# ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

## FACULTAD DE INGENIERÍA EN ELECTRICIDAD Y COMPUTACIÓN

**“**EVALUACIÓN SOBRE PANELES SOLARES E IMPLEMENTACIÓN DE PROTOTIPO CON INTERFAZ PARA MEDICIÓN DE PARÁMETROS MEDIANTE MINICOMPUTADORA RASPBERRY PI.”

**TESINA DE SEMINARIO**

Previa la obtención del Título de:

**INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES.**
**INGENIERO ELÉCTRICO ESPECIALIZACIÓN ELECTRÓNICA Y AUTOMATIZACIÓN INDUSTRIAL.**

Presentado por:

Rodrigo Raúl Lindao Suárez
Gustavo Rubén Sampedro Merchán

GUAYAQUIL – ECUADOR
2014

# AGRADECIMIENTO

A Dios por brindarme el conocimiento, la salud y la vida para alcanzar esta meta profesional.

A mis padres Félix y Sandra, quienes fueron los pilares fundamental para lograr las metas propuestas durante mi vida estudiantil. Con sus sabios consejos alcanzo uno de mis principales objetivos.

A mis compañeros quienes aportaron con su conocimiento para el desarrollo de nuestras actividades durante las extensas jornadas de estudio y al Msc. Carlos Valdivieso por su confianza y paciencia durante el desarrollo de la presente tesina.

**Rodrigo Lindao Suárez**

# AGRADECIMIENTO

Agradezco a Dios por la sabiduría y entendimiento para la realización del proyecto.

A mi profesor Msc. Carlos Valdivieso por su apoyo y conocimientos brindados para la culminación del mismo.

A mi madre Rosa Merchán y Rosa Andrade y familia por la confianza y fe. A mis amigos Cristóbal, Yuliana, Carlos, Rodrigo por ser incondicionales y apoyarme siempre. A Alexandra Caguana por ser fundamental en mi vida y creer en mí.

**Gustavo Sampedro Merchán**

# DEDICATORIA

Dedicado a todas las personas que de alguna forma brindaron su apoyo incondicional durante mi vida estudiantil, a quienes me motivaron día a día para la culminación de esta etapa.

A mi familia por su constante apoyo siempre al pendiente de cada uno de mis logros que he alcanzado y en general a todas las personas que mediante sus conocimientos e investigación desarrollan tecnología en el país.

**Rodrigo Lindao Suárez**

# DEDICATORIA

A Dios por las lecciones impartidasen la vida. A mi madre Rosa Merchán y familia por su apoyoincondicional, amor eterno y confianza en mí.

A misamigos por su gran amistad y ayuda en diferentes etapas de mi vida. A Alexandra Caguana por estar presente en mi vida y confiar en mí.

Y a todas las personas que nos apoyaron en la culminación de este proyecto.

**Gustavo Sampedro Merchán**

# TRIBUNAL DE SUSTENTACIÓN

Msc. Carlos Valdivieso A.

PROFESOR DE SEMINARIO DE GRADUACIÓN

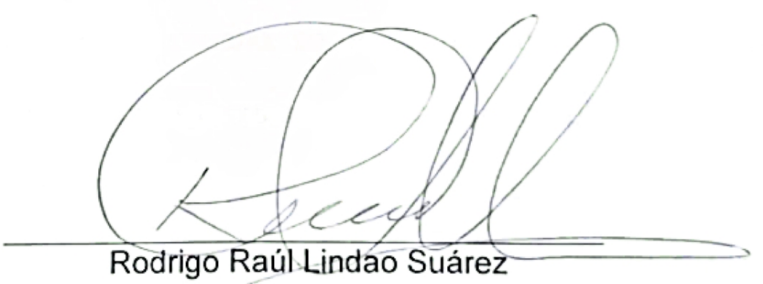
Ing. Hugo Villavicencio V.

PROFESOR DELEGADO POR LA UNIDAD ACADÉMICA.

# DECLARACIÓN EXPRESA

"La responsabilidad del contenido de esta Tesina de Grado, nos corresponde exclusivamente; y el patrimonio intelectual del mismo a la ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL"

(Reglamento de exámenes y títulos profesionales de la ESPOL)



Rodrigo Raúl Lindao Suárez

Gustavo Rubén Sampedro Merchán

# RESUMEN

La búsqueda fundamental de una fuente de energía renovable, promueve el estudio de nuevas tecnologías como la utilización de la luz solar por captación fotónica a través de paneles solares. De allí surge la necesidad de realizar la evaluación de estos instrumentos, adquiriendo los diferentes parámetros como voltaje, corriente y potencia con lo que podremos determinar cuanta energía se podría captar por día.

Nuestro sistema de adquisición de datos se desarrolla en la minicomputadora Raspberry PI cuyo sistema operativo es Raspbian basado en la plataforma LINUX, un software libre por lo que se encuentra al alcance de cualquier usuario, se puede realizar implementaciones en cualquier lenguaje de programación compatible con el sistema operativo, en nuestro caso utilizaremos el lenguaje C que nos proporcionara las principales sentencias útiles para realizar la captura de datos con el puerto GPIO esto se logra gracias a la implementación de la librería WIRINGPI que contiene la capacidad de administrar los pines de propósito general útiles durante el desarrollo de la presente tesina, para la facilitar la adopción de nuestro sistema se crea un archivo capaz de guardar cada una de las muestras tomadas durante un período de tiempo determinado por el usuario.

Los principales parámetros a obtener se fundamentan en el modelo de Duffie&Beckman utilizada para el obtener el valor de corriente generada por el panel solar en base a la corriente de cortocircuito, voltaje de circuito abierto y voltaje de entrada en el panel, se realiza la modelación de las curvas características Corriente vs Voltaje y Potencia vs Voltaje estas demuestran el comportamiento de nuestro panel, se realizan mediciones de potencia durante un período de tiempo como se muestra en los capítulos posteriores.

# ÍNDICE GENERAL

AGRADECIMIENTO ..... II

DEDICATORIA ..... IV

TRIBUNAL DE SUSTENTACIÓN ..... VI

DECLARACIÓN EXPRESA ..... VII

RESUMEN ..... VIII

ÍNDICE GENERAL ..... X

ABREVIATURAS ..... XIV

ÍNDICE DE FIGURAS ..... XV

ÍNDICE DE TABLAS ..... XVII

INTRODUCCIÓN ..... XVIII

CAPÍTULO 1

1. GENERALIDADES ..... 1

1.1 ANTECEDENTES ..... 1

1.2 PLANTEAMIENTO DEL PROBLEMA ..... 3

1.3 OBJETIVO GENERAL ..... 5

1.4 DESCRIPCIÓN DEL PROYECTO ..... 5

1.5 APLICACIONES Y VENTAJAS ..... 6

1.6 ALCANCE Y LIMITACIONES ..... 9

CAPÍTULO 2

2. FUNDAMENTO TEÓRICO ..... 11

2.1 RASPBERRY PI ..... 11

2.2 CARACTERÍSTICAS DEL RASPBERRY PI ..... 12

2.3 SISTEMAS DE ADMINISTRACIÓN LINUX ..... 14

2.3.1 ARQUITECTURA DE LINUX ..... 15

2.3.2 PUNTO DE MONTAJE ..... 15

2.3.3 COMPONENTES DE LINUX ..... 16

2.4 RASPBIAN EN RASPBERRY PI ..... 16

2.5 LENGUAJE C DE PROGRAMACIÓN ..... 17

2.5.1 GNU NANO ..... 18

2.5.2 COMPILANDO PROGRAMAS EN LENGUAJE C ..... 18

2.5.3 ARCHIVOS EN LENGUAJE C ..... 19

2.5.4 ESTRUCTURA EN LENGUAJE C ..... 20

2.6 GPIO PORT ..... 21

2.6.1 GPIO PINS ..... 22

2.6.2 UART SERIAL BUS ..... 23

2.6.3 I2C BUS ..... 23

2.6.4 SPI BUS ..... 24

2.6.5 GPIO PORT CONTROLADOS POR LENGUAJE C ..... 24

2.7 LIBRERÍA WIRINGPI ..... 24

2.7.1 INSTALACIÓN DE LIBRERÍA WIRINGPI ..... 25

2.8 FUENTE DE ENERGÍAS RENOVABLES ..... 26

2.9 ENERGÍA SOLAR ..... 27

2.10 HISTORIA DE LA ENERGÍA SOLAR FOTOVOLTAICA ..... 27

2.11 LA CÉLULA FOTOVOLTAICA ..... 28

2.12 FUNCIONAMIENTO DE UNA CÉLULA FOTOVOLTAICA ..... 30

2.13 MODELO DE DUFFIE & BECKMAN ..... 31
2.14 PARÁMETROS FUNDAMENTALES DE UNA CÉLULA FOTOVOLTAICA ..... 33

CAPÍTULO 3

3. DESARROLLO Y DISEÑO DEL PROYECTO ..... 37
3.1 INTRODUCCIÓN ..... 37
3.2 IMPLEMENTACIÓN DE LA INTERFAZ PARA MEDICIÓN DE PARÁMETROS MEDIANTE EL RASPBERRY PI ..... 38
3.2.1 DIAGRAMA DE BLOQUES DEL PROYECTO ..... 38
3.2.2 DIAGRAMA DE FLUJO PARA RUTINAS EN EL PIC 16F887 ..... 41
3.2.3 CÓDIGO FUENTE PARA RUTINAS EN EL PIC16F887 ..... 43
3.3 DIAGRAMA DE FLUJO PARA RUTINAS EN RASPBERRY PI ..... 44
3.4 CÓDIGO FUENTE PARA RASPBERRY PI PROGRAMA PRINCIPAL ..... 49
3.5 PRÁCTICA I : ROTACIÓN DE LEDS ..... 55
3.6 INTRODUCCIÓN ..... 55
3.7 DIAGRAMA DE BLOQUES PRÁCTICA I ..... 56
3.8 DIAGRAMA DE FLUJO DEL PRÁCTICA I ..... 56
3.9 CÓDIGO FUENTE DEL PROYECTO I : ROTACIÓN DE LEDS ..... 57
3.10 PRÁCTICA II: USO DEL PWM ..... 60
3.11 INTRODUCCIÓN ..... 60

3.12 DIAGRAMA DE BLOQUES PRÁCTICA II ..... 60

3.13 DIAGRAMA DE FLUJO DE PRÁCTICA II ..... 61

3.14 CÓDIGO FUENTE DE PRÁCTICA II: USO DEL PWM ..... 62

CAPÍTULO 4

4. PRUEBAS Y SIMULACIÓN DEL PROYECTO ..... 64

4.1 PROYECTO: PRUEBA DE LA INTERFAZ PARA MEDICIÓN DE PARÁMETROS DEL PANEL SOLAR CON EL RASPBERRY PI ..... 64

4.1.1 SIMULACIÓN DEL PROYECTO: FUNCIONAMIENTO PIC16F887 ..... 65

4.1.2 SIMULACIÓN DEL PROYECTO CON EL RASPBERRY PI . 66

4.2 SIMULACIÓN DE LA PRÁCTICA I: ROTACIÓN DE LEDS ..... 67

4.3 SIMULACIÓN DE LA PRÁCTICA II: USO DEL PWM ..... 68

4.4 PRUEBA EXPERIMENTAL PARA EL MODELO DE DUFFIE & BECKMAN PARA NUESTRO SISTEMA ..... 70

4.5 CURVAS CARACTERÍSTICAS DEL PANEL SOLAR ..... 73

4.6 MEDICIÓN DE PARÁMETROS DE UN PANEL SOLAR ..... 74

4.7 GRÁFICO RESULTANTE DE LA ADQUISICIÓN DE DATOS ..... 78

CONCLUSIONES ..... 80

RECOMENDACIONES ..... 83

BIBLIOGRAFÍA ..... 85

# ABREVIATURA

BGA: Ballgridarray.

GPIO: General de Entrada / Salida de uso.

GPU: Unidad de Procesamiento de Gráficos.

HDMI: High Definition Multimedia Interface.

Id: Corriente de diodo.

Isc: Corriente de corto circuito.

LAN: Red de área de local

MOSI: Salida de Datos del Master y Entrada de Datos al Slave.

MISO: Salida de Datos del Slave y Entrada de Datos al Master.

PXE: Entorno de Ejecución dePre-Arranque.

PoE: Powerover Ethernet.

PWM: Modulación de Ancho de Pulso.

RAM: Memoria de Acceso Rápido.

SPI: Interface Periféricos Serial.

SFV: Sistema FotoVoltaico.

SoC: Sistema en Chip.

USB: Bus Serie Universal

Voc: Tensión de Circuito Abierto

Wmp: Potencia pico.

W/m2: Watt por metro cuadrado

# ÍNDICE DE FIGURAS

Figura 1.1 Prueba de circuito abierto ..... 6

Figura 1.2 Prueba de cortocircuito ..... 6

Figura 2.1 Imagen Raspberry PI ..... 11

Figura 2.2 Distribución de interfaces de I/O del Raspberry Pi ..... 13

Figura 2.3 Archivos de LINUX ..... 15

Figura 2.4 Interfaz Gráfica del Raspberry PI ..... 17

Figura 2.5 Editor de texto GNU nano ..... 18

Figura 2.6 Prueba Programa Parpadeo de LED ..... 19

Figura2.7 Archivo utilizado para almacenar información ..... 20

Figura 2.8 Prototipo Raspberry Pi con LCD 16x2 ..... 21

Figura 2.9 Pines del puerto GPIO ..... 23

Figura 2.10 Creación de Programa en C y Librería wiringPi ..... 25

Figura 2.11 Modulo Solar para Arduino ..... 29

Figura 2.12 Elementos de una celda solar de Silicio ..... 30

Figura 2.13 Circuito equivalente de la célula solar ..... 31

Figura 2.14 Características de I-V de Células Solares ..... 34

Figura 3.1 Diagrama de bloque de nuestro proyecto ..... 38

Figura 3.2 Esquemático realizado en Fritzing del prototipo usado en nuestro proyecto ..... 40

Figura 3.3 Esquemático realizado en Fritzing uso de los pines del GPIO ..... 55

Figura 3.4 Diagrama bloque práctica #1 ..... 56

Figura 3.5 Esquemático realizado en Fritzing uso del PWM ..... 60

Figura 3.6 Diagrama bloque práctica #2 ..... 61

Figura 4.1 Implementación del panel solar con el Raspberry PI ..... 64

Figura 4.2 Simulación en Programa Proteus Convertidor ADC de 8 bits ..... 65

Figura 4.3 Programa ejecutado desde el Raspberry Pi ..... 66

Figura4.4 Programa ejecutado desde el Raspberry Pi Led encenderse ..... 67

Figura 4.5 Implementación práctica I se muestra durante la ejecución del programa ..... 68

Figura 4.6 Programa ejecutado desde el Raspberry Pi muestra el incremento de la intensidad ..... 68

Figura 4.7 Implementación Práctica II se muestra durante el incremento en la intensidad ..... 69

Figura 4.8 Programa ejecutado desde el Raspberry Pi muestra el decremento de la intensidad ..... 69

Figura 4.9 Simulación física muestra el decremento de la intensidad ..... 70

Figura 4.10 Curva características I-V resultante de los datos obtenidos ..... 73

Figura 4.11 Gráfica de Potencia vs Voltaje resultante de los datos obtenidos . 74

Figura 4.12 Gráfica de Voltaje Vs tiempo de los datos obtenidos ..... 78

Figura 4.13 Gráfica de Corriente vs tiempo de los datos obtenidos ..... 79

Figura 4.14 Gráfica de Potencia vs tiempo de los datos obtenidos ..... 79

# ÍNDICE DE TABLAS

Tabla 4.1 Datos de Voltaje, Corriente y Potencia del Panel Solar ..... 72

Tabla 4.2 Datos de Prueba del Panel Solar ..... 77

# INTRODUCCIÓN

Nuestro propósito es realizar la implementación de un sistema que nos permita obtener los principales parámetros que nos ayuden en la evaluación de paneles solares por medio de nuestra minicomputadora basada en el sistema operativo LINUX.

En el capítulo 1 revisaremos los antecedentes de los paneles solares, analizaremos el planteamiento del problema, cuáles serán los objetivos que deseamos obtener durante el desarrollo del mismo, y además; cuales serán nuestros alcances y limitaciones y en que aplicaciones se puede utilizar los paneles solares.

En el capítulo 2 revisaremos las características esencialmente enfocada a la librería WiringPi del Raspberry PI, las diferentes características del sistema operativos LINUX, las especificaciones de las diferentes sentencias de lenguaje C usadas dentro de nuestra implementación. Reseña historia del desarrollo de energía solar fotovoltaica, como ocurre el proceso dentro de la célula fotovoltaica para captar la energía solar. Analizaremos el modelo de Duffie&Beckman usada para calcular la corriente generada por el panel solar en base a los datos de voltajes obtenidos en las muestras.

En el capítulo 3 encontraremos el diseño, el diagrama de flujo como paso fundamental para la creación del programa, que nos ayudara a ser capaces de adquirir los datos en nuestra mini computadora Raspberry PI, además en este capítulo encontraremos diagrama de flujos y pequeños programas de prácticas que fueron fundamentales para la realización del proyecto final.

En el capítulo 4 mostraremos la implementación física del prototipo usado en la adquisición de datos y en cada práctica, se realiza prueba experimental del modelo de Duffie&Beckman mediante las cualesrealizamos la modelación de las curvas características del panel analizado, obtenemos muestras voltaje y corriente en un día para determinar cual es el valor máximo de potencia que logra captar el sistema.

# CAPÍTULO 1

## 1. GENERALIDADES

### 1.1 ANTECEDENTES

La necesidad de emplear nuevos métodos de generación de energía que sean de pequeño o nulo impacto ecológico ha dado como resultado el estudio de las energías renovables, una de ellas es el aprovechamiento de la energía solar que posee una elevada calidad energética y de insignificante impacto al ambiente además de ser inagotable a escala humana; sin embargo existen algunos problemas a la hora de dicho aprovechamiento, la energía llega a la tierra de manera dispersa y semialeatoria, estando sometida a ciclos día-noche y estaciones invierno-verano.

Debido a la enorme distancia que existe entre el sol y la tierra la Irradiación que llega a nuestro planeta es solo una pequeña parte de la potencia total emitida por el sol. En concreto, al planeta tierra llegan como valor medio 1367 (W/m2)[11]. Este valor de potencia radiante no es el

valor que llega a la superficie de la tierra, pues la atmósfera atenúa a la radiación solar a través de fenómenos de reflexión, absorción y difusión ocasionada por las moléculas de aire, ozono vapor de agua y otros gases presentes en ella, por lo tanto la irradiación en un plano horizontal ubicado sobre la superficie terrestre alcanza un valor máximo de 1000 (W/m2) durante el mediodía en un día claro, es decir el valor máximo a aprovechar [11].

La energía solar se la puede aprovechar de dos maneras: por captación térmica y captación fotónica, por captación térmica es cuando la energía solar es interceptada por una superficie absorbente y se degrada apareciendo un efecto térmico.

Por captación fotónica la radiación solar es empleada de forma directa y esto origina energía fotovoltaica. Mediante el efecto fotoeléctrico la energía de los corpúsculos constituyentes de la luz (fotones) se puede aprovechar para producir electricidad lo cual se hace con la ayuda de los llamados SFV (Sistemas Fotovoltaicos) que constan de un generador (paneles fotovoltaicos, responsables de la producción de electricidad), un acumulador (para poder almacenar energía sobrante y utilizarla cuando sea necesaria), y un regulador (elemento de control entre los equipos que consumen la energía y las partes del SFV).

Generalmente, una célula fotovoltaica tiene un grosor que varía entre los 0,25 [mm] y los 0,35 [mm] y una forma generalmente cuadrada, con una superficie aproximadamente igual a 100 [mm2][13].

Los materiales para la fabricación de los módulos fotovoltaicos solares son:

-Silicio Monocristalino: de rendimiento energético hasta 15 - 17%.

-Silicio Poli-cristalino: de rendimiento energético hasta 12 - 14 %.

-Silicio Amorfo: con rendimiento energético menor del 10 %.

-Otros materiales: Arseniuro de galio, diseleniuro de indio y cobre, telurio de cadmio.

Actualmente, el material más utilizado es el silicio monocristalino que tiene prestaciones y duración en el tiempo superiores a cualquier otro material utilizado para el mismo fin.

## 1.2 PLANTEAMIENTO DEL PROBLEMA

En nuestro país la mayor parte de la energía eléctrica es generada a través de centrales hidroeléctricas, térmicas, pero en la actualidad buscamos el uso de energías alternativas que sean menos contaminantes, más económicas para el beneficio de la población y del medio ambiente.

Las energías renovables son recursos domésticos y contribuyen a proporcionar una completa seguridad de su suministro, este tipo de fuentes de energía son virtualmente un recurso interrumpible, además de reducir el costo ambiental  a través de la generación de energía eléctrica por medio de este tipo de fuentes, también su uso es consistente con las nuevas políticas de protección del medio ambiente.

Las opciones para reducir la dependencia de las centrales hidroeléctricas y térmicas como principal fuente de generación de energía, son las llamadas energías renovables como la energía solar y energía eólica las cuales se desean implementar en el país.

Por lo cual se hace imperioso hacer un análisis sobre este tipo de energías, ¿cuál es su forma de trabajo?, ¿cuáles son los parámetros a medir?, ¿cuáles son los beneficios que ofrece, sus ventajas y sus limitaciones?

Para realizar una evaluación efectiva sobre los paneles solares obtendremos parámetros de corriente, voltaje, potencia; de los cuales graficaremos los resultados y haremos el análisis debido.

## 1.3 OBJETIVO GENERAL

Aprender el uso de técnicas utilizadas en diseño con minicomputadoras en aplicaciones de energía solar.

## 1.4 DESCRIPCIÓN DEL PROYECTO

Evaluaremos y analizaremos un panel solar, los datos de las mediciones obtenidas las mostraremos a través de la mini computadora RASPBERRY PI.

La programación de la mini computadora RASPBERRY PI se la realizará en Linux, desarrollamos un programa el cual muestra los valores de voltaje, corriente y potencia, parámetros fundamentales para evaluar un panel.

El puerto GPIO de la mini computadora nos ayudará en la adquisición de datos de voltaje, pero antes convertiremos los valores analógicos de voltaje del panel en valores digitales para poder utilizar este puerto; la conversión de dichos valores la realizamos mediante un PIC 16F887.

Utilizaremos la ecuación característica de un panel solar (ecuación de Duffie&Beckman) para la obtención de la corriente a través del voltaje, esta fórmula utiliza la corriente de cortocircuito y el voltaje de circuito abierto la cual debemos obtener.

Para la prueba de voltaje de circuito abierto colocaremos las puntas de un multímetro en los bornes del panel sin carga.

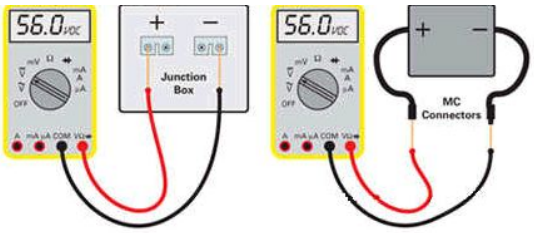



**Fig. 1.1 Prueba de circuito abierto**

Para la prueba de cortocircuito pondremos los terminales del panel en cortocircuito y mediremos la corriente que necesitamos.

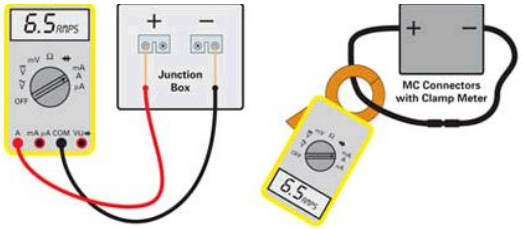



**Fig. 1.2 Prueba de cortocircuito**

## 1.5 APLICACIONES Y VENTAJAS

Las aplicaciones de los paneles solares y su utilización tienen gran importancia en el desarrollo del país, ya que permiten realizar instalaciones que alimenten sistemas que estén lejos de la red de distribución [12].

Existe una gran diversidad de aplicaciones, de las cuales podemos dividirlas en dos grupos:

- Sistemas aislados.
- Conexión a red.

Los sistemas aislados tienen como objetivo satisfacer total o parcialmente la demanda de energía eléctrica de los lugares donde la red de distribución no existe o es de difícil acceso.

Estos sistemas se lo emplea en la electrificación rural, telecomunicaciones, aplicaciones agrícolas, ganadera, iluminación pública, señalización y desarrollo rural.

Nombraremos algunas de las aplicaciones en las áreas antes mencionadas:

- Telefonía móvil.
- Repetidores de radio y televisión.
- Postes S.O.S. de carreteras.
- Telemando
- Telemetría
- Radares

- Radiotelefonía en general y para militares o puestos de vigilancia forestal.
- Telefonía rural vía satélite
- Cabinas telefónicas de uso público.
- Centrales de conmutación.
- Viviendas de uso temporal.
- Viviendas de uso permanente.
- Electrificación de refugios y albergues de montaña.
- Postas sanitarias. (iluminación, conservación de medicamentos y vacunas con frigoríficos)
- Bombeos de agua.
- Controles de riego.
- Invernaderos (automatización de ventanas e iluminación).
- Bombeos de agua para proporcionar agua al ganado.
- Electrificación de granjas. (iluminación, motores, esquiladoras)
- Electrificación de cercas.
- Carteles publicitarios.
- Farolas de alumbrado público.
- Paradas de autobuses.
- Iluminación de túneles, cuevas, etc.
- Faros y boyas de uso marítimo.
- Radiofaros y radiobalizas de uso aéreo.

- Señalización vial para señalización de curvas, obstáculos, rotondas, etc. en ciudades y carreteras mediante led's.
- Indicadores de hora y temperatura en vías públicas.

Los sistemas de conexión a red no tienen sistemas de acumulación, debido a que la energía producida por los paneles solares es canalizada hacia la red eléctrica y cuentan con sistemas de seguimiento del estado de la tensión de la red garantizando el correcto funcionamiento tanto en modo, como en tiempo de entrega de energía, evitando así situaciones peligrosas.

Por otra parte, se eliminan las baterías o acumuladores que son la parte más costosa y compleja de una instalación (ciclos de carga, vida útil, mantenimiento, etc.).

## 1.6 ALCANCE Y LIMITACIONES

Podremos analizar los parámetros de los paneles solares y dar una opinión si es conveniente usarlos en las aplicaciones que se pueden dar a este tipo de energías.

Una de las limitaciones que posee la Raspberry Pi es la memoria, si usáramos un software más robusto tendríamos problemas con el rendimiento del procesador.

Por los costos que tienen para elaborar un sistema de energía solar, utilizaremos un panel solar didáctico al cual solo se presentara a nivel de medición con el fin de analizar los resultados para una posterior comparación.

En una vivienda donde el consumo diario de energía eléctrica genere una gran demanda, el banco acumulador no podría abastecer dicha demanda, debido a que el consumo es mayor al generado y si es por las noches se debería esperar hasta que salga el sol para recargar de nuevo el banco.

Si deseamos generar mayor potencia con los paneles solares, necesitaremos mayor espacio para su instalación, puesto a mayor cantidad de células solares obtendremos mayor cantidad de energía.

# CAPÍTULO 2

## 2. FUNDAMENTO TEÓRICO

### 2.1 RASPBERRY PI

El Raspberry Pi es un ordenador que trabaja con sistema operativo Linux, es del tamaño de una tarjeta de crédito que se conecta a un monitor o un TV, y a un teclado o mouse.

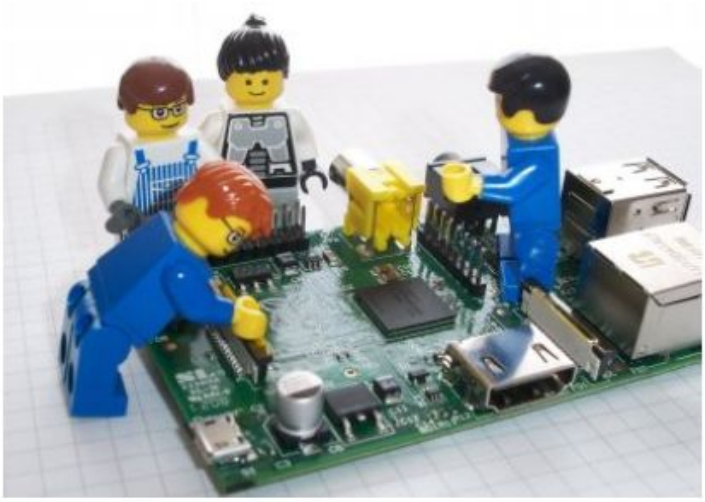


**Fig. 2.1 Imagen Raspberry PI**

**Fuente: http://trilug.org/~porter/meetings/2013-01-10_RaspberryPi.pdf**
**An Introduction to Raspberry Pi Pag. 3**

Es un PC en miniatura con procesador ARM, que se puede usar como cualquier otro PC de escritorio para ejecutar hojas de cálculo, documentos de Word, juegos, hasta reproduce vídeos en alta definición.

## 2.2 CARACTERÍSTICAS DEL RASPBERRY PI

RASPBERRY PI contiene un procesador y chip de gráficos, memoria RAM y diversas interfaces y conectores para dispositivos externos. Nuestra mini computadora opera en la misma forma que un PC estándar, que requiere un teclado, una unidad de visualización y una fuente de alimentación.

También requiere una unidad de almacenamiento masivo, pero una unidad de disco duro del tipo que se encuentra en un PC no es acorde con el tamaño de nuestra RASPBERRY PI que es de 85,60 mm x 56 mm x 21 mm. En su lugar vamos a utilizar una tarjeta de memoria flash SD utilizada normalmente en las cámaras digitales. La minicomputadora realiza el 'boot' (cargar el sistema operativo en la RAM) de esta tarjeta en la misma forma que una PC arranca en Windows desde el disco duro.

RASPBERRY PI tiene un SoC que es un Broadcom BCM2835. Este contiene un ARM1176JZFS, con punto flotante, funcionando a 700Mhz y un VideoCore 4 GPU. La GPU es capaz de reproducir la calidad BluRay,

utilizando H.264 en 40MBits / s. Tiene un núcleo 3D rápido acceso mediante el OpenGL ES2.0 y bibliotecas OpenVG suministrado.

Periféricos:

- Puertos RCA este nos permite usar nuestro TV como monitor
- 2 puertos USB para conectar los dispositivos de entrada y salida como Teclado y Mouse además se podría conectar un HUB USB para lograr más entradas USB según la necesidad del usuario.

- Posee puerto HDMI este nos permitiría observar en alta definición las actividades realizadas en televisores y monitores que soporten esta tecnología.

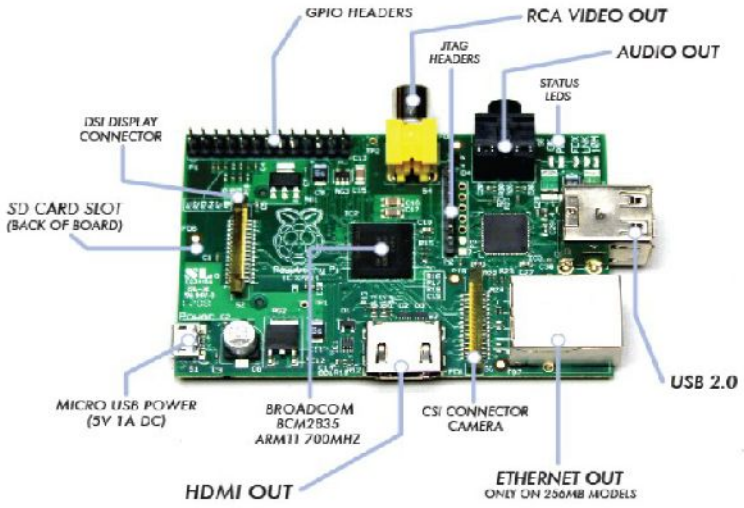



**Fig. 2.2 Distribución de interfaces de I/O del Raspberry Pi**
**Fuente: http://trilug.org/~porter/meetings/2013-01-10_RaspberryPi.pdf**
**An Introduction to Raspberry Pi Pag. 9**

Interfaces de comunicación:

- Puerto Ethernet permite establecer conexión con otros equipos que pertenezcan a la red LAN donde se encuentra conectado nuestro equipo mediante la cual pueda tener acceso a la WEB para hacer descargas de paquetes que serán de mucha ayuda para programas basados en LINUX.

  Otra opción es conectar un adaptador inalámbrico USB el cual se establece conexión con la red mediante un Access points.

- GPIO PINS (Pins de Propósito generales Entradas y Salidas) su uso principal es establecer comunicación con otras tarjetas electrónicas que nos permitan realizar ciertas aplicaciones que se requieran.

## 2.3SISTEMAS DE ADMINISTRACIÓN LINUX

LINUX es un software libre en el cual los usuarios pueden ejecutar, distribuir, estudiar, cambiar y mejorar el software. La herramienta fundamental para el sistema: el compilador para el lenguaje C (gcc, de GNU C Compiler) desarrollado por Richard Stallman, siendo este compilador usado por LinusTorvalds para desarrollar el famoso Kernel Linux.[2]

### 1.3.1 ARQUITECTURA DE LINUX

LINUX es un núcleo monolítico híbrido que engloba todos servicios del sistema, cualquier cambio a realizar en cualquier servicio requiere volver a compilar el núcleo para aplicar los nuevos cambios. Los controladores de dispositivos y las extensiones al sistema operativo se pueden cargar y descargar fácilmente como módulos, mientras el sistema continúa funcionando sin interrupciones.[2]

### 1.3.2 PUNTO DE MONTAJE

En Linux los archivos están ordenados en un gran árbol de jerarquía el cual posee como raíz el símbolo "/"

| Descripción | P. de montaje |
|---|---|
| Directorio del ROOT (Raiz) | / |
| Archivos Binarios | /bin |
| Directorio de arranque | /boot |
| Otras particiones | /data |
| Archivos de unidades | /dev |
| Archivos de configuración | /etc |
| Directorios de usuarios | /home |
| Librerias | /lib |
| Unid. removibles de montaje | /media |
| Directorios de aplicaciones | /opt |
| Directorio de administración | /root |
| Sistemas binarios | /sbin |
| Directorios de servicios | /srv |
| Area de archivos temporales | /temp |
| Recursos del sistema UNIX | /usr |
| Archivos de variables | /var |
| Archivos de procesos | /proc |
| Directorio de Información de Sist. | /sys |

**Fig.2.3 Archivos de LINUX**

**Fuente: Curso Introducción a LINUX dictado por NET service**

### 1.3.3 COMPONENTES DE LINUX

Se compone básicamente de 4 [2]:

El núcleo: es el programa principal lo usamos para ejecutar programas y gestionar dispositivos de hardware tales como los discos y las impresoras.

El shell: es la interfaz con el usuario. Recibe órdenes del usuario y las envía al núcleo para ser ejecutadas.

El sistema de archivos: se encarga de administrar la forma en que se almacenan los archivos en dispositivos de almacenamiento. Los archivos están organizados en directorios. Cada directorio puede contener un número cualquiera de subdirectorios, cada uno de los cuales puede a su vez, contener otros archivos.

Las utilidades: son programas especializados, tales como editores, compiladores y programas de comunicaciones, que realizan operaciones de computación estándar. Incluso uno mismo puede crear sus propias utilidades.

## 1.4 RASPBIAN EN RASPBERRY PI

El Raspberry PI posee un sistema operativo LINUX llamado Raspbian teniendo un ambiente de escritorio conocido como Lightweight X11 Desktop Enviroment (LXDE).

**Fig. 2.4 Interfaz Gráfica del Raspberry PI**

**Fuente: Realizado por Rodrigo Lindao Suárez**

Este escritorio muestra algunos iconos Debian Reference detalla la distribución del Linux Debian, Leafpad es un editor de texto que sirve para hacer notas rápida o escribir programas simples, LxTerminal permite usar líneas de comando en Linux sin dejar la interface gráfica, root terminal permite ingresar como un súper usuario principalmente para realizar mantenimiento de sistema.

## 1.5 LENGUAJE CDE PROGRAMACIÓN

Lenguaje C es un lenguaje de programación creado por Dennis Ritchie muy popular fue base para desarrollar muchos programas actuales siendo aún utilizado en nuestro caso para controlar el Puerto GPIO del Raspberry Pi y PIC16F887 incluso el mismo código es transferible entre Linux, Windows y las maquinas Mac OS. [3]

### 1.5.1 GNU NANO

Es nuestro editor de texto nos permitirá interactuar entre lenguaje máquina y el programador, lugar donde modificaremos las diferentes sentencias y librerías que utilizaremos en el desarrollo del proyecto.

**Fig. 2.5 Editor de texto GNU nano 2.2.6**

**Fuente: Realizado por Rodrigo Lindao Suárez**

### 1.5.2 COMPILANDO PROGRAMAS EN LENGUAJE C

Cuando ya se encuentra escrito y guardado como un archivo .cpp el programa a probar abrimos una ventana terminal y escribimos el siguiente comando para compilar [3]:

# **g++ [nombre del archivo].cpp –o [nombre que le gustaría para el programa]**

Ahora simplemente para correr el programa usamos:

# **sudo ./[ nombre que le gustaría para el programa]**

**Fig. 2.6 Prueba Programa Parpadeo de LED**

**Fuente: Realizado por Rodrigo Lindao Suarez**

### 1.5.3 ARCHIVOS EN LENGUAJE C

Para realizar nuestro sistema de adquisición de datos se necesita un lugar donde almacene los datos tomados como parámetros de tiempos y los necesarios para realizar un análisis según la necesidad del usuario. Como nuestra programación se basa únicamente en lenguaje C utilizaremos las funciones necesarias para crear el archivo que contendrá la información:

**FILE *fopen(char nombre[], char modo[] );**Mediante la apertura asociamos el fichero con una variable de tipo FILE*.

**intfprintf( FILE *f, chartexto_de_formato[], ... );** Nos permite acceder al archivo con el objetivo de insertar archivos de texto que guardaremos dentro de este.

**intfclose( FILE *fich );** Nos permite cerrar el fichero del puntero FILE*.

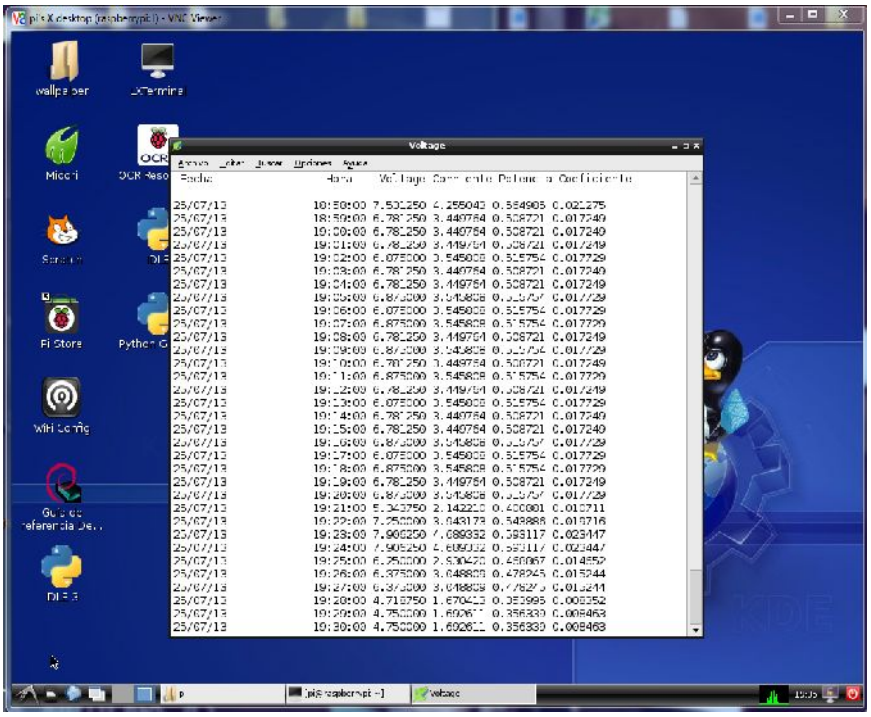



**Fig. 2.7 Archivo utilizado para almacenar información**

**Fuente: Realizado por Rodrigo Lindao Suarez**

### 1.5.4 ESTRUCTURA EN LENGUAJE C

Para obtener la fecha y hora actual de acuerdo con el tiempo local lo realizaremos por la función time esta recibirá como parámetro un puntero nulo (es decir, NULL o 0) y devolverá el tiempo actual en una variable de tipo "time_t". A continuación averiguaremos el tiempo local con la función localtimeque recibe por parámetro un puntero a una variable time_t. Esta función devuelve un puntero hacia una estructura de tipo tm, la cual utilizaremos en strftime.

Por último, la función strftime formatea una estructura de tipo tm a un formato legible por cualquier persona. El primer parámetro es una cadena,

el segundo la longitud de esta cadena, el tercero la cadena de formateo y por último la estructura que contiene el tiempo que queremos formatear.

```
inthora(char output[])
{
time_ttiempo = time(0);
struct tm *tlocal =localtime(&tiempo);
strftime(output,128,"%d/%m/%y          %H:%M:%S",tlocal);
return 0;
}
```

## 1.6 GPIO PORT

El puerto GPIO del Raspberry PI está compuesto por 26 pines, los cuales nos permite establecer conexión con hardware electrónico externo mediante el cual obtendremos muestras que nos interesan en los diferentes prototipos que analizaremos.

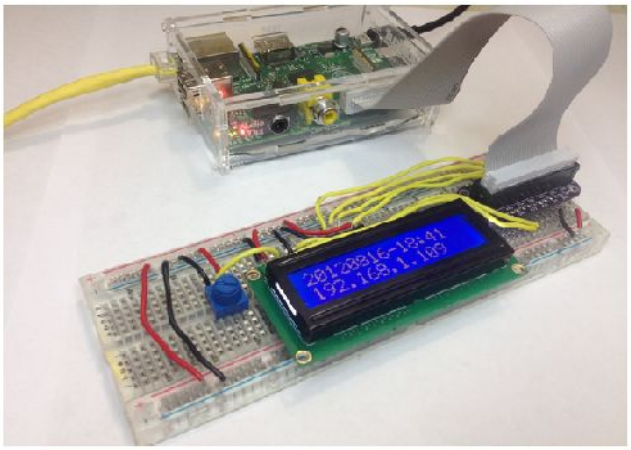


**Fig. 2.8 Prototipo Raspberry Pi con LCD 16x2 http://learn.adafruit.com/downloads/pdf/drive-a-16x2-lcd-directly-with-a-raspberry pi.pdf.Tutorial creado por MikeySklarPag. 3**

El GPIO es muy útil para controlar LEDS, motores y entre otras aplicaciones según la necesidad del usuario.[5]

Las entradas son señales enviadas hacia el dispositivo para su posterior interpretación, en cambio las salidas son señales que son producidas y enviadas hacia fuera del computador. [5]

Las entradas del Raspberry PI solo entiende señales digitales, las entradas analógicas tienen que ser procesadas antes de ingresar a nuestro dispositivo por medio de un conversor analógico digital. [5]
Los datos digitales pueden salir por medio de 2 vías seriales o paralelos, si la comunicación es paralelo se transmiten múltiple bits en cambio sí es serial puede ser transmitida por 2 bits uno se utiliza para transmisión (Tx) y el otro se utiliza para recepción (Rx). [5]

### 1.6.1 GPIO PINS

Los pins GPIO pueden ser usados como pins de propósito generales Entradas y Salidas el voltaje entregado por los pines oscila entre 0V y 3.3V. Contiene 7 pins de propósito general usados por defecto PIN 11, PIN 12, PIN 13, PIN 15, PIN 16, PIN 18 y PIN 22. Mientras el PIN 7 provee una señal de reloj puede funcionar también como pin de propósito general. [6]

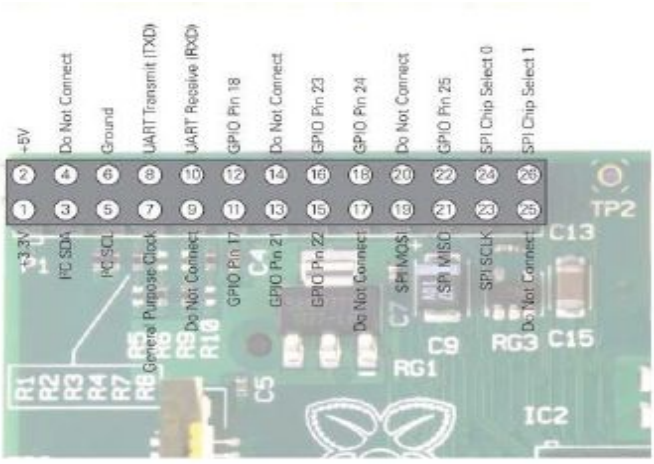



**Fig. 2.9 PINES DEL PUERTO GPIO**

**Fuente: Libro Raspberry Pi User Guide EbenUptonPag. 246**

## 1.6.2 UART SERIAL BUS

El Raspberry Pi tiene incluido un convertidor de nivel para la interfaz RS-232, en este bus serial usado como un puerto para transmitir mensajes. El pin 8 se utiliza para transportar las señales transmitidas y el pin 10 se lo utiliza para transportar las señales recibidas su rapidez puede ser cambiada en el fichero cmdline.txt se utiliza usualmente en 115.200 bits por segundos. [6]

## 1.6.3 I2C BUS

El Inter-IntegratedCircuit (I2C) bus se utiliza para proveer comunicación entre múltiple circuitos integrados estos incluyen resistencias pull-up, se puede acceder por el pin 3 provee la señal Serial Data Line (SDA) y pin 5 provee la señal del clock serial (SCL). [6]

### 1.6.4 SPI BUS

La interface periféricos serial (SPI) bus es un bus serial sincrónico diseñado para sistemas de programación de microcontroladores y otros dispositivos. Se diferencia de los buses UART y I2C, esto es un bus de 4 alambres con múltiples líneas de selección de chip cual permite comunicar con más de una tarjeta de dispositivos.

El pin 19 provee la salida maestra SPI, señal esclava de entrada (MOSI); pin 21 provee la entrada master SPI, señal esclava de salida (MISO); pin 23 provee la señal del reloj (SLCK) usada para sincronizar la comunicación; y el pin 24 y 26 provee las señal selección de chip para 2 dispositivos esclavos independientes. [6]

### 1.6.5 GPIO PORT CONTROLADOS POR LENGUAJE C

Para lograr establecer una comunicación entre nuestro lenguaje C y los pines GPIO se debe instalar ciertas librerías que existen en la WEB como la librería WiringPi. El Raspberry Pi permite periféricos y tarjetas de expansión para acceder al CPU para exponer las entradas y salidas que usaremos para nuestro proyecto.

## 1.7 LIBRERÍA WIRINGPI

La librería WiringPi escrita en C ofrece un acceso hacia los pines de propósito general es similar a la librería arduPI de Arduino soportaalgunas

funciones que esta realiza como UART, I2C,SPI y PWM desarrollada por Gordon Herdenson. [4]

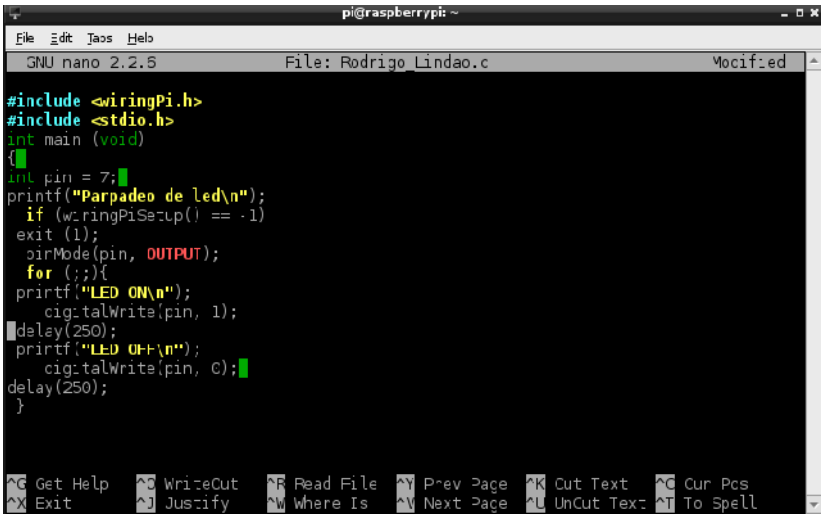



**Fig. 2.10 Creación de Programa en C y Librería wiringPi**

**Fuente: Realizado por Rodrigo Lindao Suárez**

**Ejemplo http://www.electroensaimada.com/parpadeo-led.html**

## 1.7.1 INSTALACIÓN DE LIBRERÍA WIRINGPI

Para Instalar nuestra Librería WiringPi necesitamos ciertos comandos que nos permitirán descargar los paquetes que contienen las aplicaciones para controlar el puerto GPIO del Rasberry Pi como mostramos a continuación:

- El git-core es un sistema de control de versión distribuido, nos permite guardar ciertos archivos que se cambian con frecuencia, ayudándole al usuario comparar versiones, cargar versiones anteriores y añadir nuevas versiones.

# **sudo apt-get install git-core**

- Instalamos la Librería WiringPi por medio de GIP [4]

# **git clone git://git.drogon.net/wiringPi**

- Si ya hemos usado la operación de reproducción por primera vez, entonces

# **cdwiringPi**

# **gitpullorigin**

- Para construir/instalar hay una nueva escritura simplificada [4]

# **./build**

## 1.8 FUENTE DE ENERGÍAS RENOVABLES

Se llama energía renovable la que administrada en forma adecuada, puede explotarse ilimitadamente, es decir, su cantidad disponible no disminuye a medida que se aprovecha.

La principal fuente de energía renovable es el sol, ya que el sol envía a la Tierra únicamente energía radiante, es decir, luz visible, radiación infrarroja y algo de ultravioleta. En algunos casos, en la atmosfera se convierte en una variedad de efectos, algunos de los cuales pueden ser un recurso energético como la energía eólica, la energía biomasa, la energía de las olas y la diferencia de temperaturas oceánicas, nos concentraremos en la energía solar.

## 1.9 ENERGÍA SOLAR

La energía solar es una fuente de vida y origen de la mayoría de las demás formas de energía en la Tierra. Cada año la radiación solar aporta a la Tierra la energía equivalente a varios miles de veces la cantidad de energía que consume la humanidad. Recogiendo de forma adecuada la radiación solar, esta puede transformarse en energía térmica y fotovoltaica, siendo esta última en la que nos centraremos.

## 1.10 HISTORIA DE LA ENERGÍA SOLAR FOTOVOLTAICA

La generación de electricidad con celdas fotovoltaicas solares fue descubierta en el siglo anterior, en 1839 por Edward Becquerel cuando observó casi accidentalmente la aparición de un voltaje en las terminales de un pedazo de selenio en electrolito.

El desarrollo de las células fotovoltaicas surgió durante los programas de investigación espacial, especialmente entre 1950 - 1970 debido a sus características idóneas, bajo peso, larga vida, resistencia al ambiente exterior y alta confiabilidad.

Durante los siguientes 15 años la tecnología de la celda fue mejorada y estuvo lista para que en 1975, con la llegada de la crisis petrolera y el incremento de los costos de energía, se pensara en dicha tecnología como alternativa de solución [13].

Desde entonces, los precios de las celdas fotovoltaicas han ido bajando y su eficiencia gradualmente se ha incrementado y por tanto los costos por unidad de electricidad generada por las celdas fotovoltaica empiezan a ser cada vez más bajos, hoy en día ya no solo se usa para aplicaciones espaciales sino también comerciales.

Hasta ahora, muchos de los paneles solares fotovoltaicos utilizados son celdas cristalinas o de silicio. Este tipo de módulos convierte la luz solar en electricidad con una eficiencia del 12 al 15 %. Las recientes mejoras en eficiencia probablemente influirán en una mayor utilización de estas celdas solares aunque se espera que su eficiencia vaya creciendo con el paso de los años.

Otro desarrollo reciente es el mejoramiento del comportamiento de módulos de silicio amorfo. La ventaja de un silicio amorfo es el bajo precio. Inicialmente estas celdas producían electricidad con una eficiencia muy baja (5%) pero con mejoras recientes (después de su estabilización) puede acercarse al 10% [13].

## 1.11 LA CÉLULA FOTOVOLTAICA

Son dispositivos que transforman energía solar en electricidad, la luz ingresa directamente sobre un dispositivo semiconductor de dos capas produciendo una diferencia de voltaje que genera una corriente a través de un circuito externo.

Se construye con un semiconductor tipo n y otro semiconductor tipo p de tal modo que la red cristalina del semiconductor no se interrumpa al pasar de una región a otra. [7]

**Fig. 2.11 Modulo Solar para Arduino**

**Fuente: http://www.cooking-hacks.com/index.php/solar-panel-10w.html**

La unión pn hace posible la aparición de un campo eléctrico en la célula (con la dirección del lado n al lado p) que separa los pares en los huecos, cargas positivas, los dirige hacia el contacto del lado p lo que provoca la extracción de un electrón desde el metal que constituye el contacto; los electrones, cargas negativas, los dirige hacia el contacto del lado n inyectándolos en el metal. Esto hace posible el mantenimiento de una corriente eléctrica por el circuito. [7]

## 1.12 FUNCIONAMIENTO DE UNA CÉLULA FOTOVOLTAICA.

Cuando la luz solar incide sobre un panel solar se genera (g) pares eh. Cada par individual constituye un potencial electrón circulando por el circuito exterior. Debido a esto tenemos los procesos de recombinación (r). Cada proceso de recombinación desechara uno de los pares eh generados y, por lo tanto, tendremos un electrón menos disponible para circular por el circuito exterior. De ahí que se hable del mecanismo de recombinación como un mecanismo de pérdidas para la célula. [7]

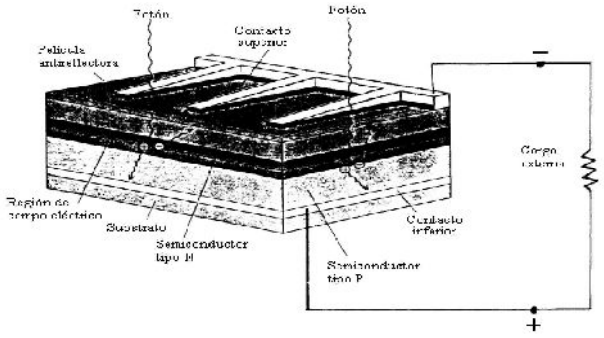



**Fig. 2.12 Elementos de una celda solar de SI**
**Fuente: www.uam.es/departamentos/ciencias/qinorg/.../Celdas%20solares.ppt**
**Presentación Placas Solares Diapos. 11**

Si existe una corriente I circulando por el circuito exterior, esta provocará una caída de tensión en la resistencia, que se traslada a los bornes de la resistencia, lo que significa que la célula debe operar a una tensión V = R·I. Esta tensión afecta a la recombinación, debido a que ésta depende

de V de manera exponencial. Entonces si la carga es muy elevada, también lo será la tensión y en consecuencia la recombinación desechará todos los procesos de generación e impedirá la circulación de la corriente.[7]

## 1.13 MODELO DE DUFFIE & BECKMAN

Una célula fotovoltaica funciona como un generador de corriente, que puede ser modelada mediante una fuente de corriente $I_L$ debida a la generación de portadores de iluminación, conectada en paralelo a un diodo, cuya corriente $I_D$ (corriente de diodo de oscuridad según la teoría de Shockley) esto se debe a la recombinación de portadores como consecuencia de la tensión entre sus extremos [10].

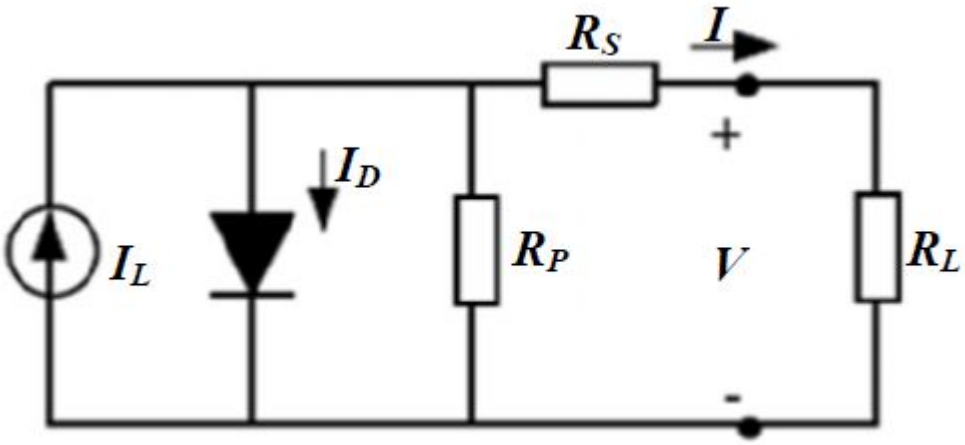



**Fig. 2.13 Circuito equivalente de la célula solar**
**Fuente: www.cip.org.pe/imagenes/temp/tesis/43324406.pdf Pag. 23**

De acuerdo a este modelo se comienza a partir de la siguiente ecuación que caracteriza el circuito equivalente de la célula solar.

$$I = I_L - I_0 \left[ e^{\frac{V+I*Rs}{Vt}} - 1 \right] - \frac{V + I * Rs}{Rp}$$

Sobre esta ecuación se harán diferentes consideraciones que la simplificarán de cuyo resultadoobtendremos una ecuación más simple con una buena aproximación del comportamiento del panel [10].

Como el término exponencial es mucho mayor que uno, el último término desaparece y debido a que $R_P \to \infty y Rs \to 0$; además en cortocircuito se cumple que $I_L = I_{sc}$ y en circuito abierto se cumple que $I_0 = I_L * e^{\frac{-Voc}{Vt}}$ la ecuación de intensidad queda como [10]:

$$I = I_{SC} * \left[ 1 - e^{\frac{V-Voc}{Vt_{panel}}} \right]$$

Isc: Intensidad de cortocircuito en condiciones estándares.

V: Tensión en bornes del Panel Solar.

Voc: Tensión de circuito abierto en condiciones estándares.

Vt: Tensión umbral de los diodos que forman el panel. Esta tensión umbral se puede extraer para un diodo como $Vt_{panel} = mVt$; donde m es el número de células del panel solar.

$$V_t = \frac{nKT}{q}$$

n: factor de idealidad del diodo (se considera 1).

K: constante de Boltzmann ($1.38 * 10^{-23}[J/K]$).

q: carga del electrón ($1.6 * 10^{-19}[C]$)

T: temperatura del panel solar.

## 1.14 PARÁMETROS FUNDAMENTALES DE UN CÉLULA FOTOVOLTAICA.

Generalmente en las hojas de características de los módulos fotovoltaicos suelen aparecer datos eléctricos que determinan el comportamiento eléctrico del módulo bajo unas condiciones estándar de medida (STC) que suelen ser, por convenio internacional, de 1000w/m2 y T=25ºC [7].

Algunas de estas especificaciones técnicas suelen ser:

- La intensidad o corriente de cortocircuito (Isc) se produce a tensión cero y se mide conectando un simple amperímetro a la salida del panel.
- La tensión de circuito abierto (Voc) es realmente la tensión máxima que puede dar un panel y se mide directamente entre bornes del panel con un voltímetro.

- La potencia pico o Wmp es simplemente el producto real máximo de la corriente y la tensión producida. Obviamente la Wpt teórica es superior a la Wmp real, eso se explica a continuación con el Factor de Forma.

El producto de Isc x Voc se representa por el área bajo la curva, donde el extremo superior derecho representa la potencia máxima teórica Wpt = Isc x Voc. Sabemos que la potencia real Wmp = Imp x Vmp será menor que la potencia teórica Wpt = Isc x Voc. Pues el factor de forma es el cociente de FF = Wmp / Wpt y su resultado, será siempre menor que uno. Este dato nos da una idea de la calidad del panel [7].

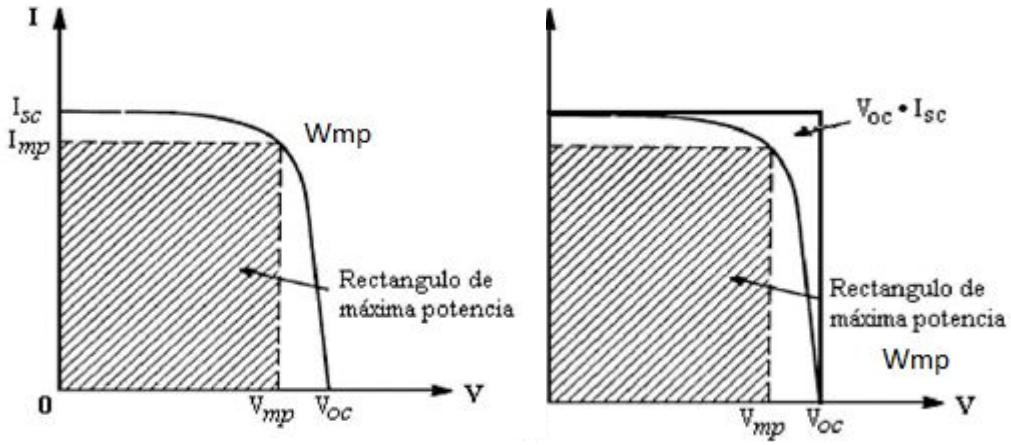



**Fig. 2.14 Características de I-V de Células Solares**

**Fuente: www.uam.es/departamentos/ciencias/qinorg/.../Celdas%20solares.ppt**

**Presentación Placas Solares Diapos. 12**

La cantidad de energía que entrega un dispositivo fotovoltaico está determinado por:

- El tipo y el área del material.
- La intensidad de la luz del sol.
- La longitud de onda de la luz del sol.

Por ejemplo, las células fotovoltaicas de silicio monocristalino actualmente no pueden convertir más del 25% de la energía solar en electricidad, porque la radiación en la región infrarroja del espectro electromagnético no tiene suficiente energía como para separar las cargas positivas y negativas en el material [7].

Las células fotovoltaicas de silicio policristalino en la actualidad tienen una eficiencia de menos del 20% y las celdas amorfas de silicio tienen actualmente una eficiencia cerca del 10%, debido a pérdidas de energía internas más altas comparadas con el de silicio monocristalino [7].

Una característica importante de las celdas fotovoltaicas es que el voltaje de la célula no depende de su tamaño, y sigue siendo bastante constante con el cambio de la intensidad de luz. La corriente en un dispositivo, sin embargo, es casi directamente proporcional a la intensidad de la luz y al tamaño. Para comparar diversas celdas se las clasifica por densidad de corriente, o amperios por centímetro cuadrado del área de la célula.

La potencia entregada por una célula solar se puede aumentar con bastante eficacia empleando un mecanismo de seguimiento para mantener el dispositivo fotovoltaico directamente frente al sol, o concentrando la luz del sol usando lentes o espejos. Sin embargo, hay límites a este proceso, debido a la complejidad de los mecanismos, y de la necesidad de refrescar las celdas. La corriente es relativamente estable a altas temperaturas, pero el voltaje se reduce, conduciendo a una caída de potencia a causa del aumento de la temperatura de la célula [7].

# CAPÍTULO 3

# 3 DESARROLLO Y DISEÑO DEL PROYECTO

## 3.1 INTRODUCCIÓN

Para la evaluación de los paneles solares se debe implementar y diseñar un programa que sea capaz de mostrar los datos adquiridos en nuestra minicomputadora, en este capítulo realizaremos la implementación física y realizaremos un diagrama de bloques del mismo, además efectuaremos el código fuente para la ejecución del proyecto y su debido diagrama de flujo que es fundamental para un mejor entendimiento.

Con el fin de obtener experiencia con el puerto GPIO de la minicomputadora, realizamos dos prácticas, que fueron rotación de Led y la función de PWM en uno de los pines del puerto.

Nuestro enfoque principal tanto para las prácticas y proyecto se estableció en el uso de la librería wiringPi, creada por Gordon Henderson en las páginas principales dedicada al uso de esta librería http://wiringpi.com/ y https://projects.drogon.net/raspberry-pi/.

## 3.2 IMPLEMENTACIÓN DE LA INTERFAZ PARA MEDICIÓN DE PARÁMETROS MEDIANTE EL RASPBERRY PI

### 3.2.1 DIAGRAMA DE BLOQUES DEL PROYECTO.

Evaluaremos un panel solar y para lograr este objetivo debemos adquirir los datos de voltaje, corriente y potencia. El panel solar cuando es expuesto a la luz produce un valor de voltaje y corriente, estos valores deben ingresar a nuestra minicomputadora, utilizamos un PIC16F887 para cumplir ese objetivo. En la siguiente figura observamos el diagrama de bloques de cómo está conformado nuestro proyecto.

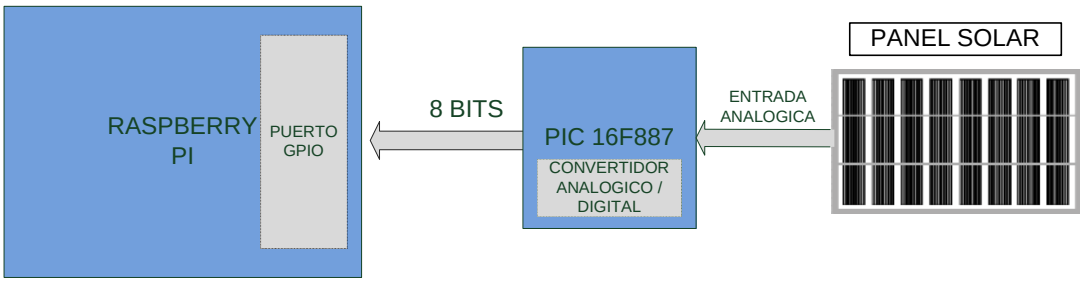



**Fig. 3.1 Diagrama de bloque de nuestro proyecto.**
**Fuente: Realizado por Rodrigo Lindao Suárez**

Bosquejamos un gráfico en el cual podemos visualizar como será conectado todos nuestros instrumentos, y detallaremos cuales son los pines que estamos utilizando en el puerto GPIO del Raspberry PI y los pines del PIC.

Del PIC16F887 utilizamos los siguientes pines:

Pin 1 - MCLR, master clear reset.

Pin 4 - Vref- , entrada Análoga que viene desde el panel solar.

Pin 11 - Vdd , alimentación +5V.

Pin 12 – Vss , GND

Pin 33 – RB0 , salida digital bit menos significativo.

Pin 34 – RB1 , salida digital.

Pin 35 – RB2 , salida digital.

Pin 36 – RB3 , salida digital.

Pin 37 – RB4 , salida digital.

Pin 38 – RB5 , salida digital.

Pin 39 – RB6 , salida digital.

Pin 40 – RB7 , salida digital bit más significativo.

Los pines del puerto GPIO son los siguientes:

Pin 2 - +5V.

Pin 6 - GND.

Pin 11 – Entrada digital bit más significativo.

Pin 12 – Entrada digital.

Pin 13 – Entrada digital.

Pin 15 – Entrada digital.

Pin 16 – Entrada digital.

Pin18 – Entrada digital.

Pin 22 – Entrada digital.

Pin 7 – Entrada digital bit más significativo.

La figura nos muestra cómo se realiza la conexión de todos los pines antes mencionados entre el Rapberry Pi y el PIC16F887.

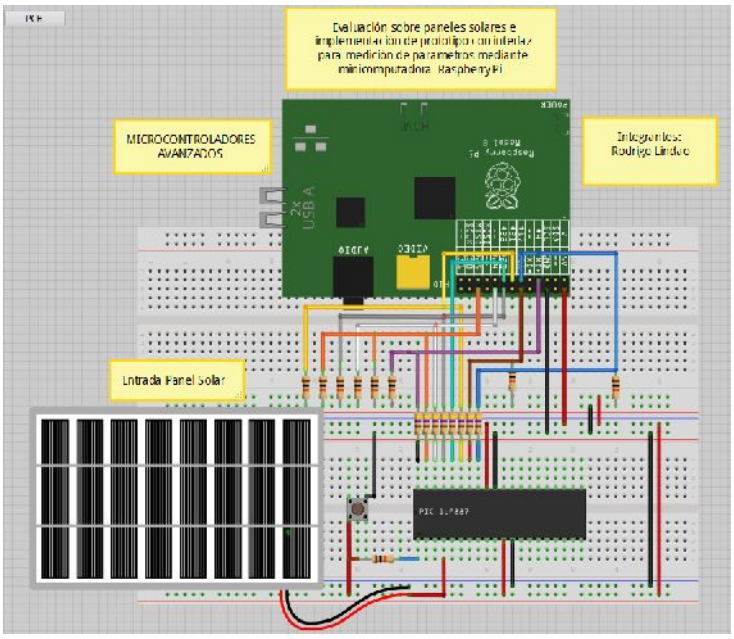


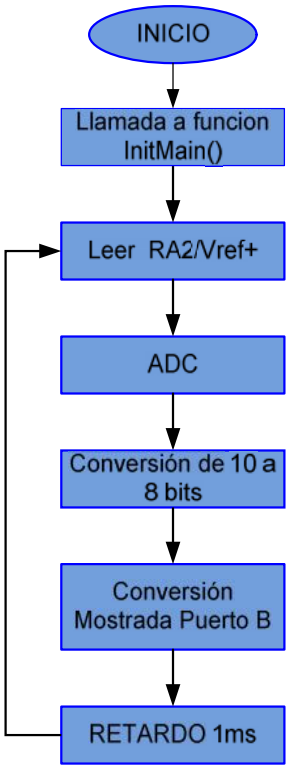

**Fig. 3.2 Esquemático realizado en Fritzing del prototipo usado en nuestro proyecto**

**Fuente: Realizado por Rodrigo Lindao Suárez**

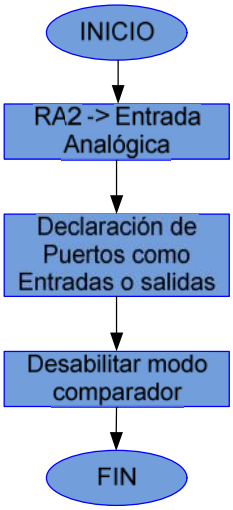
Nuestro diseño se basa fundamentalmente en adquirir datos del panel solar por medio de una señal analógica y debido a que el puerto GPIO solo tiene entradas digitales debemos utilizar un convertidor analógico/digital, para nuestro proyecto usaremos el PIC16F887como un ADC, el cual convertirá la señal analógica del panel solar en una señal digital representada en 8 bits que ingresara al puerto GPIO de la minicomputadora.

## 3.2.2 DIAGRAMA DE FLUJO PARA RUTINAS EN EL PIC 16F887

**Programa Principal**




**Función InitMain()**



## Documentación:

Para nuestro programa principal iniciamos con la declaración de los puertos los cuales serán utilizados como entrada o salida en la función InitMain(), comienza a leer el pin RA2 donde obtendremos el voltaje analógico proveniente del panel solar. Este valor es enviado hacia el convertidor analógico/digital que posee nuestro PIC16F887 en el cual realizamos la conversión hacia 8 bits que serán mostrados en cada uno de los pines correspondientes en el puerto B el cual estará 1ms para luego volver a leer un nuevo valor de voltaje en el panel solar.

### 3.2.3 CÓDIGO FUENTE PARA RUTINAS EN EL PIC 16F887

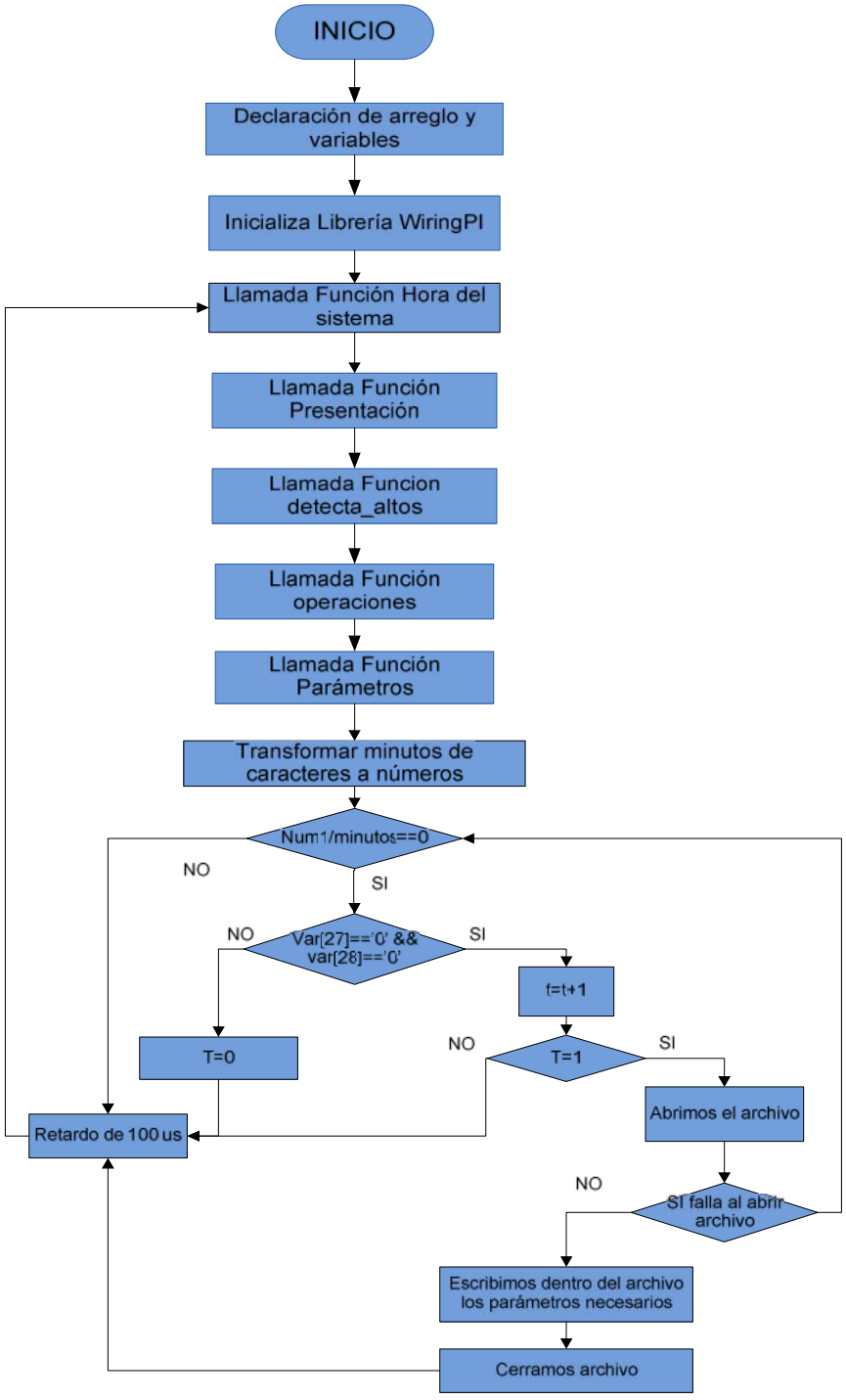
```
//Materia: Microcontroladores Avanzados
//Programa para PIC16F887 Conversión ADC 8-BITS

unsignedinttemp_res;          //declaro variables

voidInitMain()
{
    ANSEL  = 0x04;             // Configuro          PIN AN2 como entrada
analógica
    ANSELH = 0;                // Configuro otros PIN AN I/O digitales
    TRISA  = 0xFF;             // PORTA como entrada
    TRISC  = 0;                // PORTC como salida
    TRISB  = 0;                // PORTB como salida
    TRISD  = 0;                // PORTB como salida
C1ON_bit = 0;            // Desabilitarcomparador
    C2ON_bit = 0;
}
voidmain()
{
InitMain();
do {
        temp_res = ADC_Read(2);    // Obtengo el resultado de la
conversion ADC 10-bit
        temp_res = temp_res/4;      // convierto de 10 bit a 8 bit
        PORTB = temp_res;            // envio los 8 bits al Puerto B
        Delay_ms(1);                 // retardo antes de realizar una nueva
conversion
         } while(1);
return 0;
}
```

## 3.3 DIAGRAMA DE FLUJO PARA RUTINAS EN EL RASPBERRY PI

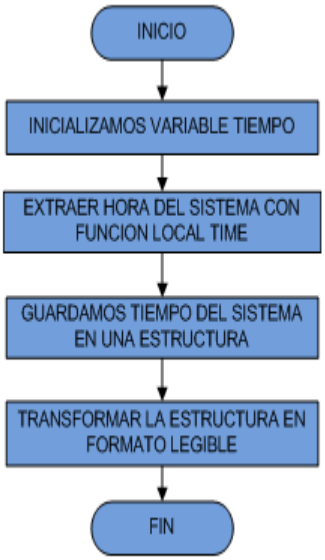
**Programa Principal**



## Documentación:

Nuestro Programa Principal realiza la función hora esta nos proporciona el tiempo y fecha real del sistema mostrada en la pantalla principal, guarda los datos en un arreglo de caracteres, luego realizamos paso a paso cada una de las llamadas a las funciones principales implementadas para nuestra proyecto que explicaremos más adelante.

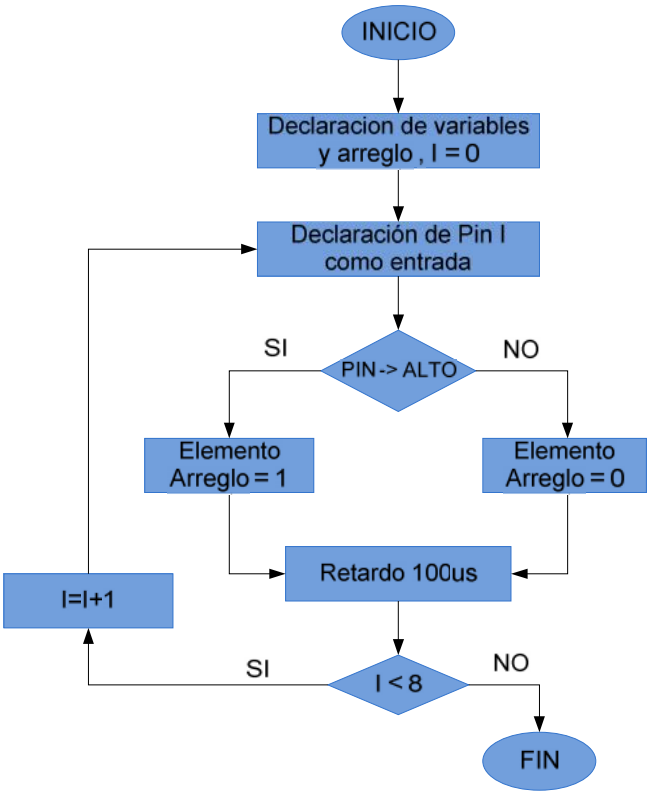
**Función hora**




Para realizar la captura automática de nuestros parámetros a analizar tenemos que transformar la hora de caracteres a números con lo cual establecemos la relación entre los minutos actuales/minutos de usuario, si el residuo=0 verificamos el elemento 27 y 28 del arreglo que contiene los segundos, si estos elementos son ceros procedemos a incrementar bandera t en 1, verificamos si t=1 para realizar la apertura del archivo, si realiza correctamente esta operación escribimos nuestros parámetros,

una vez terminada esta operación cerramos el archivo y el sistema realiza un retardo de 100us antes de una nueva lectura.

**Función  detecta altos**

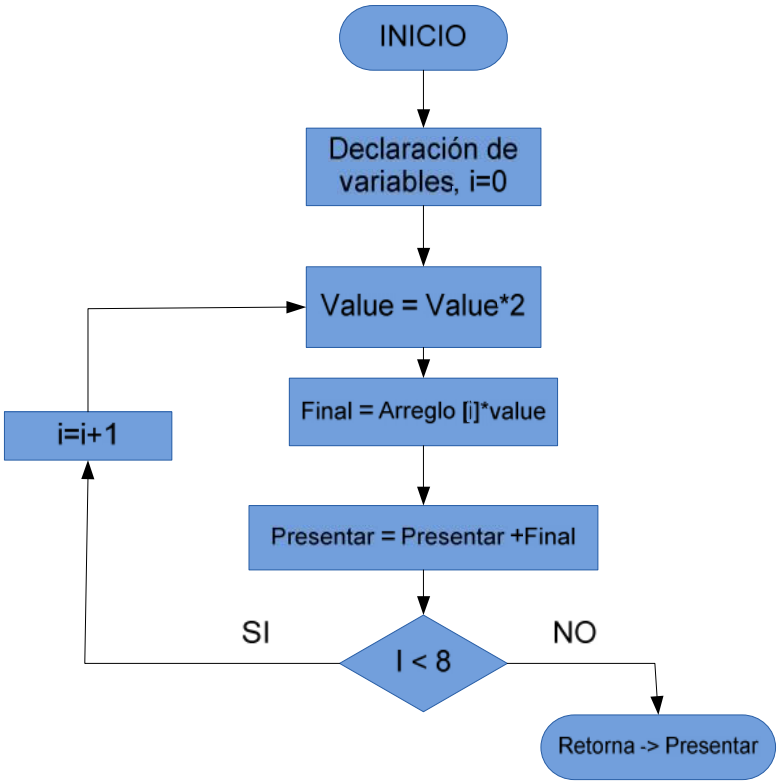



## Documentación:

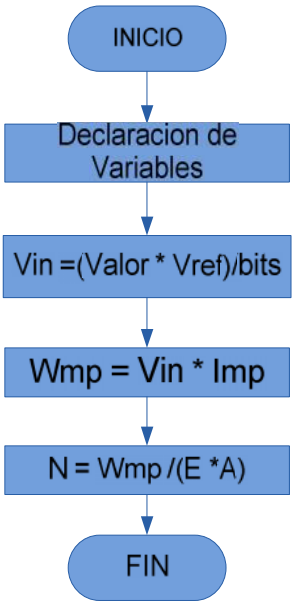
La función detecta altosse declara las variables y el arreglo, configuramos el Pin I del GPIO como entrada lo analizamos si es un alto o un bajo según esta información ingresamos nuestro datos en el arreglo esperamos 100us. Si  Pin I < 8  volvemos al análisis de un nuevo pin sino damos por terminado el proceso.

La función hora iniciamos la variable tiempo en donde se guardará el tiempo del sistema, usamos la función localtime que nos permite tomar la hora del sistema estos datos se guardarán en una estructura, para lograr obtener esta información en un formato legible usamos la función strftime.

## Función Operaciones



**Función Parámetros**




## Documentación:

La función Operaciones nos ayuda a transformar el número binario tomado del convertidor analógico / digital para convertirlo en número decimal que usaremos más adelante en nuestra función parámetros, recorremos mediante un lazo que nos permitirá analizar cada uno de los 8 bits. Si el bit es uno el valor correspondiente a la conversión será sumado a la variable Presentar que acumulará el valor total de la conversión.

La función Parámetros recibe el valor de la variable Presentar que representa la conversión digital / analógico para posteriormente calcular los parámetros correspondientes al Valor analógico obtenido a la entrada del PIC16f887, la potencia máxima y la eficiencia del panel analizado.

## 3.4 CÓDIGO FUENTE PARA RASPBERRY PI PROGRAMA PRINCIPAL

```
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include<time.h>#include <wiringPi.h>
#include<math.h>
#define Vref 8 //Voltaje entrada máximo en el panel solar
#define bits 256 //bits de entrada del raspberry 2^8
#define Isc 0.6 //corriente de cortocircuito
#define minutos 1 //indica el periodo con que los minutos serán tomados para muestra
#define Celdas 18 //números de células fotovoltaicas
#define T 298 //Temperatura de operación del panel solar
#define Constante -0.00008625 //parte de ecuación K(1.38*10-23)m(1)/q(1.60*10-19)
floatWmp,Vin,n,Imp;
voiddetecta_altos(int arreglo[])
{
inti=0;
for(i=0;i < 8;i++)
        {
pinMode(i,INPUT);
```

```
if(digitalRead(i)==HIGH)
arreglo[i]=1;
else
arreglo[i]=0;
      }
 }
intoperaciones(intarreglo[])
{
inti=0,j=0;
int  final=0,presentar=0,value=0;
for(i=0;i<8;i++)
      {
value=1;
for(j=0;j<i;j++)
value = value*2;
final = arreglo[i]*value;
presentar=presentar+final;
}
returnpresentar;
 }
voidparametros(intvalor,char output[])
 {
doubleV,Vt,t,k;
printf("\n               Dia/Mes/Ano   Horas:Minutos:Segundos\n");
printf("                %s\n",output);
```

```
        // printf("                    ResultadoBinario=%2d\n",valor)
         Vin =(((double)valor*(double)Vref)/(double)bits);
Vt=Celdas*Constante*T;
         k=(Vref-Vin)/Vt;
         t=pow(2.71,k);
         Imp=Isc*(1-t);
        // printf("%2f  %2f %2f\n",V,t,k);
Wmp=(double)Vin *(double)Imp;
printf("\n\t        Voltage        Corriente        Potencia      \n");
printf("              %2f[V]      %2f[A]      %2f[W]       \n",Vin,Imp,Wmp);
}
inthora(char output[])
{
time_ttiempo = time(0);
struct tm *tlocal =localtime(&tiempo);
strftime(output,128,"%d/%m/%y           %H:%M:%S",tlocal);
return 0;
}
voidPresentacion(void)
{
system("/usr/bin/clear");
printf("    ************************************************************\n");
printf("    * R                    Y              *******     *\n");
printf("    *     A                R              * *   * *    *\n");
printf("    *        S             R              *   *   *    *\n");
```

```
printf("   *          P      E       MICROCONTROLADORES    *  ***  *
*\n");
printf("   *          B                     *******    *\n");
printf("   *        P     E        AVANZADOS           *\n");
printf("   *     S        R                        *\n");
printf("   *   A            R                      *\n");
printf("   *R                  Y        (WELCOME)       *\n");
printf("
*****************************************************************\n\n");
printf("EVALUACION     SOBRE     PANELES     SOLARES     E
IMPLEMENTACION DE PROTOTIPO CON INTERFAZ\n");
printf("          PARA  MEDICION  DE  PARAMETROS  MEDIANTE
MINICOMPUTADORA RASPBERRY PI\n");
printf("               DESARROLLADO POR:\n\n");
printf("                     RODRIGO     LINDAO     SUAREZ
rrlindao@espol.edu.ec\n");
}
main()
{
int matriz[8],resultado,t=0,num1=0;
charvar[128];
 FILE *fa;
if(wiringPiSetup()==-1) //Iniciamos la LibreriaWiringPI
exit(1);
```

```
for(;;)
{
hora(var);
Presentacion(); //Llamamos a funciónpresentación
detecta_altos(matriz);
resultado=operaciones(matriz); //Función para transformar valor binario
parametros(resultado,var);
        num1=((var[24]-48)*10+(var[25]-48));
if(num1%minutos==0)//Tomar datos cada minuto
            {
if((var[27]=='0')&&(var[28]=='0'))//Para coger datos cada segundo
{    t++;
if(t==1)
                    {
fa=fopen ("/home/pi/Voltage","at");
if(fa==NULL){
printf("Error\n");
return;
                        }
fprintf(fa,"%s %2f %2f %2f %2f\n",var,Vin,Wmp,Imp,n);
fclose(fa);
                    }
             }
else
                    t=0;
```

```
            }
delay(100); //retardo para mostrar datos en pantalla.
}
return 0;
}
```

## 3.5 PRÁCTICA I : ROTACIÓN DE LEDS

## 3.6 INTRODUCCIÓN

Adicionalmente al tema principal de nuestro proyecto se realizaron prácticas para conocer las bondades que nos ofrece el puerto GPIO del Raspberry Pi, estas experiencias nos ayudaran a reforzar la teoría aprendida en cuanto: punteros, manejo de arreglos y retardos.

En Raspberry PI creamos una rutina para realizar el encendido y apagado de los pines del GPIO mediante un arreglo que indica: número de led, estado de ledon/off y el retardo de tiempo que permanecerá en ese estado este código fuente se extrajo dela dirección https://projects.drogon.net/raspberry-pi/examples/.

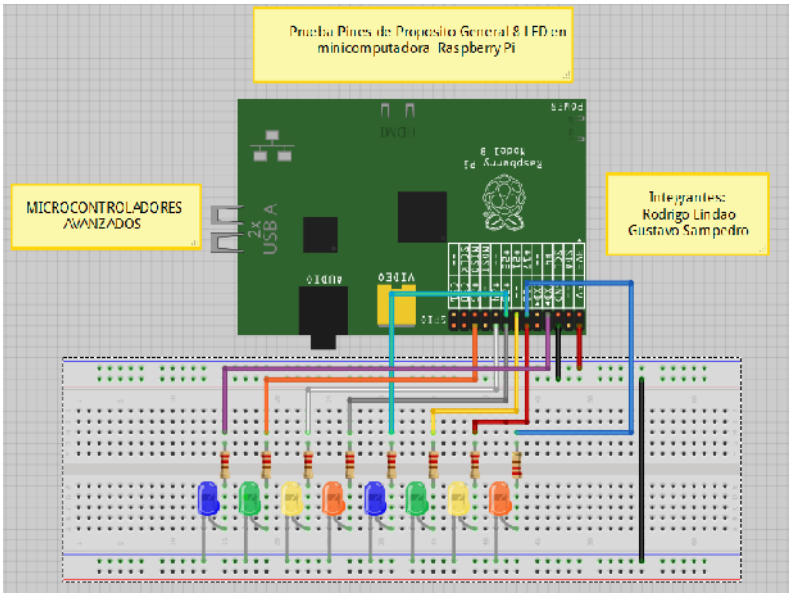



**Fig. 3.3 Esquemático realizado en Fritzing uso de los pines del GPIO**

**Fuente: Realizado por Rodrigo Lindao Suárez**

## 3.7 DIAGRAMA DE BLOQUES PRÁCTICA I

El Puerto GPIO del Raspberry Pi realiza él envió de las secuencias hacia los 8 led para comprobar el correcto funcionamiento de las operaciones realizadas en este diseño.

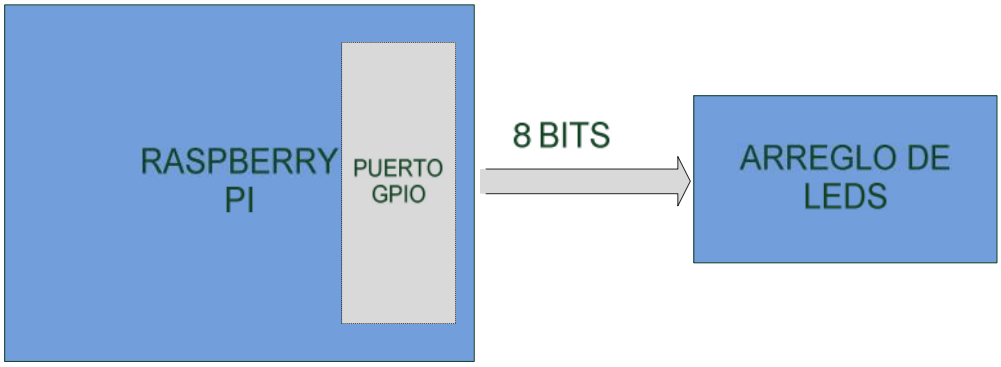



**Fig. 3.4 Diagrama bloque práctica #1**

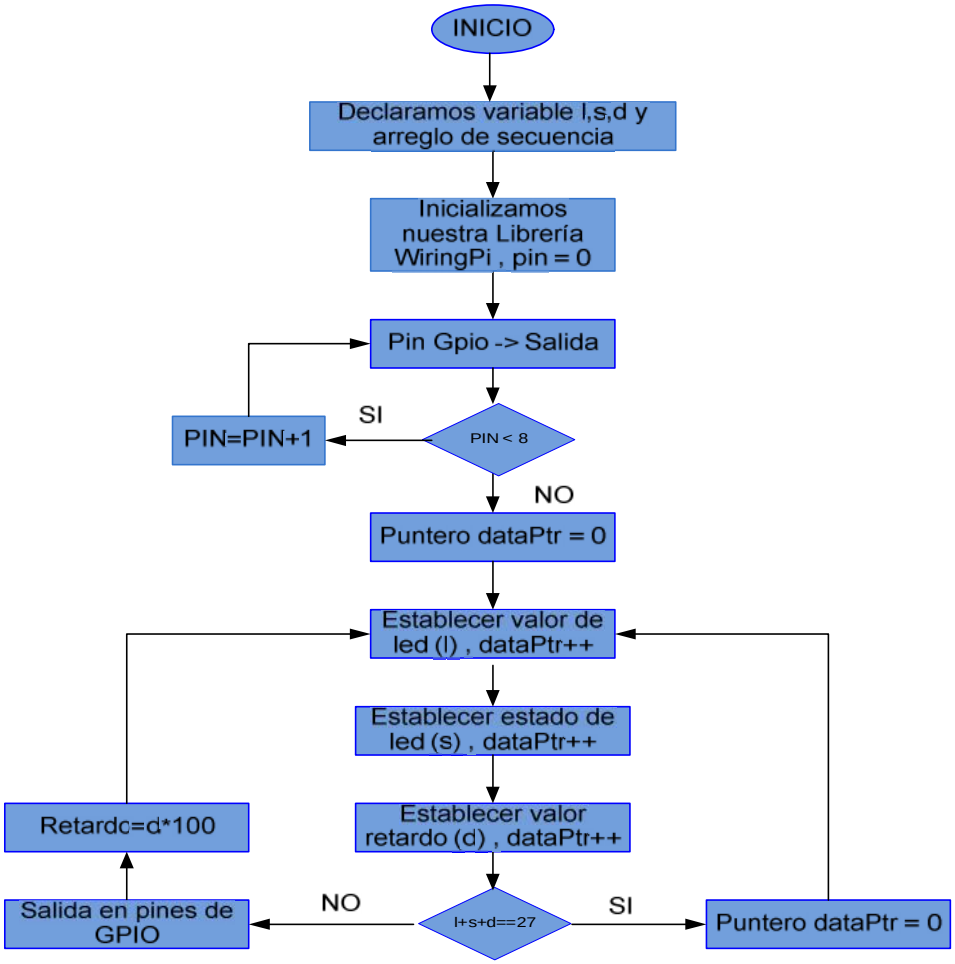
## 3.8 DIAGRAMA DE FLUJO PRÁCTICA I

### Documentación:

La estrategia utilizada para realizar el presente proyecto es utilizar el puerto GPIO del Raspberry Pi, inicializamos nuestro arreglo, las variables a utilizar y puntero para recorrer los diferentes elementos del arreglo.

Declaramos los pines del GPIO como salida y colocamos nuestro puntero en cero, luego establecemos que pin utilizaremos para enviar el estado de on/off y el tiempo (retardo) que permanecerá en este estado.

Finalmente verificamos si se encuentra dentro de nuestra secuencia establecida por el arreglo o si llega a nuestra secuencia bandera que nos indicara el regreso hacia el inicio de la secuencia, caso contrario seguirá dentro de la secuencia establecida por el arreglo.




## 3.9 CÓDIGO FUENTE DEL PRÁCTICA I : ROTACIÓN DE LEDS

```
#include <wiringPi.h>
#include <stdio.h>
#include<stdlib.h>
```

```
#include <stdint.h>

// Secuencias Simples de Datos
// Pruebas de Ledon/off  y retardos

uint8_t data [ ] =
{
//Secuencia hacia adelante
              0, 1, 1,
              1, 1, 1,
  0, 0, 0,  2, 1, 1,
  1, 0, 0,  3, 1, 1,
  2, 0, 0,  4, 1, 1,
  3, 0, 0,  5, 1, 1,
  4, 0, 0,  6, 1, 1,
  5, 0, 0,  7, 1, 1,
  6, 0, 1,
  7, 0, 1,
  0, 0, 1,                  // Retardos extra
                           // Secuencia hacia atrás
              7, 1, 1,
              6, 1, 1,
  7, 0, 0,  5, 1, 1,
  6, 0, 0,  4, 1, 1,
  5, 0, 0,  3, 1, 1,
  4, 0, 0,  2, 1, 1,
  3, 0, 0,  1, 1, 1,
  2, 0, 0,  0, 1, 1,
  1, 0, 1,
  0, 0, 1,
  0, 0, 1,                 // Retardos extra
  9, 9, 9,                // Secuencia de Finalización};
```

```
//PROGRAMA PRINCIPAL

int main (void)
{
int pin ;
intdataPtr ;
int l, s, d ;
if (wiringPiSetup () == -1)  //Iniciamos el programawiringPi
  exit (1) ;
for (pin = 0 ; pin < 8 ; ++pin)
  pinMode (pin, OUTPUT) ;  //Declaramos pines de gpio como salidas
dataPtr = 0 ;                 //Puntero que recorre el arreglo
  for (;;)
  {
             l = data [dataPtr++] ;      // define número de led
           s = data [dataPtr++] ;      // define el estado del led on /off
           d = data [dataPtr++] ;      // valor que nos definirá el retardo

           system("/usr/bin/clear");
           printf("\n\n\n\n\t\t\t************************************\n");
           printf("\t\t\t** Programa Prueba Encendido 8 Led **\n");
           printf("\t\t\t************************************\n");
           printf("\n\n\t\t\t         Led (%2x) Estado : ON\n",l+1);

if ((l + s + d) == 27)          //define termino de recorrido del puntero en el
arreglo
           {
  dataPtr = 0 ;          //inicia el puntero en cero para iniciar una nueva
secuencia on/off en led
  continue ;             //envía hacia el inicio del lazo
            }
```

```
  digitalWrite (l, s) ;          //salida en el pin definido por l y s indica si el led
está on/off
  delay (d * 100) ;            //define retardo de tiempo
  }
return 0 ;
 }
```

## 3.10 PRÁCTICA II: USO DEL PWM

## 3.11 INTRODUCCIÓN

En este programa realizamos la implementación del PWM en el Raspberry Pi en el pin específico para realizar esta operación.

El Raspberry PI creamos una rutina para realizar el incremento de la intensidad y posterior reducción de la intensidad de un led conectado a un pin del puerto GPIO del Raspberry Pi.

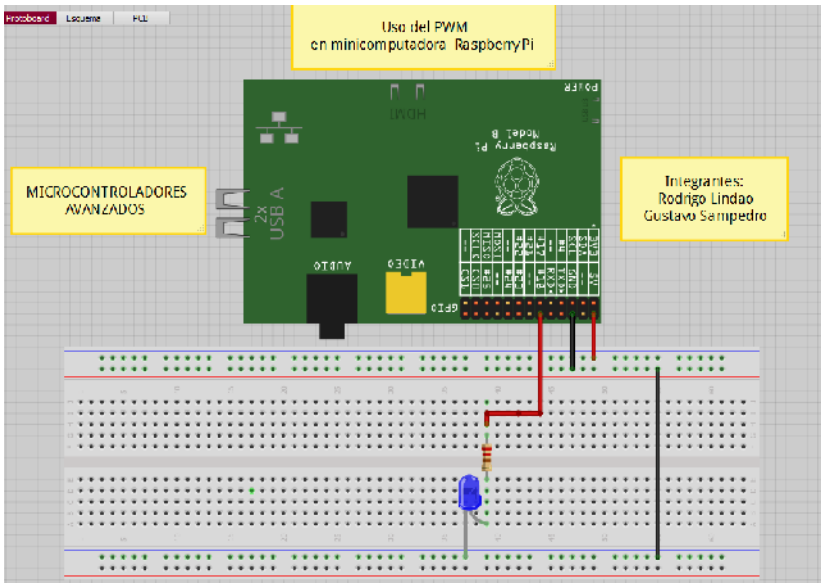



**Fig. 3.5 Esquemático realizado en Fritzing uso del PWM**
**Fuente: Realizado por Rodrigo Lindao Suárez**

## 3.12 DIAGRAMA DE BLOQUES PRÁCTICA II

En el diodo led verificaremos el cambio de frecuencia en PWM ejecutadas en el Pin PWM del Raspberry Pi.

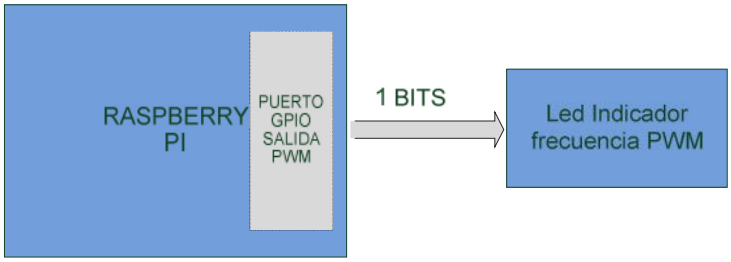
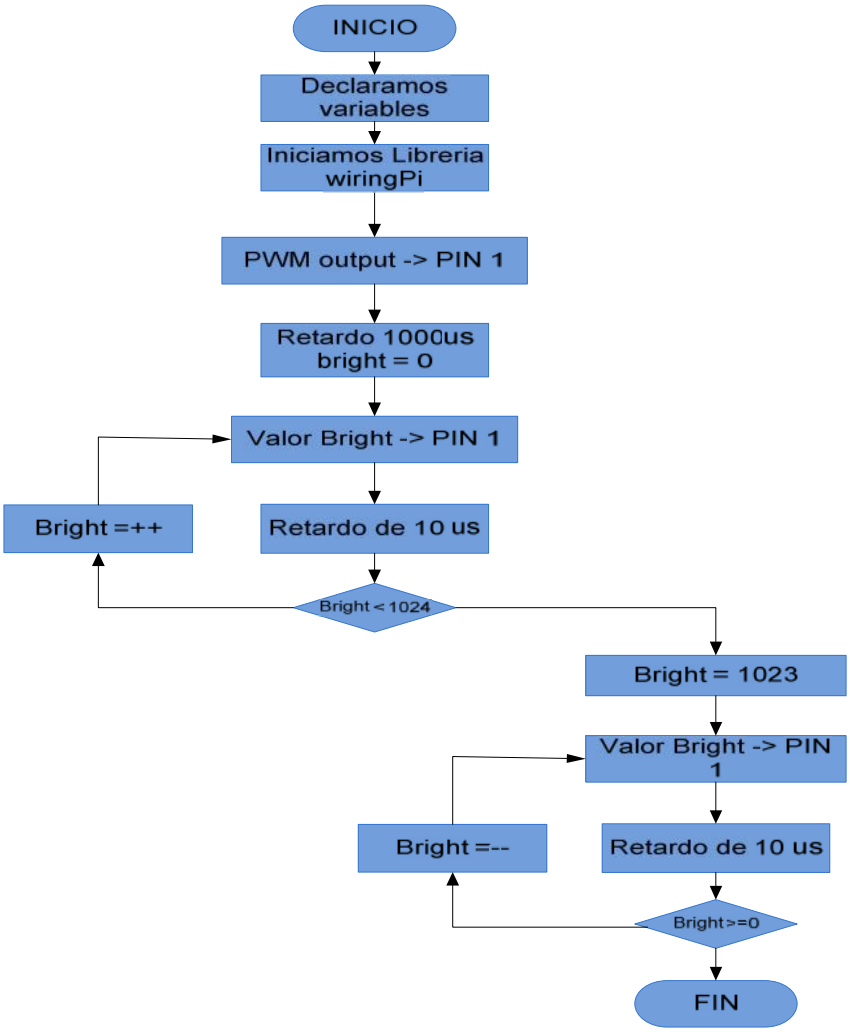



**Fig. 3.6 Diagrama bloque práctica #2**

## 3.13 DIAGRAMA DE FLUJO DEL PRÁCTICA II



## Documentación:

Declaramos nuestras variables e iniciamos la librería WiringPi, el PIN 1 del Raspberry Pi lo utilizaremos como nuestra salida PWM esperamos un tiempo de 1000us para comenzar nuestro proceso de on/off en el led. Establecemos nuestro variable bright =0 la cual usaremos como la frecuencia, la cual se incrementará periódicamente causando el aumento en la intensidad del led, luego iniciará el proceso de decremento en la frecuencia esto causa un efecto de disminución de intensidad del led.

## 3.14 CÓDIGO FUENTE DEL PRÁCTICA II: USO DEL PWM

```
#include <wiringPi.h>
#include <stdio.h>
#include<stdlib.h>
#include <stdint.h>
voidimprimir(void);
int main (void)
{
int bright ;
if (wiringPiSetup () == -1)   //IniciamosLibreriawiringPi
exit (1) ;

pinMode (1, PWM_OUTPUT) ; //pin 18 declarado como salida PWM
delay(1000);//retardo de 1000 us
imprimir(); //Funcion para imprimir presentacion
printf("\t\t\t     Incrementado Intensidad LED \n");
for (bright = 0 ; bright< 1024 ; ++bright) //lazo para realizar el incremento
en el encendido led
        {
```

```
        pwmWrite (1, bright) ;//usa pin 1 para realizar el incremento en la
intensidad
        delay (10) ;//retardo de 10us
          }
imprimir();
printf("\t\t\t\tBajando Intensidad LED\n");
for (bright = 1023 ; bright>= 0 ; --bright)//lazo para realizar el decremento
en el encendido del led
          {
        pwmWrite (1, bright) ; //usa pin 1 para realizar el decremento en la
intensidad
        delay (10) ; //retardo de 10us
          }

return 0 ;
}

void imprimir(void)  //Función para imprimir presentación
{
system ("/usr/bin/clear");
printf("\n\n\n\n\t\t\t************************************\n");
printf("\t\t\t**      Programa PWM en PIN 1     **\n");
printf("\t\t\t************************************\n");
}
```

# CAPÍTULO 4

## 3. PRUEBAS Y SIMULACIÓN DEL PROYECTO

### 4.1 PROYECTO: PRUEBA DE LA INTERFAZ PARA MEDICIÓN DE PARÁMETROS DEL PANEL SOLAR CON EL RASPBERRY PI

El panel solar es expuesto al sol durante un día para obtener los valores de voltaje, corriente y potencia y graficarlos en la curva característica de I – V.

**Figura 4.1 Implementación del panel solar con el Raspberry PI.**

En la figura 4.1 mostramos la implementación final de nuestro proyecto, donde consta nuestra minicomputadora Raspberry PI y el PIC16F887.

### 4.1.1 SIMULACIÓN DEL PROYECTO: FUNCIONAMIENTO PIC16F887

Como se mencionó en capítulos anteriores no se puede conectar directamente el panel solar al puerto GPIO sino que debe ser por medio de un ADC, por lo tanto simulamos el funcionamiento del mismo en el programa Proteus antes de grabarlo en el PIC16F887.

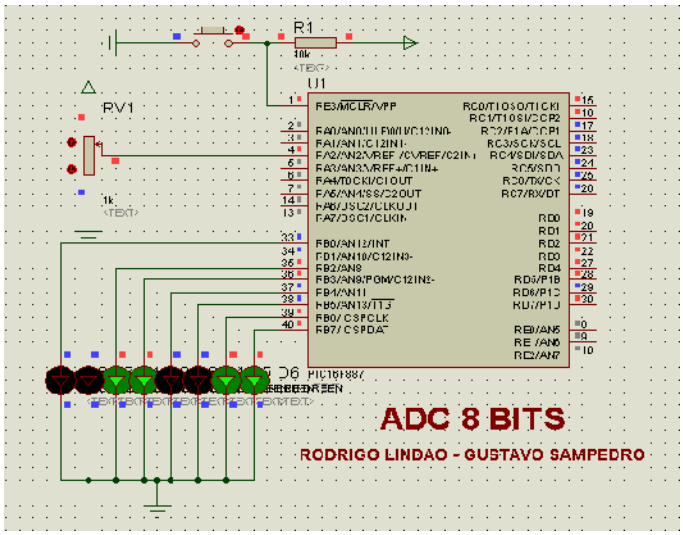



**Fig. 4.2 Simulación en Programa Proteus Convertidor ADC de 8 bits**

**Fuente: Realizado por Rodrigo Lindao Suárez**

En la figura 4.2 tenemos el PIC 16F887 donde hemos cargado el programa, el pulsador que está en la entrada 1 es el MCRL, en la entrada 4 hemos conectado un potenciómetro para simular la entrada analógica, y

para cada valor de voltaje que llegue será reflejado en los leds conectados a las salidas digitales que van desde la 33 siendo el bit menos significativo hasta la 40 que será el bit más significativo.

### 4.1.2 SIMULACIÓN DEL PROYECTO CON EL RASPBERRY PI:

Ingresamos al Raspberry PI después de compilar el programa lo ejecutamos, en la pantalla visualizamos el nombre del proyecto, los nombres de los integrantes y los parámetros del panel solar como se muestra en la figura a continuación.

**Fig. 4.3 Programa ejecutado desde el Raspberry Pi**
**Fuente: Realizado por Rodrigo Lindao Suárez**

## 4.2 SIMULACIÓN DEL PRÁCTICA I : ROTACIÓN DE LEDS

Para la mejor comprensión de la minicomputadora y sus funciones se realizaron dos prácticas como se lo menciono en el capítulo anterior, en esta práctica realizamos una interfaz para que se encienda unledy después de un tiempo se apague y encienda el siguiente led y así sucesivamente.

**Fig. 4.4 Programa ejecutado desde el Raspberry Pi muestra pin a encenderse**

**Fuente: Realizado por Rodrigo Lindao Suárez**

En la siguientes graficas se observa la ejecución del programa, observamos el encendido de los led's, la señal para el encendido de cada led proviene de los pines del puerto GPIO de nuestra mini computadora; esto nos muestra claramente una de las aplicaciones del puerto.

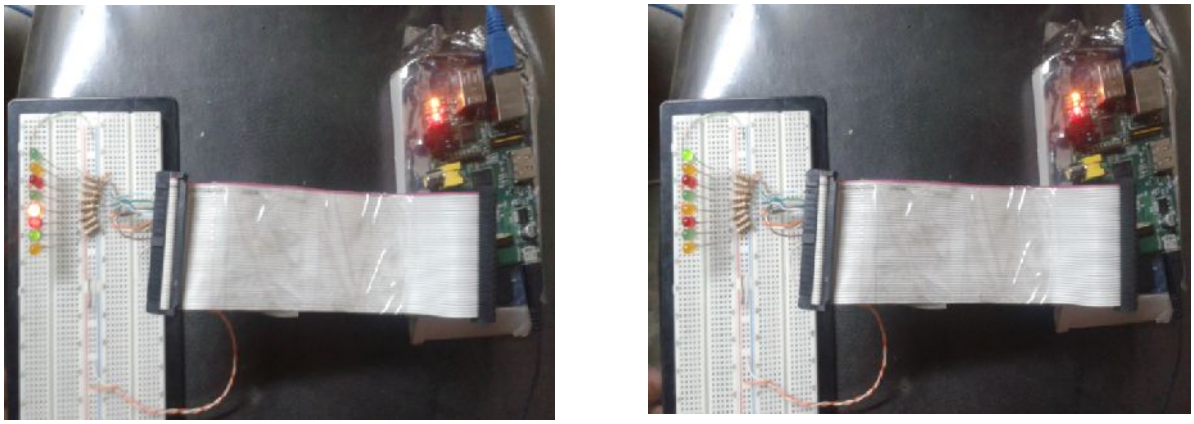


**Fig. 4.5 Implementación Práctica I se muestra durante la ejecución del programa**
**Fuente: Realizado por Rodrigo Lindao Suárez**

## 4.3 SIMULACIÓN DEL PRÁCTICA II : USO DEL PWM

Con nuestro ejemplo mostramos el uso de la función PWM la cual es una técnica usada para simular una salida analógica con una salida digital en nuestro Raspberry Pi en el pin 1 mediante el incremento y decremento de la intensidad de un led.

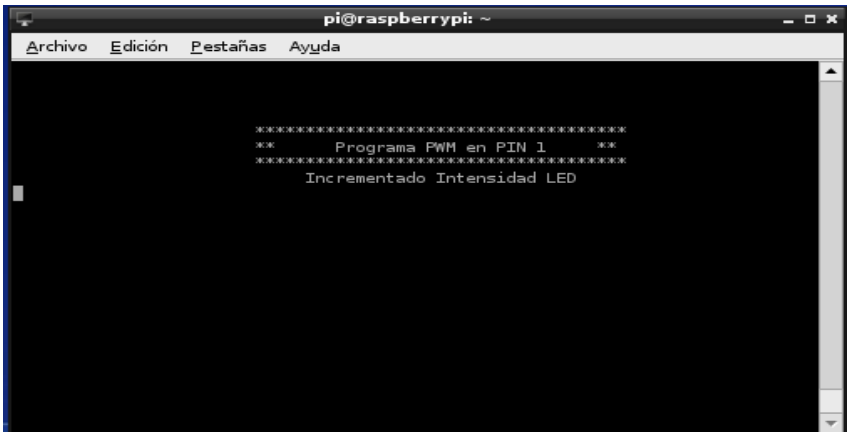



**Fig. 4.6 Programa ejecutado desde el Raspberry Pi muestra el incremento de la intensidad**
**Fuente: Realizado por Rodrigo Lindao Suárez**

En la figura 4.3 observamos la ejecución del programa en la minicomputadora en que nos indica la intensidad del led está incrementando y en la figura 4.4 se muestra la conexión que se realizó para la ejecución del mismo.

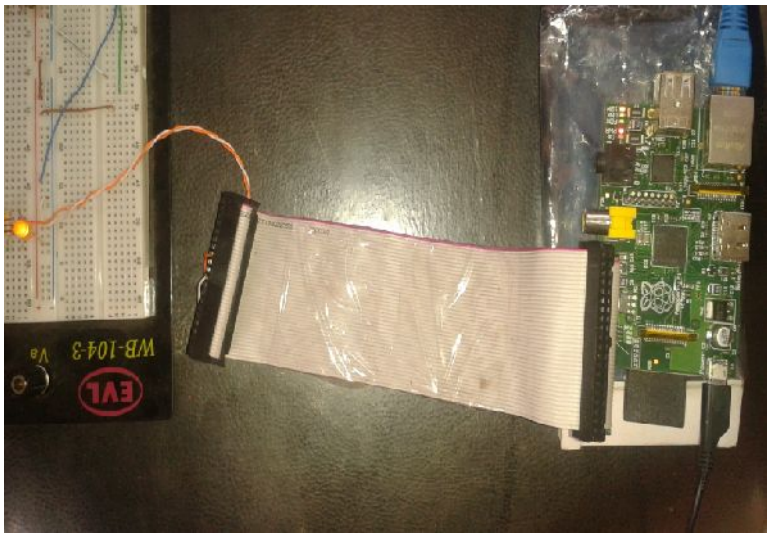


**Fig. 4.7 Implementación Práctica II muestra el incremento en la intensidad**
**Fuente: Realizado por Rodrigo Lindao Suárez**

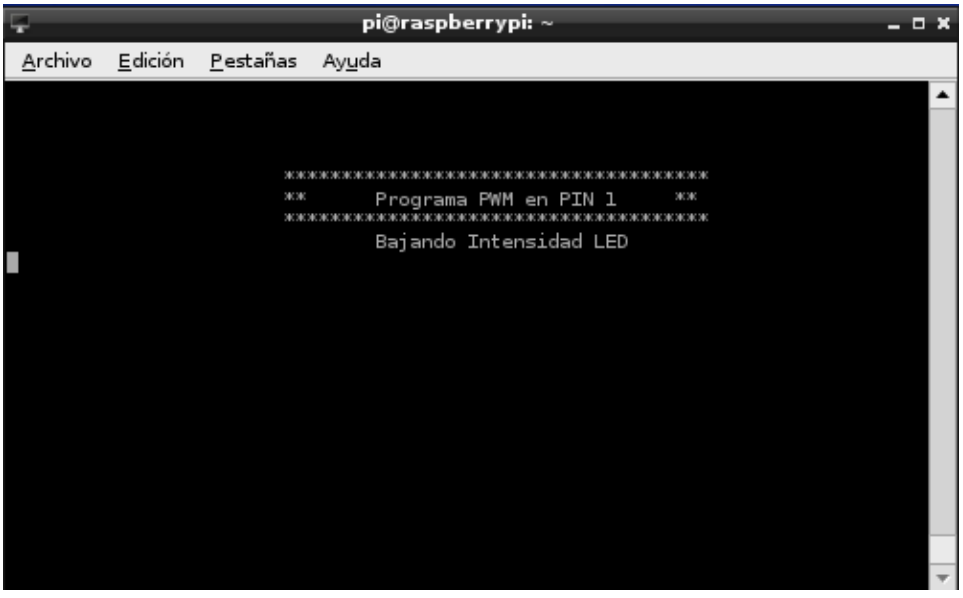



**Fig. 4.8 Programa ejecutado desde el Raspberry Pi muestra el decremento de la intensidad**
**Fuente: Realizado por Rodrigo Lindao Suárez**

En las siguientes figuras 4.5 y 4.6 vemos la ejecución del programa para el decremento de la intensidad del led y la ejecución del mismo.

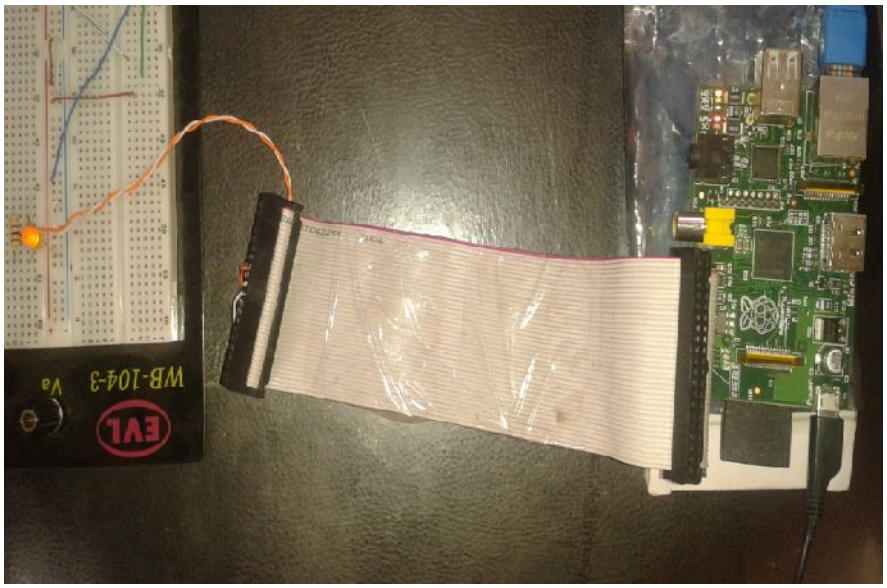


**Fig. 4.9 Programa ejecutado desde el Raspberry Pi muestra el decremento de la intensidad**

**Fuente: Realizado por Rodrigo Lindao Suárez**

## 4.4 PRUEBA EXPERIMENTAL PARA EL MODELO DE DUFFIE & BECKMAN PARA NUESTRO SISTEMA.

Con el objetivo de realizar un correcto muestreo en nuestro sistema se procedió a establecer pruebas de cortocircuito y circuito abierto para establecer parámetros necesarios en el modelo propuesto, realizando mediciones de Voltaje y Corriente durante un día soleado, cuyo propósito era obtener variables en nuestra ecuación que se especifican en la Hoja de Datos del Fabricante del Panel Solar.

Dónde:

m= 18 Número de células fotovoltaicas del panel solar.

n= 1 factor de idealidad del diodo.

K= $\mathbf{1.38 * 10^{-23}}[\boldsymbol{J/^oK}]$Constante de Boltzmann

q= $\mathbf{1.6 * 10^{-19}}[\boldsymbol{C}]$ Carga del electrón

T= 298 [ºK]  Temperatura del panel solar en condiciones normales.

$$V_{tpanel} = \frac{mnKT}{q}$$

Voc = 8[V] Voltaje de Circuito Abierto.

Isc = 0.6 [A] Corriente Cortocircuito.

V = Voltaje de Entrada Panel Solar.

I = Corriente de Entrada Panel Solar.

$$I = I_{SC} * \left[1 - e^{\frac{V-Voc}{Vt_{panel}}}\right]$$

Para nuestro análisis tomamos muestras aleatorias la cuales fueron de mucha  ayuda al momento de modelar las curvas que caracterizan a un Panel Solar que son mostradas en las Hojas de Datos de fabricante.

| Voltage | Corriente | Potencia |
|---|---|---|
| 0 | 0,599999981 | 0 |
| 0,25 | 0,599999968 | 0,149999992 |
| 0,57 | 0,599999936 | 0,341999963 |
| 0,75 | 0,599999905 | 0,449999929 |
| 1,04 | 0,599999823 | 0,623999815 |

| | | |
|---|---|---|
| 1,25 | 0,599999721 | 0,749999651 |
| 1,58 | 0,59999943 | 0,9479991 |
| 1,77 | 0,599999141 | 1,06199848 |
| 2,12 | 0,599998171 | 1,271996122 |
| 2,25 | 0,599997578 | 1,34999455 |
| 2,57 | 0,599995165 | 1,541987574 |
| 2,75 | 0,599992867 | 1,649980385 |
| 3,11 | 0,599984478 | 1,865951725 |
| 3,25 | 0,599978997 | 1,949931739 |
| 3,53 | 0,599961545 | 2,117864254 |
| 3,75 | 0,599938152 | 2,249768069 |
| 4,09 | 0,599871094 | 2,453472774 |
| 4,25 | 0,599817877 | 2,549225975 |
| 4,58 | 0,599628524 | 2,746298639 |
| 4,75 | 0,599463705 | 2,847452597 |
| 5,13 | 0,598781367 | 3,071748413 |
| 5,25 | 0,598420782 | 3,141709107 |
| 5,57 | 0,596847708 | 3,324441736 |
| 5,75 | 0,59534971 | 3,423260831 |
| 6,07 | 0,590717512 | 3,585655296 |
| 6,25 | 0,586306385 | 3,664414907 |
| 6,57 | 0,572666046 | 3,762415924 |
| 6,75 | 0,559676692 | 3,777817673 |
| 7,04 | 0,524560602 | 3,69290664 |
| 7,25 | 0,481260781 | 3,489140659 |
| 7,47 | 0,409027459 | 3,055435116 |
| 7,75 | 0,250351049 | 1,940220626 |
| 8 | 0 | 0 |

**Tabla 4.1 Datos de voltaje, corriente y potencia del Panel Solar**

## 4.5 CURVAS CARACTERÍSTICAS DEL PANEL SOLAR

Puede observarse que el valor máximo para el voltaje de salida corresponde a un valor de corriente nulo (voltaje a circuito abierto), mientras que el valor máximo para la corriente corresponde a un voltaje de salida nulo (salida cortocircuitada). Todas las curvas tienen una zona donde el valor de la corriente permanece prácticamente constante para valores crecientes del voltaje de salida, hasta que alcanzan una zona de transición.

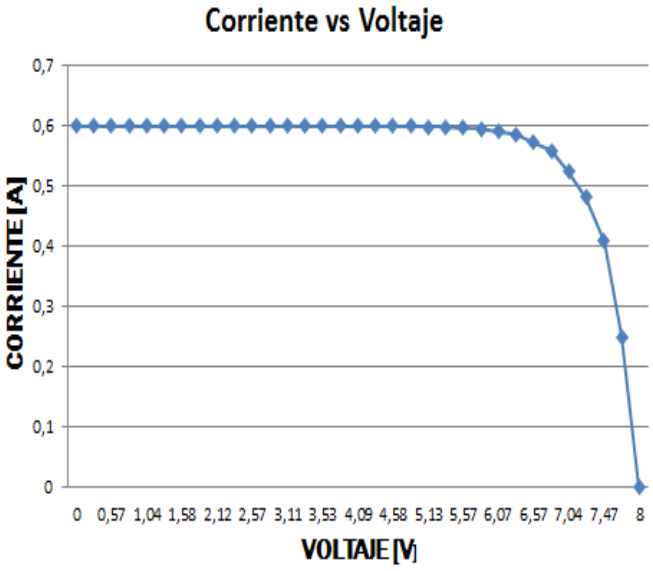



**Fig. 4.10 Curva características I-V resultante de los datos obtenidos.**

A partir de esta zona, pequeños aumentos en el voltaje de salida ocasionan bruscas disminuciones en el valor de la corriente de salida. El comienzo de la zona de transición se alcanza para menores valores del voltaje de salida cuando la temperatura de trabajo se incrementa.

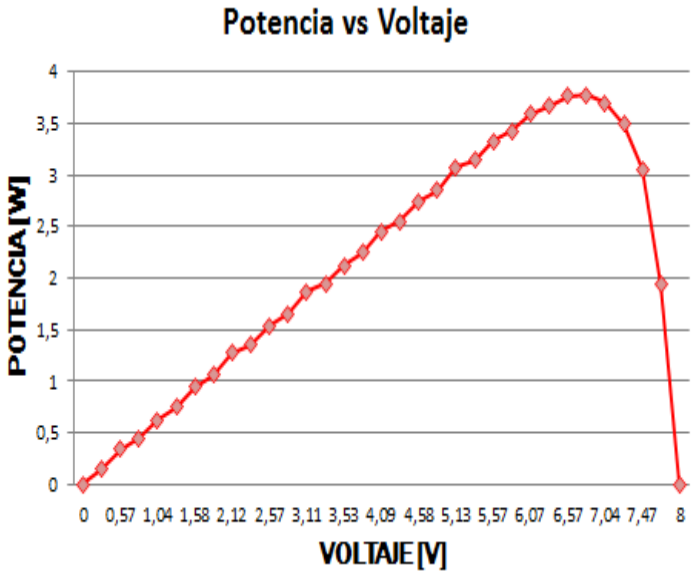



**Fig. 4.11 Grafica de Potencia vs Voltaje resultante de los datos obtenidos.**

## 4.6 MEDICIÓN DE PARÁMETROS DE UN PANEL SOLAR.

Muestras tomadas el Día Viernes, 12 de Julio /2013 con nuestro sistema de adquisición de datos desde las 10:10 a.m. hasta las 18:40 p.m. con un intervalo de captura de 5 minutos el propósito de esta actividad es notar el comportamiento durante un Día del voltaje, corriente y potencia producida por el panel solar, parámetros indispensable para realizar la evaluación del panel solar.

| Hora | Voltaje | Corriente | Potencia |
|---|---|---|---|
| 10:10 | 7,45 | 0,417101811 | 3,10740849 |
| 10:15 | 7,48 | 0,404857579 | 3,028334692 |
| 10:20 | 7,53 | 0,382602026 | 2,880993256 |
| 10:25 | 7,51 | 0,39179365 | 2,942370312 |
| 10:30 | 7,4 | 0,435825538 | 3,225108981 |
| 10:35 | 7,23 | 0,486281095 | 3,515812319 |
| 10:40 | 7,52 | 0,387247471 | 2,912100981 |

| | | | |
|---|---|---|---|
| 10:45 | 7,6 | 0,347116311 | 2,638083965 |
| 10:50 | 7,54 | 0,377855148 | 2,849027817 |
| 10:55 | 7,57 | 0,362983525 | 2,747785281 |
| 11:00 | 7,5 | 0,396242685 | 2,971820135 |
| 11:05 | 7,42 | 0,428577777 | 3,180047103 |
| 11:10 | 7,41 | 0,432240794 | 3,202904282 |
| 11:15 | 7,5 | 0,396242685 | 2,971820135 |
| 11:20 | 7,62 | 0,33595233 | 2,559956754 |
| 11:25 | 7,64 | 0,324295496 | 2,477617588 |
| 11:30 | 7,61 | 0,341594604 | 2,599534934 |
| 11:35 | 7,58 | 0,357808277 | 2,712186742 |
| 11:40 | 7,62 | 0,33595233 | 2,559956754 |
| 11:45 | 7,56 | 0,368048185 | 2,78244428 |
| 11:50 | 7,5 | 0,396242685 | 2,971820135 |
| 11:55 | 7,48 | 0,404857579 | 3,028334692 |
| 12:00 | 7,44 | 0,421010051 | 3,132314777 |
| 12:05 | 7,46 | 0,413108234 | 3,081787429 |
| 12:10 | 7,43 | 0,424834778 | 3,156522399 |
| 12:15 | 7,4 | 0,435825538 | 3,225108981 |
| 12:20 | 7,44 | 0,421010051 | 3,132314777 |
| 12:25 | 7,44 | 0,421010051 | 3,132314777 |
| 12:30 | 7,48 | 0,404857579 | 3,028334692 |
| 12:35 | 7,51 | 0,39179365 | 2,942370312 |
| 12:40 | 7,47 | 0,409027459 | 3,055435116 |
| 12:45 | 7,33 | 0,458863197 | 3,363467233 |
| 12:50 | 7,17 | 0,500103999 | 3,585745673 |
| 12:55 | 7,23 | 0,486281095 | 3,515812319 |
| 13:00 | 7,28 | 0,473311711 | 3,445709253 |
| 13:05 | 7,35 | 0,452632472 | 3,326848671 |
| 13:10 | 7,7 | 0,286145452 | 2,203319981 |
| 13:15 | 7,57 | 0,362983525 | 2,747785281 |
| 13:20 | 7,52 | 0,387247471 | 2,912100981 |
| 13:25 | 7,51 | 0,39179365 | 2,942370312 |
| 13:30 | 7,41 | 0,432240794 | 3,202904282 |
| 13:35 | 7,3 | 0,46771884 | 3,414347534 |

| | | | |
|---|---|---|---|
| 13:40 | 7,71 | 0,279292448 | 2,153344773 |
| 13:45 | 7,73 | 0,265134267 | 2,049487882 |
| 13:50 | 7,04 | 0,524560602 | 3,69290664 |
| 13:55 | 7,09 | 0,515956909 | 3,658134485 |
| 14:00 | 7 | 0,530804927 | 3,715634492 |
| 14:05 | 7,2 | 0,4934164 | 3,552598079 |
| 14:10 | 7,37 | 0,446126682 | 3,287953644 |
| 14:15 | 7,11 | 0,512246684 | 3,64207392 |
| 14:20 | 7,2 | 0,4934164 | 3,552598079 |
| 14:25 | 7,61 | 0,341594604 | 2,599534934 |
| 14:30 | 7,67 | 0,305838289 | 2,345779673 |
| 14:35 | 7,1 | 0,514121831 | 3,650264998 |
| 14:40 | 6,92 | 0,54178578 | 3,749157597 |
| 14:45 | 7 | 0,530804927 | 3,715634492 |
| 14:50 | 7,55 | 0,373004622 | 2,816184898 |
| 14:55 | 7,08 | 0,517752775 | 3,665689644 |
| 15:00 | 7,65 | 0,318275497 | 2,434807549 |
| 15:05 | 7,49 | 0,400596651 | 3,000468912 |
| 15:10 | 7,4 | 0,435825538 | 3,225108981 |
| 15:15 | 6,78 | 0,556977224 | 3,776305577 |
| 15:20 | 7,46 | 0,413108234 | 3,081787429 |
| 15:25 | 7,43 | 0,424834778 | 3,156522399 |
| 15:30 | 7,44 | 0,421010051 | 3,132314777 |
| 15:35 | 7,38 | 0,442766863 | 3,267619446 |
| 15:40 | 7,32 | 0,461879063 | 3,380954742 |
| 15:45 | 7,25 | 0,481260781 | 3,489140659 |
| 15:50 | 7,26 | 0,478668113 | 3,475130503 |
| 15:55 | 7,19 | 0,49569392 | 3,564039284 |
| 16:00 | 7,44 | 0,421010051 | 3,132314777 |
| 16:05 | 7,38 | 0,442766863 | 3,267619446 |
| 16:10 | 7,37 | 0,446126682 | 3,287953644 |
| 16:15 | 7,35 | 0,452632472 | 3,326848671 |
| 16:20 | 7,38 | 0,442766863 | 3,267619446 |
| 16:25 | 7,38 | 0,442766863 | 3,267619446 |
| 16:30 | 7,3 | 0,46771884 | 3,414347534 |

| | | | |
|---|---|---|---|
| 16:35 | 7,35 | 0,452632472 | 3,326848671 |
| 16:40 | 7,05 | 0,522913385 | 3,686539367 |
| 16:45 | 6,89 | 0,545438437 | 3,758070833 |
| 16:50 | 6,88 | 0,54660433 | 3,760637791 |
| 16:55 | 6,74 | 0,560538337 | 3,778028388 |
| 17:00 | 6,7 | 0,563804686 | 3,777491398 |
| 17:05 | 6,58 | 0,57206921 | 3,764215404 |
| 17:10 | 6,37 | 0,582254541 | 3,708961426 |
| 17:15 | 6,41 | 0,580653149 | 3,721986687 |
| 17:20 | 6,46 | 0,578446685 | 3,736765582 |
| 17:25 | 6,48 | 0,577495177 | 3,742168745 |
| 17:30 | 6,14 | 0,589202338 | 3,617702354 |
| 17:35 | 5,66 | 0,596171283 | 3,374329464 |
| 17:40 | 5,51 | 0,59723088 | 3,290742148 |
| 17:45 | 5,33 | 0,598122897 | 3,187995041 |
| 17:50 | 4,77 | 0,599440029 | 2,859328938 |
| 17:55 | 3,98 | 0,599898355 | 2,387595453 |
| 18:00 | 3,02 | 0,59998722 | 1,811961404 |
| 18:05 | 2,17 | 0,599997962 | 1,301995578 |
| 18:10 | 1,58 | 0,59999943 | 0,9479991 |
| 18:15 | 1,46 | 0,59999956 | 0,875999358 |
| 18:20 | 1,25 | 0,599999721 | 0,749999651 |
| 18:25 | 0,7 | 0,599999915 | 0,41999994 |
| 18:30 | 0,33 | 0,599999962 | 0,197999987 |
| 18:35 | 0,18 | 0,599999972 | 0,107999995 |
| 18:40 | 0,05 | 0,599999979 | 0,029999999 |

**Tabla 4.2 Datos de Prueba del Panel Solar**

## 4.7 GRÁFICO RESULTANTE DE LA ADQUISICIÓN DE DATOS

La etapa de mayor generación de potencia en nuestro panel solar se da durante las 13:00 p.m. y las 17:00 p.m. en donde notamos la mayor incidencia de la luz solar.

En la gráfica 4.12 notamos que el voltaje no varía abruptamente, y se mantiene en gran parte del día y a medida que la intensidad del sol disminuye el voltaje empieza a decaer, debido a que el material del que está compuesto el panel solar no libera la de energía que produce la electricidad.

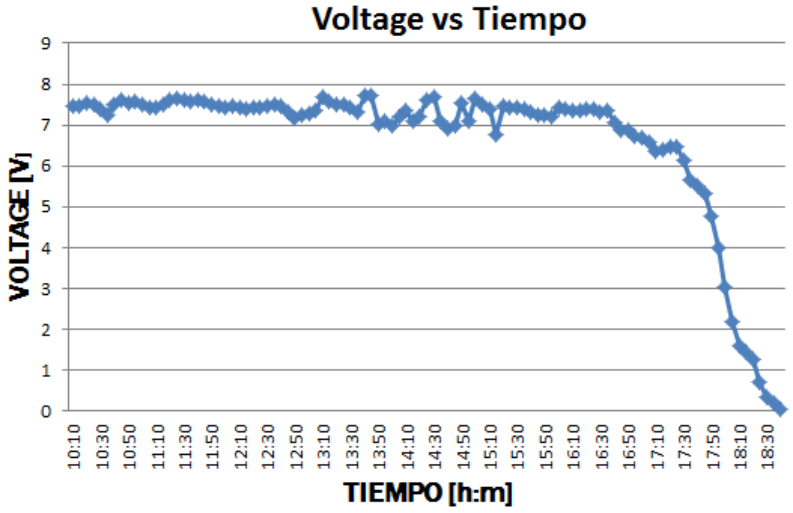



**Fig. 4.12 Gráfica de Voltaje vs Tiempo de los datos obtenidos.**

En la gráfica 4.13 que es de la corriente vs el tiempo observamos que es más inestable con respecto a la incidencia solar durante el día, al llegar el

atardecerla intensidad del sol disminuye pero la corriente se mantiene constante alcanzando la corriente de cortocircuito del panel solar.

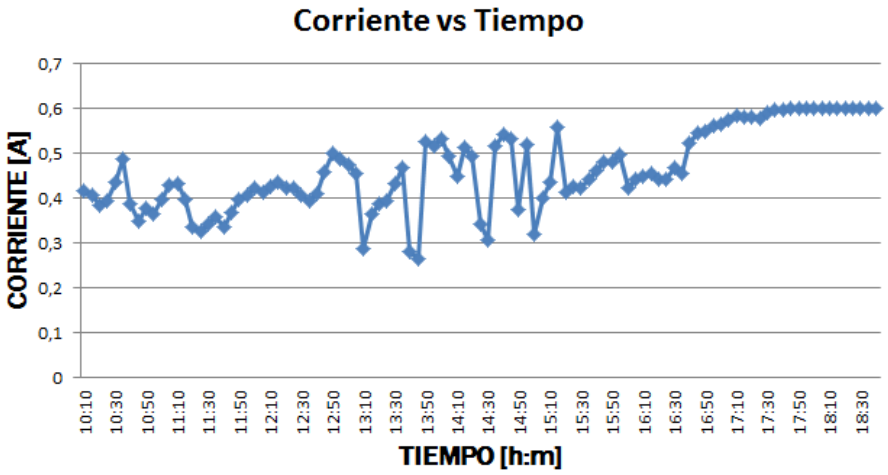



**Fig. 4.13 Gráfica de Corriente vs Tiempo de los datos obtenidos.**

En la gráfica 4.14 de potencia vs tiempo observamos que la curva es la multiplicación entre los valores de voltaje y corriente, pero a medida que la intensidad del solar disminuye la potencia disminuye, debido al producto de las dos variables.

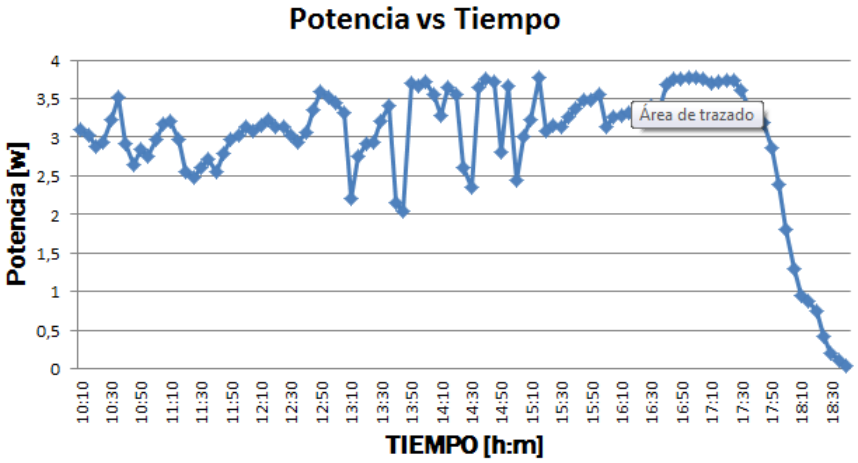



**Fig. 4.14 Grafica de Potencia vs Tiempo de los datos obtenidos.**

# CONCLUSIONES

1. El panel solar es una herramienta fundamental para la generación de energía eléctrica a partir de la luz solar, de nuestras experiencias obtuvimos valores de voltaje, corriente y potencia que en promedio son de 7V, 0.5 Amp. y 3Watts que debidamente manipulados pueden constituirse en una ayuda fundamental para aquellas viviendas que se encuentren alejadas de las redes eléctricas convencionales, si se necesita mayor potencia se deben aumentar el número de placas en el panel solar.

2. La cantidad de energía eléctrica producida por el panel solar depende de por cuantas celdas este constituida, ya que a mayor número de celdas solares mayor será mi promedio de voltaje, corriente y potencia.

3. El Raspberry proporciona funciones fundamentales en cuanto a la plataforma LINUX basándose en el lenguaje de programación C, este posee librerías útiles para realizar operaciones matemáticas complejas usadas en la modelación de la ecuación de corriente, librerías de archivos necesarias para guardar la información del usuario y librerías de tiempo útil para conocer el momento exacto de la toma de muestras.

4. Los paneles solares son importantes porque al generar electricidad fotovoltaica, creamos nuevas alternativas de generación de energía; además, la utilización de paneles solares nos ayuda a conservar el medio ambiente, debido a que impedimos la contaminación producida porruido y la quema de combustibles fósiles contribuyendo así a la reducción del cambio climático y el efecto invernadero.

5. De las pruebas realizadas con el panel solar observamos que tiene una fuerza electromotriz que depende de la intensidad de la radiación que reciba, hay que considerar que en nuestro país, a medio día sobrepasa fácilmente el nivel de radiación de los $1000W/m^2$ por lo tanto, podremos obtener mayor potencia a la especificada en la placa, esta característica debe ser mayormente explotada.

6. Para realizar pruebas con otros paneles solares de diferentes potencias, se debe medir la corriente de corto circuito y el voltaje de circuito abierto, debido a que estos datos son necesarios para la obtención de los parámetros que serán mostrados en el RaspberryPimediante el uso de la ecuación de Duffie&Beckman, adicionalmente debemos calcular las resistencias para el divisor de voltaje que se encuentra en la entrada análoga del PIC, esto se realiza para protección del mismo, pues solo puede recibir hasta 5V máximo.

# RECOMENDACIONES

1. El archivo utilizado para almacenar los parámetros necesarios para nuestro análisis debe estar ajustado de manera tal que las muestras sean tomadas cada inicio de un minuto y capaz de ser configurada por el usuario para realizar esta operación el programador debe ser capaz de analizar el arreglo de caracteres donde guardamos la hora local para lograr el efecto requerido.

2. Es necesario analizar y descargar los paquetes necesarios para nuestro sistema operativo Linux que nos facilitarán la interacción entre nuestro software y hardware del Raspberry Pi, manipular ciertas funciones que nos permiten establecer la comunicación entre el PIC 16f887 y el GPIO.

3. Los programa en Lenguaje C utilizados en esta tesina deben ser sujetas a pruebas antes de ser implementadas en el programa principal, algunas pueden causar resultados inesperados, el

programador debe tener claro el sistema de archivo de Linux para establecer el lugar exacto donde nos guarda la información requerida, el correcto uso de las librerías, contantes y variable universales para facilitar el uso a usuarios.

4. Raspberry Pi utiliza voltaje de entrada menor a 3.3 V, mientras nuestro PIC 16F887 utiliza 5V esto nos puede ocasionar daños en nuestra placa para evitar estos inconvenientes usamos resistencias a la salida de los pines del PIC, es suma importancia el regulador dc a dc usado a la entrada del panel solar para evitar las caída de voltajes.

5. Se aconseja revisar los tipos de compilaciones que existen para lenguaje C en LINUX al momento de compilar la librería para WiringPi no detecto la librería math para realizar la operación potencia pow(a,b) se necesita agregar un parámetro adicional para lograr compilar ambas librerías en un solo programa (gcc -o Proyecto **–lm**Proyecto.c -L/usr/local/lib –lwiringPi).

# BIBLIOGRAFÍA

[1] Andrew K. Dennis, Raspberry PI Home AutomationwithArduino, Packt, fecha de consulta 13 de Mayo 2013.

[2] Msc. Yucra S. Daniel, Curso de LINUX. Disponible: www.somoslibres.org. Fecha de consulta 14 de Mayo 2013.

[3] TheMagPi. Disponible:http://www.themagpi.com
Fecha de consulta 16 de Mayo 2013.

[4] WiringPi. Disponible: https://projects.drogon.net/raspberry-pi/wiringpi/
Fecha de consulta 16 de Mayo 2013.

[5] Computing At School, Raspberry Pi Education Manual, Raspberry PI Fundation. Fecha de consulta 9 de Junio 2013.

[6] EbenUpton and GarethHalfacree, Raspberry Pi Userby Guide, Wiley, Inglaterra 2012. Fecha de consulta 12 de Junio 2013.

[7]Marti Antonio,Fundamentos, dimensionado y aplicaciones de la energía solar fotovoltaica**,** Centro de Investigaciones Energéticas, Medioambientales y Tecnológicas, CIEMAT.Fecha de consulta Junio 2013.

[8] J. Ma. Angulo, Microcontroladores PIC: Diseño práctico de Aplicaciones, McGraw Hill, 1999 Segunda Parte.Fecha de consulta Junio 2013.

[9] Fecha y Hora en C. Disponible:

http://www.elrincondelc.com/nuevorincon/index.php

http:// www.holamundo.es/lenguaje/c/articulos/fecha-hora-c.html.

Fecha de consulta Julio 2013

[10] Duffie, J. A. y Beckman, W. A. (1991), Solar Engineering of Thermal Processes, 2ª Edición. Wiley-Interscience, New York.

Fecha de consulta Julio 2013.

[11] Hnos. García Noblejas,Energía solar fotovoltaica y cooperación del desarrollo, © IEPALA Editorial, 1999 pág. 27-30

Fecha de consulta Junio 2013.

[12] Artesa Grupo Elecnor, Aplicaciones de la Energía Fotovoltaica:

http://www.atersa.com/quienes.asp?param=21

Fecha de consulta Julio 2013

[13] Energía Solar Fotovoltaica: http://www.electricidad- gratuita.com/

introducción _energia_fotovoltaica.htlm.

Fecha de consulta Junio 2013.