"Responsabilidad con pensamiento positivo"

UNIVERSIDAD TECNOLÓGICA ISRAEL

TRABAJO DE TITULACIÓN

CARRERA: Sistemas Informáticos

TEMA: "Diseño y construcción de un Sistema de control de luces para una casa utilizando la placa raspberry."

AUTOR: Diego Xavier Gudiño Chalá

TUTOR: Ing. Juan Carlos Moreno Carrillo

AÑO 2014

UNIVERSIDAD TECNOLÓGICA ISRAEL

FACULTAD DE SISTEMAS INFORMÁTICOS

CERTIFICADO DE RESPONSABILIDAD

Yo, Ing. Juan Carlos Moreno Carrillo, certifico que el señor Diego Xavier Gudiño Chalá con C.I. 1713460036 realizó la presente tesis con el título "Diseño y construcción de un Sistema de control de luces para una casa utilizando la placa raspberry", y que es autor intelectual del mismo, que es original, auténtico y personal.

_____

Ing. Juan Carlos Moreno Carrillo

UNIVERSIDAD TECNOLÓGICA ISRAEL

FACULTAD DE SISTEMAS INFORMÁTICOS

ACTA DE CESIÓN DE DERECHOS

Yo, DIEGO XAVIER GUDIÑO CHALÁ, declaro conocer y aceptar la disposición de la Normativa de la Universidad Tecnológica Israel que en su parte pertinente textualmente dice: “Forma parte del Patrimonio de la Universidad la propiedad intelectual de las investigaciones, trabajos científicos o técnicos y tesis de grado que se realicen a través, o con el apoyo financiero, académico o institucional (operativo) de la Universidad”.

_____

Diego Xavier Gudiño Chalá

C.I. 1713460036

UNIVERSIDAD TECNOLÓGICA ISRAEL

FACULTAD DE SISTEMAS INFORMÁTICOS

CERTIFICADO DE AUTORÍA

El documento de tesis con título “Diseño y construcción de un Sistema de control de luces para una casa utilizando la placa raspberry”, ha sido desarrollado por Diego Xavier Gudiño Chalá con C.I. 1713460036, persona que posee los derechos de autoría y responsabilidad, restringiéndose la copia o utilización de cada uno de los productos de esta tesis sin previa autorización.

_____

Diego Xavier Gudiño Chalá

Dedicatoria

Dedico a este logro a Dios, por darme las fuerzas y el tiempo para dar este gran paso, a mi madre por su incondicional apoyo sus consejos y el cariño, eso me ayudado hacer los que soy hoy en día, y en general a todo mi familia que los llevo en el corazón.

Agradecimiento

Sería una lista muy larga para enumerar a todas las personas que me han ayudado en este proceso, en especial a Dios, por permitirme ser firme y no decaer, después de una largo tiempo fuera del país, regresar a retomar mis estudios fue muy duro, pero la vida está llena de retos y lo afronte con ayuda de mi familia amigos, y me lance nuevamente en la aventura de los estudios, ya que día a día aprendí nuevas cosas, agradezco a mis profesores que en este periodo me han brindado sus conocimientos y amistad.

Contenido

CAPÍTULO 1 1
1. PLANTEAMIENTO DEL PROBLEMA 1
1.1. Introducción 1
1.2. Delimitación del problema de investigación 2
1.2. OBJETIVOS 2
1.2.1. Objetivo principal: 2
1.2.2. Objetivos secundarios: 3
1.3. JUSTIFICACIÓN DE LA INVESTIGACIÓN 3
1.3.1. ¿Para qué sirve el trabajo de graduación? 3
1.3.2. ¿Cuál es la relevancia técnica? 3
1.3.3. ¿Ayudará a resolver algún problema práctico? 3
1.3.4. El tema es de actualidad 3
CAPÍTULO 2 4
2. PRESENTACION 4
2.1. Introducción 4
2.2. Domótica 5
2.3. Teleasistencia domótica 6
2.4. Teleasistencia social 7
2.5. Teleasistencia médica 7
2.6. Placa de control Raspberri PI 7
2.7. PIFACE 11
2.7.1. Características 11
CAPÍTULO 3 13
METODOLOGÍA DE DESARROLLO 13
3.1. Metodología 13
3.1.1. Indución 13
3.1.2 Deducción. 13
3.1.3. Análisis 13
3.1.4. Síntesis 13
3.2. Técnica de Investigación: 13

3.2.1. Observación 13
3.3. Entrevista 14
3.3.1. Calculo de la muestra 14
3.3.2 Tamaño de la muestra. 14
3.3.3 Modelo de la entrevista 15
3.3.4 Resultados de la entrevista 16
CAPÍTULO 4 18
4.1. DESARROLLO 18
4.1.1. Sistema informático 18
4.1.2. Placa de Interacción 20
4.1.3. Placa de poder 21
4.1.4. Pruebas realizadas , Programacion 22

CONCLUSIONES, RECOMENDACIONES Y BIBLIOGRAFÍA 24
5.1. CONCLUSIONES 24
5.2 RECOMENDACIONES 24
5.3 BIBLIOGRAFÍA 25
ANEXOS 26

ÍNDICE DE GRÁFICOS

Pág.

Gráfico1. Domótica ……5

Gráfico 2. Teleasistencia ……6

Gráfico 3. Placa raspberry PI ……..7

Gráfico 4. Placa PIFACE ……..12

Gráfico 5. Estadísticas ……..16

Gráfico 6. Estadísticas ……..16

Gráfico 7. Estadísticas ……...17

Gráfico 8. Python ……19

Gráfico 9. Diagrama de programación …… 19

Gráfico 10. Interfaz de usuario ……..20

Gráfico 11. PiFace ……..21

Gráfico 12. Maqueta terminada ……..22

Gráfico 13. Pruebas de realizadas ……23

CAPÍTULO 1

1.- PLANTEAMIENTO DEL PROBLEMA

1.1 Introducción

Para mejorar el estilo de vida, se ha utilizado diferentes tecnologías, como la activación de luces por medio de mensajes de texto, vía internet, controles infrarrojos, uso del bluetooth, entre otros. Donde todas estas tienen algo en común que es el uso de demasiados recursos y de espacio.

Para minimizar el costo de estos recursos se propone utilizar la placa raspberry PI, ya que tiene muchas ventajas como su tamaño e innumerables prestaciones.

La raspberry PI la cual es de reducido tamaño de 12cm * 5cm es capaz de trabajar como un computador, la misma funciona a 5 Voltios donde permite almacenar un sistema operativo y realizar muchas aplicaciones.

En nuestro país según el censo del INEN del 2012 existe un 14% de personas de la tercera edad y un 5% de personas discapacitadas. El mantener y mejorar su calidad de vida incluye aspectos tales como facilitar su vida independiente, promover una mayor integración social, aumentar sus posibilidades de educación, formación y empleo, mejorar la disponibilidad y calidad de su cuidado y asistencia.

Debe producirse una evolución socio-cultural en la forma de atender las necesidades de este grupo.

En personas con problemas de diversidad funcional o personas mayores la domótica les brinda la opción de poder ir usando sus ayudas técnicas de forma progresiva, a medida de sus problemas de movilidad o autonomía, de esta forma, su vida podrá seguir de forma casi autónoma con estos apoyos.

En el caso de personas con discapacidad lo esencial que les ofrece la domótica son los sistemas de control de entorno cuya finalidad es controlar elementos del entorno doméstico y laboral. Una persona con movilidad reducida o con cualquier

otra discapacidad puede encender sus aparatos electrónicos bien por voz, bien por mandos a distancia, o pueden realizar llamadas de emergencia o activar sus servicios de alarma si lo necesitan.

Existen varios sistemas operativos que son compatibles con la placa raspberry PI, este sistema operativo es Raspbian la cual es libre, gratuito y está basado en Linux, el mismo que fue creado por un grupo de desarrolladores que son fanáticos de la raspberry PI.

Para realizar las aplicaciones para la placa raspberry PI uno de los principales que es utilizado por los desarrolladores es el Python, el cual es un lenguaje simple.

1.2 Definición del problema de la investigación.

La tecnología en el mundo crece a pasos agigantados, apareciendo nuevas tendencias para solucionar problemas cotidianos, con el único objetivo de facilitar las labores cotidianas del ser humano principalmente personas minusválidas y de la tercera edad.

Una de estas es la de controlar fácilmente las luces de una casa sin que esta demande mucha labor o esfuerzo.

La tecnología debe ser usada para facilitar el modo de vida de las personas con discapacidad y de la tercera edad en sus actividades diarias. Reduciendo el grado de dificultad en su modo de vida.

1.2. OBJETIVOS

1.2.1 Objetivo principal:

Desarrollo de un sistema de control de luces para una casa utilizando la placa raspberry PI, mediante un lenguaje de cuarta generación.

1.2.2 Objetivos secundarios:

1.- Construir la placa de control en base al modelo de TERAUSB2.0 de propiedad de la empresa Teratronic, en base a su diseño prefabricado del diagrama esquemático

2.- Elaborar una interfaz de programación en lenguaje java para controlar mediante software el control de luces.

3.- Diseñar y construir el circuito de luces que estará ubicada en la maqueta.

1.3. JUSTIFICACIÓN DE LA INVESTIGACIÓN

1.3.1 ¿Para qué sirve el trabajo de graduación?

El presente trabajo de investigación sirve para mejorar el estilo de vida de nuestra sociedad.

1.3.2 ¿Cuál es la relevancia técnica?

La relevancia técnica que en base a un prototipo, demostrara el uso de la domótica en los hogares.

1.3.3 ¿Ayudará a resolver algún problema práctico?

Totalmente le ayuda a resolver los problemas ya que con este sistema solo la persona tiene que tener acceso a la computadora y por medio de esta tiene el control de toda la casa para prender o apagar luces, entre otras cosas,

1.3.4 El tema es de actualidad

En la actualidad, la tecnología en el mundo crece a pasos agigantados, apareciendo nuevas tendencias para solucionar problemas cotidianos, con el uso de la tecnología inalámbrica.

CAPÍTULO 2

2.- PRESENTACION

2.1 INTRODUCCION

La casa inteligente ha dejado de ser una promesa futurista para convertirse en una realidad tangible: cada vez más a menudo, las personas están convirtiendo sus hogares en máquinas punteras gracias a la popularización de sistemas de automatización que no precisan una gran inversión ni grandes cambios en su casa o estilo de vida.

La esencia de la automatización del hogar se basa en un equipo especializado que puede controlar lámparas, electrodomésticos, el calentador y el aire acondicionado, e incluso detectar en qué lugar de la casa hay personas. Una vez que estos métodos están en su sitio, no supone un gran esfuerzo, como le mostrará este libro, que algunas órdenes se ejecuten automáticamente.

Domótica. Los mejores trucos le propone técnicas asequibles y reales para mejorar considerablemente la seguridad y bienestar de su hogar: en poco tiempo será capaz de desarrollar acciones tan diversas como el control de la iluminación, inspeccionar si hay actividad en alguna zona de la casa, instalar cámaras de vigilancia o conocer en todo momento qué está ocurriendo en su hogar a través de un sencillo navegador web.

El objetivo es presentar, de forma completa, el concepto de Hogar Digital a aquellos lectores que no tienen grandes conocimientos sobre este, o servir de apoyo a los que ya lo tienen. Para ello, se da primero una pincelada de la situación de este mercado y posteriormente se describen las tecnologías, su integración y las implicaciones para el equipamiento tradicional de la vivienda, sin olvidar el concepto de servicio y su potencial en el despliegue masivo de “productos y sistemas digitales” para las nuevas viviendas.

2.2 Domótica

La Domótica en un inicio fue pensada con beneficios médicos y satisfactorios para personas tanto de tercera edad como para personas discapacitadas pero sobre todo para el cuidado de niños pequeños y por ultimo las mascotas.

Gráfico 1: Domótica[1]

Por lo cual la domótica en la actualidad proporciona un gran número de beneficios y ventajas a nuestra sociedad.

Presenta la domótica como un sistema de gestión técnica inteligente que, mediante la automatización, aporta a la vivienda tradicional la posibilidad de controlar de forma eficiente los sistemas y equipos ya instalados (alarma, televisión, teléfono, eléctrico, cocina.

Analiza detalladamente todos los aspectos que engloba el sistema domótica, desde los constituyentes y sus características, hasta la clasificación, los tipos de protocolo, las fases de instalación y el servicio posventa.

[1] Fuente: http://laelectronica2milenium.blogspot.com/2007/12/la-domotica.html

Repasa el pasado y presente de la domótica, mostrando el escenario actual, las tendencias futuras, los actores del mercado, así como su comportamiento en el mismo.

Los términos domótica, inmótica, edificios inteligentes y similares se utilizan cada vez más para las aplicaciones de las nuevas tecnologías de la información y las comunicaciones en el hogar. Aunque todavía estamos lejos de la implantación extensiva de estas técnicas en ámbitos domésticos, con frecuencia los medios de comunicación muestran edificios públicos que incorporan, además de las habituales automatizaciones de escaleras y ascensores, conceptos como la gestión del confort, el uso racional de la energía, la integración de las redes de datos y comunicaciones.

2.3 Teleasistencia domótica

Hoy en día, la teleasistencia domótica está en pleno auge. Consiste en una atención remota y personalizada a usuarios, que estando en sus hogares, requieren algún tipo de ayuda o seguimiento, utilizando para ello, las tecnologías de la información y las comunicaciones.

Gráfico 2: Teleasistencia[2]

[2] Fuente: http://www.domoticausuarios.es/la-domotica-herramienta-clave-para-el-futuro-de-la-teleasistencia/5894/

2.4 Teleasistencia social

La teleasistencia social (o telecuidados) está dirigida a personas mayores, personas discapacitadas, niños, mujeres maltratadas...ayuda dirigida a personas que requieran un seguimiento asistencial y a personas que viven solas y/o en lugares de difícil acceso. Por otro lado la teleasistencia médica que está dirigida a enfermos crónicos, paliativos o convalecientes, pacientes que necesitan un seguimiento médico especializado y con posibilidad de telemedida de constantes vitales.

2.5 Teleasistencia médica

En teleasistencia médica, se hace un seguimiento médico especializado y con posibilidad de telemedida de constantes biomédicas.

2.6 Placa de control Raspberri PI

El hardware que usaremos es la raspberry PI, para el alojamiento de la interfaz.

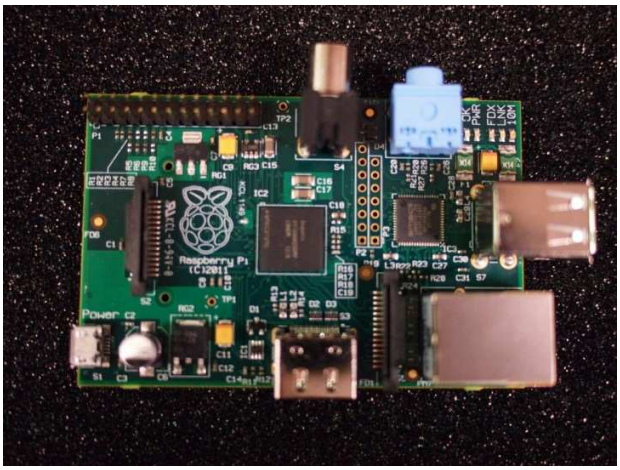


Gráfico 3: Placa raspberry PI[3]

[3] Fuente: Autor

La placa Raspberry es una pequeña tarjeta del tamaño de una tarjeta de crédito pero con las virtudes de una PC, fue diseñada en Reino Unido por una fundación cuyo nombre es el mismo de la placa.

En 2006, los primeros diseños de Raspberry Pi se basaban en el microcontrolador Atmel ATmega644. En mayo de 2009, la Fundación Raspberry Pi fue fundada en Caldecote, South Cambridgeshire, Reino Unido como una asociación caritativa que es regulada por la Comisión de Caridad de Inglaterra y Gales.

El administrador de la fundación, Eben Upton, se puso en contacto con un grupo de profesores, académicos y entusiastas de la informática para crear un ordenador con la intención de animar a los niños a aprender informática como lo hizo en 1981 el ordenador Acorn BBC Micro.15 16 El primer prototipo basado en ARM se montó en un módulo del mismo tamaño que una memoria USB. Tenía un puerto USB en un extremo y un puerto HDMI en el otro.

El Raspberry se conecta a una TV o Pantalla LCD con entrada HDMI y a un teclado. Es una pequeña PC completamente funcional que puede ser usado para muchas de las cosas como una PC de escritorio, también reproduce videos en alta definición.

El diseño contiene un chip Broadcom BCM2835, tiene un procesador central ARM1176JZF-S a 700 MHz.

Tiene un procesador gráfico (GPU) VideoCore IV, y 512 MiB de memoria RAM. El diseño no incluye un disco duro ni unidad de estado sólido, usa una tarjeta SD.

Existen descargas para arquitectura ARM, Raspbian (derivada de Debian), RISC OS 5, Arch Linux ARM (derivado de Arch Linux) y Pidora (derivado de Fedora).

Su objetivo es promover el aprendizaje del lenguaje de programación Python, 5 y otros lenguajes como Tiny BASIC, 12 C y Perl.5.

El Raspberry Pi usa mayoritariamente sistemas operativos basados en el núcleo Linux. Raspbian, una distribución derivada de Debian que está optimizada para el

hardware de Raspberry Pi, se lanzó durante julio de 2012 y es la distribución recomendada por la fundación.

A la GPU se accede mediante una imagen del firmware de código cerrado, llamado blob binario, que se carga dentro de la GPU al arrancar desde la tarjeta SD. El blob binario está asociado a los drivers Linux que también son de código cerrado. Las aplicaciones hacen llamadas a las librerías de tiempo de ejecución que son de código abierto, y estas librerías hacen llamadas a unos drivers de código abierto en el kernel de Linux. La API del driver del kernel es específica para estas librerías. Las aplicaciones que usan vídeo hacen uso de OpenMAX, las aplicaciones 3D usan OpenGL ES y las aplicaciones 2D usan OpenVG; OpenGL ES y OpenVG hacen uso de EGL y éste último, del driver de código abierto del kernel.84

El 19 de febrero de 2012, se lanzó el primer prototipo de imagen de tarjeta SD que almacenaba un sistema operativo y que podía ser cargado en una tarjeta SD. La imagen se basaba en Debian 6.0 (Squezze), con el escritorio LXDE y el navegador Midori, más algunas herramientas de programación. La imagen funcionaba bajo QEMU permitiendo que el Raspberry Pi pudiera ser emulado en otros sistemas.85

El 8 de marzo de 2012, se lanzó Raspberry Pi Fedora Remix (actualmente llamada Pidora), que en el momento de era la distribución recomendada por la fundación y fue desarrollada en la universidad de Séneca, en Canadá.

El 24 de octubre de 2012, Alex Bradbury, director de desarrollo Linux de Raspberry, anunció que todo el código del driver de la GPU Videocore que se ejecuta en ARM sería de código abierto.

El 5 de noviembre de 2012, Eben Upton anunció el lanzamiento del sistema operativo RISC OS 5 para Raspberry Pi a la comunidad, pudiéndose descargar la imagen de forma gratuita desde la web.

El 26 de septiembre de 2013, se añadió a los repositorios de Raspbian una versión oficial de Oracle Java JDK ARM con soporte para coma flotante por

hardware, que ofrece bastante más rendimiento que la versión OpenJDK ARM ya existente hasta ese momento y más compatibilidad con aplicaciones. También se anunció que esta versión de Oracle Java JDK se incluiría dentro de la distribución en futuras versiones de Raspbian.

Esta es una lista de sistemas operativos que funcionan, se han portado, o están en proceso de ser portados a Raspberry Pi:

Sistemas operativos completos:

- AROS
- Linux
- Android97
- Arch Linux ARM
- Debian Whezzy Soft-Float, versión de Debian sin soporte para coma flotante por hardware
- Firefox OS
- Gentoo Linux98
- Google Chromium OS
- Kali Linux
- Open webOS99
- PiBang Linux100 , distribución Linux derivada de Raspbian con diferente escritorio y aplicaciones
- Pidora, versión Fedora Remix optimizada101
- QtonPi, distribución linux con un framework de aplicaciones multiplataforma basado en Qt framework
- Raspbian102 , versión de Debian Wheezy para ARMv6 con soporte para coma flotante por hardware
- Slackware ARM, también conocida como ARMedslack
- Plan 9 from Bell Labs103 104
- RISC OS 52
- Unix

- FreeBSD105
- NetBSD106 107

2.7 PIFACE

PiFace Digital está diseñado para ser conectado al puerto GPIO de la Raspberry Pi. Con PiFace digital se puede detectar el estado de un interruptor, por ejemplo, un sensor de una puerta, un botón o cualquier otro tipo de interruptores. Una vez que se ha detectado el estado, usted puede escribir su propio software para el Raspberry Pi que determine cómo responder a ese estado del interruptor. Usted puede conectar las salidas a motores eléctricos, actuadores, LEDs o cualquier otra cosa que pueda imaginar.

2.7.1 Características

- Se conecta directamente al puerto GPIO de la Raspberry Pi.
- 2 relés de conmutación.
- 4 Interruptores táctiles.
- 8 entradas digitales.
- 8 salidas open-Collector.
- 8 indicadores LED.
- Fácil de programar en Python, Scratch y C.
- Emulador gráfico y simulador.
- Los relés se pueden utilizar para conmutar tensiones de hasta 20 V (máx.) y corrientes de hasta 5A (Max).

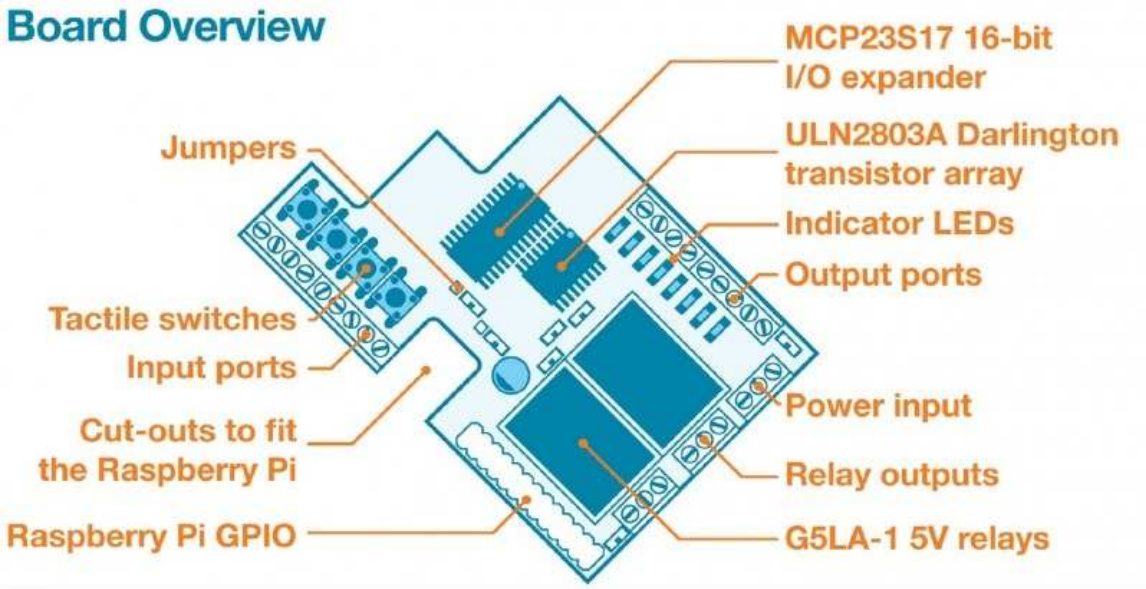


Gráfico 4: Placa PIFACE[4]

[4] Fuente: http://electronilab.co/tienda/piface-digital-io-para-raspberry-pi/#!prettyPhoto

CAPÍTULO 3

3. METODOLOGÍA DE DESARROLLO

3.1 Metodología

La metodología a usar es la siguiente:

3.1.1 Inducción

Se recopila datos para el diseño del software y hardware del sistema mediante la recolección de datos principalmente para el uso y programación de la placa electrónica Raspberry.

3.1.2 Deducción

Se analiza el camino adecuado para la solución del problema y se adaptará las posibles soluciones en función de nuestra realidad

3.1.3 Análisis

Se aplica la solucion en base a los datos obtenidos inicialmente en base a la lógica y a lo racional, lo cual servirá para mostrar y desarrollar la mejor solución del problema.

3.1.4 Síntesis

Se aplicara la solución más adecuada para el control de luces de una casa mediante la programación de puertos.

3.2 Técnica de Investigación

Las Técnica de Investigación que se va aplicar es:

3.2.1 Observación

Mediante la observación se recopila datos de la web sobre las soluciones de domótica y aplicaciones realizadas con la placa raspberry.

Se consultará documentos de la web.

Se buscará información en libros y folletos electrónicos.

3.3 Entrevista

3.3.1 Calculo de la muestra

Se ha decidido realizar una entrevista en el cantón de Quito, sector la Mariscal teniendo en cuenta que se dan las condiciones necesarias para la obtención de datos requeridos.

3.3.2 Tamaño de la muestra

Para establecer el tamaño de la muestra, es decir, el número de entrevistas que se realizaron, se tomó en cuenta la siguiente información:

- Mercado objetivo, personas que tienen casa o departamento propio, que tengan algún tipo de discapacidad, sean parte de la tercera edad, personas de 40 a 64 años y que vivan en el dentro del sector de la Mariscal; obteniendo una población de 1000 personas.
- Nivel de confianza de 90% y grado de error de 5%.

Aplicando la fórmula de la muestra:

$$n = (Z^2PQN) / (Ne^2 + ^2PQ)$$

- Nivel de confianza (Z) = 1.65
- Grado de error (e) = 0.05
- Universo (N) =1000
- Probabilidad de ocurrencia (P) = 0.5
- Probabilidad de no ocurrencia (Q) = 0.5

$n = ((1.65)^2 (0.5) (0.5) (1000)) / ((1000) (0.05)^2 + (1.65)^2 (0.5) (0.5))$

$n = ((2.72) (0.25) (1000)) / ((1000) (0.0025) + (2.72) (0.25))$

$n = 680 / 2.5 + 0.6$

n = 680 / 3.18

n = 213

Con este resultado se decidió realizar 50 entrevistas

3.3.3 Modelo de la entrevista

Para evaluar la aceptación y el impacto de la sociedad de este sistema domótica para personas de tercera edad y minusválidos, se ha realizado la siguiente entrevista.

1. ¿Usted posee algún tipo de discapacidad o pertenece a la tercera edad?
    - o Si
    - o No

    Si la respuesta es NO pace a la tercera pregunta.
2. ¿Le gustaría tener un sistema domótico que le permita controlar de manera más fácil a su hogar?
    - o Si
    - o No
3. Estaría dispuesto a comprar un sistema domótico para mejorar su calidad de vida.
    - o Si
    - o No

3.3.4 Resultados de la entrevista

Pregunta #1

SI	NO
36	14

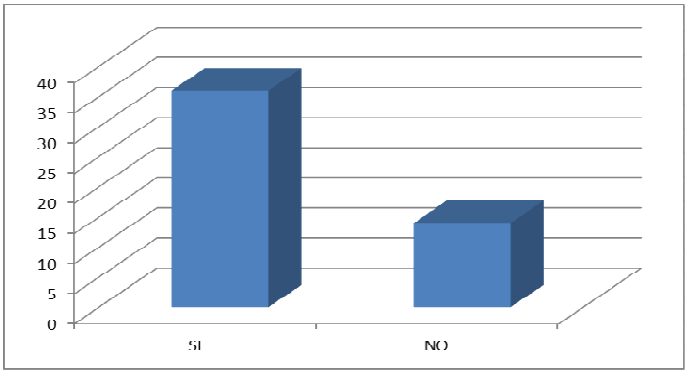


Gráfico 5: Estadísticas[5]

Pregunta #2

SI	NO
40	10

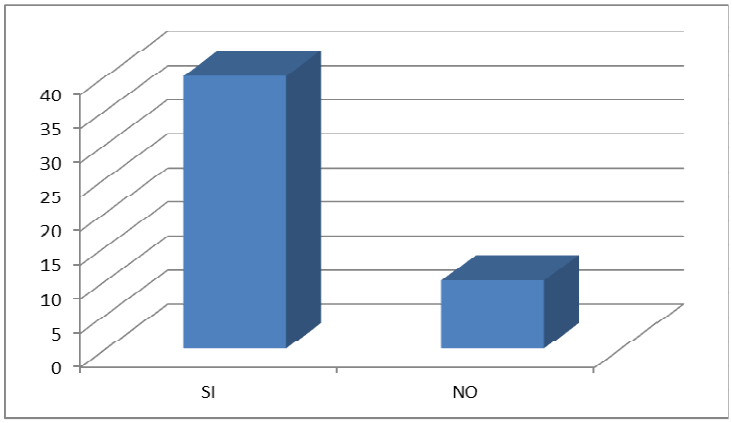


Gráfico 6: Estadísticas[6]

[5] Fuente: Autor
[6] Fuente: Autor

Pregunta #3

SI	NO
30	20

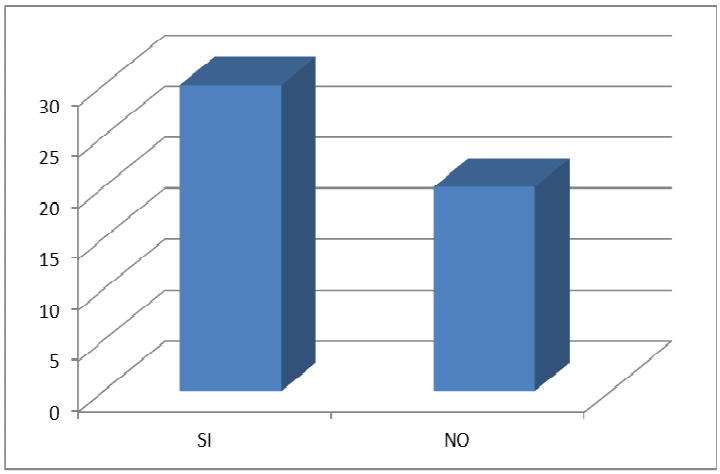


Gráfico 7: Estadísticas[7]

De acuerdo a los resultados obtenidos con la entrevista realizada, se palpa que la sociedad en especial las personas de tercera edad y con discapacidad si están dispuestos a adquirir un sistema domótica para mejor la calidad de vida.

[7] Fuente: Autor

CAPÍTULO 4

4.1 DESARROLLO

Este trabajo se orienta al uso de nuevas tecnologías como es la utilización de la placa Raspberry, esta placa posee la principal característica de alojar en su interior un sistema operativo LINUX, esta se encuentra recientemente en el mercado mundial desde noviembre del 2012, donde no existe mucho desarrollo.

El tema propuesto ayudará al usuario a controlar de manera amigable las escenas de luminosidad de su hogar, reduciendo costos y espacio por medio de la utilización de la Raspberry.

El sistema completo consta de tres partes:

4.1.1 SISTEMA INFORMÁTICO

El sistema informático de control de luces RaspControl, es la interfaz que permite conectar al usuario con el sistema.

El programa permitirá accionar las luces cuando el usuario pulse un evento de botón en la interfaz. Esta fue realizado en python según las características de la raspberry, ya que es multiplataforma, es decir puede ser ejecutado en el sistema operativo Linux.

La Placa Raspberry servirá como servidor del sistema para ello es mencionaremos las características y bondades de esta placa.

Cabe recalcar que la ejecución de drivers necesarios para conectarse con los periféricos externos se lo realiza de diferente manera en cada Sistema Operativo.

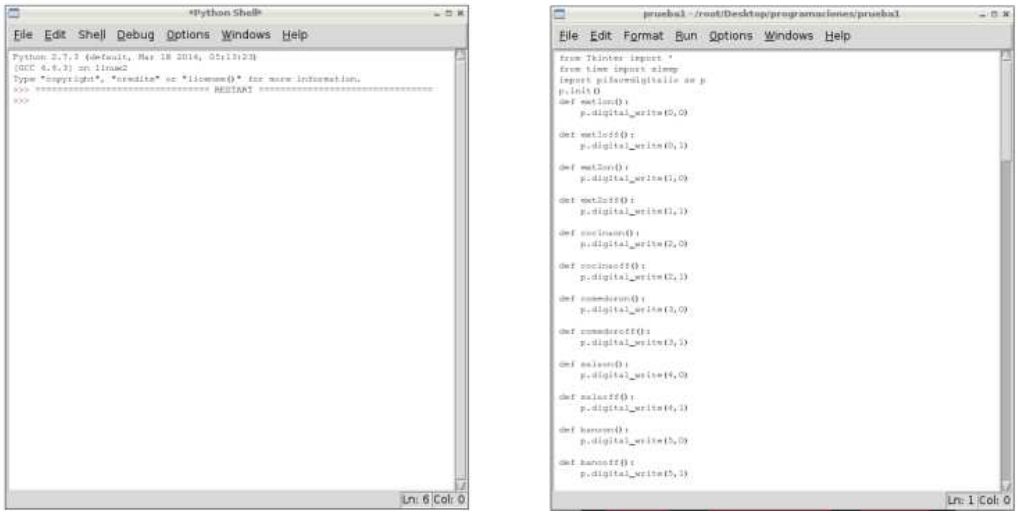


Gráfico 8: Python[8]

A continuación se explica la rutina de la aplicación.

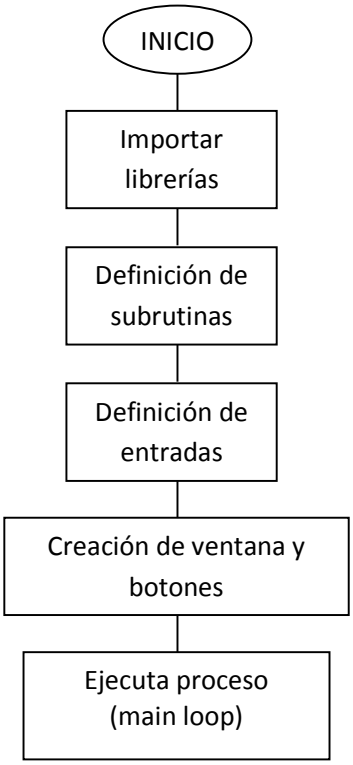


Gráfico 9: Diagrama de programación

[8] Fuente: Autor

Luego de haber realizado la programación la cual se la puede observar en los anexos, se tiene la siguiente interfaz donde el usuario puede ejercer el control del sistema domótico.

Gráfico 10: Interfaz de usuario[9]

4.1.2 Placa de Interacción

La placa de interacción es una tarjeta electrónica que permite conectar a la placa Raspberry con el sistema eléctrico de la casa, en este caso la maqueta.

Para que el Sistema operativo reconozca esta tarjeta es necesario instalar un driver, el cuál creará un puerto virtual serial en la Raspberry y permitirá enviar datos seriales para el control de Luces.

[9] Fuente: Autor

Esta tarjeta posee relés para activar las luces de la maqueta, también tiene pulsadores para simular entradas con sus respectivas borneras.

Gráfico 11: PiFace[10]

4.1.3 Placa de poder

Esta fase corresponde a la instalación eléctrica de la maqueta, con sus respectivas luces.

El sistema constará de una maqueta de una vivienda de un piso que servirá para simular las escenas de luz del interior.

Para el funcionamiento del sistema implementado en la maqueta es necesario tener una toma de voltaje de 110V AC para los focos de 110V AC instalados, los mismos que nos proporciona directo la red eléctrica, para la simulación de encendido de las luces ambientes se ha colocado unos leds, los cuales funcionan a 3VDC los mismos que se alimentan de la placa raspberry pi.

La maqueta simulará una casa habitable para la demostración del correcto funcionamiento del sistema.

[10] http://piregistration.element14.com/signup.html

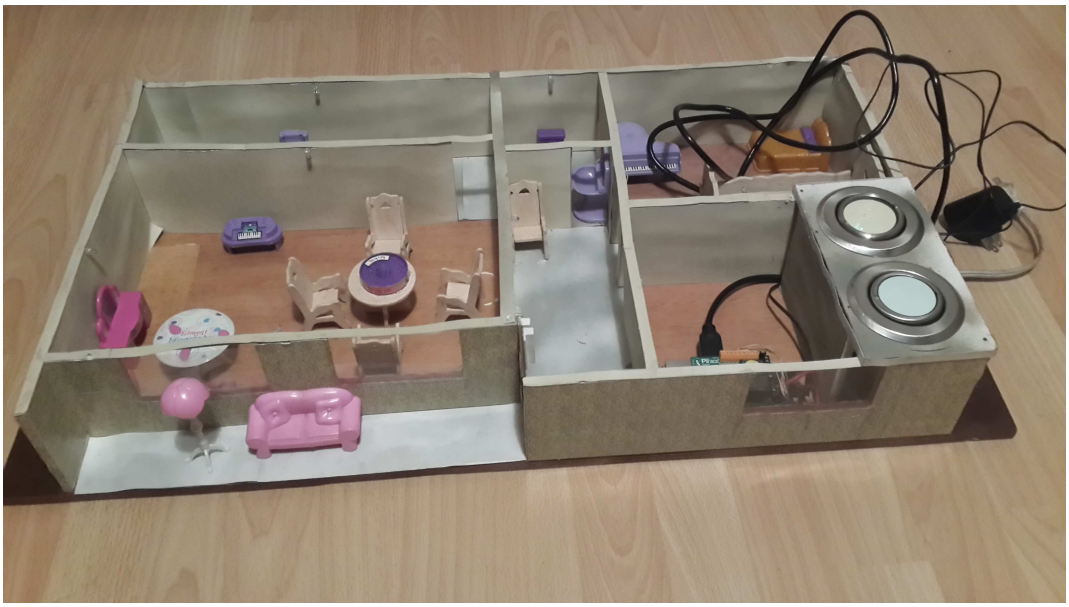


Gráfico 12: Maqueta terminada[11]

Pruebas realizadas

Las pruebas que se hicieron al sistema domótico en un domicilio rodeado de diferentes interferencias de artefactos eléctricos y electrónicos tales como el horno microondas, transmisión bluetooth de teléfonos móviles, teléfono inalámbrico.

Para realizar acabo estas pruebas, se pusieron a funcionar todos los artefactos anteriormente mencionados, donde el sistema domótico funciono correctamente, para lo cual se realizaron se enviaron 20 comandos desde la aplicación de sistema y se ejecutaron el 100% de los mismos sin ningún inconveniente.

Con las pruebas realizadas tenemos un sistema bien fiable y robusto.

Ver Anexos.

[11] Fuente: Autor

Gráfico 13: Pruebas de realizadas[12]

[12] Fuente: Autor

5.- CONCLUSIONES

5.1 CONCLUSIONES

1.- Se concluye que el sistema es muy recomendable para ser usado por personas de la tercera edad, niños y discapacitados, tomando en cuenta que estas tengan un raciocinio normal.

2.- El sistema ayudará al usuario a controlar de manera amigable las escenas de luminosidad de su hogar, reduciendo costos y espacio.

3.- El Sistema es de fácil utilización para personas de tercera edad y discapacitadas debido a su interfaz amigable.

4.- El sistema cumple con los objetivos planteados al inicio del proyecto por lo que es totalmente funcional.

5.2 RECOMENDACIONES

1.- El sistema debe funcionar con una batería de 5v o a su ver un adaptador para poder alimentar a la placa y poner en funcionamiento el sistema

2.- Para realizar una completa instalación en una casa u hogar es necesario de una persona de altos conocimientos en electrónica.

3.- Para instalar la interfaz en otras placas raspberry es necesario de un Ingeniero en Sistemas con conocimientos en sistemas operativos LINUX.

4.- En caso de que se dañe la placa raspberry es necesario adquirir otra y nuevamente cargar el programa de instalación mediante comandos Unix.

5.- De igual manera para instalar la placa de interfaz es necesario de un especialista para instalar los Drivers en el Sistema.

5.3 BIBLIOGRAFÍA

a) Escalona-Franco, Gabriel (2011). FILOSOFÍA, IDENTIFICACIÓN Y RACIONALIZACIÓN DE ALARMAS EN SCADA APLICADO A LA DOMÓTICA DE UN HOTEL ENERGY management (2 Ed.).Londres

b) Pedro M. Alonso, (2010). Casas inteligentes gracias a la domótica, Gale Power Search (3 Ed.) Madrid

c) Henríquez, Mauricio (2011). Control Automático de Condiciones Ambientales en Domótica usando Redes Neuronales Artificiales, Networks. (English) (1 ED.) Chile

Web: http://laelectronica2milenium.blogspot.com/2007/12/la-domotica.html

Web:http://www.domoticausuarios.es/la-domotica-herramienta-clave-para-el-futuro-de-la-teleasistencia/5894/

Web:http://es.wikipedia.org/wiki/Raspberry_Pi

Web:http://electronilab.co/tienda/piface-digital-io-para-raspberry-pi/#!prettyPhoto

Web:http://www.farnell.com/datasheets/1524403.pdf

Web:http://www.element14.com/community/docs/DOC-55183/l/piface-digital-datasheetpdf

ANEXOS

RASPBERRY PI

Quick Start Guide

The Raspberry Pi – Single Board Computer

Source: Raspberry Pi & Wiki

Chapter 1: RPi Hardware Basic Setup

Typical Hardware You Will Need

While the RPi can be used without any additional hardware (except perhaps a power supply of some kind), it won't be much use as a general computer. As with any normal PC, it is likely you will need some additional hardware.

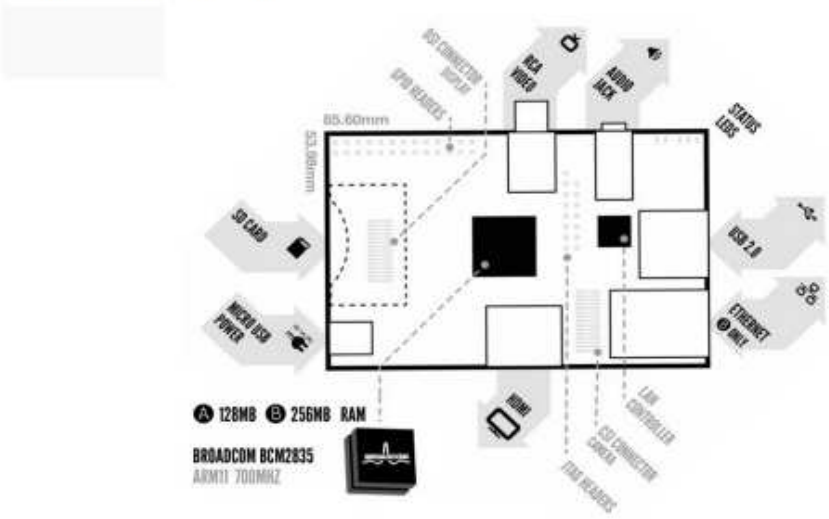
The following are more or less essential:

- Raspberry Pi board
- Prepared Operating System SD Card
- USB keyboard
- Display (with HDMI, DVI, Composite or SCART input)
- Power Supply
- Cables

Highly suggested extras include:

- USB mouse
- Internet connectivity - a USB WiFi adaptor (Model A/B) or a LAN cable (Model B)
- Powered USB Hub
- Case

Connecting Together



You can use the diagram to connect everything together, or use the following instructions:

1. Plug the preloaded SD Card into the Pi.
2. Plug the USB keyboard and mouse into the Pi, perhaps via a USB Hub. Connect the Hub to power, if necessary.

3. Plug the video cable into the screen (TV) and into the Pi.
4. Plug your extras into the Pi (USB WiFi, Ethernet cable, hard drive etc.). This is where you may really need a USB Hub.
5. Ensure that your USB Hub (if any) and screen are working.
6. Plug the power source into the main socket.
7. With your screen on, plug the other end of the power source into the Pi.
8. The Pi should boot up and display messages on the screen.

It is always recommended to connect the MicroUSB Power to the unit last (while most connections can be made live, it is best practice to connect items such as displays/h/w pin connections with the power turned off).

The RPi may take a long time to boot when powered-on for the first time, so be patient!

Prepared Operating System SD Card

As the RPi has no internal storage or built-in operating system it requires an SD-Card that is set up to boot the RPi.

- You can create your own preloaded card using any suitable SD card you have. Be sure to backup any existing data on the card.
- Preloaded SD cards will be available from the RPi Shop.

This guide will assume you have a preloaded SD card.

Keyboard & Mouse

Most standard USB keyboards and mice will work with the RPi. Wireless keyboard/mice should also function, and only require a single USB port for an RF dongle. In order to use a Bluetooth keyboard or mouse you would need to use a Bluetooth dongle, which again uses a single port.

Remember that the Model A has a single USB port and the Model B only has two (typically a keyboard and mouse will use a USB port each).

Display

There are two main connection options for the RPi display, *HDMI* (high definition) and *Composite* (low definition).

- HD TVs and most LCD Monitors can be connected using a full-size 'male' HDMI cable, and with an inexpensive adaptor if DVI is used. HDMI versions 1.3 and 1.4 are supported, and a version 1.4 cable is recommended. The RPi outputs audio and video via HMDI, but does not support HDMI input.
- Older TVs can be connected using Composite (a yellow-to-yellow cable) or via SCART (using a Composite to SCART adaptor). PAL and NTSC TVs are supported. When using composite video, audio is available from a 3.5mm (1/8 inch) socket, and can be sent to your TV, to headphones, or to an amplifier. To send audio your TV,

you will need a cable which adapts from 3.5mm to double (red and white) RCA connectors.

Note: There is no VGA output available, so older VGA monitors will require an expensive adaptor.

Using an HDMI to DVI-D (digital) adaptor plus a DVI to VGA adaptor will not work. HDMI does not supply the DVI-A (analogue) needed to convert to VGA - converting an HDMI or DVI-D source to VGA (or component) needs an active converter. (It can work out cheaper to buy a new monitor.) The lack of VGA has been acknowledged as a priority issue.

Power Supply

The unit uses a Micro USB connection to power itself (only the power pins are connected - so it will not transfer data over this connection). A standard modern phone charger with a micro-USB connector will do, but needs to produce at least 700mA at 5 volts. Check your power supply's ratings carefully. Suitable mains adaptors will be available from the RPi Shop and are recommended if you are unsure what to use.

You can use a range of other power sources (assuming they are able to provide enough current ~700mA):

- Computer USB Port or powered USB hub (will depend on power output)
- Special wall warts with USB ports
- Mobile Phone Backup Battery (will depend on power output) (in theory - needs confirmation)

To use the above, you'll need a USB A 'male' to USB micro 'male' cable - these are often shipped as data cables with MP3 players.

Cables

You will probably need a number of cables in order to connect your RPi up.

1. Micro-B USB Power Cable
2. HDMI-A or Composite cable, plus DVI adaptor or SCART adaptor if required, to connect your RPi to the Display/Monitor/TV of your choice.
3. Audio cable, this is not needed if you use a HDMI TV/monitor.
4. Ethernet/LAN Cable

Additional Peripherals

You may decide you want to use various other devices with your RPi, such as Flash Drives/Portable Hard Drives, Speakers etc.

Internet Connectivity

This may be an Ethernet/LAN cable (standard RJ45 connector) or a USB WiFi adaptor. The RPi ethernet port is auto-sensing which means that it may be connected to a router or directly to another computer (without the need for a crossover cable).

USB-Hub

In order to connect additional devices to the RPi, you may want to obtain a USB Hub, which will allow multiple devices to be used.

It is recommended that a **powered** hub is used - this will provide any additional power to the devices without affecting the RPi itself.

USB version 2.0 is recommended. USB version 1.1 is fine for keyboards and mice, but may not be fast enough for other accessories.

Case

Since the RPi is supplied without a case, it will be important to ensure that you do not use it in places where it will come into contact with conductive metal or liquids, unless suitably protected.

Expansion & Low Level Peripherals

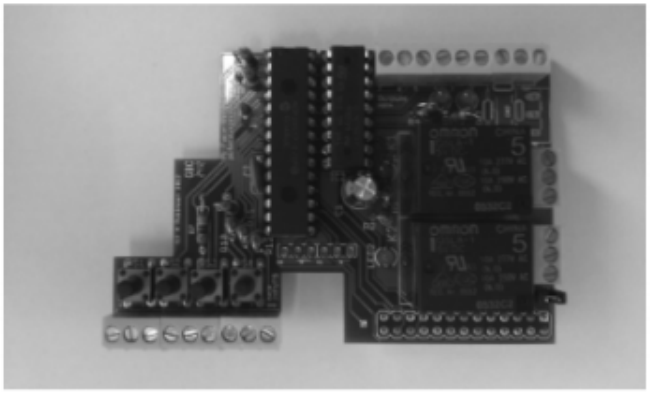
If you plan on making use of the low level interfaces available on the RPi, then ensure you have suitable header pins for the GPIO (and if required JTAG) suitable for your needs.

Also if you have a particular low-level project in mind, then ensure you design in suitable protection circuits to keep your RPi safe.

PIFACE

PiFace

PiFace Digital Datasheet



Version 2

MANCHESTER 1824

The University of Manchester

School of Computer Science

PiFace Overview

PiFace Digital is the first in a range of interfaces that allows your Raspberry Pi to control input and output devices in the real world. It allows the Raspberry Pi to read switches connected to the PiFace, such as a door sensor or pressure pad. With easy to write code, you can make your Raspberry Pi drive outputs, powering motors, LEDs and anything else you can imagine to respond to the inputs.

Summary

Board footprint	56mm x 84.5mm
Digital outputs	8
Digital inputs	8
Raspberry Pi pins	3v3, 5v, 0v, SPI MOSI, SPI MISO, SPI SCLK, SPI10 CE0 N, SPI10 CE1 N

Components

8x **3 pin screw terminal**

1x **26 pin female connector**

4x **3mm red LED**

12x **330Ω Resistor**

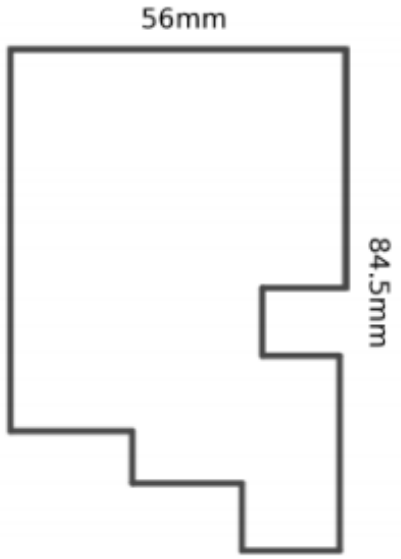
4x **6mm x 6mm push button**

1x **100nF Ceramic capacitor**

1x **100µF Electrolytic capacitor**

2x **OMRON G5LA-1 5VDC relay**

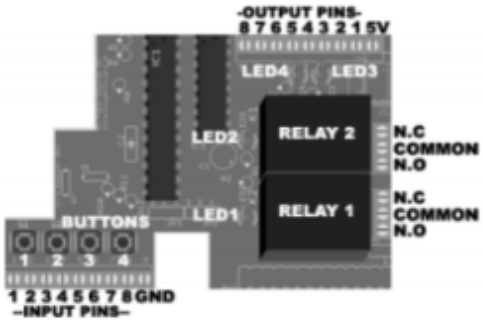
1x **MCP23S17-E/SP - Specialised 16bit Input/Output SPI interface chip**



PiFace The board

Overview

The PiFace board has 8 inputs pins located at the front of the board, and 8 output pins located at the rear of the board. It has 6 pins that are tied to its 2 relays (3 pins per relay).



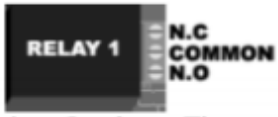
Buttons

The PiFace has 4 onboard momentary pushbuttons. These are connected to input pins 1-4 respectively. These are non-latching buttons so the circuit is complete only while the button is being pressed. These pins will still read other inputs connected to the pin terminals and if either or both are in their on state, the pin will read as 1, if both are off, the pin is read as 0.

LEDs

The PiFace has 4 onboard red LEDs. These are connected to output pins 1-4 respectively. These pins will still control other outputs through their pin terminals, while controlling these LEDs.

Relays



The PiFace has 2 relays. These are connected to output pins 1 & 2 respectively. These pins will still control other outputs through their pin terminals, while controlling these relays.

A relay is an electromechanical switch. Each relay has three pin terminals connected to it, N.C (normally closed), COM (common) and N.O (normally open). Normally N.C is connected to COM and N.O is not connected to anything. If you perform a digital write to pin 1 (or pin 2 for relay 2) N.O and COM will be connected and N.C will not be connected to anything. Setting pin 1 LOW (off) will return the relay to its natural state. This is useful for controlling different voltaged items without having to properly interface them with the PiFace, as the two are not electrically connected.

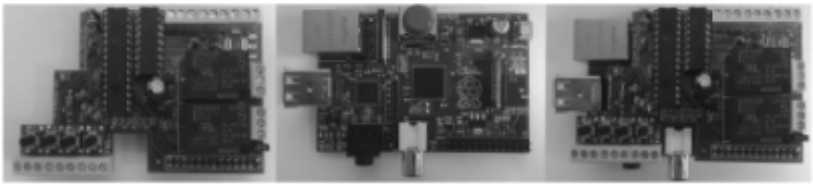
PiFace

Fitting the PiFace

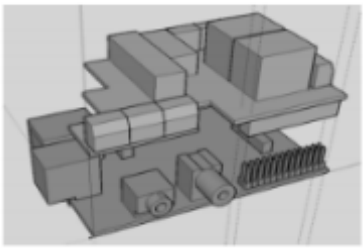
You should stick the small plastic bumper on top of the HDMI port on the Pi. This is will support the PiFace and stop it touching the Raspberry Pi to avoid short circuiting anything on either device.

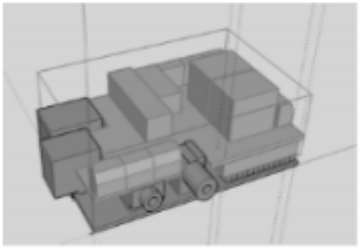
*The part number for the bumper is PD.2125G

On the Raspberry Pi there is a set of 26 General Purpose Input/Output (GPIO) pins in 2 rows of 13, some of these are used by the PiFace. These pins slot into the female connector on the underside of the PiFace. To attach the PiFace to the Pi align the PiFace as show below, with the yellow composite video connector enclosed by the cut-out square on the PiFace.



Then gently lower the PiFace onto the Pi, while aligning the GPIO pins with the PiFace's connector.





While doing this you should be observant of other components (especially the composite video connector) on the Raspberry Pi to ensure they are fitting around the PiFace and avoid damaging either device.

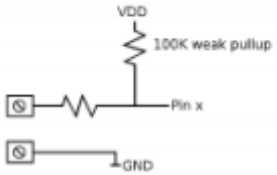
PiFace

Inputs & Outputs

Inputs and outputs are connected via the screw terminals located around the board. The terminals have 3mm flat head screws. To connect a wire to a pin terminal, unscrew the screw until the screw head is flush with the plastic which encases it. Insert the wire into the corresponding pinhole on the terminal's outfacing side and then tighten the screw until it bites gently.

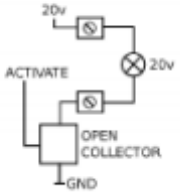
Inputs

The 8 digital inputs have a weak internal pullup. They register an input when they are pulled down to ground. There is a resistor on the input pin to limit the current.



Outputs

The 8 digital outputs are buffered through the ULN2803A open-collector chip. When actuated the outputs sink (all currnt to flow to ground), think of it as a switch with one terminal connected to ground. To connect a circuit you must connect on end of the output to 5v and the other to an output pin. When the output pin is set high the current will go from 5v to the grounded output pin.

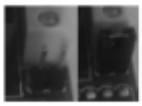


Ground & 5v

The PiFace has a ground pin by the inputs and a 5v VCC pin by the outputs.

Power Jumper

Next to relay 1 there are two vertical pins, these are for switching power modes. If connected by a standard jumper then the PiFace and Raspberry Pi can be powered from the same source (either from the Raspberry Pi's power source or PiFace's). If these pins are not connected then the two devices are powerred separately.



PiFace Programming

The PiFace is controlled with the pfio package which communicates with the PiFace board over the SPI bus. The python pfio library uses functions in the SPI module to send specific commands to the PiFace. To use PiFace from your programs, you need to import the pfio package.

How to use pfio in python

To import and use the pfio package in your python programs you must include the following lines near the top of your program.

```
>>> import piface.pfio as pfio      #Import the library
>>> pfio.init()                     #Initialise the board
>>> pfio.digital_write(1,1)         #Set pin 1 to high (example)
```

How to use pfio in C

There is a C pfio library with the same functions and structure as the python pfio. This is for people who are more experienced in programming. Include the C pfio header file in your PiFace C programs.

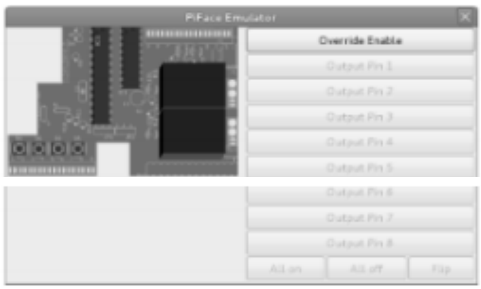
```
#include <piface/pfio.h>

int main(void){
   pfio_init();
   pfio_digital_write(1, 1);
}
```

Scratch

You can use a modified version of Scratch with PiFace.
(documentation provided elsewhere)

Emulator

The pfio has a built-in emulator that allows you to run code intended for the PiFace but without the board being connected. On startup the emulator detects whether the PiFace board is connected, if not it runs in simulator mode, otherwise it runs in controller mode. We recommend you do not connect or disconnect PiFace with the Pi powered, if you do you must close and re-run the emulator so it can detect the board and enter the correct mode.



Behind the scenes:

The python pfio package uses a C SPI module to communicate with the PiFace board. Advanced users can use this module directly to control the PiFace board.

5

Código de programación

```
from Tkinter import *                    //importar la librería Tkinter para grafico de botones, etiquetas

from time import sleep                   // importar la librería time para tiempos de espera si fueran necesarios

import pifacedigitalio as p  //importar la librería pifacedigitalio y llamarla p esta librería permite
                                         // controlar la tarjeta piface de la raspberry pi

p.init()                 // inicializamos la librería pifacedigitalio

// OUTPUTS

// las entradas y salidas de la piface tienen lógica invertida

//p.digital_write(a,b) donde (a) es la salida de la piface y (b) es el valor lógico de la salida



def ext1on():                    // definimos una subrutina llamado "ext1on"
        p.digital_write(0,0)  //se llama a la librería pifacedigitalio para realizar
                                         //una escritura de tipo digital con un valor de 0L en el puerto 0 de salidas de la piface


def ext1off():             // definimos una subrutina llamado "ext1on"
        p.digital_write(0,1)  //se llama a la librería pifacedigitalio para realizar
                                         //una escritura de tipo digital con un valor de 1L en el puerto 0 de salidas de la piface
```

```
def ext2on():              // definimos una subrutina llamado "ext2on"
    p.digital_write(1,0)   //se llama a la librería pifacedigitalio para realizar
                           //una escritura de tipo digital con un valor de 0L en el puerto 1 de salidas de la piface

def ext2off():             // definimos una subrutina llamado "ext1on"
    p.digital_write(1,0)   //se llama a la librería pifacedigitalio para realizar
                           //una escritura de tipo digital con un valor de 1L en el puerto 1 de salidas de la piface

//realizamos este procedimiento con todos los demás botones que se quiere realizar

//INPUTS

def sensor():              // definimos una subrutina llamado "sensor"
    m = p.digital_read(0)  // asignamos a la variable (m) el valor que nos entrega al llamar a la librería
                           // pifacedigitalio de la entrada 0
    n = p.digital_read(1)  // asignamos a la variable (n) el valor que nos entrega al llamar a la librería
                           // pifacedigitalio de la entrada 1
    o = p.digital_read(2)  // asignamos a la variable (o) el valor que nos entrega al llamar a la librería
                           // pifacedigitalio de la entrada 2
```

q = p.digital_read(3) // asignamos a la variable (q) el valor que nos entrega al llamar a la librería

// pifacedigitalio de la entrada 3

if m == 0: // realizamos una comparación de valores de la variable para realizar las subrutinas

toggleLabel["text"]= "PUERTA ABIERTA" //toggleLabel["text"] es la etiqueta que se crea para obtener

//en la interfaz el estado del sensor y se coloca PUERTA ABIERTA

// si el sensor se encuentra en 0L

if m == 1:

toggleLabel["text"]= "PUERTA CERRADA" //toggleLabel["text"] es la etiqueta que se crea para obtener

//en la interfaz el estado del sensor y se coloca PUERTA CERRADA

//si el sensor se encuentra en 1L

if n == 0: // realizamos una comparación de valores de la variable para realizar las subrutinas

toggleLabel1["text"]= "PUERTA ABIERTA" //toggleLabel["text"] es la etiqueta que se crea para obtener

//en la interfaz el estado del sensor y se coloca PUERTA ABIERTA

// si el sensor se encuentra en 0L

if n == 1:

toggleLabel1["text"]= "PUERTA CERRADA" //toggleLabel["text"] es la etiqueta que se crea para obtener

//en la interfaz el estado del sensor y se coloca PUERTA CERRADA

//si el sensor se encuentra en 1L

// se realiza lo mismo con el resto de variables teniendo en cuenta el no colocar en la misma etiqueta

root = Tk() //realiza un root a la librería Tk ya que necesita permisos de administrador

root.title("DOMOTICA") // coloca el título de la ventana que vamos a crear

toggleLabel = Label(root,text="SENSOR PUERTA") // crea una etiqueta con el nombre Label y de texto SENSOR PUERTA

toogleLabel.pack(side=TOP) // coloca la etiqueta en el TOP de la ventana

toggleLabel1 = Label(root,text="SENSOR PUERTA1") // crea una etiqueta con el nombre Label1 y de texto SENSOR PUERTA1

toogleLabel.pack(side=TOP) // coloca la etiqueta en el TOP de la ventana

// se realiza lo mismo para todas las etiquetas que son de los sensores

toggleButton0 = Button(root,text=["sensor"],command=sensor) //crea un botón llamado Button0 de texto sensor, el cual al presionar realizara la

//subrutina sensor

toogleButton0.pack(side=LEFT) // coloca el botón en el LEFT de la ventana

toggleButton = Button(root,text=["ext1ON"],command=ext1on) //crea un botón llamado Button de texto ext1ON, el cual al presionar realizara la

//subrutina ext1on

toogleButton.pack(side=LEFT) // coloca el botón en el LEFT de la ventana

quitButton = Button(root,text="salir",command=exit) //crea un botón del tipo salir el cual al presionar cierra la ventana de ejecución

toogleButton.pack(side=TOP) // coloca el botón en el LEFT de la ventana

//se realiza lo mismo para todos los botones que deseamos ubicar

root.mainloop() // realiza un bucle para que siempre se este ejecutando el archivo

Manual de Usuario

Ventana de escritorio raspberry PI

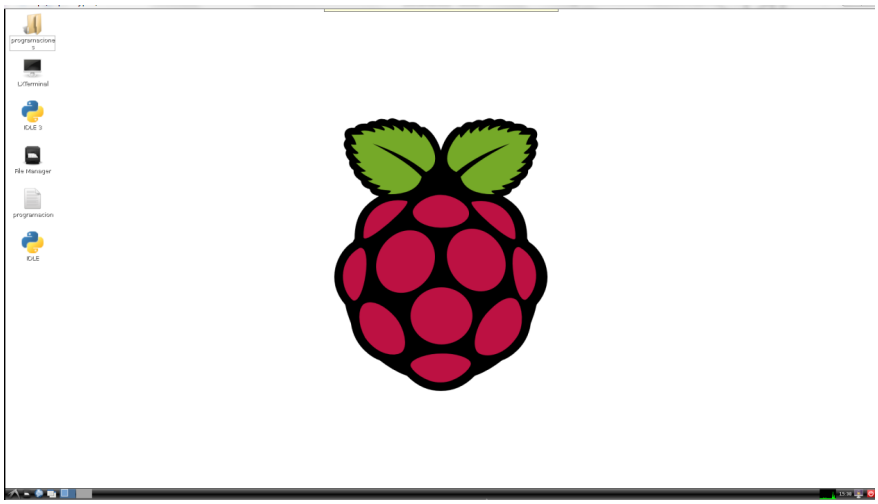


Imagen A1: Escritorio raspbeery pi

Abrimos el IDLE

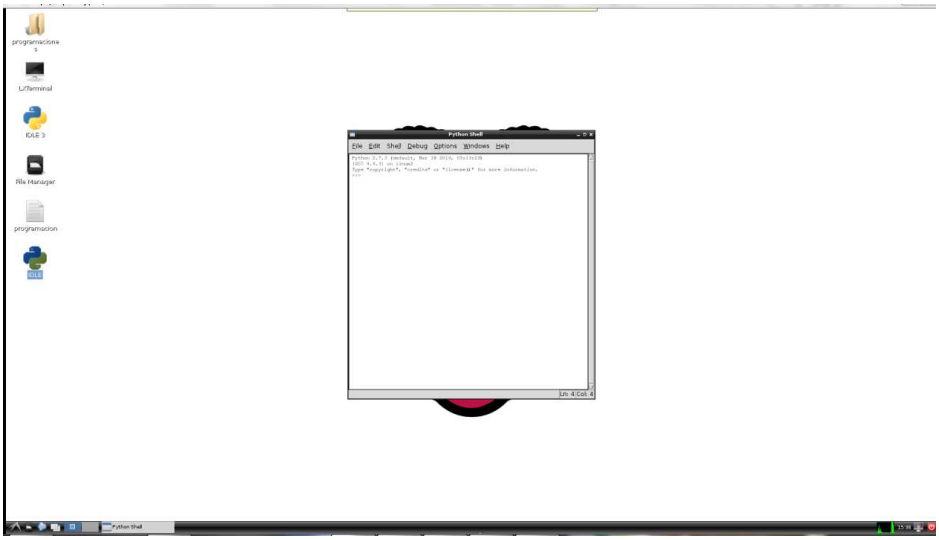


Imagen A2: Idle

Abrimos el documento de programación en python

Imagen A3: Python

Se encuentra en la carpeta del sistema root/desktop/programaciones/prueba1

Imagen A4: Directorio de carpeta

Archivo prueba1

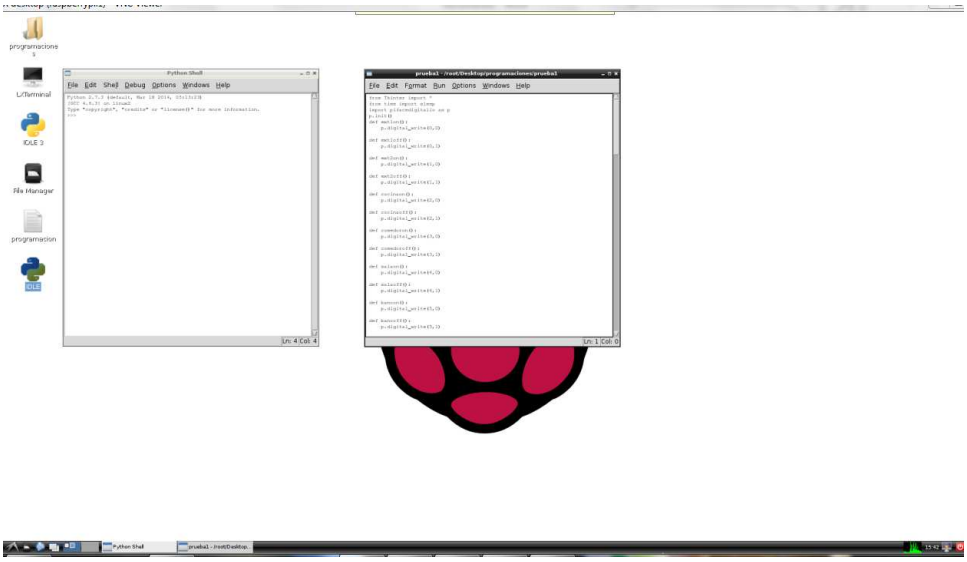


Imagen A5: Archivo prueba

Corremos el programa en run >> run module

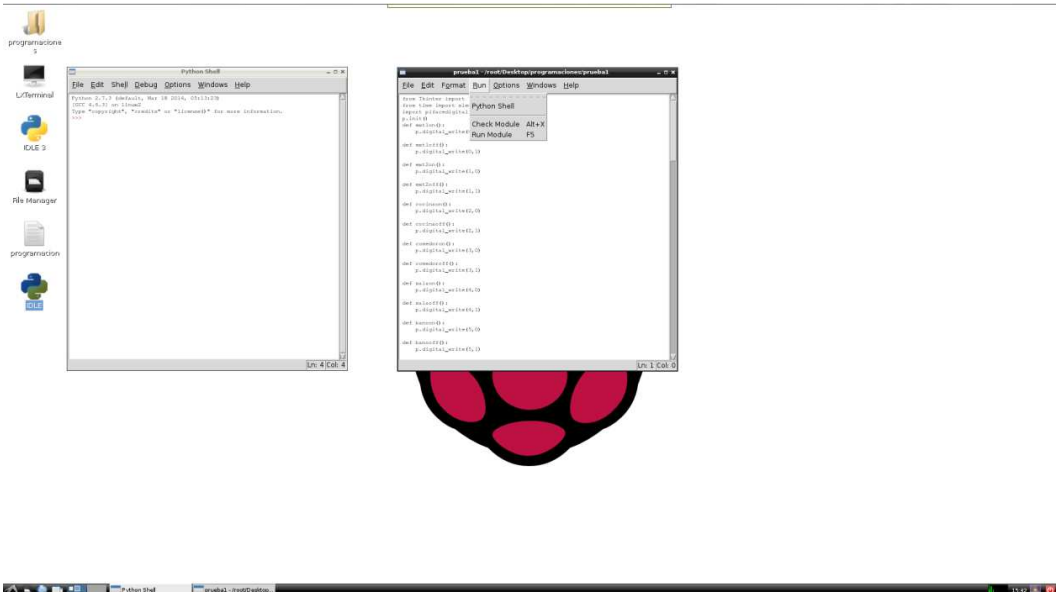


Imagen A6: Run module

Se ejecutara la interfaz que se a realizado

Imagen A7: Interfaz ejecutada

Para probar el sistema pulsamos el botón sensor lo cual nos permitirá detectar si el sensor de la puerta se encuentra abierto o cerrado

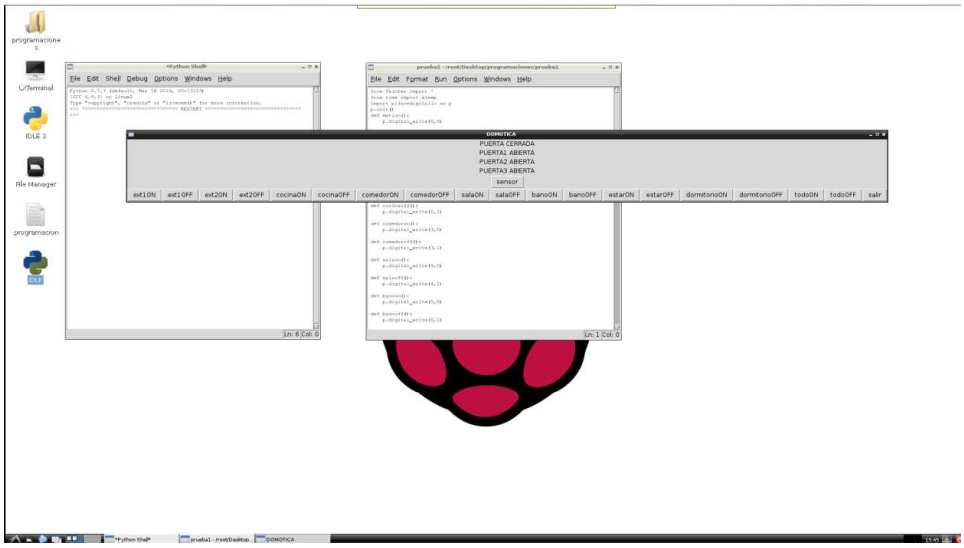


Imagen A8: Ejecucion de botones

Nota: Solo cambiara el primer label ya que se tiene un solo sensor de puerta pero se ha colocado las demás entradas que cuenta la piface ya que si se desea conectar sensores adicionales estos funcionarán sin la necesidad de realizar cambios en la programación de la raspberry

Cada luz tiene un botón de encendido y apagado al igual que cuenta con botones de todo encendido y todo apagado.

El botón salir permite cerrar el programa que se encuentra ejecutando.

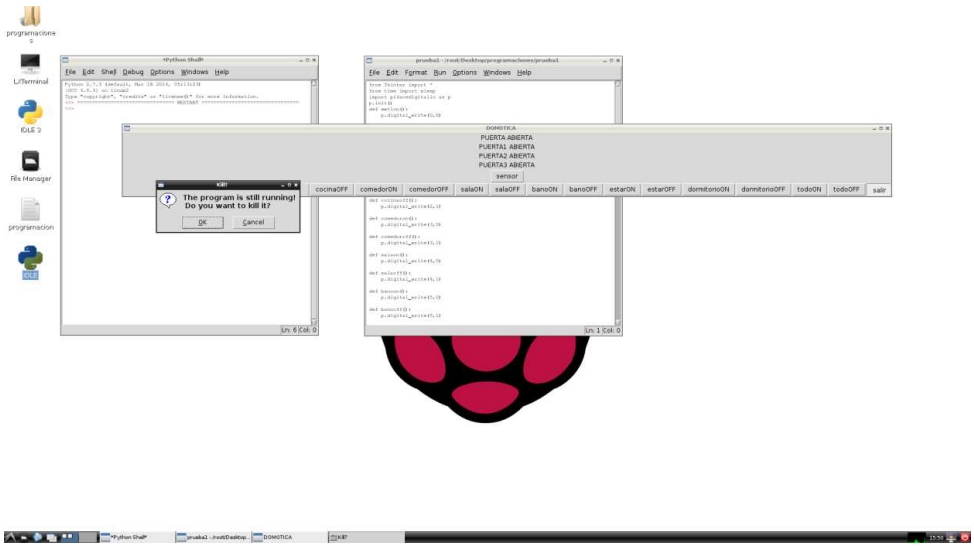


Imagen A9: Salir interfaz