

**IMPLEMENTACIÓN DE MÓDULOS DIDÁCTICOS PARA PROGRAMACIÓN DE
SISTEMAS ARDUINO**

**LUIS GERMAN BERROCAL VILLA
FREDY ALEXANDER SEPULVEDA
JOHN BAYRON MESA LONDOÑO**



**INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO
TECNOLOGÍA ELÉCTRICA
MEDELLIN
2014**

**IMPLEMENTACIÓN DE MÓDULOS DIDÁCTICOS PARA PROGRAMACIÓN DE
SISTEMAS ARDUINO**

**Trabajo de grado para optar al título de
*TECNÓLOGO ELECTRICISTA***

**ASESOR METODOLÓGICO Y TEMÁTICO
Carlos Alberto Valencia Hernández
*Ingeniero en Instrumentación y Control***



**INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO
TECNOLOGÍA ELÉCTRICA
MEDELLÍN
2014**

NOTA DE ACEPTACIÓN

Nota de Aceptación:

Firma del presidente del jurado

Firma del jurado

Firma del jurado

Medellín, noviembre 2014

DEDICATORIA

A nuestros padres, A toda la familia con Amor.

Dedicatoria especial a los docentes que con su conocimiento y experiencia contribuyeron en nuestra formación y a nuestros coequiperos, siempre luchando por alcanzar el mismo ideal.

AGRADECIMIENTO

Agradecemos a nuestros profesores que con sus aportes y directrices nos guiaron por el campo del conocimiento, en especial para el asesor metodológico y temático quien contribuyó para que sacáramos adelante este proyecto de investigación, el cual nos servirá de experiencia para la presentación de futuros proyectos en nuestra vida profesional y por ende en la vida profesional.

Expresar nuestro sincero agradecimiento a la Institución Universitaria Pascual Bravo, por tener en su pensum la materia que nos apasiona.

Por donde pasemos, dejaremos huella, generando con ello beneficios en la productividad y competitividad de las empresas.

TABLA DE CONTENIDO

	Pág.
IMPLEMENTACIÓN DE MÓDULOS DIDÁCTICOS PARA PROGRAMACIÓN DE SISTEMAS ARDUINO	1
INTRODUCCIÓN	12
1. JUSTIFICACIÓN.....	13
2. PLANTEAMIENTO DEL PROBLEMA.....	14
3. OBJETIVOS.....	15
3.1 GENERAL	15
3.2. ESPECÍFICOS	15
4. MARCO TEÓRICO	16
4.1 SISTEMA EMBEBIDO.....	16
4.1.1 Ventajas de los sistemas embebidos sobre las soluciones industriales tradicionales.....	17
4.2 ARDUINO.....	18
4.2.1 Tipos de arduino	18
4.3 RELÉ	21
5. METODOLOGÍA	24
6. RESULTADOS DEL PROYECTO.....	25
6.1 DISEÑO DE LA ESTRUCTURA.....	25
6.2 IMPLEMENTACIÓN MODULO DIDÁCTICO.....	28
6.3 REALIZACIÓN DE LAS PRUEBAS DE FUNCIONAMIENTO	31
7. CONCLUSIONES Y RECOMENDACIONES	34

BIBLIOGRAFÍA.....35

ANEXOS.....36

LISTA ANEXOS

	Pág.
ANEXO A – ARDUINO MEGA DISTRIBUCIÓN DE PINES.....	37
ANEXO B – ARDUINO UNO.....	38
ANEXO C - ARDUINO UNO DISTRIBUCIÓN DE PINES	40
ANEXO D – ARDUINO MINI.....	41
ANEXO E - PLANO MÓDULO DE CUATRO RELEVOS	42
ANEXO F - TELERUPTOR.....	43
ANEXO G - START STOP.....	44
ANEXO H - ARRANQUE SECUENCIAL (LIFO).....	45
ANEXO I - ESTRELLA DELTA	47
ANEXO J - CONTROL DE TEMPERATURA.....	49
ANEXO K- PRUEBA DEL MÓDULO	51

LISTA ILUSTRACIONES

	Pág.
Ilustración 1. Sistema embebido	16
Ilustración 2. Arduino Mega	20
Ilustración 3. Arduino Uno.....	20
Ilustración 4. Arduino Mini.....	21
Ilustración 5. Relé	23
Ilustración 6. Sensor de temperatura	23
Ilustración 7. Módulo Arduino uno entradas y salidas	26
Ilustración 8. Relé	26
Ilustración 9. Sensor de temperatura	27
Ilustración 10. Maletín de aluminio.....	27
Ilustración 11. Conector hembra tipo banana, accionamientos tipo start NA, accionamientos tipo stop NC, protoboard, tablero acrílico.....	28

LISTA FOTOGRAFÍAS

	Pág.
Fotografía 1. Inicio del ensamblaje	29
Fotografía 2. Instalación del arduino.....	29
Fotografía 3. Panel frontal casi terminado	30
Fotografía 4. Construcción de los módulos.....	30
Fotografía 5. Cableado interno del módulo arduino.	31
Fotografía 6. Arduino instalado en su maleta	31
Fotografía 7 Moduló en funcionamiento # 1.....	32
Fotografía 8. Moduló en funcionamiento # 2.....	33

EQUIPO DE TRABAJO

Luis German Berrocal Villa

Cédula 8.125.791

Carné 2012.104.097

Estudiante de Tecnología Eléctrica

John Bayron Mesa Londoño

Cédula 71.736.722

Carné 2011.204.045

Estudiante de Tecnología Eléctrica

Fredy Alexander Sepúlveda Vélez

Cédula 98.715.756

Carné 2013.2041.76

Estudiante de Tecnología Eléctrica

INTRODUCCIÓN

La tecnología es el conjunto de conocimientos técnicos, científicamente descritos y ordenados para la adaptación del hombre a su entorno; ésta le brinda las herramientas para solucionar de forma parcial o total una necesidad o un problema. La tecnología siempre ha acompañado a la humanidad a lo largo de la historia, ha permitido establecer sociedades y culturas. Los conocimientos acumulados a lo largo de las generaciones se han ido retroalimentando de tal forma que tecnologías ya establecidas han generado otras emergentes.

Los sistemas autómatas generaron una mejora en la calidad de vida del ser humano tanto en el hogar como en el ambiente laboral, han entregado confort, bienestar y seguridad. En la actualidad sería prácticamente impensable un proceso sin una etapa de automatización, está tan integrado a nuestras vidas que no lo percibimos, por ejemplo el proceso de una lavadora, un televisor, un equipo de sonido, un reloj, una nevera, entre otras.

El conocimiento y la manipulación de estos sistemas, en ámbitos académicos, se ha vuelto un tema relevante y en el caso concreto de arduino, es utilizado por personas de diferentes disciplinas, por su facilidad de programación, sencillez, robustez y bajo costo; es utilizado en sistemas de encendido y apagado de luces, en sistemas de calefacción, en dispositivos para discapacitados, en el campo del arte, en juegos de luces, controles domóticos y, en tiempos recientes, en automatización y control industrial. Convirtiéndose en una herramienta flexible que se puede utilizar en una gran cantidad de aplicaciones y procesos.

1. JUSTIFICACIÓN

La implementación de un conjunto de módulos didácticos para programación sobre sistemas arduino responde a la necesidad que presenta la Institución Universitaria Pascual Bravo de capacitar a los docentes y estudiantes en el uso de las nuevas tecnologías en las áreas de electrónica de potencia, programación, circuitos eléctricos y automatización.

Este proyecto beneficia al 16.17% de los estudiantes matriculados en la institución pertenecientes al departamento de eléctrica; también tendrá incidencia en algunas asignaturas del pensum académico tales como: control II, PLC, circuitos lógicos entre otros, lo que traerá consigo la actualización y cualificación en el rol del tecnólogo electricista Pascualino.

2. PLANTEAMIENTO DEL PROBLEMA

La Institución Universitaria Pascual Bravo no cuenta con módulos pedagógicos de sistemas embebidos que ayuden a la actualización y cualificación del rol profesional.

3. OBJETIVOS

3.1 GENERAL

- Implementar un conjunto de módulos didácticos para programación de sistemas arduino.

3.2. ESPECÍFICOS

- Diseñar la estructura de los módulos didácticos arduino.
- Implementar módulos didácticos, prácticos y pedagógicos.
- Realizar pruebas de funcionamiento

4. MARCO TEÓRICO

4.1 SISTEMA EMBEBIDO

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

En un sistema embebido la mayoría de los componentes se encuentran incluidos en la placa base (la tarjeta de vídeo, audio, módem) y muchas veces los dispositivos resultantes no tienen el aspecto de lo que se suele asociar a una computadora. Algunos ejemplos de sistemas embebidos podrían ser dispositivos como un taxímetro, un sistema de control de acceso, la electrónica que controla una máquina expendedora o el sistema de control de una fotocopiadora, entre otras múltiples aplicaciones¹. (Class y Altadill Izura)

Ilustración 1. Sistema embebido



Fuente: Imagen tomada de <http://ramon-gzz.blogspot.com/2012/02/introduccion-sistemas-integrados.html>).

¹ Fundación Wikipedia. Sistema embebido. (En línea). http://es.wikipedia.org/wiki/Sistema_embellido [Citado en noviembre 4 de 2014].

Por lo general los sistemas embebidos se pueden programar directamente en el lenguaje ensamblador del micro-controlador o microprocesador incorporado sobre el mismo, o también, utilizando los compiladores específicos, pueden utilizarse lenguajes como C o C++; en algunos casos, cuando el tiempo de respuesta de la aplicación no es un factor crítico, también pueden usarse lenguajes interpretados como JAVA.

Puesto que los sistemas embebidos se pueden fabricar por decenas de millares o por millones de unidades, una de las principales preocupaciones es reducir los costes.

4.1.1 Ventajas de los sistemas embebidos sobre las soluciones industriales tradicionales. Los equipos industriales de medida y control tradicionales están basados en un microprocesador con un sistema operativo privativo o específico para la aplicación correspondiente. Dicha aplicación se programa en ensamblador para el microprocesador dado o en lenguaje C, realizando llamadas a las funciones básicas de ese sistema operativo que en ciertos casos ni siquiera llega a existir. Con los modernos sistemas de cómputo se llega a integrar el mundo del computador personal compatible con las aplicaciones industriales. Ello implica numerosas ventajas:

- Posibilidad de utilización de sistemas operativos potentes que ya realizan numerosas tareas: comunicaciones por redes de datos, soporte gráfico, concurrencia con lanzamiento de threads, entre otros. Estos sistemas operativos pueden ser los mismos utilizados para computador personal compatibles (Linux, Windows, MS-DOS) con fuertes exigencias en hardware o bien ser una versión reducida de los mismos con características orientadas a los computadores personales embebidos.
- Al utilizar dichos sistemas operativos se pueden encontrar fácilmente herramientas de desarrollo software potentes así como numerosos programadores que las dominan, dada la extensión mundial de las aplicaciones para computador personal compatibles.
- Reducción en el precio de los componentes hardware y software debido a la gran cantidad de computadores personales en el mundo².

² Fundación Wikipedia. Sistema embebido. (En línea).
http://es.wikipedia.org/wiki/Sistema_embebido [Citado en noviembre 4 de 2014].

4.2 ARDUINO

Arduino es una plataforma de hardware libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280, ATmega8 por su sencillez y bajo coste que permiten el desarrollo de múltiples diseños. Por otro lado el software consiste en un entorno de desarrollo que implementa el lenguaje de programación Processing/Wiring y el cargador de arranque que es ejecutado en la placa³.

4.2.1 Tipos de arduino

- **Arduino mega 2560:** Está basada en el microcontrolador ATmega2560, esta tarjeta es la sucesora de la Arduino MEGA1280 la tarjeta arduino con mayor número de puertos de propósito general, Tiene 54 I/O digital de los cuales se pueden usar 14 PWMs, 16 canales de entradas análogas, 4 UARTs (hardware serial ports), a 16 MHz, integra una conexión a USB, entrada de alimentación tipo Jack y conector ICSP, su uso es muy sencillo, solo se requiere conectarla al computador por medio del cable USB. Aplicaciones: Debido a que los sistemas arduino viene listos para usar se puede aplicar en múltiples campos de la electrónica y control como, sensorica, robótica, dispositivos de visualización entre otros. No requiere programador externo, para programarlo basta con conectar la tarjeta por medio del puerto USB, tener listo el programa arduino y pulsar la opción de Upload⁴. (educativa)

La principal diferencia entre la ARDUINO MEGA 2560 y la ARDUINO MEGA es el uso del ATMEGA8U2 como interface USB, lo que le permite una comunicación más rápida, para entorno Windows (Más información ver anexo 1)⁵.

- **Arduino uno:** es la versión mejorada de su predecesor Duemilanove. Incluye función de autoreset, protección de sobrecargas, conector USB para programarlo, totalmente montado con componentes miniatura SMD

³ Fundación Wikipedia. Arduino. (En línea). <http://es.wikipedia.org/wiki/Arduino>
[Citado en noviembre 12 de 2014].

⁴ Arduino. Arduino Mega. (En línea). <http://arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardMega>
[Citado en 2014].

⁵ Dinamo electronics. Arduino mega 2560. (en línea).
http://www.dynamoelectronics.com/index.php?product_id=917&page=shop.product_details&category_id=82&flypage=dynamo.tpl&option=com_virtuemart&Itemid=58.

(salvo el microcontrolador, para poder cambiarlo fácilmente) y nuevo bootloader OptiBoot a 155kbps⁶.

La placa se entrega completamente ensamblada y probada con un microcontrolador AVR ATmega328 con un cristal de cuarzo de 16Mhz. El microcontrolador se entrega con un bootloader que permite su programación sin necesidad de ningún tipo de programado externo (Más información ver anexo 2)⁷. (Cadiz)

- **Arduino mini:** es una placa electrónica pequeña originalmente basado en el ATmega168, pero ahora suministra con el 328, destinado a circular por placas universales y cuando el espacio es un bien escaso. Cuenta con 14 pines digitales de entrada/salida (de los cuales 6 pueden utilizarse para salidas PWM), 8 entradas analógicas, y un oscilador de cristal de 16 MHz. Se puede programar con el adaptador de serie USB u otro USB o RS232 a TTL adaptador serie⁸.

Cada uno de los 14 pines digitales en el Mini se puede utilizar como una entrada o salida. Operan a 5 voltios. Cada pin puede proporcionar o recibir un máximo de 40 mA y tiene una resistencia pull-up interna (desconectada por defecto) de 20-50 KΩ. Pasadores 3, 5, 6, 9, 10, y 11 pueden proporcionar una salida PWM. En todo caso, además de la Mini USB (u otro) adaptador está conectado a los pines 0 y 1, que interfiere con la comunicación USB, evitando nuevo código sea cargado u otra comunicación con el ordenador (Más información ver anexo 4)⁹. (Pomares Baeza)

⁶ Menos media y más comunicación. Placa arduino uno. (En línea). <http://www.menosmedia.org/spip.php?article43>. Diciembre 2012

⁷ Arduino. Arduino uno. (En línea). <http://arduino.cc/en/Main/arduinoBoardUno> [Citado en 2014].

⁸ Arduino. Arduino uno. (En línea). <http://arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardMini> [Citado en 2014].

⁹ Tienda bricogeek. Arduino Pro Mini 328 - 5V/16MHz. (En línea). <http://tienda.bricogeek.com/arduino/233-arduino-pro-mini-328-5v-16mhz.html>

Ilustración 2. Arduino Mega



Fuente: Imagen tomada de <http://arduino.cc/es/pmwiki.php?n=Main/ArduinoBoardMega>

Ilustración 3. Arduino Uno



Fuente: Imagen tomada de http://arduino.cc/en/uploads/Main/ArduinoUno_R3_Front_450px.jpg

Ilustración 4. Arduino Mini

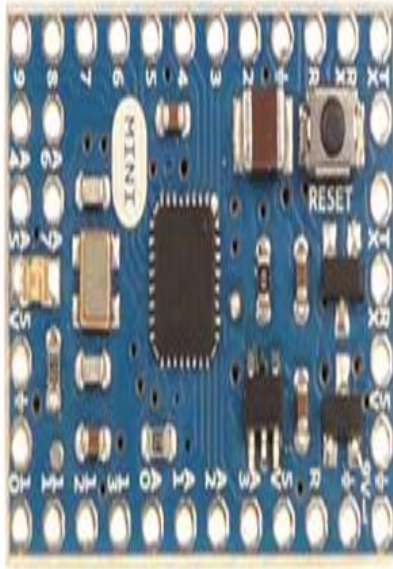


Imagen tomada de http://arduino.cc/en/uploads/Main/Mini05_front_450px.jpg

4.3 RELÉ

El relé o relevador es un dispositivo electromecánico. Funciona como un interruptor controlado por un circuito eléctrico en el que, por medio de una bobina y un electroimán, se acciona un juego de uno o varios contactos que permiten abrir o cerrar otros circuitos eléctricos independientes. Fue inventado por Joseph Henry en 1835¹⁰.

Dado que el relé es capaz de controlar un circuito de salida de mayor potencia que el de entrada, puede considerarse, en un amplio sentido, como un amplificador eléctrico. Como tal se emplearon en telegrafía, haciendo la función de repetidores que generaban una nueva señal con corriente procedente de pilas locales a partir de la señal débil recibida por la línea. Se les llamaba "relevadores". De ahí "relé".

- **Relé srd-05vdc-sl-c:** este tipo de relé cuenta con las siguientes bobdades y/o características:

¹⁰ Fundación Wikipedia. Relé. (En línea). <http://es.wikipedia.org/wiki/Rel%C3%A9>. [Citado en noviembre 20 de 2014].

- Carga nominal: 10A 250VAC/28VDC, 10A 125VAC/28VDC, 10^a 125VAC/28VDC
- Vida eléctrica: 100.000
- Vida mecánica: 10.000.000
- Bobina Tensión nominal: 3-48VDC
- Potencia de la bobina: 0.36W, 0.45W
- Tipo de montaje: PCB¹¹

5.4 SENSOR DE TEMPERATURA

Un sensor de temperatura es un dispositivo capaz de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y transformarlas en variables eléctricas. Las variables de instrumentación pueden ser por ejemplo: temperatura, intensidad lumínica, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, movimiento, pH, entre otros; las variables eléctricas son corriente y voltaje¹².

- **LM 35:** es un sensor de temperatura integrado de precisión, cuya tensión de salida es linealmente proporcional a temperatura en °C (grados centígrados). Este sensor tiene una ventaja sobre los sensores de temperatura lineal calibrada en grados Kelvin ya que el usuario no está obligado a restar una gran tensión constante para obtener grados centígrados.
- El LM35 no requiere ninguna calibración externa o ajuste para proporcionar una precisión típica de ± 1.4 °C a temperatura ambiente y ± 3.4 °C a lo largo de su rango de temperatura (de -55 a 150 °C). El dispositivo se ajusta y calibra durante el proceso de producción. La baja impedancia de salida, la salida lineal y la precisa calibración inherente, permiten la creación de circuitos de lectura o control especialmente sencillos. El LM35 puede funcionar con alimentación simple o alimentación doble (+ y -).

Requiere sólo 60 μ A para alimentarse, y bajo factor de auto-calentamiento, menos de 0,1 °C en aire estático. El LM35 está preparado para trabajar en una gama de temperaturas que abarca desde los- 55 °C bajo cero a 150 °C,

¹¹ Matius. Rele miniatura alta potencia SRD-12VDC-SL-C. (En línea). <http://www.matius.net/etienda/rele-miniatura-alta-potencia-srd12vdcslc-p-1224.html>. [Citado en noviembre 23 de 2014].

¹² Fundación Wikipedia. Sensor. (En línea). <http://es.wikipedia.org/wiki/Sensor>. [Citado en noviembre 22 de 2014].

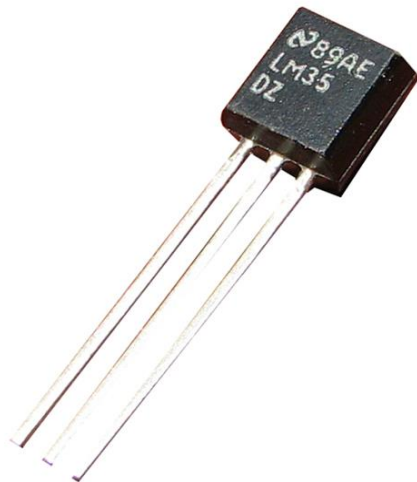
mientras que el LM35C está preparado para trabajar entre $-40\text{ }^{\circ}\text{C}$ y $110\text{ }^{\circ}\text{C}$ (con mayor precisión)¹³.

Ilustración 5. Relé



Fuente: imagen tomada de http://www.satistronics.com/mini-5v-dc-power-relay-srd5vdcslc-songle_p2159.html

Ilustración 6. Sensor de temperatura



Fuente: Imagen tomada de <https://www.google.com.co>

¹³ Página de electrónica de Carlos Díaz. LM35 - Sensor de temperatura de precisión. (En línea). <http://electronica.webcindario.com/componentes/lm35.htm>.

5. METODOLOGÍA

Para comenzar con la ejecución de este trabajo de grado se debe convenir con el asesor del proyecto los días y horarios en los cuales se llevarán a cabo los espacios de reunión; con la finalidad de comenzar la planeación y definir la estrategia para la puesta en marcha del trabajo, además de establecer el cronograma de actividades.

El programa de tecnología eléctrica brinda a sus estudiantes los conocimientos, herramientas y destrezas para la ejecución de este proyecto que consiste en la implementación de módulos didácticos para programación de sistemas arduino, que serán suministrados al laboratorio del departamento de eléctrica para la utilidad de los tecnólogos en formación de la Institución Universitaria Pascual Bravo.

El proyecto se desarrollará en las siguientes etapas:

Etapas 1: Preparación

- Revisión de ciber-grafía sobre información relacionada con los sistemas embebidos y arduino.
- Selección del tipo de arduino a implementar.
- Establecimiento de cronograma de actividades para el desarrollo del proyecto.

Etapas 2: Diseño y ejecución

- Realización de pruebas de funcionamiento del conjunto de módulos arduino en el laboratorio del departamento de eléctrica de la Institución Universitaria Pascual bravo.
- Diseño y construcción de módulos didácticos para programación de sistemas arduino el laboratorio del departamento de eléctrica de la Institución Universitaria Pascual bravo.

Etapas 3: Evaluación y socialización

- Evaluación del funcionamiento del conjunto de módulos arduino en el laboratorio del departamento de eléctrica de la Institución Universitaria Pascual bravo.
- Socialización del producto final al asesor de práctica.

6. RESULTADOS DEL PROYECTO

En este capítulo se revisarán las diferentes etapas que se llevaron a cabo para la implementación del conjunto de módulos didácticos.

6.1 DISEÑO DE LA ESTRUCTURA

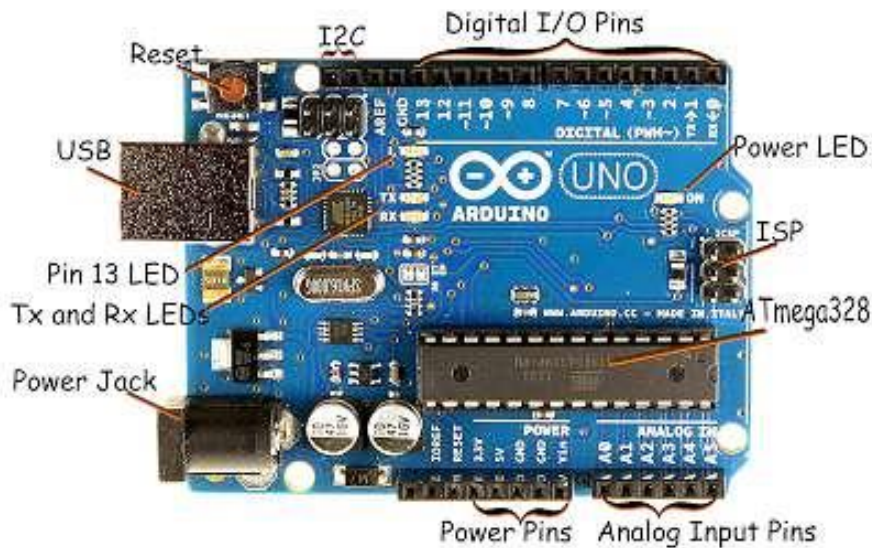
El modulo didáctico será concebido de forma que cumpla los siguientes requerimientos: que sea portable y de fácil transporte, que brinde seguridad para cada uno de sus componentes, que se pueda utilizar en diferentes carreras, que sea amigable con el usuario, que sus componentes sean de fácil adquisición, debe tener su propia fuente de alimentación, que estén identificados sus componentes, que se puedan implementar procesos con manejo de potencia industrial (220 AC, 5A) y por último que se pueda censar variables como la temperatura.

Teniendo en cuenta lo anterior se eligieron los siguientes componentes:

Módulo arduino, se seleccionó el módulo Arduino Uno R3 por las siguientes razones

- Posee un buen desempeño a bajo coste comparado con otros sistemas embebidos como BeagleBone, Raspberry Pi.
- Es el muy fácil de adquirir a nivel local.
- En caso de daño, a diferencia de otros modelos no hay que cambiar toda la tarjeta, solo el chip.
- Cuenta con cinco entradas analógicas y 14 entradas/salidas digitales que son suficientes para los diferentes proyectos planteados en la Institución Universitaria Pascual Bravo.

Ilustración 7. Módulo Arduino uno entradas y salidas



Fuente: <http://www.arduinoclassroom.com/images/A101C1/Figure%201%20-%20Arduino%20Uno%20R3-400.jpg>

Relé, se seleccionó un módulo de cuatro relevos (referencia srd-05vdc-sl-c) por que cuenta con las siguientes características.

- Voltaje máximo en la carga de 250VAC.
- Corriente máxima en la carga 10A.
- Voltaje de operación 5VDC.
- Potencia de bobina 0.36W

El voltaje y la corriente soportados por el relé son suficientes para energizar una amplia gama de contactores ya que la corriente de llamada de este es de 1.5A y la de trabajo es de 90mA. Y su voltaje de operación 220VAC.

Ilustración 8. Relé

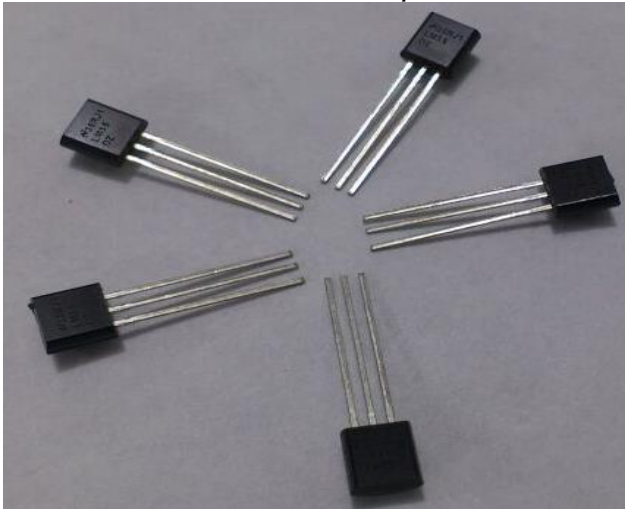


Fuente: Los autores

Sensor de temperatura, se seleccionó el sensor de temperatura LM35, es ideal para los proyectos en laboratorio por su amplio rango de temperatura, es de fácil adquisición en el mercado, cuenta con las siguientes características:

- Rango de temperatura de -55°C hasta 150°C
- Medición directa en grados Celsius.
- Rango de alimentación entre 4V hasta 20V que cumple con la suministrada por el arduino (5V).
- Suministra una salida de comportamiento lineal que puede ser interpretada por el módulo arduino.

Ilustración 9. Sensor de temperatura



Fuente: Los autores

Ilustración 10. Maletín de aluminio

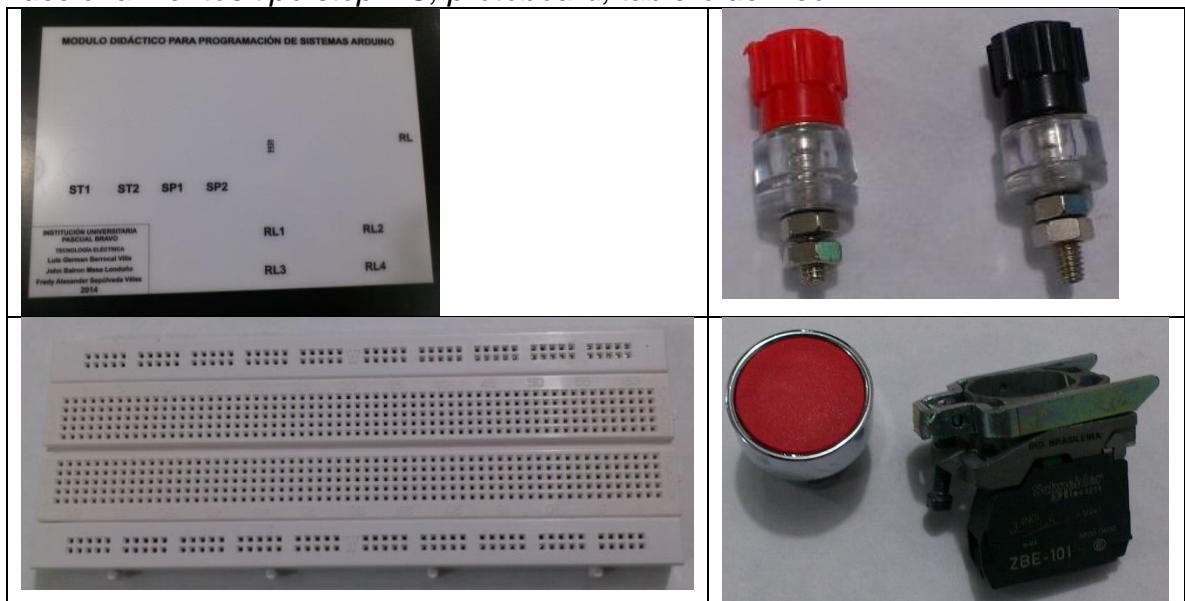


Fuente: Los autores

Maletín de aluminio, sus dimensiones son treinta y siete centímetros de largo veintiséis de ancho y trece de profundidad. Contendrá el tablero en acrílico y sus componentes además la circuitería interna necesaria para su funcionamiento, se seleccionó un maletín porque facilita el transporte del módulo y protegerá sus componentes además cuenta con un toma hembra lateral.

Además se le agregaron 12 salidas hembra tipo banana (tres por rele) para conectar los diferentes equipos en la universidad, incluiremos dos accionamientos tipo start NA y dos accionamientos tipo stop NC. Para comandar con ellos diversos ejemplos de control. Se incluirá una protoboard con el fin de que sea flexible y contribuya a la ejecución de otras ideas de proyecto, Un tablero acrílico que soportara los el arduino, el módulo de relés, los pulsadores, la protoboard y las salidas tipo banana, estará rotulada para la adecuada identificación de los componentes empotrados en ella.

Ilustración 11. Conector hembra tipo banana, accionamientos tipo start NA, accionamientos tipo stop NC, protoboard, tablero acrílico.



Fuente: Los autores

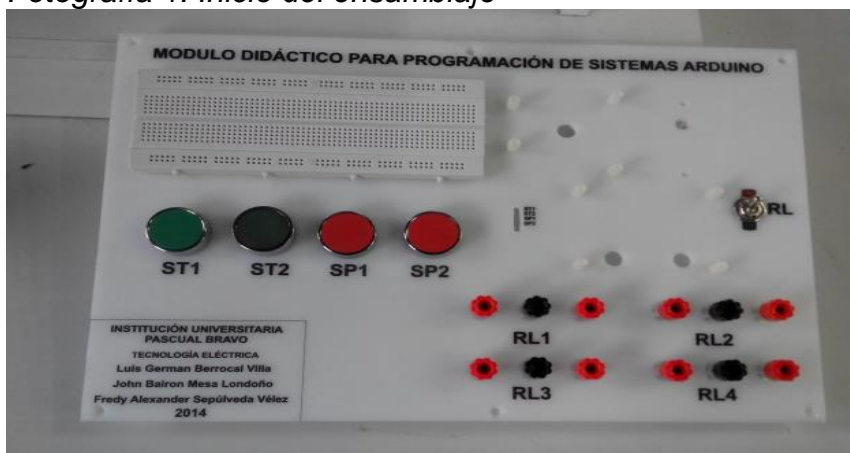
6.2 IMPLEMENTACIÓN MODULO DIDÁCTICO

Se compraron los maletines según la medida de estos se diseñaron los montajes en los acrílicos para luego mandarlos a elaborar en una empresa especializada. Luego de entregados los acrílicos se realizaron las perforaciones pertinentes.

Para cada uno de los componentes, se consiguieron los componentes y se instalaron en los módulos con tornillos espaciadores, la protoboard queda adherida al acrílico, además se instalaron las terminales tipo banana y en la parte trasera del acrílico se adiciono las terminales que salen de los pulsadores.

Se instalaron en el acrílico los pulsadores industriales. Se perforo el maletín para adicionarle la clavija de entrada de potencia, se cableo el acrílico por su parte trasera, los relevos se cablearon con cable vehicular, se instalaron los soportes en el maletín para asegurar el acrílico se adecuo el sistema eléctrico en el maletín para luego atornillar la placa acrílica en el interior de los módulos.

Fotografía 1. Inicio del ensamblaje



Fuente: Los autores

Fotografía 2. Instalación del arduino



Fuente: Los autores

Fotografía 3. Panel frontal casi terminado



Fuente: Los autores

Fotografía 4. Construcción de los módulos



Fuente: los autores

Fotografía 5. Cableado interno del módulo arduino.



Fuente: Los autores

Fotografía 6. Arduino instalado en su maleta



Fuente: Los autores

6.3 REALIZACIÓN DE LAS PRUEBAS DE FUNCIONAMIENTO

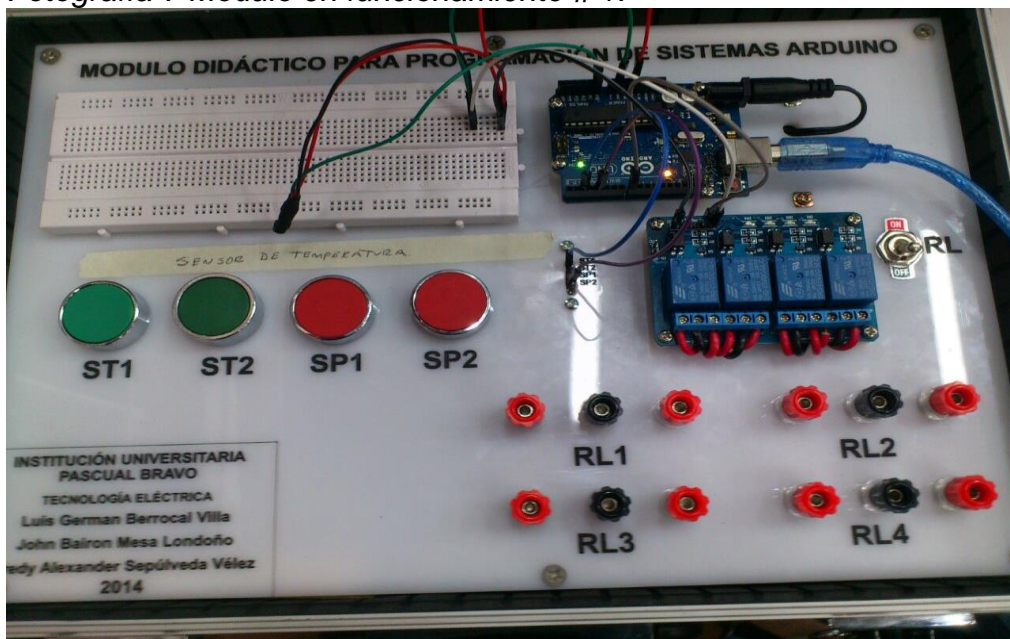
Para probar los módulos es necesario realizar primero el proceso de programación. El software de programación oficial de arduino Arduino IDE se puede descargar de la página oficial (<http://arduino.cc/>), se estableció como

prueba un algoritmo que pruebe todos los pulsadores, el módulo arduino y todas las salidas de los relevos el algoritmo funciona de la siguiente manera.

Al presionar cualquier botón ST (botones verdes) todos los relevos se activaran y al presionar cualquier SP (botones rojos) todos los relevos se desactivaran.

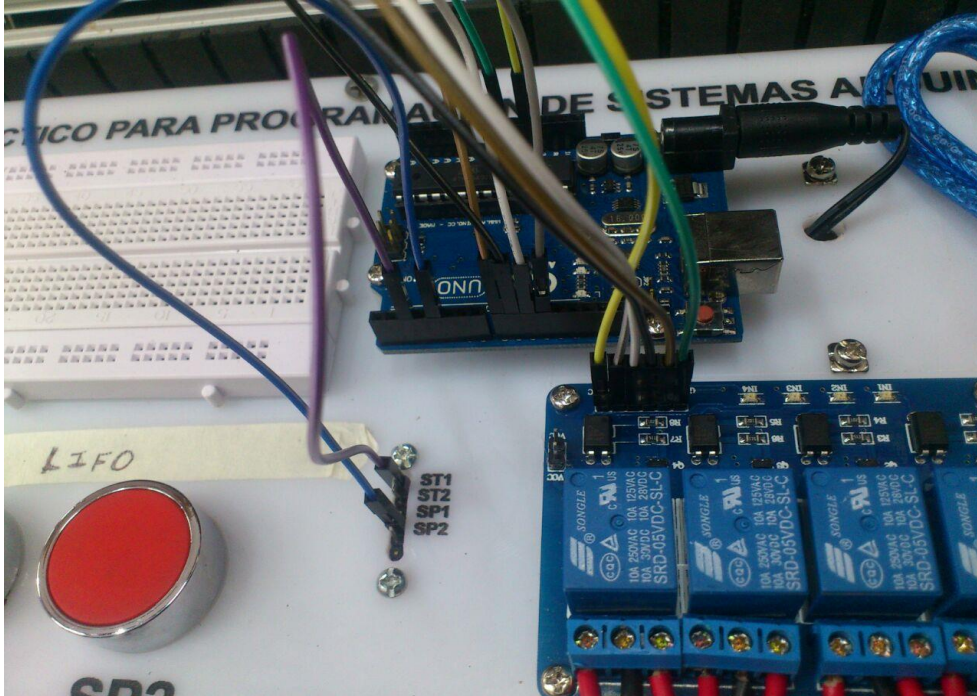
Se procedió a cargar el modulo con el algoritmo propuesto y se realizaron las pruebas se revisó el buen funcionamiento del módulo, los accionamientos y se verifico la continuidad en los relevos los, los resultados son los esperados.

Fotografía 7 Moduló en funcionamiento # 1.



Fuente: Los autores

Fotografía 8. Moduló en funcionamiento # 2.



Fuente: Los autores

7. CONCLUSIONES Y RECOMENDACIONES

- La revisión cibergráfica permitió el estudio de software de programación del dispositivo y se pudo crear un concepto claro de cómo programarlo.
- A través del desarrollo del proyecto se evidencio la utilidad de arduino para la enseñanza en diferentes institutos educativos.
- Este proyecto ayuda a la cualificación del profesional para que tenga las herramientas adecuadas y actualizadas que estén a la vanguardia de la tecnología, esto hace que el profesional tenga más fundamentos a la hora de incursionar en el campo laboral.
- Es importante recordar las recomendaciones al respecto de los pines configurados como salida (OUTPUT) con pinMode se dice que están, en estado de baja impedancia. Esto significa que puede proporcionar una cantidad sustancial de corriente a otros circuitos. Los pines del Atmega pueden proporcionar corriente positiva o proporcionar corriente negativa de hasta 40 mA (miliamperios) a otros dispositivos o circuitos. Esta es suficiente corriente para la brillante luz de un LED (no se olvide de la resistencia en serie) o para utilizar muchos sensores por ejemplo; sin embargo, no es conveniente utilizar las salidas directamente en los dispositivos con inductancias, como relés, solenoides o motores.
- Los cortocircuitos en los pines de Arduino o intentos de extraer mucha corriente de ellos, pueden dañar o destruir los transistores de salida en el pin, pudiendo dañar completamente el chip Atmega. A menudo, esto se traducirá en un pin del microcontrolador "muerto", pero el resto del chip seguirá funcionando adecuadamente. Por esta razón es buena idea conectar los pines de salida a otros dispositivos con resistencias de 470Ω o 1k, limitando la corriente máxima que desde los pines es requerida para una aplicación particular.
- Se debe tener presente que los botones de paro (stop) son de configuración NC y los de marcha (start) son de configuración NA
- Para que un relevo en el módulo de relevos se active la salida digital del arduino se debe configurar en (LOW) y para desactivarlo la salida digital se debe configurar en (HIGH). Esto quiere decir que si hay voltaje en la salida del módulo arduino el relé se desactivara y si hay ausencia de voltaje se activara.

BIBLIOGRAFÍA

- Cádiz, Universidad de. «[www.arduino.cc/en/Practica_1:Comenzando con Arduino.](http://www.arduino.cc/en/Practica_1:Comenzando_con_Arduino)» s.f. http://www.uca.es/recursos/doc/Unidad_Innovacion/Innovacion_Docente/ANEXOS_2011_2012/22232441_310201212102.pdf. 27 de 11 de 014. <<http://www.uca.es/recursos/doc/>>.
- Class, Peter y Pello Xabier Altadill Izura. «Tutorial de C++ o el diario de Peter Class.» *Tutorial de C++*. 19 de 11 de 2004. 27 de 11 de 2014. <<http://es.tldp.org/Manuales-LuCAS/doc-tutorial-c++/doc-tutorial-c++.pdf>>.
- Educativa, Robótica. «Libro Básico de Arduino.» *libro-en-pdf-arduino-basico-tienda-de.html*. Primera. Bogotá, Abril de 2012. 27 de 11 de 2014. <<http://ludicaconarduino.blogspot.com>>.
- Pomares Baeza, Jorge. «Manual de programación arduino.» *Manual de arduino*. Ed. Universitat d'Alacant. Prod. Grupo de Innovación Educativa en Automática. Alicante, 2009. <<http://rua.ua.es/>>.

ANEXOS

ANEXO A – ARDUINO MEGA DISTRIBUCIÓN DE PINES

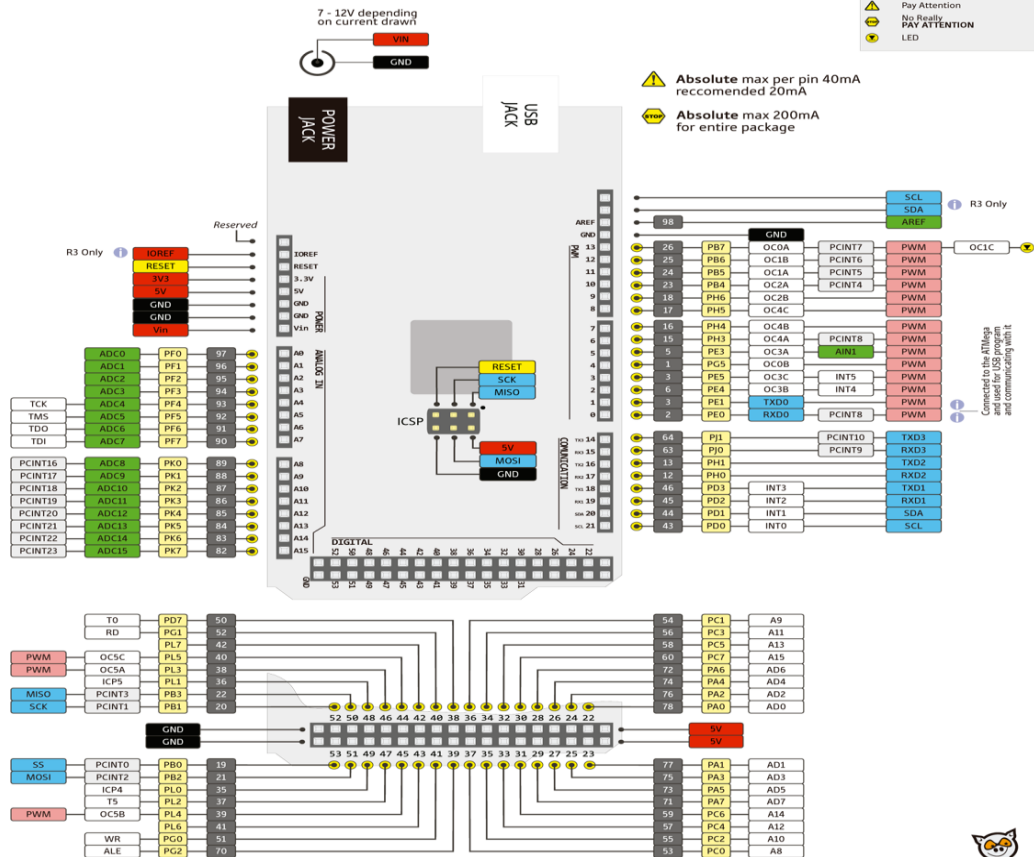
THE UNOFFICIAL
ARDUINO MEGA
PINOUT DIAGRAM

LEGEND

- GND GND
- POWER POWER
- CONTROL CONTROL
- PHYSICAL PIN PHYSICAL PIN
- PORT PIN PORT PIN
- ATMEGA PIN FUNC ATMEGA PIN FUNC
- DIGITAL PIN DIGITAL PIN
- ANALOG-RELATED PIN ANALOG-RELATED PIN
- PWM PIN PWM PIN
- SERIAL PIN SERIAL PIN

General Information

- ! Pay Attention
- ! Not Really
- ! PAY ATTENTION
- ! LED



www.pigbiass.com
CC BY-NC-SA
30 JAN 2013

Fuente: Imagen tomada de <http://spainlabs.com>

Características técnicas del arduino mega:

- **Procesador:** ATmega2560
- **Velocidad de reloj:** 16Mhz
- **Puertos I/O:** 54
- **Memoria Flash:** 256K
- **Modo de programación:** USB por software arduino.
- **Voltaje de entrada:** 7-12Vdc
- **Voltaje de operación:** 5Vdc

ANEXO B – ARDUINO UNO

Dispone de 14 pines que pueden configurarse como entrada o salida y a los que puede conectarse cualquier dispositivo que sea capaz de transmitir o recibir señales digitales de 0 y 5 V.

Además dispone de entradas y salidas analógicas. Mediante las entradas analógicas podemos obtener datos de sensores en forma de variaciones continuas de un voltaje. Las salidas analógicas suelen utilizarse para enviar señales de control en forma de señales PWM.

Arduino UNO es la última versión de la placa, existen dos variantes, la Arduino UNO convencional y la Arduino UNO SMD. La única diferencia entre ambas es el tipo de microcontrolador que montan¹⁴.

Cada uno de los 14 pines digitales se puede usar como entrada o como salida. Funcionan a 5V, cada pin puede suministrar hasta 40 mA. La intensidad máxima de entrada también es de 40 mA.

Cada uno de los pines digitales dispone de una resistencia de pull-up interna de entre 20KΩ y 50 KΩ que está desconectada, salvo que nosotros indiquemos lo contrario.

Arduino también dispone de 6 pines de entrada analógicos que trasladan las señales a un convertor analógico/digital de 10 bits.

Pines especiales de entrada y salida:

- RX y TX: Se usan para transmisiones serie de señales TTL.
- Interrupciones externas: Los pines 2 y 3 están configurados para generar una interrupción en el atmega. Las interrupciones pueden dispararse cuando se encuentra un valor bajo en estas entradas y con flancos de subida o bajada de la entrada.
- PWM: Arduino dispone de 6 salidas destinadas a la generación de señales PWM de hasta 8 bits.

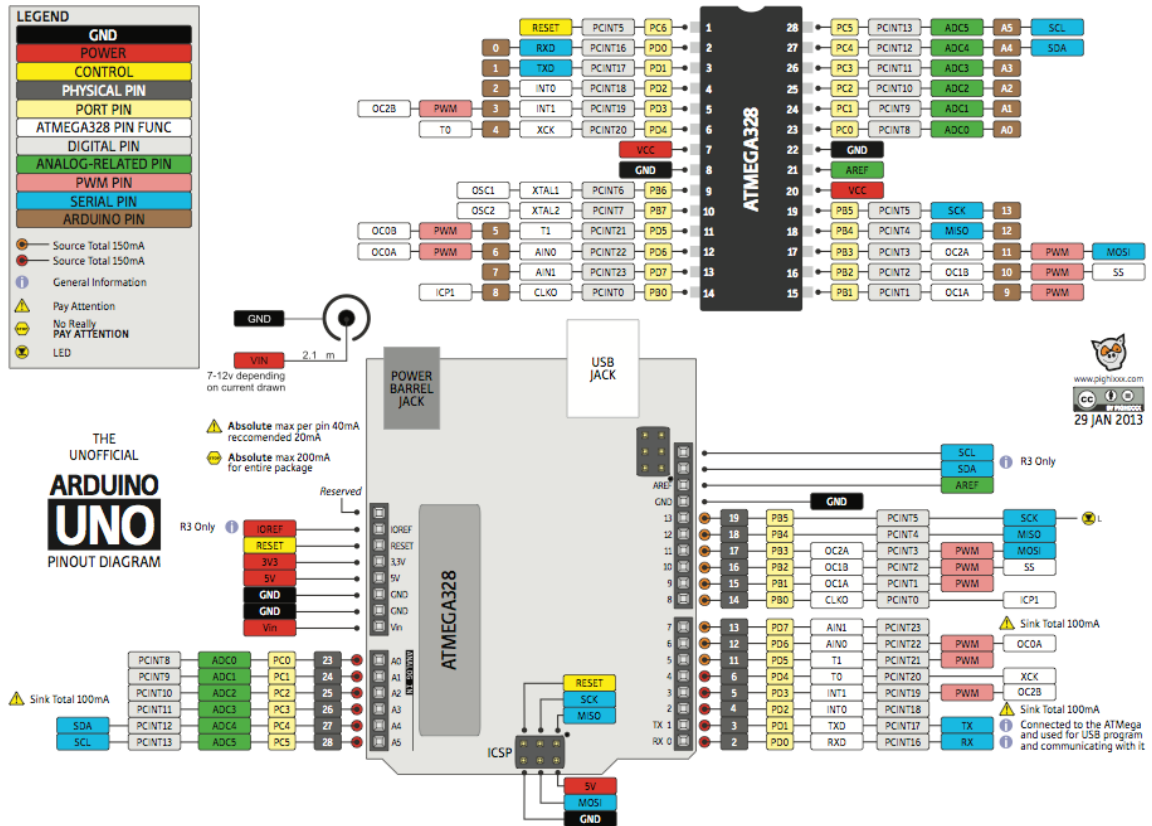
¹⁴ Arduino. Arduino uno. (En línea). <http://arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardMini> [Citado en 2014].

- SPI: Los pines 10, 11, 12 y 13 pueden utilizarse para llevar a cabo comunicaciones SPI, que permiten trasladar información full dúplex en un entorno Maestro/Esclavo.
- I2C: Permite establecer comunicaciones a través de un bus I2C. El bus I2C es un producto de Phillips para interconexión de sistemas embebidos. Actualmente se puede encontrar una gran diversidad de dispositivos que utilizan esta interfaz, desde pantallas LCD, memorias EEPROM y sensores¹⁵.

¹⁵ Menos media y más comunicación. Placa arduino uno. (En línea). <http://www.menosmedia.org/spip.php?article43>. Diciembre 2012.

¹⁵ Arduino. Arduino uno. (En línea). <http://arduino.cc/en/Main/arduinoBoardUno> [Citado en 2014].

ANEXO C - ARDUINO UNO DISTRIBUCIÓN DE PINES

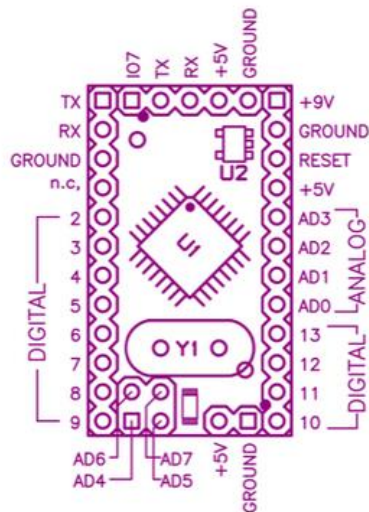


Fuente: Imagen tomada de http://www.arduteka.com/wp-content/uploads/2013/02/Arduino_Uno_pinout.png

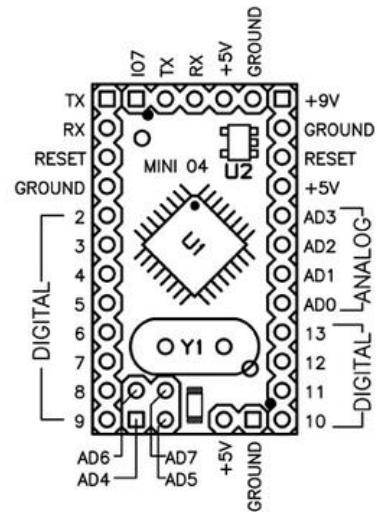
Características técnicas del arduino uno:

- **Microcontrolador:** ATmega328 Voltage: 5V
- **Voltage entrada** (recomendado): 7-12V
- **Voltage entrada** (limites): 6-20V
- **Digital I/O Pins:** 14 (de los cuales 6 son salida PWM)
- **Entradas Analógicas:** 6
- **DC Current per I/O Pin:** 40 mA
- **DC Current 3.3V Pin:** 50 mA
- **Flash Memory:** 32 KB (ATmega328) de los cuales 0.5 KB son utilizados para el arranque
- **SRAM:** 2 KB (ATmega328)
- **EEPROM:** 1 KB (ATmega328)
- **Clock Speed:** 16 MHz

ANEXO D – ARDUINO MINI



Pin out of the Arduino Mini 03. (older versions are compatible, but missing the IO7 header at the top)



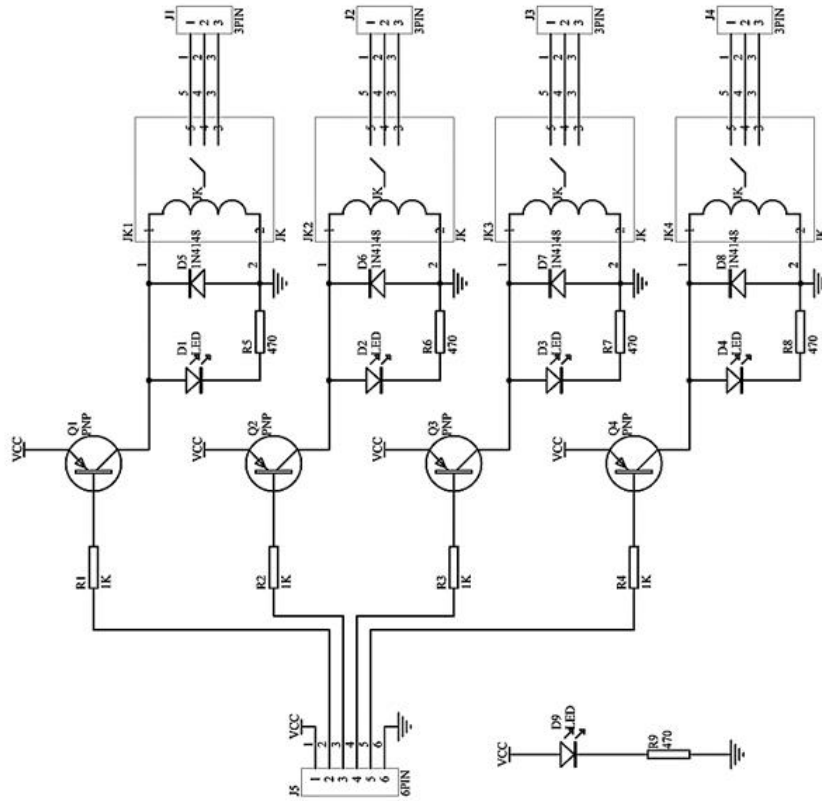
Pin out of the Arduino Mini 04 and 05. (Note that the ground pin on the left has moved down one pin.)

Fuente: Ilustración Arduino Mini distribución de pines

Características del arduino mini:

- Chip ATmega328 a 16MHz con cristal de cuarzo externo (tolerancia: 0.5%)
- Auto-reset
- Regulador 5V integrado
- Max: 150mA por salida
- Protección de sobrecarga
- Protección contra inversión de polaridad
- Entrada DC de 5V hasta 12V
- LED de power y estado

ANEXO E - PLANO MÓDULO DE CUATRO RELEVOS



Fuente: <http://www.yourduino.com/docs/4-relay-shield-diagram.jpg>

ANEXO F - TELERUPTOR

El siguiente código consiste en prender y apagar un equipo con un solo botón.

```
int pulsador=2;
int led=8;
int cnt_etapa=0; //contador de etapas y estado//
void setup(){
  pinMode(pulsador,INPUT);
  pinMode(led,OUTPUT);
  digitalWrite(led,HIGH);
}
void loop(){
  if(digitalRead(pulsador)==HIGH&&cnt_etapa==0){ //detecta pulsador on
    digitalWrite(led,LOW);
    cnt_etapa=1;
    delay(500);
  }
  if(digitalRead(pulsador)==LOW&&cnt_etapa==1){ //detecta pulsador off
    cnt_etapa=2;
    delay(500);
  }
  if(digitalRead(pulsador)==HIGH && cnt_etapa==2){ //detecta pulsador on
    digitalWrite(led,HIGH);
    cnt_etapa=3;
    delay(500);
  }
  if(digitalRead(pulsador)==LOW && cnt_etapa==3){ //detecta pulsador off
    cnt_etapa=0;
    delay(500);
  }
}
```

ANEXO G - START STOP

El siguiente código consiste encender un equipo con un start y detenerlo con un stop.

```
int arranque=2,paro=4,salida=8;
void setup()
{
  pinMode(arranque,INPUT);
  pinMode(paro,INPUT);
  pinMode(salida,OUTPUT);
  digitalWrite(salida,HIGH);
}
void loop()
{
  if(digitalRead(arranque) && digitalRead(paro))
    digitalWrite(salida, LOW);
  if(digitalRead(arranque) && !digitalRead(paro))
  {
    digitalWrite(salida,HIGH);
    delay(1000);
  }
  if(!digitalRead(paro))
    digitalWrite(salida,HIGH);
}
```

ANEXO H - ARRANQUE SECUENCIAL (LIFO)

El siguiente código consiste en encender cuatro máquinas en orden secuencial y apagarlas en orden inverso con la condición que la última máquina sea la primera en detenerse.

```
int b=0,st1=2,sp1=4,r1=8,r2=9,r3=10,r4=11;
long time,control;
void setup()
{
  pinMode(st1,INPUT);
  pinMode(sp1,INPUT);
  pinMode(r1,OUTPUT);
  pinMode(r2,OUTPUT);
  pinMode(r3,OUTPUT);
  pinMode(r4,OUTPUT);
  digitalWrite(r1,HIGH);
  digitalWrite(r2,HIGH);
  digitalWrite(r3,HIGH);
  digitalWrite(r4,HIGH);
}
void loop()
{
  if(digitalRead(st1)&&digitalRead(sp1))
  {
    b=1;
    digitalWrite(r1,LOW);
    control=millis();
  }
  if(digitalRead(st1)&&!digitalRead(sp1))
  {
    b=0;
    digitalWrite(r1,HIGH);
    delay(1000);
  }
  if(b==1)
  {
    time=millis();
    if(time>=control+4000)
    {
      digitalWrite(r2,LOW);
    }
  }
  if(time>=control+8000)
  {
```

```

digitalWrite(r13,LOW);
}
if(time>=control+12000)
{
digitalWrite(r14,LOW);
}
}

if(!digitalRead(sp1))
{
b=0;
time=0;
digitalWrite(r14,HIGH);
control=millis();
}
if(b==0)
{
time=millis();
if(time>=control+4000)
{
digitalWrite(r13,HIGH);
}
if(time>=control+8000)
{
digitalWrite(r12,HIGH);
}
if(time>=control+12000)
{
digitalWrite(r11,HIGH);
}
}
}
}

```

ANEXO I - ESTRELLA DELTA

El siguiente código consiste en encender un motor en estrella y luego conectarlo en delta.

```
int b=0,st1=2,sp1=4,rl1=8,rl2=9,rl3=10;
long time,control;
void setup()
{
  pinMode(st1,INPUT);
  pinMode(sp1,INPUT);
  pinMode(rl1,OUTPUT);
  pinMode(rl2,OUTPUT);
  pinMode(rl3,OUTPUT);
  digitalWrite(rl1,HIGH);
  digitalWrite(rl2,HIGH);
  digitalWrite(rl3,HIGH);
}
void loop()
{
  if(digitalRead(st1)&&digitalRead(sp1))
  {
    b=1;
    digitalWrite(rl1,LOW);
    digitalWrite(rl2,LOW);
    control=millis();
  }
  if(digitalRead(st1)&&!digitalRead(sp1))
  {
    b=0;
    digitalWrite(rl1,HIGH);
    digitalWrite(rl2,HIGH);
    digitalWrite(rl3,HIGH);
    delay(1000);
  }
  if(b==1)
  {
    time=millis();
    if(time>=control+20000)
    {
      digitalWrite(rl2,HIGH);
    }
  }
  if(time>=control+20200)
  {
```

```
    digitalWrite(r13,LOW);  
  }  
  
  }  
  if(!digitalRead(sp1))  
  {  
    b=0;  
    digitalWrite(r11,HIGH);  
    digitalWrite(r12,HIGH);  
    digitalWrite(r13,HIGH);  
  }  
}
```


ANEXO J - CONTROL DE TEMPERATURA

El siguiente código consiste en controlar un sistema de temperatura con un start y un stop con la condición de que cuando el medio a sensar supere los 28°C. Se ordene el encendido de del sistema de ventilación.

```
float temperatura=0,time,control,suma=0;
int fan=8,st=2,sp=4,b=0,contador=0;
void setup(){

  Serial.begin (9600);
  pinMode(st,INPUT);
  pinMode(sp,INPUT);
  pinMode(fan,OUTPUT);
  digitalWrite(fan,HIGH);
}

void loop()
{
  if(digitalRead(st)&&digitalRead(sp))
  {
    b=1;
    control=millis();
  }
  if(digitalRead(st)&&!digitalRead(sp))
  {
    b=0;
    digitalWrite(fan,HIGH);
    delay(1000);
  }
  if(b==1)
  {
    time=millis();
    if(time>=control+1000)
    {
      control=millis();
      temperatura = (5.0 * analogRead(0)*100.0)/1023.0;
      Serial.println (temperatura);
      Serial.println (b);
      if(temperatura>=28)
      {
        digitalWrite(fan,LOW);
      }
    }
  }
  else
```

```
{
  digitalWrite(fan,HIGH);
}
}
}
if(!digitalRead(sp))
{
  b=0;
  digitalWrite(fan,HIGH);
}
}
```

ANEXO K- PRUEBA DEL MÓDULO

Código de prueba de los módulos arduino que cuando se acciona uno de los dos Start todos los relevos se activan y cuando se accionan uno de los dos stop, se desactivan.

```
int st1=2,st2=3,sp1=4,sp2=5,A=8,B=9,C=10,D=11;
void setup()
{
  pinMode(st1,INPUT);
  pinMode(st2,INPUT);
  pinMode(sp1,INPUT);
  pinMode(sp2,INPUT);
  pinMode(A,OUTPUT);
  pinMode(B,OUTPUT);
  pinMode(C,OUTPUT);
  pinMode(D,OUTPUT);
  digitalWrite(A,HIGH);
  digitalWrite(B,HIGH);
  digitalWrite(C,HIGH);
  digitalWrite(D,HIGH);
}
void loop()
{
  if(digitalRead(st1)==1 || digitalRead(st2)==1)
    digitalWrite(A, LOW),digitalWrite(B, LOW),digitalWrite(C, LOW), digitalWrite(D,
LOW);

  if(digitalRead(sp1)==0 || !digitalRead(sp2)==1)

digitalWrite(A,HIGH),digitalWrite(B,HIGH),digitalWrite(C,HIGH),digitalWrite(D,HIG
H);
}
```