14-bit	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩		
12-bit	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩				
10-bit	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩						

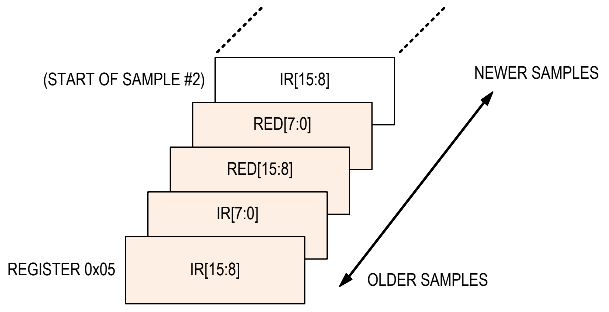


Figure 2. Graphical Representation of the FIFO Data Register

The locations to store new data, and the read pointer for reading data, are used to control the flow of data in the FIFO. The write pointer increments every time a new sample is added to the FIFO. The read pointer is incremented automatically every time a sample is read from the FIFO. To reread a sample from the FIFO, decrement its value by one and read the data register again.

The SpO_2 write/read pointers should be cleared (back to 0x0) upon entering SpO_2 mode or heart-rate mode, so that there is no old data represented in the FIFO. The pointers are not automatically cleared when changing modes, but they are cleared if V_{DD} is power cycled so that the V_{DD} voltage drops below its UVLO voltage.

Pseudo-Code Example of Reading Data from FIFO

First transaction: Get the FIFO_WR_PTR:

```
START;
Send device address + write mode
Send address of FIFO_WR_PTR;
REPEATED_START;
Send device address + read mode
Read FIFO_WR_PTR;
STOP;
```

The central processor evaluates the number of samples to be read from the FIFO:

```
NUM_AVAILABLE_SAMPLES = FIFO_WR_PTR – FIFO_RD_PTR
(Note: pointer wrap around should be taken into account)
NUM_SAMPLES_TO_READ = < less than or equal to NUM_AVAILABLE_SAMPLES >
```

Second transaction: Read `NUM_SAMPLES_TO_READ` **samples from the FIFO**:

```
START;
Send device address + write mode
Send address of FIFO_DATA;
REPEATED_START;
Send device address + read mode
for (i = 0; i < NUM_SAMPLES_TO_READ; i++) {
Read FIFO_DATA;
Save IR[15:8];
Read FIFO_DATA;
Save IR[7:0];
Read FIFO_DATA;
Save R[15:8];
Read FIFO_DATA;
Save R[7:0];
     }
STOP;
```

Third transaction: Write to FIFO_RD_PTR register. If the second transaction was successful, FIFO_RD_PTR points to the next sample in the FIFO, and this third transaction is not necessary. Otherwise, the processor updates the FIFO_RD_PTR appropriately, so that the samples are reread.

```
START;
Send device address + write mode
Send address of FIFO_RD_PTR;
Write FIFO_RD_PTR;
STOP;
```

Mode Configuration (0x06)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Mode Configuration	SHDN	RESET			TEMP_ EN	MODE[2:0]			0x06	0x00	R/W

Bit 7: Shutdown Control (SHDN)

The part can be put into a power-save mode by setting this bit to one. While in power-save mode, all registers retain their values, and write/read operations function as normal. All interrupts are cleared to zero in this mode.

Bit 6: Reset Control (RESET)

When the RESET bit is set to one, all configuration, threshold, and data registers are reset to their power-on-state. The only exception is writing both RESET and TEMP_EN bits to one at the same time since temperature data registers 0x16 and 0x17 are not cleared. The RESET bit is cleared automatically back to zero after the reset sequence is completed.

Bit 3: Temperature Enable (TEMP_EN)

This is a self-clearing bit which, when set, initiates a single temperature reading from the temperature sensor. This bit is cleared automatically back to zero at the conclusion of the temperature reading when the bit is set to one in heart rate or SpO_2 mode.

Bits 2:0: Mode Control

These bits set the operating state of the MAX30100. Changing modes does not change any other setting, nor does it erase any previously stored data inside the data registers.

Table 3. Mode Control

MODE[2:0]	MODE
000	Unused
001	Reserved (Do not use)
010	HR only enabled
011	SPO_2 enabled
100–111	Unused

SpO2 Configuration (0x07)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
SPO_2 Configuration		SPO2_HI_RES_EN	Reserved	SPO2_SR[2:0]			LED_PW[1:0]		0x07	0x00	R/W

Bit 6: SpO_2 High Resolution Enable (SPO2_HI_RES_EN)

Set this bit high. The SpO_2 ADC resolution is 16-bit with 1.6ms LED pulse width.

Bit 5: Reserved. Set low (default).

Bit 4:2: SpO_2 Sample Rate Control

These bits define the effective sampling rate, with one sample consisting of one IR pulse/conversion and one RED pulse/conversion.

The sample rate and pulse width are related, in that the sample rate sets an upper bound on the pulse width time. If the user selects a sample rate that is too high for the selected LED_PW setting, the highest possible sample rate will instead be programmed into the register.

Bits 1:0: LED Pulse Width Control

These bits set the LED pulse width (the IR and RED have the same pulse width), and therefore, indirectly set the integration time of the ADC in each sample. The ADC resolution is directly related to the integration time.

Table 4. SpO_2 Sample Rate Control

SPO2_SR[2:0]	SAMPLES (PER SECOND)
000	50
001	100
010	167
011	200
100	400
101	600
110	800
111	1000

Table 5. LED Pulse Width Control

LED_PW[1:0]	PULSE WIDTH (µs)	ADC RESOLUTION (BITS)
00	200	13
01	400	14
10	800	15
11	1600	16

LED Configuration (0x09)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
LED Configuration	RED_PA[3:0]				IR_PA[3:0]				0x09	0x00	R/W

Bits 7:4: Red LED Current Control

These bits set the current level of the Red LED as in Table 6.

Bits 3:0: IR LED Current Control

These bits set the current level of the IR LED as in Table 6.

Table 6. LED Current Control

Red_PA[3:0] OR IR_PA[3:0]	TYPICAL LED CURRENT (mA)*
0000	0.0
0001	4.4
0010	7.6
0011	11.0
0100	14.2
0101	17.4
0110	20.8
0111	24.0
1000	27.1
1001	30.6
1010	33.8
1011	37.0
1100	40.2
1101	43.6
1110	46.8
1111	50.0

*Actual measured LED current for each part can vary widely due to the proprietary trim methodology.

Temperature Data (0x16–0x17)

REGISTER	B7	B6	B5	B4	B3	B2	B1	B0	REG ADDR	POR STATE	R/W
Temp_Integer	TINT[7:0]								0x16	0x00	R/W
Temp_Fraction					TFRAC[3:0]				0x17	0x00	R/W

Temperature Integer

The on-board temperature ADC output is split into two registers, one to store the integer temperature and one to store the fraction. Both should be read when reading the temperature data, and the following equation shows how to add the two registers together:

$$T_{MEASURED} = T_{INTEGER} + T_{FRACTION}$$

This register stores the integer temperature data in two's complement format, where each bit corresponds to degree Celsius.

Table 7. Temperature Integer

REGISTER VALUE (hex)	TEMPERATURE (°C)
0x00	0
0x00	+1
...	...
0x7E	+126
0x7F	+127
0x80	-128
0x81	-127
...	...
0xFE	-2
0xFF	-1

Temperature Fraction

This register stores the fractional temperature data in increments of 0.0625°C (1/16th of a degree).

If this fractional temperature is paired with a negative integer, it still adds as a positive fractional value (e.g., -128°C + 0.5°C = -127.5°C).

Applications Information

Sampling Rate and Performance

The MAX30100 ADC is a 16-bit sigma delta converter. The ADC sampling rate can be configured from 50sps to 1ksps. The maximum sample rate for the ADC depends on the selected pulse width, which in turn, determines the ADC resolution. For instance, if the pulse width is set to 200μs, then the ADC resolution is 13 bits and all sample rates from 50sps to 1ksps are selectable. However, if the pulse width is set to 1600μs, then only sample rates of 100sps and 50sps can be set. The allowed sample rates for both SpO_2 and HR mode are summarized in Table 8 and Table 9.

Power Considerations

The LEDs in MAX30100 are pulsed with a low duty cycle for power savings, and the pulsed currents can cause ripples in the LED power supply. To ensure these pulses do not translate into optical noise at the LED outputs, the power supply must be designed to handle peak LED current. Ensure that the resistance and inductance from the power supply (battery, DC/DC converter, or LDO) to the device LED+ pins is much smaller than 1Ω, and that there is at least 1μF of power-supply bypass capacitance to a low impedance ground plane. The decoupling capacitor should be located physically as close as possible to the MAX30100 device.

In the heart-rate only mode, the red LED is inactive, and only the IR LED is used to capture optical data and determine the heart rate. This mode allows power savings due to the red LED being off; in addition, the IR_LED+ power supply can be reduced to save power because the forward voltage of the IR LED is significantly less than that of the red LED.

The average I_{DD} and LED current as function of pulse width and sampling rate is summarized in Table 10 to Table 13.

Table 8. SpO_2 Mode (Allowed Settings)

SAMPLES (per second)	PULSE WIDTH (μs)			
	200	400	800	1600
50	O	O	O	O
100	O	O	O	O
167	O	O	O	
200	O	O	O	
400	O	O		
600	O			
800	O			
1000	O			
Resolution (bits)	13	14	15	16

Table 9. Heart-Rate Mode (Allowed Settings)

SAMPLES (per second)	PULSE WIDTH (μs)			
	200	400	800	1600
50	O	O	O	O
100	O	O	O	O
167	O	O	O	
200	O	O	O	
400	O	O		
600	O	O		
800	O	O		
1000	O	O		
Resolution (bits)	13	14	15	16

Table 10. SpO_2 Mode: Average IDD Current (µA) R_PA = 0x3, IR_PA = 0x3

SAMPLES (per second)	PULSE WIDTH (µs)			
	200	400	800	1600
50	628	650	695	782
100	649	691	776	942
167	678	748	887	
200	692	775	940	
400	779	944		
600	865			
800	952			
1000	1037			

Table 11. SpO_2 Mode: Average LED Current (mA) R_PA = 0x3, IR_PA = 0x3

SAMPLES (per second)	PULSE WIDTH (µs)			
	200	400	800	1600
50	0.667	1.332	2.627	5.172
100	1.26	2.516	4.96	9.766
167	2.076	4.145	8.173	
200	2.491	4.93	9.687	
400	4.898	9.765		
600	7.319			
800	9.756			
1000	12.17			

Table 12. Heart-Rate Mode: Average IDD Current (µA) IR_PA = 0x3

SAMPLES (per second)	PULSE WIDTH (µs)			
	200	400	800	1600
50	608	616	633	667
100	617	634	669	740
167	628	658	716	831
200	635	670	739	876
400	671	740	878	
600	707	810		
800	743	881		
1000	779	951		

Table 13. Heart-Rate Mode: Average LED Current (mA) IR_PA = 0x3

SAMPLES (per second)	PULSE WIDTH (µs)			
	200	400	800	1600
50	0.256	0.511	1.020	2.040
100	0.512	1.022	2.040	4.077
167	0.854	1.705	3.404	6.795
200	1.023	2.041	4.074	8.130
400	2.042	4.074	8.123	
600	3.054	6.089		
800	4.070	8.109		
1000	5.079	10.11		

Hardware Interrupt

The active-low interrupt pin pulls low when an interrupt is triggered. The pin is open-drain and requires a pullup resistor or current source to an external voltage supply (up to +5V from GND). The interrupt pin is not designed to sink large currents, so the pullup resistor value should be large, such as 4.7kΩ.

The internal FIFO stores up to 16 samples, so that the system processor does not need to read the data after every sample. Temperature data may be needed to properly interpret SpO_2 data, but the temperature does not need to be sampled very often—once a second or every few seconds should be sufficient. In heart-rate mode temperature information is not necessary.

Table 14. Red LED Current Settings vs. LED Temperature Rise

RED LED CURRENT SETTING	RED LED DUTY CYCLE (% OF LED PULSE WIDTH TO SAMPLE TIME)	ESTIMATED TEMPERATURE RISE (ADD TO TEMPERATURE SENSOR MEASUREMENT) (°C)
0001 (3.1mA)	8	0.1
1111 (35mA)	8	2
0001 (3.1mA)	16	0.3
1111 (35mA)	16	4
0001 (3.1mA)	32	0.6
1111 (35mA)	32	8

Timing for Measurements and Data Collection

Timing in SpO_2 Mode

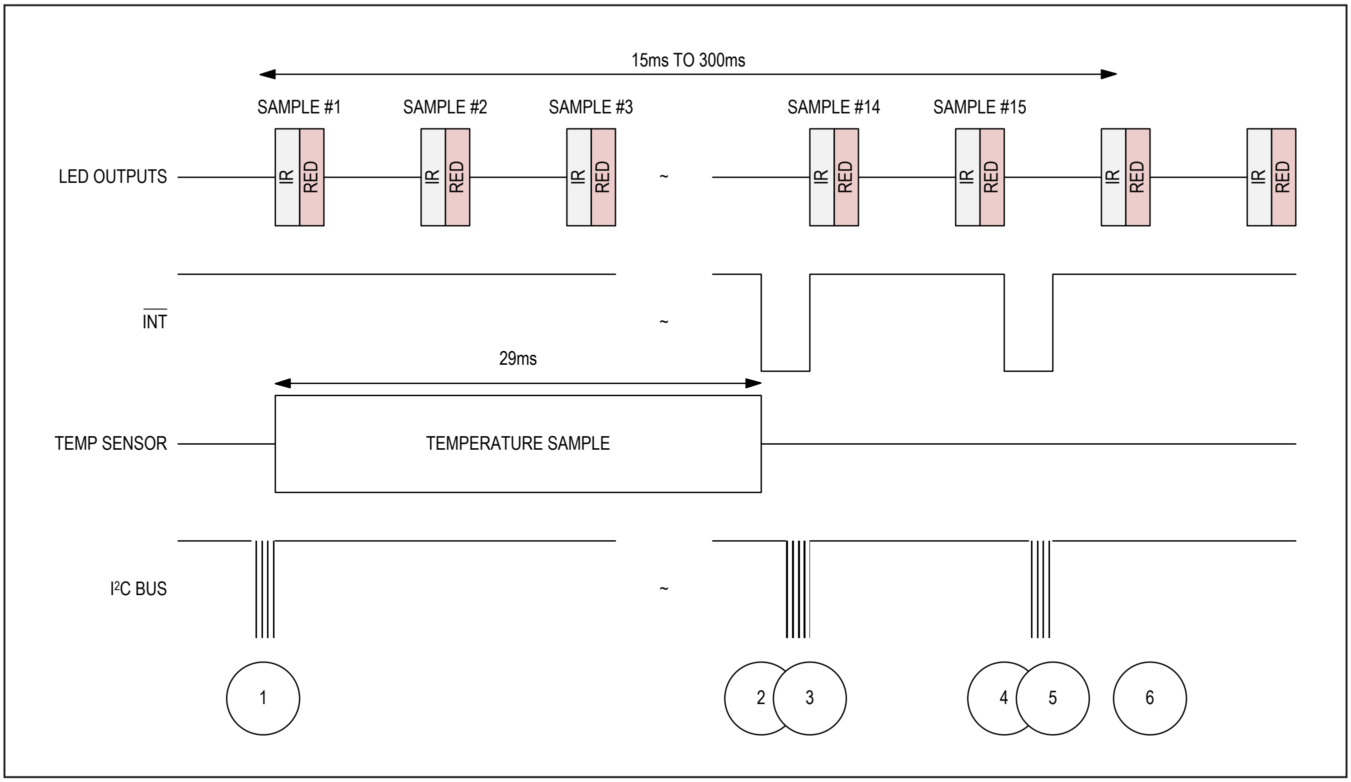


Figure 3. Timing for Data Acquisition and Communication When in SpO_2 Mode

Table 15. Events Sequence for Figure 3 in SpO_2 Mode

EVENT	DESCRIPTION	COMMENTS
1	Enter into SpO_2 mode. Initiate a temperature measurement.	I^2C Write Command Sets MODE[2:0] = 0x03. At the same time, set the TEMP_EN bit to initiate a single temperature measurement. Mask the SPO2_RDY Interrupt.
2	Temperature measurement complete, interrupt generated	TEMP_RDY interrupt triggers, alerting the central processor to read the data.
3	Temp data is read, interrupt cleared	
4	FIFO is almost full, interrupt generated	Interrupt is generated when the FIFO has only one empty space left.
5	FIFO data is read, interrupt cleared	
6	Next sample is stored	New sample is stored at the new read pointer location. Effectively, it is now the first sample in the FIFO.

Timing in Heart-Rate Mode

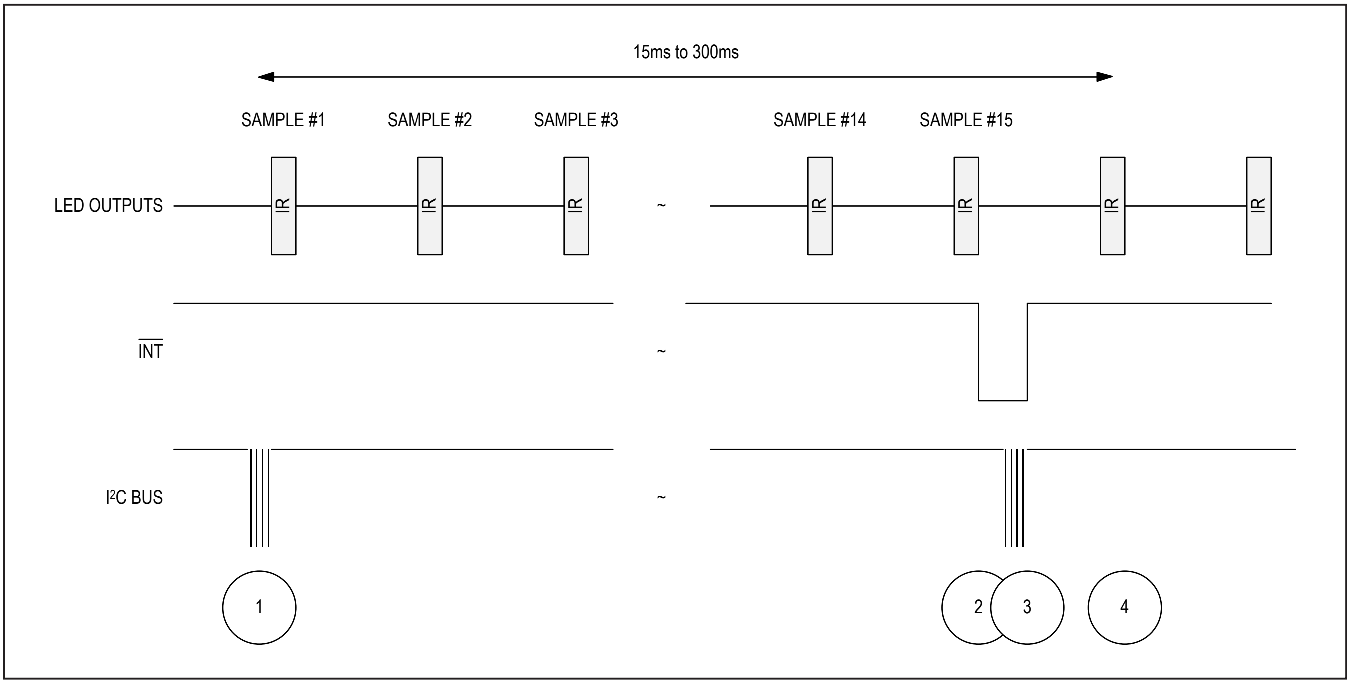


Figure 4. Timing for Data Acquisition and Communication When in Heart Rate Mode

Table 16. Events Sequence for Figure 4 in Heart-Rate Mode

EVENT	DESCRIPTION	COMMENTS
1	Enter into heart rate mode	I^2C Write Command Sets MODE[2:0] = 0x02. Mask the HR_RDY interrupt.
2	FIFO is almost full, interrupt generated	Interrupt is generated when the FIFO has only one empty space left.
3	FIFO data is read, interrupt cleared	
4	Next sample is stored	New sample is stored at the new read pointer location. Effectively, it is now the first sample in the FIFO.

Power Sequencing and Requirements

Power-Up Sequencing

Figure 5 shows the recommended power-up sequence for the MAX30100.

It is recommended to power the V_{DD} supply first, before the LED power supplies (R_LED+, IR_LED+). The interrupt and I^2C pins can be pulled up to an external voltage even when the power supplies are not powered up.

After the power is established, an interrupt occurs to alert the system that the MAX30100 is ready for operation. Reading the I^2C interrupt register clears the interrupt, as shown in Figure 5.

Power-Down Sequencing

The MAX30100 is designed to be tolerant of any power-supply sequencing on power-down.

I^2C Interface

The MAX30100 features an I^2C/SMBus-compatible, 2-wire serial interface consisting of a serial data line (SDA) and a serial clock line (SCL). SDA and SCL facilitate communication between the MAX30100 and the master at clock rates up to 400kHz. Figure 1 shows the 2-wire interface timing diagram. The master generates SCL and initiates data transfer on the bus. The master device writes data to the MAX30100 by transmitting the proper slave address followed by data. Each transmit sequence is framed by a START (S) or REPEATED START (Sr) condition and a STOP (P) condition. Each word transmitted to the MAX30100 is 8 bits long and is followed by an acknowledge clock pulse. A master reading data from the MAX30100 transmits the proper slave address followed by a series of nine SCL pulses.

The MAX30100 transmits data on SDA in sync with the master-generated SCL pulses. The master acknowledges receipt of each byte of data. Each read sequence is framed by a START (S) or REPEATED START (Sr) condition, a not acknowledge, and a STOP (P) condition. SDA operates as both an input and an open-drain output. A pullup resistor, typically greater than 500Ω, is required on SDA. SCL operates only as an input. A pullup resistor, typically greater than 500Ω, is required on SCL if there are multiple masters on the bus, or if the single master has an open-drain SCL output.

Bit Transfer

One data bit is transferred during each SCL cycle. The data on SDA must remain stable during the high period of the SCL pulse. Changes in SDA while SCL is high are control signals. See the *START and STOP Conditions* section.

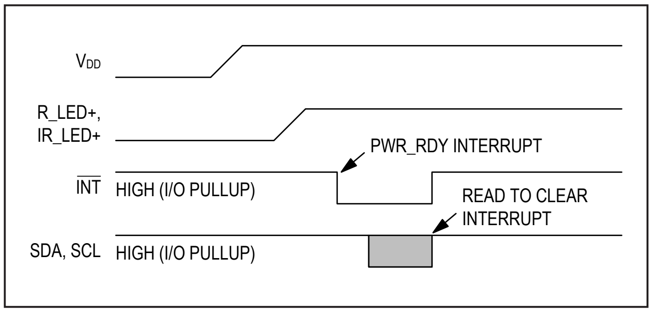


Figure 5. Power-Up Sequence of the Power-Supply Rails

START and STOP Conditions

SDA and SCL idle high when the bus is not in use. A master initiates communication by issuing a START condition. A START condition is a high-to-low transition on SDA with SCL high. A STOP condition is a low-to-high transition on SDA while SCL is high (Figure 6). A START condition from the master signals the beginning of a transmission to the MAX30100. The master terminates transmission, and frees the bus, by issuing a STOP condition. The bus remains active if a REPEATED START condition is generated instead of a STOP condition.

Early STOP Conditions

The MAX30100 recognizes a STOP condition at any point during data transmission except if the STOP condition occurs in the same high pulse as a START condition. For proper operation, do not send a STOP condition during the same SCL high pulse as the START condition.

Slave Address

A bus master initiates communication with a slave device by issuing a START condition followed by the 7-bit slave ID. When idle, the MAX30100 waits for a START condition followed by its slave ID. The serial interface compares each slave ID bit by bit, allowing the interface to power down and disconnect from SCL immediately if an incorrect slave ID is detected. After recognizing a START condition followed by the correct slave ID, the MAX30100 is ready to accept or send data. The LSB of the slave ID word is the Read/$\overline{\text{Write}}$ (R/$\overline{\text{W}}$) bit. R/$\overline{\text{W}}$ indicates whether the master is writing to or reading data from the MAX30100. R/$\overline{\text{W}}$ = 0 selects a write condition, R/$\overline{\text{W}}$ = 1 selects a read condition). After receiving the proper slave ID, the MAX30100 issues an ACK by pulling SDA low for one clock cycle.

The MAX30100 slave ID consists of seven fixed bits, B7–B1 (set to 0b1010111). The most significant slave ID bit (B7) is transmitted first, followed by the remaining bits. Table 18 shows the possible slave IDs of the device.

Acknowledge

The acknowledge bit (ACK) is a clocked 9th bit that the MAX30100 uses to handshake receipt each byte of data when in write mode (Figure 7). The MAX30100 pulls down SDA during the entire master-generated 9th clock pulse if the previous byte is successfully received. Monitoring ACK allows for detection of unsuccessful data transfers. An unsuccessful data transfer occurs if a receiving device is busy or if a system fault has occurred. In the event of an unsuccessful data transfer, the bus master will retry communication. The master pulls down SDA during the 9th clock cycle to acknowledge receipt of data when the MAX30100 is in read mode. An acknowledge is sent by the master after each read byte to allow data transfer to continue. A not-acknowledge is sent when the master reads the final byte of data from the MAX30100, followed by a STOP condition.

Table 17. Slave ID Description

B7	B6	B5	B4	B3	B2	B1	B0	WRITE ADDRESS	READ ADDRESS
1	0	1	0	1	1	1	R/$\overline{\text{W}}$	0xAE	0xAF

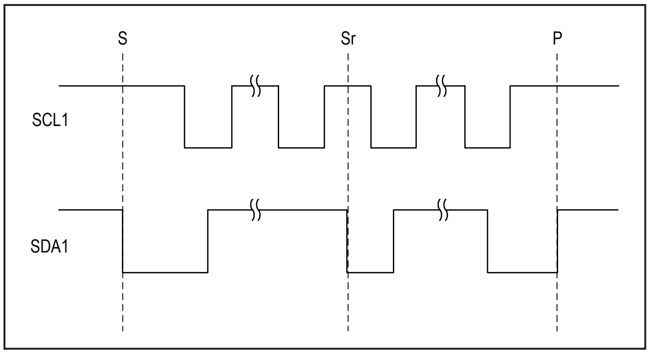


Figure 6. START, STOP, and REPEATED START Conditions

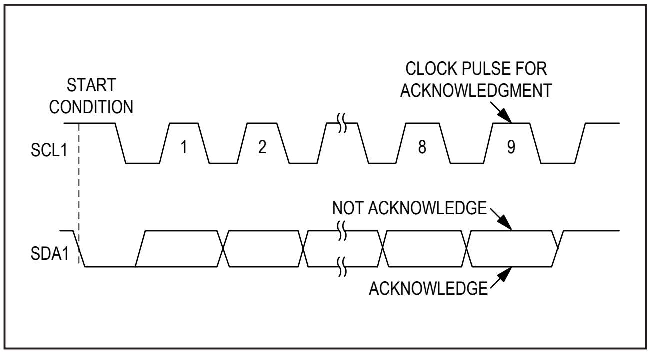


Figure 7. Acknowledge

Write Data Format

For the write operation, send the slave ID as the first byte followed by the register address byte and then one or more data bytes. The register address pointer increments automatically after each byte of data received. For example, the entire register bank can be written by at one time. Terminate the data transfer with a STOP condition. The write operation is shown in Figure 8.

The internal register address pointer increments automatically, so writing additional data bytes fill the data registers in order.

Read Data Format

For the read operation, two I2C operations must be performed. First, the slave ID byte is sent followed by the I2C register that you wish to read. Then a REPEATED START (Sr) condition is sent, followed by the read slave ID. The MAX30100 then begins sending data beginning with the register selected in the first operation. The read pointer increments automatically, so the MAX30100 continues sending data from additional registers in sequential order until a STOP (P) condition is received. The exception to this is the FIFO_DATA register, at which the read pointer no longer increments when reading additional bytes. To read the next register after FIFO_DATA, an I2C write command is necessary to change the location of the read pointer.

An initial write operation is required to send the read register address.

Data is sent from registers in sequential order, starting from the register selected in the initial I2C write operation. If the FIFO_DATA register is read, the read pointer does not automatically increment, and subsequent bytes of data contain the contents of the FIFO.

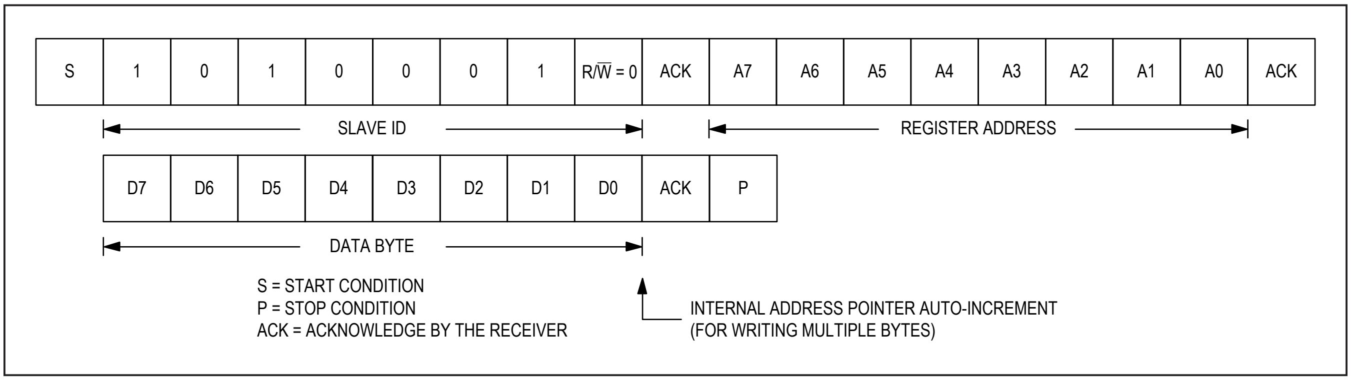


Figure 8. Writing One Data Byte to the MAX30100

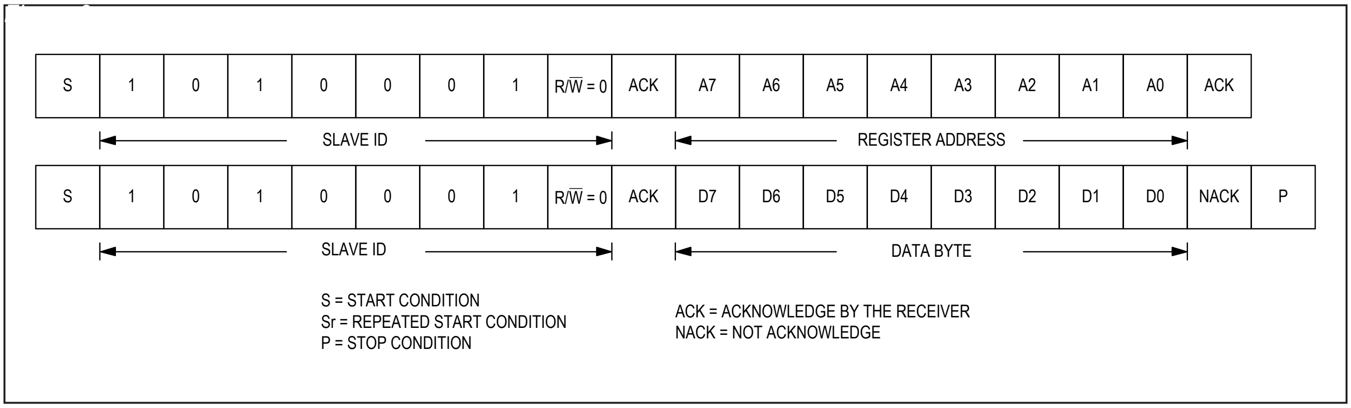


Figure 9. Reading One Byte of Data from the MAX30100

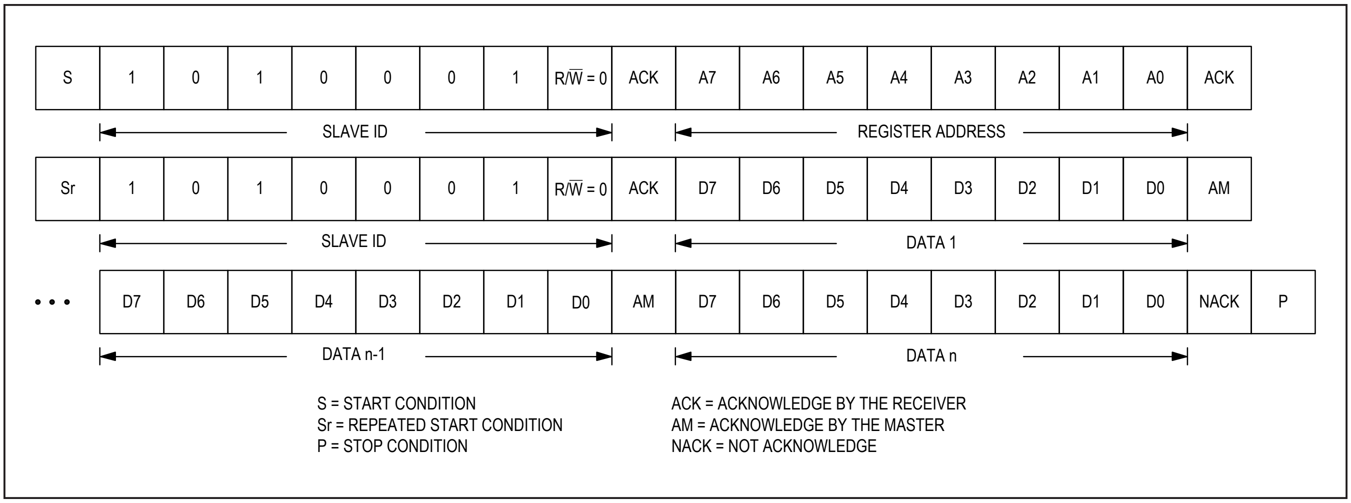
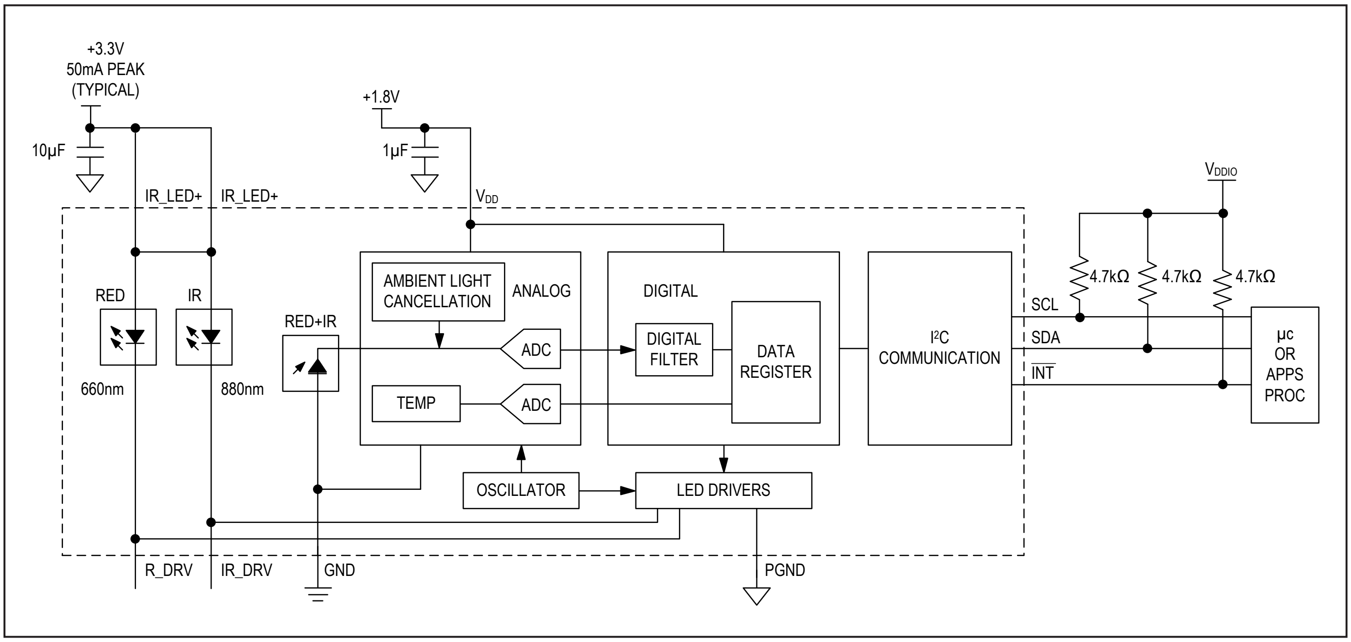


Figure 10. Reading Multiple Bytes of Data from the MAX30100

Typical Application Circuit



Ordering Information

PART	TEMP RANGE	PIN-PACKAGE
MAX30100EFD+	-40°C to +85°C	14 OESIP (0.8mm pitch)

+Denotes a lead(Pb)-free/RoHS-compliant package.

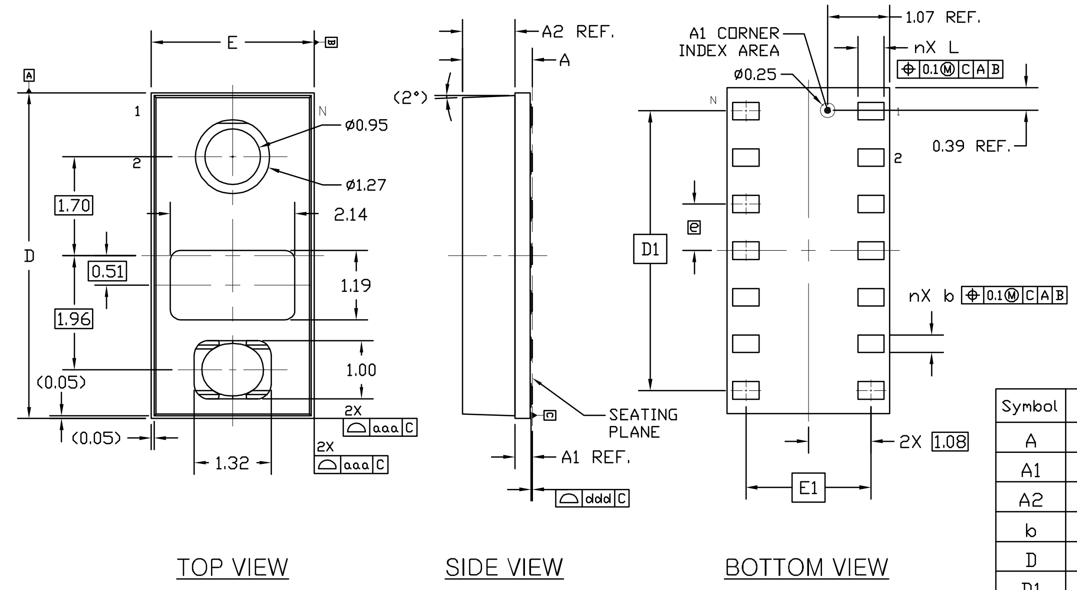
Chip Information

PROCESS: BiCMOS

Package Information

For the latest package outline information and land patterns (footprints), go to **www.maximintegrated.com/packages**. Note that a “+”, “#”, or “-” in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

PACKAGE TYPE	PACKAGE CODE	OUTLINE NO.	LAND PATTERN NO.
14 OESIP	F142D5+2	21-0880	90-0461



Symbol	MIN	NOM	MAX
A	1.1	1.2	1.3
A1	0.3 REF.		
A2	0.9 REF.		
b	0.25	0.30	0.35
D	5.5	5.6	5.7
D1	4.8 BSC		
E	2.7	2.8	2.9
E1	2.15 BSC		
e	0.8 BSC		
L	0.40	0.45	0.50
N	14		
aaa	0.1		
ddd	0.08		

NOTES:

1. ALL DIMENSIONS ARE IN MILLIMETERS. ANGLES ARE IN DEGREES.
2. DIMENSIONING & TOLERANCING CONFORM TO ASME Y14.5M—1994.
3. N IS THE TOTAL NUMBER OF TERMINALS.
4. MATERIAL MUST BE COMPLIANT WITH MAXIM SPECIFICATION 10-0131 FOR SUBSTANCE CONTENT, MUST BE Eu ROHS COMPLIANT WITHOUT EXEMPTION AND PB-FREE.
5. ALL DIMENSIONS APPLY TO Pb FREE (+) PARTS.
6. PACKAGE CODE: F142D5+2

—DRAWING NOT TO SCALE—

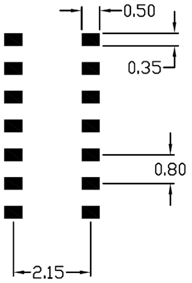
maxim
integrated

TITLE:
PACKAGE OUTLINE, 14L
OLGA, 2.8x5.6x1.2mm

APPROVAL	DOCUMENT CONTROL NO.	REV.	
	21-0880	B	1/1

Package Information (continued)

For the latest package outline information and land patterns (footprints), go to **www.maximintegrated.com/packages**. Note that a “+”, “#”, or “-” in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.



NOTES:
1. REFERENCE PKG. OUTLINE: 21-0880.
2. LAND PATTERN COMPLIES TO: IPC7351A.
3. TOLERANCE: +/- 0.02 MM.
4. ALL DIMENSIONS APPLY TO PbFREE (+) PKG. CODE ONLY
5. ALL DIMENSIONS IN MM.

-DRAWING NOT TO SCALE-

maxim integrated™

This document (including dimensions, notes & specs) is a recommendaton based on typical circuit board manufacturing parameters. Since land pattern design depend on many factors unknown to Maxim (eg. user's board manufacturing specs), user must determine suitability for use.
This document is subject to change without notice.
Contact technical support at http://www.maxim-ic.com/support for further questions.

TITLE:
PACKAGE LAND PATTERN,
[F142D5+2] OLGA

APPROVAL	DOCUMENT CONTROL NO.	REV.	
	90-0461	A	1/1

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	9/14	Initial release	—

For pricing, delivery, and ordering information, please contact Maxim Direct at 1-888-629-4642, or visit Maxim Integrated's website at www.maximintegrated.com.

ANEXO B

Oxygen Sensor (Medical) P/N : PSR-11-75-KE7

Replacement oxygen sensor with excellent stability and accuracy for use in Maxtec OM-25A, Fluke VT Mobile and Vaportherm Precision Flow.

PSR-11-75-KE1

OEM Equipment:
Fluke VT Mobile
Maxtec OM-25A
Vaportherm Precision Flow

Replacement for:
Maxtec: MAX-250
Teledyne: R29
Sensoronics: SS-250

Highlights:

Signal Output:	10-14 mV
Response Time:	13 Sec
Expected Life in Air:	60 months
Warranty:	16 months
Connector:	3” wire Plug 2 position connector TE connectivity/AMP

FDA CLEARED CE Made in USA

Advanced galvanic type oxygen sensor with excellent stability and accuracy under stringent applications. All sensors are subjected to the most extensive stability test, output in air, 30" of water column pressure test and stability at 100% oxygen. The wide range of oxygen sensors offered by CareOx, LLC. are "Made in USA"

Technical Specifications

Measuring Range	0-100%
Accuracy	+/-2% of Full Scale
Signal Output	10-14 mV
Linearity	+/-2% of Full Scale
Response T90	13 sec
Temperature Coefficient	Compensated
Operating Temp	0 to 45°C
Recommended Storage3	0 to 25°C
Shelf Life 4	6 months
Humidity Non-condensing	0-99% RH
Expected Life	60 months
Warranty 5	16 months
Electrical Conn	3” wire Plug 2 position connector TE connectivity/AMP

*** Conditions - Specification validated during design and in pursuit of improvement are subject to change without notice**

1. At constant temperature and pressure.
2. In air (20.9% oxygen) at 25°C and 1 atm.
3. Sensor may be stored up to 55°C on an intermittent basis, for example, during transportation.
4. In original Package at 25°C and 1 atm.
5. Under normal operating conditions, the sensor is warranted to be free of defects in material and workmanship for the specified period provided the sensor is properly installed and operated. The sole remedy for sensor determined to be defective by CareOx, LLC. is limited to replacing the sensor. CareOx, LLC. will not be liable for buyer’s negligence, misapplication, abuse or accident.

ANEXO C

Ultra-Small, Low-Power, 16-Bit Analog-to-Digital Converter with Internal Reference

Check for Samples: ADS1113 ADS1114 ADS1115

FEATURES

- ULTRA-SMALL QFN PACKAGE: 2mm × 1,5mm × 0,4mm
- WIDE SUPPLY RANGE: 2.0V to 5.5V
- LOW CURRENT CONSUMPTION: Continuous Mode: Only 150µA Single-Shot Mode: Auto Shut-Down
- PROGRAMMABLE DATA RATE: 8SPS to 860SPS
- INTERNAL LOW-DRIFT VOLTAGE REFERENCE
- INTERNAL OSCILLATOR
- INTERNAL PGA
- I^2C™ INTERFACE: Pin-Selectable Addresses
- FOUR SINGLE-ENDED OR TWO DIFFERENTIAL INPUTS (ADS1115)
- PROGRAMMABLE COMPARATOR (ADS1114 and ADS1115)

APPLICATIONS

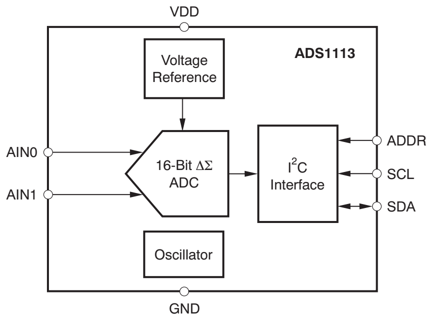
- PORTABLE INSTRUMENTATION
- CONSUMER GOODS
- BATTERY MONITORING
- TEMPERATURE MEASUREMENT
- FACTORY AUTOMATION AND PROCESS CONTROLS

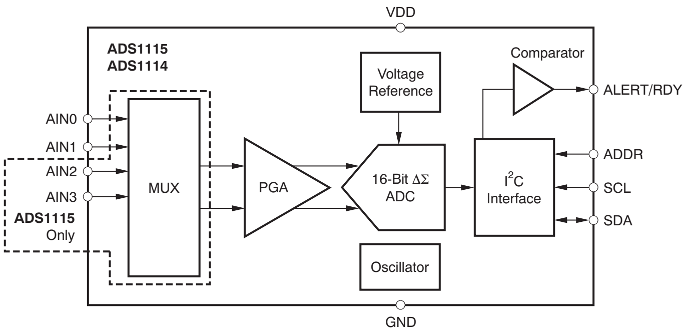
DESCRIPTION

The ADS1113, ADS1114, and ADS1115 are precision analog-to-digital converters (ADCs) with 16 bits of resolution offered in an ultra-small, leadless QFN-10 package or an MSOP-10 package. The ADS1113/4/5 are designed with precision, power, and ease of implementation in mind. The ADS1113/4/5 feature an onboard reference and oscillator. Data are transferred via an I^2C-compatible serial interface; four I^2C slave addresses can be selected. The ADS1113/4/5 operate from a single power supply ranging from 2.0V to 5.5V.

The ADS1113/4/5 can perform conversions at rates up to 860 samples per second (SPS). An onboard PGA is available on the ADS1114 and ADS1115 that offers input ranges from the supply to as low as ±256mV, allowing both large and small signals to be measured with high resolution. The ADS1115 also features an input multiplexer (MUX) that provides two differential or four single-ended inputs.

The ADS1113/4/5 operate either in continuous conversion mode or a single-shot mode that automatically powers down after a conversion and greatly reduces current consumption during idle periods. The ADS1113/4/5 are specified from –40°C to +125°C.





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.

ABSOLUTE MAXIMUM RATINGS(1)

	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/N9J	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°C to +125°C, VDD = 3.3V, and Full-Scale (FS) = ±2.048V, unless otherwise noted.
Typical values are at +25°C.

PARAMETER	TEST CONDITIONS	ADS1113, ADS1114, ADS1115			UNIT
		MIN	TYP	MAX	
ANALOG INPUT					
Full-scale input voltage(1)	$V_{IN} = (AIN_P) – (AIN_N)$		±4.096/PGA		V
Analog input voltage	AIN_P or AIN_N to GND	GND		VDD	V
Differential input impedance			See Table 2		
Common-mode input impedance	FS = ±6.144V(1)		10		MΩ
	FS = ±4.096V(1), ±2.048V		6		MΩ
	FS = ±1.024V		3		MΩ
	FS = ±0.512V, ±0.256V		100		MΩ
SYSTEM PERFORMANCE					
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	DR = 8SPS, FS = ±2.048V, best fit(2)			1	LSB
Offset error	FS = ±2.048V, differential inputs		±1	±3	LSB
	FS = ±2.048V, single-ended inputs		±3		LSB
Offset drift	FS = ±2.048V		0.005		LSB/°C
Offset power-supply rejection	FS = ±2.048V		1		LSB/V
Gain error(3)	FS = ±2.048V at 25°C		0.01	0.15	%
Gain drift(3)	FS = ±0.256V		7		ppm/°C
	FS = ±2.048V		5	40	ppm/°C
	FS = ±6.144V(1)		5		ppm/°C
Gain power-supply rejection			80		ppm/V
PGA gain match(3)	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 FS = ±0.256V		105		dB
	At dc and FS = ±2.048V		100		dB
	At dc and FS = ±6.144V(1)		90		dB
	f_{CM} = 60Hz, DR = 8SPS		105		dB
	f_{CM} = 50Hz, DR = 8SPS		105		dB
DIGITAL INPUT/OUTPUT					
Logic level					
V_{IH}		0.7VDD		5.5	V
V_{IL}		GND – 0.5		0.3VDD	V
V_{OL}	I_{OL} = 3mA	GND	0.15	0.4	V
Input leakage					
I_H	V_{IH} = 5.5V			10	μA
I_L	V_{IL} = GND	10			μA

(1) This parameter expresses the full-scale range of the ADC scaling. In no event should more than VDD + 0.3V be applied to this device.
(2) 99% of full-scale.
(3) Includes all errors from onboard PGA and reference.

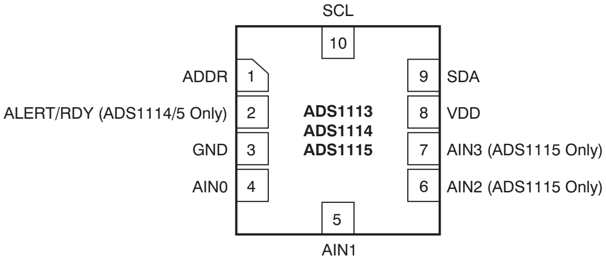
ELECTRICAL CHARACTERISTICS (continued)

All specifications at –40°C to +125°C, VDD = 3.3V, and Full-Scale (FS) = ±2.048V, unless otherwise noted. Typical values are at +25°C.

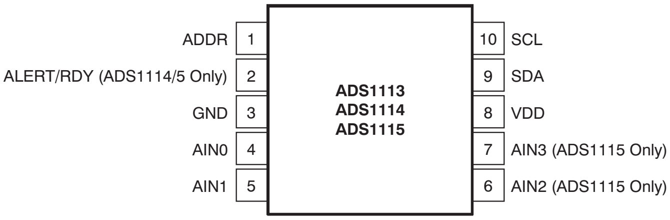
PARAMETER	TEST CONDITIONS	ADS1113, ADS1114, ADS1115			UNIT
		MIN	TYP	MAX	
POWER-SUPPLY REQUIREMENTS					
Power-supply voltage		2		5.5	V
Supply current	Power-down current at 25°C		0.5	2	μA
	Power-down current up to 125°C			5	μA
	Operating current at 25°C		150	200	μA
	Operating current up to 125°C			300	μA
Power dissipation	VDD = 5.0V		0.9		mW
	VDD = 3.3V		0.5		mW
	VDD = 2.0V		0.3		mW
TEMPERATURE					
Storage temperature		–60		+150	°C
Specified temperature		–40		+125	°C

PIN CONFIGURATIONS

RUG PACKAGE
QFN-10
(TOP VIEW)



DGS PACKAGE
MSOP-10
(TOP VIEW)



PIN DESCRIPTIONS

PIN #	DEVICE			ANALOG/ DIGITAL INPUT/ OUTPUT	DESCRIPTION
	ADS1113	ADS1114	ADS1115		
1	ADDR	ADDR	ADDR	Digital Input	I^{2}C slave address select
2	NC(1)	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.

TIMING REQUIREMENTS

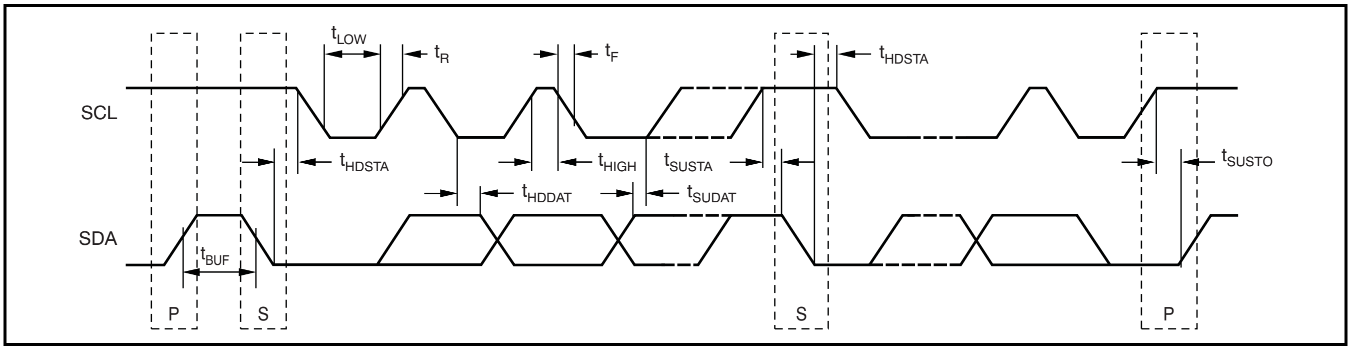


Figure 1. I^{2}C Timing Diagram

Table 1. I^{2}C Timing Definitions

PARAMETER		FAST MODE		HIGH-SPEED MODE		UNIT
		MIN	MAX	MIN	MAX	
SCL operating frequency	f_{SCL}	0.01	0.4	0.01	3.4	MHz
Bus free time between START and STOP condition	t_{BUF}	600		160		ns
Hold time after repeated START condition. After this period, the first clock is generated.	t_{HDSTA}	600		160		ns
Repeated START condition setup time	t_{SUSTA}	600		160		ns
Stop condition setup time	t_{SUSTO}	600		160		ns
Data hold time	t_{HDDAT}	0		0		ns
Data setup time	t_{SUDAT}	100		10		ns
SCL clock low period	t_{LOW}	1300		160		ns
SCL clock high period	t_{HIGH}	600		60		ns
Clock/data fall time	t_F		300		160	ns
Clock/data rise time	t_R		300		160	ns

TYPICAL CHARACTERISTICS

At T_A = +25°C and VDD = 3.3V, unless otherwise noted.

OPERATING CURRENT vs TEMPERATURE

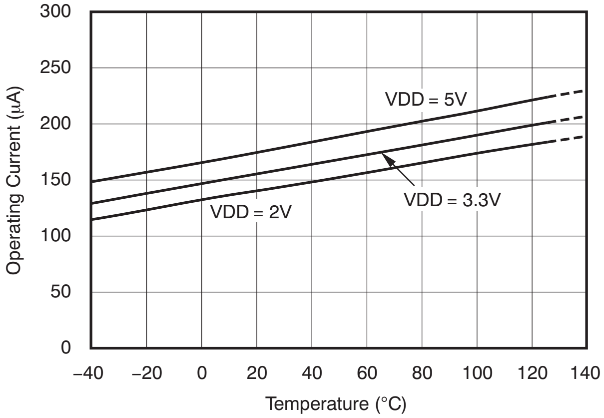


Figure 2.

SHUTDOWN CURRENT vs TEMPERATURE

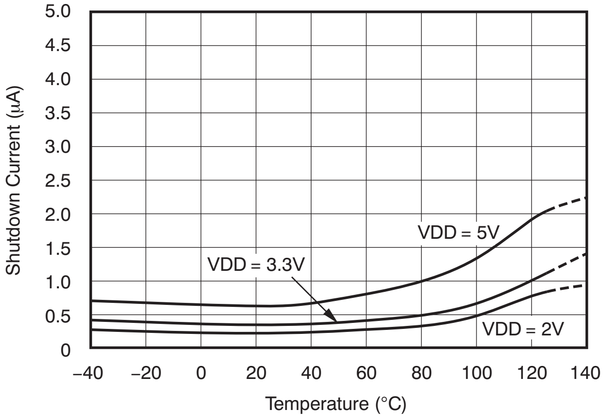


Figure 3.

SINGLE-ENDED OFFSET ERROR vs TEMPERATURE(1)

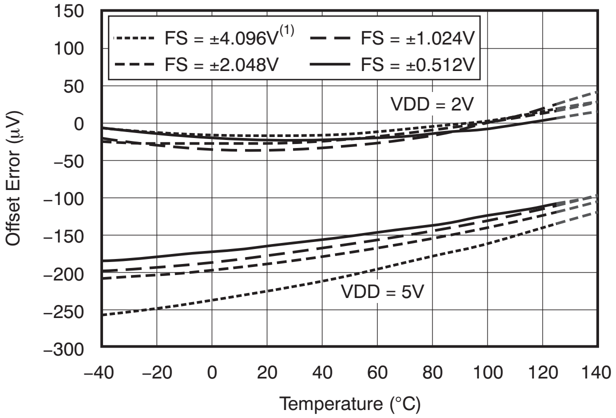


Figure 4.

DIFFERENTIAL OFFSET vs TEMPERATURE

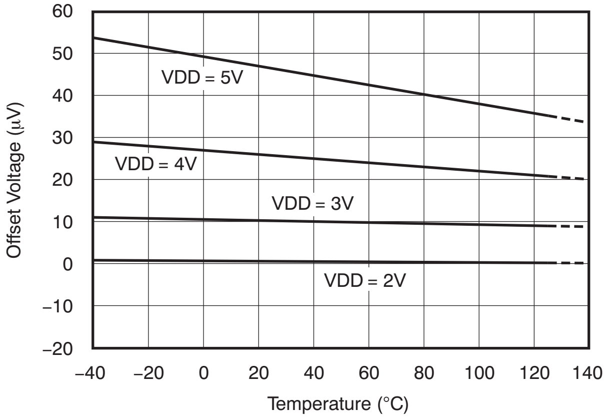


Figure 5.

GAIN ERROR vs TEMPERATURE

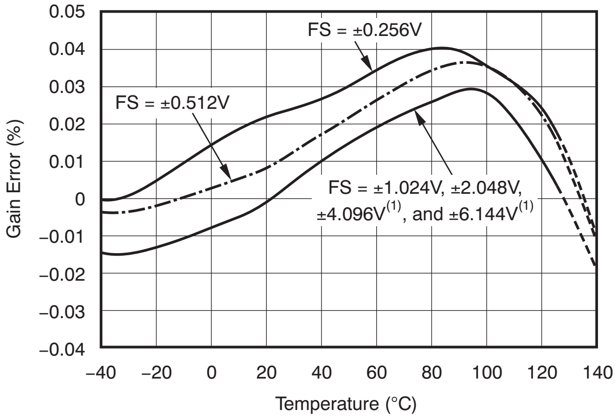


Figure 6.

GAIN ERROR vs SUPPLY

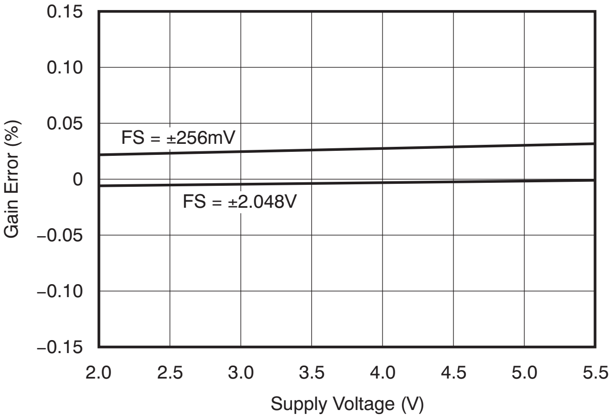


Figure 7.

---

(1) This parameter expresses the full-scale range of the ADC scaling. In no event should more than VDD + 0.3V be applied to this device.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C and VDD = 3.3V, unless otherwise noted.

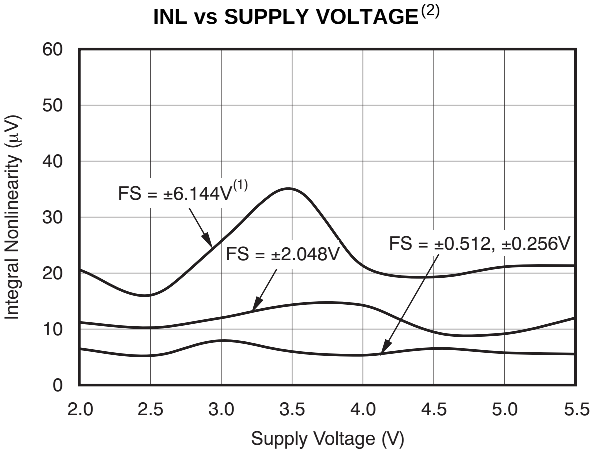


Figure 8.

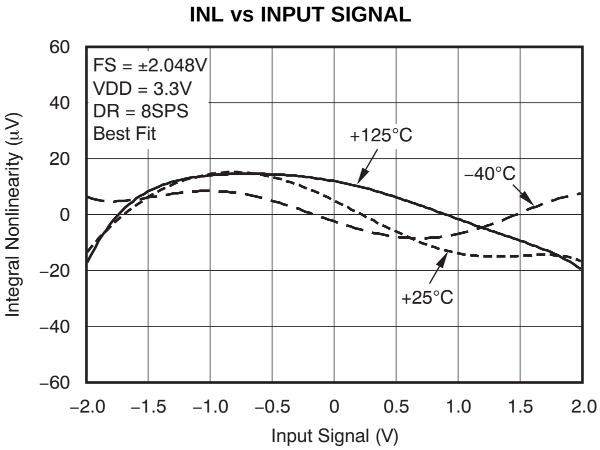


Figure 9.

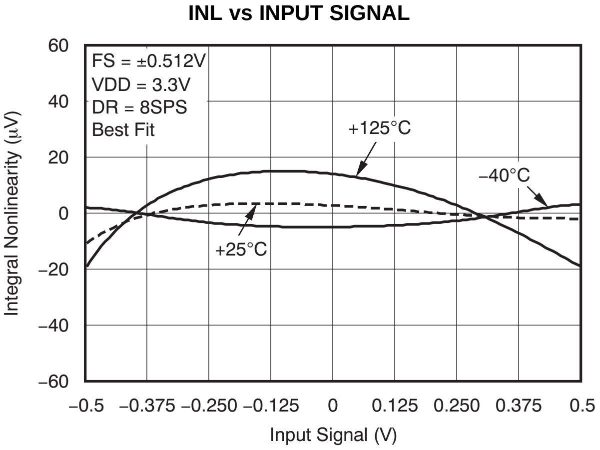


Figure 10.

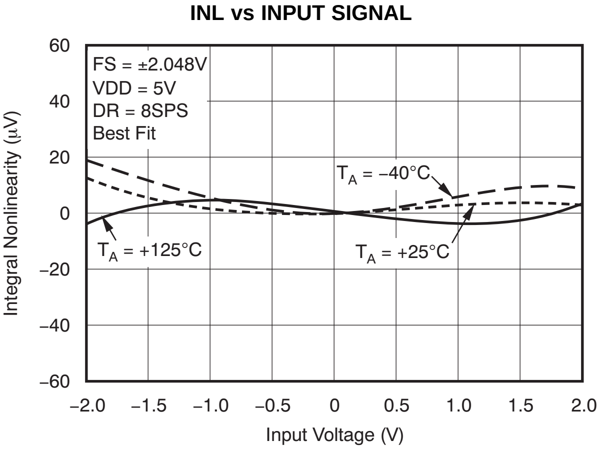


Figure 11.

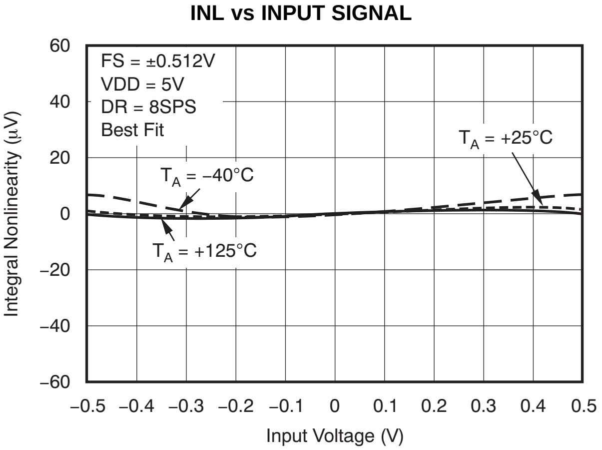


Figure 12.

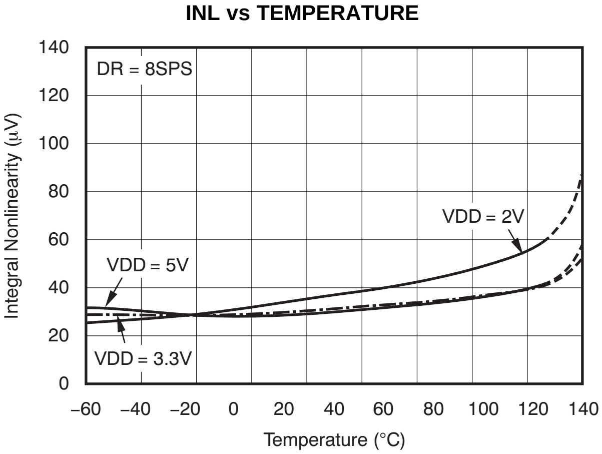


Figure 13.

---

(2) This parameter expresses the full-scale range of the ADC scaling. In no event should more than VDD + 0.3V be applied to this device.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C and VDD = 3.3V, unless otherwise noted.

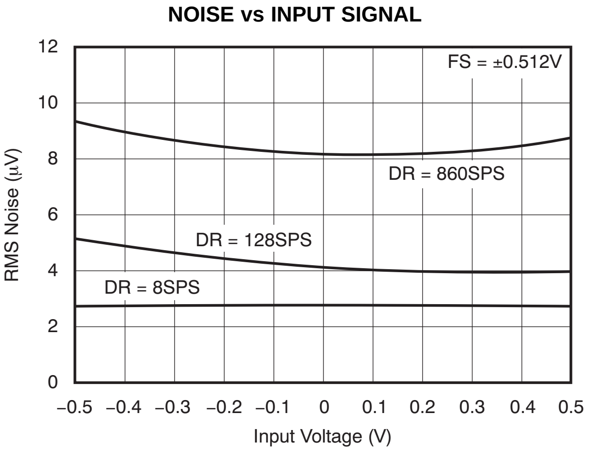


Figure 14.

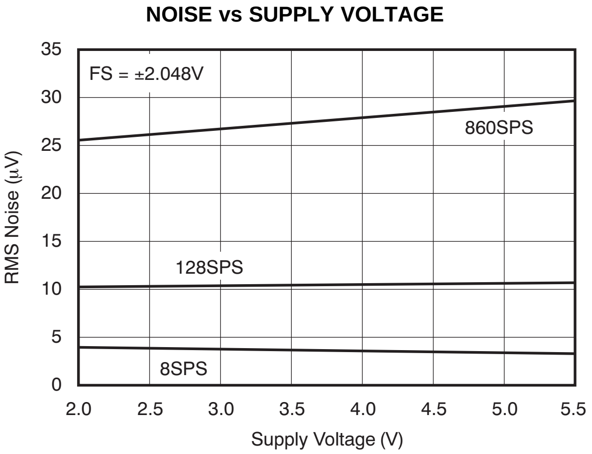


Figure 15.

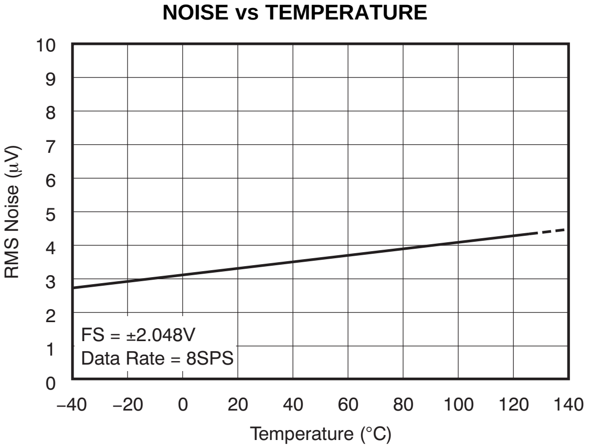


Figure 16.

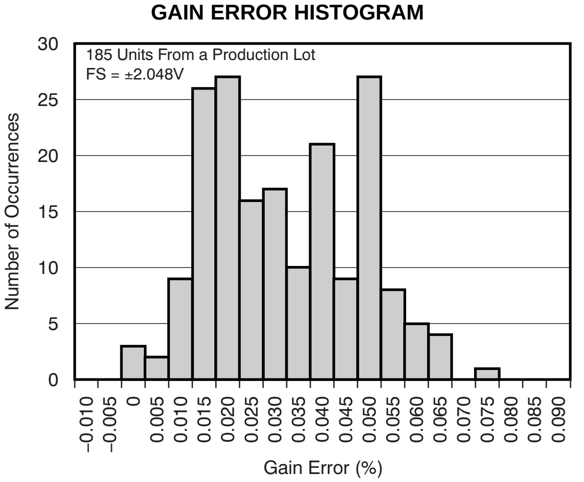


Figure 17.

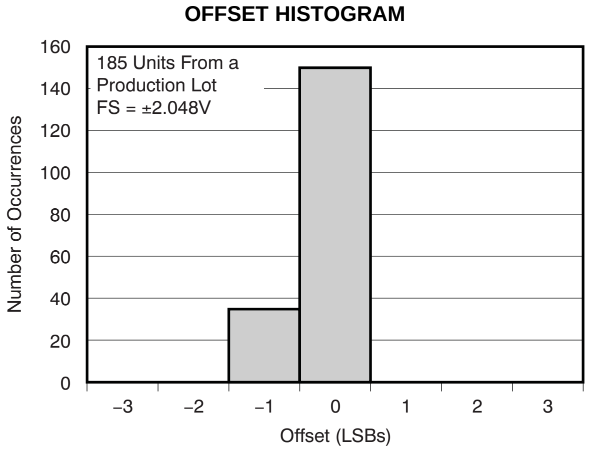


Figure 18.

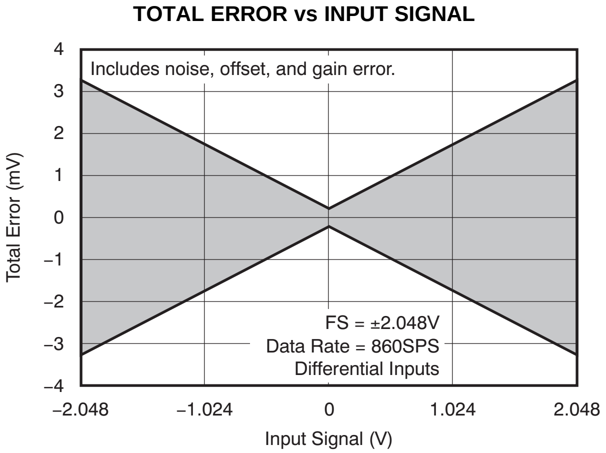


Figure 19.

TYPICAL CHARACTERISTICS (continued)

At T_A = +25°C and VDD = 3.3V, unless otherwise noted.

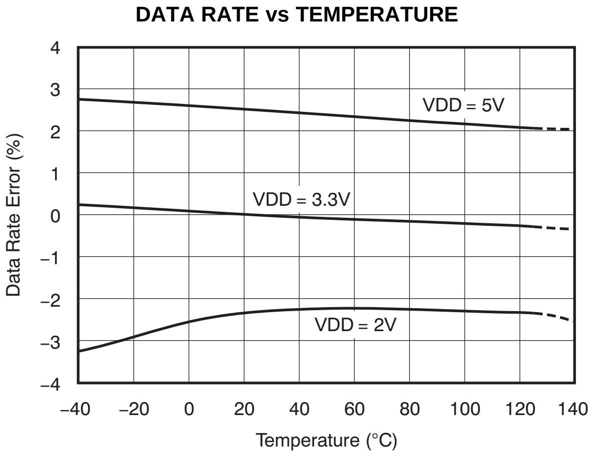


Figure 20.

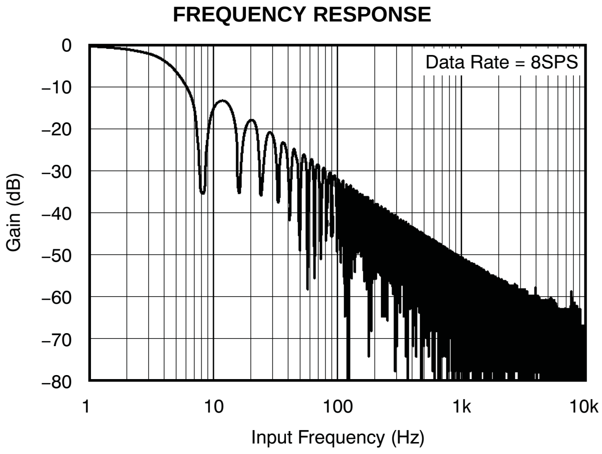


Figure 21.

OVERVIEW

The ADS1113/4/5 are very small, low-power, 16-bit, delta-sigma (ΔΣ) analog-to-digital converters (ADCs). The ADS1113/4/5 are extremely easy to configure and design into a wide variety of applications, and allow precise measurements to be obtained with very little effort. Both experienced and novice users of data converters find designing with the ADS1113/4/5 family to be intuitive and problem-free.

The ADS1113/4/5 consist of a ΔΣ analog-to-digital (A/D) core with adjustable gain (excludes the ADS1113), an internal voltage reference, a clock oscillator, and an I^{2}C interface. An additional feature available on the ADS1114/5 is a programmable digital comparator that provides an alert on a dedicated pin. All of these features are intended to reduce required external circuitry and improve performance. Figure 22 shows the ADS1115 functional block diagram.

The ADS1113/4/5 A/D core measures a differential signal, V_{IN}, that is the difference of AIN_P and AIN_N. A MUX is available on the ADS1115. This architecture results in a very strong attenuation in any common-mode signals. The converter core consists of a differential, switched-capacitor ΔΣ modulator followed by a digital filter. Input signals are compared to the internal voltage reference. The digital filter receives a high-speed bitstream from the modulator and outputs a code proportional to the input voltage.

The ADS1113/4/5 have two available conversion modes: single-shot mode and continuous conversion mode. In single-shot mode, the ADC performs one conversion of the input signal upon request and stores the value to an internal result register. The device then enters a low-power shutdown mode. This mode is intended to provide significant power savings in systems that only require periodic conversions or when there are long idle periods between conversions. In continuous conversion mode, the ADC automatically begins a conversion of the input signal as soon as the previous conversion is completed. The rate of continuous conversion is equal to the programmed data rate. Data can be read at any time and always reflect the most recent completed conversion.

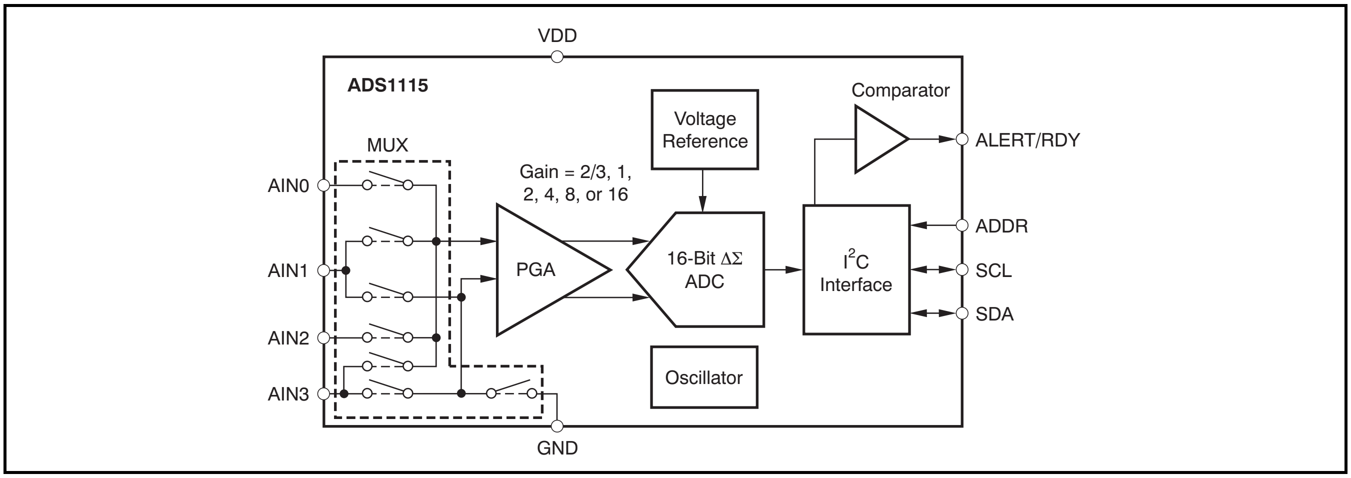


Figure 22. ADS1115 Functional Block Diagram

QUICKSTART GUIDE

This section provides a brief example of ADS1113/4/5 communications. Refer to subsequent sections of this data sheet for more detailed explanations. Hardware for this design includes: one ADS1113/4/5 configured with an I^{2}C address of 1001000; a microcontroller with an I^{2}C interface (TI recommends the MSP430F2002); discrete components such as resistors, capacitors, and serial connectors; and a 2V to 5V power supply. Figure 23 shows the basic hardware configuration.

The ADS1113/4/5 communicate with the master (microcontroller) through an I^{2}C interface. The master provides a clock signal on the SCL pin and data are transferred via the SDA pin. The ADS1113/4/5 never drive the SCL pin. For information on programming and debugging the microcontroller being used, refer to the device-specific product data sheet.

The first byte sent by the master should be the ADS1113/4/5 address followed by a bit that instructs the ADS1113/4/5 to listen for a subsequent byte. The second byte is the register pointer. Refer to Table 9 for a register map. The third and fourth bytes sent from the master are written to the register indicated in the second byte. Refer to Figure 30 and Figure 31 for read and write operation timing diagrams, respectively. All read and write transactions with the ADS1113/4/5 must be preceded by a start condition and followed by a stop condition.

For example, to write to the configuration register to set the ADS1113/4/5 to continuous conversion mode and then read the conversion result, send the following bytes in this order:

Write to Config register:

First byte: 0b10010000 (first 7-bit I^{2}C address followed by a low read/write bit)

Second byte: 0b00000001 (points to Config register)

Third byte: 0b10000100 (MSB of the Config register to be written)

Fourth byte: 0b10000011 (LSB of the Config register to be written)

Write to Pointer register:

First byte: 0b10010000 (first 7-bit I^{2}C address followed by a low read/write bit)

Second byte: 0b00000000 (points to Conversion register)

Read Conversion register:

First byte: 0b10010001 (first 7-bit I^{2}C address followed by a high read/write bit)

Second byte: the ADS1113/4/5 response with the MSB of the Conversion register

Third byte: the ADS1113/4/5 response with the LSB of the Conversion register

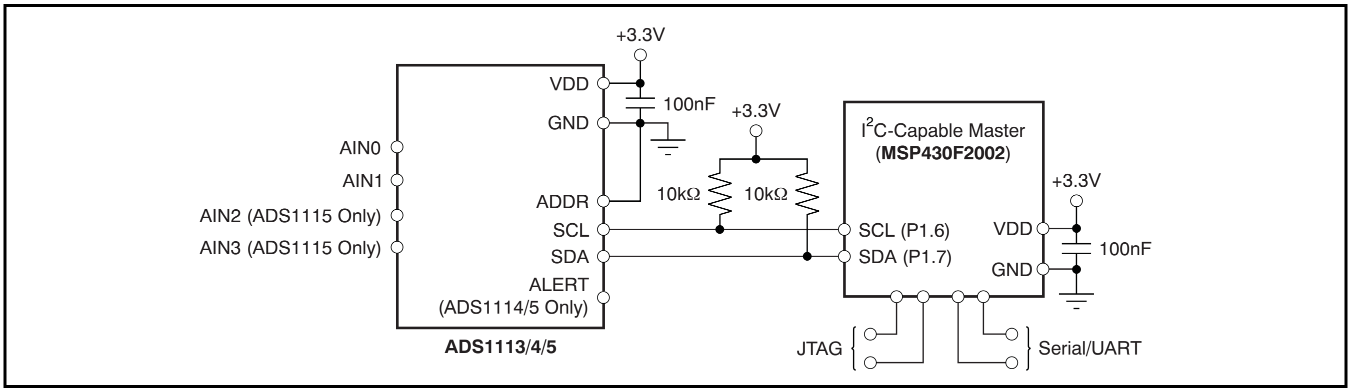


Figure 23. Basic Hardware Configuration

MULTIPLEXER

The ADS1115 contains an input multiplexer, as shown in Figure 24. Either four single-ended or two differential signals can be measured. Additionally, AIN0 and AIN1 may be measured differentially to AIN3. The multiplexer is configured by three bits in the Config register. When single-ended signals are measured, the negative input of the ADC is internally connected to GND by a switch within the multiplexer.

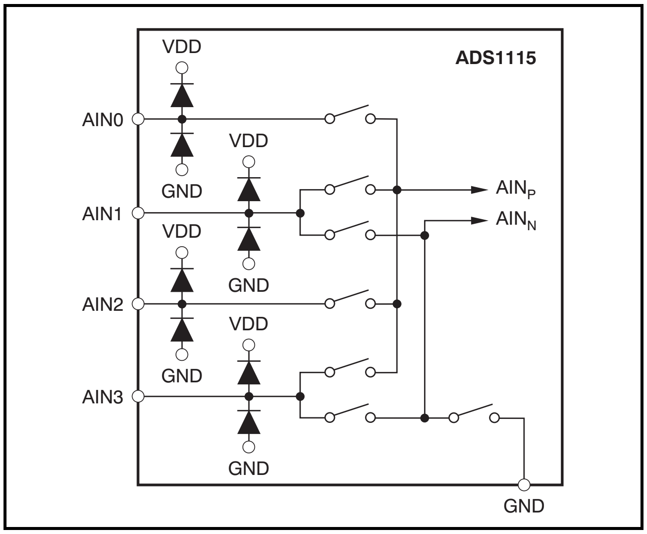


Figure 24. ADS1115 MUX

The ADS1113 and ADS1114 do not have a multiplexer. Either one differential or one single-ended signal may be measured with these devices. For single-ended measurements, connect the AIN1 pin to GND. Note that in subsequent sections of this data sheet, AIN_P refers to AIN0 and AIN_N refers to AIN1 for the ADS1113 and ADS1114.

When measuring single-ended inputs it is important to note that the negative range of the output codes are not used. These codes are for measuring negative differential signals such as (AIN_P – AIN_N) < 0. ESD diodes to VDD and GND protect the inputs on all three devices (ADS1113, ADS1114, and ADS1115). To prevent the ESD diodes from turning on, the absolute voltage on any input must stay within the following range:

GND – 0.3V < AINx < VDD + 0.3V

If it is possible that the voltages on the input pins may violate these conditions, external Schottky clamp diodes and/or series resistors may be required to limit the input current to safe values (see the Absolute Maximum Ratings table).

Also, overdriving one unused input on the ADS1115 may affect conversions taking place on other input pins. If overdrive on unused inputs is possible, again it is recommended to clamp the signal with external Schottky diodes.

ANALOG INPUTS

The ADS1113/4/5 use a switched-capacitor input stage where capacitors are continuously charged and then discharged to measure the voltage between AIN_P and AIN_N. The capacitors used are small, and to external circuitry the average loading appears resistive. This structure is shown in Figure 26. The resistance is set by the capacitor values and the rate at which they are switched. Figure 25 shows the on/off setting of the switches illustrated in Figure 26. During the sampling phase, S_1 switches are closed. This event charges C_{A1} to AIN_P, C_{A2} to AIN_N, and C_B to (AIN_P – AIN_N). During the discharge phase, S_1 is first opened and then S_2 is closed. Both C_{A1} and C_{A2} then discharge to approximately 0.7V and C_B discharges to 0V. This charging draws a very small transient current from the source driving the ADS1113/4/5 analog inputs. The average value of this current can be used to calculate the effective impedance (R_{eff}) where $R_{eff} = V_{IN}/I_{AVERAGE}$.

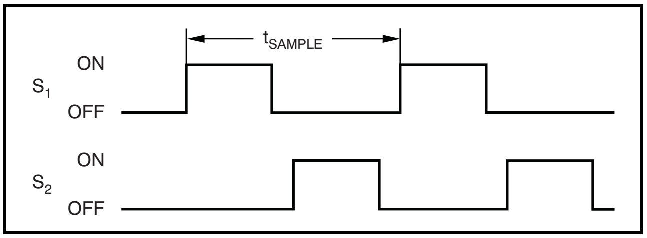


Figure 25. S_1 and S_2 Switch Timing for Figure 26

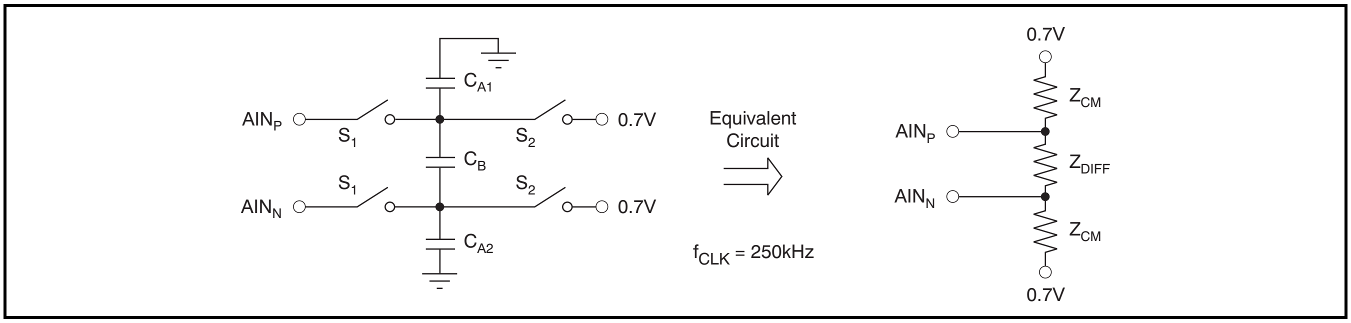


Figure 26. Simplified Analog Input Circuit

The common-mode input impedance is measured by applying a common-mode signal to shorted AIN$_P$ and AIN$_N$ inputs and measuring the average current consumed by each pin. The common-mode input impedance changes depending on the PGA gain setting, but is approximately 6MΩ for the default PGA gain setting. In Figure 26, the common-mode input impedance is Z$_{CM}$.

The differential input impedance is measured by applying a differential signal to AIN$_P$ and AIN$_N$ inputs where one input is held at 0.7V. The current that flows through the pin connected to 0.7V is the differential current and scales with the PGA gain setting. In Figure 26, the differential input impedance is Z$_{DIFF}$. Table 2 describes the typical differential input impedance.

Table 2. Differential Input Impedance

FS (V)	DIFFERENTIAL INPUT IMPEDANCE
±6.144V[1]	22MΩ
±4.096V[1]	15MΩ
±2.048V	4.9MΩ
±1.024V	2.4MΩ
±0.512V	710kΩ
±0.256V	710kΩ

1. This parameter expresses the full-scale range of the ADC scaling. In no event should more than VDD + 0.3V be applied to this device.

The typical value of the input impedance cannot be neglected. Unless the input source has a low impedance, the ADS1113/4/5 input impedance may affect the measurement accuracy. For sources with high output impedance, buffering may be necessary. Active buffers introduce noise, and also introduce offset and gain errors. All of these factors should be considered in high-accuracy applications.

Because the clock oscillator frequency drifts slightly with temperature, the input impedances also drift. For many applications, this input impedance drift can be ignored, and the values given in Table 2 for typical input impedance are valid.

FULL-SCALE INPUT

A programmable gain amplifier (PGA) is implemented before the ΔΣ core of the ADS1114/5. The PGA can be set to gains of 2/3, 1, 2, 4, 8, and 16. Table 3 shows the corresponding full-scale (FS) ranges. The PGA is configured by three bits in the Config register. The ADS1113 has a fixed full-scale input range of ±2.048V. The PGA = 2/3 setting allows input measurement to extend up to the supply voltage when VDD is larger than 4V. Note though that in this case (as well as for PGA = 1 and VDD < 4V), it is not possible to reach a full-scale output code on the ADC. Analog input voltages may never exceed the analog input voltage limits given in the Electrical Characteristics table.

Table 3. PGA Gain Full-Scale Range

PGA SETTING	FS (V)
2/3	±6.144V[1]
1	±4.096V[1]
2	±2.048V
4	±1.024V
8	±0.512V
16	±0.256V

1. This parameter expresses the full-scale range of the ADC scaling. In no event should more than VDD + 0.3V be applied to this device.

DATA FORMAT

The ADS1113/4/5 provide 16 bits of data in binary twos complement format. The positive full-scale input produces an output code of 7FFFh and the negative full-scale input produces an output code of 8000h. The output clips at these codes for signals that exceed full-scale. Table 4 summarizes the ideal output codes for different input signals. Figure 27 shows code transitions versus input voltage.

Table 4. Input Signal versus Ideal Output Code

INPUT SIGNAL, V_{IN} ($AIN_P – AIN_N$)	IDEAL OUTPUT CODE[1]
$\geq$ FS $(2^{15} – 1)/2^{15}$	7FFFh
$+FS/2^{15}$	0001h
0	0
$–FS/2^{15}$	FFFFh
$\leq$ –FS	8000h

1. Excludes the effects of noise, INL, offset, and gain errors.

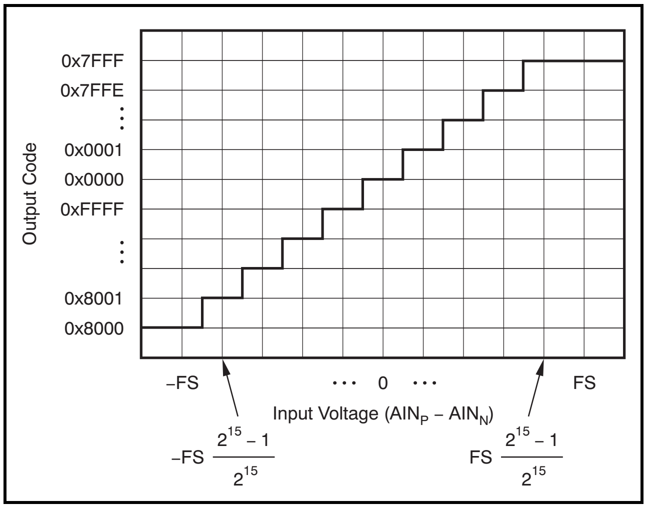


Figure 27. ADS1113/4/5 Code Transition Diagram

ALIASING

As with any data converter, if the input signal contains frequencies greater than half the data rate, aliasing occurs. To prevent aliasing, the input signal must be bandlimited. Some signals are inherently bandlimited. For example, the output of a thermocouple, which has a limited rate of change. Nevertheless, they can contain noise and interference components. These components can fold back into the sampling band in the same way as with any other signal.

The ADS1113/4/5 digital filter provides some attenuation of high-frequency noise, but the digital Sinc filter frequency response cannot completely replace an anti-aliasing filter. For a few applications, some external filtering may be needed; in such instances, a simple RC filter is adequate.

When designing an input filter circuit, be sure to take into account the interaction between the filter network and the input impedance of the ADS1113/4/5.

OPERATING MODES

The ADS1113/4/5 operate in one of two modes: continuous conversion or single-shot. In continuous conversion mode, the ADS1113/4/5 continuously perform conversions. Once a conversion has been completed, the ADS1113/4/5 place the result in the Conversion register and immediately begins another conversion. In single-shot mode, the ADS1113/4/5 wait until the OS bit is set high. Once asserted, the bit is set to '0', indicating that a conversion is currently in progress. Once conversion data are ready, the OS bit reasserts and the device powers down. Writing a '1' to the OS bit during a conversion has no effect.

RESET AND POWER-UP

When the ADS1113/4/5 powers up, a reset is performed. As part of the reset process, the ADS1113/4/5 set all of the bits in the Config register to the respective default settings.

The ADS1113/4/5 respond to the I^2C general call reset command. When the ADS1113/4/5 receive a general call reset, an internal reset is performed as if the device had been powered on.

DUTY CYCLING FOR LOW POWER

For many applications, the improved performance at low data rates may not be required. For these applications, the ADS1113/4/5 support duty cycling that can yield significant power savings by periodically requesting high data rate readings at an effectively lower data rate. For example, an ADS1113/4/5 in power-down mode with a data rate set to 860SPS could be operated by a microcontroller that instructs a single-shot conversion every 125ms (8SPS). Because a conversion at 860SPS only requires about 1.2ms, the ADS1113/4/5 enter power-down mode for the remaining 123.8ms. In this configuration, the ADS1113/4/5 consume about 1/100th the power of the ADS1113/4/5 operated in continuous conversion mode. The rate of duty cycling is completely arbitrary and is defined by the master controller. The ADS1113/4/5 offer lower data rates that do not implement duty cycling and offer improved noise performance if it is needed.

COMPARATOR (ADS1114/15 ONLY)

The ADS1114/5 are each equipped with a customizable comparator that can issue an alert on the ALERT/RDY pin. This feature can significantly reduce external circuitry for many applications. The comparator can be implemented as either a traditional comparator or a window comparator via the COMP_MODE bit in the Config register. When implemented as a traditional comparator, the ALERT/RDY pin asserts (active low by default) when conversion data exceed the limit set in the high threshold register. The comparator then deasserts when the input signal falls below the low threshold register value. In window comparator mode, the ALERT/RDY pin asserts if conversion data exceed the high threshold register or fall below the low threshold register.

In either window or traditional comparator mode, the comparator can be configured to latch once asserted by the COMP_LAT bit in the Config register. This setting causes the assertion to remain even if the input signal is not beyond the bounds of the threshold registers. This latched assertion can be cleared by issuing an SMBus alert response or by reading the Conversion register. The COMP_POL bit in the Config register configures the ALERT/RDY pin as active high or active low. Operational diagrams for the comparator modes are shown in Figure 28 and Figure 29.

The comparator can be configured to activate the ALERT/RDY pin after a set number of successive readings exceed the threshold. The comparator can be configured to wait for one, two, or four readings beyond the threshold before activating the ALERT/RDY pin by changing the COMP_QUE bits in the Config register. The COMP_QUE bits can also disable the comparator function.

CONVERSION READY PIN (ADS1114/5 ONLY)

The ALERT/RDY pin can also be configured as a conversion ready pin. This mode of operation can be realized if the MSB of the high threshold register is set to '1' and the MSB of the low threshold register is set to '0'. The COMP_POL bit continues to function and the COMP_QUE bits can disable the pin; however, the COMP_MODE and COMP_LAT bits no longer control any function. When configured as a conversion ready pin, ALERT/RDY continues to require a pull-up resistor. When in continuous conversion mode, the ADS1113/4/5 provide a brief (~8µs) pulse on the ALERT/RDY pin at the end of each conversion. When in single-shot shutdown mode, the ALERT/RDY pin asserts low at the end of a conversion if the COMP_POL bit is set to '0'.

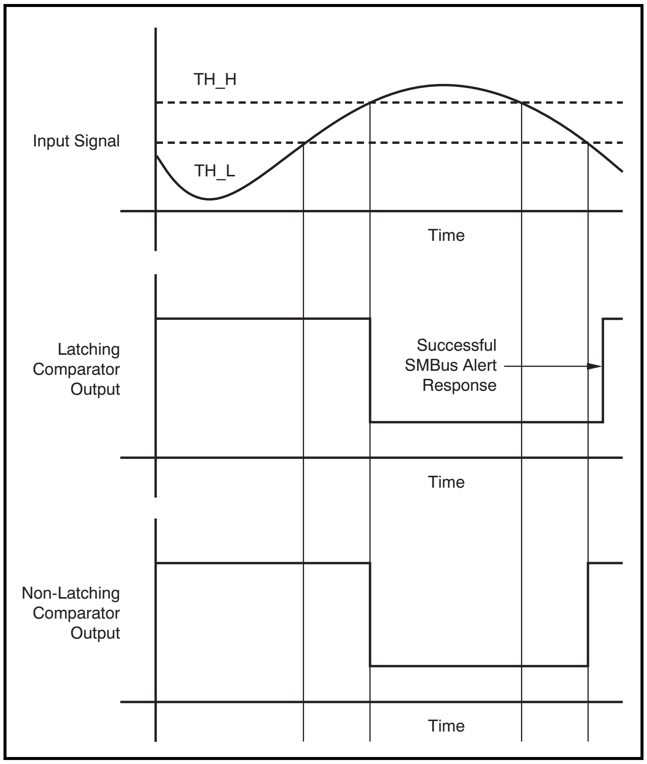


Figure 28. Alert Pin Timing Diagram When Configured as a Traditional Comparator

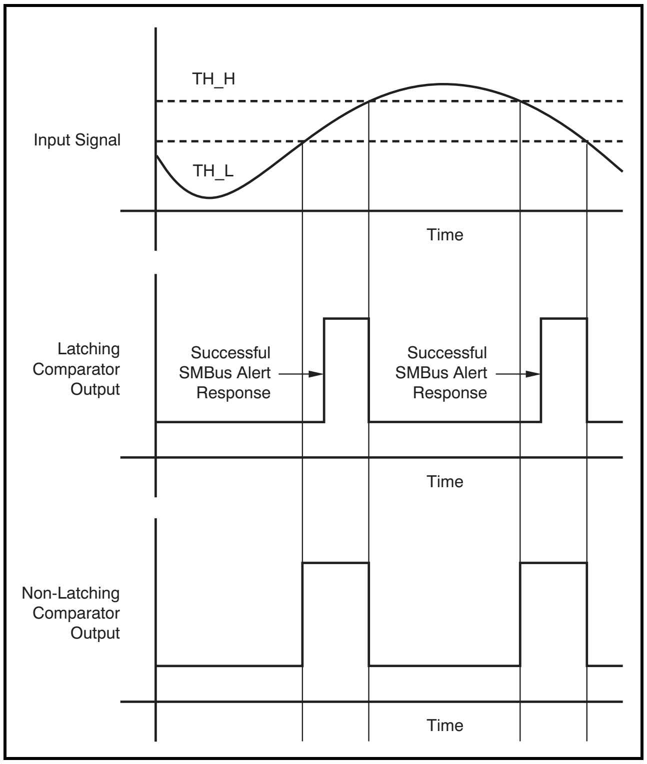


Figure 29. Alert Pin Timing Diagram When Configured as a Window Comparator

SMBus ALERT RESPONSE

When configured in latching mode (COMP_LAT = '1' in the Config register), the ALERT/RDY pin can be implemented with an SMBus alert. The pin asserts if the comparator detects a conversion that exceeds an upper or lower threshold. This interrupt is latched and can be cleared only by reading conversion data, or by issuing a successful SMBus alert response and reading the asserting device I^{2}C address. If conversion data exceed the upper or lower thresholds after being cleared, the pin reasserts. This assertion does not affect conversions that are already in progress. The ALERT/RDY pin, as with the SDA pin, is an open-drain pin. This architecture allows several devices to share the same interface bus. When disabled, the pin holds a high state so that it does not interfere with other devices on the same bus line.

When the master senses that the ALERT/RDY pin has latched, it issues an SMBus alert command (00011001) to the I^{2}C bus. Any ADS1114/5 data converters on the I^{2}C bus with the ALERT/RDY pins asserted respond to the command with the slave address. In the event that two or more ADS1114/5 data converters present on the bus assert the latched ALERT/RDY pin, arbitration during the address response portion of the SMBus alert decides which device clears its assertion. The device with the lowest I^{2}C address always wins arbitration. If a device loses arbitration, it does not clear the comparator output pin assertion. The master then repeats the SMBus alert response until all devices have had the respective assertions cleared. In window comparator mode, the SMBus alert status bit indicates a '1' if signals exceed the high threshold and a '0' if signals exceed the low threshold.

I^{2}C INTERFACE

The ADS1113/4/5 communicate through an I^{2}C interface. I^{2}C is a two-wire open-drain interface that supports multiple devices and masters on a single bus. Devices on the I^{2}C bus only drive the bus lines low by connecting them to ground; they never drive the bus lines high. Instead, the bus wires are pulled high by pull-up resistors, so the bus wires are high when no device is driving them low. This way, two devices cannot conflict; if two devices drive the bus simultaneously, there is no driver contention.

Communication on the I^{2}C bus always takes place between two devices, one acting as the master and the other as the slave. Both masters and slaves can read and write, but slaves can only do so under the direction of the master. Some I^{2}C devices can act as masters or slaves, but the ADS1113/4/5 can only act as slave devices.

An I^{2}C bus consists of two lines, SDA and SCL. SDA carries data; SCL provides the clock. All data are transmitted across the I^{2}C bus in groups of eight bits. To send a bit on the I^{2}C bus, the SDA line is driven to the appropriate level while SCL is low (a low on SDA indicates the bit is zero; a high indicates the bit is one). Once the SDA line settles, the SCL line is brought high, then low. This pulse on SCL clocks the SDA bit into the receiver shift register. If the I^{2}C bus is held idle for more than 25ms, the bus times out.

The I^{2}C bus is bidirectional: the SDA line is used for both transmitting and receiving data. When the master reads from a slave, the slave drives the data line; when the master sends to a slave, the master drives the data line. The master always drives the clock line. The ADS1113/4/5 never drive SCL, because they cannot act as a master. On the ADS1113/4/5, SCL is an input only.

Most of the time the bus is idle; no communication occurs, and both lines are high. When communication is taking place, the bus is active. Only master devices can start a communication and initiate a START condition on the bus. Normally, the data line is only allowed to change state while the clock line is low. If the data line changes state while the clock line is high, it is either a START condition or a STOP condition. A START condition occurs when the clock line is high and the data line goes from high to low. A STOP condition occurs when the clock line is high and the data line goes from low to high.

After the master issues a START condition, it sends a byte that indicates which slave device it wants to communicate with. This byte is called the *address byte*. Each device on an I^{2}C bus has a unique 7-bit address to which it responds. The master sends an address in the address byte, together with a bit that indicates whether it wishes to read from or write to the slave device.

Every byte transmitted on the I^{2}C bus, whether it is address or data, is acknowledged with an *acknowledge* bit. When the master has finished sending a byte (eight data bits) to a slave, it stops driving SDA and waits for the slave to acknowledge the byte. The slave acknowledges the byte by pulling SDA low. The master then sends a clock pulse to clock the acknowledge bit. Similarly, when the master has finished reading a byte, it pulls SDA low to acknowledge this to the slave. It then sends a clock pulse to clock the bit. (The master always drives the clock line.)

A *not-acknowledge* is performed by simply leaving SDA high during an acknowledge cycle. If a device is not present on the bus, and the master attempts to address it, it receives a not-acknowledge because no device is present at that address to pull the line low.

When the master has finished communicating with a slave, it may issue a STOP condition. When a STOP condition is issued, the bus becomes idle again. The master may also issue another START condition. When a START condition is issued while the bus is active, it is called a repeated START condition.

See the *Timing Requirements* section for a timing diagram showing the ADS1113/4/5 I^2C transaction.

I^2C ADDRESS SELECTION

The ADS1113/4/5 have one address pin, ADDR, that sets the I^2C address. This pin can be connected to ground, VDD, SDA, or SCL, allowing four addresses to be selected with one pin as shown in Table 5. The state of the address pin ADDR is sampled continuously.

Table 5. ADDR Pin Connection and Corresponding Slave Address

ADDR PIN	SLAVE ADDRESS
Ground	1001000
VDD	1001001
SDA	1001010
SCL	1001011

I^2C GENERAL CALL

The ADS1113/4/5 respond to the I^2C general call address (0000000) if the eighth bit is '0'. The devices acknowledge the general call address and respond to commands in the second byte. If the second byte is 00000110 (06h), the ADS1113/4/5 reset the internal registers and enter power-down mode.

I^2C SPEED MODES

The I^2C bus operates at one of three speeds. Standard mode allows a clock frequency of up to 100kHz; fast mode permits a clock frequency of up to 400kHz; and high-speed mode (also called Hs mode) allows a clock frequency of up to 3.4MHz. The ADS1113/4/5 are fully compatible with all three modes.

No special action is required to use the ADS1113/4/5 in standard or fast mode, but high-speed mode must be activated. To activate high-speed mode, send a special address byte of *00001xxx* following the START condition, where *xxx* are bits unique to the Hs-capable master. This byte is called the Hs master code. (Note that this is different from normal address bytes; the eighth bit does not indicate read/write status.) The ADS1113/4/5 do not acknowledge this byte; the I^2C specification prohibits acknowledgment of the Hs master code. Upon receiving a master code, the ADS1113/4/5 switch on Hs mode filters, and communicate at up to 3.4MHz. The ADS1113/4/5 switch out of Hs mode with the next STOP condition.

For more information on high-speed mode, consult the I^2C specification.

SLAVE MODE OPERATIONS

The ADS1113/4/5 can act as either slave receivers or slave transmitters. As a slave device, the ADS1113/4/5 cannot drive the SCL line.

Receive Mode:

In slave receive mode the first byte transmitted from the master to the slave is the address with the $R/\overline{W}$ bit low. This byte allows the slave to be written to. The next byte transmitted by the master is the register pointer byte. The ADS1113/4/5 then acknowledge receipt of the register pointer byte. The next two bytes are written to the address given by the register pointer. The ADS1113/4/5 acknowledge each byte sent. Register bytes are sent with the most significant byte first, followed by the least significant byte.

Transmit Mode:

In slave transmit mode, the first byte transmitted by the master is the 7-bit slave address followed by the high $R/\overline{W}$ bit. This byte places the slave into transmit mode and indicates that the ADS1113/4/5 are being read from. The next byte transmitted by the slave is the most significant byte of the register that is indicated by the register pointer. This byte is followed by an acknowledgment from the master. The remaining least significant byte is then sent by the slave and is followed by an acknowledgment from the master. The master may terminate transmission after any byte by not acknowledging or issuing a START or STOP condition.

WRITING/READING THE REGISTERS

To access a specific register from the ADS1113/4/5, the master must first write an appropriate value to the Pointer register. The Pointer register is written directly after the slave address byte, low $R/\overline{W}$ bit, and a successful slave acknowledgment. After the Pointer register is written, the slave acknowledges and the master issues a STOP or a repeated START condition.

When reading from the ADS1113/4/5, the previous value written to the Pointer register determines the register that is read from. To change which register is read, a new value must be written to the Pointer register. To write a new value to the Pointer register, the master issues a slave address byte with the R/$\overline{\text{W}}$ bit low, followed by the Pointer register byte. No additional data need to be transmitted, and a STOP condition can be issued by the master. The master may now issue a START condition and send the slave address byte with the R/$\overline{\text{W}}$ bit high to begin the read. Table 10 details this sequence. If repeated reads from the same register are desired, there is no need to continually send Pointer register bytes, because the ADS1113/4/5 store the value of the Pointer register until it is modified by a write operation. However, every write operation requires the Pointer register to be written.

REGISTERS

The ADS1113/4/5 have four registers that are accessible via the I^{2}C port. The Conversion register contains the result of the last conversion. The Config register allows the user to change the ADS1113/4/5 operating modes and query the status of the devices. Two registers, Lo_thresh and Hi_thresh, set the threshold values used for the comparator function.

POINTER REGISTER

The four registers are accessed by writing to the Pointer register byte; see Figure 30. Table 6 and Table 7 indicate the Pointer register byte map.

Table 6. Register Address

BIT 1	BIT 0	REGISTER
0	0	Conversion register
0	1	Config register
1	0	Lo_thresh register
1	1	Hi_thresh register

CONVERSION REGISTER

The 16-bit register contains the result of the last conversion in binary twos complement format. Following reset or power-up, the Conversion register is cleared to '0', and remains '0' until the first conversion is completed.

The register format is shown in Table 8.

CONFIG REGISTER

The 16-bit register can be used to control the ADS1113/4/5 operating mode, input selection, data rate, PGA settings, and comparator modes. The register format is shown in Table 9.

Table 7. Pointer Register Byte (Write-Only)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
0	0	0	0	0	0	Register address	

Table 8. Conversion Register (Read-Only)

BIT	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NAME	D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0

Table 9. Config Register (Read/Write)

BIT	15	14	13	12	11	10	9	8
NAME	OS	MUX2	MUX1	MUX0	PGA2	PGA1	PGA0	MODE
BIT	7	6	5	4	3	2	1	0
NAME	DR2	DR1	DR0	COMP_MODE	COMP_POL	COMP_LAT	COMP_QUE1	COMP_QUE0

Default = 8583h.

Bit [15] **OS: Operational status/single-shot conversion start**

This bit determines the operational status of the device.
This bit can only be written when in power-down mode.

For a write status:
0 : No effect
1 : Begin a single conversion (when in power-down mode)
For a read status:
0 : Device is currently performing a conversion
1 : Device is not currently performing a conversion

Bits [14:12]	**MUX[2:0]: Input multiplexer configuration (ADS1115 only)** These bits configure the input multiplexer. They serve no function on the ADS1113/4. 000 : AIN_P = AIN0 and AIN_N = AIN1 (default) 001 : AIN_P = AIN0 and AIN_N = AIN3 010 : AIN_P = AIN1 and AIN_N = AIN3 011 : AIN_P = AIN2 and AIN_N = AIN3 100 : AIN_P = AIN0 and AIN_N = GND 101 : AIN_P = AIN1 and AIN_N = GND 110 : AIN_P = AIN2 and AIN_N = GND 111 : AIN_P = AIN3 and AIN_N = GND
Bits [11:9]	**PGA[2:0]: Programmable gain amplifier configuration (ADS1114 and ADS1115 only)** These bits configure the programmable gain amplifier. They serve no function on the ADS1113. 000 : FS = ±6.144V(1) 001 : FS = ±4.096V(1) 010 : FS = ±2.048V (default) 011 : FS = ±1.024V 100 : FS = ±0.512V 101 : FS = ±0.256V 110 : FS = ±0.256V 111 : FS = ±0.256V
Bit [8]	**MODE: Device operating mode** This bit controls the current operational mode of the ADS1113/4/5. 0 : Continuous conversion mode 1 : Power-down single-shot mode (default)
Bits [7:5]	**DR[2:0]: Data rate** These bits control the data rate setting. 000 : 8SPS 001 : 16SPS 010 : 32SPS 011 : 64SPS 100 : 128SPS (default) 101 : 250SPS 110 : 475SPS 111 : 860SPS
Bit [4]	**COMP_MODE: Comparator mode (ADS1114 and ADS1115 only)** This bit controls the comparator mode of operation. It changes whether the comparator is implemented as a traditional comparator (COMP_MODE = '0') or as a window comparator (COMP_MODE = '1'). It serves no function on the ADS1113. 0 : Traditional comparator with hysteresis (default) 1 : Window comparator
Bit [3]	**COMP_POL: Comparator polarity (ADS1114 and ADS1115 only)** This bit controls the polarity of the ALERT/RDY pin. When COMP_POL = '0' the comparator output is active low. When COMP_POL='1' the ALERT/RDY pin is active high. It serves no function on the ADS1113. 0 : Active low (default) 1 : Active high
Bit [2]	**COMP_LAT: Latching comparator (ADS1114 and ADS1115 only)** This bit controls whether the ALERT/RDY pin latches once asserted or clears once conversions are within the margin of the upper and lower threshold values. When COMP_LAT = '0', the ALERT/RDY pin does not latch when asserted. When COMP_LAT = '1', the asserted ALERT/RDY pin remains latched until conversion data are read by the master or an appropriate SMBus alert response is sent by the master, the device responds with its address, and it is the lowest address currently asserting the ALERT/RDY bus line. This bit serves no function on the ADS1113. 0 : Non-latching comparator (default) 1 : Latching comparator
Bits [1:0]	**COMP_QUE: Comparator queue and disable (ADS1114 and ADS1115 only)** These bits perform two functions. When set to '11', they disable the comparator function and put the ALERT/RDY pin into a high state. When set to any other value, they control the number of successive conversions exceeding the upper or lower thresholds required before asserting the ALERT/RDY pin. They serve no function on the ADS1113. 00 : Assert after one conversion 01 : Assert after two conversions 10 : Assert after four conversions 11 : Disable comparator (default)

(1) This parameter expresses the full-scale range of the ADC scaling. In no event should more than VDD + 0.3V be applied to this device.

Lo_thresh AND Hi_thresh REGISTERS

The upper and lower threshold values used by the comparator are stored in two 16-bit registers. These registers store values in the same format that the output register displays values; that is, they are stored in twos complement format. Because it is implemented as a digital comparator, special attention should be taken to readjust values whenever PGA settings are changed.

A secondary conversion ready function of the comparator output pin can be realized by setting the Hi_thresh register MSB to '1' and the Lo_thresh register MSB to ‘0’. However, in all other cases, the Hi_thresh register must be larger than the Lo_thresh register. The threshold register formats are shown in Table 10. When set to RDY mode, the ALERT/RDY pin outputs the OS bit when in single-shot mode and pulses when in continuous conversion mode.

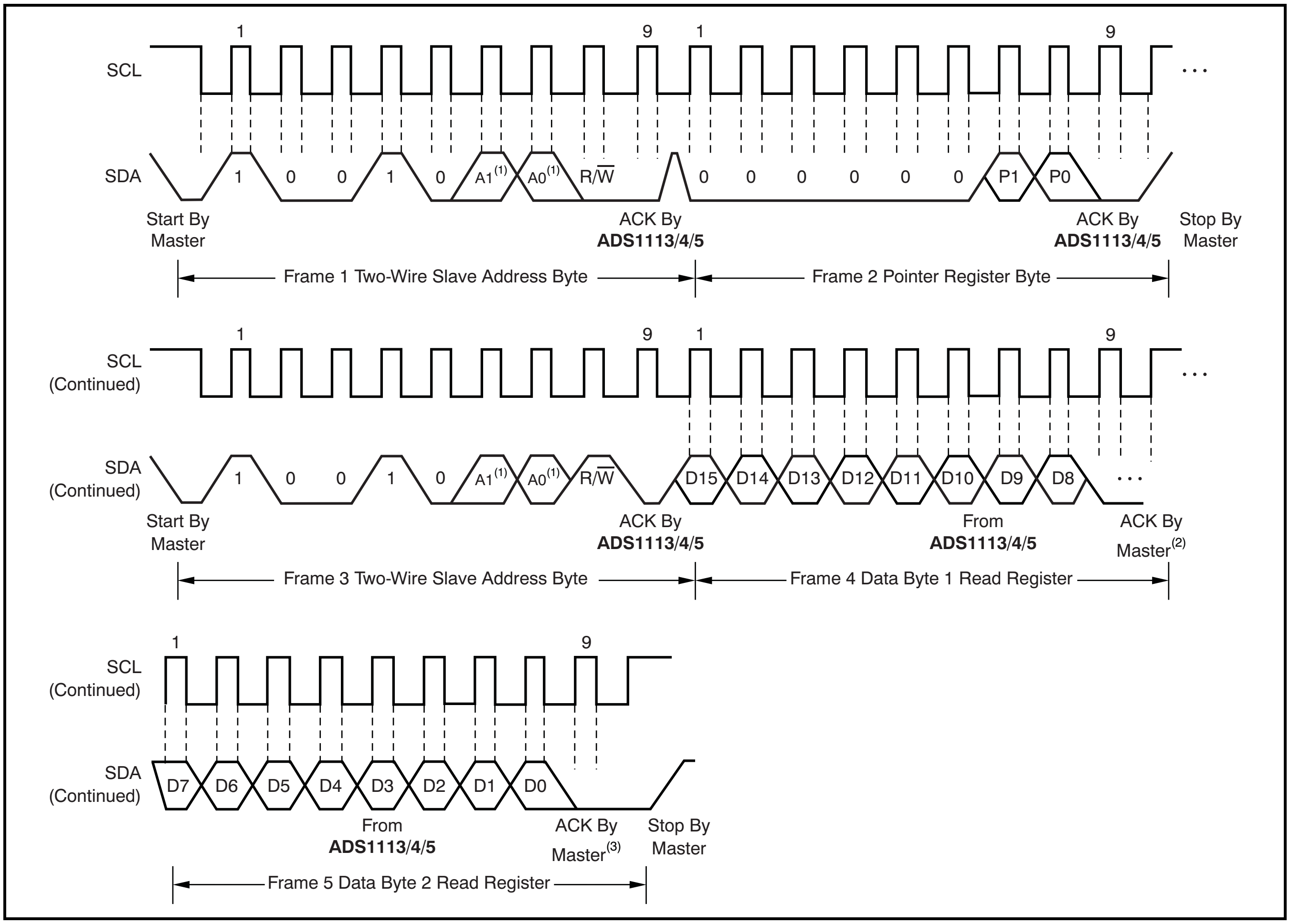
Table 10. Lo_thresh and Hi_thresh Registers

REGISTER	**Lo_thresh (Read/Write)**							
BIT	15	14	13	12	11	10	9	8
NAME	Lo_thresh15	Lo_thresh14	Lo_thresh13	Lo_thresh12	Lo_thresh11	Lo_thresh10	Lo_thresh9	Lo_thresh8
BIT	7	6	5	4	3	2	1	0
NAME	Lo_thresh7	Lo_thresh6	Lo_thresh5	Lo_thresh4	Lo_thresh3	Lo_thresh2	Lo_thresh1	Lo_thresh0

REGISTER	**Hi_thresh (Read/Write)**							
BIT	15	14	13	12	11	10	9	8
NAME	Hi_thresh15	Hi_thresh14	Hi_thresh13	Hi_thresh12	Hi_thresh11	Hi_thresh10	Hi_thresh9	Hi_thresh8
BIT	7	6	5	4	3	2	1	0
NAME	Hi_thresh7	Hi_thresh6	Hi_thresh5	Hi_thresh4	Hi_thresh3	Hi_thresh2	Hi_thresh1	Hi_thresh0

Lo_thresh default = 8000h.

Hi_thresh default = 7FFFh.

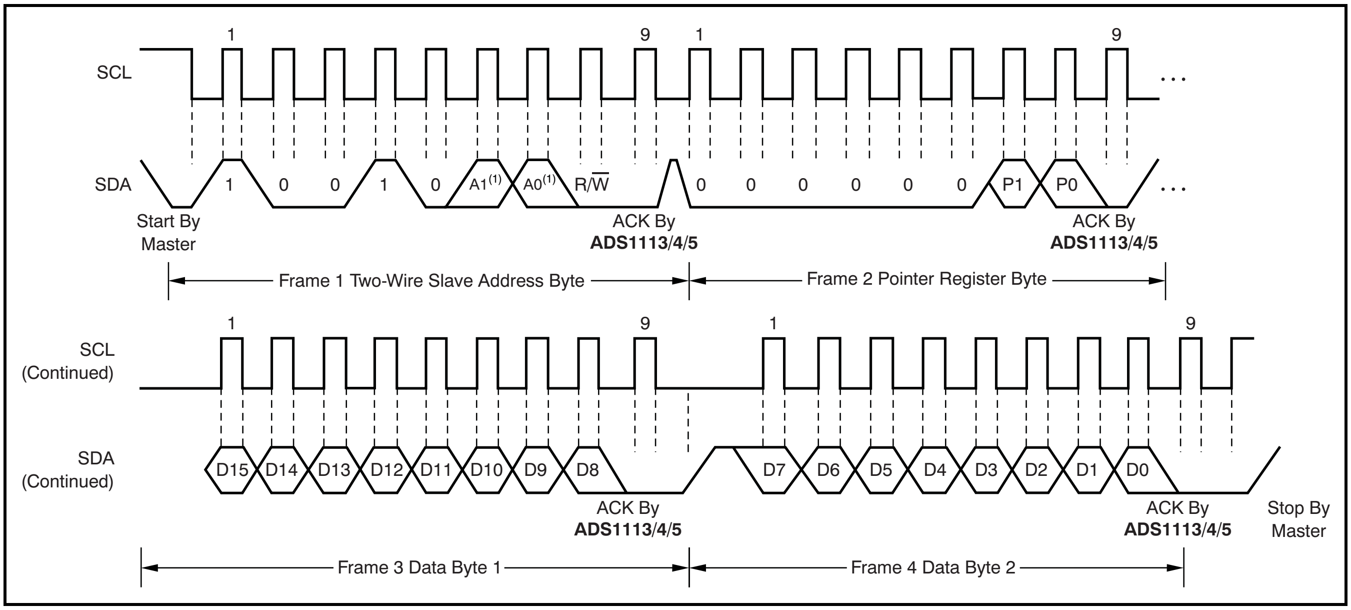


(1) The values of A0 and A1 are determined by the ADDR pin.

(2) Master can leave SDA high to terminate a single-byte read operation.

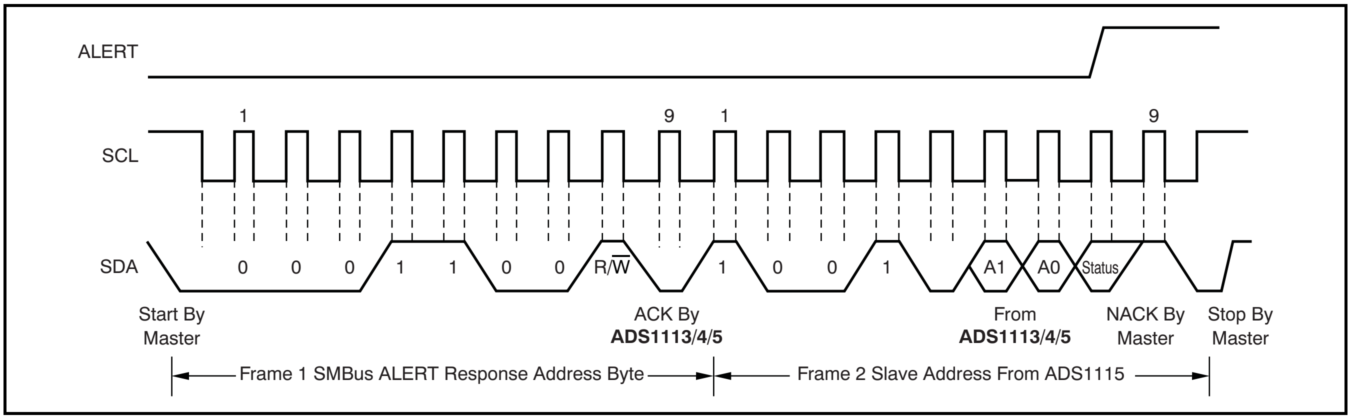
(3) Master can leave SDA high to terminate a two-byte read operation.

Figure 30. Two-Wire Timing Diagram for Read Word Format



(1) The values of A0 and A1 are determined by the ADDR pin.

Figure 31. Two-Wire Timing Diagram for Write Word Format



(1) The values of A0 and A1 are determined by the ADDR pin.

Figure 32. Timing Diagram for SMBus ALERT Response

APPLICATION INFORMATION

The following sections give example circuits and suggestions for using the ADS1113/4/5 in various situations.

BASIC CONNECTIONS

For many applications, connecting the ADS1113/4/5 is simple. A basic connection diagram for the ADS1115 is shown in Figure 33.

The fully differential voltage input of the ADS1113/4/5 is ideal for connection to differential sources with moderately low source impedance, such as thermocouples and thermistors. Although the ADS1113/4/5 can read bipolar differential signals, they cannot accept negative voltages on either input. It may be helpful to think of the ADS1113/4/5 positive voltage input as *noninverting*, and of the negative input as *inverting.*

When the ADS1113/4/5 are converting data, they draw current in short spikes. The 0.1μF bypass capacitor supplies the momentary bursts of extra current needed from the supply.

The ADS1113/4/5 interface directly to standard mode, fast mode, and high-speed mode I^{2}C controllers. Any microcontroller I^{2}C peripheral, including master-only and non-multiple-master I^{2}C peripherals, can operate with the ADS1113/4/5. The ADS1113/4/5 do not perform clock-stretching (that is, they never pull the clock line low), so it is not necessary to provide for this function unless other clock-stretching devices are on the same I^{2}C bus.

Pull-up resistors are required on both the SDA and SCL lines because I^{2}C bus drivers are open-drain. The size of these resistors depends on the bus operating speed and capacitance of the bus lines. Higher-value resistors consume less power, but increase the transition times on the bus, limiting the bus speed. Lower-value resistors allow higher speed at the expense of higher power consumption. Long bus lines have higher capacitance and require smaller pull-up resistors to compensate. The resistors should not be too small; if they are, the bus drivers may not be able to pull the bus lines low.

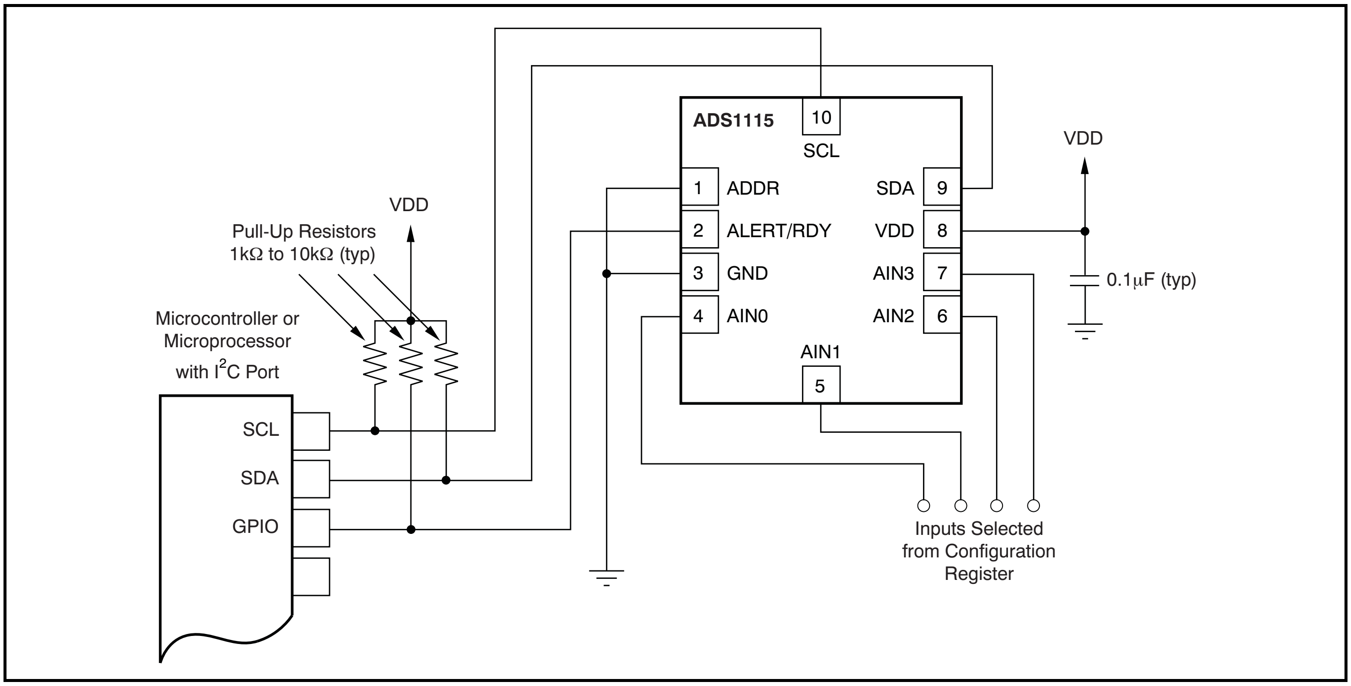


Figure 33. Typical Connections of the ADS1115

CONNECTING MULTIPLE DEVICES

Connecting multiple ADS1113/4/5s to a single bus is simple. Using the address pin, the ADS1113/4/5 can be set to one of four different I^{2}C addresses. An example showing three ADS1113/4/5 devices is given in Figure 35. Up to four ADS1113/4/5s (using different address pin configurations) can be connected to a single bus.

Note that only one set of pull-up resistors is needed per bus. The pull-up resistor values may need to be lowered slightly to compensate for the additional bus capacitance presented by multiple devices and increased line length.

The TMP421 and DAC8574 devices detect the respective I^{2}C bus addresses based on the states of pins. In the example, the TMP421 has the address 0101010, and the DAC8574 has the address 1001100. Consult the DAC8574 and TMP421 data sheets, available at www.ti.com, for further details.

USING GPIO PORTS FOR COMMUNICATION

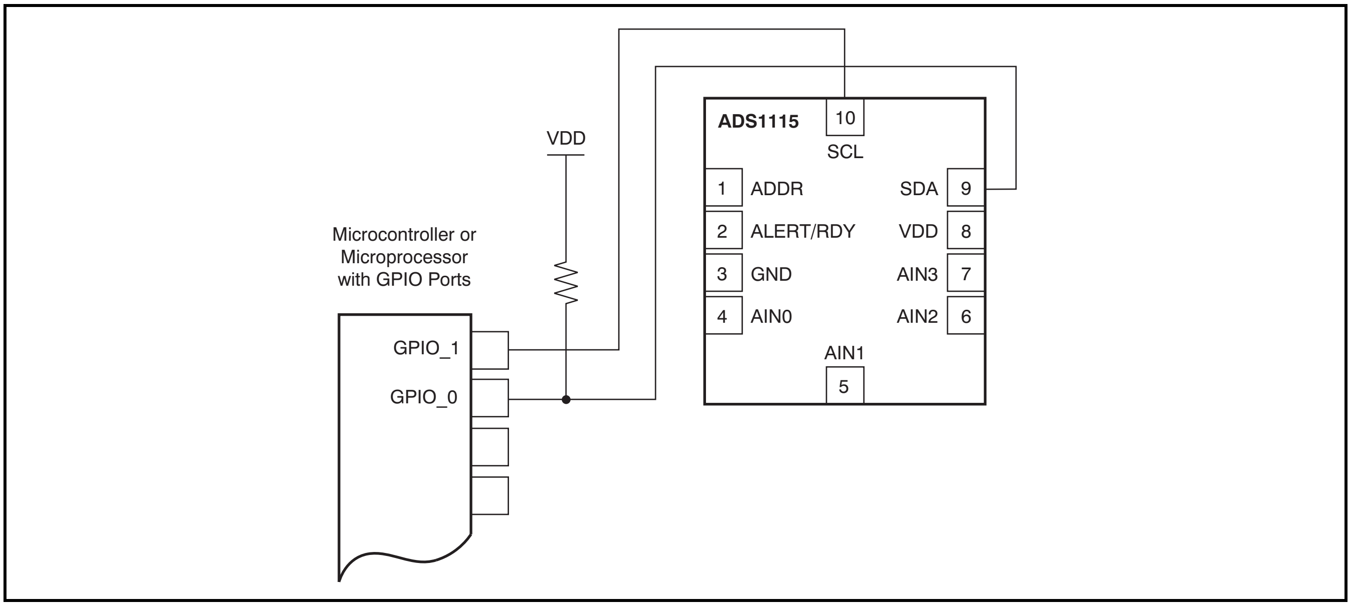
Most microcontrollers have programmable input/output (I/O) pins that can be set in software to act as inputs or outputs. If an I^{2}C controller is not available, the ADS1113/4/5 can be connected to GPIO pins and the I^{2}C bus protocol simulated, or *bit-banged*, in software. An example of this configuration for a single ADS1113/4/5 is shown in Figure 34.

Bit-banging I^{2}C with GPIO pins can be done by setting the GPIO line to '0' and toggling it between input and output modes to apply the proper bus states. To drive the line low, the pin is set to output '0'; to let the line go high, the pin is set to input. When the pin is set to input, the state of the pin can be read; if another device is pulling the line low, this configuration reads as a '0' in the port input register.

Note that no pull-up resistor is shown on the SCL line. In this simple case, the resistor is not needed; the microcontroller can simply leave the line on output, and set it to '1' or '0' as appropriate. This action is possible because the ADS1113/4/5 never drive the clock line low. This technique can also be used with multiple devices, and has the advantage of lower current consumption as a result of the absence of a resistive pull-up.

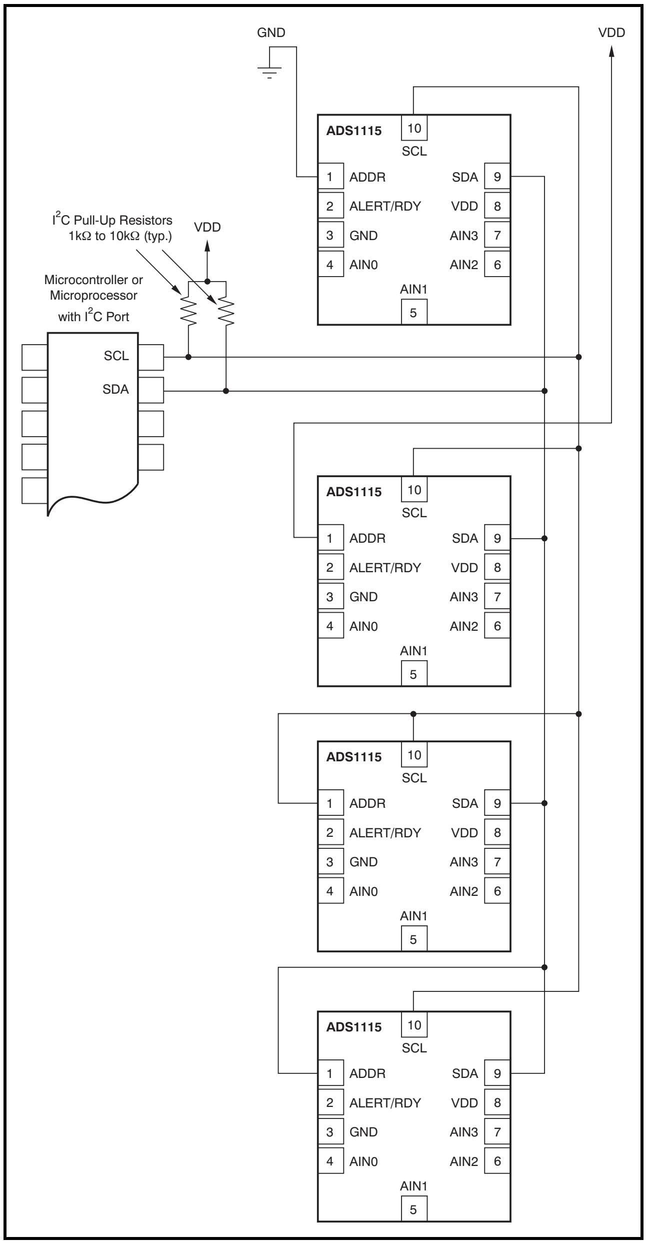
If there are any devices on the bus that may drive the clock lines low, this method should not be used; the SCL line should be high-Z or '0' and a pull-up resistor provided as usual.

Some microcontrollers have selectable strong pull-up circuits built in to the GPIO ports. In some cases, these circuits can be switched on and used in place of an external pull-up resistor. Weak pull-ups are also provided on some microcontrollers, but usually these are too weak for I^{2}C communication. If there is any doubt about the matter, test the circuit before committing it to production.



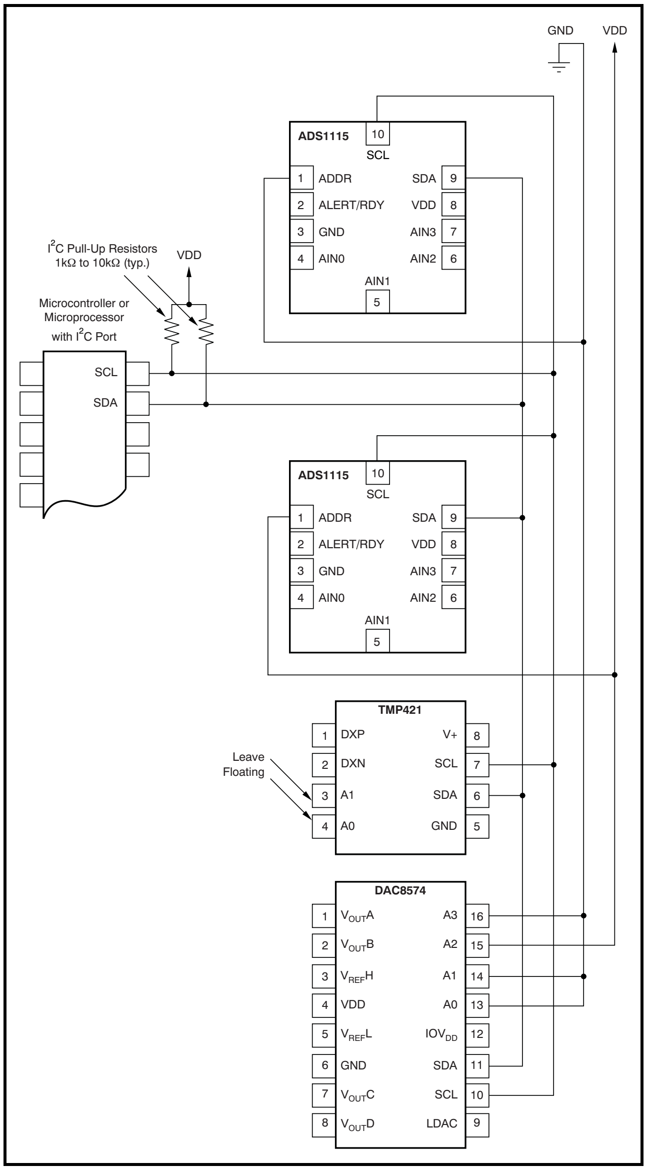
NOTE: ADS1113/4/5 power and input connections omitted for clarity.

Figure 34. Using GPIO with a Single ADS1115



NOTE: ADS1113/4/5 power and input connections omitted for clarity. The ADDR pin selects the I^{2}C address.

Figure 35. Connecting Multiple ADS1113/4/5s



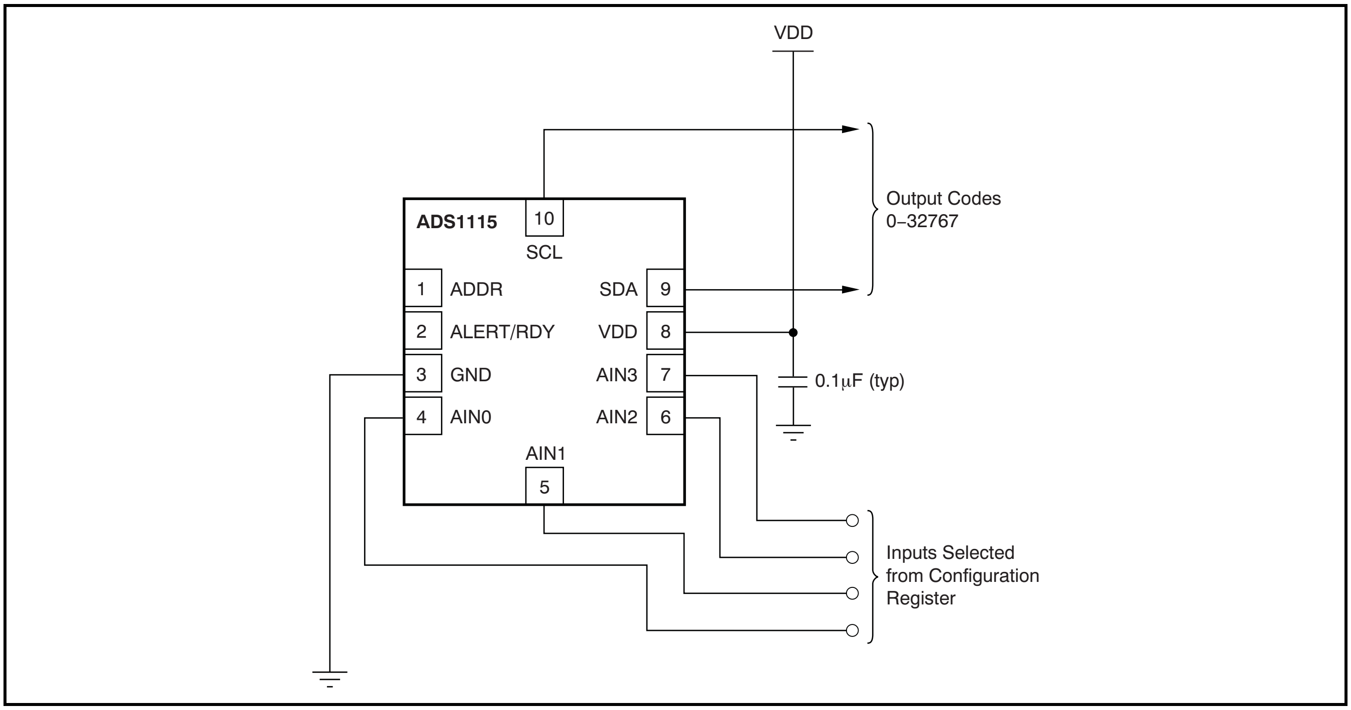
NOTE: ADS1113/4/5 power and input connections omitted for clarity. ADDR, A3, A2, A1, and A0 select the I^{2}C addresses.

Figure 36. Connecting Multiple Device Types

SINGLE-ENDED INPUTS

Although the ADS1115 has two differential inputs, the device can easily measure four single-ended signals. Figure 37 shows a single-ended connection scheme. The ADS1115 is configured for single-ended measurement by configuring the MUX to measure each channel with respect to ground. Data are then read out of one input based on the selection on the configuration register. The single-ended signal can range from 0V to supply. The ADS1115 loses no linearity anywhere within the input range. Negative voltages cannot be applied to this circuit because the ADS1115 can only accept positive voltages.

The ADS1115 input range is bipolar differential with respect to the reference. The single-ended circuit shown in Figure 37 covers only half the ADS1115 input scale because it does not produce differentially negative inputs; therefore, one bit of resolution is lost.



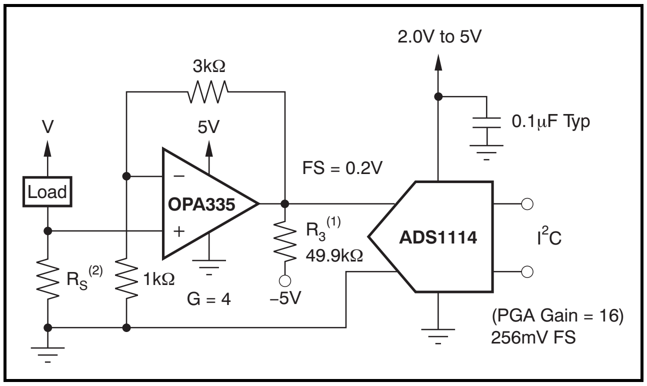
NOTE: Digital and address pin connections omitted for clarity.

Figure 37. Measuring Single-Ended Inputs

LOW-SIDE CURRENT MONITOR

Figure 38 shows a circuit for a low-side shunt-type current monitor. The circuit monitors the voltage across a shunt resistor, which is sized as small as possible while giving a measurable output voltage. This voltage is amplified by an OPA335 low-drift op amp, and the result is read by the ADS1114/5.

It is suggested that the ADS1114/5 be operated at a gain of 8. The gain of the OPA335 can then be set lower. For a gain of 16, the op amp should be set up to give a maximum output voltage no greater than 0.256V. If the shunt resistor is sized to provide a maximum voltage drop of 50mV at full-scale current, the full-scale input to the ADS1114/5 is 0.2V.



(1) Pull-down resistor to allow accurate swing to 0V.

(2) R_S is sized for a 50mV drop at full-scale current.

Figure 38. Low-Side Current Measurement

The ADS1113/4/5 are fabricated in a small-geometry, low-voltage process. The analog inputs feature protection diodes to the supply rails. However, the current-handling ability of these diodes is limited, and the ADS1113/4/5 can be permanently damaged by analog input voltages that remain more than approximately 300mV beyond the rails for extended periods. One way to protect against overvoltage is to place current-limiting resistors on the input lines. The ADS1113/4/5 analog inputs can withstand momentary currents as large as 100mA.

If the ADS1113/4/5 are driven by an op amp with high-voltage supplies, such as ±12V, protection should be provided, even if the op amp is configured so that it does not output out-of-range voltages. Many op amps drift to one of the supply rails immediately when power is applied, usually before the input has stabilized; this momentary spike can damage the ADS1113/4/5. This incremental damage results in slow, long-term failure, which can be disastrous for permanently installed, low-maintenance systems.

If an op amp or other front-end circuitry is used with an ADS1113/4/5, performance characteristics must be taken into account when designing the application.

REVISION HISTORY

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision A (August 2009) to Revision B	**Page**
• Deleted Operating Temperature bullet from Features section	1
• Deleted *Operating temperature range* parameter from Absolute Maximum Ratings table	2
• Deleted *Operating temperature* parameter from Temperature section of Electrical Characteristics table	4
• Changed Figure 2 to reflect maximum operating temperature	6
• Changed Figure 3 to reflect maximum operating temperature	6
• Changed Figure 4 to reflect maximum operating temperature	6
• Changed Figure 5 to reflect maximum operating temperature	6
• Changed Figure 6 to reflect maximum operating temperature	6
• Changed +140°C to +125°C in Figure 9	7
• Changed +140°C to +125°C in Figure 10	7
• Changed +140°C to +125°C in Figure 11	7
• Changed +140°C to +125°C in Figure 12	7
• Changed Figure 13 to reflect maximum operating temperature	7
• Changed Figure 16 to reflect maximum operating temperature	8
• Changed Figure 20 to reflect maximum operating temperature	9

PACKAGE OPTION ADDENDUM

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead/ Ball Finish	MSL Peak Temp (3)	Samples (Requires Login)
ADS1113IDGSR	ACTIVE	MSOP	DGS	10	2500	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-2-260C-1 YEAR	Add to cart
ADS1113IDGST	ACTIVE	MSOP	DGS	10	250	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-2-260C-1 YEAR	Add to cart
ADS1113IRUGR	ACTIVE	X2QFN	RUG	10	3000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM	Add to cart
ADS1113IRUGT	ACTIVE	X2QFN	RUG	10	250	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM	Add to cart
ADS1114IDGSR	ACTIVE	MSOP	DGS	10	2500	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-2-260C-1 YEAR	Add to cart
ADS1114IDGST	ACTIVE	MSOP	DGS	10	250	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-2-260C-1 YEAR	Add to cart
ADS1114IRUGR	ACTIVE	X2QFN	RUG	10	3000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM	Add to cart
ADS1114IRUGT	ACTIVE	X2QFN	RUG	10	250	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM	Add to cart
ADS1115IDGSR	ACTIVE	MSOP	DGS	10	2500	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-2-260C-1 YEAR	Add to cart
ADS1115IDGST	ACTIVE	MSOP	DGS	10	250	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-2-260C-1 YEAR	Add to cart
ADS1115IRUGR	ACTIVE	X2QFN	RUG	10	3000	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM	Add to cart
ADS1115IRUGT	ACTIVE	X2QFN	RUG	10	250	Green (RoHS & no Sb/Br)	CU NIPDAU	Level-1-260C-UNLIM	Add to cart

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) Eco Plan - The planned eco-friendly classification: Pb-Free (RoHS), Pb-Free (RoHS Exempt), or Green (RoHS & no Sb/Br) - please check http://www.ti.com/productcontent for the latest availability information and additional product content details.
TBD: The Pb-Free/Green conversion plan has not been defined.

Pb-Free (RoHS): TI's terms "Lead-Free" or "Pb-Free" mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TI Pb-Free products are suitable for use in specified lead-free processes.
Pb-Free (RoHS Exempt): This component has a RoHS exemption for either 1) lead-based flip-chip solder bumps used between the die and package, or 2) lead-based die adhesive used between the die and leadframe. The component is otherwise considered Pb-Free (RoHS compatible) as defined above.
Green (RoHS & no Sb/Br): TI defines "Green" to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material)

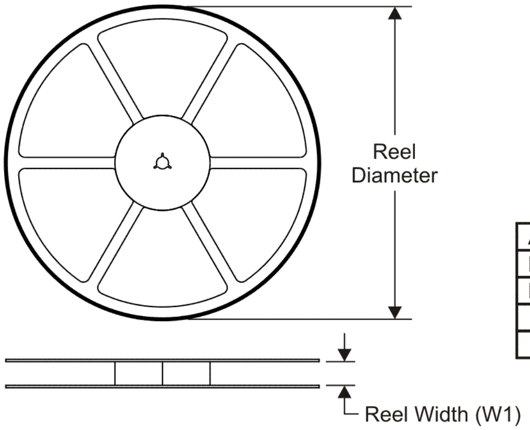
[3] MSL, Peak Temp. -- The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

Important Information and Disclaimer: The information provided on this page represents TI's knowledge and belief as of the date that it is provided. TI bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TI has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TI and TI suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

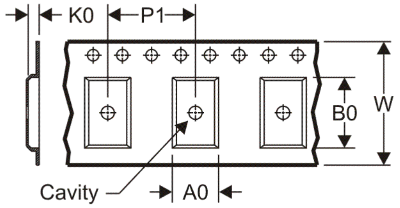
In no event shall TI's liability arising out of such information exceed the total purchase price of the TI part(s) at issue in this document sold by TI to Customer on an annual basis.

TAPE AND REEL INFORMATION

REEL DIMENSIONS

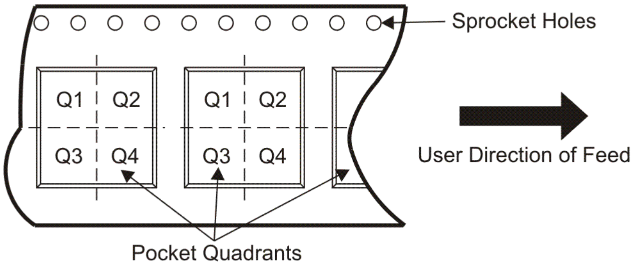


TAPE DIMENSIONS



A0	Dimension designed to accommodate the component width
B0	Dimension designed to accommodate the component length
K0	Dimension designed to accommodate the component thickness
W	Overall width of the carrier tape
P1	Pitch between successive cavity centers

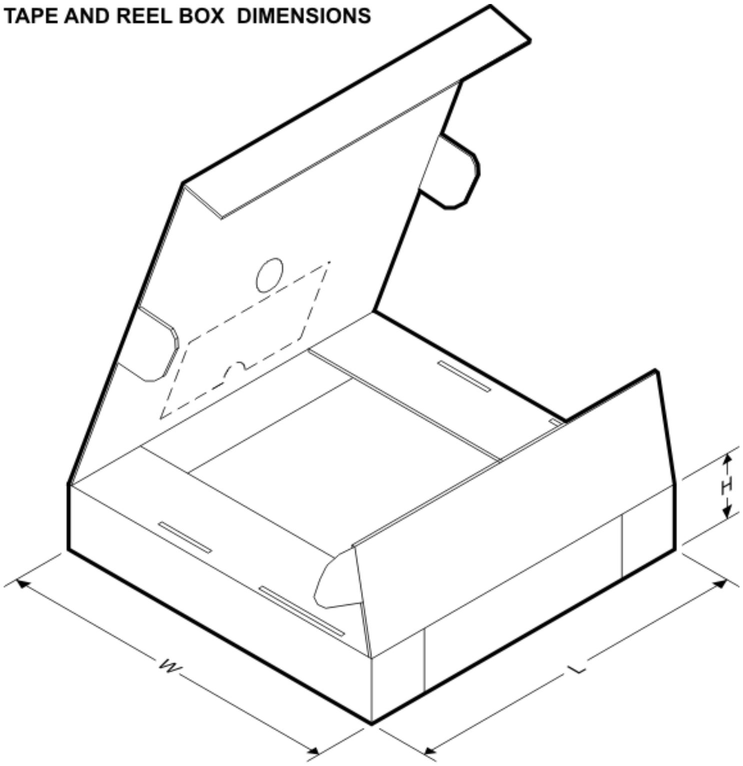
QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Reel Diameter (mm)	Reel Width W1 (mm)	A0 (mm)	B0 (mm)	K0 (mm)	P1 (mm)	W (mm)	Pin1 Quadrant
ADS1113IDGSR	MSOP	DGS	10	2500	330.0	12.4	5.3	3.3	1.3	8.0	12.0	Q1
ADS1113IDGST	MSOP	DGS	10	250	180.0	12.4	5.3	3.3	1.3	8.0	12.0	Q1
ADS1113IRUGR	X2QFN	RUG	10	3000	179.0	8.4	1.75	2.25	0.65	4.0	8.0	Q1
ADS1113IRUGT	X2QFN	RUG	10	250	179.0	8.4	1.75	2.25	0.65	4.0	8.0	Q1
ADS1114IDGSR	MSOP	DGS	10	2500	330.0	12.4	5.3	3.3	1.3	8.0	12.0	Q1
ADS1114IDGST	MSOP	DGS	10	250	180.0	12.4	5.3	3.3	1.3	8.0	12.0	Q1
ADS1114IRUGR	X2QFN	RUG	10	3000	179.0	8.4	1.75	2.25	0.65	4.0	8.0	Q1
ADS1114IRUGT	X2QFN	RUG	10	250	179.0	8.4	1.75	2.25	0.65	4.0	8.0	Q1
ADS1115IDGSR	MSOP	DGS	10	2500	330.0	12.4	5.3	3.3	1.3	8.0	12.0	Q1
ADS1115IDGST	MSOP	DGS	10	250	180.0	12.4	5.3	3.3	1.3	8.0	12.0	Q1
ADS1115IRUGR	X2QFN	RUG	10	3000	179.0	8.4	1.75	2.25	0.65	4.0	8.0	Q1
ADS1115IRUGT	X2QFN	RUG	10	250	179.0	8.4	1.75	2.25	0.65	4.0	8.0	Q1

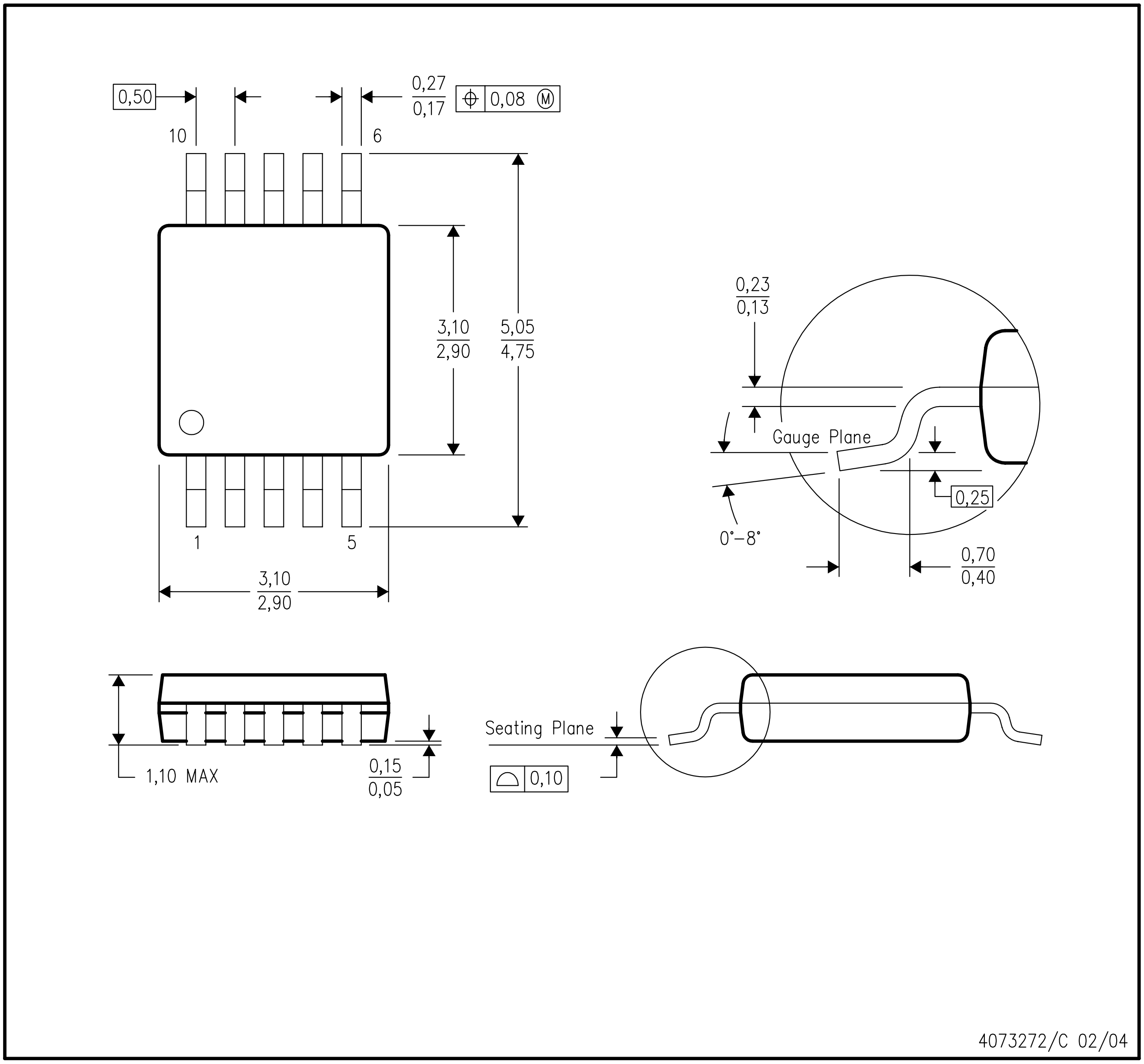
TAPE AND REEL BOX DIMENSIONS



*All dimensions are nominal

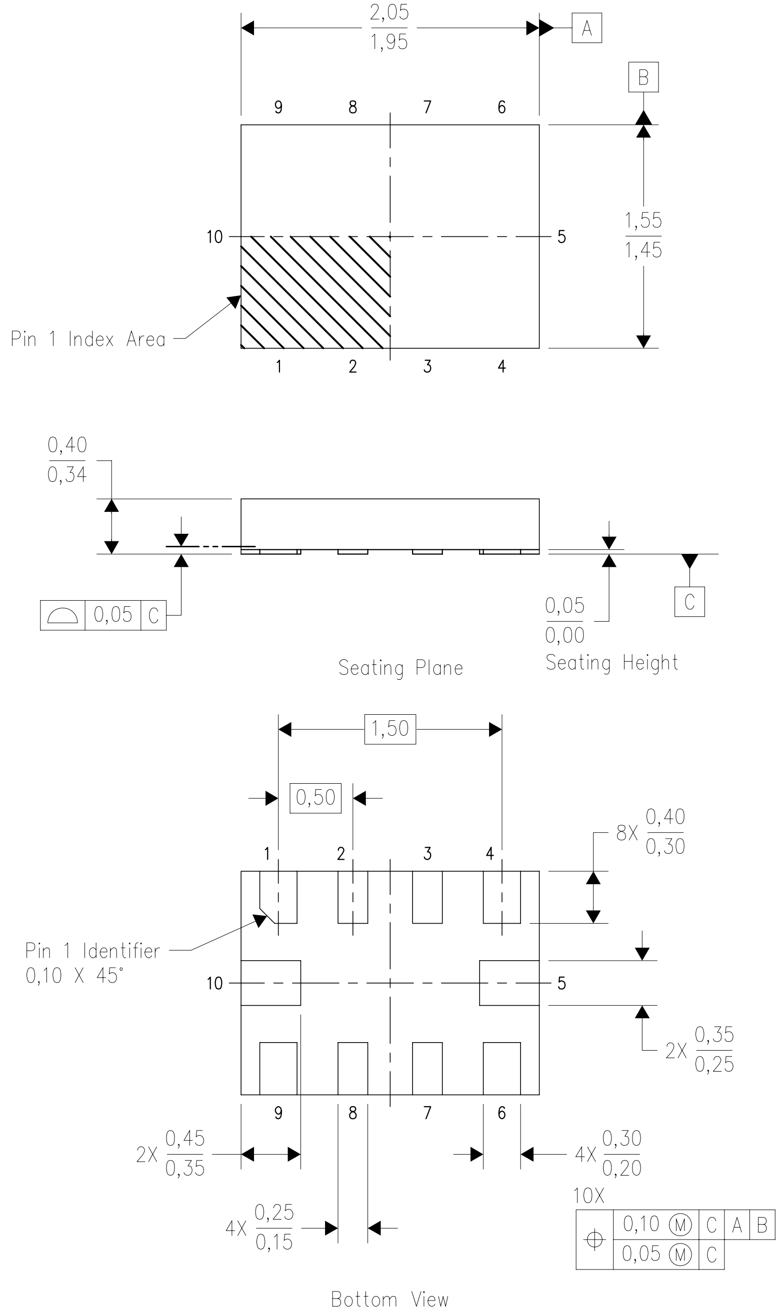
Device	Package Type	Package Drawing	Pins	SPQ	Length (mm)	Width (mm)	Height (mm)
ADS1113IDGSR	MSOP	DGS	10	2500	370.0	355.0	55.0
ADS1113IDGST	MSOP	DGS	10	250	195.0	200.0	45.0
ADS1113IRUGR	X2QFN	RUG	10	3000	203.0	203.0	35.0
ADS1113IRUGT	X2QFN	RUG	10	250	203.0	203.0	35.0
ADS1114IDGSR	MSOP	DGS	10	2500	370.0	355.0	55.0
ADS1114IDGST	MSOP	DGS	10	250	195.0	200.0	45.0
ADS1114IRUGR	X2QFN	RUG	10	3000	203.0	203.0	35.0
ADS1114IRUGT	X2QFN	RUG	10	250	203.0	203.0	35.0
ADS1115IDGSR	MSOP	DGS	10	2500	370.0	355.0	55.0
ADS1115IDGST	MSOP	DGS	10	250	195.0	200.0	45.0
ADS1115IRUGR	X2QFN	RUG	10	3000	203.0	203.0	35.0
ADS1115IRUGT	X2QFN	RUG	10	250	203.0	203.0	35.0

DGS (S–PDSO–G10) PLASTIC SMALL–OUTLINE PACKAGE



NOTES:
- A. All linear dimensions are in millimeters.
- B. This drawing is subject to change without notice.
- C. Body dimensions do not include mold flash or protrusion.
- D. Falls within JEDEC MO–187 variation BA.

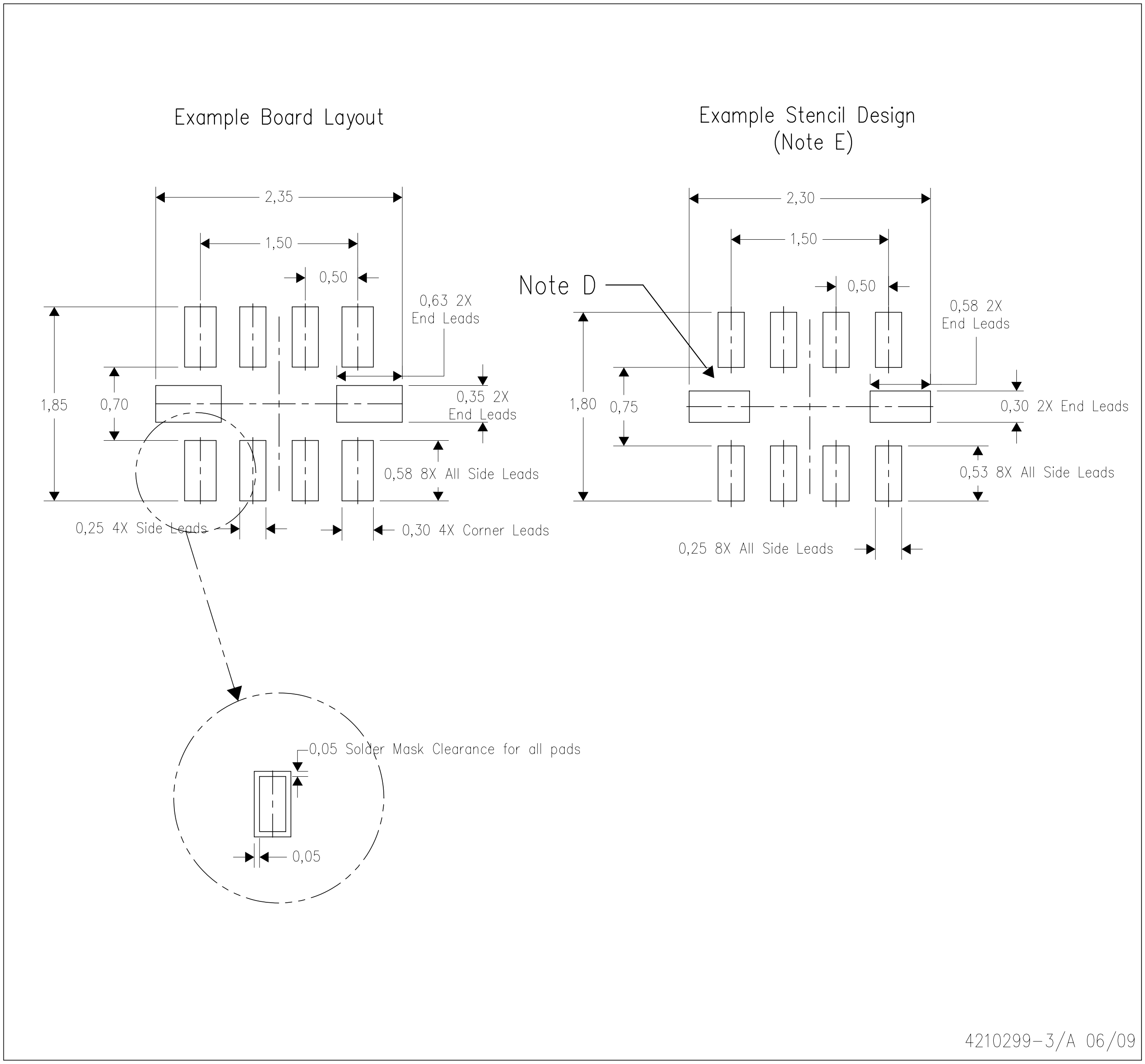
RUG (R-PQFP-N10) PLASTIC QUAD FLATPACK



4208528-3/B 04/2008

NOTES:
- A. All linear dimensions are in millimeters. Dimensioning and tolerancing per ASME Y14.5M-1994.
- B. This drawing is subject to change without notice.
- C. QFN (Quad Flatpack No-Lead) package configuration.
- D. This package complies to JEDEC MO-288 variation X2EFD.

RUG (R–PQFP–N10)



NOTES:
- A. All linear dimensions are in millimeters.
- B. This drawing is subject to change without notice.
- C. Publication IPC–7351 is recommended for alternate designs.
- D. Customers should contact their board fabrication site for minimum solder mask web tolerances between signal pads.
- E. Maximum stencil thickness 0,127 mm (5 mils). All linear dimensions are in millimeters.
- F. Laser cutting apertures with trapezoidal walls and also rounding corners will offer better paste release. Customers should contact their board assembly site for stencil design recommendations. Refer to IPC 7525 for stencil design considerations.
- G. Side aperture dimensions over–print land for acceptable area ratio > 0.66. Customer may reduce side aperture dimensions if stencil manufacturing process allows for sufficient release at smaller opening.

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI’s standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of TI information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

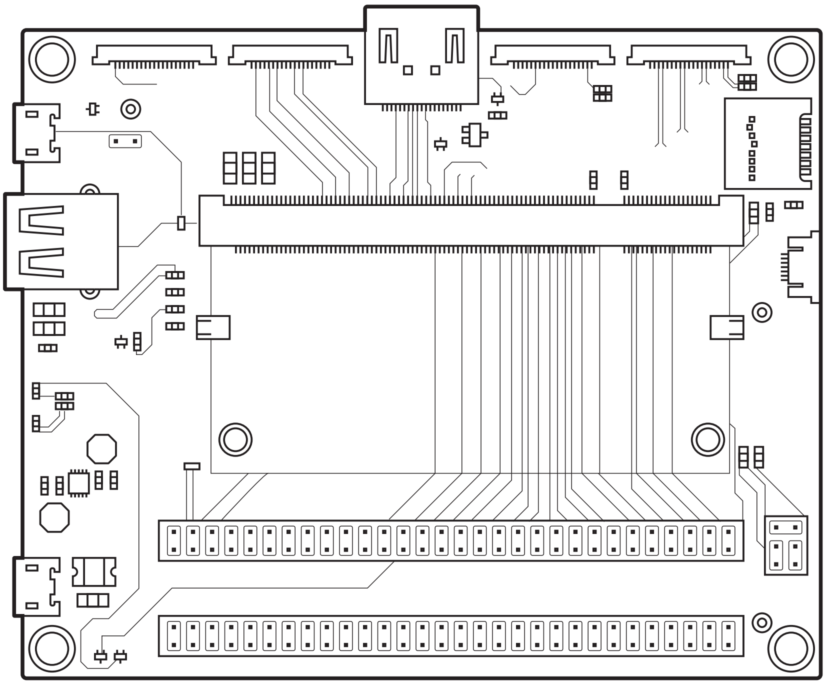
Products		**Applications**	
Audio	www.ti.com/audio	Communications and Telecom	www.ti.com/communications
Amplifiers	amplifier.ti.com	Computers and Peripherals	www.ti.com/computers
Data Converters	dataconverter.ti.com	Consumer Electronics	www.ti.com/consumer-apps
DLP® Products	www.dlp.com	Energy and Lighting	www.ti.com/energy
DSP	dsp.ti.com	Industrial	www.ti.com/industrial
Clocks and Timers	www.ti.com/clocks	Medical	www.ti.com/medical
Interface	interface.ti.com	Security	www.ti.com/security
Logic	logic.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Power Mgmt	power.ti.com	Transportation and Automotive	www.ti.com/automotive
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Mobile Processors	www.ti.com/omap		
Wireless Connctivity	www.ti.com/wirelessconnectivity		
		TI E2E Community Home Page	e2e.ti.com

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

ANEXO D

OFFICIAL
MANUFACTURER

RASPBERRY Pi 3B+
COMPUTE MODULE 3



www.okdo.com

OVERVIEW

The Raspberry Pi Compute Module (CM1), Compute Module 3 (CM3) and Compute Module 3 Lite (CM3L) are DDR2-SODIMM-mechanically-compatible System on Modules (SoMs) containing pro- cessor, memory, eMMC Flash (for CM1 and CM3) and supporting power circuitry. These modules allow a designer to leverage the Raspberry Pi hardware and software stack in their own custom systems and form factors. In addition these module have extra IO interfaces over and above what is available on the Raspberry Pi model A/B boards opening up more options for the designer.

The CM1 contains a BCM2835 processor (as used on the original Raspberry Pi and Raspberry Pi B+ models), 512MByte LPDDR2 RAM and 4Gbytes eMMC Flash. The CM3 contains a BCM2837 pro- cessor (as used on the Raspberry Pi 3), 1Gbyte LPDDR2 RAM and 4Gbytes eMMC Flash. Finally the CM3L product is the same as CM3 except the eMMC Flash is not fitted, and the SD/eMMC interface pins are available for the user to connect their own SD/eMMC device.

Note that the BCM2837 processor is an evolution of the BCM2835 processor. The only real differences are that the BCM2837 can address more RAM (up to 1Gbyte) and the ARM CPU complex has been upgraded from a single core ARM11 in BCM2835 to a Quad core Cortex A53 with dedicated 512Kbyte L2 cache in BCM2837. All IO interfaces and peripherals stay the same and hence the two chips are largely software and hardware compatible.

The pinout of CM1 and CM3 are identical. Apart from the CPU upgrade and increase in RAM the other significant hwardware differences to be aware of are that CM3 has grown from 30mm to 31mm in height, the VBAT supply can now draw significantly more power under heavy CPU load, and the HDMI HPD N 1V8 (GPIO46 1V8 on CM1) and EMMC EN N 1V8 (GPIO47 1V8 on CM1) are now driven from an IO expander rather than the processor. If a designer of a CM1 product has a suitably specified VBAT, can accomodate the extra 1mm module height increase and has followed the design rules with respect to GPIO46 1V8 and GPIO47 1V8 then a CM3 should work fine in a board designed for a CM1.

RASPBERRY Pi 3B+

COMPUTE MODULE 3

FEATURES

Hardware

- Low cost
- Low power
- High availability
- High reliability
  - – Tested over millions of Raspberry Pis Produced to date
  - – Module IO pins have 35u hard gold plating

Peripherals

- 48x GPIO
- 2x I2C
- 2x SPI
- 2x UART
- 2x SD/SDIO
- 1x HDMI 1.3a
- 1x USB2 HOST/OTG
- 1x DPI (Parallel RGB Display) • 1x NAND interface (SMI)
- 1x 4-lane CSI Camera Interface (up to 1Gbps per lane)
- 1x 2-lane CSI Camera Interface (up to 1Gbps per lane)
- 1x 4-lane DSI Display Interface (up to 1Gbps per lane)
- 1x 2-lane DSI Display Interface (up to 1Gbps per lane)

Software

- ARMv6 (CM1) or ARMv7 (CM3, CM3L) Instruction Set
- Mature and stable Linux software stack
  - – Latest Linux Kernel support
  - – Many drivers upstreamed
  - – Stable and well supported userland
  - – Full availability of GPU functions using standard APIs

BLOCK DIAGRAM

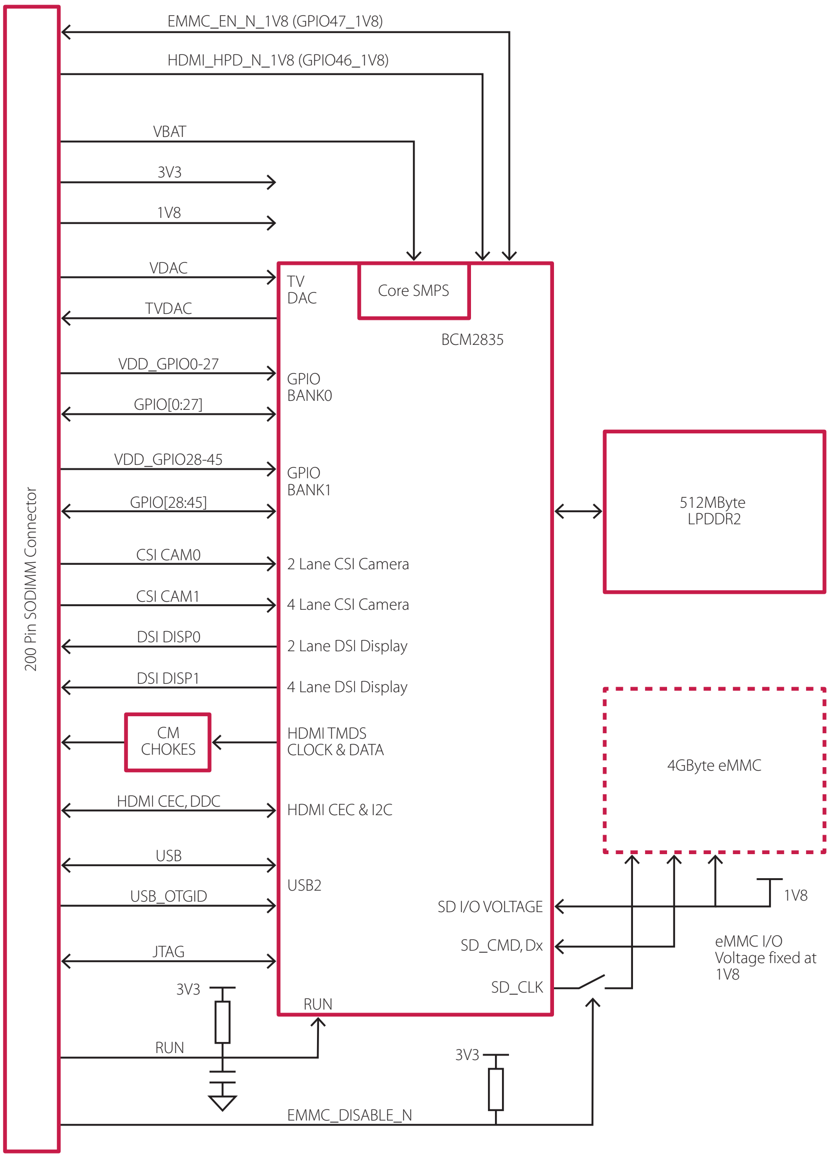


Figure 1: CM1 Block Diagram

BLOCK DIAGRAM

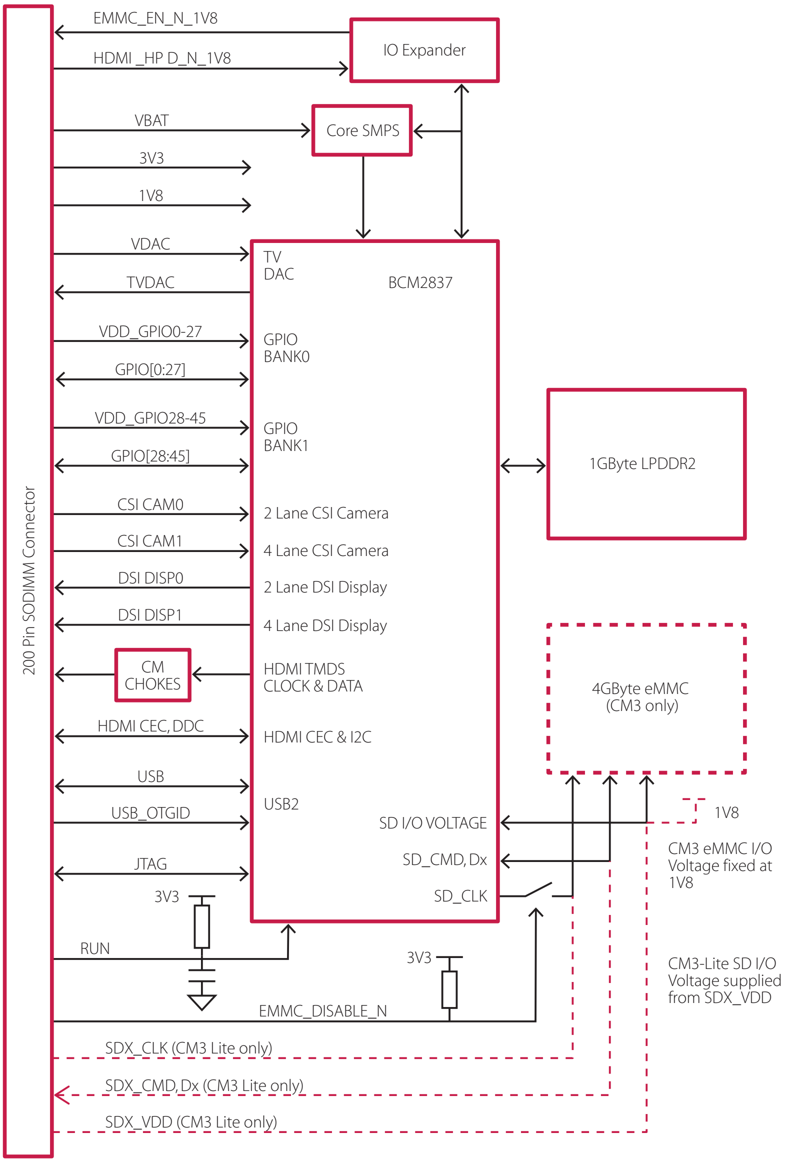


Figure 2: CM3/CM3L Block Diagram

MECHANICAL SPECIFICATION

The Compute Modules conform to JEDEC MO-224 mechanical specification for 200 pin DDR2 (1.8V) SODIMM modules (with the exception that the CM3, CM3L modules are 31mm in height rather than 30mm of CM1) and therefore should work with the many DDR2 SODIMM sockets available on the market. (Please note that the pinout of the Compute Module is not the same as a DDR2 SODIMM module; they are not electrically compatible.)

The SODIMM form factor was chosen as a way to provide the 200 pin connections using a standard, readily available and low cost connector compatible with low cost PCB manufacture.
The maximum component height on the underside of the Compute Module is 1.2mm. The maximum component height on the top side of the Compute Module is 1.5mm. The Compute Module PCB thickness is 1.0mm +/- 0.1mm.

Note that the location and arrangement of components on the Compute Module may change slightly over time due to revisions for cost and manufacturing considerations; however, maximum component heights and PCB thickness will be kept as specified.

Figure 3 gives the CM1 mechanical dimensions. Figure 4 gives the CM3 and CM3L mechanical dimensions.

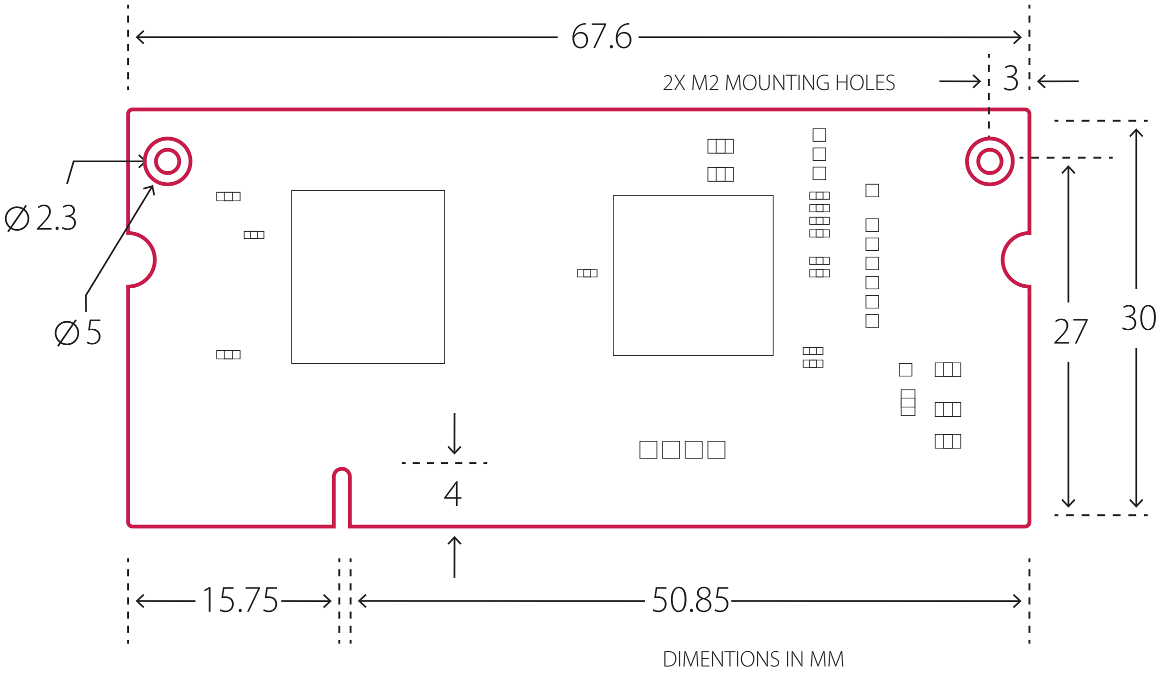


Figure 3: CM1 Mechanical Dimensions

MECHANICAL SPECIFICATION

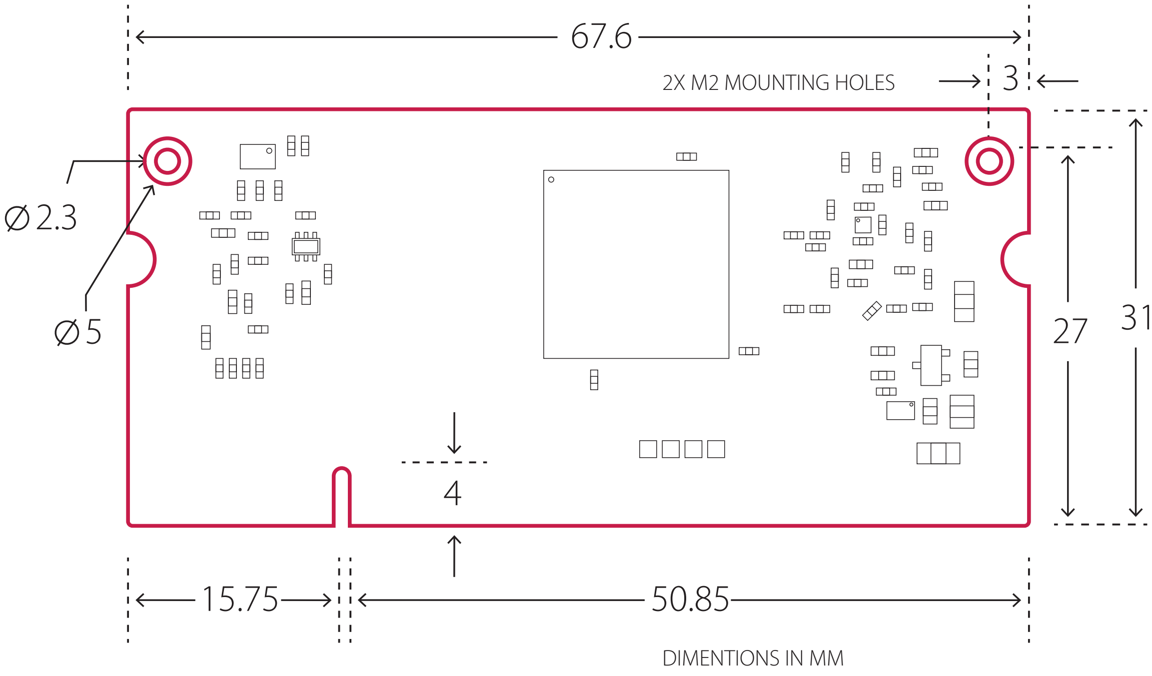


Figure 4: CM3 and CM3L Mechanical Dimensions

RASPBERRY Pi 3B+

COMPUTE MODULE 3

PIN ASSIGNMENTS

CM1	CM3-Lite	CM3	PIN		PIN	CM3	CM3-Lite	CM1
GND			1		2	EMMC_DISABLE_N		
GPIO0			3		4	NC	SDX_VDD	NC
GPIO1			5		6	NC	SDX_VDD	NC
GND			7		8	GND		NC
GPIO2			9		10	NC	SDX_CLK	NC
GPIO3			11		12	NC	SDX_CMD	NC
GND			13		14	GND		NC
GPIO4			15		16	NC	SDX_D0	NC
GPIO5			17		18	NC	SDX_D1	NC
GND			19		20			NC
GPIO6			21		22	NC	SDX_D2	NC
GPIO7			23		24	NC	SDX_D3	NC
GND			25		26	GND		
GPIO8			27		28	GPIO28		
GPIO9			29		30	GPIO29		
GND			31		32	GND		
GPIO10			33		34	GPIO30		
GPIO11			35		36	GPIO31		
GND			37		38	GND		
GPIO0-27_VDD			39		40	GPIO0-27_VDD		
			KEY					
GPIO28-45_VDD			41		42	GPIO28-45_VDD		
GND			43		44	GND		
GPIO12			45		46	GPIO32		
GPIO13			47		48	GPIO33		
GND			49		50	GND		
GPIO14			51		52	GPIO34		
PGIO15			53		54	GPIO35		
GND			55		56	GND		
GPIO16			57		58	GPIO36		
GPIO17			59		60	GPIO37		
GND			61		62	GND		
GPIO18			63		64	GPIO38		
GPIO19			65		66	GPIO39		
GND			67		68	GND		
GPIO20			69		70	GPIO40		
GPIO21			71		72	GPIO41		
GND			73		74	GND		
GPIO22			75		76	GPIO42		
GPIO23			77		78	GPIO43		
GND			79		80	GND		
GPIO24			81		82	GPIO44		
GPIO25			83		84	GPIO45		
GND			85		86	GND		
GPIO26			87		88	HDMI_HPD_N_1V8		GPIO46_1V8
GPIO27			89		90	EMMC_EN_N_1V8		GPIO47_1V8
GND			91		92	GND		
DSI0_DN1			93		94	DSI1_DP0		
DSI0_DP1			95		96	DSI1_DN0		
GND			97		98	GND		
DSI0_DN0			99		100	DSI1_CP		
DSI0_DP0			101		102	DSI1_CN		
GND			103		104	GND		
DSI0_CN			105		106	DSI1_DP3		
DSI0_CP			107		18	DSI1_DN3		
GND			109		110	GND		
HDMI_CLK_N			111		112	DSI1_DP2		
HDMI_CLK_P			113		114	DSI1_DN2		
GND			115		116	GND		
HDMI_D0_N			117		118	DSI1_DP1		
HDMI_D0_P			119		120	DSI1_DN1		
GND			121		122	GND		
HDMI_D1_N			123		124	NC		
HDMI_D1_P			125		126	NC		
GND			127		128	NC		
HDMI_D2_N			129		130	NC		
HDMI_D2_P			131		132	NC		
GND			133		134	GND		
CAM1_DP3			135		136	CAM0_DP0		
CAM1_DN3			137		138	CAM0_DN0		
GND			139		140	GND		
CAM1_DP2			141		142	CAM0_CP		
CAM1_DN2			143		144	CAM0_CN		
GND			145		146	GND		
CAM1_CP			147		148	CAM0_DP1		
CAM1_CN			149		150	CAM0_DN1		
GND			151		152	GND		
CAM1_DP1			153		154	NC		
CAM1_DN1			155		156	NC		
GND			157		158	NC		
CAM1_DP0			159		160	NC		
CAM1_DN0			161		162	NC		
GND			163		164	GND		
USB_DP			165		166	TVDAC		
USB_DM			167		168	USB_OTGID		
GND			169		170	GND		
HDMI_CEC			171		172	VC_TRST_N		
HDMI_SDA			173		174	VC_TDI		
HDMI_SCL			175		176	VC_TMS		
RUN			177		178	VC_TDO		
VDD_CORE (DO NOT CONNECT)			179		180	VC_TCK		
GND			181		182	GND		
1V8			183		184	1V8		
1V8			185		186	1V8		
GND			187		188	GND		
VDAC			189		190	VDAC		
3V3			191		192	3V3		
3V3			193		194	3V3		
GND			195		196	GND		
VBAT			197		189	VBAT		
VBAT			199		200	VBAT		

Table 2: Compute Module SODIMM Connector Pinout

Table 2 gives the Compute Module pinout and Table 3 gives the Compute Module pin functions.

PIN ASSIGNMENTS

Pin Name	DIR	Voltage Ref	PDN[a] State	If Unused	Description/Notes
RUN and Boot Control (see text for usage guide)					
RUN	I	3V3[b]	Pull High	Leave open	Has internal 10k pull up
EMMC_DISABLE_N	I	3V3[b]	Pull High	Leave open	Has internal 10k pull up
EMMC_EN_N_1V8	O	1V8	Pull High	Leave open	Has internal 2k2 pull up
GPIO					
GPIO[27:0]	I/O	GPIO0-27_VDD	Pull or Hi-Z[c]	Leave open	GPIO Bank 0
GPIO[45:28]	I/O	GPIO28-45_VDD	Pull or Hi-Z[c]	Leave open	GPIO Bank 1
Primary SD Interface[d,e]					
SDX_CLK	O	SDX_VDD	Pull High	Leave open	Primary SD interface CLK
SDX_CMD	I/O	SDX_VDD	Pull High	Leave open	Primary SD interface CMD
SDX_Dx	I/O	SDX_VDD	Pull High	Leave open	Primary SD interface DATA
USB Interface					
USB_Dx	I/O	-	Z	Leave open	Serial interface
USB_OTGID	I	3V3		Tie to GND	OTG pin detect
HDMI Interface					
HDMI_SCL	I/O	3V3[b]	Z[f]	Leave open	DDC Clock (5.5V tolerant)
HDMI_SDA	I/O	3V3[b]	Z[f]	Leave open	DDC Data (5.5V tolerant)
HDMI_CEC	I/O	3V3	Z	Leave open	CEC (has internal 27k pull up)
HDMI_CLKx	O	-	Z	Leave open	HDMI serial clock
HDMI_Dx	O	-	Z	Leave open	HDMI serial data
HDMI_HPD_N_1V8	I	1V8	Pull High	Leave open	HDMI hotplug detect
CAM0 (CSI0) 2-lane Interface					
CAM0_Cx	I	-	Z	Leave open	Serial clock
CAM0_Dx	I	-	Z	Leave open	Serial data
CAM1 (CSI1) 4-lane Interface					
CAM1_Cx	I	-	Z	Leave open	Serial clock
CAM1_Dx	I	-	Z	Leave open	Serial data
DSI0 (Display 0) 2-lane Interface					
DSI0_Cx	O	-	Z	Leave open	Serial clock
DSI0_Dx	O	-	Z	Leave open	Serial data
DSI1 (Display 1) 4-lane Interface					
DSI1_Cx	O	-	Z	Leave open	Serial clock
DSI1_Dx	O	-	Z	Leave open	Serial data
TV Out					
TVDAC	O	-	Z	Leave open	Composite video DAC output
JTAG Interface					
TMS	I	3V3	Z	Leave open	Has internal 50k pull up
TRST_N	I	3V3	Z	Leave open	Has internal 50k pull up
TCK	I	3V3	Z	Leave open	Has internal 50k pull up
TDI	I	3V3	Z	Leave open	Has internal 50k pull up
TDO	O	3V3	O	Leave open	Has internal 50k pull up

Table 3: Pin Functions

[a] The PDN column indicates power-down state (when RUN pin LOW)
[b] Must be driven by an open-collector driver
[c] GPIO have software enabled pulls which keep state over power-down
[d] Only available on Lite variants
[e] The CM will always try to boot from this interface first
[f] Requires external pull-up resistor to 5V as per HDMI spec

ELECTRICAL SPECIFICATION

Caution! Stresses above those listed in Table 4 may cause permanent damage to the device. This is a stress rating only; functional operation of the device under these or any other conditions above those listed in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Symbol	Parameter	Minimum	Maximum	Unit
VBAT	Core SMPS Supply	-0.5	6.0	v
3V3	3V3 Supply Voltage	-0.5	4.10	v
1V8	1V8 Supply Voltage	-0.5	2.10	v
VDAC	TV DAC Supply	-0.5	4.10	v
GPIO0-27_VDD	GPIO0-27 I/O Supply Voltage	-0.5	4.10	v
GPIO28-45_VDD	GPIO28-27 I/O Supply Voltage	-0.5	4.10	v
SDX_VDD	Primary SD/eMMC Supply Voltage	-0.5	4.10	v

Table 4: Absolute Maximum Ratings

DC Characteristics are defined in Table 5

Symbol	Parameter	Conditions	Minimum	Typical	Maximum	Unit
V_{IL}	Input low voltage[a]	VDD_IO = 1.8V VDD_IO = 2.7V	- -	- -	0.6 0.8	V V
V_{IH}	Input high voltage[a]	VDD_IO = 1.8V VDD_IO = 2.7V	1.0 1.3	- -	- -	V V
I_{IL}	Input leakage current	TA = +85°C	-	-	5	μA
C_{IN}	Input capacitance	-	-	5	-	pF
V_{OL}	Output low voltage[b]	VDD_IO = 1.8V, IOL = -2mA VDD_IO = 2.7V, IOL = -2mA	- -	- -	0.2 0.15	V V
V_{OH}	Output high voltage[b]	VDD_IO = 1.8V, IOH = 2mA VDD_IO = 2.7V, IOH = 2mA	1.6 2.5	- -	- -	V V
I_{OL}	Output low current[c]	VDD_IO = 1.8V, VO = 0.4V VDD_IO = 2.7V, VO = 0.4V	12 17	- -	- -	mA mA
I_{OH}	Output high current[c]	VDD_IO = 1.8V, VO = 1.4V VDD_IO = 2.7V, VO = 2.3V	10 16	- -	- -	mA mA
R_{PU}	Pullup resistor	-	50	-	65	kΩ
R_{PD}	Pulldown resistor	-	50	-	65	kΩ

Table 5: DC Characteristics

[a] Hysteresis enabled
[b] Default drive strength (8mA)
[c] Maximum drive strength (16mA)

ELECTRICAL SPECIFICATION

AC Characteristics are defined in Table 6 and Fig. 5.

Pin Name	Symbol	Parameter	Minimum	Typical	Maximum	Unit
Digital outputs	t_{rise}	10-90% rise time[a]	-	1.6	-	ns
Digital outputs	t_{fall}	90-10% fall time[a]	-	1.7	-	ns
GPCLK	t_{JOSC}	Oscillator-derived GPCLK cycle-cycle jitter (RMS)	-	-	48	ps
GPCLK	t_{JPLL}	PLL-derived GPCLK cycle-cycle jitter (RMS)	-	-	20	ps

Table 6: Digital I/O Pin AC Characteristics

[a] Default drive strength, CL = 5pF, VDD IOx = 3.3V

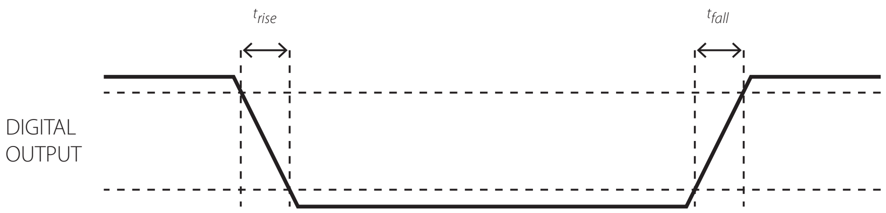


Figure 5: Digital IO Characteristics

POWER SUPPLIES

The Compute Module has six separate supplies that must be present and powered at all times; you cannot leave any of them unpowered, even if a specific interface or GPIO bank is unused. The six supplies are as follows:

1. VBAT is used to power the BCM283x processor core. It feeds the SMPS that generates the chip core voltage.
2. 3V3 powers various BCM283x PHYs, IO and the eMMC Flash.
3. 1V8 powers various BCM283x PHYs, IO and SDRAM.
4. VDAC powers the composite (TV-out) DAC.
5. GPIO0-27 VREF powers the GPIO 0-27 IO bank.
6. GPIO28-45 VREF powers the GPIO 28-45 IO bank.

Supply	Descripion	Minimum	Typical	Maximum	Unit
VBAT	Core SMPS Supply	2.5	-	5.0+5%	V
3V3	3V3 Supply Voltage	3.3-5%	3.3	3.3+5%	V
1V8	1V8 Supply Voltage	1.8-5%	1.8	1.8+5%	V
VDAC	TV DAC Supply[a]	2.5-5%	2.8	3.3+5%	V
GPIO0-27_VDD	GPIO0-27 I/O Supply Voltage	1.8-5%	-	3.3+5%	V
GPIO28-45_VDD	GPIO28-27 I/O Supply Voltage	1.8-5%	-	3.3+5%	V
SDX_VDD	Primary SD/eMMC Supply Voltage	1.8-5%	-	3.3+5%	V

Table 7: Power Supply Operating Ranges

[a] Requires a clean 2.5-2.8V supply if TV DAC is used, else connect to 3V3

Supply Sequencing

Supplies should be staggered so that the highest voltage comes up first, then the remaining voltages in descending order. This is to avoid forward biasing internal (on-chip) diodes between supplies, and causing latch-up. Alternatively supplies can be synchronised to come up at exactly the same time as long as at no point a lower voltage supply rail voltage exceeds a higher voltage supply rail voltage.

POWER SUPPLIES

Power Requirements

Exact power requirements will be heavily dependent upon the individual use case. If an on-chip subsys- tem is unused, it is usually in a low power state or completely turned off. For instance, if your application does not use 3D graphics then a large part of the core digital logic will never turn on and need power. This is also the case for camera and display interfaces, HDMI, USB interfaces, video encoders and decoders, and so on.

Powerchain design is critical for stable and reliable operation of the Compute Module. We strongly recommend that designers spend time measuring and verifying power requirements for their particular use case and application, as well as paying careful attention to power supply sequencing and maximum supply voltage tolerance.

Table 8 specifies the recommneded minimum power supply outputs required to power the Compute Module.

Supply	Minimum Requirement	Unit
VBAT (CM1)	2000[a]	mW
VBAT (CM3,3L)	3500[a]	mW
3V3	250	mA
1V8	250	mA
VDAC	25	mA
GPIO0-27_VDD	50[b]	mA
GPIO28-45_VDD	50[b]	mA
SDX VDD	50[b]	mA

Table 8: Mimimum Power Supply Requirements

[a] Recommended minimum. Actual power drawn is very dependent on use-case

[b] Each GPIO can supply up to 16mA, aggregate current per bank must not exceed 50mA

POWER SUPPLIES

Booting

The 4GB eMMC Flash device on CM3 is directly connected to the primary BCM2837 SD/eMMC interface. These connections are not accessible on the module pins. On CM3L this SD interface is available on the SDX_pins.

When initially powered on, or after the RUN pin has been held low and then released, the BCM2837 will try to access the primary SD/eMMC interface. It will then look for a file called bootcode.bin on the primary partition (which must be FAT) to start booting the system. If it cannot access the SD/eMMC device or the boot code cannot be found, it will fall back to waiting for boot code to be written to it over USB; in other words, its USB port is in slave mode waiting to accept boot code from a suitable host.

A USB boot tool is available on Github which allows a host PC running Linux to write the BCM2837 boot code over USB to the module. That boot code then runs and provides access to the SD/eMMC as a USB mass storage device, which can then be read and written using the host PC. Note that a Raspberry Pi can be used as the host machine. For those using Windows a precompiled and packeged tool is available. For more information see here.
The Compute Module has a pin called EMMC_DISABLE_N which when shorted to GND will disable the SD/eMMC interface (by physically disconnecting the SD_CMD pin), forcing BCM2837 to boot from USB. Note that when the eMMC is disabled in this way, it takes a couple of seconds from powering up for the processor to stop attempting to talk to the SD/eMMC device and fall back to booting from USB.

Note that once booted over USB, BCM2837 needs to re-enable the SD/eMMC device (by releasing EMMC_DISABLE_N) to allow access to it as mass storage. It expects to be able to do this by driving the EMMC_EN_N 1V8 pin LOW, which at boot is initially an input with a pull up to 1V8. If an end user wishes to add the ability to access the SD/eMMC over USB in their product, similar circuitry to that used on the Compute Module IO Board to enable/disable the USB boot and SD/eMMC must be used; that is, EMMC_DISABLE_N pulled low via MOSFET(s) and released again by MOSFET, with the gate controlled by EMMC EN N 1V8. Ensure you use MOSFETs suitable for switching at 1.8V (i.e. use a device with gate threshold voltage, Vt, suitable for 1.8V switching).

PERIPHERALS

GPIO

BCM283x has in total 54 GPIO lines in 3 separate voltage banks. All GPIO pins have at least two alternative functions within the SoC. When not used for the alternate peripheral function, each GPIO pin may be set as an input (optionally as an interrupt) or an output. The alternate functions are usually peripheral I/Os, and most peripherals appear twice to allow flexibility on the choice of I/O voltage.

On CM1, CM3 and CM3L bank2 is used on the module to connect to the eMMC device and, on CM3 and CM3L, for an on-board I2C bus (to talk to the core SMPS and control the special function pins). On CM3L most of bank 2 is exposed to allow a user to connect their choice of SD card or eMMC device (if required).

Bank0 and 1 GPIOs are available for general use. GPIO0 to GPIO27 are bank 0 and GPIO28-45 make up bank1. GPIO0-27_VDD is the power supply for bank0 and GPIO28-45_VDD is the power supply for bank1. SDX_VDD is the supply for bank2 on CM3L. These supplies can be in the range 1.8V-3.3V (see Table 7) and are not optional; each bank must be powered, even when none of the GPIOs for that bank are used.

Note that the HDMI_HPD_N 1V8 and EMM_ EN_N 1V8 pins (on CM1 these were called GPIO46_1V8 and GPIO47_1V8 respectively) are 1.8V IO and are used for special functions (HDMI hot plug detect and boot control respectively). Please do not use these pins for any other purpose, as the software for the Compute Module will always expect these pins to have these special functions. If they are unused please leave them unconnected.

All GPIOs except GPIO28, 29, 44 and 45 have weak in-pad pull-ups or pull-downs enabled when the device is powered on. It is recommended to add off-chip pulls to GPIO28, 29, 44 and 45 to make sure they never float during power on and initial boot.

POWER SUPPLIES

GPIO Alternate Functions

GPIO	Default Pull	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5
0	High	SDA0	SA5	PCLK	-	-	-
1	High	SCL0	SA4	DE	-	-	-
2	High	SDA1	SA3	LCD_VSYNC	-	-	-
3	High	SCL1	SA2	LCD_HSYNC	-	-	-
4	High	GPCLK0	SA1	DPI_D0	-	-	ARM_TDI
5	High	GPCLK1	SA0	DPI_D1	-	-	ARM_TDO
6	High	GPCLK2	SOE_N	DPI_D2	-	-	ARM_RTCK
7	High	SPI0_CE1_N	SWE_N	DPI_D3	-	-	-
8	High	SPI0_CE0_N	SD0	DPI_D4	-	-	-
9	Low	SPI0_MISO	SD1	DPI_D5	-	-	-
10	Low	SPI0_MOSI	SD2	DPI_D6	-	-	-
11	Low	SPI0_SCLK	SD3	DPI_D7	-	-	-
12	Low	PWM0	SD4	DPI_D8	-	-	ARM_TMS
13	Low	PWM1	SD5	DPI_D9	-	-	ARM_TCK
14	Low	TXD0	SD6	DPI_D10	-	-	TXD1
15	Low	RXD0	SD7	DPI_D11	-	-	RXD1
16	Low	FL0	SD8	DPI_D12	CTS0	SPI1_CE2_N	CTS1
17	Low	FL1	SD9	DPI_D13	RTS0	SPI1_CE1_N	RTS1
18	Low	PCM_CLK	SD10	DPI_D14	-	SPI1_CE0_N	PWM0
19	Low	PCM_FS	SD11	DPI_D15	-	SPI1_MISO	PWM1
20	Low	PCM_DIN	SD12	DPI_D16	-	SPI1_MOSI	GPCLK0
21	Low	PCM_DOUT	SD13	DPI_D17	-	SPI1_SCLK	GPCLK1
22	Low	SD0_CLK	SD14	DPI_D18	SD1_CLK	ARM_TRST	-
23	Low	SD0_CMD	SD15	DPI_D19	SD1_CMD	ARM_RTCK	-
24	Low	SD0_DAT0	SD16	DPI_D20	SD1_DAT0	ARM_TDO	-
25	Low	SD0_DAT1	SD17	DPI_D21	SD1_DAT1	ARM_TCK	-
26	Low	SD0_DAT2	TE0	DPI_D22	SD1_DAT2	ARM_TDI	-
27	Low	SD0_DAT3	TE1	DPI_D23	SD1_DAT3	ARM_TMS	-

Table 9: GPIO Bank0 Alternate Functions

POWER SUPPLIES

GPIO Alternate Functions

GPIO	Default Pull	ALT0	ALT1	ALT2	ALT3	ALT4	ALT5
28	None	SDA0	SA5	PCM_CLK	FL0	-	-
29	None	SCL0	SA4	PCM_FS	FL1	-	-
30	Low	TE0	SA3	PCM_DIN	CTS0	-	CTS1
31	Low	FL0	SA2	PCM_DOUT	RTS0	-	RTS1
32	Low	GPCLK0	SA1	RING_OCLK	TXD0	-	TXD1
33	Low	FL1	SA0	TE1	RXD0	-	RXD1
34	High	GPCLK0	SOE_N	TE2	SD1_CLK	-	-
35	High	SPI0_CE1_N	SWE_N	-	SD1_CMD	-	-
36	High	SPI0_CE0_N	SD0	TXD0	SD1_DAT0	-	-
37	Low	SPI0_MISO	SD1	RXD0	SD1_DAT1	-	-
38	Low	SPI0_MOSI	SD2	RTS0	SD1_DAT2	-	-
39	Low	SPI0_SCLK	SD3	CTS0	SD1_DAT3	-	-
40	Low	PWM0	SD4	-	SD1_DAT4	SPI2_MISO	TXD1
41	Low	PWM1	SD5	TE0	SD1_DAT5	SPI2_MOSI	RXD1
42	Low	GPCLK1	SD6	TE1	SD1_DAT6	SPI2_SCLK	RTS1
43	Low	GPCLK2	SD7	TE2	SD1_DAT7	SPI2_CE0_N	CTS1
44	None	GPCLK1	SDA0	SDA1	TE0	SPI2_CE1_N	-
45	None	PWM1	SCL0	SLC1	TE1	SPI2_CE2_N	-

Table 10: GPIO Bank1 Alternate Functions

Table 9 and Table 10 detail the default pin pull state and available alternate GPIO functions. Most of these alternate peripheral functions are described in detail in the Broadcom Peripherals Specification document and have Linux drivers available.

POWER SUPPLIES

Secondary Memory Interface (SMI)

The SMI peripheral is an asynchronous NAND type bus supporting Intel mode80 type transfers at 8 or 16 bit widths and available in the ALT1 positions on GPIO banks 0 and 1 (see Table 9 and Table 10). It is not publicly documented in the Broadcom Peripherals Specification but a Linux driver is available in the Raspberry Pi Github Linux repository (bcm2835_smi.c in linux/drivers/misc).

Display Parallel Interface (DPI)

A standard parallel RGB (DPI) interface is available on bank 0 GPIOs. This up-to-24-bit parallel inter- face can support a secondary display. Again this interface is not documented in the Broadcom Peripherals Specification but documentation can be found here.

SD/SDIO Interface

The BCM283x supports two SD card interfaces, SD0 and SD1.

The first (SD0) is a proprietary Broadcom controller that does not support SDIO and is the primary interface used to boot and talk to the eMMC or SDX_x signals.

The second interface (SD1) is standards compliant and can interface to SD, SDIO and eMMC devices; for example on a Raspberry Pi 3 it is used to talk to the on-board BCM43438 WiFi device in SDIO mode.

Both interfaces can support speeds up to 50MHz single ended (SD High Speed Mode).

CSI (MIPI Serial Camera)

Currently the CSI interface is not openly documented and only CSI camera sensors supported by the official Raspberry Pi firmware will work with this interface. Supported sensors are the OmniVision OV5647 and Sony IMX219.

It is recommended to attach other cameras via USB.

DSI (MIPI Serial Display)

Currently the DSI interface is not openly documented and only DSI displays supported by the official Raspberry Pi firmware will work with this interface.

Displays can also be added via the parallel DPI interface which is available as a GPIO alternate function - see Table 9 and Section 9.1.3

POWER SUPPLIES

USB

The BCM283x USB port is On-The-Go (OTG) capable. If using either as a fixed slave or fixed master, please tie the USB_OTGID pin to ground.

The USB port (Pins USB_DP and USB_DM) must be routed as 90 ohm differential PCB traces.

Note that the port is capable of being used as a true OTG port however there is no official documentation. Some users have had success making this work.

HDMI

BCM283x supports HDMI V1.3a.

It is recommended that users follow a similar arrangement to the Compute Module IO Board circuitry for HDMI output.

The HDMI CK_P/N (clock) and D0-D2_P/N (data) pins must each be routed as matched length 100 ohm differential PCB traces. It is also important to make sure that each differential pair is closely phase matched. Finally, keep HDMI traces well away from other noise sources and as short as possible.

Failure to observe these design rules is likely to result in EMC failure.

Composite (TV Out)

The TVDAC pin can be used to output composite video (PAL or NTSC). Please route this signal away from noise sources and use a 75 ohm PCB trace.

Note that the TV DAC is powered from the VDAC supply which must be a clean supply of 2.5-2.8V. It is recommended users generate this supply from 3V3 using a low noise LDO.

If the TVDAC output is not used VDAC can be connected to 3V3, but it must be powered even if the TV-out functionality is unused.

POWER SUPPLIES

Thermals

The BCM283x SoC employs DVFS (Dynamic Voltage and Frequency Scaling) on the core voltage. When the processor is idle (low CPU utilisation), it will reduce the core frequency and voltage to reduce current draw and heat output. When the core utilisation exceeds a certain threshold the core votlage is increased and the core frequency is boosted to the maximum working frerquency. The voltage and frequency are throttled back when the CPU load reduces back to an 'idle' level OR when the silicon temperature as mesured by the on-chip temperature sensor exceeds 85C (thermal throttling).

A designer must pay careful attention to the thermal design of products using the CM3/CM3L so that performance is not artificially curtailed due to the processor thermal throttling, as the Quad ARM complex in the BCM2837 can generate significant heat output.

Temperature Range

The operating temperature range of the module is set by the lowest maximum and highest minimum of any of the components used.

The eMMC and LPDDR2 have the narrowest range, these are rated for -25 to +80 degrees Celsius. Therefore the nominal range for the CM3 and CM3L is -25C to +80C.

However, this range is the maximum for the silicon die; therefore, users would have to take into account the heat generated when in use and make sure this does not cause the temperature to exceed 80 degrees Celsius.

Availability

Raspberry Pi guarantee availability of CM1, CM3 and CM3 Lite until at least January 2023.

Support

For support please see the hardware documentation section of the Raspberry Pi website and post questions to the Raspberry Pi forum.

ANEXO E

1. ALGORITMO, COMUNICACIÓN CON EL OXÍMETRO DE PULSO

A continuación, se muestra el código para la comunicación con el oxímetro de pulso.

```
import time
import serial, sys
from time import sleep

import time
import board
import busio
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn
import digitalio
import pwmio
import MAX30100

mx30 = MAX30100.MAX30100()
mx30.enable_spo2()
mx30.read_sensor()

mx30.ir, mx30.red

hb = int(mx30.ir / 100)
spo2 = int(mx30.red / 100)

if mx30.ir != mx30.buffer_ir:
    print("{:>2}\t{:>3}".format("RC:", hb))
if mx30.red != mx30.buffer_red:
    print("{:>5}\t{:>3}".format("SPO2:", spo2))
time.sleep(2)
print("OXÍMETRO DE PULSO LISTO")
```

2. ALGORITMO DE LA CELDA DE OXÍGENO

A continuación, se muestra el código para la comunicación con la celda de oxígeno.

```
import time
import board
import busio
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn

# Creación del bus I2C en la RPI
i2c = busio.I2C(board.SCL, board.SDA)

# Creación de la variable con el ADC ADS1115
```

```
ads = ADS.ADS1115(i2c)

# Creación de variable con la entrada P0 del ADS1115
chan = AnalogIn(ads, ADS.P0)

print("{:>5}\t{:>5}\t{:>5}".format("raw", "v","CELDA O2"))

while True:
    #creación de variable para cálculo del FiO2
    OOM = 20.9*((chan.voltage) / 0.010)
    print("{:>5}\t{:>5.5f}\t{:>5.5f}".format(chan.value, chan.voltage,OOM))
    time.sleep(2)
```

3. ALGORITMO, CONTROL DE VÁLVULAS

```
import time
import serial, sys
from time import sleep

import time
import board
import busio
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn
import digitalio
import pwmio
import MAX30100


PWM1 = pwmio.PWMOut(board.D18, frequency=4, duty_cycle=0) # Creación de PWM
con frecuencia 0.25 Hz
PWM2 = pwmio.PWMOut(board.D13, frequency=1, duty_cycle=0 ) # Creación de PWM
con frecuencia 0.25 Hz


def control_motores(FiO2):
    O2 = (50 * (FiO2 - 0.21)) / 0.79  # Porcentaje de ciclo de trabajo para O2
    Aire = 50 - O2  # Porcentaje de ciclo de trabajo para aire

    PWM1.duty_cycle = O2*(65535/100)  # Ciclo de trabajo
    PWM2.duty_cycle = Aire*(65535/100) # Ciclo de trabajo
```

4. ALGORITMO PARA COMUNICACIÓN CON LA PLATAFORMA THINGSBOARD

```
import time
import MAX30100
```

```
import paho.mqtt.client as mqtt
import json

THINGSBOARD_HOST = '192.168.100.139'
ACCESS_TOKEN = '123456789'
# Data capture and upload interval in seconds.
INTERVAL=2

sensor_data = {'freq': 0, 'spO2': 0}

next_reading = time.time()

client = mqtt.Client()

# Set access token
client.username_pw_set(ACCESS_TOKEN)

# Connect to ThingsBoard using default MQTT port and 60 seconds keepalive
interval
client.connect(THINGSBOARD_HOST, 1883, 60)

client.loop_start()

#Initialize sensor
mx30 = MAX30100.MAX30100()
mx30.enable_spo2()
try:
    while True:

        mx30.read_sensor()
        mx30.ir, mx30.red

        hb = int(mx30.ir / 100)
        spo2 = int(mx30.red / 100)

        if mx30.ir != mx30.buffer_ir:
            print("Pulse:", hb)
            sensor_data['freq'] = hb

        if mx30.red != mx30.buffer_red:
            print("SPO2:", spo2)
            sensor_data['spO2'] = spo2

        client.publish('v1/devices/me/telemetry', json.dumps(sensor_data), 1)

        next_reading += INTERVAL
        sleep_time = next_reading-time.time()
        if sleep_time > 0:
            time.sleep(sleep_time)
except KeyboardInterrupt:
    pass

client.loop_stop()
client.disconnect()
```

5. ALGORITMO COMPLETO DE DISPOSITIVOS Y COMPORTAMIENTO DEL SISTEMA

```
import time
import serial, sys

from time import sleep

import time
import board
import busio
import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn
import digitalio
import pwmio
import MAX30100

listaSPO2_PAO2 = [
   {
      'SpO2_t':100,
      'PaO2_t':100,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':99,
      'PaO2_t':96,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':98,
      'PaO2_t':92,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':97,
      'PaO2_t':88,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':96,
      'PaO2_t':84,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':95,
      'PaO2_t':80,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':94,
      'PaO2_t':76,
        'FiO2_t':0.21
   },
    {
      'SpO2_t':93,
      'PaO2_t':72,
```

```
     'FiO2_t':0.21
},
{
   'SpO2_t':92,
   'PaO2_t':68,
     'FiO2_t':0.21
},
 {
   'SpO2_t':91,
   'PaO2_t':64,
     'FiO2_t':0.28
},
 {
   'SpO2_t':90,
   'PaO2_t':60,
     'FiO2_t':0.28
},
 {
   'SpO2_t':89,
   'PaO2_t':58.5,
     'FiO2_t':0.28
},
{
   'SpO2_t':88,
   'PaO2_t':57,
     'FiO2_t':0.28
},
 {
   'SpO2_t':87,
   'PaO2_t':55.5,
     'FiO2_t':0.32
},
 {
   'SpO2_t':86,
   'PaO2_t':54,
     'FiO2_t':0.32
},
 {
   'SpO2_t':85,
   'PaO2_t':52.5,
     'FiO2_t':0.32
},
 {
   'SpO2_t':84,
   'PaO2_t':51,
     'FiO2_t':0.4
},
 {
   'SpO2_t':83,
   'PaO2_t':49.5,
     'FiO2_t':0.4
},
 {
   'SpO2_t':82,
   'PaO2_t':48,
     'FiO2_t':0.6
},
 {
   'SpO2_t':81,
```

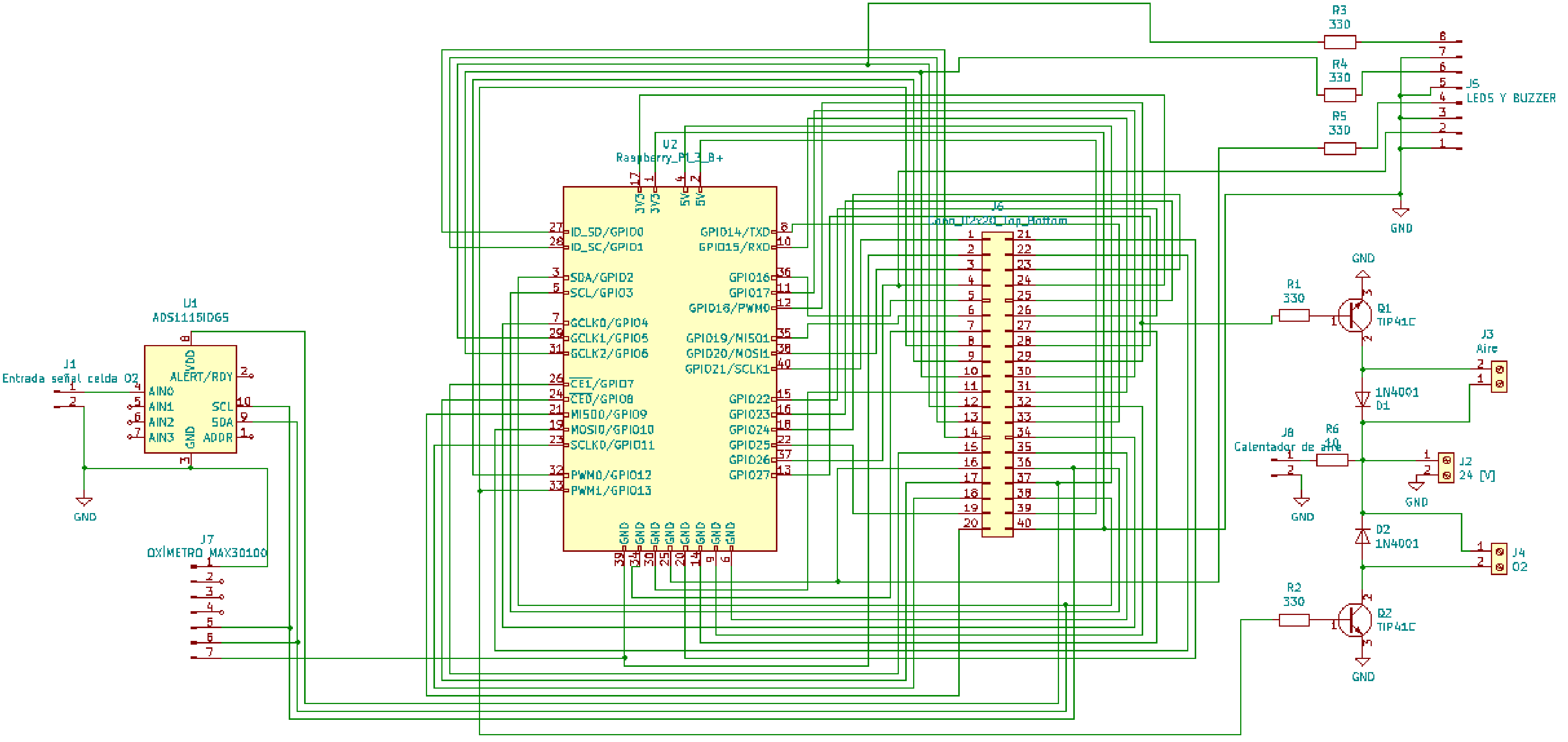
```
      'PaO2_t':46.5,
        'FiO2_t':0.6
   },
    {
      'SpO2_t':80,
      'PaO2_t':45,
        'FiO2_t':0.6
   },

]
# timeout para acabar una lectura y pasar a otro estado
timeout = 1

# Inicializar GPIO
LED_ROJO = digitalio.DigitalInOut(board.D19)
LED_ROJO.direction = digitalio.Direction.OUTPUT
LED_AZUL = digitalio.DigitalInOut(board.D6)
LED_AZUL.direction = digitalio.Direction.OUTPUT
LED_VERDE = digitalio.DigitalInOut(board.D5)
LED_VERDE.direction = digitalio.Direction.OUTPUT
BUZZER = digitalio.DigitalInOut(board.D11)
BUZZER.direction = digitalio.Direction.OUTPUT


PWM1 = pwmio.PWMOut(board.D18, frequency=4, duty_cycle=0) # Creación de PWM
con frecuencia 0.25 Hz
PWM2 = pwmio.PWMOut(board.D13, frequency=1, duty_cycle=0 ) # Creación de PWM
con frecuencia 0.25 Hz
# Creación del bus I2C en la RPI
i2c = busio.I2C(board.SCL, board.SDA)

# Creación de la variable con el ADC ADS1115
ads = ADS.ADS1115(i2c)

# Creación de variable con la entrada P0 del ADS1115
chan = AnalogIn(ads, ADS.P0)

print("{:>5}\t{:>5}\t{:>5}".format("raw", "v","CELDA O2"))

#creación de variable para cálculo del FiO2
OOM = 21*((chan.voltage) / 0.010)
print("{:>5}\t{:>5.3f}\t{:>5.3f}".format(chan.value, chan.voltage,OOM))
time.sleep(2)
print("CELDA OXIGENO LISTA")

# Creación de la variable I2C PULSOXIMETRO
mx30 = MAX30100.MAX30100()
mx30.enable_spo2()
mx30.read_sensor()

mx30.ir, mx30.red

hb = int(mx30.ir / 100)
spo2 = int(mx30.red / 100)

if mx30.ir != mx30.buffer_ir:
    print("{:>2}\t{:>3}".format("RC:", hb))
if mx30.red != mx30.buffer_red:
    print("{:>5}\t{:>3}".format("SPO2:", spo2))
```

```
time.sleep(2)
print("OXÍMETRO DE PULSO LISTO")


def loop(timeout):
    cont = 0
    conectado = 0
    timeout_start = time.time()
    while time.time() < timeout_start + timeout:

        mx30.read_sensor()

        mx30.ir, mx30.red

        hb = int(mx30.ir / 100)
        spo2 = int(mx30.red / 100)
        spo2 = spo2/1.4
        if spo2 >= 100:
            spo2 = 99
        if mx30.ir != mx30.buffer_ir:
            print("Pulse:", hb);
        if mx30.red != mx30.buffer_red:
            print("SPO2:", int(spo2));


        if hb != 0:
            conectado = 1
            LED_VERDE.value = True
            if (cont == 0):
                rc_inicial = hb
                spO2_inicial = spo2
            rc = hb
            spO2 = spo2
            if int(spo2)>= 80:
                for item in listaSPO2_PAO2:
                    if item["SpO2_t"] == int(spO2):

                        print("FiO2 calculado:", item["FiO2_t"])

                        print("{:>9.9}\t{:>5.3}".format("Celda O2:", OOM))
                        fiO2 = item["FiO2_t"]
                print("Medición completada!")
                cont += 1
            else:
                conectado = 0
                spO2 = -1
                spO2_inicial = -1
                fiO2 = -1
                print("SpO2 muy bajo, EMERGENCIA, PACIENTE REQUIERE
VENTILACIÓN MECÁNICA")
                LED_ROJO.value = True
                BUZZER.value = True
        else:
            conectado = 0
            spO2 = -1
            spO2_inicial = -1
            fiO2 = -1
            print("No hay datos recibidos. Verificar conexión del oxímetro de
pulso o del paciente")
```

```
                LED_ROJO.value = True
                BUZZER.value = True

            time.sleep(5)
        return spO2, spO2_inicial, fiO2, conectado

    def control_motores(FiO2):
        O2 = (50 * (FiO2 - 0.21)) / 0.79  # Porcentaje de ciclo de trabajo para O2
        Aire = 50 - O2  # Porcentaje de ciclo de trabajo para aire

        PWM1.duty_cycle = O2*(65535/100)  # Ciclo de trabajo
        PWM2.duty_cycle = Aire*(65535/100) # Ciclo de trabajo


    LED_ROJO.value = True
    LED_AZUL.value = True
    LED_VERDE.value = True
    BUZZER.value = True
    time.sleep(2)
    LED_ROJO.value = False
    LED_AZUL.value = False
    LED_VERDE.value = False
    BUZZER.value = False
    time.sleep(2)

    sys.stdout.write("Iniciando programa de Control \n")
    sys.stdout.flush()
    conectado = 0
    while conectado == 0:
        spO2, spO2_inicial, fiO2, conectado = loop(timeout)

    control_motores(fiO2)
    FiO2 = fiO2

    while True:
        timeout = 10 #poner minutos #60 * 10   # [seconds]
        spO2, spO2_inicial, fiO2, conectado = loop(timeout)
        LED_ROJO.value = False
        LED_AZUL.value = False
        LED_VERDE.value = False
        BUZZER.value = False
        if (conectado != 0):
            if(int(spO2) < 80): # Ventilación mecánica
                print("Paciente requiere ventilación mecánica")
                FiO2 = fiO2
                LED_ROJO.value = True
                LED_AZUL.value = True
                LED_VERDE.value = True
                BUZZER.value = True
                print("FiO2:", fiO2)

            elif(int(spO2) >= 92): # 92% límite inferior del paciente
                print("Saturación Normal")
                FiO2 = fiO2
                print("CASO 1")

            elif(int(spO2)>int(spO2_inicial)): # paciente mejoró
                print("Paciente evoluciona adecuadamente")
                FiO2 = fiO2
```

```
    print("CASO 2")

elif(int(spO2)>=88):
    FiO2 = 0.32
    LED_AZUL.value = True

    print("Paciente recibiendo mayor FiO2 que la inicial ")
    print("CASO 3")
    print("Nuevo valor FiO2:", FiO2)

elif(int(spO2)>=85):
    FiO2 = 0.36
    LED_AZUL.value = True
    print("Paciente recibiendo mayor FiO2 que la inicial ")
    print("CASO 4")
    print("Nuevo valor FiO2:", FiO2)

elif(int(spO2)>=83):
    FiO2 = 0.52
    LED_AZUL.value = True
    print("Paciente recibiendo mayor FiO2 que la inicial ")
    print("CASO 5")
    print("Nuevo valor FiO2:", FiO2)

elif(int(spO2)>=80):
    FiO2 = 0.60
    LED_ROJO.value = True
    LED_AZUL.value = True
    BUZZER.value = True
    print("Paciente requiere ventilación mecánica ")
    print("CASO 6")
    print("Nuevo valor FiO2:", FiO2)

control_motores(FiO2)
```

ANEXO F



Anexo F. Esquemático del sistema

ANEXO G

ENCUESTA PARA LA EVALUACIÓN MÉDICA DEL SISTEMA DE CONTROL AUTOMÁTICO PARA OXIGENOTERAPIA NO INVASIVA DE ALTO FLUJO

Con el propósito de atender las diferentes limitantes durante el tratamiento con oxigenoterapia de alto flujo, como dependencia absoluta de un profesional en la salud; uso de diferentes interfaces de acuerdo a los litros de oxígeno por minuto o FiO2 requerida, como cánula nasal, mascarilla de alto flujo, mascarilla vénturi, muchos de ellos incómodos; se ha construido un SISTEMA DE CONTROL AUTOMÁTICO PARA OXIGENOTERAPIA NO INVASIVA DE ALTO FLUJO , el cual puede ajustar en tiempo real, la FiO2 que recibirá el paciente, de acuerdo a su saturación de oxígeno SpO2 y frecuencia cardíaca.

En la presente encuesta se medirá el criterio de un médico especialista con respecto al sistema, luego de haberlo empleado en un paciente sano.

- Usabilidad:

¿Tuvo dificultad para aprender a utilizar el sistema de control automático de oxigenoterapia de alto flujo?

[✓ 1] 2 3 4 5

¿El rendimiento fue óptimo?

1 2 3 [✓ 4] 5

- Confiabilidad:

Indique el nivel de seguridad del dispositivo según su apreciación

1 2 [✓ 3] 4 5

¿El sistema le ha alertado correctamente cuando ha existido alguna complicación con el paciente?

1 2 3 4 [✓ 5]

- Efectividad:

¿El sistema de control automático de oxigenoterapia de alto flujo le permitió visualizar la información?

1 2 3 4 [✓ 5]

¿El sistema de control automático de oxigenoterapia de alto flujo le permitió visualizar y escuchar alarmas?

1 2 3 4 [✓ 5]

¿La FiO2 real entregada por el dispositivo se ajusta al rango empleado por usted al someter a un paciente a la oxigenoterapia de alto flujo convencional?

1 2 3 4 [✓ 5]

- Utilidad:

¿Considera usted que el sistema puede ser empleado en ambientes hospitalarios?

1 2 3 4 [✓ 5]

¿Considera usted que el sistema puede ser empleado en ambientes no hospitalarios?

1 2 [✓ 3] 4 5

¿Facilitaría el sistema la atención oportuna a un paciente que recibe oxigenoterapia de alto flujo?

1 2 3 4 [✓ 5]

- Aceptación:

¿En las condiciones actuales, sometería a sus pacientes al sistema desarrollado?

1 [✓ 2] 3 4 5

¿En las condiciones actuales, recomendaría el uso del sistema desarrollado?

1 [✓ 2] 3 4 5