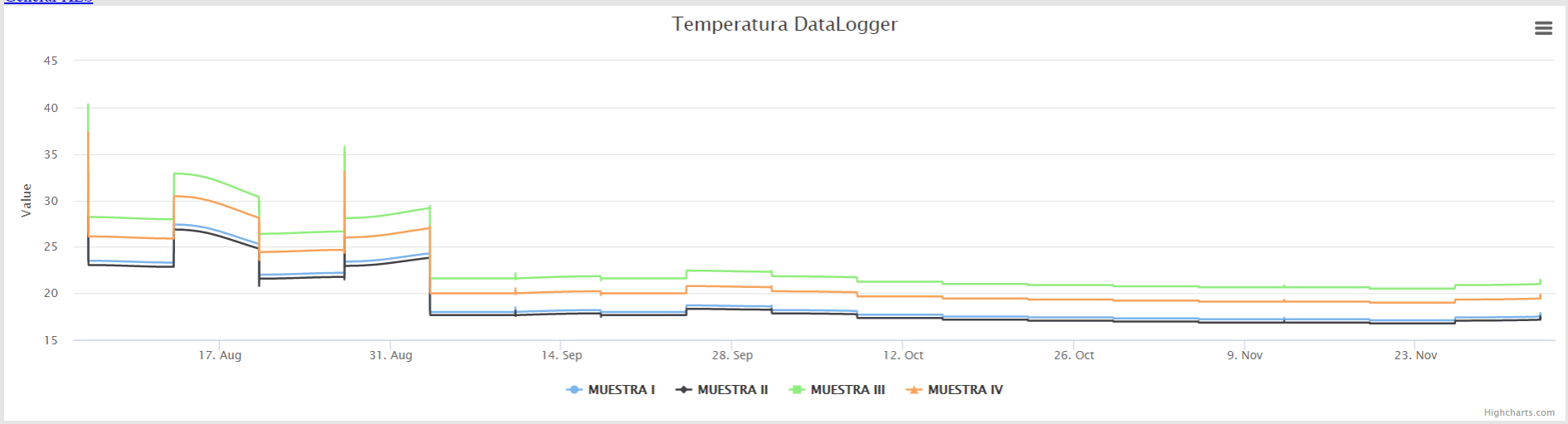
Visualizacion de resultados en el Dashboard (Temperatura)

Figura 56:Visualización Temperatura

TEMPERATURA | LUMINOSIDAD | VERSUS

N	TIEMPO	T. MUESTRA I	T. MUESTRA II	T. MUESTRA III	T. MUESTRA IV
214	2020-12-03 05:55:15	17.7	17.35	21.24	19.67
213	2020-12-03 05:50:15	17.6	17.25	21.12	19.56
212	2020-12-03 05:45:15	17.6	17.25	21.12	19.56
211	2020-12-03 05:40:15	17.5	17.16	21	19.44
210	2020-12-03 05:35:15	17.6	17.25	21.12	19.56

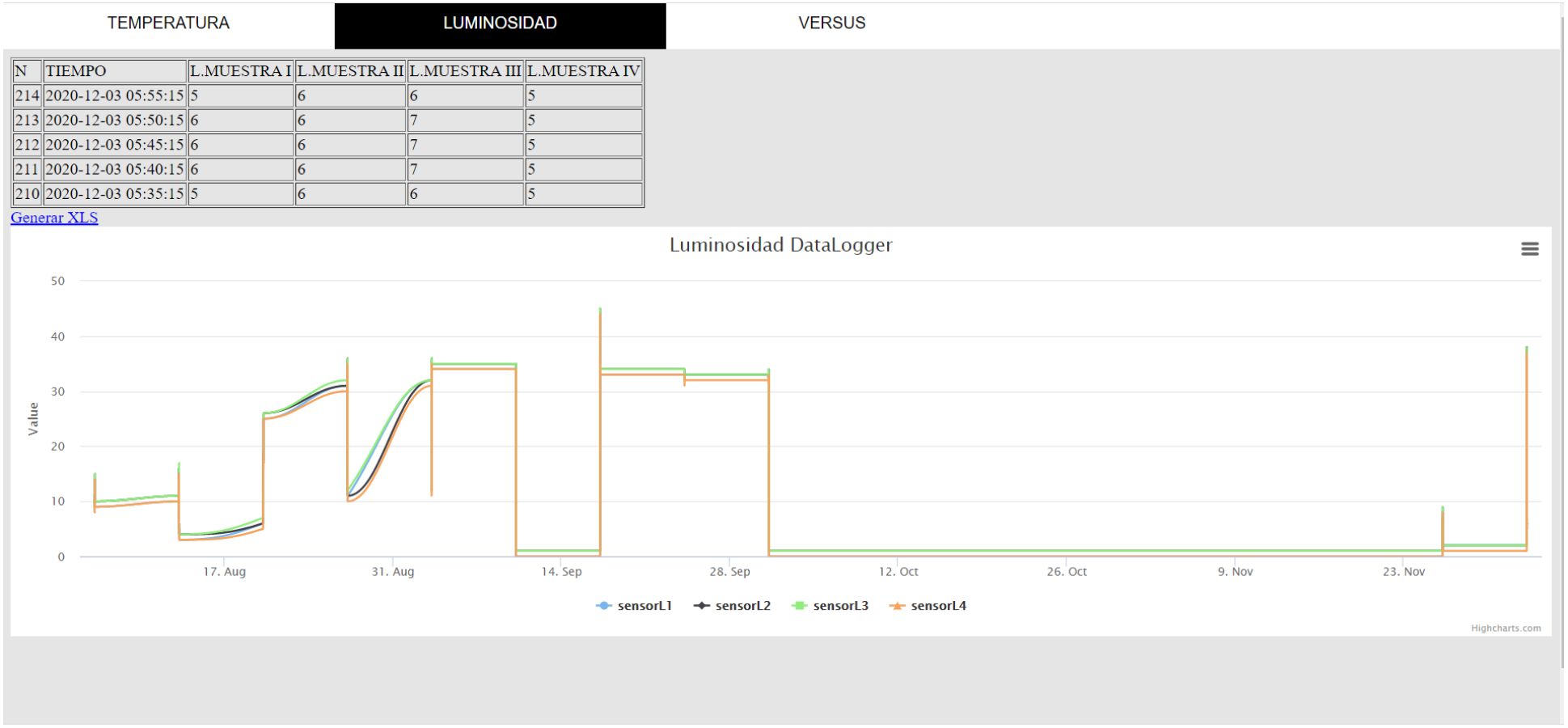
Generar XLS



Fuente 81: (Autoría propia)

Visualizacion de resultados en el Dashboard (Luminosidad)

Figura 57:Visualización Luminosidad



Fuente 82: (Autoría propia)

3.3. Arquitectura e implementación

3.3.1. Implementación del hardware

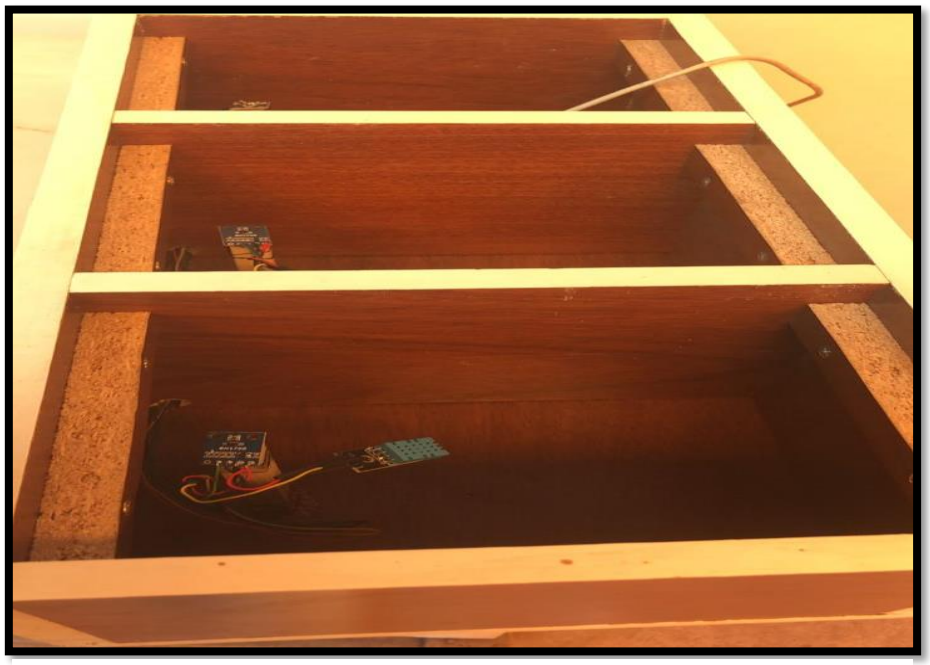
En el desarrollo en implementación de la herramienta Dashboard se ha creado una infraestructura física (Hardware) que hace posible la captura de datos y está compuesto de componentes electrónicos que están distribuidos en un circuito electrónico de funcionamiento que a continuación es mostrado:

Descripción de dispositivos electrónicos, y componentes del Dashboard:

1. Nodemcu
2. DHT11.PCB: Sensor humedad temperatura. Cantidad (04)
3. BH1750.PCB: Sensor de Luz. Cantidad (04)
4. Multiplexor UX: MM74HC4053 Cantidad (1/2)
5. Multiplexor UY: MM74HC4053 Cantidad (1/2)
6. Cables de circuito FTP (10 mts)
7. Rack de malamina (1 plancha)
8. Tornillos corrugados (50 unidades)

Foto: Dashboard

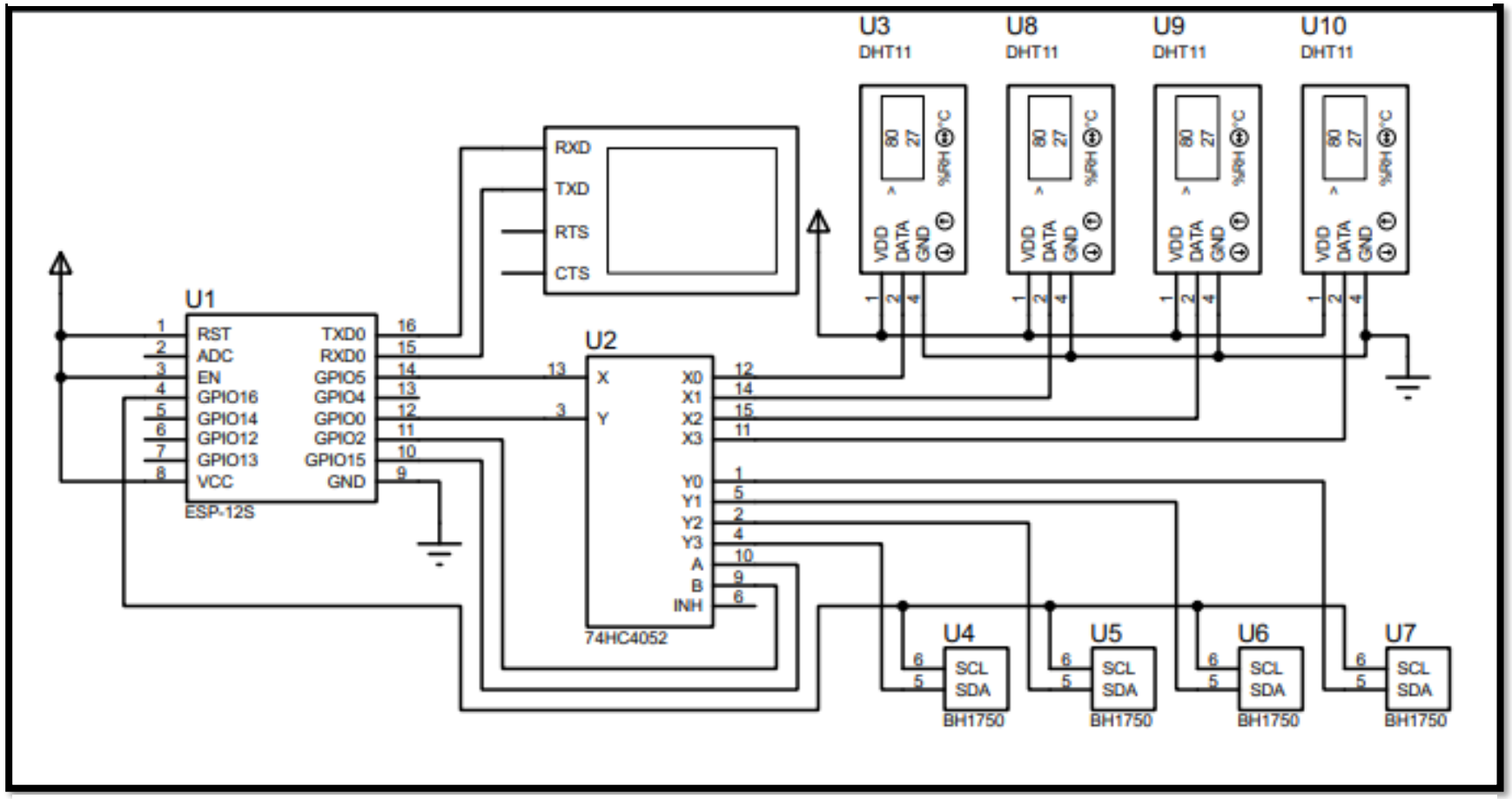
Figura 58:Foto física del Dashboard



Fuente 83:(Autoría propia)

Plano: circuito electrónico Dashboard

Figura 59:Plano circuito electrónico Dashboard



Fuente 84:(Autoría propia)

Capítulo 4: Resultados

Después de realizar las diferentes pruebas en el funcionamiento de la herramienta de monitoreo y control de datos (Dashboard) se observó los siguientes resultados:

4.1. Comprobación de la prospectiva

- Se desarrolló una herramienta de monitoreo y control de datos en un entorno de funcionamiento web que automatiza la visualización de los datos proporcionados por los sensores de medición de LUMINOSIDAD Y TEMPERATURA en tiempo real denominado (Dashboard).
- La herramienta de monitoreo y control de datos (Dashboard) utiliza dispositivos electrónicos a la vanguardia con tecnología de conexión de sensores MCI y protocolo serial de periférico MCU del mundo digital dentro de su infraestructura de hardware.
- La herramienta de monitoreo y control de datos (Dashboard) utiliza código de reutilizable de los FRAMEWORK proporcionados por GOOGLE
- La herramienta de monitoreo y control de datos (Dashboard) realiza reportes específicos en formatos XLS, PDF que sirven de ayuda para el usuario que realiza la medición de datos.
- La herramienta de monitoreo y control de datos (Dashboard) realiza el almacenamiento referente a los datos generados por los sensores de medición en un dominio privado a su alcance.
- La herramienta de monitoreo y control de datos (Dashboard) reduce considerablemente el tiempo de monitoreo de datos, ya que al poder hacer mediciones simultaneas de diferentes sustancias ahorra el tiempo de uso del aparato.

4.2. Cumplimiento de objetivos

El objetivo principal de esta investigación de desarrollo aplicada fue la implementación de una herramienta de software-hardware "Dashboard" de captura de datos simultáneos en condiciones homogéneas de las muestras de los materiales de construcción, para el monitoreo de la luminosidad y temperatura en tiempo real en el laboratorio de Concreto y Materiales de la EPIC. Lo cual se logró en los ámbitos propuestos de “software y hardware”.

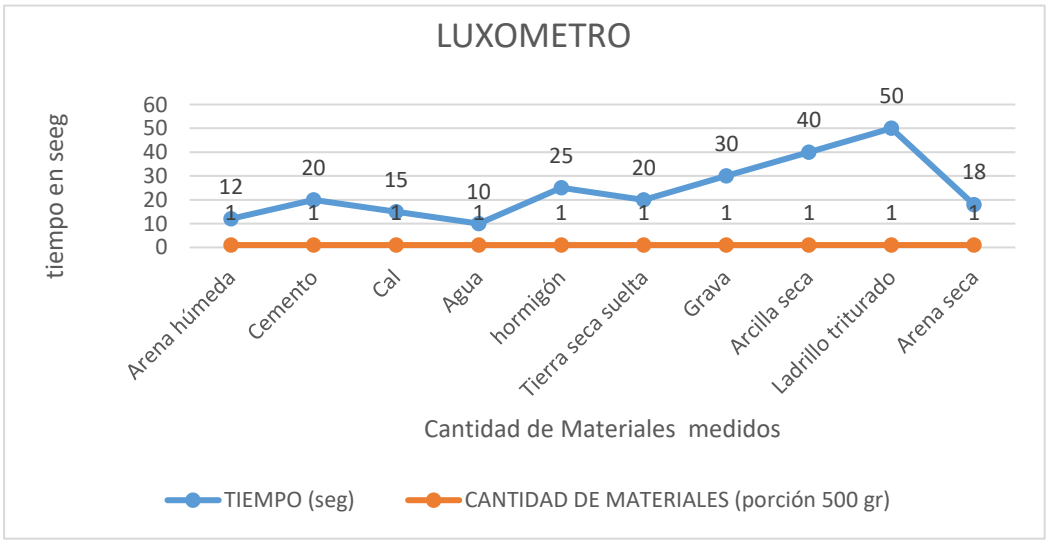
Se ha elaborado un cuadro de comparación de uso entre el Luxómetro y la herramienta de monitoreo y control de datos (Dashboard), la medición de comparación fue el tiempo que utiliza cada sistema para determinar la luminosidad y la temperatura, porque esta variable de tiempo fue propuesta en el objetivo de la investigación.

Tabla 26:CAPTURA LUXÓMETRO DE VARIABLES LUMINOSIDAD TEMPERATURA

LUXOMETRO (LUMINOSIDAD, TEMPERATURA)		
MATERIALES	TIEMPO (seg)	CANTIDAD DE MATERIALES (porción 500 gr)
Arena húmeda	12	1
Cemento	20	1
Cal	15	1
Agua	10	1
hormigón	25	1
Tierra seca suelta	20	1
Grava	30	1
Arcilla seca	40	1
Ladrillo triturado	50	1
Arena seca	18	1

Fuente 85:(Autoría propia)

Figura 60:Visualización Luxómetro



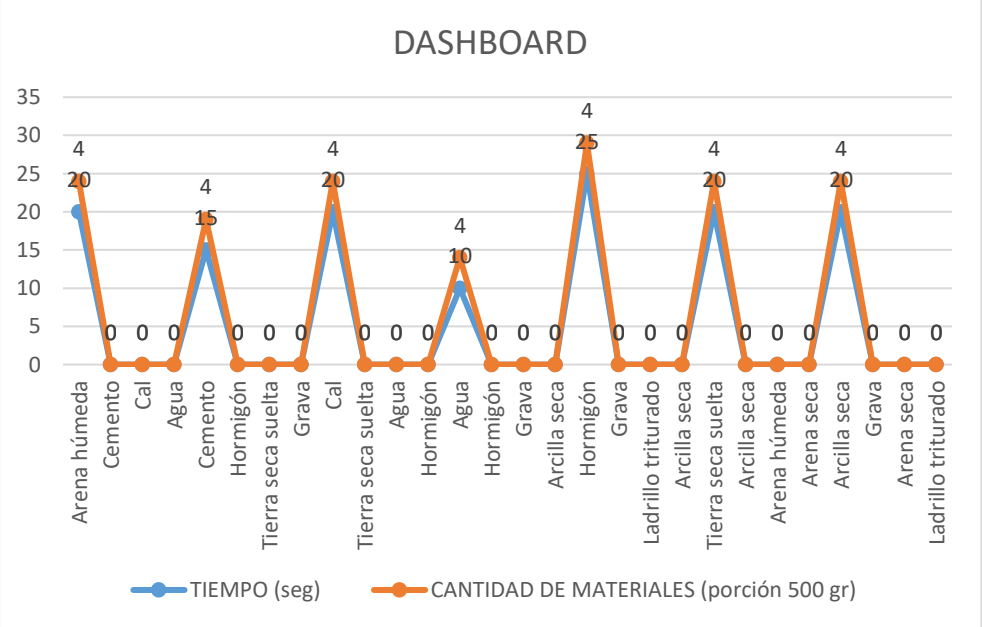
Fuente 86:(Autoría propia

Figura 61:captura Dashboard de Variables luminosidad tempera

DASHBOARD (LUMINOSIDAD, TEMPERATURA)		
MATERIALES	TIEMPO (seg)	CANTIDAD DE MATERIALES (porción 500 gr)
Arena húmeda Cemento Cal Agua	20	4
Cemento Hormigón Tierra seca suelta Grava	15	4
Cal Tierra seca suelta Agua Hormigón	20	4
Agua Hormigón Grava Arcilla seca	10	4
Hormigón Grava Ladrillo triturado Arcilla seca	25	4
Tierra seca suelta Arcilla seca Arena húmeda Arena seca	20	4
Arcilla seca Grava Arena seca Ladrillo triturado	20	4

Fuente 87: (Autoría propia)

Figura 62:Visualización Dashboard



Fuente 88:(Autoría propia)

4.3. Contribuciones (impacto)

- La herramienta de monitoreo y control de datos (Dashboard) esta alojada en un dominio privado y disponible para quien usa el Dashboard. Como hosting web produce un servicio de información constante de todas mediciones almacenadas en el dominio a través de la consulta a una base de datos en tiempo real. Se puede acceder desde cualquier dispositivo electrónico como Celular Tablet Ipad, computadora, etc., proporcionando información estratégica en cualquier momento y en cualquier lugar.
- El objetivo principal de la tesis fue el desarrollo e implantación de la herramienta de monitoreo y control de datos (Dashboard), que tuvo un origen de investigación MULTIDISPLINARIO, ya que esta tesis es el complemento funcional de la tesis de la Escuela Profesional de ingeniería Civil de nombre: "EVALUACIÓN COMPARATIVA DE LAS PROPIEDADES FÍSICAS Y MECÁNICAS DE UN CONCRETO TRADICIONAL, CON RESPECTO A UN CONCRETO TRASLUCIDO, REEMPLAZANDO EL AGREGADO FINO POR VIDRIO MOLIDO EN DIFERENTES PORCENTAJES" presentado por los bachilleres Aucca Cruz, Bryan, Carbajal Arriola, Daniel Enrique.
- La infraestructura del equipo tanto en software como en hardware contribuye al fortalecimiento de los laboratorios de Escuela Profesional de Ingeniería Civil.

Conclusiones

En la elaboración del presente trabajo de investigación de tipo desarrollo aplicado se ha llegado a las siguientes conclusiones:

1. La implementación física y lógica de la herramienta Dashboard, se logró a través de la construcción de una arquitectura física con dispositivos electrónicos que utilizan tecnologías de conexión de sensores MCI y protocoles serial periférico MCU. En la arquitectura de desarrollo de software utilizó distintas herramientas como MVC, Cliente/Servidor, junto a la metodología ágil SCRUM, las cuales nos permitieron alcanzar los objetivos planteados en nuestra investigación.
2. Se diseño un nuevo modelo de negocio en cuanto a la captura de datos simultáneos para la medición de muestras de materiales de construcción. El cual satisface los requerimientos analizados en el modelo de negocio que utiliza el luxómetro como instrumento de medición.
3. Se implementó una herramienta de servicio funcional de arquitectura cliente servidor con alojamiento y servicio de hosting remoto y dominio privado, capaz de brindar información en tiempo real en cualquier momento y desde cualquier lugar que se le solicite. Brindando las credenciales de acceso al Dashboard al jefe del laboratorio del aula 107 de la EPIC.
4. Se diseñó la base de datos en el SGBDD MySQL, con las tablas funcionales y versátiles para el funcionamiento del sistema, y para realizar los diferentes procesos que involucran la visualización de datos capturados por los sensores de temperatura y luminosidad, permitiendo el funcionamiento del sistema sin problemas en cuanto al manejo de datos y desarrollo del aplicativo.
5. Los conceptos, la funcionalidad, la aplicación, los entornos de desarrollo, fueron recopilados de los antecedentes de las tesis mencionadas en el capítulo 2 del marco teórico ya sea en el ámbito nacional e internacional. Aclarando de mejor manera la visión de desarrollo de nuestro Dashboard. Dándole una sostenibilidad operativa cuando lo requiera el estudiante.
6. Al finalizar la investigación me contribuyo y amplio conocimientos referente a las metodologías de desarrollo de software (BPM: RAD y SCRUM), además sobre los formatos de documentación (APA 7ma edición). Así mismo me enseño a trabajar en equipo con tiempos establecidos para las entregas de tareas.

7. En el transcurso del desarrollo del trabajo de investigación, personalmente aprendí el manejo y la aplicación de dos metodologías de desarrollo; una de ellas para el desarrollo de nuevos procesos vinculadas al negocio y la otra metodología para el desarrollo del software de aplicación. Además, ver la formalidad de como se conceptualiza y desarrolla una tesis en la modalidad de aplicación.

Recomendaciones

1. Para el mejoramiento de la herramienta de monitoreo y control de datos (Dashboard) se propone la creación de un framework orientado exclusivamente a la visualización de indicadores de temperatura y luminosidad y estos componentes serán los más idóneos y garantizarán la visualización de los datos como en su segundad.
2. Agregar funcionalidades como la minería datos e inteligencia artificial a fin de evaluar patrones de comportamiento que sirvan para detectar posibles errores en la información encontrada y sugieran cambios oportunos y actualización de variables de funcionamiento en el Dashboard
3. Crear una plataforma digital que pueda integrar a distintos Dashboards la cual permitirá alcanzar información de monitoreo de datos, tratando de introducir el uso del Dashboards para distintos negocios o rubros de producción en la ciudad del Cusco.
4. Dar apoyo funcional a la creación de proyectos tecnológicos de menor costo como el nuestro. Ya que en el mercado comercial un LUXOMETRO cuesta 10 veces más que nuestro proyecto de Investigación de desarrollo aplicada: Dashboard.

Glosario

Acceso: Operación de identificación de un dato, un registro o una entidad que pertenece a un fichero. Proceso necesario para acceder a una instrucción almacenada en memoria para ejecutarla.

Alerta: Hace referencia a una situación de vigilancia o atención. Un estado o una señal de alerta es un aviso para que se extremen las precauciones o se incremente la vigilancia. En nuestro proyecto la alerta se genera a partir de un dispositivo dado como perdido o robado.

Automatización: Sistema tecnológico basado en la ingeniería y la informática, que proporciona una optimización de los procesos productivos mediante la regulación automática (autorreguladores).

Bit: El bit es la unidad mínima de información empleada en informática, en cualquier dispositivo digital, o en la teoría de la información. Con él, podemos representar dos valores cualesquiera, como verdadero o falso, abierto o cerrado, blanco o negro, norte o sur, masculino o femenino, rojo o azul, etc. Basta con asignar uno de esos valores al estado de "apagado" (0), y el otro al estado de "encendido" (1).

Briqueta: Es un producto 100 % ecológico y renovable, catalogado como bioenergía sólida, que viene en forma cilíndrica o de ladrillo y sustituye a la leña con muchas ventajas.

Byte: Es una unidad de información digital equivalente a cuatro bits (binary digit, dígito binario) originalmente y posteriormente como estándar se adoptó que 1 byte equivale a ocho bits. La palabra byte proviene de bite, que significa mordisco, como la cantidad más pequeña de datos que un ordenador podía "morder" a la vez. El símbolo de byte es un B mayúscula, para distinguir de bit, cuyo símbolo es b minúscula. El byte se utiliza generalmente en las áreas de informática y telecomunicaciones, en esta última se denomina comúnmente octeto, que proviene del francés octet, derivado del latín octo y del griego okto, que significa ocho, diferenciando así el byte de 8 bits de otros bytes con diferente equivalencia de bits.

Cloud Storage: El almacenamiento en la nube, del inglés cloud storage, es un modelo de almacenamiento de datos basado en redes de computadoras, ideado en los años 1960,

donde los datos están alojados en espacios de almacenamiento virtualizados, por lo general aportados por terceros.

Colas: Las colas se utilizan en sistemas informáticos, transportes y operaciones de investigación (entre otros), dónde los objetos, personas o eventos son tomados como datos que se almacenan y se guardan mediante colas para su posterior procesamiento. Este tipo de estructura de datos abstracta se implementa en lenguajes orientados a objetos mediante clases, en forma de listas enlazadas.

Dashboard es una representación gráfica de las principales métricas o KPIs que intervienen en la consecución de los objetivos de una estrategia de Inbound Marketing. Esta herramienta nos permite visualizar el problema y favorecer la toma de decisiones orientada a mejorar los posibles errores que podamos estar cometiendo.

Embebido: Un sistema embebido o empotrado (integrado, incrustado) es un sistema de computación diseñado para realizar una o algunas pocas funciones dedicadas, frecuentemente en un sistema de computación en tiempo real. Al contrario de lo que ocurre con los ordenadores de propósito general (como por ejemplo una computadora personal o PC) que están diseñados para cubrir un amplio rango de necesidades, los sistemas embebidos se diseñan para cubrir necesidades específicas.

Formulario: En Internet y sistemas informáticos, los formularios pueden ser llenados de forma online con validación de datos, y son muy útiles para las encuestas, registración de usuarios, ingreso a sistemas, suscripciones, etc. Los formularios por Internet son llamados formularios web, y generalmente son hechos a través de etiquetas HTML, aunque también existen otros medios como Flash, Java, etc.

Ide: Un entorno de desarrollo integrado o entorno de desarrollo interactivo, en inglés Integrated Development Environment (IDE), es una aplicación informática que proporciona servicios integrales para facilitarle al desarrollador o programador el desarrollo de software.

Iteración: En programación, cuando el bloque de instrucciones de un bucle se ejecuta, se dice que se ha producido una iteración.

Jquery: Es una biblioteca multiplataforma de JavaScript, que permite simplificar la manera de interactuar con los documentos HTML, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas web.

Mcu: Un microcontrolador es un circuito integrado que en su interior contiene una unidad central de procesamiento (CPU), unidades de memoria (RAM y ROM), puertos de entrada y salida y periféricos. Estas partes están interconectadas dentro del microcontrolador, y en conjunto forman lo que se le conoce como microcomputadora. Se puede decir con toda propiedad que un microcontrolador es una microcomputadora completa encapsulada en un circuito integrado.

Módulo: Un módulo es un software que agrupa un conjunto de subprogramas y estructuras de datos. Los módulos son unidades que pueden ser compiladas por separado y los hace reusables y permite que múltiples programadores trabajen en diferentes módulos en forma simultánea, produciendo ahorro en los tiempos de desarrollo.

Mvc: Es una propuesta de diseño de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario. Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

Open source: Es el término con el que se conoce al software distribuido y desarrollado libremente. El código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones éticas y morales las cuales destacan en el llamado software libre.

Puerto com: Un puerto serie o puerto en serie es una interfaz de comunicaciones de datos digitales, frecuentemente utilizado por computadoras y periféricos, donde la información es transmitida bit a bit, enviando un solo bit a la vez; en contraste con el puerto paralelo que envía varios bits simultáneamente.

Product owner: El rol del Product Owner, puede venir de parte del cliente o dentro de la empresa misma, depende. Generalmente no se aconseja que el Product Owner sea parte también del equipo de desarrollo, o el Scrum Master mismo, sus intereses se pueden ver enfrentados, pero esto puede variar según el caso. Por defecto, probablemente venga de parte del cliente.

Product backlog: El Product Backlog es el conjunto de requisitos funcionales y no funcionales que debe cumplir el producto una vez entregado. No se requiere que esté completo al momento de su creación, basta con definir aquellos requisitos que se conozcan en su momento y alentar a su crecimiento continuo o su modificación.

Proceso: Es una mínima unidad de ejecución de una tarea determinada, siguiendo un conjunto de instrucciones que ha sido especificado, utilizando un recurso del sistema y partiendo de la base de un estado determinado, para lo cual se obtiene lógicamente un resultado afín al deseado.

Registro: En un fichero de bases de datos, cada uno de los elementos en que se divide. Contiene los datos de un elemento de los descritos en el fichero, dividiendo la información en campos. Generalmente es el contenido de un formulario completo del formato escogido para la recogida de datos.

Reporte: Son informes que organizan y exhiben la información contenida en una base de datos. Su función es aplicar un formato determinado a los datos para mostrarlos por medio de un diseño atractivo y que sea fácil de interpretar por los usuarios.

Rs232: RS-232 (Recommended Standard 232, en español: "Estándar Recomendado 232"), también conocido como EIA/TIA RS-232C, es una interfaz que designa una norma para el intercambio de datos binarios serie entre un DTE (Data Terminal Equipment, "Equipo Terminal de Datos"), como por ejemplo una computadora, y un DCE (Data Communication Equipment, "Equipo de Comunicación de Datos"), por ejemplo, un módem. Existen otros casos en los que también se utiliza la interfaz RS-232. Una definición equivalente publicada por la UIT se denomina V.24.

Rs485: Estándar de comunicaciones multipunto de la EIA. Es una especificación eléctrica (de la capa física en el modelo OSI) de las conexiones half-duplex, two-wire y multipint serial. RS485 sólo especifica características eléctricas de una unidad, pero no especifica o recomienda ningún protocolo de datos.

Rup: El Rational Unified Process o Proceso Unificado de Racional. Es un proceso de ingeniería de software que suministra un enfoque para asignar tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de software de alta y de mayor calidad para satisfacer las necesidades de los usuarios que tienen un cumplimiento al final dentro de un límite de tiempo y presupuesto previsible. Es una metodología de desarrollo iterativo que es enfocada hacia “diagramas de los casos de uso, y manejo de los riesgos y el manejo de la arquitectura” como tal.

Scrum: El Scrum es un proceso de la Metodología Ágil que se usa para minimizar los riesgos durante la realización de un proyecto, pero de manera colaborativa. Entre las

ventajas se encuentran la productividad, calidad y que se realiza un seguimiento diario de los avances del proyecto, logrando que los integrantes estén unidos, comunicados y que el cliente vaya viendo los avances.

Script: Un script, archivo de órdenes, archivo de procesamiento por lotes o, cada vez más aceptado en círculos profesionales y académicos, guion es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano. Los guiones son casi siempre interpretados, pero no todo programa interpretado es considerado un guion. El uso habitual de los guiones es realizar diversas tareas como combinar componentes, interactuar con el sistema operativo o con el usuario. Por este uso es frecuente que los intérpretes de órdenes sean a la vez intérpretes de este tipo de programas.

Sprint: Sprint es el nombre que va a recibir cada uno de los ciclos o iteraciones que vamos a tener dentro de dentro de un proyecto Scrum.

Sdk: (Software Development Kit - Kit de desarrollo de software o devkit). Un SDK es un conjunto de herramientas y programas de desarrollo que permite al programador crear aplicaciones para un determinado paquete de software, estructura de software, plataforma de hardware, sistema de computadora, consulta de videojuego, sistema operativo o similar.

Sensor: Es un dispositivo que es capaz de leer, guardar e identificar datos. Todos los sensores cuentan mínimamente con una pieza que es sensible al tacto, el ambiente, (Que es el sensor en si, aunque luego hacen falta ciertas partes electrónicas) Estos dispositivos se han hecho populares a raíz de que los últimos smartphones y tablets han incorporado dicho sistema pues son los que mayor seguridad aportan.

Servidor: En informática, un servidor es un tipo de software que realiza ciertas tareas en nombre de los usuarios. El término servidor ahora también se utiliza para referirse al ordenador físico en el cual funciona ese software, una máquina cuyo propósito es proveer datos de modo que otras máquinas puedan utilizar esos datos.

Sgbd: (Sistema de gestión de base de datos) o en inglés Database management system (DBMS), es una agrupación de programas que sirven para definir, construir y manipular una base de datos.

Sprint: Es una división de un conjunto de tareas componen un proyecto en un periodo de tiempo. Normalmente los sprints suelen ser de 15 a 30 días). En cada sprint se busca

poder tener una parte del proyecto que sea “entregable” para poder ver la evolución del mismo hasta llegar al producto final.

Trazabilidad: Un conjunto de procedimientos preestablecidos y autosuficientes que permiten conocer el histórico, la ubicación y la trayectoria de un producto, o lote de productos a lo largo de la cadena de suministros, en un momento dado y a través de unas herramientas determinadas

Webservice: Un servicio web (en inglés, web service o web services) es una tecnología que utiliza un conjunto de protocolos y estándares que sirven para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programación diferentes, y ejecutadas sobre cualquier plataforma, pueden utilizar los servicios web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS y W3C son los comités responsables de la arquitectura y reglamentación de los servicios Web.

Referencias

Achour, M. (2013). Manual de PHP. E.E.U.U.

Affiliates, O. C. (2016). *Workbench Reference Manual.* Obtenido de http://dev.mysql.com/

Anexo-06. (s.f.). *Código fuente la herramienta de monitoreo y control de datos.*

Angel Cobo, P. G. (2005). *PHP Y MySQL*. Obtenido de https://www.editdiazdesantos.com/wwwdat/pdf/9788479787066.pdf

Baeza, J. P. (2009). *Manual Arduino*. Obtenido de https://rua.ua.es/dspace/bitstream/10045/11833/1/arduino.pdf

Bilicki, E. &. (s.f.).

Bizagi Limited. (2016). *Bizagi BPM Suite*. Obtenido de Solución para Gestión de procesos de negocio (BPM): http://www.bizagi.com

Cabanillas, A. R. (2010). *DISEÑO DE UN PROTOTIPO DE UN SISTEMA DE TRAZABILIDAD DE GANADO USANDO RFID.* Lima - Perú.

Cádiz, U. d. (2018). *Comenzando con arduino*. Obtenido de http://www.uca.es/recursos/doc/Unidades/Unidad_Innovacion/Innovacion_Docente/ANEXOS_2011_2012/22232441_310201212102.pdf

Casillas Santillan, L., Ginesta, M., & Perez Mora, O. (2009). *Bases de Datos en Mysql.* Barcelona: www.uoc.edu.

Castilla, D. (2014). *A Hybrid Approach Using RUP and Scrum as a Software Development Strategy.* Florida.

Chawla, K. (. (1998). *Composite materials.* New York: Springer.

Choque, J. C. (2019).

corbacho-perez. (s.f.). *anexo .*

Corcuera, P. (s.f.). *Programacion Arduino*. Obtenido de https://personales.unican.es/corcuerp/progucont/slides/Arduino.pdf

CÓRDOVA FORERO, J. A. (2014). *IMPLEMENTACIÓN DE UN SISTEMA DE MATRÍCULAS Y PAGOS PARA EL CENTRO DE INFORMÁTICA DE LA UNIVERSIDAD CÉSAR VALLEJO.* Lima.

Córdova Zamora, M. (2000). *Estadística descriptiva e inferencial* (Cuarta ed.). Lima, Perú: Moshera S.R.L. Recuperado el 21 de 09 de 2017

Delgado, W. M. (2015). *ASP.NET MVC 6 - UNA GUÍA INTRODUCCTORIA.* Obtenido de https://leanpub.com/aspnetmvc6-unaguaintroductoria

Gallego, R., & Sanz, J. C. (2001). Diccionario Akal del color. En R. Gallego, & J. C. Sanz.

García, A. A. (Diciembre de 2007). *Desarrollo de herramientas web de gestión docente* . Obtenido de http://repositorio.upct.es/bitstream/handle/10317/179/pfc2475.pdf

Gardey, J. P. (2012). Definición de temperatura. Mexico.

Gauchat, J. D. (2012). Barcelona: MARCOMBO S.A.

Gauchat, J. D. (s.f.). *El gran libro de HTML 5, CSS3 Y JAVASCRIPT*. Obtenido de https://gutl.jovenclub.cu/wp-content/uploads/2013/10/El+gran+libro+de+HTML5+CSS3+y+Javascrip.pdf

Gianni, R. d. (2011). *El libro del BPM 2011.*

gonzalez gisela, c. c. (s.f.). *Calor y temperatura*. Obtenido de http://www.elgeometra.com.ar/fisica/trabajos%20practicos/2010/B/CALOR.pdf

Group, The PHP. (10 de 05 de 2018). *PHP*. Obtenido de http://php.net/manual/es/preface.php

Huamán, C. E. (2015). *Reyna .*

Jacobson, I., Booch, G., & Rumbaugh, J. (1999). *El proceso unificado de desarrollo de software.* Madrid: Pearson Educación S.A.

Java - Oracle. (2016). *JAVA*. Obtenido de https://www.java.com/es/

Jiménez, R. E. (8 de Enero de 2015). *Metodologías Ágiles de Desarrollo de Software Aplicadas a la*. Obtenido de https://core.ac.uk/download/pdf/80296686.pdf

Kerzner. (2013). Dashboard diseños.

Kniberg, H. (2007). *Scrum y XP desde las trincheras. Como hacemos Scrum.* Nueva York: C4Media, editor de InfoQ.com. .

LUXÓMETRO 1335 . (s.f.). Obtenido de http://www.crana.org/themed/crana/files/docs/033/042/medidornivelilluminacion_3.pdf

M. López et al., ". M. (2009). Proceedings of the.

M., C. M. (2006). *RFID: El código de barras inteligente para Bibliotecas.* Santiago - Chile.

Maria Loureiro, J. P. (julio de 2017). *Taller robotica libre con arduino*. Obtenido de Arduino y el open Source: http://educalab.es/documents/10180/640047/TallerRoboticaLibreArduino.pdf/c77adbfd-606a-4fbe-acd4-11630927b5a4

materiales de laboratorio. (s.f.). Obtenido de https://materialeslaboratorio.com/luxometro/

Medidas de a luz. (s.f.). Obtenido de http://www2.fisica.unlp.edu.ar/materias/FEIII/OLD/2012/IntensidadLuminosa.pdf

Meneses, C. P. (2012). *DISEÑO DE UN SISTEMA DE CONTROL DE ACTIVOS PARA EL ALMACÉN DE ELECTRÓNICA DE LA PONTIFICIA UNIVERSIDAD CATÓLICA DEL PERÚ UTILIZANDO RFID.* Lima - Perú.

Mera, H. M. (2013). *MERA, HEINER MAHECHA.*

Mestras, J. P. (s.f.). *PHP Aplicaciones web/sistemas web*. Obtenido de https://www.fdi.ucm.es/profesor/jpavon/web/33-PHP.pdf

Metodologia para el diseño de Dashboard. (2013). En kerzner, *Metodologia para el diseño de Dashboard.*

Microsoft. (2012). *Microsoft Developer Network*. Obtenido de Documentación: https://msdn.microsoft.com/es-es/library/fx6bk1f4(v=vs.100).aspx

Microsoft. (2016). *Developer Network*. Obtenido de https://msdn.microsoft.com/es-es/library/bb545450.aspx

Murphey, R. (Agosto de 2013). *Fundamentos de jQuery.* Obtenido de http://librojquery.com/

Ofimática Profesional - Corinne Hervo. (2013). *Microsoft Excel 2013.* Barcelona: Ediciones ENI.

Oracle. (2016). *NetBeans IDE*. Obtenido de https://netbeans.org/index_es.html

Oracle. (2016). *Oracle Documentación*. Obtenido de http://www.oracle.com/technetwork/es/database/enterprise-edition/documentation/index.html

PBWORK. (2008). *Cursos de Ingeniería de Software.* Obtenido de https://ingsw.pbworks.com/f/Ciclo+de+Vida+del+Software.pdf

Pérez, J. E. (17 de Diciembre de 2010). Obtenido de http://www.jesusda.com/docs/ebooks/introduccion_ajax.pdf

Pérez, J. E. (17 de Diciembre de 2010). *Introducción a XHTML.* Obtenido de http://www.jesusda.com/docs/ebooks/introduccion_xhtml.pdf

Ralph M. Stair, George W. Reynolds. (2001). *Principios de Sistema de Información. Enfoque administrativo.* Madrid: Ediciones Paraninfo.

Rancel, M. R. (2013). *Aprende a programar*. Obtenido de ww.aprenderaprogramar.com

ROMO, P. O. (2016).

Salazar Caraballo, L. (2013). *Asuntos de la Ingeniería de software.* Obtenido de http://www.gazafatonarioit.com

Sanchez, J. A. (2008). Sistema de control de acceso con RFID. México D.F., México.

Sanchez, J. (s.f.). *MySQL guía rápida*. Obtenido de http://www.cartagena99.com/recursos/programacion/apuntes/mysql.pdf

SAS, DOINTECH. (2015). *DOINTECH SAS, Automatización, seguridad y control*. Obtenido de http://www.dointech.com.co/control-de-acceso.html

Schwaber, K., & Sutherland, J. (Julio de 2013). *Scrum guides.* Obtenido de La guía de Scrum: http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf

SIGMA ELECTRÓNICA LTDA. (2016). *SIGMA ELECTRÓNICA LTDA*. Obtenido de http://www.sigmaelectronica.net/sl500l-p-1971.html

SOFTENG. (2016). *SOFTENG*. Obtenido de Metodología Scrum para desarrollo de software - aplicaciones complejas: https://www.softeng.es/es-es/empresa/metodologias-de-trabajo/metodologia-scrum.html

Sparx Systems. (2016). *Sparx Systems*. Obtenido de Enterprise Architect - Herramienta de diseño UML: http://www.sparxsystems.com.ar/products/ea.html

Trigas Gallego, M. (s.f.). *Gestión de proyectos Informáticos.* Obtenido de Metodología SCRUM: http://www.scrumguides.org/docs/scrumguide/v1/Scrum-Guide-ES.pdf

Universidad de Alicante. (2015). *Servicio de Informática ASP.NET MVC 3 Framework*. Obtenido de Modelo vista controlador (MVC): https://si.ua.es/es/documentacion/asp-net-mvc-3/1-dia/modelo-vista-controlador-mvc.html

zona, M. (septiembre de 2014). *Manual de arduino programacion y conc*

eptos basicos.

Anexos

- **Anexo 01:** ficha de observación. de sprint: historia usuario, tarea, prueba

Figura 63:Anexo 01 Formatos de Usuarios y tareas

FICHA HISTORIA USUARIO:

Historia de Usuario	
ID	HUXX
NOMBRE	
PRIORIDAD	
RIESGO	
DESCRIPCION	
VALIDACION	

FICHA TAREAS

TAREA	
ID	TXX
Historia del usuario	
ESTADO	
DESCRIPCION	

FICHA PRUEBAS

Nombre de caso de prueba:	Iniciar herramienta Dashboard
Entradas:	
Condiciones:	

Fuente 89:(Autoría propia)

- **Anexo 02: FOTOS**

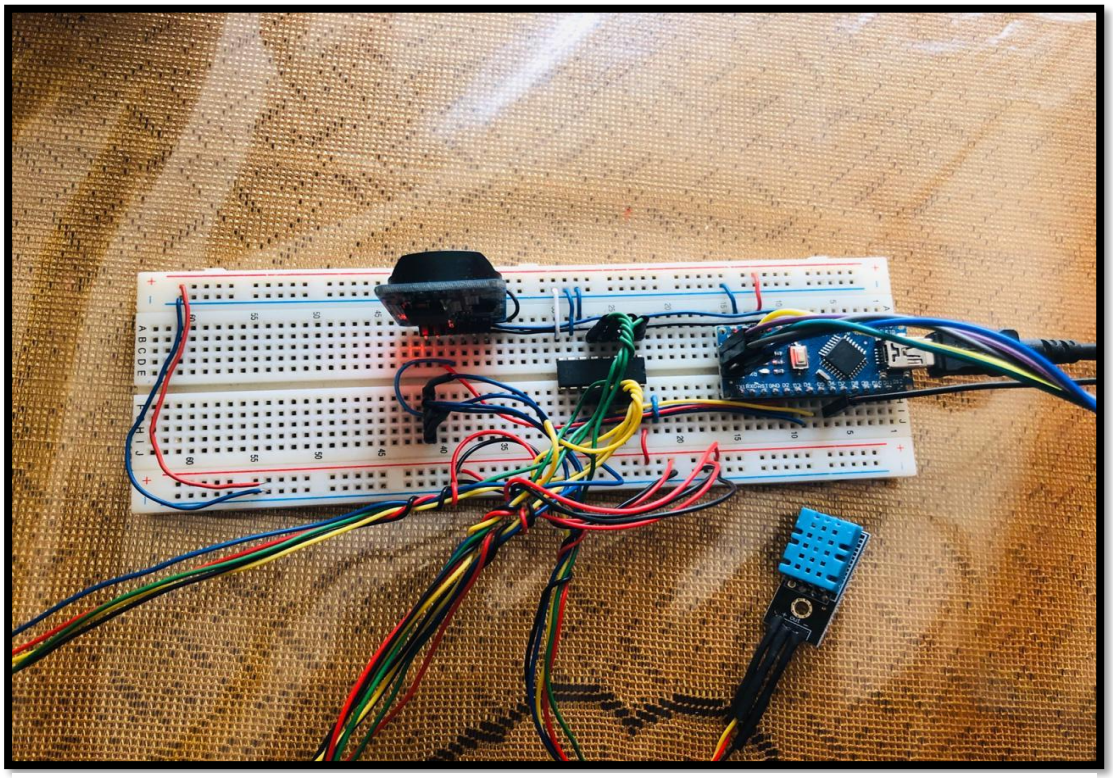
Figura 64:Anexo 02 Fotos sensores

Fuente 90:(Autoría propia)

Figura 65:Anexo 03 Fotos sensores

Fuente 91:(Autoría propia)

Figura 66:: Foto del circuito



Fuente 92:(Autoría propia)

Figura 67:: Foto de la maqueta

Fuente 93:(Autoría propia)

Anexo 03: CD4051BC • CD4052BC • CD4053BC Single 8-Channel Analog Multiplexer/Demultiplexer • Dual 4-Channel Analog Multiplexer/Demultiplexer • Triple 2-Channel Analog Multiplexer/Demultiplexer

FAIRCHILD SEMICONDUCTOR™

November 1983
Revised April 2002

CD4051BC • CD4052BC • CD4053BC
Single 8-Channel Analog Multiplexer/Demultiplexer • Dual 4-Channel Analog Multiplexer/Demultiplexer • Triple 2-Channel Analog Multiplexer/Demultiplexer

General Description

The CD4051BC, CD4052BC, and CD4053BC analog multiplexers/demultiplexers are digitally controlled analog switches having low "ON" impedance and very low "OFF" leakage currents. Control of analog signals up to $15V_{p-p}$ can be achieved by digital signal amplitudes of 3–15V. For example, if V_{DD} = 5V, V_{SS} = 0V and V_{EE} = –5V, analog signals from –5V to +5V can be controlled by digital inputs of 0–5V. The multiplexer circuits dissipate extremely low quiescent power over the full V_{DD}–V_{SS} and V_{DD}–V_{EE} supply voltage ranges, independent of the logic state of the control signals. When a logical "1" is present at the inhibit input terminal all channels are "OFF".

CD4051BC is a single 8-channel multiplexer having three binary control inputs, A, B, and C, and an inhibit input. The three binary signals select 1 of 8 channels to be turned "ON" and connect the input to the output.

CD4052BC is a differential 4-channel multiplexer having two binary control inputs, A and B, and an inhibit input. The two binary input signals select 1 or 4 pairs of channels to be turned on and connect the differential analog inputs to the differential outputs.

CD4053BC is a triple 2-channel multiplexer having three separate digital control inputs, A, B, and C, and an inhibit input. Each control input selects one of a pair of channels which are connected in a single-pole double-throw configuration.

Features

- Wide range of digital and analog signal levels: digital 3 – 15V, analog to $15V_{p-p}$
- Low "ON" resistance: 80Ω (typ.) over entire $15V_{p-p}$ signal-input range for $V_{DD} - V_{EE}$ = 15V
- High "OFF" resistance: channel leakage of ±10 pA (typ.) at $V_{DD} - V_{EE}$ = 10V
- Logic level conversion for digital addressing signals of 3 – 15V ($V_{DD} - V_{SS}$ = 3 – 15V) to switch analog signals to 15 V_{p-p} ($V_{DD} - V_{EE}$ = 15V)
- Matched switch characteristics: ΔR_{ON} = 5Ω (typ.) for $V_{DD} - V_{EE}$ = 15V
- Very low quiescent power dissipation under all digital-control input and supply conditions: 1 μ W (typ.) at $V_{DD} - V_{SS} = V_{DD} - V_{EE}$ = 10V
- Binary address decoding on chip

Ordering Code:

Order Number	Package Number	Package Description
CD4051BCM	M16A	16-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-012, 0.150" Narrow
CD4051BCSJ	M16D	16-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
CD4051BCMTC	MTC16	16-Lead Thin Shrink Small Outline Package (TSSOP), JEDEC MO-153, 4.4mm Wide
CD4051BCN	N16E	16-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide
CD4052BCM	M16A	16-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-012, 0.150" Narrow
CD4052BCSJ	M16D	16-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
CD4052BCN	N16E	16-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide
CD4053BCM	M16A	16-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-012, 0.150" Narrow
CD4053BCSJ	M16D	16-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
CD4053BCN	N16E	16-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide

Devices also available in Tape and Reel. Specify by appending the suffix letter "X" to the ordering code.

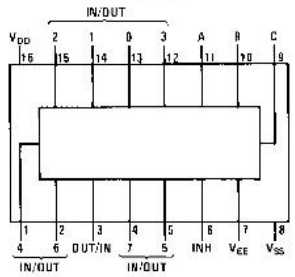
 DS005662 www.fairchildsemi.com

CD4051BC • CD4052BC • CD4053BC Single 8-Channel Analog Multiplexer/Demultiplexer • Dual 4-Channel Analog Multiplexer/Demultiplexer • Triple 2-Channel Analog Multiplexer/Demultiplexer

Connection Diagrams

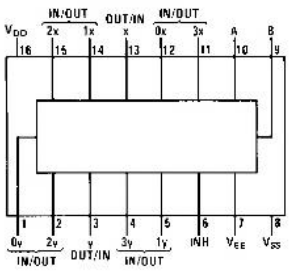
Pin Assignments for DIP and SOIC

CD4051BC



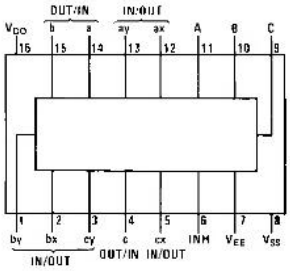
TOP VIEW

CD4052BC



TOP VIEW

CD4053BC



TOP VIEW

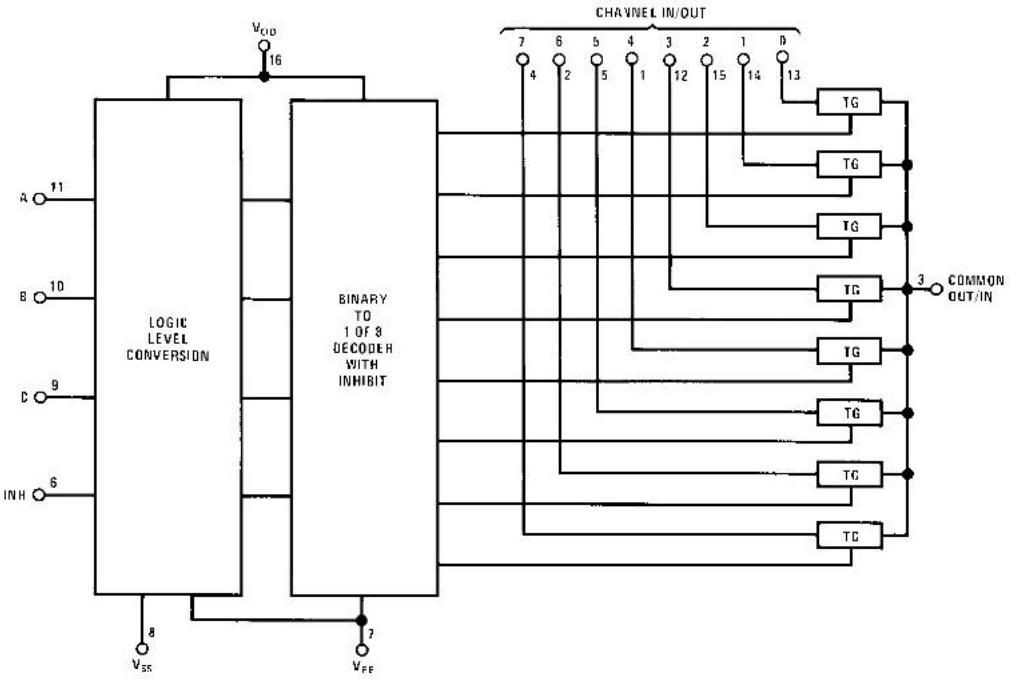
Truth Table

INPUT STATES				"ON" CHANNELS		
INHIBIT	C	B	A	CD4051B	CD4052B	CD4053B
0	0	0	0	0	0X, 0Y	cx, bx, ax
0	0	0	1	1	1X, 1Y	cx, bx, ay
0	0	1	0	2	2X, 2Y	cx, by, ax
0	0	1	1	3	3X, 3Y	cx, by, ay
0	1	0	0	4		cy, bx, ax
0	1	0	1	5		cy, bx, ay
0	1	1	0	6		cy, by, ax
0	1	1	1	7		cy, by, ay
1	*	*	*	NONE	NONE	NONE

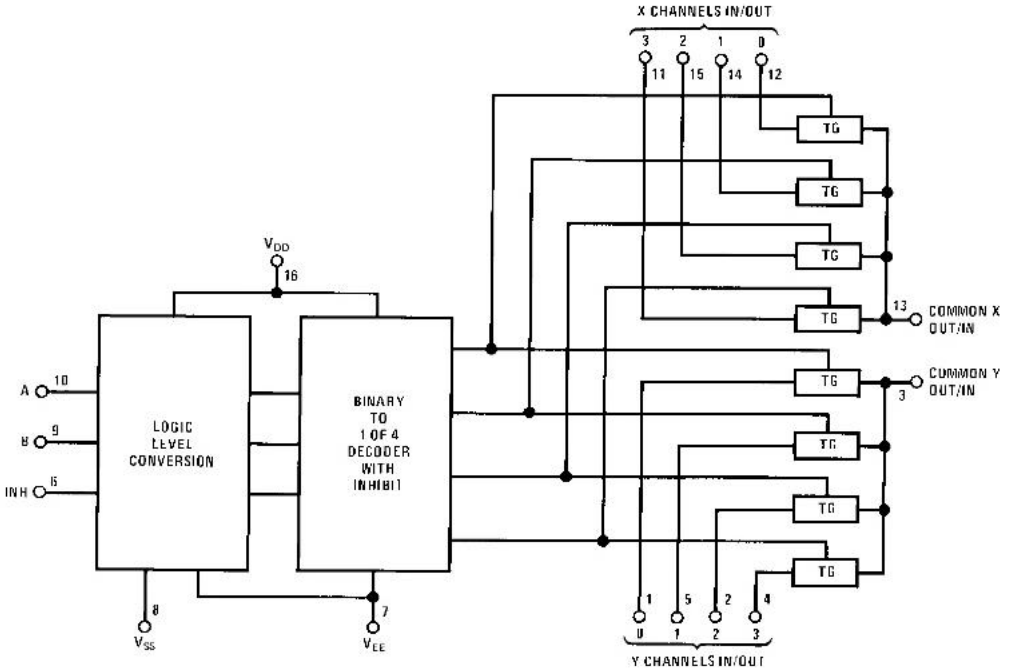
*Don't Care condition.

Logic Diagrams

CD4051BC

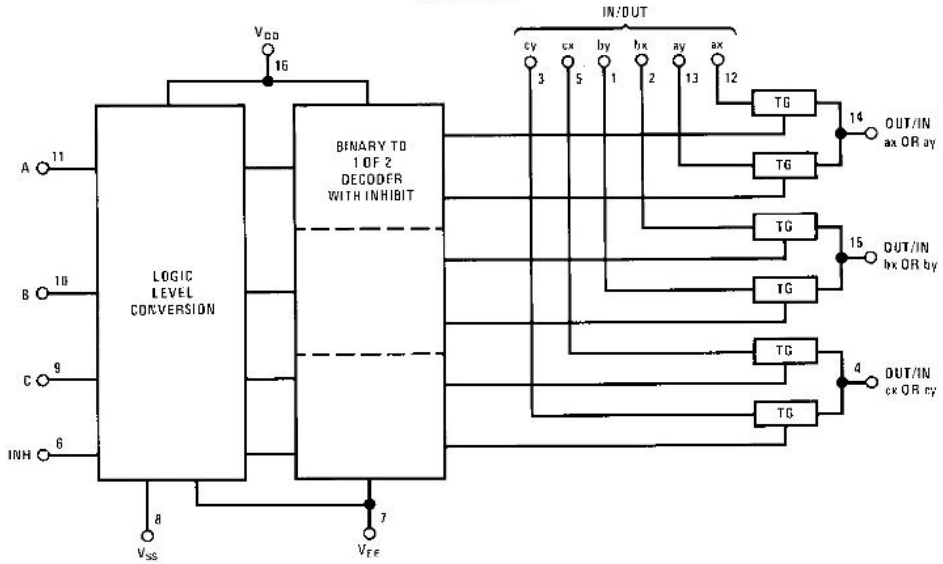


CD4052BC



Logic Diagrams (Continued)

CD4053BC



Absolute Maximum Ratings (Note 1)

DC Supply Voltage (V_{DD})	−0.5 V_{DC} to +18 V_{DC}
Input Voltage (V_{IN})	−0.5 V_{DC} to V_{DD} +0.5 V_{DC}
Storage Temperature Range (T_S)	−65°C to +150°C
Power Dissipation (P_D)	
Dual-In-Line	700 mW
Small Outline	500 mW
Lead Temperature (T_L) (soldering, 10 seconds)	260°C

Recommended Operating Conditions

DC Supply Voltage (V_{DD})	+5 V_{DC} to +15 V_{DC}
Input Voltage (V_{IN})	0V to V_{DD} V_{DC}
Operating Temperature Range (T_A)	
CD4051BC/CD4052BC/CD4053BC	−55°C to +125°C

Note 1: "Absolute Maximum Ratings" are those values beyond which the safety of the device cannot be guaranteed. Except for "Operating Temperature Range" they are not meant to imply that the devices should be operated at these limits. The Electrical Characteristics tables provide conditions for actual device operation.

DC Electrical Characteristics (Note 2)

Symbol	Parameter	Conditions		−55°C		−25°			125°C		Units
				Min	Max	Min	Typ	Max	Min	Max	
Control A, B, C and Inhibit											
I_{IN}	Input Current	V_{DD} = 15V, V_{IN} = 0V	V_{EE} = 0V		−0.1		-10^{-5}	−0.1		−1.0	µA
		V_{DD} = 15V, V_{IN} = 15V	V_{EE} = 0V		0.1		10^{-5}	0.1		1.0	
I_{DD}	Quiescent Device Current	V_{DD} = 5V			5			5		150	µA
		V_{DD} = 10V			10			10		300	
		V_{DD} = 15V			20			20		600	
Signal Inputs (V_{IS}) and Outputs (V_{OS})											
R_{ON}	"ON" Resistance (Peak for $V_{EE} \le V_{IS} \le V_{DD}$)	R_L = 10 kΩ (any channel selected)	V_{DD} = 2.5V, V_{EE} = −2.5V or V_{DD} = 5V, V_{EE} = 0V		800		270	1050		1300	Ω
			V_{DD} = 5V, V_{EE} = −5V or V_{DD} = 10V, V_{EE} = 0V		310		120	400		550	Ω
			V_{DD} = 7.5V, V_{EE} = −7.5V or V_{DD} = 15V, V_{EE} = 0V		200		80	240		320	Ω
ΔR_{ON}	Δ "ON" Resistance Between Any Two Channels	R_L = 10 kΩ (any channel selected)	V_{DD} = 2.5V, V_{EE} = −2.5V or V_{DD} = 5V, V_{EE} = 0V				10				Ω
			V_{DD} = 5V V_{EE} = −5V or V_{DD} = 10V, V_{EE} = 0V				10				Ω
			V_{DD} = 7.5V, V_{EE} = −7.5V or V_{DD} = 15V, V_{EE} = 0V				5				Ω
	"OFF" Channel Leakage Current, any channel "OFF"	V_{DD}=7.5V, O/I=±7.5V, I/O=0V	V_{EE}=−7.5V		±50		±0.01	±50		±500	nA
	"OFF" Channel Leakage Current, all channels "OFF" (Common OUT/IN)	Inhibit = 7.5V V_{DD} = 7.5V, V_{EE} = −7.5V, O/I = 0V	CD4051		±200		±0.08	±200		±2000	nA
			D4052		±200		±0.04	±200		±2000	
		I/O = ±7.5V	CD4053		±200		±0.02	±200		±2000	

DC Electrical Characteristics (Continued)

Symbol	Parameter	Conditions	−55°C		+25°			125°C		Units
			Min	Max	Min	Typ	Max	Min	Max	
Control Inputs A, B, C and Inhibit										
V_{IL}	LOW Level Input Voltage	$V_{EE} = V_{SS}$ R_L = 1 kΩ to V_{SS}								
		I_{IS}<2 µA on all OFF Channels								
		$V_{IS} = V_{DD}$ thru 1 kΩ								
		V_{DD} = 5V		1.5			1.5		1.5	
		V_{DD} = 10V		3.0			3.0		3.0	V
		V_{DD} = 15V		4.0			4.0		4.0	
V_{IH}	HIGH Level Input Voltage	V_{DD} = 5	3.5		3.5			3.5		
		V_{DD} = 10	7		7			7		V
		V_{DD} = 15	11		11			11		

Note 2: All voltages measured with respect to V_{SS} unless otherwise specified.

AC Electrical Characteristics (Note 3)

$T_A = 25°C$, $t_r = t_f = 20$ ns, unless otherwise specified.

Symbol	Parameter	Conditions	V_{DD}	Min	Typ	Max	Units
t_{PZH}, t_{PZL}	Propagation Delay Time from Inhibit to Signal Output (channel turning on)	$V_{EE} = V_{SS} = 0V$, $R_L = 1\ k\Omega$, $C_L = 50$ pF	5V		600	1200	ns
			10V		225	450	
			15V		160	320	
t_{PHZ}, t_{PLZ}	Propagation Delay Time from Inhibit to Signal Output (channel turning off)	$V_{EE} = V_{SS} = 0V$, $R_L = 1\ k\Omega$, $C_L = 50$ pF	5V		210	420	ns
			10V		100	200	
			15V		75	150	
C_{IN}	Input Capacitance						
	Control input				5	7.5	pF
	Signal Input (IN/OUT)				10	15	
C_{OUT}	Output Capacitance (common OUT/IN)						
	CD4051	$V_{EE} = V_{SS} = 0V$	10V		30		pF
	CD4052		10V		15		
	CD4053		10V		8		
C_{IOS}	Feedthrough Capacitance				0.2		pF
C_{PD}	Power Dissipation Capacitance						
	CD4051				110		pF
	CD4052				140		
	CD4053				70		
Signal Inputs (V_{IS}) and Outputs (V_{OS})							
	Sine Wave Response (Distortion)	$R_L = 10\ k\Omega$, $f_{IS} = 1$ kHz, $V_{IS} = 5\ V_{p-p}$, $V_{EE} = V_{SI} = 0V$	10V		0.04		%
	Frequency Response, Channel "ON" (Sine Wave Input)	$R_L = 1\ k\Omega$, $V_{EE} = 0V$, $V_{IS} = 5V_{p-p}$, $20 \log_{10} V_{OS}/V_{IS} = -3$ dB	10V		40		MHz
	Feedthrough, Channel "OFF"	$R_L = 1\ k\Omega$, $V_{EE} = V_{SS} = 0V$, $V_{IS} = 5V_{p-p}$, $20 \log_{10} V_{OS}/V_{IS} = -40$ dB	10V		10		MHz
	Crosstalk Between Any Two Channels (frequency at 40 dB)	$R_L = 1\ k\Omega$, $V_{EE} = V_{SS} = 0V$, $V_{IS}(A) = 5V_{p-p}$, $20 \log_{10} V_{OS}(B)/V_{IS}(A) = -40$ dB (Note 4)	10V		3		MHz
t_{PHL}, t_{PLH}	Propagation Delay Signal Input to Signal Output	$V_{EE} = V_{SS} = 0V$, $C_L = 50$ pF	5V		25	55	ns
			10V		15	35	
			15V		10	25	
Control Inputs, A, B, C and Inhibit							
	Control Input to Signal Crosstalk	$V_{EE} = V_{SS} = 0V$, $R_L = 10\ k\Omega$ at both ends of channel. Input Square Wave Amplitude = 10V	10V		65		mV (peak)
t_{PHL}, t_{PLH}	Propagation Delay Time from Address to Signal Output (channels "ON" or "OFF")	$V_{EE} = V_{SS} = 0V$, $C_L = 50$ pF	5V		500	1000	ns
			10V		180	360	
			15V		120	240	

Note 3: AC Parameters are guaranteed by DC correlated testing.

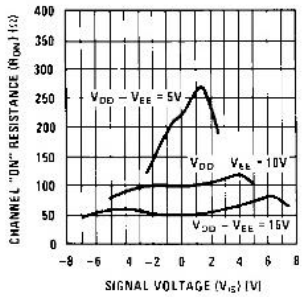
Note 4: A, B are two arbitrary channels with A turned "ON" and B "OFF".

Special Considerations

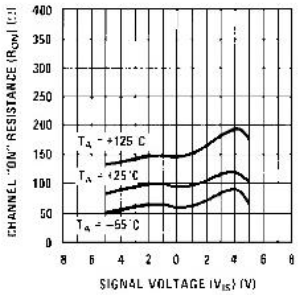
In certain applications the external load-resistor current may include both V_{DD} and signal-line components. To avoid drawing V_{DD} current when switch current flows into IN/OUT pin, the voltage drop across the bidirectional switch must not exceed 0.6V at $T_A \leq 25°C$, or 0.4V at $T_A > 25°C$ (calculated from R_{ON} values shown). No V_{DD} current will flow through R_L if the switch current flows into OUT/IN pin.

Typical Performance Characteristics

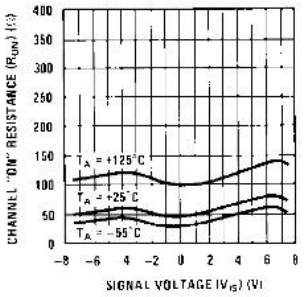
"ON" Resistance vs Signal Voltage for $T_A = 25°C$



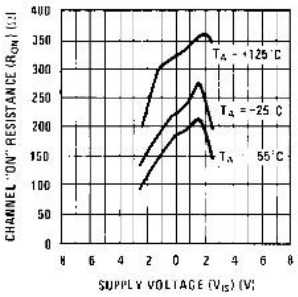
"ON" Resistance as a Function of Temperature for $V_{DD} - V_{EE} = 10V$



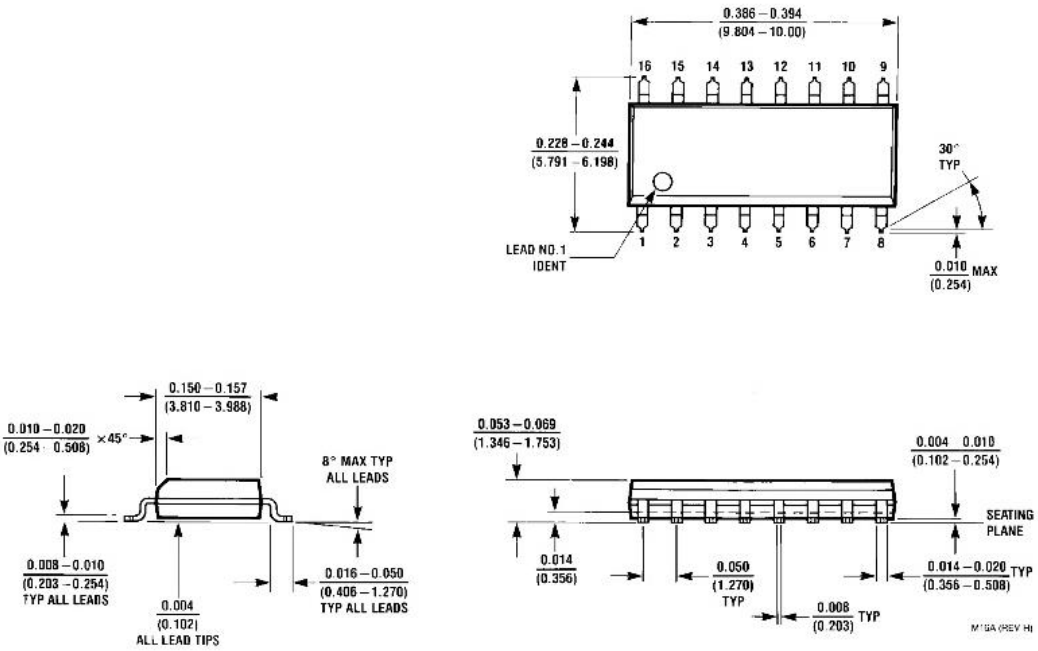
"ON" Resistance as a Function of Temperature for $V_{DD} - V_{EE} = 15V$



"ON" Resistance as a Function of Temperature for $V_{DD} - V_{EE} = 5V$

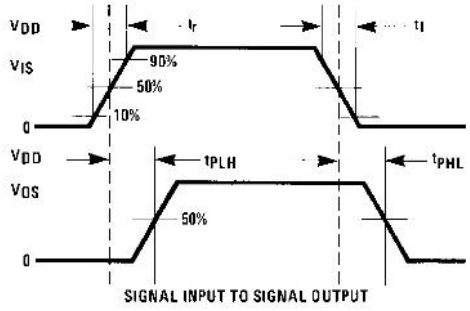


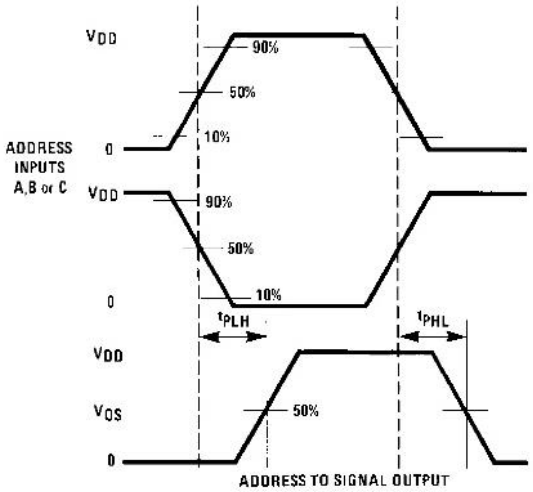
Physical Dimensions inches (millimeters) unless otherwise noted

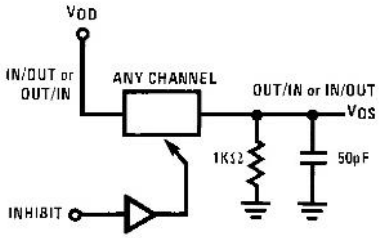


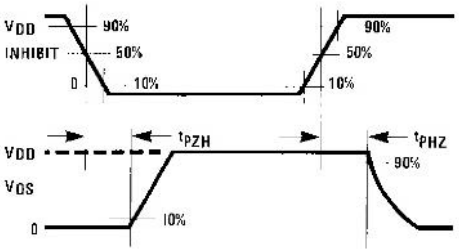
16-Lead Small Outline Integrated Circuit (SOIC), JEDEC MS-012, 0.150" Narrow
Package Number M16A

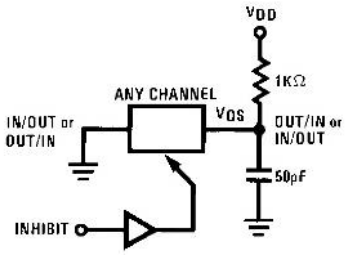
Switching Time Waveforms

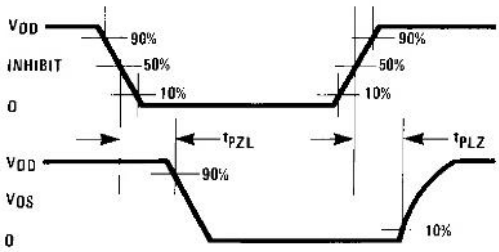




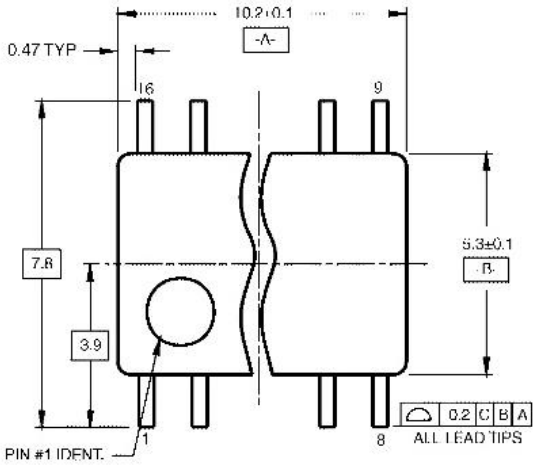


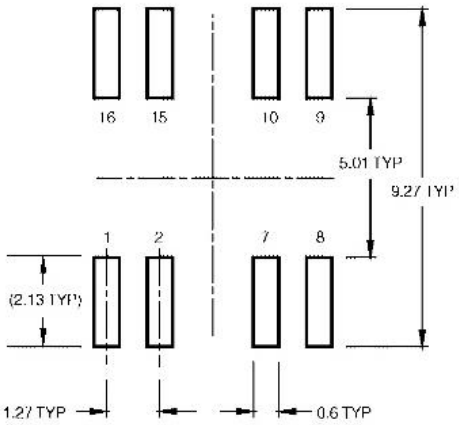




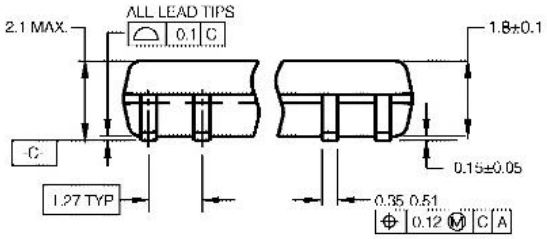


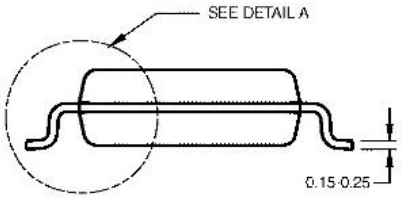
Physical Dimensions inches (millimeters) unless otherwise noted (Continued)





LAND PATTERN RECOMMENDATION





DIMENSIONS ARE IN MILLIMETERS

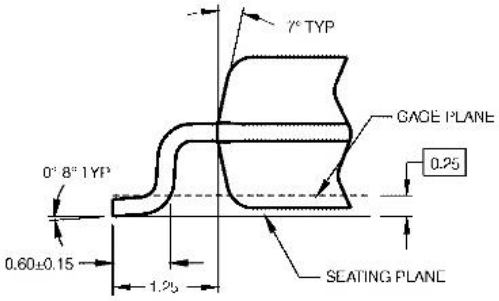
NOTES:

A. CONFORMS TO EIAJ EDR 7320 REGISTRATION, ESTABLISHED IN DECEMBER, 1998.

B. DIMENSIONS ARE IN MILLIMETERS.

C. DIMENSIONS ARE EXCLUSIVE OF BURRS, MOLD FLASH, AND TIE BAR EXTRUSIONS.

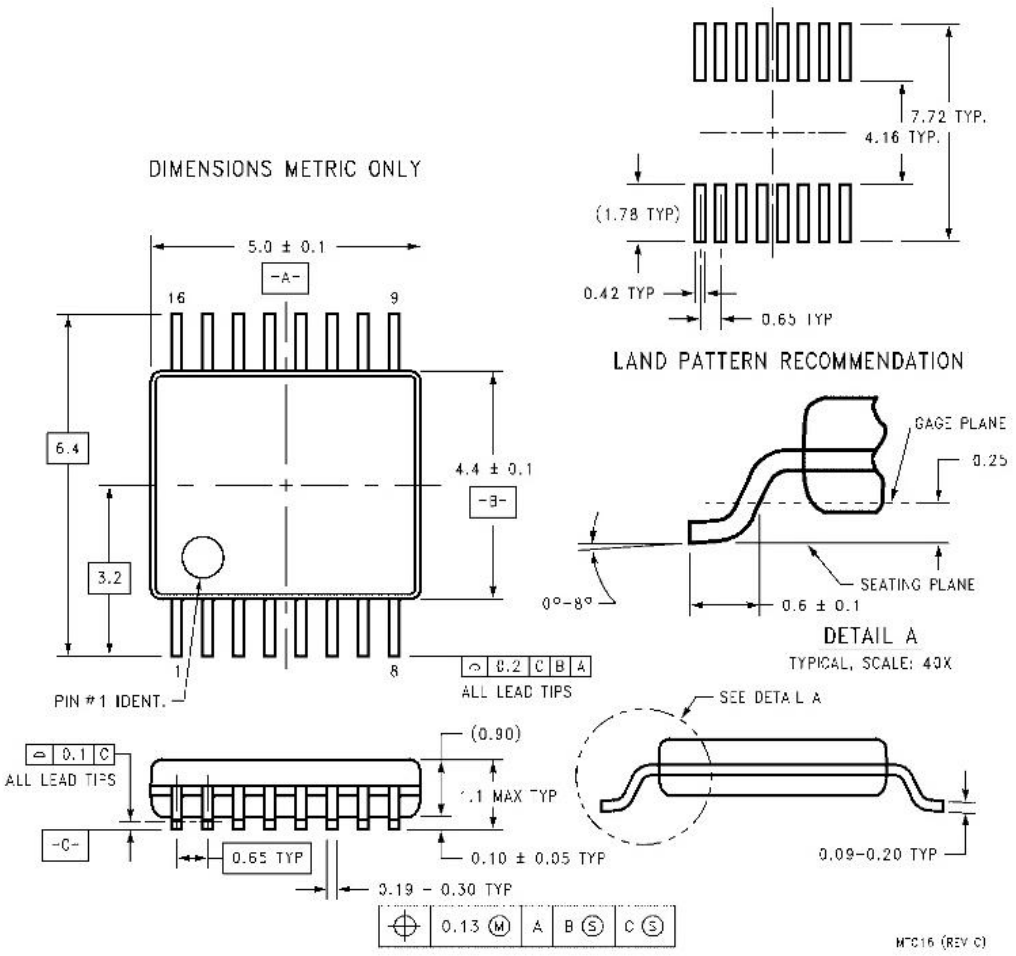
M16DRevB1



DETAIL A

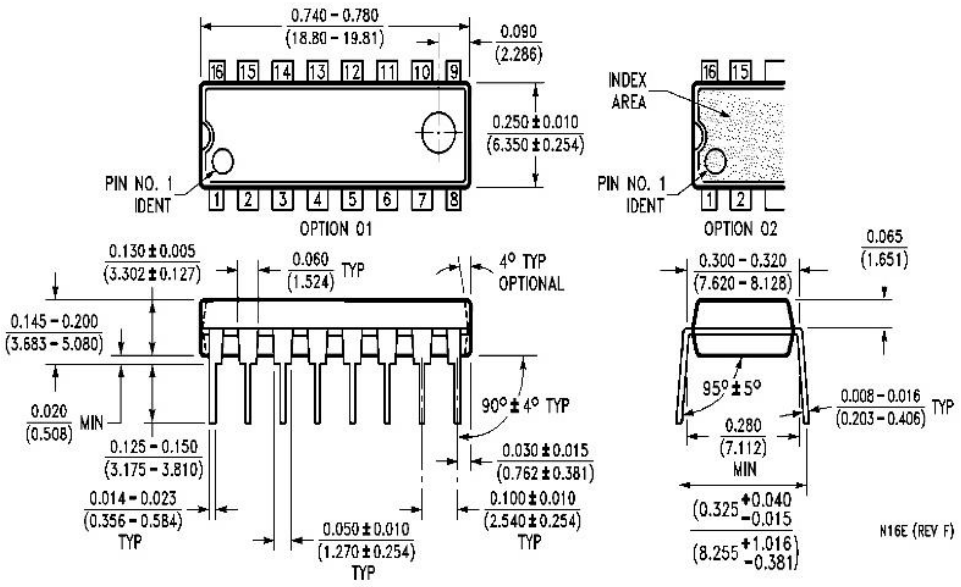
16-Lead Small Outline Package (SOP), EIAJ TYPE II, 5.3mm Wide
Package Number M16D

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



16-Lead Thin Shrink Small Outline Package (TSSOP), JEDEC MO-153, 4.4mm Wide
Package Number MTC16

Physical Dimensions inches (millimeters) unless otherwise noted (Continued)



16-Lead Plastic Dual-In-Line Package (PDIP), JEDEC MS-001, 0.300" Wide
Package Number N16E

➢ ANEXO 04: Digital 16 bit Serial OutPut Type Ambient Sensor IC

ROHM SEMICONDUCTOR

Technical Note

Ambient Light Sensor IC Series

Digital 16bit Serial Output Type Ambient Light Sensor IC

Pb Free ROHM Electronic Components

RoHS Directive Compliance

BH1750FVI No.11046EDT01

●Descriptions

BH1750FVI is an digital Ambient Light Sensor IC for I^2C bus interface. This IC is the most suitable to obtain the ambient light data for adjusting LCD and Keypad backlight power of Mobile phone. It is possible to detect wide range at High resolution. (1 - 65535 lx).

●Features

1) I^2C bus Interface (f / s Mode Support)
2) Spectral responsibility is approximately human eye response
3) Illuminance to Digital Converter
4) Wide range and High resolution. (1 - 65535 lx)
5) Low Current by power down function
6) 50Hz / 60Hz Light noise reject-function
7) 1.8V Logic input interface
8) No need any external parts
9) Light source dependency is little. (ex. Incandescent Lamp. Fluorescent Lamp. Halogen Lamp. White LED. Sun Light)
10) It is possible to select 2 type of I^2C slave-address.
11) Adjustable measurement result for influence of optical window
(It is possible to detect min. 0.11 lx, max. 100000 lx by using this function.)
12) Small measurement variation (+/- 20%)
13) The influence of infrared is very small.

●Applications

Mobile phone, LCD TV, NOTE PC, Portable game machine, Digital camera, Digital video camera, PDA, LCD display

●Absolute Maximum Ratings

Parameter	Symbol	Ratings	Units
Supply Voltage	Vmax	4.5	V
Operating Temperature	Topr	-40～85	℃
Storage Temperature	Tstg	-40～100	℃
SDA Sink Current	Imax	7	mA
Power Dissipation	Pd	260※	mW

※ 70mm × 70mm × 1.6mm glass epoxy board. Derating in done at 3.47mW/℃ for operating above Ta=25℃.

●Operating Conditions

Parameter	Symbol	Ratings			Units
		Min.	Typ.	Max.	
Vcc Voltage	Vcc	2.4	3.0	3.6	V
I^2C Reference Voltage	VDVI	1.65	-	Vcc	V

●**Electrical Characteristics (Vcc = 3.0V, DVI = 3.0V, Ta = 25℃, unless otherwise noted)**

Parameter	Symbol	Limits			Units	Conditions
		Min.	Typ.	Max.		
Supply Current	Icc1	–	120	190	μA	Ev = 100 lx ※1
Powerdown Current	Icc2	–	0.01	1.0	μA	No input Light
Peak Wave Length	λp	–	560	–	nm	
Measurement Accuracy	S/A	0.96	1.2	1.44	times	Sensor out / Actual lx EV = 1000 lx ※1, ※2
Dark (0 lx) Sensor out	S0	0	0	3	count	H-Resolution Mode ※3
H-Resolution Mode Resolution	rHR	–	1	–	lx	
L-Resolution Mode Resolution	rLR	–	4	–	lx	
H-Resolution Mode Measurement Time	tHR	–	120	180	ms	
L-Resolution Mode Measurement Time	tLR	–	16	24	ms	
Incandescent / Fluorescent Sensor out ratio	rIF	–	1	–	times	EV = 1000 lx
ADDR Input 'H' Voltage	VAH	0.7 * VCC	–	–	V	
ADDR Input 'L' Voltage	VAL	–	–	0.3 * VCC	V	
DVI Input 'L' Voltage	VDVL	–	–	0.4	V	
SCL, SDA Input 'H' Voltage 1	VIH1	0.7 * DVI	–	–	V	DVI ≧ 1.8V
SCL, SDA Input 'H' Voltage 2	VIH2	1.26	–	–	V	1.65V ≦ DVI <1.8V
SCL, SDA Input 'L' Voltage 1	VIL1	–	–	0.3 * DVI	V	DVI ≧ 1.8V
SCL, SDA Input 'L' Voltage 2	VIL2	–	–	DVI – 1.26	V	1.65V ≦ DVI < 1.8V
SCL, SDA, ADDR Input 'H' Current	IIH	–	–	10	μA	
SCL, SDA, ADDR Input 'L' Current	IIL	–	–	10	μA	
I²C SCL Clock Frequency	fSCL	–	–	400	kHz	
I²C Bus Free Time	tBUF	1.3	–	–	μs	
I²C Hold Time (repeated) START Condition	tHDSTA	0.6	–	–	μs	
I²C Set up time for a Repeated START Condition	tSUSTA	0.6	–	–	μs	
I²C Set up time for a Repeated STOP Condition	tSUSTD	0.6	–	–	μs	
I²C Data Hold Time	tHDDAT	0	–	0.9	μs	
I²C Data Setup Time	tSUDAT	100	–	–	ns	
I²C 'L' Period of the SCL Clock	tLOW	1.3	–	–	μs	
I²C 'H' Period of the SCL Clock	tHIGH	0.6	–	–	μs	
I²C SDA Output 'L' Voltage	VOL	0	–	0.4	V	IOL = 3 mA

※1 White LED is used as optical source.
※2 Measurement Accuracy typical value is possible to change '1' by "Measurement result adjustment function".
※3 Use H-resolution mode or H-resolution mode2 if dark data (less than 10 lx) is need.

●**Reference Data**

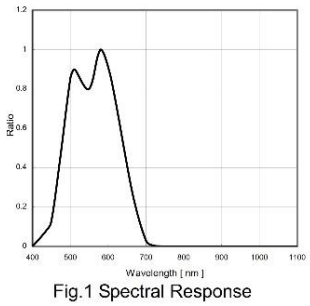


Fig.1 Spectral Response

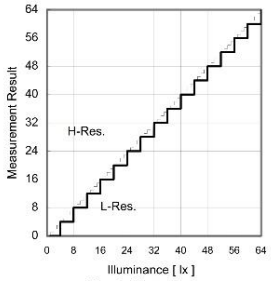


Fig.2 Illuminance - Measurement Result 1

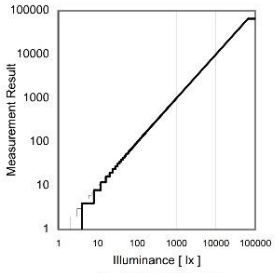


Fig.3 Illuminance - Measuremnet Result 2

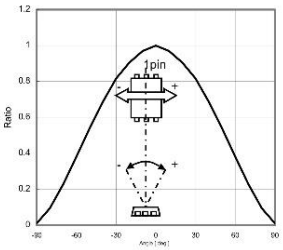


Fig.4 Directional Characteristics 1

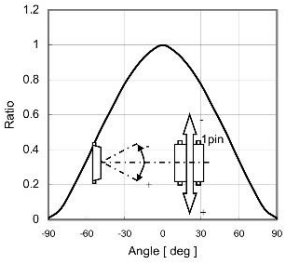


Fig.5 Directional Characteristics 2

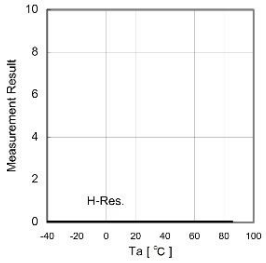


Fig.6 Dark Response

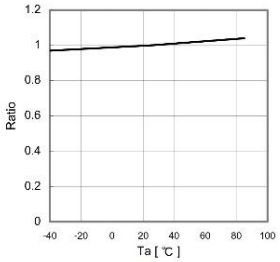


Fig.7 Measurement Accuracy Temperature Dependency

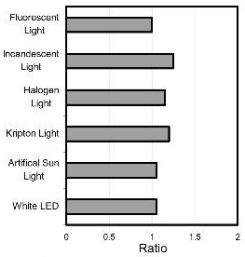


Fig.8 Light Source Dependency (Fluorescent Light is set to '1')

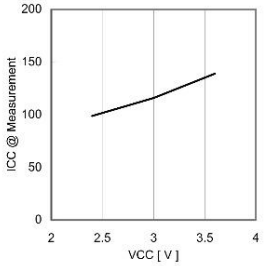


Fig.9 VCC - ICC (During measurement)

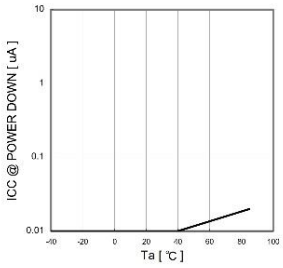


Fig.10 VCC – ICC@0 Lx (POWER DOWN)

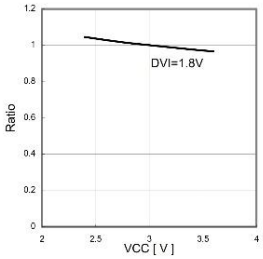


Fig.11 Measurement Result VCC Dependency

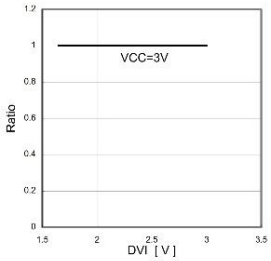
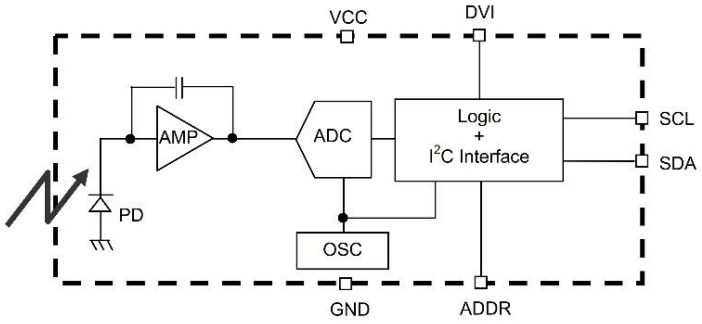


Fig.12 Measurement Result DVI Dependency

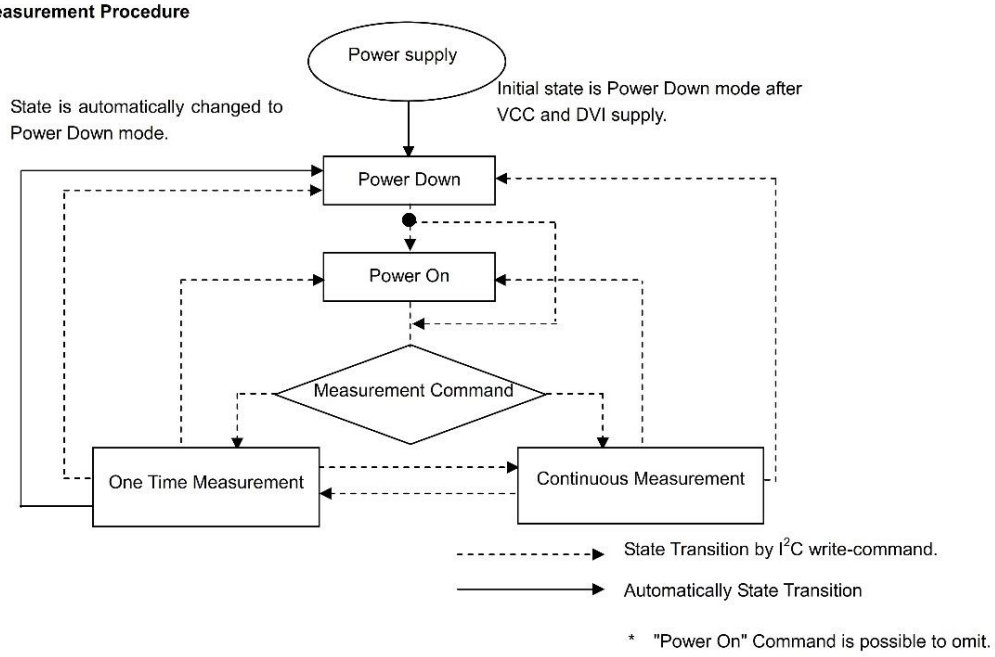
●Block Diagram



●Block Diagram Descriptions

- PD
  Photo diode with approximately human eye response.
- AMP
  Integration-OPAMP for converting from PD current to Voltage.
- ADC
  AD converter for obtainment Digital 16bit data.
- Logic + I^2C Interface
  Ambient Light Calculation and I^2C BUS Interface. It is including below register.
  Data Register → This is for registration of Ambient Light Data. Initial Value is "0000_0000_0000_0000".
  Measurement Time Register → This is for registration of measurement time. Initial Value is "0100_0101".
- OSC
  Internal Oscillator (typ. 320kHz). It is CLK for internal logic.

●Measurement Procedure



* "Power On" Command is possible to omit.

●**Instruction Set Architecture**

Instruction	Opecode	Comments
Power Down	0000_0000	No active state.
Power On	0000_0001	Waiting for measurement command.
Reset	0000_0111	Reset Data register value. Reset command is not acceptable in Power Down mode.
Continuously H-Resolution Mode	0001_0000	Start measurement at 1lx resolution. Measurement Time is typically 120ms.
Continuously H-Resolution Mode2	0001_0001	Start measurement at 0.5lx resolution. Measurement Time is typically 120ms.
Continuously L-Resolution Mode	0001_0011	Start measurement at 4lx resolution. Measurement Time is typically 16ms.
One Time H-Resolution Mode	0010_0000	Start measurement at 1lx resolution. Measurement Time is typically 120ms. It is automatically set to Power Down mode after measurement.
One Time H-Resolution Mode2	0010_0001	Start measurement at 0.5lx resolution. Measurement Time is typically 120ms. It is automatically set to Power Down mode after measurement.
One Time L-Resolution Mode	0010_0011	Start measurement at 4lx resolution. Measurement Time is typically 16ms. It is automatically set to Power Down mode after measurement.
Change Measurement time (High bit)	01000_MT[7,6,5]	Change measurement time. ※ Please refer "adjust measurement result for influence of optical window."
Change Masurement time (Low bit)	011_MT[4,3,2,1,0]	Change measurement time. ※ Please refer "adjust measurement result for influence of optical window."

※ Don't input the other opecode.

●Measurement mode explanation

Measurement Mode	Measurement Time.	Resolurtion
H-resolution Mode2	Typ. 120ms.	0.5 lx
H-Resolution Mode	Typ. 120ms.	1 lx.
L-Resolution Mode	Typ. 16ms.	4 lx.

We recommend to use H-Resolution Mode.
Measurement time (integration time) of H-Resolution Mode is so long that some kind of noise(including in 50Hz / 60Hz noise) is rejected. And H-Resolution Mode is 1 l x resolution so that it is suitable for darkness (less than 10 lx)
H-resolution mode2 is also suitable to detect for darkness.

●**Explanation of Asynchronous reset and Reset command "0000_0111"**

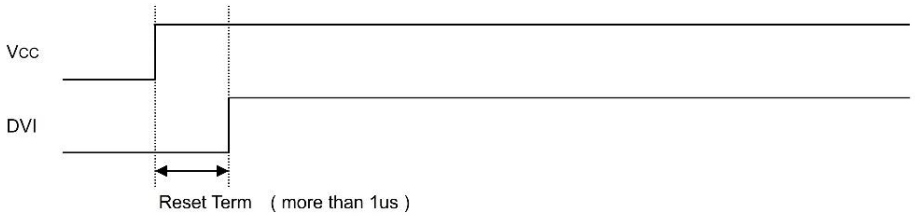
1) Asynchronous reset
All registers are reset. It is necessary on power supply sequence. Please refer "Timing chart for VCC and DVI power supply sequence" in this page. It is power down mode during DVI = 'L'.

2) Reset command
Reset command is for only reset Illuminance data register. (reset value is '0') It is not necessary even power supply sequence.It is used for removing previous measurement result. This command is not working in power down mode, so that please set the power on mode before input this command.

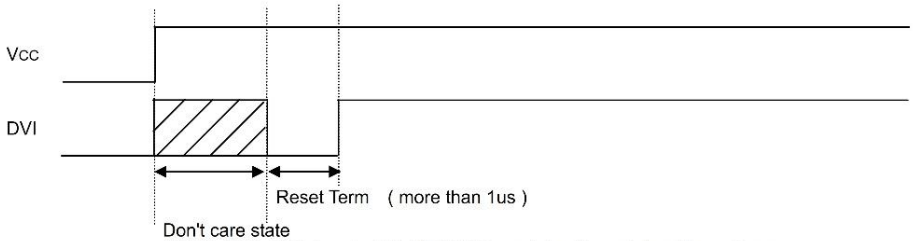
●Timing chart for VCC and DVI power supply sequence

DVI is I^2C bus reference voltage terminal. And it is also asynchronous reset terminal. It is necessary to set to 'L' after Vcc is supplied. In DVI 'L' term, internal state is set to Power Down mode.

1) Recommended Timing chart1 for VCC and DVI supply.



2) Timing chart2 for VCC and DVI supply.
(If DVI rises within 1μs after VCC supply)



Don't care state
ADDR, SDA, SCL is not stable if DVI 'L' term (1us) is not given by systems.
In this case, please connect the resisters (approximately 100kOhm) to ADDR without directly connecting to VCC or GND,
because it is 3 state buffer for Internal testing.

●Measurement sequence example from "Write instruction" to "Read measurement result"

ex1) Continuously H-resolution mode (ADDR = 'L')

from Master to Slave

from Slave to Master

① Send "Continuously H-resolution mode " instruction

ST	0100011	0	Ack	00010000	Ack	SP

② Wait to complete 1st H-resolution mode measurement.(max. 180ms.)

③ Read measurement result.

ST	0100011	1	Ack	High Byte [15:8]	Ack

Low Byte [7:0]	$\overline{\text{Ack}}$	SP

How to calculate when the data High Byte is "10000011" and Low Byte is "10010000"
$(2^{15} + 2^{9} + 2^{8} + 2^{7} + 2^{4}) / 1.2 \fallingdotseq 28067$ [lx]

The result of continuously measurement mode is updated.(120ms.typ at H-resolution mode, 16ms.typ at L-resolution mode)

ex2) One time L-resolution mode (ADDR = 'H')

① Send "One time L-resolution mode " instruction

ST	1011100	0	Ack	00100011	Ack	SP

② Wait to complete L-resolution mode measurement.(max. 24ms.)

③ Read measurement result

ST	1011100	1	Ack	High Byte [15:8]	Ack

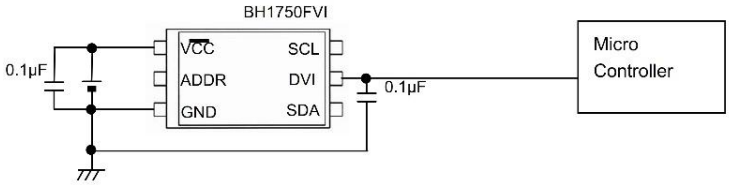
Low Byte [7:0]	$\overline{\text{Ack}}$	SP

How to calculate when the data High Byte is "00000001" and Low Byte is "00010000"
$(2^{8} + 2^{4}) / 1.2 \fallingdotseq 227$ [lx]
In one time measurement, Statement moves to power down mode after measurement completion.If updated result is need then please resend measurement instruction.

●**Application circuit example of DVI terminal**

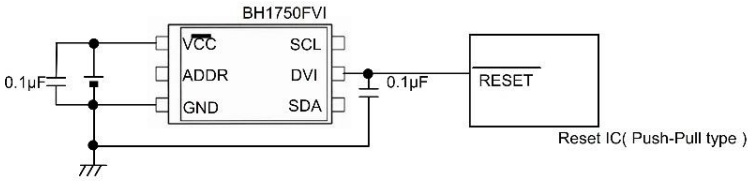
The DVI terminal is an asynchronous reset terminal. Please note that there is a possibility that IC doesn't operate normally if the reset section is not installed after the start-up of Vcc.
(Please refer to the paragraph of "Timing chart for Vcc and DVI power supply sequence")
The description concerning SDA and the terminal SCL is omitted in this application circuit example. Please design the application the standard of the I2C bus as it finishes being satisfactory. Moreover, the description concerning the terminal ADDR is omitted. Please refer to the paragraph of "Timing chart for Vcc and DVI power supply sequence" about the terminal ADDR design.

ex 1) The control signal line such as CPU is connected.

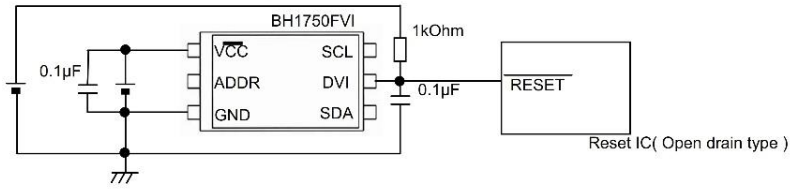


ex 2) Reset IC is used.

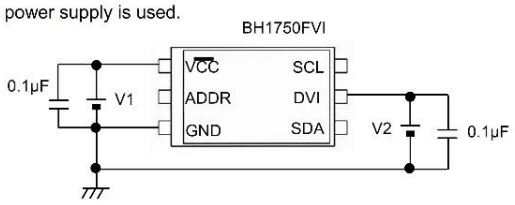
1, For Reset IC of the Push-Pull type



2, For Reset IC of the Open drain output

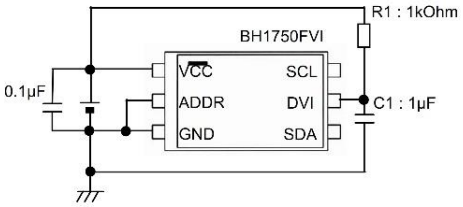


ex 3) A different power supply is used.



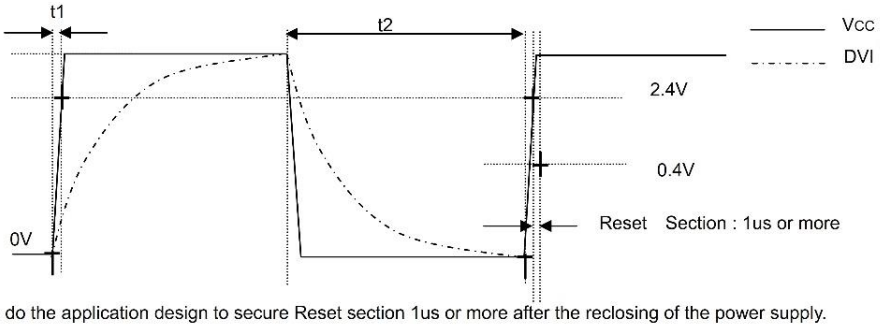
※ Power supply of DVI must stand up later than power supply of VCC stand up, because it is necessary to secure reset section (1μs or more).

ex 4) LPF using CR is inserted between VCC and DVI.
This method has the possibility that the Reset section of turning on the power supply can not satisfied. cannot be satisfied.
Please design the set considering the characteristic of the power supply enough.



◆ Notes when CR is inserted between VCC and DVI

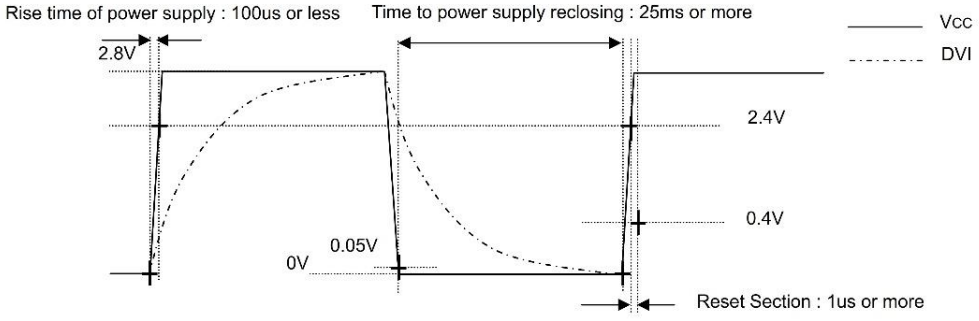
※ Please note that there is a possibility that reset section (1μs) can not be satisfied because the power supply is turned on when the rise time of VCC is slow
※ When VCC is turned off, the DVI voltage becomes higher than VCC voltage but IC destruction is not occred if recommended constant (R1 = 1kOhm, C1 = 1μF) is used.
※ Please note that there is a possibility that Reset section (1μsec) cannot be satisfied if wait time is not enough long after turning off VCC. (It is necessary to consider DVI voltage level after turning off VCC.)



・Please do the application design to secure Reset section 1us or more after the reclosing of the power supply.

◆ Example of designing set when CR (C = 1μF, R = 1kΩ) is inserted between VCC and DVI with VCC=2.8V

①The rise time to 0→2.4V of VCC must use the power supply of 100μs or less.
②Please wait 25ms or more after VCC turn off (VCC <= 0.05V), because it is necessary to secure reset section (1μs or more).

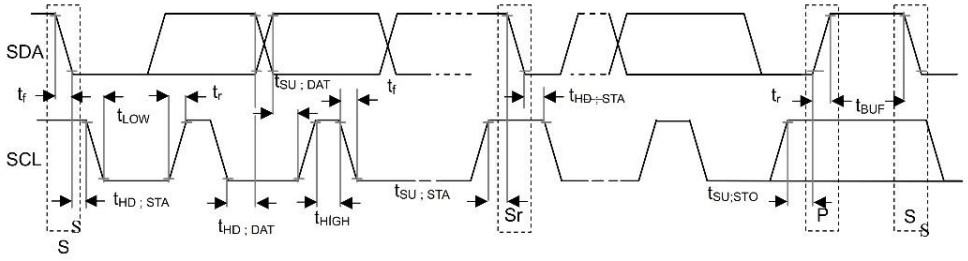


・ Please do the application design to secure Reset section 1us or more after the reclosing of the power supply.

●I²C Bus Access

1) I²C Bus Interface Timing chart

Write measurement command and Read measurement result are done by I²C Bus interface. Please refer the formally specification of I²C Bus interface, and follow the formally timing chart.



2) Slave Address

Slave Address is 2 types, it is determined by ADDR Terminal

ADDR = 'H' (ADDR ≧ 0.7VCC) → "1011100"

ADDR = 'L' (ADDR ≦ 0.3VCC) → "0100011"

3) Write Format

BH1750FVI is not able to accept plural command without stop condition. Please insert SP every 1 Opecode.

ST	Slave Address	R/W 0	Ack	Opecode	Ack	SP

4) Read Format

ST	Slave Address	R/W 1	Ack	High Byte [15:8] 2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^{9} 2^{8}	Ack

Low Byte [7:0] 2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0}	$\overline{\text{Ack}}$	SP

■ from Master to Slave □ from Slave to Master

ex)

High Byte = "1000_0011"

Low Byte = "1001_0000"

$(2^{15} + 2^{9} + 2^{8} + 2^{7} + 2^{4}) / 1.2 \fallingdotseq 28067$ [lx]

* I²C BUS is trademark of Phillips Semiconductors. Please refer formality specification.

●Adjust measurement result for influence of optical window. (sensor sensitivity adjusting)

BH1750FVI is possible to change sensor sensitivity. And it is possible to cancel the optical window influence (difference with / without optical window) by using this function. Adjust is done by changing measurement time. For example, when transmission rate of optical window is 50% (measurement result becomes 0.5 times if optical window is set), influence of optical window is ignored by changing sensor sensitivity from default to 2 times

Sensor sensitivity is shift by changing the value of MTreg (measurement time register). MTreg value has to set 2 times if target of sensor sensitivity is 2 times. Measurement time is also set 2 times when MTreg value is changed from default to 2 times.

ex) Procedure for changing target sensor sensitivity to 2 times.

Please change Mtreg from "0100_0101" (default) to "1000_1010" (default * 2).

1) Changing High bit of MTreg

ST	Slave Address	R/W 0	Ack	01000_100	Ack	SP

2) Changing Low bit of MTreg

ST	Slave Address	R/W 0	Ack	011_01010	Ack	SP

3) Input Measurement Command

ST	Slave Address	R/W 0	Ack	0001_0000	Ack	SP

* This example is High Resolution mode, but it accepts the other measurement.

4) After about 240ms, measurement result is registered to Data Register.
(High Resolution mode is typically 120ms, but measurement time is set twice.)

The below table is seeing the changable range of MTreg.

		Min.	Typ.	Max.
changeable range of MTreg	binary	0001_1111 (sensitivity : default * 0.45)	0100_0101 default	1111_1110 (sensitivity : default * 3.68)
	decimal	31 (sensitivity : default * 0.45)	69 default	254 (sensitivity : default * 3.68)

It is possilbe to detect 0.23lx by using this function at H-resolution mode. And it is possilbe to detect 0.11lx by using this function at H-resolution mode2.

The below formula is to calculate illuminance per 1 count.

H-reslution mode : Illuminance per 1 count (lx / count) = 1 / 1.2 *(69 / X)
H-reslution mode2 : Illuminance per 1 count (lx / count) = 1 / 1.2 *(69 / X) / 2

1.2 : Measurement accuracy
69 : Default value of MTreg (dec)
X : MTreg value

The below table is seeing the detail of resolution.

Mtreg の値	lx / count at H-resolutin mode	lx / count at H-resolution mode2
0001_1111	1.85	0.93
0100_0101	0.83	0.42
1111_1110	0.23	0.11

●H-Resolution Mode2

H-resolution mode2 is 0.5lx (typ.) resolution mode. It is suitable if under less than 10 lx measure ment data is necessary. This measurement mode supports " Adjust measurement result for influence of optical window ". Please refer it. It is possible to detect min. 0.11 lx by using H-resolution mode2.

○ Instruction set architecture for H-resolution mode2

Instruction	Opecode	Comments
Continuously H-Resolution Mode2	0001_0001	Start measurement at 0.5lx resolution. Measurement Time is typically 120ms.
One Time H-Resolution Mode2	0010_0001	Start measurement at 0.5lx resolution. Measurement Time is typically 120ms. It is automatically set to Power Down mode after measurement.

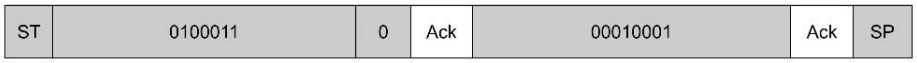
○ Measurement sequence example from "Write instruction" to "Read measurement result"

ex) Continuously H-resolution mode2 (ADDR = 'L')

from Master to Slave

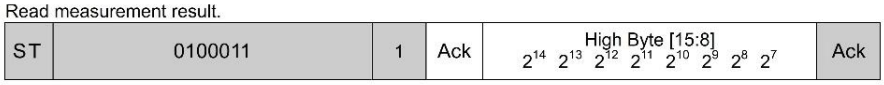
from Slave to Master

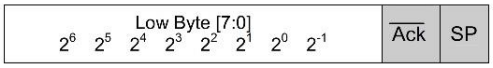
① Send "Continuously H-resolution mode2 " instruction



② Wait to complete 1st H-resolution mode2 measurement.(max. 180ms.)

③ Read measurement result.





How to calculate when the data High Byte is "00000000" and Low Byte is "00010010"

$(2^{3} + 2^{0}) / 1.2 \fallingdotseq 7.5$ [lx]

●H-Resolution Mode2

H-resolution mode2 is 0.5lx (typ.) resolution mode. It is suitable if under less than 10 lx measure ment data is necessary. This measurement mode supports " Adjust measurement result for influence of optical window ". Please refer it. It is possible to detect min. 0.11 lx by using H-resolution mode2.

○ Instruction set architecture for H-resolution mode2

Instruction	Opecode	Comments
Continuously H-Resolution Mode2	0001_0001	Start measurement at 0.5lx resolution. Measurement Time is typically 120ms.
One Time H-Resolution Mode2	0010_0001	Start measurement at 0.5lx resolution. Measurement Time is typically 120ms. It is automatically set to Power Down mode after measurement.

○ Measurement sequence example from "Write instruction" to "Read measurement result"

ex) Continuously H-resolution mode2 (ADDR = 'L')

from Master to Slave

from Slave to Master

① Send "Continuously H-resolution mode2 " instruction

ST	0100011	0	Ack	00010001	Ack	SP

② Wait to complete 1st H-resolution mode2 measurement.(max. 180ms.)

③ Read measurement result.

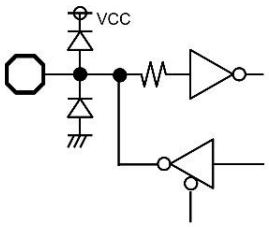
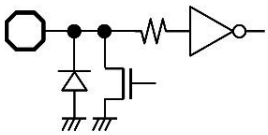
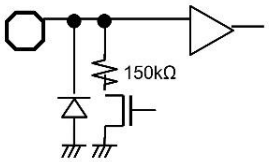
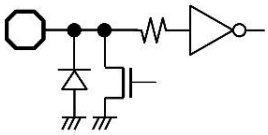
ST	0100011	1	Ack	High Byte [15:8] 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^{9} 2^{8} 2^{7}	Ack

Low Byte [7:0] 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0} 2^{-1}	$\overline{\text{Ack}}$	SP

How to calculate when the data High Byte is "00000000" and Low Byte is "00010010"

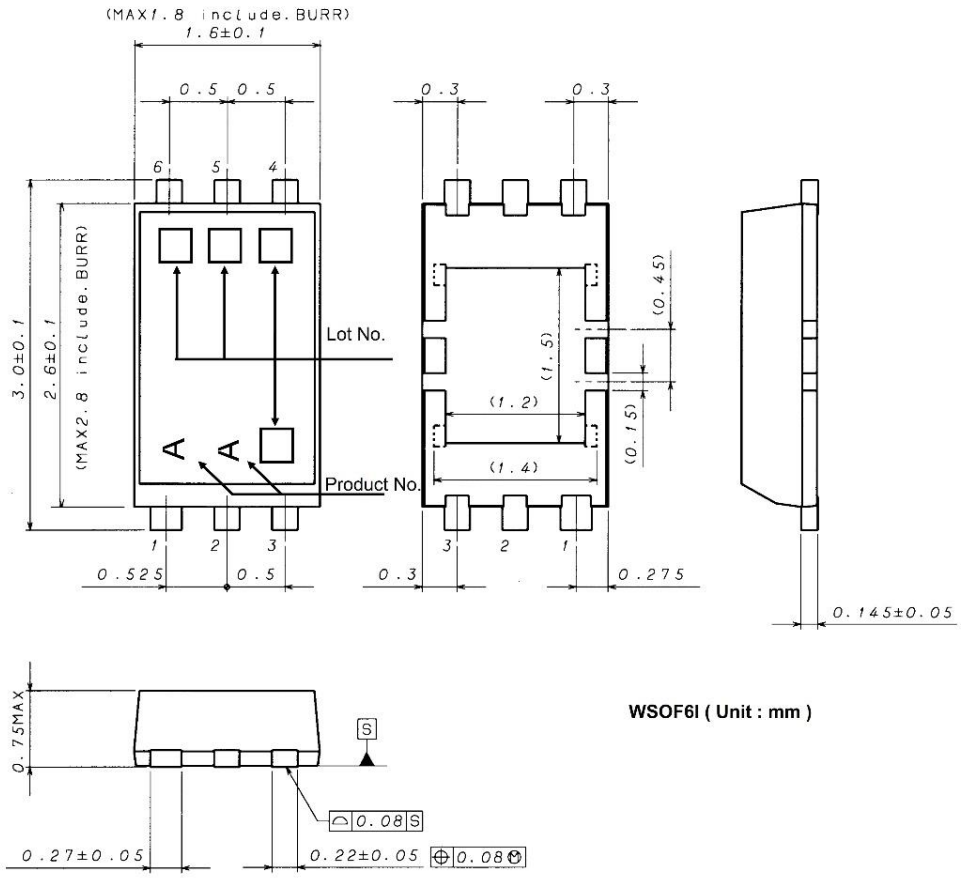
$(2^{3} + 2^{0}) / 1.2 \fallingdotseq 7.5$ [lx]

●**Terminal Description**

PIN No.	Terminal Name	Equivalent Circuit	Function
1	VCC		Power Supply Terminal
2	ADDR		I^2C Slave-address Terminal ADDR = 'H' (ADDR ≧ 0.7Vcc) "1011100" ADDR = 'L' (ADDR ≦ 0.3Vcc) "0100011" ADDR Terminal is designed as 3 state buffer for internal test. So that please take care of Vcc and DVI supply procedure.Please see P6.
3	GND		GND Terminal
4	SDA		I^2C bus Interface SDA Terminal
5	DVI		SDA, SCL Reference Voltage Terminal And DVI Terminal is also asynchronous Reset for internal registers.So that please set to 'L' (at least 1μs, DVI <= 0.4V) after Vcc is supplied. BH1750FVI is pulled down by 150kOhm while DVI = 'L'.
6	SCL		I^2C bus Interface SCL Terminal

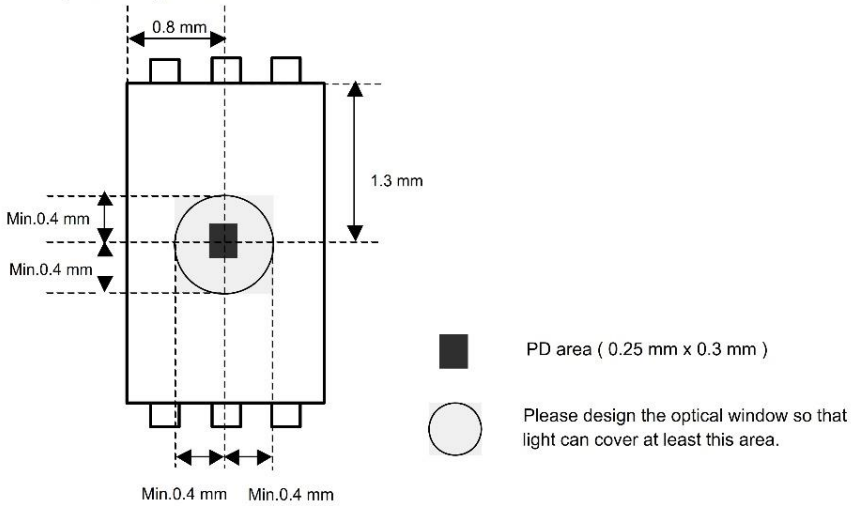
※These values are design-value, not guaranteed.

●Package Outlines



WSOF6I (Unit : mm)

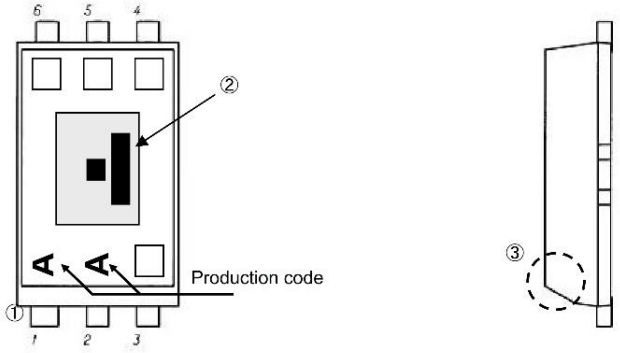
●About an optical design on the device



●The method of distinguishing 1pin.

There is some method of distinguishing 1pin.
① Distinguishing by 1Pin wide-lead
② Distinguishing by die pattern
③ Distinguishing by taper part of 1-3pin side

② (by die patern) is the easiest method to distinguish by naked eye.



●**Cautions on use**

1) Absolute Maximum Ratings
An excess in the absolute maximum ratings, such as supply voltage (Vmax), temperature range of operating conditions (Topr), etc., can break down devices, thus making impossible to identify breaking mode such as a short circuit or an open circuit. If any special mode exceeding the absolute maximum ratings is assumed, consideration should be given to take physical safety measures including the use of fuses, etc.

2) GND voltage
Make setting of the potential of the GND terminal so that it will be maintained at the minimum in any operating state. Furthermore, check to be sure no terminals are at a potential lower than the GND voltage including an actual electric transient.

3) Short circuit between terminals and erroneous mounting
In order to mount ICs on a set PCB, pay thorough attention to the direction and offset of the ICs. Erroneous mounting can break down the ICs. Furthermore, if a short circuit occurs due to foreign matters entering between terminals or between the terminal and the power supply or the GND terminal, the ICs can break down.

4) Operation in strong electromagnetic field
Be noted that using ICs in the strong electromagnetic field can malfunction them.

5) Inspection with set PCB
On the inspection with the set PCB, if a capacitor is connected to a low-impedance IC terminal, the IC can suffer stress. Therefore, be sure to discharge from the set PCB by each process. Furthermore, in order to mount or dismount the set PCB to/from the jig for the inspection process, be sure to turn OFF the power supply and then mount the set PCB to the jig. After the completion of the inspection, be sure to turn OFF the power supply and then dismount it from the jig. In addition, for protection against static electricity, establish a ground for the assembly process and pay thorough attention to the transportation and the storage of the set PCB.

6) Input terminals
In terms of the construction of IC, parasitic elements are inevitably formed in relation to potential. The operation of the parasitic element can cause interference with circuit operation, thus resulting in a malfunction and then breakdown of the input terminal. Therefore, pay thorough attention not to handle the input terminals; such as to apply to the input terminals a voltage lower than the GND respectively, so that any parasitic element will operate. Furthermore, do not apply a voltage to the input terminals when no power supply voltage is applied to the IC. In addition, even if the power supply voltage is applied, apply to the input terminals a voltage lower than the power supply voltage or within the guaranteed value of electrical characteristics.

7) Thermal design
Perform thermal design in which there are adequate margins by taking into account the power dissipation (Pd) in actual states of use.

8) Treatment of package
Dusts or scratch on the photo detector may affect the optical characteristics. Please handle it with care.

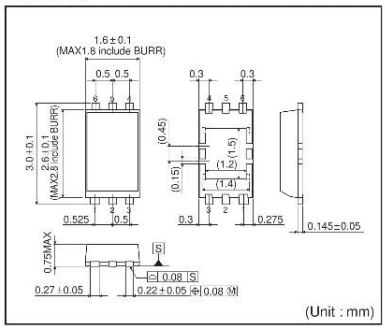
9) Rush current
When power is first supplied to the CMOS IC, it is possible that the internal logic may be unstable and rush current may flow instantaneously. Therefore, give special consideration to power coupling capacitance, power wiring, width of GND wiring, and routing of connections.

10) The exposed central pad on the back side of the package
There is an exposed central pad on the back side of the package. But please do it non connection. (Don't solder, and don't do electrical connection) Please mount by Footprint dimensions described in the Jisso Information for WSOF6I. This pad is GND level, therefore there is a possibility that LSI malfunctions and heavy-current is generated.

●**Ordering part number**

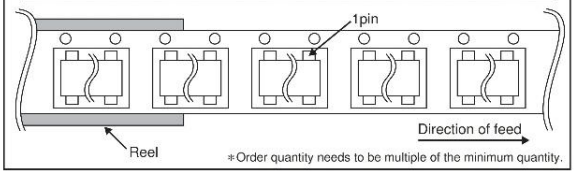
B	H	1	7	5	0	F	V	I	-	T	R
Part No.		Part No.				Package FVI: WSOF6I				Packaging and forming specification TR: Embossed tape and reel	

WSOF6I



<Tape and Reel information>

Tape	Embossed carrier tape
Quantity	3000pcs
Direction of feed	TR (The direction is the 1pin of product is at the upper right when you hold reel on the left hand and you pull out the tape on the right hand)



∗Order quantity needs to be multiple of the minimum quantity.

Notice

Precaution on using ROHM Products

1. Our Products are designed and manufactured for application in ordinary electronic equipments (such as AV equipment, OA equipment, telecommunication equipment, home electronic appliances, amusement equipment, etc.). If you intend to use our Products in devices requiring extremely high reliability (such as medical equipment (Note 1), transport equipment, traffic equipment, aircraft/spacecraft, nuclear power controllers, fuel controllers, car equipment including car accessories, safety devices, etc.) and whose malfunction or failure may cause loss of human life, bodily injury or serious damage to property ("Specific Applications"), please consult with the ROHM sales representative in advance. Unless otherwise agreed in writing by ROHM in advance, ROHM shall not be in any way responsible or liable for any damages, expenses or losses incurred by you or third parties arising from the use of any ROHM's Products for Specific Applications.

(Note1) Medical Equipment Classification of the Specific Applications

JAPAN	USA	EU	CHINA
CLASSⅢ	CLASSⅢ	CLASSⅡb	CLASSⅢ
CLASSⅣ		CLASSⅢ	

2. ROHM designs and manufactures its Products subject to strict quality control system. However, semiconductor products can fail or malfunction at a certain rate. Please be sure to implement, at your own responsibilities, adequate safety measures including but not limited to fail-safe design against the physical injury, damage to any property, which a failure or malfunction of our Products may cause. The following are examples of safety measures:
   [a] Installation of protection circuits or other protective devices to improve system safety
   [b] Installation of redundant circuits to reduce the impact of single or multiple circuit failure

3. Our Products are designed and manufactured for use under standard conditions and not under any special or extraordinary environments or conditions, as exemplified below. Accordingly, ROHM shall not be in any way responsible or liable for any damages, expenses or losses arising from the use of any ROHM's Products under any special or extraordinary environments or conditions. If you intend to use our Products under any special or extraordinary environments or conditions (as exemplified below), your independent verification and confirmation of product performance, reliability, etc, prior to use, must be necessary:
   [a] Use of our Products in any types of liquid, including water, oils, chemicals, and organic solvents
   [b] Use of our Products outdoors or in places where the Products are exposed to direct sunlight or dust
   [c] Use of our Products in places where the Products are exposed to sea wind or corrosive gases, including Cl_2, H_2S, NH_3, SO_2, and NO_2
   [d] Use of our Products in places where the Products are exposed to static electricity or electromagnetic waves
   [e] Use of our Products in proximity to heat-producing components, plastic cords, or other flammable items
   [f] Sealing or coating our Products with resin or other coating materials
   [g] Use of our Products without cleaning residue of flux (even if you use no-clean type fluxes, cleaning residue of flux is recommended); or Washing our Products by using water or water-soluble cleaning agents for cleaning residue after soldering
   [h] Use of the Products in places subject to dew condensation

4. The Products are not subject to radiation-proof design.

5. Please verify and confirm characteristics of the final or mounted products in using the Products.

6. In particular, if a transient load (a large amount of load applied in a short period of time, such as pulse. is applied, confirmation of performance characteristics after on-board mounting is strongly recommended. Avoid applying power exceeding normal rated power; exceeding the power rating under steady-state loading condition may negatively affect product performance and reliability.

7. De-rate Power Dissipation (Pd) depending on Ambient temperature (Ta). When used in sealed area, confirm the actual ambient temperature.

8. Confirm that operation temperature is within the specified range described in the product specification.

9. ROHM shall not be in any way responsible or liable for failure induced under deviant condition from what is defined in this document.

Precaution for Mounting / Circuit board design

1. When a highly active halogenous (chlorine, bromine, etc.) flux is used, the residue of flux may negatively affect product performance and reliability.

2. In principle, the reflow soldering method must be used; if flow soldering method is preferred, please consult with the ROHM representative in advance.

For details, please refer to ROHM Mounting specification

Precautions Regarding Application Examples and External Circuits

1. If change is made to the constant of an external circuit, please allow a sufficient margin considering variations of the characteristics of the Products and external components, including transient characteristics, as well as static characteristics.

2. You agree that application notes, reference designs, and associated data and information contained in this document are presented only as guidance for Products use. Therefore, in case you use such information, you are solely responsible for it and you must exercise your own independent verification and judgment in the use of such information contained in this document. ROHM shall not be in any way responsible or liable for any damages, expenses or losses incurred by you or third parties arising from the use of such information.

Precaution for Electrostatic

This Product is electrostatic sensitive product, which may be damaged due to electrostatic discharge. Please take proper caution in your manufacturing process and storage so that voltage exceeding the Products maximum rating will not be applied to Products. Please take special care under dry condition (e.g. Grounding of human body / equipment / solder iron, isolation from charged objects, setting of Ionizer, friction prevention and temperature / humidity control).

Precaution for Storage / Transportation

1. Product performance and soldered connections may deteriorate if the Products are stored in the places where:
   [a] the Products are exposed to sea winds or corrosive gases, including Cl2, H2S, NH3, SO2, and NO2
   [b] the temperature or humidity exceeds those recommended by ROHM
   [c] the Products are exposed to direct sunshine or condensation
   [d] the Products are exposed to high Electrostatic

2. Even under ROHM recommended storage condition, solderability of products out of recommended storage time period may be degraded. It is strongly recommended to confirm solderability before using Products of which storage time is exceeding the recommended storage time period.

3. Store / transport cartons in the correct direction, which is indicated on a carton with a symbol. Otherwise bent leads may occur due to excessive stress applied when dropping of a carton.

4. Use Products within the specified time after opening a humidity barrier bag. Baking is required before using Products of which storage time is exceeding the recommended storage time period.

Precaution for Product Label

QR code printed on ROHM Products label is for ROHM's internal use only.

Precaution for Disposition

When disposing Products please dispose them properly using an authorized industry waste company.

Precaution for Foreign Exchange and Foreign Trade act

Since our Products might fall under controlled goods prescribed by the applicable foreign exchange and foreign trade act, please consult with ROHM representative in case of export.

Precaution Regarding Intellectual Property Rights

1. All information and data including but not limited to application example contained in this document is for reference only. ROHM does not warrant that foregoing information or data will not infringe any intellectual property rights or any other rights of any third party regarding such information or data. ROHM shall not be in any way responsible or liable for infringement of any intellectual property rights or other damages arising from use of such information or data.:

2. No license, expressly or implied, is granted hereby under any intellectual property rights or other rights of ROHM or any third parties with respect to the information contained in this document.

Other Precaution

1. This document may not be reprinted or reproduced, in whole or in part, without prior written consent of ROHM.

2. The Products may not be disassembled, converted, modified, reproduced or otherwise changed without prior written consent of ROHM.

3. In no event shall you use in any way whatsoever the Products and the related technical information contained in the Products or this document for any military purposes, including but not limited to, the development of mass-destruction weapons.

4. The proper names of companies or products described in this document are trademarks or registered trademarks of ROHM, its affiliated companies or third parties.

General Precaution

1. Before you use our Products, you are requested to carefully read this document and fully understand its contents. ROHM shall not be in any way responsible or liable for failure, malfunction or accident arising from the use of any ROHM's Products against warning, caution or note contained in this document.

2. All information contained in this document is current as of the issuing date and subject to change without any prior notice. Before purchasing or using ROHM's Products, please confirm the latest information with a ROHM sales representative.

3. The information contained in this document is provided on an "as is" basis and ROHM does not warrant that all information contained in this document is accurate and/or error-free. ROHM shall not be in any way responsible or liable for any damages, expenses or losses incurred by you or third parties resulting from inaccuracy or errors of or concerning such information.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

ROHM Semiconductor:
BH1750FVI-TR

- **ANEXO 05: DHT11 Humidity & Temperature Sensor**

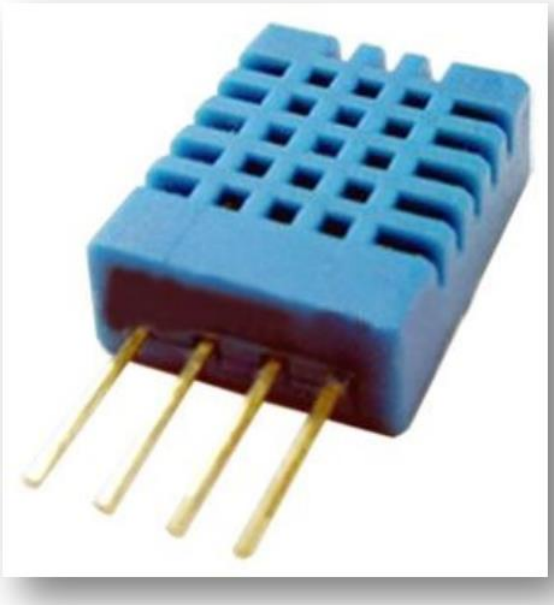
DHT11 Humidity & Temperature Sensor

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output.

DHT 11 Humidity & Temperature Sensor

1. Introduction

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability. This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high-performance 8-bit microcontroller, offering excellent quality, fast response, anti-interference ability and cost-effectiveness.



Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programmes in the OTP memory, which are used by the sensor's internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up-to-20 meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect and special packages can be provided according to users' request.

2. Technical Specifications:

Overview:

Item	Measurement Range	Humidity Accuracy	Temperature Accuracy	Resolution	Package
DHT11	20-90%RH 0-50 ℃	±5%RH	±2℃	1	4 Pin Single Row

Page | 3

Detailed Specifications:

Parameters	Conditions	Minimum	Typical	Maximum
Humidity				
Resolution		1%RH	1%RH	1%RH
			8 Bit	
Repeatability			±1%RH	
Accuracy	25℃		±4%RH	
	0-50℃			±5%RH
Interchangeability	Fully Interchangeable			
Measurement Range	0℃	30%RH		90%RH
	25℃	20%RH		90%RH
	50℃	20%RH		80%RH
Response Time (Seconds)	1/e(63%)25℃, 1m/s Air	6 S	10 S	15 S
Hysteresis			±1%RH	
Long-Term Stability	Typical		±1%RH/year	
Temperature				
Resolution		1℃	1℃	1℃
		8 Bit	8 Bit	8 Bit
Repeatability			±1℃	
Accuracy		±1℃		±2℃
Measurement Range		0℃		50℃
Response Time (Seconds)	1/e(63%)	6 S		30 S

3. Typical Application (Figure 1)

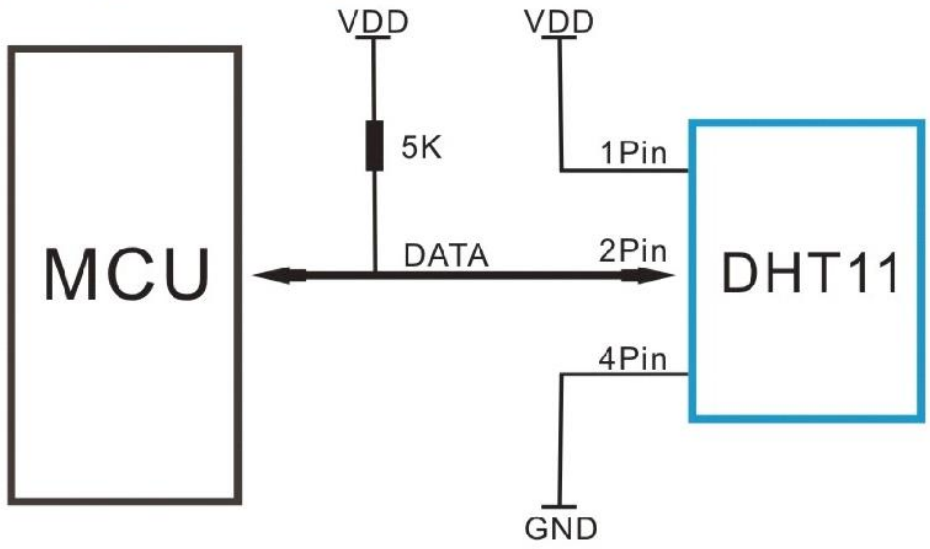


Figure 1 Typical Application

Note: 3Pin – Null; MCU = Micro-computer Unite or single chip Computer

When the connecting cable is shorter than 20 metres, a 5K pull-up resistor is recommended; when the connecting cable is longer than 20 metres, choose a appropriate pull-up resistor as needed.

4. Power and Pin

DHT11's power supply is 3-5.5V DC. When power is supplied to the sensor, do not send any instruction to the sensor in within one second in order to pass the unstable status. One capacitor valued 100nF can be added between VDD and GND for power filtering.

5. Communication Process: Serial Interface (Single-Wire Two-Way)

Single-bus data format is used for communication and synchronization between MCU and DHT11 sensor. One communication process is about 4ms.

Data consists of decimal and integral parts. A complete data transmission is **40bit**, and the sensor sends **higher data bit** first.
Data format: 8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data + 8bit check sum. If the data transmission is right, the check-sum should be the last 8bit of "8bit integral RH data + 8bit decimal RH data + 8bit integral T data + 8bit decimal T data".

5.1 Overall Communication Process (Figure 2, below)

When MCU sends a start signal, DHT11 changes from the low-power-consumption mode to the running-mode, waiting for MCU completing the start signal. Once it is completed, DHT11 sends a response signal of 40-bit data that include the relative humidity and temperature information to MCU. Users can choose to collect (read) some data. Without the start signal from MCU, DHT11 will not give the response signal to MCU. Once data is collected, DHT11 will change to the low-power-consumption mode until it receives a start signal from MCU again.

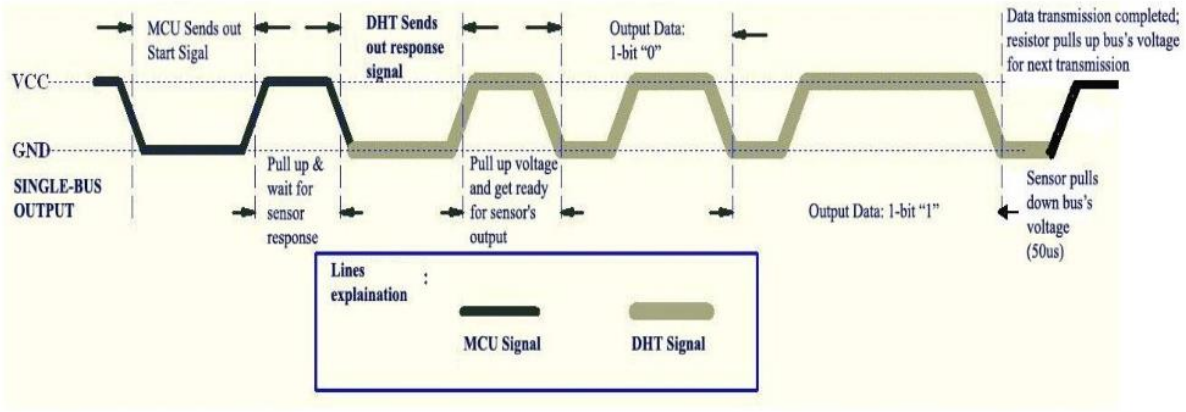


Figure 2 Overall Communication Process

5.2 MCU Sends out Start Signal to DHT (Figure 3, below)

Data Single-bus free status is at high voltage level. When the communication between MCU and DHT11 begins, the programme of MCU will set Data Single-bus voltage level from high to low and this process must take at least 18ms to ensure DHT's detection of MCU's signal, then MCU will pull up voltage and wait 20-40us for DHT's response.

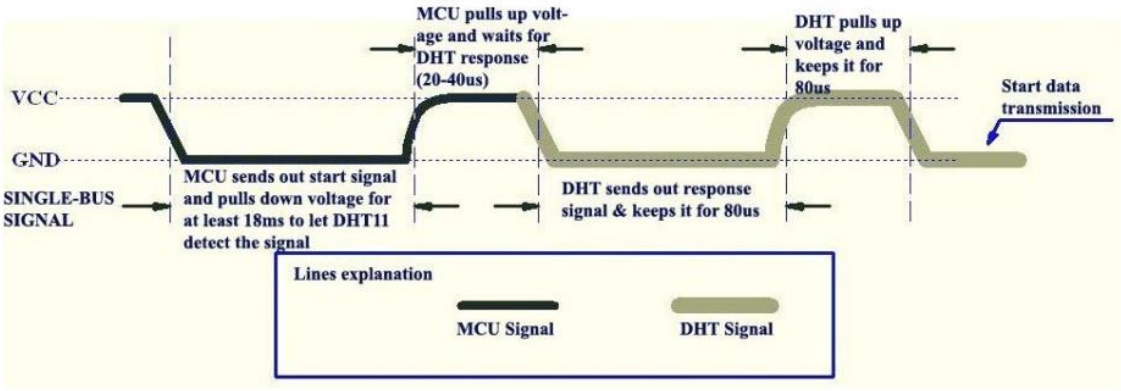


Figure 3 MCU Sends out Start Signal & DHT Responses

5.3 DHT Responses to MCU (Figure 3, above)

Once DHT detects the start signal, it will send out a low-voltage-level response signal, which lasts 80us. Then the programme of DHT sets Data Single-bus voltage level from low to high and keeps it for 80us for DHT's preparation for sending data.

When DATA Single-Bus is at the low voltage level, this means that DHT is sending the response signal. Once DHT sent out the response signal, it pulls up voltage and keeps it for 80us and prepares for data transmission.

When DHT is sending data to MCU, every bit of data begins with the 50us low-voltage-level and the length of the following high-voltage-level signal determines whether data bit is "0" or "1" (see Figures 4 and 5 below).

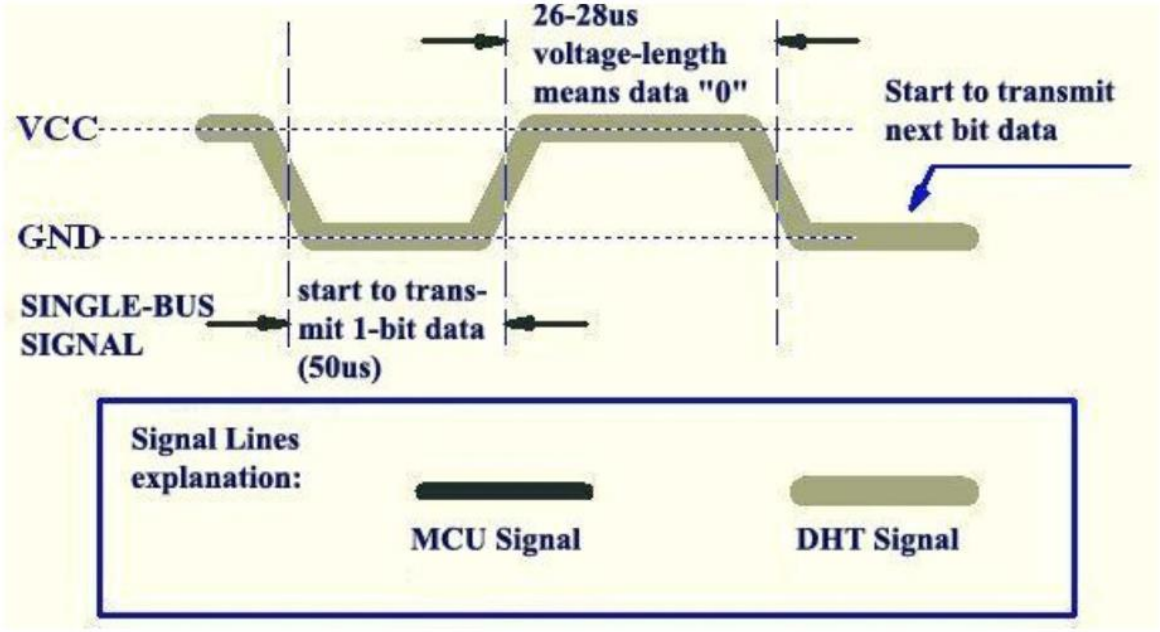


Figure 4 Data "0" Indication

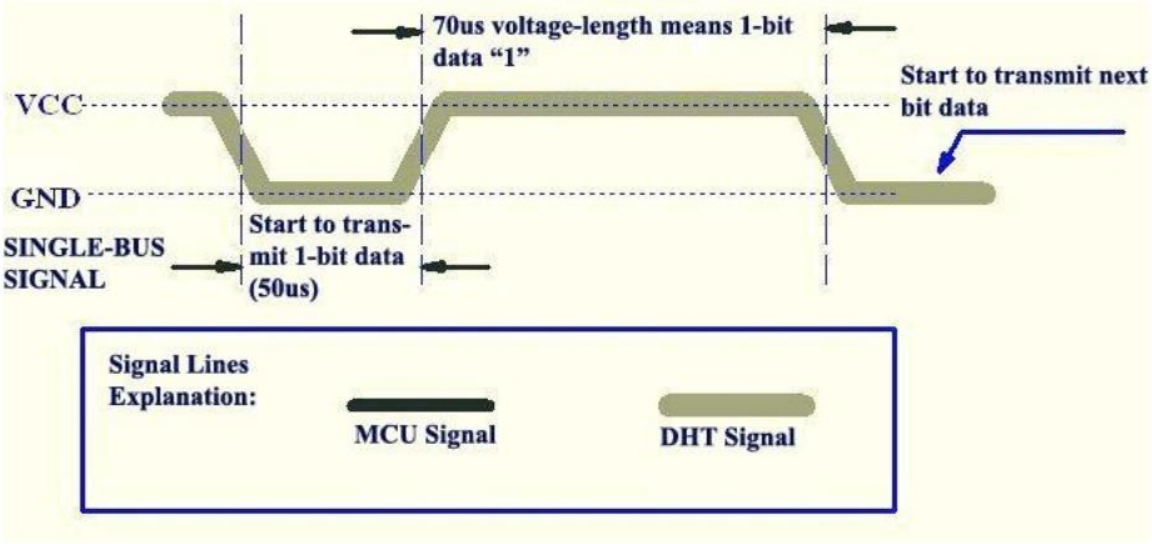


Figure 5 Data "1" Indication

If the response signal from DHT is always at high-voltage-level, it suggests that DHT is not responding properly and please check the connection. When the last bit data is transmitted, DHT11 pulls down the voltage level and keeps it for 50us. Then the Single-Bus voltage will be pulled up by the resistor to set it back to the free status.

6. Electrical Characteristics

VDD=5V, T = 25℃ (unless otherwise stated)

	Conditions	Minimum	Typical	Maximum
Power Supply	DC	3V	5V	5.5V
Current Supply	Measuring	0.5mA		2.5mA
	Average	0.2mA		1mA
	Standby	100uA		150uA
Sampling period	Second	1		

Note: Sampling period at intervals should be no less than 1 second.

7. Attentions of application

(1) Operating conditions

Applying the DHT11 sensor beyond its working range stated in this datasheet can result in 3%RH signal shift/discrepancy. The DHT11 sensor can recover to the calibrated status gradually when it gets back to the normal operating condition and works within its range. Please refer to (3) of

this section to accelerate its recovery. Please be aware that operating the DHT11 sensor in the non-normal working conditions will accelerate sensor's aging process.

(2) Attention to chemical materials

Vapor from chemical materials may interfere with DHT's sensitive-elements and debase its sensitivity. A high degree of chemical contamination can permanently damage the sensor.

(3) Restoration process when (1) & (2) happen

Step one: Keep the DHT sensor at the condition of Temperature 50~60Celsius, humidity <10%RH for 2 hours;
Step two:K keep the DHT sensor at the condition of Temperature 20~30Celsius, humidity >70%RH for 5 hours.

(4) Temperature Affect

Relative humidity largely depends on temperature. Although temperature compensation technology is used to ensure accurate measurement of RH, it is still strongly advised to keep the humidity and temperature sensors working under the same temperature. DHT11 should be mounted at the place as far as possible from parts that may generate heat.

(5) Light Affect

Long time exposure to strong sunlight and ultraviolet may debase DHT's performance.

(6) Connection wires

The quality of connection wires will affect the quality and distance of communication and high quality shielding-wire is recommended.

(7) Other attentions

* Welding temperature should be bellow 260Celsius and contact should take less than 10 seconds.
* Avoid using the sensor under dew condition.
* Do not use this product in safety or emergency stop devices or any other occasion that failure of DHT11 may cause personal injury.
* Storage: Keep the sensor at temperature 10-40℃, humidity <60%RH.

Disclaimer

This is a translated version of the manufacturer's data sheet. OSEPP is not responsible for the accuracy of the translated information.

Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

OSEPP Electronics:
HUMI-01

- **ANEXO 06: Código fuente la herramienta de monitoreo y control de datos (Dashboard)**

❖ Modelos del sistema.

```
//Aduino: //configuración Sensor01 ARCHIVO: arduinov3.ino

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

//------------------VARIABLES GLOBALES-------------------------

int contconexion = 0;

//const char *ssid = "corbachop";

//const char *password = "Cor*Ba:cHop:D";

const char *ssid = "Iphone Mijail";

const char *password = "123456789a";

unsigned long previousMillis = 0;

float sensorT1 = 1.4;

float sensorL1 = 14;

float sensorT2 = 50;

float sensorL2 = 35;

float sensorT3 = 0.5;

float sensorL3 = 15;

float sensorT4 = 7.5;

float sensorL4 = 4.5;

char host[48];

String strhost = "datalogger.mcorbachop.com";

String strurl = "/enviardatos.php";


//-------Función para Enviar Datos a la Base de Datos SQL--------

String enviardatos(String datos) {

  String linea = "error";

  WiFiClient client;
```

```
strhost.toCharArray(host, 49);
if (!client.connect(host, 80)) {
  Serial.println("Fallo de conexion");
  return linea;
}
client.print(String("POST ") + strurl + " HTTP/1.0" + "\r\n" +
             "Host: " + strhost + "\r\n" +
             "Accept: */*" + "*\r\n" +
             "Content-Length: " + datos.length() + "\r\n" +
             "Content-Type: application/x-www-form-urlencoded" + "\r\n" +
             "\r\n" + datos);
delay(10);
Serial.print("Enviando datos a SQL...");
unsigned long timeout = millis();
while (client.available() == 0) {
  if (millis() - timeout > 5000) {
    Serial.println("Cliente fuera de tiempo!");
    client.stop();
    return linea;
  }
}
// Lee todas las lineas que recibe del servidor y las imprime por la terminal serial
while(client.available()){
  linea = client.readStringUntil('\r');
}
Serial.println(linea);
return linea;
}
```

```
//--------------------------------------------------------------------

void setup() {

  // Inicia Serial
  Serial.begin(115200);
  Serial.println("");

  // Conexión WIFI
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED and contconexion <50) {
  //Cuenta hasta 50 si no se puede conectar lo cancela
    ++contconexion;
    delay(500);
    Serial.print(".");
  }
  if (contconexion <50) {


      Serial.println("");
      Serial.println("WiFi conectado");
      Serial.println(WiFi.localIP());
  }
  else {
      Serial.println("");
      Serial.println("Error de conexion");
  }
}
```

```
//-------------------------LOOP------------------------------
void loop() {


  unsigned long currentMillis = millis();


  if (currentMillis - previousMillis >= 10000) { //envia la temperatura cada 10 segundos
    previousMillis = currentMillis;
    int analog = analogRead(17);
    float temp = analog*0.322265625;
    Serial.println(temp);
    enviardatos("sensorT1=" + String(sensorT1, 2) +" &sensorL1=" + String(sensorL1, 2) + "&sensorT2=" + String(sensorT2,2 )+ "&sensorL2=" + String(sensorL2, 2) + "&sensorT3=" + String(sensorT3, 2) + "&sensorL3=" + String(sensorL3, 2) + "&sensorT4=" + String(sensorT4, 2) + "&sensorL4=" + String(sensorL4, 2));
    sensorT1 = sensorT1 * 1.01 ;
    sensorL1 = sensorL1 + 2.5;
    sensorT2 = sensorT2 / 1.02 ;
    sensorL2 = sensorL2 + 0.59;
    sensorT3 = sensorT3 * 1.20 ;
    sensorL3 = sensorL3 + 0.82;
    sensorT4 = sensorT4 / 0.9 ;
    sensorL4 = sensorL4 - 0.45;


  }
}
//ADUINO: //CONFIGURACIÓN SENSOR02 ARCHIVO: ARDUINOV2.INO
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
// nuevo codigo
#include <Wire.h>
```

```
#include <BH1750.h>
#include <DHT.h>
#define DHTPIN D3
#define DHTTYPE DHT11
BH1750 lightMeter;
// Inicializamos el sensor DHT11
DHT dht(DHTPIN, DHTTYPE);

//-------------------VARIABLES GLOBALES-------------------------
int contconexion = 0;

//const char *ssid = "corbachop";
//const char *password = "Cor*Ba:cHop:D";
const char *ssid = "Iphone Mijail";
const char *password = "123456789a";
unsigned long previousMillis = 0;
float sensorT1 = 1.4;
float sensorL1 = 14;
float sensorT2 = 50;
float sensorL2 = 35;
float sensorT3 = 0.5;
float sensorL3 = 15;
float sensorT4 = 7.5;
float sensorL4 = 4.5;
char host[48];
String strhost = "datalogger.mcorbachop.com";
String strurl = "/enviardatos.php";
//-------Función para Enviar Datos a la Base de Datos SQL--------
String enviardatos(String datos) {
```

```
String linea = "error";
WiFiClient client;
strhost.toCharArray(host, 49);
if (!client.connect(host, 80)) {
  Serial.println("Fallo de conexion");
  return linea;
}
client.print(String("POST ") + strurl + " HTTP/1.0" + "\r\n" +
             "Host: " + strhost + "\r\n" +
             "Accept: */*" + "*\r\n" +
             "Content-Length: " + datos.length() + "\r\n" +
             "Content-Type: application/x-www-form-urlencoded" + "\r\n" +
             "\r\n" + datos);
delay(10);
Serial.print("Enviando datos a SQL...");
unsigned long timeout = millis();
while (client.available() == 0) {
  if (millis() - timeout > 5000) {
    Serial.println("Cliente fuera de tiempo!");
    client.stop();
    return linea;
  }
}
// Lee todas las lineas que recibe del servidor y las imprime por la terminal serial
while(client.available()){
  linea = client.readStringUntil('\r');
}
Serial.println(linea);
return linea;
```

```
}

//------------------------------------------------------------------------

void setup() {

 // Inicia Serial
 Serial.begin(115200);
 Serial.println("");
 Wire.begin();
 dht.begin();
 lightMeter.begin();
 Serial.println(F("BH1750 Test"));
 // Conexión WIFI
 WiFi.begin(ssid, password);
 while (WiFi.status() != WL_CONNECTED and contconexion <50) {
 //Cuenta hasta 50 si no se puede conectar lo cancela
  ++contconexion;
  delay(500);
  Serial.print(".");
 }
 if (contconexion <50) {

    Serial.println("");
    Serial.println("WiFi conectado");
    Serial.println(WiFi.localIP());
 }
 else {
    Serial.println("");
```

```
      Serial.println("Error de conexion");
  }
}
//-------------------------LOOP------------------------------
void loop() {
 float t = dht.readTemperature();
 float lux = lightMeter.readLightLevel();
  //Serial.print("Light: ");
  if ( isnan(t)) {
    Serial.println("Error obteniendo los datos del sensor DHT11");
    return;
  }
  unsigned long currentMillis = millis();

  if (currentMillis - previousMillis >= 10000) { //envia la temperatura cada 10 segundos
    previousMillis = currentMillis;
    sensorT1 = t ;
    sensorL1 = lux;
    sensorT2 = sensorT2 / 1.02 ;
    sensorL2 = sensorL2 + 0.59;
    sensorT3 = sensorT3 * 1.20 ;
    sensorL3 = sensorL3 + 0.82;
    sensorT4 = sensorT4 / 0.9 ;
    sensorL4 = sensorL4 - 0.45;
    enviardatos("sensorT1=" + String(sensorT1, 2) +" &sensorL1=" + String(sensorL1,
2) + "&sensorT2=" + String(sensorT2,2 )+ "&sensorL2=" + String(sensorL2, 2) +
"&sensorT3=" + String(sensorT3, 2) + "&sensorL3=" + String(sensorL3, 2) +
"&sensorT4=" + String(sensorT4, 2) + "&sensorL4=" + String(sensorL4, 2));


  }
```

```
}

//ADUINO: //CONFIGURACIÓN SENSOR03 ARCHIVO: ARDUINOV3.INO
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <Wire.h>
#include <BH1750.h>
#include <DHT.h>
#define DHTPIN D3
#define DHTTYPE DHT11
BH1750 lightMeter;
DHT dht(DHTPIN, DHTTYPE);

//-------------------VARIABLES GLOBALES-------------------------
int contconexion = 0;
const char *ssid = "corbachop";
const char *password = "Cor*Ba:cHop:D";
//const char *ssid = "Iphone Mijail";
//const char *password = "123456789a";
unsigned long previousMillis = 0;
float sensorT1 = 1.4;
float sensorL1 = 14;
float sensorT2 = 50;
float sensorL2 = 35;
float sensorT3 = 0.5;
float sensorL3 = 15;
float sensorT4 = 7.5;
float sensorL4 = 4.5;
char host[48];
```

```
String strhost = "datalogger.mcorbachop.com";
String strurl = "/enviardatos.php";


//-------Función para Enviar Datos a la Base de Datos SQL--------
String enviardatos(String datos) {
  String linea = "error";
  WiFiClient client;
  strhost.toCharArray(host, 49);
  if (!client.connect(host, 80)) {
    Serial.println("Fallo de conexion");
    return linea;
  }
client.print(String("POST ") + strurl + " HTTP/1.0" + "\r\n" +
             "Host: " + strhost + "\r\n" +
             "Accept: */*" + "*\r\n" +
             "Content-Length: " + datos.length() + "\r\n" +
             "Content-Type: application/x-www-form-urlencoded" + "\r\n" +
             "\r\n" + datos);
delay(10);
Serial.print("Enviando datos a SQL...");
  unsigned long timeout = millis();
  while (client.available() == 0) {
    if (millis() - timeout > 5000) {
      Serial.println("Cliente fuera de tiempo!");
      client.stop();
      return linea;
    }
  }
  // Lee todas las lineas que recibe del servidor y las imprime por la terminal serial
```

```
  while(client.available()){
    linea = client.readStringUntil('\r');
  }
  Serial.println(linea);
  return linea;
}
void setup() {
  pinMode(D5,OUTPUT);
  pinMode(D6,OUTPUT);
  // Inicia Serial
  Serial.begin(115200);
  Serial.println("");
  Wire.begin();
  dht.begin();
  //lightMeter.begin();
  Serial.println(F("BH1750 Test"));
  // Conexión WIFI
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED and contconexion <50) {
  //Cuenta hasta 50 si no se puede conectar lo cancela
    ++contconexion;
    delay(500);
    Serial.print(".");
  }
  if (contconexion <50) {
      Serial.println("");
      Serial.println("WiFi conectado");
      Serial.println(WiFi.localIP());
  }
```

```
  else {
     Serial.println("");
     Serial.println("Error de conexion");
  }

}
//-------------------------LOOP------------------------------
// D5 ---> B
// D6 ---> A
float leerTem(char a)
{
  switch (a) {
  case 0:
        digitalWrite(D5,LOW);
        digitalWrite(D6,LOW);
    break;
  case 1:
        digitalWrite(D5,LOW);
        digitalWrite(D6,HIGH);
    break;
  case 2:
        digitalWrite(D5,HIGH);
        digitalWrite(D6,LOW);
    break;
  case 3:
        digitalWrite(D5,HIGH);
        digitalWrite(D6,HIGH);
    break;
  default:
```

```
    // statements
    break;
}
float t = dht.readTemperature();
return t;
}

float leerLux(char a)
{
  switch (a) {
  case 0:
        digitalWrite(D5,LOW);
        digitalWrite(D6,LOW);
    break;
  case 1:
        digitalWrite(D5,LOW);
        digitalWrite(D6,HIGH);
    break;
  case 2:
        digitalWrite(D5,HIGH);
        digitalWrite(D6,LOW);
    break;
  case 3:
        digitalWrite(D5,HIGH);
        digitalWrite(D6,HIGH);
    break;
  default:
    // statements
    break;
```

```
}
lightMeter.begin();
float lux = lightMeter.readLightLevel();
return lux;
}
void loop() {
  //float t = dht.readTemperature();
 // float lux = lightMeter.readLightLevel();
  //Serial.print("Light: ");


  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= 300000) { //envia la temperatura cada 10 segundos
    previousMillis = currentMillis;
    sensorT1 = leerTem(0);
    sensorL1 = leerLux(0);
    sensorT2 = leerTem(1);
    sensorL2 = leerLux(1);
    sensorT3 = leerTem(2);
    sensorL3 = leerLux(2);
    sensorT4 = leerTem(3);
    sensorL4 = leerLux(3);
    enviardatos("sensorT1=" + String(sensorT1, 2) +" &sensorL1=" + String(sensorL1, 2) + "&sensorT2=" + String(sensorT2,2 )+ "&sensorL2=" + String(sensorL2, 2) + "&sensorT3=" + String(sensorT3, 2) + "&sensorL3=" + String(sensorL3, 2) + "&sensorT4=" + String(sensorT4, 2) + "&sensorL4=" + String(sensorL4, 2));
  }
}
//ADUINO: //CONFIGURACIÓN SENSOR04 ARCHIVO: ARDUINOV4.INO
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
```

```
#include <Wire.h>
#include <BH1750.h>
#include <DHT.h>
#define DHTPIN D3
#define DHTTYPE DHT11
BH1750 lightMeter;
DHT dht(DHTPIN, DHTTYPE);

//-------------------VARIABLES GLOBALES-------------------------
int contconexion = 0;
// Seleccionamos el Wifi al que nos conectáramos
const char *ssid = "corbachop";
const char *password = "Cor*Ba:cHop:D";
//const char *ssid = "Iphone Mijail";
//const char *password = "123456789a";
unsigned long previousMillis = 0;
//crear las variables para los sensores
float sensorT1;
float sensorL1;
float sensorT2;
float sensorL2;
float sensorT3;
float sensorL3;
float sensorT4;
float sensorL4;
char host[48];
//ubicacion del hosting
String strhost = "datalogger.mcorbachop.com";
String strurl = "/enviardatos.php";
```

```
//-------Función para Enviar Datos a la Base de Datos SQL--------
String enviardatos(String datos) {
  String linea = "error";
  WiFiClient client;
  strhost.toCharArray(host, 49);
  if (!client.connect(host, 80)) {
    Serial.println("Fallo de conexion");
    return linea;
  }
client.print(String("POST ") + strurl + " HTTP/1.0" + "\r\n" +
             "Host: " + strhost + "\r\n" +
             "Accept: */*" + "*\r\n" +
             "Content-Length: " + datos.length() + "\r\n" +
             "Content-Type: application/x-www-form-urlencoded" + "\r\n" +
             "\r\n" + datos);
delay(10);
Serial.print("Enviando datos a SQL...");
  unsigned long timeout = millis();
  while (client.available() == 0) {
    if (millis() - timeout > 5000) {
      Serial.println("Cliente fuera de tiempo!");
      client.stop();
      return linea;
    }
  }
  // Lee todas las lineas que recibe del servidor y las imprime por la terminal serial
  while(client.available()){
    linea = client.readStringUntil('\r');
  }
```

```
  Serial.println(linea);
  return linea;
}
void setup() {
  pinMode(D5,OUTPUT);
  pinMode(D6,OUTPUT);
  // Inicia Serial
  Serial.begin(115200);
  Serial.println("");
  Wire.begin();
  dht.begin();
  //lightMeter.begin();
  Serial.println(F("BH1750 Test"));
  // Conexión WIFI
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED and contconexion <50) {
  //Cuenta hasta 50 si no se puede conectar lo cancela
    ++contconexion;
    delay(500);
    Serial.print(".");
  }
  if (contconexion <50) {
      Serial.println("");
      Serial.println("WiFi conectado");
      Serial.println(WiFi.localIP());
  }
  else {
      Serial.println("");
      Serial.println("Error de conexion");
```

```
  }
}
//-------------------------LOOP-------------------------------
// arduino        color                      0   1   2    3
// D5 ---> B  cable amarillo                 12  14  15  11
// D6 ---> A  abble verde                    1   5   2   4
float leerTem(char a)
{
  switch (a) {
  case 0:
        digitalWrite(D5,LOW);
        digitalWrite(D6,LOW);
    break;
  case 1:
        digitalWrite(D5,LOW);
        digitalWrite(D6,HIGH);
    break;
  case 2:
        digitalWrite(D5,HIGH);
        digitalWrite(D6,LOW);
    break;
  case 3:
        digitalWrite(D5,HIGH);
        digitalWrite(D6,HIGH);
    break;
  default:
    // statements
    break;
}
```

```
float t = dht.readTemperature();
return t;
}
float leerLux(char a)
{
 switch (a) {
 case 0:
      digitalWrite(D5,LOW);
      digitalWrite(D6,LOW);
  break;
 case 1:
      digitalWrite(D5,LOW);
      digitalWrite(D6,HIGH);
  break;
 case 2:
      digitalWrite(D5,HIGH);
      digitalWrite(D6,LOW);
  break;
 case 3:
      digitalWrite(D5,HIGH);
      digitalWrite(D6,HIGH);
  break;
 default:
  // statements
  break;
}
lightMeter.begin();
float lux = lightMeter.readLightLevel();
return lux;
```

```
}
void loop() {
// float t = dht.readTemperature();
// float lux = lightMeter.readLightLevel();
// Serial.print("Light: ");
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= 10000) { //envia la temperatura cada 300000 milisegundos
    previousMillis = currentMillis;
    sensorT1 = leerTem(0);
    sensorL1 = leerLux(0);
    sensorT2 = leerTem(1);
    sensorL2 = leerLux(1);
    sensorT3 = leerTem(2);
    sensorL3 = leerLux(2);
    sensorT4 = leerTem(3);
    sensorL4 = leerLux(3);
    enviardatos("sensorT1=" + String(sensorT1, 2) +" &sensorL1=" + String(sensorL1, 2) + "&sensorT2=" + String(sensorT2,2 )+ "&sensorL2=" + String(sensorL2, 2) + "&sensorT3=" + String(sensorT3, 2) + "&sensorL3=" + String(sensorL3, 2) + "&sensorT4=" + String(sensorT4, 2) + "&sensorL4=" + String(sensorL4, 2));
  }
}
```

//MYSQL: //CONSULTAS BDD ARCHIVO: COSULTAS.SQL

```
DROP TABLE IF EXISTS `tdata`;
CREATE TABLE IF NOT EXISTS `tdata` (
  `id_data` int(11) NOT NULL AUTO_INCREMENT,
  `time_data` timestamp NOT NULL DEFAULT current_timestamp(),
  `sensorT1` double NOT NULL,
  `sensorL1` int(11) NOT NULL,
  `sensorT2` double NOT NULL,
```

```
  `sensorL2` int(11) NOT NULL,
  `sensorT3` double NOT NULL,
  `sensorL3` int(11) NOT NULL,
  `sensorT4` double NOT NULL,
  `sensorL4` int(11) NOT NULL,
  PRIMARY KEY (`id_data`)
) ENGINE=MyISAM AUTO_INCREMENT=1 DEFAULT CHARSET=utf8;
```

tdata

```
INSERT INTO `tdata`
        (`id_data`, `time_data`, `sensorT1`, `sensorL1`, `sensorT2`, `sensorL2`,
`sensorT3`, `sensorL3`, `sensorT4`, `sensorL4`)
VALUES      (NULL, current_timestamp(), '30', '31', '32', '33', '34', '35', '36', '37');
```

//PHP: PRESENTACION ARCHIVO INDEX.PHP

```
<!doctype html>
<html>
<head><meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>DataloggerUAC</title>
<link rel="shortcut icon" href="img/logo.ico" />
<link rel="stylesheet" href="/css/estilos.css" />
</head>
<body>
        <input type="checkbox" id="cerrar">
        <label for="cerrar" id="btn-cerrar">X</label>
        <div class="modal">
        <div class="contenido">
        <h2>Cúbrete al toser, unidos por un Cusco saludable</h2>
        <img src="/img/covit2.png" width="800"
        height="480">
```

```
            </div>
            </div>
<div id="main">
  <div id="nav">
            <div id="menu1" class="menu">TEMPERATURA</div>
            <div id="menu2" class="menu">LUMINOSIDAD</div>
            <div id="menu3" class="menu">VERSUS</div>
  </div>
  <iframe id="content" src="">iFrames not supported</iframe>
</div>
<script src="http://ajax.googleapis.com/ajax/libs/jquery/2.0.0/jquery.min.js"></script>
<script>
  $(document).ready(function(e) {
            $('#menu1').on('click', function(){

                        $('#content').attr('src', '/data/temperatura.php');
            });
            $('#menu2').on('click', function(){
                        $('#content').attr('src', '/data/luminosidad.php');

            });
            $('#menu3').on('click', function(){
                        $('#content').attr('src', '/data/ok.php');

            });
  });
</script>
</body>
</html>
```

```
//CSS: PRESENTACION ESTILOS.CSS
body {
    text-align: center;
    font-family: sans-serif;
    margin: 0;
}
.menu {
            width: 20%;
            height: 25px;
            float: left;
            padding: 10px;
            text-align: center;
            background: #fff;
            color: #000;
  }
.menu:hover {
            background: #000;
            color: #fff;
  }
  #content {
            clear: both;
            background: #e5e5e5;
            padding: 0;
            overflow-y: scroll;
            width: 100%;
            height: 700px;
            border: 0;
  }
.modal {
```

```
    width: 100%;
    height: 100vh;
    background: rgba(0,0,0,0.8);

    position: absolute;
    top: 0;
    left: 0;

    display: flex;

    animation: modal 2s 3s forwards;
    visibility: hidden;
    opacity: 0;
}

.contenido {
    margin: auto;
    width: 70%;
    height: 90%;
    background: white;
    border-radius: 10px;
}

#cerrar {
    display: none;
}

#cerrar + label {
    position: fixed;
```

```
    color: #fff;
    font-size: 25px;
    z-index: 50;
    background: darkred;
    height: 40px;
    width: 40px;
    line-height: 40px;
    border-radius: 50%;
    right: 150px;
    top: 150px;
    cursor: pointer;
    animation: modal 2s 3s forwards;
    visibility: hidden;
    opacity: 0;
}

#cerrar:checked + label, #cerrar:checked ~ .modal {
    display: none;
}
@keyframes modal {
    100% {
        visibility: visible;
        opacity: 1;
    }
}
```

//PHP: CONEXIÓN BDD ARCHIVO CONEXIÓN.PHP

```
<?php
function conexioncloud(){
                $servidor="db4free.net";
```

```
                $usuario="corbachoroot";
                $password="corbahop123";
                $bd="dbdatalogger";
                $conexion=mysqli_connect($servidor,$usuario,$password,$bd);
                return $conexion;
        }
function conexioncorback(){
                $servidor="mcorbachop.com";
                $usuario="mcorbachop_corback";
                $password="corback1234";
                $bd="mcorbachop_dbdatalogger";
                $conexion=mysqli_connect($servidor,$usuario,$password,$bd);
                return $conexion;
        }
?>
```

//PHP: DATALUMIN.PHP

```
<?php
  header('Content-type:application/xls');
  header('Content-Disposition: attachment; filename=data_luminosidad.xls');
   include("conexion.php");
   $conexion=conexioncorback();
   $sql="SELECT * FROM tdata";
  $result=mysqli_query($conexion,$sql);
?>
<table border="1">
  <tr style="background-color:red;">
                <th>id</th>
                <th>timer</th>
```

```
                    <th>sensorL1</th>
                    <th>sensorL2</th>
                    <th>sensorL3</th>
                    <th>sensorL4</th>
  </tr>
  <?php
          while ($row=mysqli_fetch_assoc($result)) {
                    ?>
                        <tr>
                              <td><?php echo $row['id_data'] ?></td>
                      <td><?php echo $row['time_data'] ?></td>
                      <td><?php echo $row['sensorL1'] ?></td>
                      <td><?php echo $row['sensorL2'] ?></td>
                      <td><?php echo $row['sensorL3'] ?></td>
                      <td><?php echo $row['sensorL4'] ?></td>
                        </tr>
                    <?php
          }
  ?>
</table>
```

//PHP: DATATEMP.PHP

```
<?php
  header('Content-type:application/xls');
  header('Content-Disposition: attachment; filename=data_temperatura.xls');
   include("conexion.php");
   $conexion=conexioncorback();
   $sql="SELECT * FROM tdata";
  $result=mysqli_query($conexion,$sql);
?>
```

```
<table border="1">
  <tr style="background-color:red;">
                <th>id</th>
                <th>timer</th>
                <th>sensorT1</th>
                <th>sensorT2</th>
                <th>sensorT3</th>
                <th>sensorT4</th>
  </tr>
  <?php
        while ($row=mysqli_fetch_assoc($result)) {
                ?>
                        <tr>
                                <td><?php echo $row['id_data'] ?></td>
                        <td><?php echo $row['time_data'] ?></td>
                        <td><?php echo $row['sensorT1'] ?></td>
                        <td><?php echo $row['sensorT2'] ?></td>
                        <td><?php echo $row['sensorT3'] ?></td>
                        <td><?php echo $row['sensorT4'] ?></td>
                           </tr>
                <?php
        }
  ?>
</table>
```

//PHP: ENVIARDATOS.PHP

```
<?php
include("conexion.php");
        #$conexion=conexionlocal();
        $conexion=conexioncorback();
```

```
        #$conexion=conexioncloud();
        $T1=$_POST['sensorT1'];
        $L1=$_POST['sensorL1'];
    $T2=$_POST['sensorT2'];
        $L2=$_POST['sensorL2'];
        $T3=$_POST['sensorT3'];
        $L3=$_POST['sensorL3'];
        $T4=$_POST['sensorT4'];
        $L4=$_POST['sensorL4'];
        $sql="INSERT INTO tdata
(sensorT1,sensorL1,sensorT2,sensorL2,sensorT3,sensorL3,sensorT4,sensorL4)
                    VALUES ('$T1','$L1','$T2','$L2','$T3','$L3','$T4','$L4')";
        $result=mysqli_query($conexion,$sql);
?>
```

//PHP: LUMINOSIDAD.PHP

```
<?php
   include("conexion.php");
   $conexion=conexioncorback();
 ?>
  <script type="text/javascript">
  function actualizar(){location.reload(true);}
//Función para actualizar cada 4 segundos(4000 milisegundos)
  setInterval("actualizar()",300000);
  </script>
<!DOCTYPE html>
<html>
<head>
<title>DataLogeer</title>
</head>
```

```
<body>
                    <table border="1" >
                    <tr>
                        <td>id</td>
                        <td>timer</td>
                        <td>sensorL1</td>
                        <td>sensorL2</td>
                        <td>sensorL3</td>
                        <td>sensorL4</td>
                    </tr>

                    <?php
                    $sql="SELECT * FROM tdata ORDER BY id_data
DESC LIMIT 5";
                    $result=mysqli_query($conexion,$sql);

                    while($mostrar=mysqli_fetch_array($result)){
                     ?>

                    <tr>
                        <td><?php echo $mostrar['id_data'] ?></td>
                        <td><?php echo $mostrar['time_data'] ?></td>
                        <td><?php echo $mostrar['sensorL1'] ?></td>
                        <td><?php echo $mostrar['sensorL2'] ?></td>
                        <td><?php echo $mostrar['sensorL3'] ?></td>
                        <td><?php echo $mostrar['sensorL4'] ?></td>
                    </tr>
                    <?php
                    }
```

```
                                        ?>
                                       </table>
                                        <div class="col text-center">
          <a href="datalumi.php">
            Generar XLS
          </a>
        </div>
</form>
</body>
</html>
<?php
 function conectarBD(){
          $server = "mcorbachop.com";
          $usuario = "mcorbachop_corback";
          $pass = "corback1234";
          $BD = "mcorbachop_dbdatalogger";
          //variable que guarda la conexión de la base de datos
          $conexion = mysqli_connect($server, $usuario, $pass, $BD);
          //Comprobamos si la conexión ha tenido exito
          if(!$conexion){
            echo 'Ha sucedido un error inexperado en la conexion de la base de
datos<br>';
          }
          //devolvemos el objeto de conexión para usarlo en las consultas
          return $conexion;
    }
    /*Desconectar la conexion a la base de datos*/
    function desconectarBD($conexion){
          //Cierra la conexión y guarda el estado de la operación en una variable
```

```
        $close = mysqli_close($conexion);
        //Comprobamos si se ha cerrado la conexión correctamente
        if(!$close){
          echo 'Ha sucedido un error inexperado en la desconexion de la base de datos<br>';
        }
        //devuelve el estado del cierre de conexión
        return $close;
    }
  //Devuelve un array multidimensional con el resultado de la consulta
   function getArraySQL($sql){
      //Creamos la conexión
      $conexion = conectarBD();
      //generamos la consulta
      if(!$result = mysqli_query($conexion, $sql)) die();

      $rawdata = array();
      //guardamos en un array multidimensional todos los datos de la consulta
      $i=0;
      while($row = mysqli_fetch_array($result))
      {
        //guardamos en rawdata todos los vectores/filas que nos devuelve la consulta
        $rawdata[$i] = $row;
        $i++;
      }
      //Cerramos la base de datos
      desconectarBD($conexion);
      //devolvemos rawdata
      return $rawdata;
```

```
    }
    //Sentencia SQL
$sql = "SELECT sensorL1,sensorL2,sensorL3,sensorL4,time_data from tdata;";
//Array Multidimensional
$rawdata = getArraySQL($sql);
//Adaptar el tiempo
for($i=0;$i<count($rawdata);$i++){
    $time = $rawdata[$i]["time_data"];
    $date = new DateTime($time);
    $rawdata[$i]["time"]=$date->getTimestamp()*1000;
}
?>
<HTML>
<BODY>
<meta charset="utf-8">
<!-- Latest compiled and minified JavaScript -->
<script src="https://code.jquery.com/jquery.js"></script>
    <!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
<script src="http://code.highcharts.com/stock/highstock.js"></script>
<script src="http://code.highcharts.com/modules/exporting.js"></script>
<div id="container">
</div>
<script type='text/javascript'>
$(function () {
    $(document).ready(function() {
        Highcharts.setOptions({
            time: {
                    timezone: 'America/Peru'
                }
```

```
});

var chart;
$('#container').highcharts({
    chart: {
        type: 'spline',
        animation: Highcharts.svg, // don't animate in old IE
        marginRight: 10,
        events: {
            load: function() {


            }
        }
    },
    title: {
        text: 'Luminosidad DataLogger'
    },
    xAxis: {
        type: 'datetime',
        tickPixelInterval: 150
    },
    yAxis: {
        title: {
            text: 'Value'
        },
        plotLines: [{
            value: 0,
            width: 1,
            color: '#808080'
```

```
    }]
},
tooltip: {
    formatter: function() {
            return '<b>'+ this.series.name +'</b><br/>'+
            Highcharts.dateFormat('%Y-%m-%d %H:%M:%S', this.x) +'<br/>'+
            Highcharts.numberFormat(this.y, 2);
    }
},
legend: {
    enabled: true
},
exporting: {
    enabled: true
},
series: [{
    name: 'sensorL1',
        data: (function() {
            var data = [];
        <?php
            for($i = 0 ;$i<count($rawdata);$i++){
        ?>
            data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo
$rawdata[$i]["sensorL1"];?>]);
        <?php } ?>
    return data;
        })()
},{
    name: 'sensorL2',
```

```
                data: (function() {
                    var data = [];
                <?php
                    for($i = 0 ;$i<count($rawdata);$i++){
                ?>
                data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo $rawdata[$i]["sensorL2"];?>]);
                <?php } ?>
            return data;
                })()
        },{
            name: 'sensorL3',
                data: (function() {
                    var data = [];
                <?php
                    for($i = 0 ;$i<count($rawdata);$i++){
                ?>
                data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo $rawdata[$i]["sensorL3"];?>]);
                <?php } ?>
            return data;
                })()
        },{
            name: 'sensorL4',
                data: (function() {
                    var data = [];
                <?php
                    for($i = 0 ;$i<count($rawdata);$i++){
                ?>
```

```
                data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo
$rawdata[$i]["sensorL4"];?>]);
                <?php } ?>
              return data;
                })()
          }]
       });
   });
});
</script>
</html>
//TEMPERATURA.PHP
<?php
   include("conexion.php");
   $conexion=conexioncorback();
 ?>
 <script type="text/javascript">
  function actualizar(){location.reload(true);}
//Función para actualizar cada 5 minutos segundos(300000 milisegundos)
  setInterval("actualizar()",60000);
</script>
<!DOCTYPE html>
<html>
<head><meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<title>DataLogeer</title>
</head>
<body>
          <table border="1" >
           <tr>
```

```
            <td>id</td>
            <td>timer</td>
            <td>sensorT1</td>
            <td>sensorT2</td>
            <td>sensorT3</td>
            <td>sensorT4</td>
</tr>

<?php
$sql="SELECT * FROM tdata ORDER BY id_data DESC LIMIT 5";
$result=mysqli_query($conexion,$sql);

while($mostrar=mysqli_fetch_array($result)){
 ?>

<tr>
            <td><?php echo $mostrar['id_data'] ?></td>
            <td><?php echo $mostrar['time_data'] ?></td>
            <td><?php echo $mostrar['sensorT1'] ?></td>
            <td><?php echo $mostrar['sensorT2'] ?></td>
            <td><?php echo $mostrar['sensorT3'] ?></td>
            <td><?php echo $mostrar['sensorT4'] ?></td>
 </tr>
<?php
}
 ?>
</table>

<div class="col text-center">
```

```
        <a href="datatemp.php">
          Generar XLS
        </a>
      </div>
</form>
</body>
</html>
<?php
 function conectarBD(){
        $server = "mcorbachop.com";
        $usuario = "mcorbachop_corback";
        $pass = "corback1234";
        $BD = "mcorbachop_dbdatalogger";
        //variable que guarda la conexión de la base de datos
        $conexion = mysqli_connect($server, $usuario, $pass, $BD);
        //Comprobamos si la conexión ha tenido exito
        if(!$conexion){
          echo 'Ha sucedido un error inexperado en la conexion de la base de
datos<br>';
        }
        //devolvemos el objeto de conexión para usarlo en las consultas
        return $conexion;
   }
   /*Desconectar la conexion a la base de datos*/
   function desconectarBD($conexion){
        //Cierra la conexión y guarda el estado de la operación en una variable
        $close = mysqli_close($conexion);
        //Comprobamos si se ha cerrado la conexión correctamente
        if(!$close){
```

```
            echo 'Ha sucedido un error inexperado en la desconexion de la base de datos<br>';
        }
        //devuelve el estado del cierre de conexión
        return $close;
    }
  //Devuelve un array multidimensional con el resultado de la consulta
   function getArraySQL($sql){
      //Creamos la conexión
      $conexion = conectarBD();
      //generamos la consulta
      if(!$result = mysqli_query($conexion, $sql)) die();

      $rawdata = array();
      //guardamos en un array multidimensional todos los datos de la consulta
      $i=0;
      while($row = mysqli_fetch_array($result))
      {
         //guardamos en rawdata todos los vectores/filas que nos devuelve la consulta
         $rawdata[$i] = $row;
         $i++;
      }
      //Cerramos la base de datos
      desconectarBD($conexion);
      //devolvemos rawdata
      return $rawdata;
   }
   //Sentencia SQL
$sql = "SELECT time_data,sensorT1,sensorT2,sensorT3,sensorT4 from tdata;";
```

```
//Array Multidimensional
$rawdata = getArraySQL($sql);
//Adaptar el tiempo

for($i=0;$i<count($rawdata);$i++){
    $time = $rawdata[$i]["time_data"];
    $date = new DateTime($time);
    $rawdata[$i]["time"]=$date->getTimestamp()*1000;
}
?>
<HTML>
<BODY>

<!-- Latest compiled and minified JavaScript -->
<script src="https://code.jquery.com/jquery.js"></script>
    <!-- Importo el archivo Javascript de Highcharts directamente desde su servidor -->
<script src="http://code.highcharts.com/stock/highstock.js"></script>
<script src="http://code.highcharts.com/modules/exporting.js"></script>
<div id="container">
</div>
<script type='text/javascript'>
$(function () {
    $(document).ready(function() {
        Highcharts.setOptions({
            time: {
                    timezone: 'America/Peru'
                }
        });
```

```
var chart;
$('#container').highcharts({
    chart: {
        type: 'spline',
        animation: Highcharts.svg, // don't animate in old IE
        marginRight: 10,
        events: {
            load: function() {


            }
        }
    },
    title: {
        text: 'Temperatura DataLogger'
    },
    xAxis: {
        type: 'datetime',
        tickPixelInterval: 150
    },
    yAxis: {
        title: {
            text: 'Value'
        },
        plotLines: [{
            value: 0,
            width: 1,
            color: '#808080'
        }]
    },
```

```
tooltip: {
    formatter: function() {
        return '<b>'+ this.series.name +'</b><br/>'+
        Highcharts.dateFormat('%Y-%m-%d %H:%M:%S', this.x) +'<br/>'+
        Highcharts.numberFormat(this.y, 2);
    }
},
legend: {
    enabled: true
},
exporting: {
    enabled: true
},
series: [{
    name: 'sensorT1',
    data: (function() {
      var data = [];
       <?php
          for($i = 0 ;$i<count($rawdata);$i++){
       ?>
          data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo
$rawdata[$i]["sensorT1"];?>]);
          <?php } ?>
    return data;
    })()
},{
    name: 'sensorT2',
        data: (function() {
          var data = [];
```

```
                <?php
                    for($i = 0 ;$i<count($rawdata);$i++){
                ?>
                data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo
$rawdata[$i]["sensorT2"];?>]);
                <?php } ?>
            return data;
                })()
        },{
            name: 'sensorT3',
                data: (function() {
                    var data = [];
                <?php
                    for($i = 0 ;$i<count($rawdata);$i++){
                ?>
                data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo
$rawdata[$i]["sensorT3"];?>]);
                <?php } ?>
            return data;
                })()
        },{
            name: 'sensorT4',
                data: (function() {
                    var data = [];
                <?php
                    for($i = 0 ;$i<count($rawdata);$i++){
                ?>
                data.push([<?php echo $rawdata[$i]["time"];?>,<?php echo
$rawdata[$i]["sensorT4"];?>]);
                <?php } ?>
```

```
            return data;
                })()
            }]
        });
    });
});
</script>
</html>
```

- **ANEXO 07: Tabla de costos**

TABLA DE COSTOS				
Elaboración del Dashboard	**Materiales**	**Cantidad**	**P/u**	**Total**
	Nodemcu	1	60	60
	DHT11.PCB: Sensor humedad temperatura	4	7.5	30
	BH1750.PCB: Sensor de Luz.	4	6.5	26
	Multiplexor UX: MM74HC4053	1	2.5	2.5
	Cables de circuito FTP	10	1	10
	Rack de melamina	1	75	75
	Tornillos corrugados	50	0.1	5
	Protoboard	1	15	15
	Sub-Total			223.5
Servicios en la nube	**Materiales**	**cantidad**	**p/u**	**total**
	Hosting	1	200	200
	Dominio	1	100	100
	Sub-Total			300
TOTAL				**523.5**