



Universidad de San Carlos de Guatemala  
Facultad de Ingeniería  
Escuela de Ingeniería Mecánica Eléctrica

**PROPUESTA DE UN DISPOSITIVO DE CONTROL PARENTAL PARA EL  
TIEMPO DE USO DE CONSOLAS DE VIDEOJUEGOS EN UN HOGAR**

**Didier Carlos Roberto Tenas Morales**

Asesorado por el Inga. Ingrid Salome Rodríguez de Loukota

Guatemala, septiembre de 2017

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA



FACULTAD DE INGENIERÍA

**PROPUESTA DE UN DISPOSITIVO DE CONTROL PARENTAL PARA EL  
TIEMPO DE USO DE CONSOLAS DE VIDEOJUEGO EN UN HOGAR**

TRABAJO DE GRADUACIÓN

PRESENTADO A LA JUNTA DIRECTIVA DE LA  
FACULTAD DE INGENIERÍA

POR

**DIDIER CARLOS ROBERTO TENAS MORALES**

ASESORADO POR EL INGA. INGRID SALOME RODRIGUEZ DE LOUKOTA

AL CONFERÍRSELE EL TÍTULO DE

**INGENIERO EN ELECTRÓNICA**

GUATEMALA, SEPTIEMBRE DE 2017

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA  
FACULTAD DE INGENIERÍA



**NÓMINA DE JUNTA DIRECTIVA**

DECANO	Ing. Pedro Antonio Aguilar Polanco
VOCAL I	Ing. Angel Roberto Sic García
VOCAL II	Ing. Pablo Christian de León Rodríguez
VOCAL III	Ing. José Milton de León Bran
VOCAL IV	Br. Jurgen Andoni Ramírez Ramírez
VOCAL V	Br. Oscar Humberto Galicia Nuñez
SECRETARIA	Inga. Lesbia Magalí Herrera López

**TRIBUNAL QUE PRACTICÓ EL EXAMEN GENERAL PRIVADO**

DECANO	Ing. Murphy Olympos Paiz Recinos
EXAMINADOR	Ing. Otto Fernando Andrino González
EXAMINADOR	Ing. José Antonio de León Escobar
EXAMINADORA	Inga. Wendy Nora Miranda López
SECRETARIO	Ing. Hugo Humberto Rivera Pérez

## **HONORABLE TRIBUNAL EXAMINADOR**

En cumplimiento con los preceptos que establece la ley de la Universidad de San Carlos de Guatemala, presento a su consideración mi trabajo de graduación titulado:

### **PROPUESTA DE UN DISPOSITIVO DE CONTROL PARENTAL PARA EL TIEMPO DE USO DE CONSOLAS DE VIDEOJUEGOS EN UN HOGAR**

Tema que me fuera asignado por la Dirección de la Escuela de Ingeniería Mecánica Eléctrica, con fecha 26 de marzo de 2013.

**Didier Carlos Roberto Tenas Morales**

Guatemala 4 de febrero de 2016

Ingeniero  
Carlos Eduardo Guzmán Salazar  
Coordinador del Área de Electrónica  
Escuela de Ingeniería Mecánica Eléctrica  
Facultad de Ingeniería, USAC.


Estimado Ingeniero Guzmán.

Me permito dar aprobación al trabajo de graduación titular: **"Propuesta de un dispositivo de control parental para el tiempo de uso de consolas de videojuegos en un hogar"**, del señor Didier Carlos Roberto Tenas Morales, por considerar que cumple con los requisitos establecidos.

Por tanto, el autor de este trabajo de graduación y, yo, como su asesora, nos hacemos responsables por el contenido y conclusiones del mismo.

Sin otro particular, me es grato saludarle.

Atentamente,

  
Inga. Ingrid Rodríguez de Loukota  
Colegiada 5,356  
Asesora

Ingrid Rodríguez de Loukota  
Ingeniera en Electrónica  
colegiado 5356



FACULTAD DE INGENIERIA

REF. EIME 11 2017.  
Guatemala, 29 de noviembre 2016.

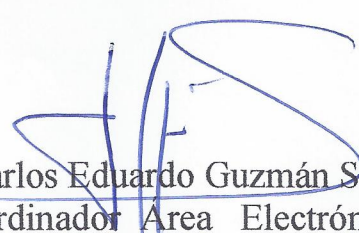
Señor Director  
Ing. Francisco Javier González López  
Director Escuela de Ingeniería Mecánica Eléctrica  
Facultad de Ingeniería, USAC.

Señor Director:

Me permito dar aprobación al trabajo de Graduación titulado:  
**PROPUESTA DE UN DISPOSITIVO DE CONTROL  
PARENTAL PARA EL TIEMPO DE USO DE CONSOLAS DE  
VIDEOJUEGOS EN UN HOGAR,** del estudiante Didier  
Carlos Roberto Tenas Morales, que cumple con los requisitos  
establecidos para tal fin.

Sin otro particular, aprovecho la oportunidad para saludarle.

Atentamente,  
ID Y ENSEÑAD A TODOS

  
Ing. Carlos Eduardo Guzmán Salazar  
Coordinador Área Electrónica



STO



FACULTAD DE INGENIERIA

REF. EIME 11. 2017.

El Director de la Escuela de Ingeniería Mecánica Eléctrica, después de conocer el dictamen del Asesor, con el Visto bueno del Coordinador de Área, al trabajo de Graduación del estudiante: **DIDIER CARLOS ROBERTO TENAS MORALES** Titulado: **PROPUESTA DE UN DISPOSITIVO DE CONTROL PARENTAL PARA EL TIEMPO DE USO DE CONSOLAS DE VIDEOJUEGOS EN UN HOGAR,** procede a la autorización del mismo.

  
Ing. Francisco Javier González López



GUATEMALA, 27 DE MARZO 2017.





Ref. DTG.D.419.2017

El Decano de la Facultad de Ingeniería de la Universidad de San Carlos de Guatemala, luego de conocer la aprobación por parte del Director de la Escuela de Ingeniería Mecánica Eléctrica al trabajo de graduación titulado: **PROPUESTA DE UN DISPOSITIVO DE CONTROL PARENTAL PARA EL TIEMPO DE USO DE CONSOLAS DE VIDEOJUEGOS EN UN HOGAR**, presentado por el estudiante universitario: **Didier Carlos Roberto Tenas Morales** y después de haber culminado las revisiones previas bajo la responsabilidad de las instancias correspondientes, se autoriza la impresión del mismo.

IMPRÍMASE.

Ing. Pedro Antonio Aguilar Polanco  
Decano



Guatemala, septiembre, de 2017



## **ACTO QUE DEDICO A:**

### **Dios**

Por ser quien coloco muchos obstáculos en mi camino para forjarme como profesional y como ser humano.

### **Mis padres**

Leticia Morales y Carlos Tenas, quienes me han apoyado incondicionalmente en todo momento.

### **Mis hermanos**

Sugey Tenas y Diego Tenas, quienes me han incentivado en todo momento a lograr el objetivo.

### **Personas especiales y amigos**

Por ser las personas que la vida te pone en el camino para ayudarte y motivarte a superar todo reto y aprender de ellos.

## **AGRADECIMIENTOS A:**

**Universidad de San  
Carlos de Guatemala**

Por ser la universidad que me forjó como profesional.

**Facultad de Ingeniería**

Por ser la facultad en donde encontré mi profesión para la vida.

**Mis amigos de  
la facultad**

Quienes se convirtieron en personas muy apreciadas e importantes en mi vida.

**Inga. Ingrid Rodriguez  
de Loucota**

Por su valiosa ayuda en la asesoría de mi trabajo de graduación.

**Compañeros y amigos**

Por brindarme su amistad y ayuda necesaria para superar cada reto de la carrera.

## ÍNDICE GENERAL

ÍNDICE DE ILUSTRACIONES.....	V
LISTA DE SÍMBOLOS .....	IX
GLOSARIO .....	XI
RESUMEN.....	XVII
OBJETIVOS.....	XIX
INTRODUCCIÓN.....	XXI
1. VIDEOJUEGOS .....	1
1.1. Descripción de los videojuegos .....	1
1.2. El mercado de los videojuegos.....	3
1.2.1. Principales productos de consolas .....	3
1.2.2. Los jugadores .....	4
1.2.3. Edades de uso de videojuegos.....	4
1.2.4. Marcas de videojuegos más utilizadas .....	6
1.3. Efectos en la salud de las personas .....	9
1.3.1. Los videojuegos y la responsabilidad .....	10
1.3.2. ¿Qué problemas puede ocasionarle a los niños el uso excesivo de los videojuegos? .....	11
1.3.3. Horas recomendadas de uso de videojuegos.....	12
2. LA TECNOLOGÍA ARDUINO.....	13
2.1. ¿Qué puede hacer Arduino? .....	13
2.1.1. <i>Wiring</i> .....	14
2.1.2. <i>Processing</i> .....	14
2.2. Los microcontroladores de Arduino .....	15

2.2.1.	El microcontrolador .....	15
2.2.1.1.	Características .....	16
2.2.2.	Modelos de Arduino .....	17
2.2.3.	Microcontroladores AVR de Atmel .....	18
2.3.	Arduino Mega 2560 .....	19
2.3.1.	Alimentación.....	21
2.3.2.	Entradas y salidas .....	22
2.3.3.	Programación .....	23
3.	TECNOLOGÍA COMPLEMENTARIA.....	25
3.1.	Entorno gráfico.....	25
3.1.1.	LCD 16X2.....	25
3.1.1.1.	Características principales .....	25
3.1.1.2.	Descripción de pines .....	26
3.1.2.	GLCD .....	27
3.1.3.	Pantalla LCD TFT de Arduino .....	29
3.1.3.1.	Bibliotecas utilizadas para la pantalla táctil .....	31
3.2.	Manipulación .....	31
3.2.1.	Pulsadores .....	31
3.2.1.1.	Funcionamiento.....	32
3.2.2.	Teclado matriciales 4x4.....	33
3.2.3.	Pantalla LCD táctil TFT .....	35
3.2.3.1.	Tipos .....	35
3.2.4.	Manipulación con dispositivos móviles con sistema operativo Android .....	37
3.2.4.1.	Comunicación <i>Bluetooth</i> .....	38
3.3.	Control de tiempo y fechas.....	39
3.3.1.	Temporizador .....	39

3.3.2.	Arduino DS1307 con AT24C32.....	39
3.4.	Control de encendido/apagado de TV y consola de videojuego .....	40
3.4.1.	Relé .....	41
3.4.2.	Sensores Infrarrojos .....	42
3.4.2.1.	Fotodiodo receptor TSOP48 .....	43
3.5.	Seguridad .....	43
3.5.1.	Contraseña .....	44
3.5.2.	Balance entre seguridad y comodidad.....	45
3.5.3.	Factores en la seguridad de un sistema de contraseñas .....	45
3.5.3.1.	Posibilidades de que algún atacante pueda adivinar o inventar la contraseña .....	46
3.5.3.2.	Formas de almacenar contraseñas .....	46
3.5.4.	Almacenamiento de datos en memorias EEPROM .....	47
3.5.4.1.	Librería EEPROM .....	48
3.5.5.	Procedimientos para cambiar las contraseñas .....	48
3.5.6.	Diseño de <i>software</i> protegido .....	49
4.	DISEÑO DEL DISPOSITIVO.....	51
4.1.	Descripción del diseño del dispositivo .....	51
4.2.	Partes del dispositivo.....	52
4.2.1.	Controlador principal.....	52
4.2.2.	Entorno gráfico y manipulación del dispositivo .....	53
4.2.3.	Control de encendido/apagado de TV y consola de videojuego .....	56
4.2.4.	Seguridad .....	63

4.2.5.	Control de tiempo y fecha.....	70
4.2.5.1.	Control parental.....	75
CONCLUSIONES.....		81
RECOMENDACIONES .....		83
BIBLIOGRAFÍA.....		85



## ÍNDICE DE ILUSTRACIONES

### FIGURAS

1.	Gráfica de edades de los niños cuando utilizan los videojuegos .....	5
2.	Tipos de equipo de videojuegos.....	6
3.	Marcas de consolas de videojuegos más vendidas en Japón.....	7
4.	Consolas de videojuegos más vendidas en Japón .....	7
5.	Ventas de consolas de videojuegos a nivel mundial en 2010 .....	8
6.	Ventas de consolas de videojuegos en América en 2010 .....	8
7.	Niños utilizando un videojuego.....	10
8.	Microcontrolador .....	16
9.	Placa de Arduino Mega 2560 .....	20
10.	Símbolo de un pulsador .....	32
11.	Diagrama de un teclado matricial de 4x4 .....	33
12.	Decodificador de teclado de 4x4 .....	34
13.	Pantalla capacitiva y principio de funcionamiento .....	36
14.	Pantalla LCD TFT de Arduino .....	37
15.	Diagrama del Arduino DS1307.....	40
16.	Diferentes formas físicas de un relé .....	41
17.	Diagrama de un relé.....	42
18.	Partes del dispositivo .....	51
19.	Diagrama de conexión de Arduino mega con la TFT LCD 3.2” .....	53
20.	Ejemplo de botón dibujado en la pantalla LCD TFT .....	55
21.	Coordenadas de botones en pantalla TFT LCD 3.2” .....	55
22.	Diagrama de escalabilidad del menú del dispositivo .....	56

23.	Diagrama de circuito de control de alimentación de TV y consola de videojuego .....	57
24.	Mensaje presionar botón de apagado de TV .....	58
25.	Señal IR de control remoto de TV.....	58
26.	Código ingresado.....	59
27.	Verificación del código de TV .....	59
28.	Verificación de TV apagado con código ingresado .....	60
29.	Diagrama conexión de infrarrojo emisor y receptor. ....	61
30.	Ingresar la clave a ser guardada .....	63
31.	Seguridad al ingresar los dígitos.....	63
32.	Ejemplo de 5 dígitos de contraseña ingresada .....	64
33.	Confirmar contraseña .....	64
34.	Contraseñas no son iguales .....	65
35.	Solicitud de contraseña del dispositivo .....	66
36.	Clave incorrecta .....	66
37.	Máximo número de intentos para ingresar al dispositivo .....	67
38.	Sistema Bloqueado.....	67
39.	Cambiar clave o fecha y hora .....	68
40.	Modificar clave .....	68
41.	Clave modificada con éxito .....	69
42.	Diagrama de conexión del módulo RTC con el Arduino Mega .....	70
43.	Ingresar hora del dispositivo .....	72
44.	Ejemplo de hora ingresada del dispositivo.....	73
45.	Ingresar fecha del dispositivo .....	73
46.	Ejemplo de fecha ingresada al sistema .....	74
47.	Seleccionar día actual de la semana .....	74
48.	Elegir plan de control .....	75
49.	Menú plan de control semanal o diario .....	76
50.	Hora inicial de juego .....	76

51.	Hora final de juego .....	77
52.	Horario semanal .....	78
53.	Seleccionar día de plan de control diario .....	78

## TABLAS

I.	Marcas de consolas .....	3
II.	Características de los modelos de Arduino .....	17
III.	Características el Arduino Mega 2560 .....	20
IV.	Sintaxis básicas del lenguaje Arduino .....	23
V.	Descripción de pines .....	26
VI.	Descripción de pines .....	30
VII.	Declaración de variables y comandos para la librería IRremote.h .....	61
VIII.	Sentencias para obtener hora y fecha del reloj RTC.....	72



## LISTA DE SÍMBOLOS

Símbolo	Significado
<b>KB</b>	Kilobyte
<b>Kb</b>	Kilobit
<b>KHz</b>	Kilohercio
<b>MHz</b>	Megahercio
<b>M</b>	Metro
<b>mA</b>	Miliamperios
<b>mm</b>	Milímetro
<b>mW</b>	Microwatt
<b>μC</b>	Microcontrolador
<b>nm</b>	Nanómetro
<b>V</b>	Voltios





## GLOSARIO

<b>AC</b>	Corriente alterna ( <i>Alternating Current</i> , por sus siglas en inglés), es la corriente eléctrica en la que la magnitud y el sentido varían cíclicamente.
<b>APK</b>	<i>Application package file</i> ; un archivo con extensión apk, es un paquete para el sistema operativo Android, es básicamente un archivo comprimido ZIP con diferente extensión.
<b>AREF</b>	Nombre que recibe un pin de Arduino, la tensión aplicada en este pin será la que haga que el conversor analógico a digital de su máxima lectura (1 023) cuando lea una tensión igual a la aplicada en ese pin.
<b>AVR</b>	Familia de microcontroladores RISC del fabricante estadounidense Atmel.
<b>BIT</b>	Acrónimo de <i>binary digit</i> (dígito binario). Un bit es un dígito del sistema de numeración binario. Las unidades de almacenamiento tienen por símbolo bit, en el se utilizan solo dos dígitos:
<b>CIU</b>	<i>The Competitive Intelligence Unit</i> , firma de consultoría e investigación de mercado de alcance

regional y con amplia experiencia en las comunicaciones, infraestructura y tecnología.

<b>CPU</b>	Unidad central de procesamiento (del inglés: <i>central processing unit</i> , CPU) o procesador. Componente principal del ordenador y otros dispositivos programables, que interpreta las instrucciones contenidas en los programas y procesa los datos
<b>DAC</b>	Conversor de señal digital a analógico (del inglés: <i>Digital to Analogue Converter</i> ). Dispositivo para convertir señales digitales con datos binarios en señales de corriente o de tensión analógica.
<b>DC</b>	Corriente continua (en inglés DC, de <i>direct current</i> ) se refiere al flujo continuo de carga eléctrica a través de un conductor entre dos puntos de distinto potencial que no cambia de sentido con el tiempo.
<b>DSP</b>	Procesador digital de señales o DSP (siglas en inglés de <i>digital signal processor</i> ), sistema basado en un procesador o microporocesorador que posee un conjunto de instrucciones, un <i>hardware</i> y un <i>software</i> optimizados para aplicaciones operaciones numéricas a muy alta velocidad.
<b>EEPROM</b>	Siglas de <i>electrically erasable programmable read-only memory</i> (ROM programable y borrada eléctricamente).Tipo de memoria ROM que puede

ser programada, borrada y reprogramada eléctricamente, a diferencia de la EPROM que ha de borrarse mediante un aparato que emite rayos ultravioleta.

**FTDI**

Futuros dispositivos internacionales de tecnología (en inglés: *future technology devices international*). Fabricante de procesadores que se hizo famoso por sus chips conversores USB-Serie. Las placas Arduino hacen uso de este dispositivo para conectar a diferentes tipos de ordenadores, es por esto que se necesitan drivers al instalar Arduino.

**GND**

Tierra (en inglés: *ground*); puede referirse al punto de referencia en un circuito eléctrico a partir del cual se miden las tensiones, un camino de retorno común para la corriente eléctrica o de una conexión física directa a la tierra.

**LED**

Diodo emisor de luz (del acrónimo en inglés LED, *light emitting diode*). Componente optoelectrónico pasivo y, más contritivamente, un diodo que emite luz.

**PIC**

Familia de microcontroladores tipo RISC fabricados por *Microchip Technology Inc.* y derivados del PIC1650, originalmente, desarrollado por la división de microelectrónica de *General Instrument*.

<b>PWM</b>	Modulación por ancho de pulsos (también conocida como PWM, siglas en inglés de <i>pulse width modulación</i> ) de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica (una senoidal o cuadrada, por ejemplo).
<b>RAM</b>	La memoria de acceso aleatorio (en inglés: <i>random access memory</i> ) se utiliza como memoria de trabajo para el sistema operativo, los programas y la mayoría del software. Es allí donde se cargan todas las instrucciones que ejecuta el procesador y otras unidades de cómputo.
<b>ROM</b>	La memoria de solo lectura, conocida también como ROM (acrónimo en inglés de <i>read only memory</i> ), es un medio de almacenamiento utilizado en ordenadores y dispositivos electrónicos, que permite solo la lectura de la información y no su escritura, independientemente de la presencia o no de una fuente de energía.
<b>RTC</b>	Reloj en tiempo real (en inglés, <i>real time clock</i> , RTC) es un reloj de un ordenador, incluido en un circuito integrado, que mantiene la hora actual.
<b>SDK</b>	Kit de desarrollo de <i>software</i> o SDK (siglas en inglés de <i>software development kit</i> ); es generalmente un conjunto de herramientas de desarrollo de <i>software</i>

que le permite al programador crear aplicaciones para un sistema concreto.

## **SIMD**

En computación, SIMD (del inglés *single instrucción, múltiple data*, en español: una instrucción, múltiples datos); técnica empleada para conseguir paralelismo a nivel de datos.

## **SPI**

Del inglés *serial peripheral interface*, es un estándar de comunicaciones, usado principalmente para la transferencia de información entre circuitos integrados en equipos electrónicos.

## **UART**

Son las siglas de *universal asynchronous receiver transmitter* (en español, Transmisor Receptor asíncrono universal); controla los puertos y dispositivos serie.

## **USB**

Bus universal en serie (del inglés: universal serial bus); estándar industrial que define los cables, conectores y protocolos usados en un bus para conectar, comunicar y proveer de alimentación eléctrica entre ordenadores y periféricos y dispositivos electrónicos.

## **TFT-LCD**

*Thin film transistor-liquid crystal display*, pantalla de cristal líquido de transistores de película fina, usa tecnología de transistor de película delgada para mejorar su calidad de imagen.





## **RESUMEN**

El primer capítulo del trabajo de investigación describe todo lo relacionado en los videojuegos: el significado del vocablo y los dispositivos a los que se les llama videojuegos.

En el mercado de los videojuegos existe, tres marcas famosas de fabricantes de consolas, las más cotizadas indica en el contenido hay. Una gráfica las edades de los niños que dedican más horas a los videojuegos.

De igual manera, se advierte que el uso excesivo de las consolas de videojuegos por parte de los niños puede tener efectos preocupantes en su salud física, mental y social; en consecuencias se recomienda el número de horas que da forma saludable puedan los niños utilizar la consola de videojuegos.

En el segundo capítulo, se describe todo lo referente a la tecnología Arduino; una tecnología de punta para este proyecto, la base sobre la cual está construido el dispositivo de control parental. El sistema tecnológico Arduino cuenta con diversos modelos de placas, cada una con un modelo distinto de microprocesador, se describen en este capítulo. También, se describe el modelo de placa Arduino Mega 2560, utilizado en el dispositivo.

Se describe, también, el nivel de voltaje, sus características de memoria, sus puertos de entrada y salida la sintaxis de programación.

En el capítulo tercero, se menciona en detalle, todas las tecnologías complementarias con las que interactuará el dispositivo Arduino Mega 2560. Estas tecnologías, por razones académicas, fueron divididas en cuatro áreas: la primera, que se refiere al entorno gráfico, y ésta se refiere a la forma como se visualizarán los datos del dispositivo; segunda, se refiere a la manipulación: cómo se controlará el dispositivo en la práctica, con un teclado matricial o pantalla táctil; la tercera área, constituye el control de tiempo y fechas, se indica la tecnología para que el dispositivo cuente con un reloj, además, las opciones para activar o desactivar la televisión y la consola de videojuego, con la tecnología seguridad y disponibilidad para aplicar al dispositivo.

Por último, el cuarto capítulo presenta el diseño del dispositivo: sus partes y sus funciones. Se indicará, asimismo, la forma de manipular el dispositivo, su relación con el entorno gráfico la sintaxis de programación para su manipulación. Se presentan los diagramas de conexión del Arduino Mega 2560 con las diferentes tecnologías complementarias utilizadas; se muestra la forma como el dispositivo controla la TV y la consola de videojuegos, también la seguridad del dispositivo para evitar que el niño lo manipule con plena libertad, lo que constituye la naturaleza misma del proyecto.

## **OBJETIVOS**

### **General**

Proponer el diseño de un dispositivo que permita a los padres controlar la cantidad de horas que los niños utiliza las consolas de videojuegos.

### **Específicos**

1. Crear, diseñar o investigar métodos de almacenamiento de información para que los padres controlen las horas que los niños jueguen durante el día, semana o mes.
2. Proponer una forma de apagar la TV sin causar algún daño.
3. Proponer mecanismos para que el usuario conozca el tiempo de juego transcurrido.



## INTRODUCCIÓN

En la actualidad los videojuegos tienen un papel muy importante en la vida de los niños incluso influye en su educación; también, generan un descontrol en sus formas de vida, en especial, cuando existe abuso del tiempo dedicado los videojuegos, inclusive causas ansiedad, depresión y otros males en su sistema nervioso, como efectos negativos de tipo psicológico.

También, se ha establecido, que el uso excesivo de los videojuegos en niños y jóvenes produce otros efectos nocivos de tipo social y físico en su organismo. En cuanto a los de tipo social, provocan falta de interés en sus relaciones interpersonales aislamiento de la sociedad. En cuanto a los de tipo físico, la falta de interés en la práctica de algún deporte, proclives, en consecuencia a la obesidad y a enfermedades de tipo cardiovasculares.

Sin embargo, desde un punto de vista positivo, los videojuegos por sí mismos no son dañinos para los niños, siempre y cuando sean utilizados de forma moderada y bajo el control y responsabilidad de los padres de familia, quienes tienen el deber de controlar las horas que sus hijos dedican a los videojuegos, asimismo, los juegos que los menores utilizan durante su tiempo libre.

Por la razón anterior surge la propuesta de crear un dispositivo electrónico con el cual los padres de familia dispongan de una herramienta útil de carácter preventivo el beneficio para sus hijos y familia; este dispositivo facilita llevar el control de las horas de juego que los niños dedican diariamente para regular, desde cualquier lugar los horarios cuando los hijos utilizarán la consola de

videojuegos; dispone de todas las herramientas de seguridad electrónica para que solo el padre pueda manipularlo.

En ese orden de ideas, se plantea la propuesta de un dispositivo con espacio suficiente para almacenar horarios, contraseñas y variables para la programación de este dispositivo.

En caso necesario se propone un proyecto innovador de suma importancia y utilidad, en especial, para padres de familia que, derivado de la vida moderna y las demandas laborales que impone la globalización, están obligados a salir por largos períodos de tiempo de sus hogares alejados de la importante interrelación con sus niños.



# **1. VIDEOJUEGOS**

Un videojuego es un conjunto de obras artísticas, literarias y científicas creadas por un autor o varios bajo la dirección de una persona natural o jurídica; estas obras son programadas posteriormente y compiladas gracias al *software* sin ser considerado el videojuego como un programa de ordenador ni como una obra audiovisual, con un tratamiento diferenciado.

El videojuego está creado para el entretenimiento en general o para el desarrollo de disciplinas profesionales, educativas o de salud, basado en la interacción entre una o varias personas por medio de un controlador y un aparato electrónico que ejecuta el videojuego; este dispositivo electrónico puede ser una computadora, una videoconsola o un dispositivo portátil (teléfono móvil o tableta, por ejemplo), conocidos como plataformas. Aunque, usualmente el término video en la palabra videojuego se refiere en sí a un visualizador de gráficos rasterizados; actualmente se utiliza para usar cualquier tipo de visualizador.

## **1.1. Descripción de los videojuegos**

El término de videojuegos se aplica a todo tipo de juego digital interactivo, con independencia de su soporte.

Hay videojuegos sencillos y otros más complejos, algunos son capaces de narrar historias y acontecimientos usando audio y video creados *ex profeso*; esto demuestra que el videojuego es otra manifestación del arte.

Al dispositivo de entrada usado para manipular un videojuego se lo conoce como controlador de videojuego. Por ejemplo, un controlador de consola podría únicamente consistir de un botón y una palanca de mando. No obstante, otro podría presentar una docena de botones y una o más palancas. Frecuentemente, los primeros juegos informáticos hacían uso de un teclado para llevar a cabo una interacción e incluso se necesitaba que el usuario adquiriera una palanca por separado que tenía un botón como mínimo.

Varios juegos de computadora modernos permiten, y en algunos casos exigen, que el usuario use un teclado y un ratón de forma simultánea. Por lo general, los videojuegos hacen uso de otras maneras de proveer la interactividad e información al jugador. El audio es casi universal; se usan dispositivos de reproducción de sonido: altavoces y auriculares.

Desde su aparición, a principios de la década de los setenta en California, los videojuegos han alcanzado una gran importancia cultural y económica, han sido un importante factor de innovación tecnológica, los precursores del encuentro entre la informática y la televisión; son, también, un antecedente directo de los actuales sistemas multimedia interactivos. Además, han ocupado una parte importante del tiempo dedicado al ocio en los países industrializados.

El volumen de negocio de la industria del videojuego a nivel mundial es, en la actualidad, uno de los más importantes del sector informático y de comunicaciones; compañías como Microsoft o Sony compiten ferozmente para lanzar al mercado máquinas cada vez más sofisticadas; los padres se limitan a menudo, a proporcionar el dinero a sus hijos para comprar la consola o el videojuego.

Lo cierto, es que millones de niños en el mundo consumen muchas horas a la semana absorbidos por combates, disparos, agresiones, persecuciones, asaltos y guerras galácticas.

## **1.2. El mercado de los videojuegos**

De acuerdo con un estudio elaborado por la Unidad de Inteligencia Competitiva (CIU, por sus siglas en inglés), empresa de consultoría e investigación de mercado, en 2008, la industria de los videojuegos generó ventas por \$.47 000 000 000 en todo el mundo. En este escenario global, México es el mercado más importante en América Latina.

México ocupa el cuarto lugar mundial en términos de consumo y el primero en América Latina. Para tener una idea, el consumo en México ronda entre los \$.850 000 000 y los \$.900 000 000 anuales.

### **1.2.1. Principales productos de consolas**

En la actualidad, existen tres principales productores de consolas para videojuegos:

Tabla I. **Marcas de consolas**

<b>Empresa</b>	<b>Consolas</b>
Nintendo	Wii
Microsoft	Xbox
Sony	Play Station

Fuente: Brújula de compra. *Un gasto nada menos: videojuegos.*

[http://www.profeco.gob.mx/encuesta/brujula/bruj\\_2010/bol170\\_videojuegos.asp](http://www.profeco.gob.mx/encuesta/brujula/bruj_2010/bol170_videojuegos.asp) Consulta: 19 de mayo de 2014.

Según la consultora CIU, los consumidores tienen mayor preferencia por el *Xbox 360* de *Microsoft* (61 %), en segundo lugar, el *Wii* de *Nintendo* (22 %) y en tercero *Play Station 3* de *Sony* (17 %).

La marca *Nintendo* tiene su mayor base de usuarios en los niños de edades entre los 6 y 11 años, *Xbox 360* y *Play Station 3* en el rango entre 12 y 17 años.

Las preferencias obedecen al precio de la consola, a la disponibilidad de juegos para cada plataforma y al diseño de en cada plataforma.

### **1.2.2. Los jugadores**

Hay dos tipos de videojugadores:

- Casual: usuario relativamente nuevo, que dedica pocas horas a la semana al juego; lo utiliza para pasar el tiempo de forma divertida y busca videojuegos fáciles de utilizar.
- 
- Gamer o intensivo: jugador experimentado que tiene una gran pasión por esta actividad; le dedica muchas horas y tiene gustos diversificados, en cuanto a géneros y plataformas. Es un consumidor asiduo de las novedades que ofrece el mercado.

### **1.2.3. Edades de uso de videojuegos**

La DGEC (Dirección de Gestión de Evaluación de la Calidad) de Profeco (Procuraduría Federal del Consumidor, en México) realizó un sondeo, del 30 de

marzo al 5 de abril de 2010, a 300 personas con hijos entre 6 y 15 años que utilizan videojuegos.

El objetivo fue conocer sus hábitos de compra con relación a las consolas y videojuegos, además de la asistencia a locales donde pueden jugar.

De acuerdo con los resultados, 38 % mencionó que sus hijos tienen de 1 a 3 años de usar videojuegos; mientras que 31 % más de 5 años, principalmente, como se muestra a continuación:

Figura 1. **Gráfica de edades de los niños cuando utilizan los videojuegos**



Fuente: Brújula de compra. *Un gasto nada menos: videojuegos.*

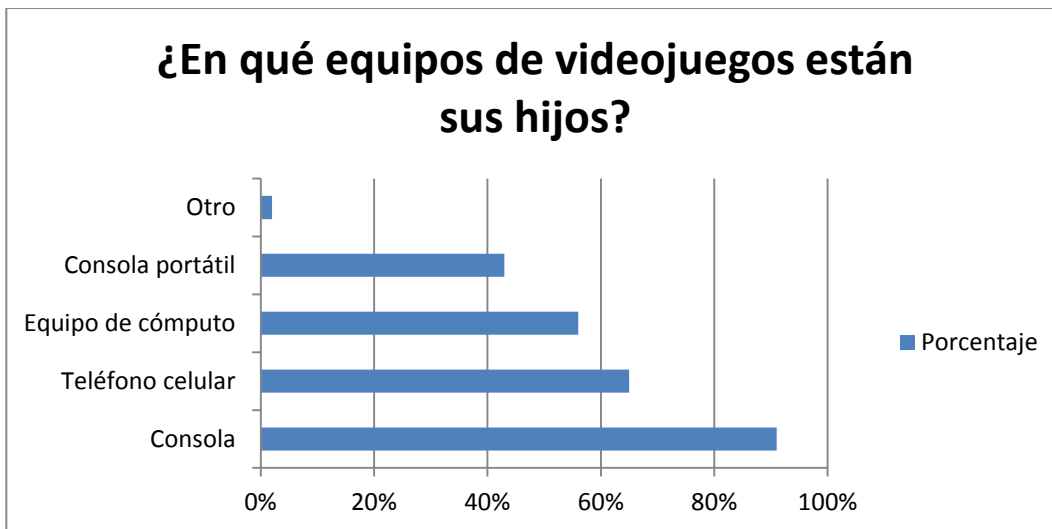
[http://www.profeco.gob.mx/encuesta/brujula/bruj\\_2010/bol170\\_videojuegos.asp](http://www.profeco.gob.mx/encuesta/brujula/bruj_2010/bol170_videojuegos.asp).

Consulta: 20 de mayo de 2014.

#### 1.2.4. Marcas de videojuegos más utilizadas

Como se observa en la figura 1, de los equipos que se utilizan para videojuegos, la mayoría (91 %) comentó que sus hijos usan consola, seguido de celular (65 %) y equipo de cómputo (56 %).

Figura 2. Tipos de equipo de videojuegos

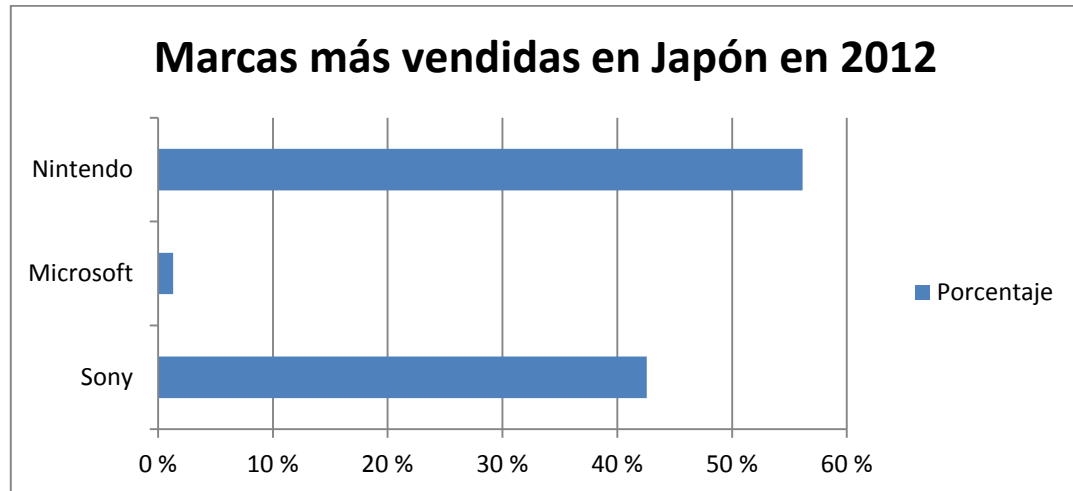


Fuente: Brújula de compra. *Un gasto nada menos: videojuegos.*

[http://www.profeco.gob.mx/encuesta/brujula/bruj\\_2010/bol170\\_videojuegos.asp](http://www.profeco.gob.mx/encuesta/brujula/bruj_2010/bol170_videojuegos.asp). Consulta: 20 de mayo de 2014.

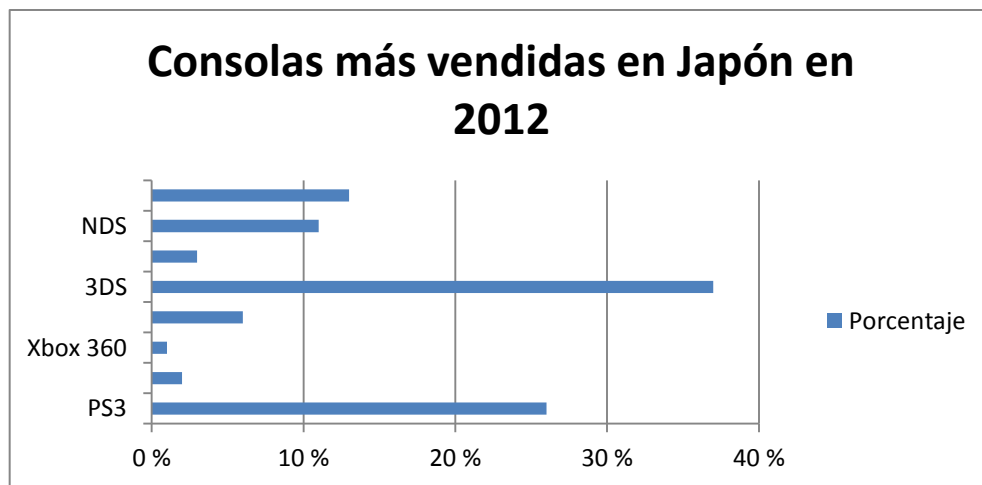
*Media Create*, empresa japonesa que recoge y analiza los datos de la industria del entretenimiento digital, público una lista de las consolas más vendidas en Japón durante el año 2012.

Figura 3. **Marcas de consolas de videojuegos más vendidas en Japón**



Fuente: Glovedia. *Las consolas más vendidas del 2012 en Japón*. <http://gt.globedia.com/consolas-videojuegos-vendidos-2012-japon>. Consulta: 20 de mayo de 2014.

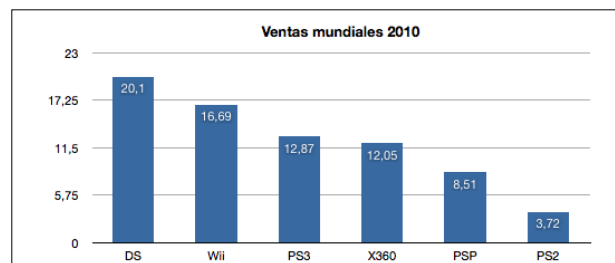
Figura 4. **Consolas de videojuegos más vendidas en Japón**



Fuente: Glovedia. *Las consolas más vendidas del 2012 en Japón*. <http://gt.globedia.com/consolas-videojuegos-vendidos-2012-japon>. Consulta: 20 de mayo de 2014.

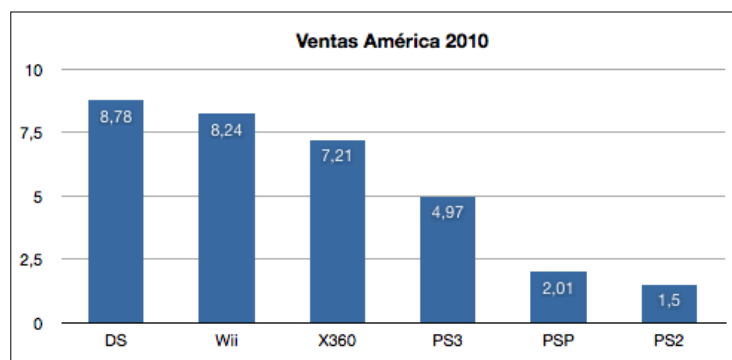
En las ventas las consolas de videojuegos, a nivel mundial en 2010 de la más variedad ha sido DS por quinto año consecutivo. Tampoco, es de extrañar que sea *Wii* la consola que le siga los pasos; lo que sí llama poderosamente la atención, es lo cercanas que han estado en ventas *PlayStation 3* y *Xbox 360*, ganando esta batalla particular Sony por poco más de 800 000 unidades de diferencia.

Figura 5. **Ventas de consolas de videojuegos a nivel mundial en 2010**



Fuente: Glovedia. *Las consolas más vendidas del 2010 en Japón*. <http://gt.globedia.com/consolas-videojuegos-vendidos-2010-japon>. Consulta: 20 de mayo de 2014.

Figura 6. **Ventas de consolas de videojuegos en América en 2010**



Fuente: Glovedia. *Las consolas más vendidas del 2010 en Japón*. <http://gt.globedia.com/consolas-videojuegos-vendidos-2010-japon>. Consulta: 20 de mayo de 2014.



En América, la diferencia entre DS y *Wii* es mucho menor, tanto que la doméstica ha estado cerca de superar a la portátil; aquí *Microsoft* le gana la partida a *Sony*, por bastante diferencia. El continente americano es también el territorio donde menos PSP se han vendido durante 2010.

### **1.3. Efectos en la salud de las personas**

El exceso en el uso de videojuegos puede ser nocivo para la salud, el sedentarismo es la carencia de actividad física fuerte, como utilizar cualquier deporte, que por lo general pone al organismo humano en situación vulnerable ante enfermedades especialmente cardíacas y sociales.

Se presenta con mayor frecuencia en sociedades altamente tecnificadas en donde todo está pensado para evitar grandes esfuerzos físicos.

La forma en que los niños, adolescentes y jóvenes se entretienen y divierten en la actualidad puede generar serios daños en su salud. Especialistas del área de prestaciones médicas, el uso excesivo de videojuegos, celulares y computadoras puede ser acusa de sedentarismo y, en consecuencia, de obesidad.

El niño obeso muy probablemente tiene cifras más altas de presión, de colesterol y por la propia obesidad es propenso a desarrollar diabetes.

El problema no se atribuye al uso, sino a las largas horas y sesiones que se dan con las consolas junto con la ausencia de ejercicio; además, se agrega el consumo de comidas no saludables de alto contenido calórico, frituras, galletas, golosinas y bebidas gaseosas.

### **1.3.1. Los videojuegos y la responsabilidad**

A los videojuegos se les acusan de ser la causa de trastornos psiquiátricos y de comportamiento, de provocar actitudes violentas, adicción, etc. sector de los videojuegos se han defendido argumentando que una cosa es el videojuego y otra la responsabilidad, y no falta razón.

Los videojuegos sirven para pasar ratos estupendos, solo y en compañía. Pero no siempre se reúnen las condiciones adecuadas para ponerse al mando de cualquier videojuego. La Asociación de Pediatría Americana recomienda, y base en las evidencias científicas, que los niños y adolescentes no superen un límite de dos horas de exposición a la pantalla de la televisión o la videoconsola.

La madurez personal, y psicológica y el ambiente son aspectos a tener en cuenta a la hora de evaluar los efectos de los videojuegos en los trastornos. El exceso de videojuegos deriva en problemas más graves como la adicción; incluso determinados juegos representan para algunas personas todo un peligro. Pero, obviamente, la responsabilidad recae en las personas y los tutores de los niños y adolescentes.

**Figura 7. Niños utilizando un videojuego**



Fuente: El Makuaz. *Con todo el poder de Sheringan*. <http://m4ku4z.net/>. Consulta: 8 de enero de 2013.

### **1.3.2. ¿Qué problemas puede ocasionarles a los niños el uso excesivo de los videojuegos?**

- La atención puesta en el juego desarrolla agotamiento y cansancio del sistema nervioso; el juego en exceso puede producirle nerviosismo, ansiedad o depresión.
- Aislamiento de la sociedad: por estar jugando, el individuo se limita a hacer otras actividades: estudiar, pasar más tiempo con la familia, salir con sus amigos o hacer deportes.
- Su comportamiento puede tornarse violento: por influencia de los juegos violentos.
- Pelea demasiado con los otros miembros de la familia, por ejemplo, con sus hermanos; generalmente, estas peleas son causadas por el tiempo de uso de las consolas de videojuegos.
- Bajo rendimiento académico: se desinteresa por las actividades escolares y tiene problemas de atención en la escuela.
- El niño corre riesgo de sedentarismo nocivo: que sumado al consumo de alimentos perjudiciales desencadenan la aparición de sobrepeso y obesidad.

Pueden aparecerle molestias en los ojos con síntomas de irritación, ya que no realiza los movimientos de parpadeo que lubrican la córnea. Esto se debe a la atención que le presta a los juegos.

- Debido a la postura del niño cuando juega, se pueden presentar dolores musculares o vicios posturales perjudiciales para un organismo en crecimiento.

### **1.3.3. Horas recomendadas de uso de videojuegos**

Se recomienda no más de cuatro horas a la semana; por lo tanto, se debe establecer un horario para jugar y limitar el número de horas usar los videojuegos. Se puede aumentar el tiempo de juego durante el fin de semana o en época de vacaciones. Hay más tiempo para el ocio, pero no se debe exceder.

Algunos expertos señalan que lo recomendable es fijar un límite de entre 4 y 5 horas semanales. Valorar qué es lo más adecuado según el grado de madurez del niño.

No dejar que el niño use la consola poco antes de irse a la cama. Los videojuegos son muy estimulantes y pueden hacer que al pequeño tenga problemas de sueño.

## 2. LA TECNOLOGIA ARDUINO

Arduino es una plataforma de creación de prototipos electrónicos de código abierto basado en *hardware* de y *software* de fácil uso, flexible. Está dirigido a artistas, diseñadores, aficionados y cualquier persona interesada en la creación de objetos o entornos interactivos.

### 2.1. ¿Qué puede hacer Arduino?

Arduino puede detectar el medio ambiente mediante la recepción de la entrada de una variedad de sensores y puede afectar a sus alrededores por las luces de control, motores y otros actuadores. El microcontrolador en la placa se programa mediante el lenguaje de programación Arduino (basado en *Wiring*) y el entorno de desarrollo Arduino (basado en *Processing*). Proyectos de Arduino pueden ser independientes o se pueden comunicar con el *software* que se ejecuta en un ordenador (por ejemplo, *Flash*, *Processing*, *MaxMSP*).

Los circuitos pueden ser armados a mano o pueden ser comprados ya preensamblados; el *software* puede ser descargado de forma gratuita. Los diseños de referencia de hardware son disponibles bajo una licencia de código abierto, que son libres de adaptarlos a sus necesidades.

Arduino recibió una mención honorífica en la sección Digital Communities del Ars Electronica Prix 2006. Los fundadores de Arduino son: Massimo Banzi , David Cuartielles, Tom Igoe, Gianluca Martino, y David Mellis.

### **2.1.1.     *Wiring***

*Wiring* es un marco de programación de código abierto para los microcontroladores.

*Wiring* permite la escritura de *software* multiplataforma para controlar los dispositivos conectados a una amplia gama de placas de microcontrolador para crear todo tipo de codificación creativa, objetos interactivos, espacios o experiencias físicas. El marco es creado cuidadosamente con diseñadores y artistas para fomentar una comunidad de principiantes a través de expertos y todo el mundo para compartir ideas, conocimientos y su experiencia colectiva. Hay miles de estudiantes, artistas, diseñadores, investigadores y aficionados que utilizan cableado para el aprendizaje, la creación de prototipos y la producción de un trabajo profesional terminado.

### **2.1.2.     *Processing***

Es un lenguaje de programación, entorno de desarrollo y la comunidad en línea. Desde el año 2001, la producción ha promovido la alfabetización de *software* dentro de las artes visuales y la cultura visual dentro de la tecnología. Inicialmente, creado para servir como un cuaderno de bocetos de *software* y para enseñar los fundamentos de programación de computadoras dentro de un contexto visual, su producción evolucionó hasta convertirse en una herramienta de desarrollo para los profesionales. Actualmente hay decenas de miles de estudiantes, artistas, diseñadores, investigadores y aficionados que utilizan el procesamiento para el aprendizaje, la creación de prototipos y producción.

## **2.2. Los microcontroladores de Arduino**

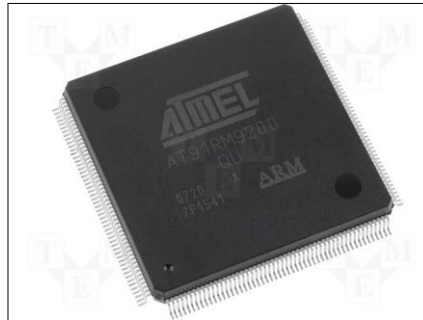
El microcontrolador fue inventado por *Texas Instruments* en la década de 1970, casi al mismo tiempo que el primer microprocesador que estaba siendo inventado en Intel. Los primeros microcontroladores eran simplemente microprocesadores con una función de memoria, como la memoria RAM y ROM. Más tarde, los microcontroladores se desarrollaron en una amplia gama de dispositivos diseñados para aplicaciones de sistemas integrados específicos en dispositivos: automóviles, teléfonos móviles y electrodomésticos.

### **2.2.1. El microcontrolador**

Un microcontrolador (abreviado  $\mu\text{C}$ , UC o MCU) es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales que cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida.

Algunos microcontroladores pueden utilizar palabras de cuatro *bits* y funcionan a velocidad de reloj con frecuencias tan bajas como 4 kHz, con un consumo de baja potencia (mW o microvatios). Por lo general, tendrá la capacidad para mantener la funcionalidad a la espera de un evento como pulsar un botón o de otra interrupción, el consumo de energía durante el sueño (reloj de la CPU y los periféricos de la mayoría) puede ser solo nanovatios, lo que hace que muchos sean adecuados para aplicaciones con batería de larga duración. Otros microcontroladores pueden servir para roles de rendimiento crítico, donde sea necesario actuar más como un procesador digital de señal (DSP), con velocidades de reloj y consumo de energía más altos.

**Figura 8. Microcontrolador**



Fuente: <http://www.tme.eu/>. Consulta: 10 de enero de 2013.

Al ser fabricadas, la memoria ROM del microcontrolador no posee datos. Para que pueda controlar algún proceso es necesario generar o crear y luego grabar en la EEPROM o equivalente del microcontrolador, algún programa que puede ser escrito en lenguaje ensamblador u otro lenguaje para microcontroladores; sin embargo, para que el programa pueda ser grabado en la memoria del microcontrolador, debe ser codificado en sistema numérico hexadecimal que es finalmente el sistema que hace trabajar al microcontrolador cuando es alimentado con el voltaje adecuado y asociado a dispositivos analógicos y discretos para su funcionamiento.

#### **2.2.1.1. Características**

Los microcontroladores son diseñados para reducir el costo económico y el consumo de energía de un sistema en particular. Por eso, el tamaño de la unidad central de procesamiento, la cantidad de memoria y los periféricos incluidos dependerán de la aplicación. El control de un electrodoméstico sencillo como una batidora, utilizará un procesador muy pequeño (4 u 8 bits) porque sustituirá a un autómata finito. En cambio, un reproductor de música y/o vídeo



digital (MP3 o MP4) requerirá de un procesador de 32 o 64 *bits* y de uno o más codificadores-decodificadores de señal digital (audio y/o vídeo). El control de un sistema de frenos ABS (*antilock brake system*, sistema antibloqueo de ruedas o frenos antibloqueo) se basa normalmente en un microcontrolador de 16 *bits*, al igual que el sistema de control electrónico del motor en un automóvil.

## 2.2.2. Modelos de Arduino

La tabla II muestra una comparación rápida entre las características de todos los modelos de Arduino.

Tabla II. Características de los modelos de Arduino

Nombre	Procesador	Reloj CPU	In/Out Analógicas	Digital IO/PWM	EEPROM [KB]
Uno	ATmega328	16 Mhz	6/0	14/6	1
Due	AT91SAM3X8E	84 Mhz	12/2	54/12	-
Leonardo	ATmega32u4	16 Mhz	12/0	20/7	1
Mega 2560	ATmega2560	16 Mhz	16/0	54/15	4
Mega ADK	ATmega2560	16 Mhz	16/0	54/15	4
Micro	ATmega32u4	16 Mhz	12/0	20/7	1
Mini	ATmega328	16 Mhz	8/0	14/6	1
Nano	ATmega168 ATmega328	16 Mhz	8/0	14/6	0,51 2
Ethernet	ATmega328	16 Mhz	6/0	14/4	1
Esplora	ATmega32u4	16 Mhz	-	-	1
ArduinoBT	ATmega328	16 Mhz	6/0	14/6	1
Fio	ATmega328P	8 Mhz	8/0	14/6	1
Pro (168)	ATmega168	8 Mhz	6/0	14/6	0,51 2
Pro (328)	ATmega328	16 Mhz	6/0	14/6	1
Pro Mini	ATmega168	8 Mhz 16Mhz	6/0	14/6	0,51 2
LilyPad	ATmega168V ATmega328V	8 Mhz	6/0	14/6	0,51 2
LilyPad USB	ATmega32u4	8 Mhz	4/0	9/4	1
LilyPad Simple	ATmega328	8 Mhz	4/0	9/4	1
LilyPad SimpleSnap	ATmega328	8 Mhz	4/0	9/4	1

Fuente: *Arduino*. <http://Arduino.cc/en/Products.Compare>. Consulta: 21 de mayo de

2014

### 2.2.3. Microcontroladores AVR de Atmel

Los AVR son una familia de microcontroladores RISC del fabricante estadounidense Atmel. La arquitectura de los AVR fue concebida por dos estudiantes en el Norwegian Institute of Technology, posteriormente, refinada y desarrollada en Atmel Norway, la empresa subsidiaria de Atmel, fundada por los dos arquitectos del chip. Cuenta con bastantes aficionados debido a su diseño simple y la facilidad de programación. Se pueden dividir en los siguientes grupos:

- ATxmega: procesadores muy potentes con 16 a 384 kB de memoria flash programable, encapsulados de 44, 64 y 100 pines (A4, A3, A1), capacidad de DMA (acceso directo a memoria), eventos, criptografía y amplio conjunto de periféricos con DAC (convertor de digital a analógico).
- ATmega: microcontroladores AVR grandes con 4 a 256 kB de memoria flash programable, encapsulados de 28 a 100 pines, conjunto de instrucciones extendido (multiplicación y direccionamiento de programas mayores) y amplio conjunto de periféricos.
- ATtiny: pequeños microcontroladores AVR con 0,5 a 8 kB de memoria flash programable, encapsulados de 6 a 20 pines y un limitado set de periféricos.
- AT90USB: ATmega integrado con controlador USB.
- AT90CAN: ATmega con controlador de bus CAN (protocolo de comunicación, *controller area network*).

- Tipos especiales: algunos modelos especiales, por ejemplo, para el control de los cargadores de baterías, pantallas LCD y los controles de los motores o la iluminación.
- AT90S: tipos obsoletos, los AVR clásicos.

Bajo el nombre AVR32, Atmel tiene una arquitectura RISC de 32 *bits* con soporte de DSP y SIMD. A pesar de la similitud de sus nombres y logotipos, las dos arquitecturas tienen poco en común.

El AVR es una CPU de arquitectura Harvard. Tiene 32 registros de 8 *bits*. Algunas instrucciones solo operan en un subconjunto de estos registros. La concatenación de los 32 registros, los registros de entrada/salida y la memoria de datos conforman un espacio de direcciones unificado, al cual se accede a través de operaciones de carga/almacenamiento. A diferencia de los microcontroladores PIC, el stack se ubica en este espacio de memoria unificado y no está limitado a un tamaño fijo.

### **2.3. Arduino Mega 2560**

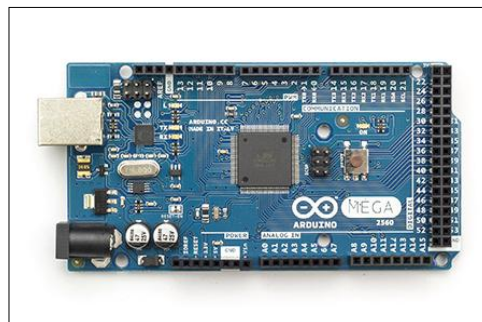
Arduino es una plataforma de *hardware* libre, basada en una placa con un microcontrolador y un entorno de desarrollo, diseñada para facilitar el uso de la electrónica en proyectos multidisciplinarios.

El Arduino Mega 2560 es una placa electrónica basada en el ATmega2560 (microcontrolador). Tiene 54 entradas / salidas digitales (de los cuales 15 pueden utilizarse para salidas PWM), 16 entradas analógicas, 4 UART (puertas seriales), un 16MHz oscilador de cristal, una conexión USB, un conector de alimentación, un header ICSP y un botón de reset. Contiene todo lo necesario

para apoyar el microcontrolador, simplemente conectarlo a un ordenador con un cable USB, o alimentarla con un adaptador de corriente AC a DC para empezar.

El Arduino Mega 2560 difiere de todas las placas anteriores ya que no utiliza el chip controlador de USB a serial FTDI. En su lugar, ofrece las ATmega16U2 programado como convertidor USB a serie.

Figura 9. **Placa de Arduino Mega 2560**



Fuente: *Arduino*. <http://Arduino.cc/es/Main/ArduinoBoardMega#.Uy0GrKh5POQ>. Consulta: 21 de mayo de 2014.

Tabla III. **Características de Arduino Mega 2560**

Característica	Descripción
Microcontrolador	ATmega2560
Tensión de funcionamiento	5V
Voltaje de entrada (recomendado)	7-12V
Voltaje de entrada (límites)	6-20V
Pines digitales de entrada / salida	54 (de los cuales 15 proporcionan salida PWM)
Pines de entrada analógica	16
Corriente continua para las entradas / salidas pin	40 mili amperios (mA)
Corriente de la CC para Pin 3.3V	50 mili amperios (mA)
Memoria flash	256 KB, 8 KB utilizado por el gestor de arranque
SRAM	8 KB
EEPROM	4 KB
Velocidad del reloj	16 MHz

Fuente: *Arduino*. <http://Arduino.cc/es/Main/ArduinoBoardMega#.Uy0GrKh5POQ>. Consulta: 25 de mayo de 2014.

### **2.3.1. Alimentación**

El Mega Arduino puede ser alimentado a través de la conexión USB o con una fuente de alimentación externa. La fuente de alimentación se selecciona automáticamente.

Alimentación externa (no USB) de potencia puede venir con un adaptador de AC-DC o la batería. El adaptador se puede conectar al conectar un centro positivo enchufe de 2,1 mm en el conector de alimentación de la placa.

La tarjeta puede funcionar con un suministro externo de 6 a 20 voltios. Si se proporcionan menos de 7V, no obstante, el pin de 5V puede suministrar menos de 5 voltios. Si se utiliza más de 12V, el regulador de voltaje se puede sobrecalentar y dañar la placa. El rango recomendado es de 7 a 12 voltios. Los pines de alimentación son:

- VIN. La tensión de entrada a la placa Arduino cuando se trata de utilizar una fuente de alimentación externa (en contraposición a 5 voltios de la conexión USB o de otra fuente de alimentación regulada). Se puede suministrar tensión a través de este pin o, si el suministro de tensión a través de la toma de poder, acceder a ella a través de este pin.
- 5V. Este pin como salida una 5V regulado por el regulador en el tablero. El tablero puede ser alimentado ya sea desde el conector de alimentación de CC (7 - 12), el conector USB (5V) o el pasador de VIN del tablero (7-12V). El suministro de tensión a través de los 5V o 3,3 V pines no pasa por el regulador, y puede dañar su tablero. No se aconseja hacer esto.

- 3,3 V. Un suministro de 3,3 voltios generados por el regulador a bordo. El drenaje actual máximo es de 50 mA.
- GND, pin para tierra.
- IOREF. Este pin de la placa Arduino proporciona la referencia de tensión con la que opera el microcontrolador. Un escudo configurado puede leer el voltaje pin IOREF y seleccionar la fuente de alimentación adecuada o habilitar traductores tensión en las salidas para trabajar con los 5V o 3,3 V.

### **2.3.2. Entradas y salidas**

Cada uno de los 54 pines digitales en el Mega se puede utilizar como una entrada o salida, utilizando las funciones `pinMode()`, `digitalWrite()`, y `digitalRead()`. Operan a 5 voltios. Cada pin puede proporcionar o recibir un máximo de 40 mA y tiene una resistencia de pull-up (desconectado por defecto) de 20-50 kOhms.

El Mega 2560 tiene 16 entradas analógicas, cada una de las cuales proporcionan 10 bits (en binario) de resolución (es decir, 1,024 valores diferentes). Por defecto se miden desde el suelo a 5 voltios, aunque es posible cambiar el extremo superior de su rango utilizando el pin AREF y función `analogReference()`.

### 2.3.3. Programación

La plataforma Arduino se programa mediante el uso de un lenguaje propio basado en el lenguaje de programación de alto nivel de procesamiento. Sin embargo, es posible utilizar otros lenguajes de programación y aplicaciones populares en Arduino, debido a que Arduino usa la transmisión serial de datos soportada por la mayoría de los lenguajes mencionados. Para los que no soportan el formato serie de forma nativa, es posible utilizar software intermediario que traduzca los mensajes enviados por ambas partes para permitir una comunicación fluida.

Arduino está basado en C y soporta todas las funciones del estándar C y algunas de C++ (C y C++ son lenguajes de programación). En la tabla IV se muestra un resumen con la estructura y sintaxis del lenguaje Arduino.

Tabla IV. **Sintaxis básicas del lenguaje Arduino**

Descripción	Sintaxis
Delimitadores	;, {}
Comentarios	//, /* */
Cabeceras	#define, #include
Operadores aritméticos	+, -, *, /, %
Asignación	=
Operadores de comparación	==, !=, <, >, <=, >=
Operadores booleanos	&&,   , !
Operadores de acceso a punteros	*, &
Operadores de bits	&,  , ^, ~, <<, >>
Incremento y decremento de variables	++, --
Asignación y operación	+=, -=, *=, /=, &=,  =

Fuente: *Arduino*. <http://Arduino.cc/en/Reference/HomePage>.

Consulta: 20 de octubre de 2014.





### **3. TECNOLOGÍA COMPLEMENTARIA**

Abarca todas aquellas tecnologías necesarias para poder visualizar la información del dispositivo, ser manipulado para ingresar datos o navegar por el menú y proteger el dispositivo para que no pueda ser modificados los horarios ingresados.

#### **3.1. Entorno gráfico**

Para poder mostrarle información al usuario del dispositivo como tiempo, menú, opciones, etc, se contemplan las siguientes opciones:

##### **3.1.1. LCD 16X2**

La pantalla de cristal líquido o LCD (*liquid crystal display*) es un dispositivo microcontrolado de visualización gráfico para la presentación de caracteres, símbolos o incluso dibujos (en algunos modelos); en este caso dispone de 2 filas de 16 caracteres cada una y cada carácter dispone de una matriz de 5x7 puntos (píxeles), aunque los hay de otro número de filas y caracteres. Este dispositivo está gobernado internamente por un microcontrolador y regula todos los parámetros de presentación, este modelo es el más comúnmente usado y esta información se basará en el manejo de este u otro LCD compatible.

##### **3.1.1.1. Características principales**

- Pantalla de caracteres ASCII, además, de los caracteres Kanji y griegos

- Desplazamiento de los caracteres hacia la izquierda o la derecha
- Proporciona la dirección de la posición absoluta o relativa del carácter
- Memoria de 40 caracteres por línea de pantalla
- Movimiento del cursor y cambio de su aspecto
- Permite que el usuario pueda programar 8 caracteres
- Conexión a un procesador usando un interfaz de 4 u 8 *bits*

### 3.1.1.2. Descripción de pines

Para comunicarse con la pantalla LCD se puede hacer por medio de sus pines de entrada de dos maneras posibles: con bus de 4 bits o con bus de 8 bits; en la tabla IV se observan las dos maneras posibles de conexión el LCD con un microcontrolador.

Tabla V. Descripción de pines

Núm. de <sup>o</sup> PIN	Abreviatura	Descripción
1	VSS	GND
2	VDD	VDD (5 V)
3	V0	Contraste ( 0 a +5V )
4	RS	Registro de control/Registro de datos: RS=0 selección registro de control RS=1 selección registro de datos
5	R/W	Lectura/Escritura: R/W=0 escritura ( <i>write</i> ) R/W=1 lectura ( <i>read</i> )
6	E	Habilitación: E=0 deshabilitado E=1 habilitado
7-14	D0-D7	Bus de datos bidireccional

Fuente: Scribd. *Manejo del LCD*. <http://es.scribd.com/doc/44252680/LCD-16X2>.

Consulta: 29 de enero de 2015.

### 3.1.2. GLCD

Una pantalla gráfica de cristal líquido o GLCD (acrónimo del inglés *graphic liquid crystal display*) es una pantalla plana formada por una matriz de píxeles monocromos colocados delante de una fuente de luz o reflectora. A menudo se utiliza en dispositivos electrónicos de pilas, ya que utiliza cantidades muy pequeñas de energía eléctrica; hay versiones de pantallas con diferentes controladores embebidos: Samsung KS0107, Samsung KS0108 o el Toshiba T6963.

Dispone de una memoria RAM interna del mismo tamaño de la capacidad que dispone la pantalla, por ejemplo, si una pantalla tiene un tamaño de 128 píxeles de largo por 64 píxeles de alto (128 x 64) tiene una memoria RAM interna de la misma capacidad (128 x 64). Los diferentes tamaños que existen son:

- 240 x 128
- 240 x 64
- 160 x 128
- 128 x 128
- 128 x 66
- 128 x 64
- 96 x 65

Para una GLCD de 128 x 64 los pines de conexión son los siguientes:

- PIN 1: VSS (conexión a tierra).
- PIN 2: VDD (conexión de alimentación  $\pm 5$  V).

- PIN 3: V0 (voltaje de salida negativa, por lo general es usada con un potenciómetro con el PIN 18 para el ajuste del contraste de los pixeles).
- PIN 4: D/I (datos de E/S para el cambio de registro).
- PIN 5: R/W (determina si los datos se van a leer o escribir).
- PIN 6: E (habilita la comunicación con la GLCD).
- PIN 7 - 14 (especifica un dato de 8-Bits de información).
- PIN15: CS1 (indica si se selecciona la primera mitad de la pantalla, pixeles 0-63).
- PIN16: CS2 (indica si se selecciona la segunda mitad de la pantalla, pixeles 64-127).
- PIN17: RESETB (señal de reinicio, funciona de varias formas dependiendo de la ocasión).
- PIN18: VEE (conexión de ajuste de contraste de los pixeles).
- PIN19: A (conexión positiva de la luz de fondo, por lo general son +5V).
- PIN20: K (conexión negativa de la luz de fondo, por lo general es tierra).

### 3.1.3. Pantalla LCD TFT de Arduino

TFT-LCD (*thin film transistor-liquid crystal display*, pantalla de cristal líquido de transistores de película fina) es una variante de pantalla de cristal líquido (LCD) que usa tecnología de transistor de película delgada (TFT) para mejorar su calidad de imagen. Las LCD de TFT son un tipo de LCD de matriz activa, aunque esto es generalmente sinónimo de LCD. Son usados en televisores, visualizadores de pantalla plana y proyectores. En informática, los monitores de TFT han desplazado la tecnología de CRT (*cathode ray tube*, tubo de rayos catódicos); comúnmente, disponibles en tamaños de 12 a 30 pulgadas.

La pantalla TFT de Arduino es una pantalla LCD retroiluminada con encabezados. Puede dibujar texto, imágenes y formas a la pantalla con la biblioteca de TFT. Hay una ranura para tarjetas micro-SD de a bordo en la parte posterior de la pantalla que pueden, entre otras cosas, las imágenes de mapa de bits para almacenar la pantalla para mostrar.

Encabezados de la pantalla están diseñados para encajar en el zócalo en la parte delantera de la Arduino Esplora, pero es compatible con cualquier Arduino basado en AVR (Uno, Leonardo, etc.) o con el Arduino Mega.

Las características son las siguientes:

- Pantalla LCD de 3,2" pulgadas
- Panel táctil resistivo
- Soporte lector de tarjeta TF
- Soporte de librería gráfica de código abierto

Las especificaciones de la pantalla LCD TFT de Arduino son:

- Resolución de 320 x 240, formato RGB565.
- LCD de 8 bits de bus de datos.
- Soporte lector de tarjeta TF.
- Controlador de panel táctil ADS7843.
- La pantalla funciona con 5 VDC.
- La ranura micro-SD se puede acceder a través de la biblioteca de la tarjeta SD.
- La retroiluminación LED es regulable mediante PWM.

Tabla VI. Descripción de pines

TFT LCD Pines	Arduino UNO Pines	Arduino Mega 2560 Pines	Descripción
0-7	0-7 (PD0-PD7)	0-7 (PE0,PE1,PE4,PE5,PG5,PE3,PH3,PH4)	Bus de datos [0:7]
9	9	9	TF_CS
10	10	10	Touch_Select(TCS)
11	11	11	MOSI (DIN)
12	12	12	MISO(DOUT)
A0	A0	A0	Chipselect(CS)
A1	A1	A1	Data/Command_Select D/C(RS)
A2	A2	A2	Write (WR)
A3	A3	A3	Reset (RST)
A4	A4	A4	Touch_IRQ (IRQ)
A5	A5	A5	Latch signal (ALE)

Fuente: Guía del usuario del escudo LCD. [http://www.openhardware.pe/wp-content/uploads/2013/03/3inch2\\_Arduino\\_LCD\\_Shield\\_UG.pdf](http://www.openhardware.pe/wp-content/uploads/2013/03/3inch2_Arduino_LCD_Shield_UG.pdf). Consulta: 20 de octubre de 2014.

El controlador de la pantalla táctil y el lector de la tarjeta TF comparte el mismo hardware de puertos SPI (Pin11, Pin12 y Pin13); son activados por diferentes señales de selección (la pantalla táctil utiliza el Pin10 mientras el lector TF utiliza Pin9).

#### **3.1.3.1. Bibliotecas utilizadas para la pantalla táctil**

La biblioteca de Arduino TFT extiende el Adafruit GFX y bibliotecas ST7735 Adafruit que se basa. La biblioteca GFX es responsable de las rutinas de dibujo, mientras que la ST7735 biblioteca es específica para la pantalla de Arduino.

La biblioteca TFT se basa en la librería SPI que debe ser incluida en todo programa se utilice la pantalla. Si desea usar la tarjeta SD, es necesario incluir la biblioteca SD también.

### **3.2. Manipulación**

Para poder ingresar datos, seleccionar un menú, apagar y encender el dispositivo, es necesario conocer las siguientes piezas para aplicarlas al diseño.

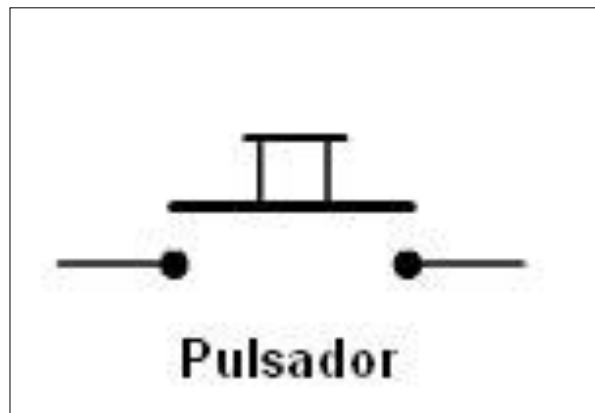
#### **3.2.1. Pulsadores**

Un botón o pulsador es un dispositivo utilizado para realizar cierta función. Los botones son de diversas formas y tamaño y se encuentran en todo tipo de dispositivos, aunque principalmente en aparatos eléctricos y electrónicos.

Los botones son por lo general activados al ser pulsados con un dedo. Permiten el flujo de corriente mientras son accionados. Cuando ya no se presiona vuelve a su posición de reposo.

Puede ser un contacto normalmente abierto en reposo NA o NO (*normally open* en inglés), o con un contacto normalmente cerrado en reposo NC.

Figura 10. **Símbolo de un pulsador**



Fuente: Informática enfurecida. *Elementos de control*. <http://electricidad-tekno.blogspot.com/2009/11/blog-post.html>. Consulta: 12 de febrero de 2013.

### **3.2.1.1. Funcionamiento**

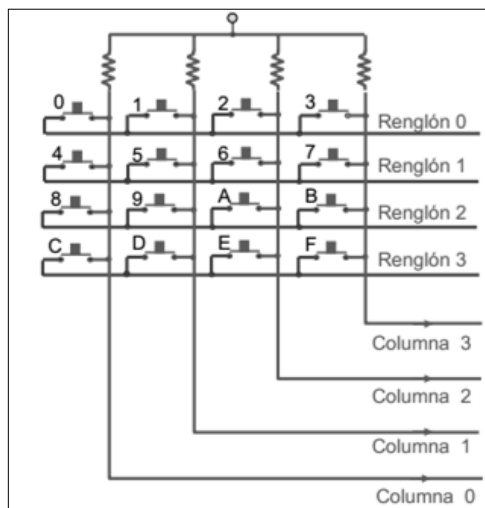
El botón de un dispositivo electrónico funciona, por lo general, como un interruptor eléctrico, es decir, en su interior tiene dos contactos, al ser pulsado uno, se activará la función inversa de la que en ese momento realiza; si es un dispositivo NA (normalmente abierto) será cerrado, si es un dispositivo NC (normalmente cerrado) será abierto.



### 3.2.2. Teclados matriciales 4x4

Los teclados matriciales son ensamblados en forma de matriz.

Figura 11. Diagrama de un teclado matricial de 4x4



Fuente: El teclado matricial. *Microcontroladores*. [http://galia.fc.uaslp.mx/~cantocar/microcontroladores/SLIDES\\_8051\\_PDF/20\\_MATRI.PDF](http://galia.fc.uaslp.mx/~cantocar/microcontroladores/SLIDES_8051_PDF/20_MATRI.PDF). Consulta: 12 de febrero de 2013.

En la figura 11 se muestra un teclado como una matriz de 4x4: 16 teclas configuradas en 4 columnas y 4 renglones.

Cuando no se ha oprimido ninguna tecla, (todas las teclas abiertas) no hay conexión entre renglones y columnas. Cuando se oprime una tecla se hace una conexión entre la columna y el renglón de la tecla.

Muchos teclados comerciales ya traen incluido su decodificador que escanea el teclado; una tecla es presionada, regresa un número que identifica

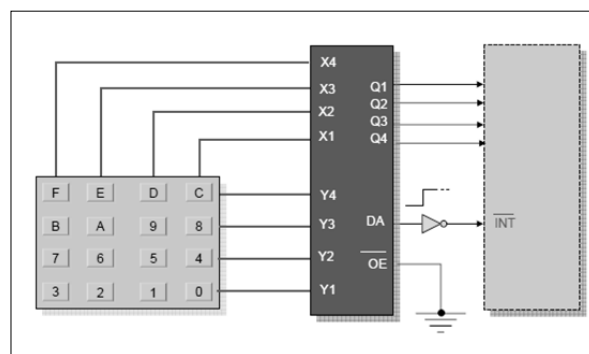
la tecla. Otra alternativa es adquirir por separado un chip decodificador y conectarlo al teclado.

El decodificador mostrado tiene 8 entradas: las 4 entradas X son conectadas a las 4 columnas del teclado y las 4 entradas Y son conectadas a los 4 renglones. No se muestran los capacitores que gobiernan la rapidez a la que se escanea el teclado.

Cuando se oprime una tecla el código de 4 bits de la tecla (con 16 teclas, los códigos están entre 0000 y 1111 en binario) aparecerá en las 4 líneas de salida y la línea de datos disponible (DA) se pone en bajo. Si se conecta a una línea de interrupción, el microprocesador será interrumpido cuando se oprima alguna tecla. La rutina de servicio de la interrupción, entonces lee los 4 bits y procesa el dato.

El chip del decodificador se encarga de eliminar el rebote de las teclas, lo que libera al programador de esta responsabilidad, esto es una ventaja al usar un chip decodificador.

Figura 12. **Decodificador de teclado de 4x4**



Fuente: El teclado matricial. *Microcontroladores*. [http://galia.fc.uaslp.mx/~cantocar/microcontroladores/SLIDES\\_8051\\_PDF/20\\_MATRI.PDF](http://galia.fc.uaslp.mx/~cantocar/microcontroladores/SLIDES_8051_PDF/20_MATRI.PDF). Consulta: 12 de febrero de 2013.

### **3.2.3. Pantalla LCD táctil TFT**

Una pantalla táctil es una pantalla que mediante un toque directo sobre su superficie permite la entrada de datos y órdenes al dispositivo. A su vez, muestra los resultados introducidos previamente; actuando como periférico de entrada y salida de datos, así como emulador de datos interinos erróneos al no tocarse efectivamente. Este contacto también se puede realizar por medio de un lápiz óptico u otras herramientas similares. Actualmente, hay pantallas táctiles que pueden instalarse sobre una pantalla normal, de cualquier tipo (LCD, monitores y televisores CRT, plasma, etc.).

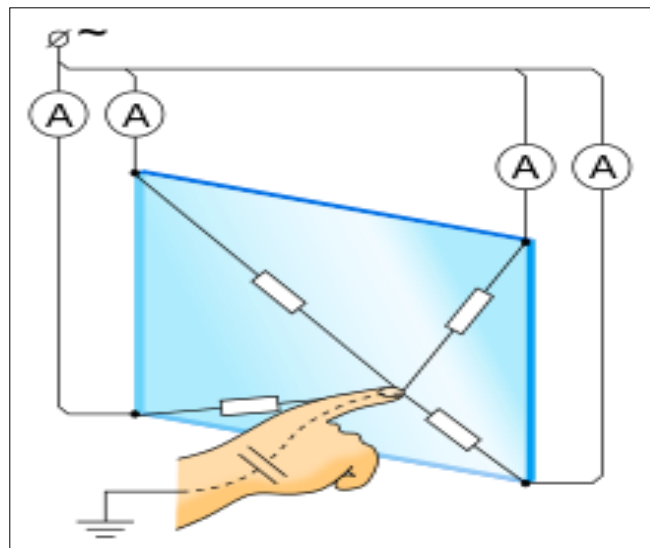
#### **3.2.3.1. Tipos**

Según la tecnología, hay dos tipos de pantallas táctiles de uso habitual:

- Resistivas: son más baratas y no les afectan el polvo ni el agua salada, además de ser más precisas, pueden ser usadas con un puntero o con el dedo. Sin embargo, tienen hasta un 25 % menos de brillo y son más gruesas, por lo que están siendo sustituidas por otras en los dispositivos móviles que precisan un tamaño y un peso ajustado y mayor brillo en la pantalla por la posibilidad de estar expuestos a la luz directa del sol.
- Capacitivas: basadas en sensores capacitivos, consisten en una capa de aislamiento eléctrico, como el cristal, recubierto con un conductor transparente, como el ITO (*tin-doped indium oxide*). Como el cuerpo humano es también un conductor eléctrico, tocando la superficie de la pantalla resulta una distorsión del campo electrostático de la pantalla, la cual es medida por el cambio de capacitancia (capacidad eléctrica).

Diferentes tecnologías pueden ser usadas para determinar en qué posición de la pantalla fue hecho el toque. La posición es enviada al controlador para el procesamiento. La calidad de imagen es mejor, tienen mejor respuesta y algunas permiten el uso de varios dedos a la vez (*multitouch*). Sin embargo, son más caras y no se pueden usar con puntero normal, sino con uno especial para las pantallas capacitivas.

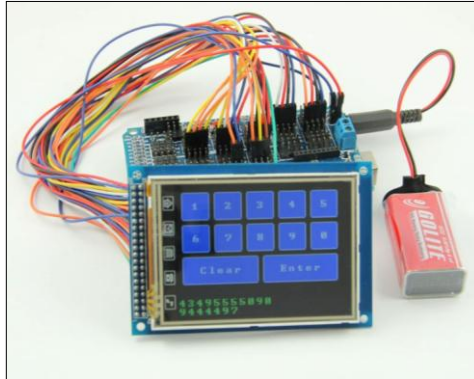
Figura 13. **Pantalla capacitiva y principio de funcionamiento**



Fuente: *Pantalla táctil*. [http://es.wikipedia.org/wiki/Pantalla\\_t%C3%A1ctil](http://es.wikipedia.org/wiki/Pantalla_t%C3%A1ctil). Consulta: 25 de febrero de 2015.

Pantalla LCD TFT es un módulo de pantalla táctil LCD. Tiene una interfaz 40 pines, tarjeta SD y lector de diseño de flash. Es un módulo de gran alcance y multifuncional para los proyectos con pantalla táctil. Está diseñado con un controlador táctil SSD1289, formado por una interfaz de 8/16bit, fácil de manipular.

Figura 14. **Pantalla LCD TFT de Arduino**



Fuente: Geeetech. <http://www.geeetech.com/2012/04/interface-3-2tft-lcd-module-to-Arduino/>.

Consulta: 25 de febrero de 2013.

#### **3.2.4. Manipulación con dispositivos móviles con sistema operativo Android**

Android es un sistema operativo basado en el kernel de Linux diseñado principalmente para dispositivos móviles con pantalla táctil: teléfonos inteligentes o tabletas.

Las aplicaciones se desarrollan habitualmente en el lenguaje Java con *Android Software Development Kit (Android SDK)*, pero están disponibles otras herramientas de desarrollo, incluyendo un kit de desarrollo nativo para aplicaciones o extensiones en C o C++, *Google App Inventor*, un entorno visual para programadores novatos y varios marcos de aplicaciones basadas en la web multiteléfono.

El desarrollo de aplicaciones para *Android* no requiere aprender lenguajes complejos de programación. Todo lo que se necesita es un conocimiento

aceptable de Java y estar en posesión del kit de desarrollo de *software* o SDK provisto por Google el cual se puede descargar gratuitamente.

Todas las aplicaciones están comprimidas en formato APK, que se pueden instalar sin dificultad desde cualquier explorador de archivos en la mayoría de dispositivos.

#### **3.2.4.1. Comunicación *Bluetooth***

Bluetooth es la norma que define un estándar global de comunicación inalámbrica que posibilita la transmisión entre diferentes equipos mediante un enlace por radiofrecuencia. Los principales objetivos de norma son:

- Facilitar las comunicaciones entre equipos móviles y fijos
- Eliminar cables y conectores
- Ofrecer la posibilidad de crear pequeñas redes inalámbricas

La solución que se da respecto a este método de comunicación consta de dos dispositivos:

- Un adaptador de USB a *Bluetooth*
- Un adaptador de *Bluetooth* a puerto serie RS232: PROMI-SD202

La configuración de estos dispositivos se realiza mediante un software que se incluyen con estos. Las velocidades de comunicación pueden ser desde 1 200 – 115 200 bps.

### **3.3. Control de tiempo y fechas**

Para poder controlar el tiempo de utilización de la consola de videojuego se puede hacer uso de los siguientes términos y dispositivos.

#### **3.3.1. Temporizador**

Un temporizador o minuterero es un dispositivo con frecuencia programable que permite medir el tiempo. En tiempos pasados, el ser humano utilizó, principalmente, los relojes de arena. Luego estos fueron sustituidos por relojes convencionales y, más tarde, por un dispositivo íntegramente electrónico. Cuando transcurre el tiempo configurado se hace saltar una alarma o alguna otra función a modo de advertencia.

En la actualidad, la mayor parte de los aparatos electrónicos, teléfonos móviles o computadoras personales, cuentan con una función de temporizador.

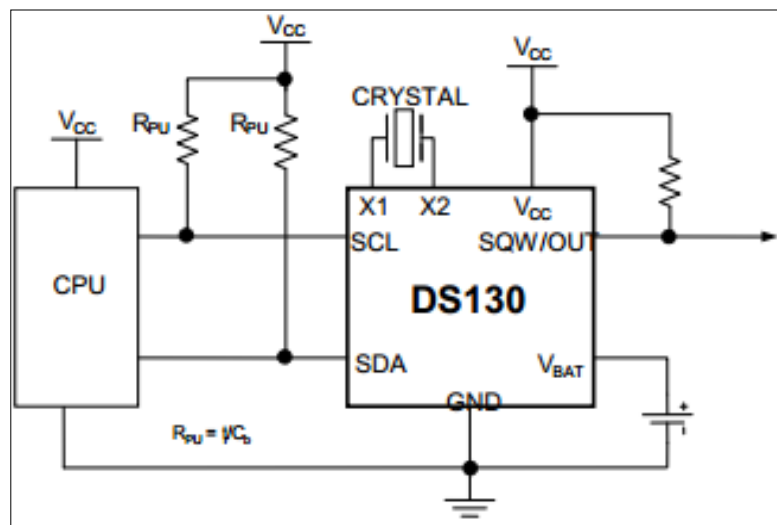
Un temporizador puede utilizarse, también, como un simulador de presencia que permite que un aparato electrónico (radio o luz) permanezca encendido durante un tiempo predeterminado, con el fin de prevenir robos. Igualmente, puede utilizarse para que un dispositivo conectado a la corriente eléctrica se conecte o desconecte en un momento dado. Esto es especialmente útil para aquellos aparatos que no cuentan con un temporizador propio o que no pueden programarse.

#### **3.3.2. Arduino DS1307 con AT24C32**

El DS1307 serial reloj de tiempo real (RTC) se caracteriza por ser de bajo consumo de energía, totalmente código binario decimal (*binary-coded decimal* BCD) reloj/calendario más 56 bytes de NV SRAM. Direcciones y datos son

transferidos serialmente a través de un I2C, bus bidireccional. El reloj/calendario provee segundos, minutos, horas, días, fechas, meses e información de años. El final de las fechas de los meses es automáticamente ajustada por mes con poco menos de 31 días, incluyendo los años bisiestos. El reloj opera en formato de 24 horas o 12 horas con indicador de AM/PM. El DS1307 tiene un *built-in* circuito sensor de fuente de poder que detecta si la fuente falla y automáticamente cambia a una fuente de backup.

Figura 15. **Diagrama del Arduino DS1307**



Fuente: *Apagado de televisión y consola*. [http://www.electronica60norte.com/detalle.php?sku=430Control de Encendido/](http://www.electronica60norte.com/detalle.php?sku=430Control%20de%20Encendido/) Consulta: 25 de febrero de 2013.

### 3.4. **Control de encendido/apagado de TV y consola de videojuego**

En esta parte del dispositivo es necesario controlar el voltaje AC que alimenta a la televisión, también su apagado y encendido. Para lo cual, se pueden utilizar los siguientes dispositivos:



### 3.4.1. 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 1835.

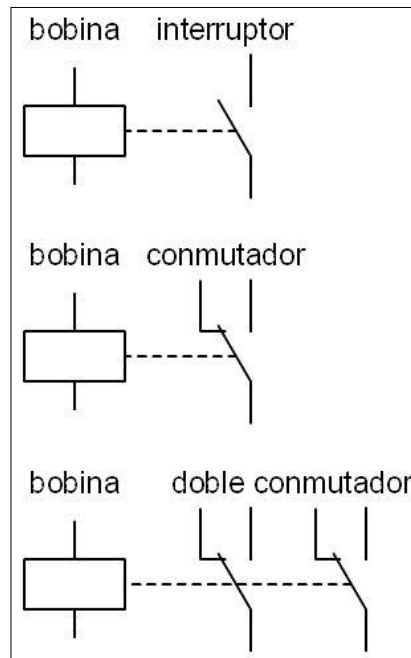
Dado que el relé es capaz de controlar un circuito de salida de mayor potencia que el de entrada, puede considerarse, en un amplio sentido, como un amplificador eléctrico. Como tal se emplearon en telegrafía, haciendo la función de repetidores que generaban una nueva señal con corriente procedente de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

Figura 16. Diferentes formas físicas de un relé



Fuente: Relé. <http://es.wikipedia.org/wiki/Rel%C3%A9>. Consulta: 28 de febrero de 2013.

Figura 17. **Diagrama de un relé**



Fuente: *Símbolos eléctricos de relés*. <http://www.simbologia-electronica.com/simbolos-electricos-electronicos/simbolos-reles.htm>. Consulta: 28 de febrero de 2013.

### 3.4.2. **Sensores Infrarrojos**

El sensor es un dispositivo electrónico/mecánico/químico que mapea un atributo ambiental resultando una medida cuantizada, normalmente un nivel de tensión eléctrica.

Particularmente, el sensor infrarrojo es un dispositivo electrónico capaz de medir la radiación electromagnética infrarroja de los cuerpos en su campo de visión. Todos los cuerpos reflejan una cierta cantidad de radiación, esta resulta invisible para los ojos pero no para estos aparatos electrónicos, ya que se encuentran en el rango del espectro justo por debajo de la luz visible.

#### **3.4.2.1. Fotodiodo receptor TSOP48**

La serie TSOP48 remite a los receptores infrarrojos para sistemas de control remoto. Un diodo PIN y un preamplificador están montados en un bastidor de conductores, el paquete de epoxi actúa como un filtro de infrarrojos. La señal de salida demodulada directamente puede ser decodificada por un microprocesador. El principal beneficio de la TSOP48 es la compatibilidad con todos los formatos de datos de control remoto IR. Este componente no ha sido calificado de acuerdo a las especificaciones de automoción, las características del TSOP48 son:

- Baja corriente de alimentación
- Detector de Fotos y preamplificador en un paquete
- Filtro interno para PCM Frecuencia
- Mejora Blindaje contra EMI
- Voltaje de alimentación: 2,7 a 5,5 V
- Mejora la inmunidad contra la luz ambiente
- Insensible a la alimentación de tensión de rizado y ruido

#### **3.5. Seguridad**

En la parte de seguridad del dispositivo, es necesario analizar la aplicación de los factores que se mencionarán a continuación.

### 3.5.1. Contraseña

Una contraseña o clave es una forma de autenticación que utiliza información secreta para controlar el acceso hacia algún recurso. La contraseña debe mantenerse en secreto para a quienes no se les permite el acceso. A quienes desean acceder a la información se les solicita una clave; si conocen o no conocen la contraseña, se concede o se niega el acceso a la información según sea el caso.

En la era tecnológica, las contraseñas son usadas comúnmente para controlar el acceso a sistemas operativos de computadoras protegidas, teléfonos celulares, decodificadores de TV por cable, cajeros automáticos, etc. Una computadora típica puede hacer uso de contraseñas para diferentes propósitos: conexiones a cuentas de usuario, acceder al correo electrónico de los servidores, acceder a bases de datos, redes, y páginas web e incluso para leer noticias en los periódicos (diarios) electrónicos.

En la lengua inglesa, se tienen dos denominaciones distintivas para las contraseñas: *password* (palabra de acceso) y *pass code* (código de acceso); la primera no implica necesariamente usar alguna palabra existente (sin embargo, es normal emplear alguna palabra familiar o de fácil memorización por parte del usuario), la primera suele asociarse también al uso de códigos alfanuméricos (también llamado PIT *personal identification text*), la segunda frecuentemente se liga a la utilización de algún código numérico (PIN, *personal identification number*). Esto ocurre igualmente en la lengua española ya que en ocasiones clave y contraseña se usan indistintamente.

### **3.5.2. Balance entre seguridad y comodidad**

Con excepciones, cuanto más fácil de recordar sea una contraseña, implicará una mayor facilidad para un atacante adivinarla. Es decir, si algún recurso está protegido por una contraseña, entonces, la seguridad se incrementa con el consecuente aumento de molestia para los usuarios. El nivel de seguridad es inherente dada una política de contraseñas en particular, que está influida por diversos factores. Sin embargo, no existe un método único que sea el mejor para definir un balance adecuado entre seguridad y comodidad de acceso.

Algunos sistemas protegidos por contraseñas plantean pocos o ningún riesgo a los usuarios si Estos se revelan, por ejemplo, una contraseña que permita el acceso a la información de un sitio *web* gratuito. Otros plantean un modesto riesgo económico o de privacidad. En estos casos, una contraseña utilizada para acceder al correo electrónico o alguna contraseña para algún teléfono celular. Aun así, en otras situaciones, puede tener consecuencias severas, si la contraseña es revelada. Por ejemplo, como las situaciones para limitar el acceso de expedientes sobre tratamientos del sida o el control de estaciones de energía.

### **3.5.3. Factores en la seguridad de un sistema de contraseñas**

La seguridad de un sistema protegido por contraseña depende de varios factores. El sistema debe estar diseñado para sondear la seguridad.

### **3.5.3.1. Posibilidades de que algún atacante pueda adivinar o inventar la contraseña**

La posibilidad de que algún atacante pueda proporcionar una contraseña que adivinó es un factor clave al determinar la seguridad de un sistema. Algunos sistemas imponen un límite de tiempo después de que sucede un pequeño número de intentos fallidos de proporcionar la clave. Al no tener otras vulnerabilidades, estos sistemas pueden estar relativamente seguros con simples contraseñas, mientras estas no sean fácilmente deducibles, al no asignar datos fácilmente conocidos como nombres de familiares o de mascotas, el número de matrícula del automóvil o contraseñas sencillas como administrador o 1234. Solo para citar algunos casos.

Otros sistemas almacenan o transmiten una pista a modo de sugerencia de recordatorio de la contraseña, de manera que la propia pista puede ser fundamental para el acceso de algún atacante. Cuando esto ocurre (y suele ser común), el atacante intentará suministrar contraseñas frecuentemente en una alta proporción, quizás utilizando listas extensamente conocidas de contraseñas comunes.

### **3.5.3.2. Formas de almacenar contraseñas**

Algunos sistemas almacenan contraseñas como archivos de texto. Si algún atacante gana acceso al archivo que contiene las contraseñas, entonces, todas estas se encontrarán comprometidas. Si algunos usuarios emplean la misma contraseña para diferentes cuentas, estarán comprometidas de igual manera. Los mejores sistemas almacenan las contraseñas en una forma de protección criptográfica. Así, el acceso a la contraseña será más difícil para algún espía que haya ganado el acceso interno al sistema, aunque la validación todavía sigue siendo posible.

Un esquema criptográfico común almacena solamente el texto de la contraseña codificado, conocido como *hash*. Cuando un usuario teclea la contraseña en este tipo de sistema, se genera a partir de la contraseña y mediante un algoritmo el código *hash* equivalente para esa contraseña, y si el resultante (*hash*) coincide con el valor almacenado, se permite el acceso al usuario.

El texto codificado de la contraseña se crea al aplicar una función criptográfica usando la contraseña. Otro valor es el conocido en el idioma inglés como *salt*. El *salt* previene que los atacantes construyan una lista de valores para contraseñas comunes. Las funciones criptográficas más comunes son la MD5 y SHA1. Una versión modificada de DES fue utilizada en los primeros sistemas Unix.

#### **3.5.4. Almacenamiento de datos en memorias EEPROM**

EPROM son las siglas de *erasable programmable read-only memory* (ROM programable borrrable). Es un tipo de chip de memoria ROM no volátil inventado por el ingeniero Dov Frohman. Está formada por celdas de FAMOS (*floating gate avalanche-injection metal-oxide semiconductor*) o transistores de puerta flotante, cada uno viene de fábrica sin carga, por lo que son leídos como 1 (por esa razón, una EPROM sin grabar se lee como FF en todas sus celdas).

Las memorias EPROM se programan mediante un dispositivo electrónico que proporciona voltajes superiores a los normalmente utilizados en los circuitos electrónicos. Las celdas que reciben carga se leen entonces como un cero 0.

Una vez programada, una EPROM se puede borrar solamente mediante exposición a una fuerte luz ultravioleta. Esto es debido a que los fotones de la luz excitan a los electrones de las celdas provocando que se descarguen. Las EPROM se reconocen fácilmente por una ventana transparente en la parte alta del encapsulado, a través de la cual se puede ver el chip de silicio que admite la luz ultravioleta durante el borrado.

Una EPROM programada retiene sus datos durante diez o veinte años y se puede leer un número ilimitado de veces. Para evitar el borrado accidental por la luz del sol, la ventana de borrado debe permanecer cubierta. Las antiguas BIOS de los ordenadores personales eran frecuentemente EPROM y la ventana de borrado estaba habitualmente cubierta por una etiqueta que contenía el nombre del productor de la BIOS, su revisión y una advertencia de copyright.

#### **3.5.4.1. Librería EEPROM**

El microcontrolador de la placa Arduino tiene 4KB de memoria EEPROM, cuyos valores se mantienen cuando la placa está apagada (como un pequeño disco duro). Esta librería permite leer y escribir esos bytes

#### **3.5.5. Procedimientos para cambiar las contraseñas**

Usualmente, un sistema debe proveer una manera de cambiar una contraseña, porque el usuario sospeche que la contraseña actual ha sido descubierta o como medida de precaución. Si la nueva contraseña es introducida en el sistema de una manera no cifrada, la seguridad puede haberse perdido incluso antes de que la nueva contraseña haya sido instalada en la base de datos. Si la nueva contraseña fue revelada a un empleado de



confianza, se gana poco. Algunas páginas web incluyen la opción de recordar la contraseña de un usuario de una manera no cifrada al mandárselo por correo electrónico.

Los sistemas de administración de identidad se utilizan cada vez más para automatizar la emisión de reemplazos para contraseñas perdidas. La identidad del usuario se verifica al realizar algunas preguntas y compararlas con las que se tienen almacenadas. Preguntas típicas son: ¿dónde naciste?, ¿cuál es tu película favorita?, ¿cuál es el nombre de tu mascota? En muchos casos las respuestas a estas preguntas pueden ser adivinadas, también, pueden ser determinadas con un poco de investigación u obtenidas a través de estafa con ingeniería social. Mientras que muchos usuarios han sido advertidos para que nunca revelen su contraseña, muy pocos consideran el nombre de su película favorita para requerir este tipo de seguridad.

### **3.5.6.      Diseño de *software* protegido**

Técnicas comunes utilizadas para mejorar la seguridad de sistemas de software protegidas por contraseñas incluyen:

- No repetir la contraseña en la pantalla de visualización cuando se está accediendo.
- Permitir contraseñas de una longitud adecuada (algunos sistemas de Unix limitan contraseñas a 8 caracteres).
- Obligar a que la contraseña tenga algún carácter especial y algún número.

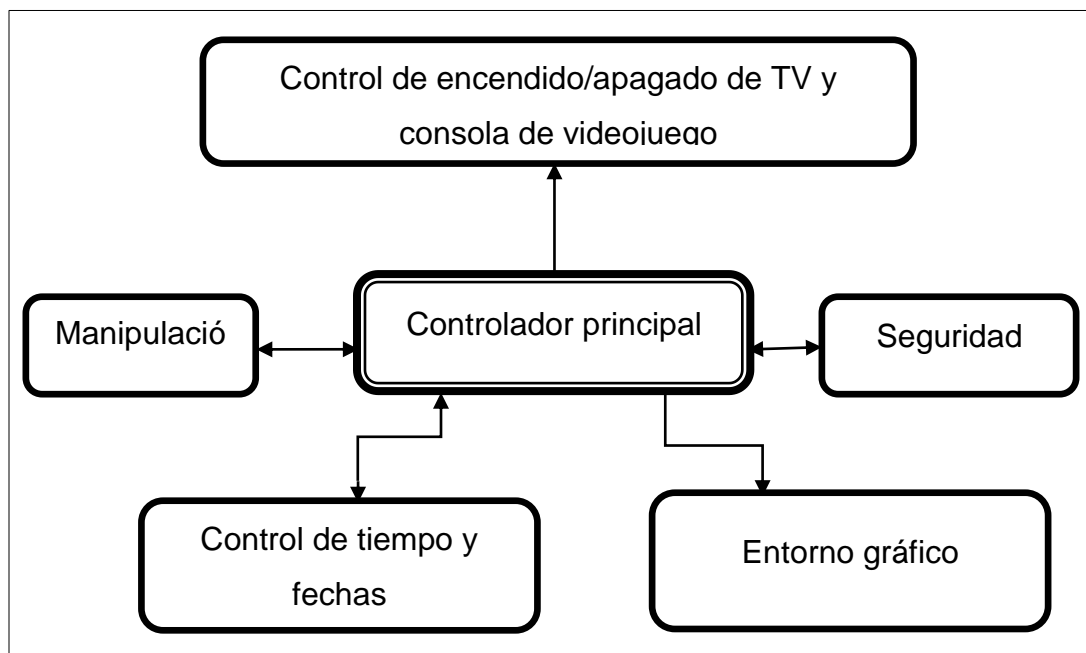
- Requerir a los usuarios volver a ingresar su contraseña después de un período de inactividad.
- Requerir periódicamente cambios de contraseña.
- Asignar contraseñas al azar.
- Proveer una opción alternativa al uso de teclados.
- Al cambiar la contraseña, comprobar que no se parece a las contraseñas anteriormente usadas.

## 4. DISEÑO DEL DISPOSITIVO

### 4.1. Descripción del diseño del dispositivo

El dispositivo está dividido en 5 partes, las cuales interactúan con el controlador principal, como lo muestra el esquema:

Figura 18. Partes del dispositivo



Fuente: elaboración propia.

Este esquema describe a grandes rasgos las diferentes partes del dispositivo; cada una abarca los recursos y las herramientas necesarias para que el dispositivo trabaje de forma adecuada, sea fácil de interactuar con el

usuario y cumpla con los requerimientos necesarios de control de las horas de juego de los niños.

El controlador principal es el que gobierna el dispositivo, es el encargado de controlar y recibir información de las diferentes partes del dispositivo, por lo tanto, como se observa en el esquema de la figura 18 hay flechas de doble vía y algunas solo son de una vía; esto se debe a que en algunas partes el controlador principal recibe y envía información; en otras partes solo recibe o envía información.

## **4.2. Partes del dispositivo**

### **4.2.1. Controlador principal**

El controlador principal es un Arduino Mega 2560; fue elegido debido a la cantidad de puertos digitales con los que cuenta, puertos I2C, cantidad de memoria EEPROM y velocidad de reloj de 16MHZ. Con todo, se logra cumplir con los requerimientos necesarios para el controlador principal que está encargado de interpretar y enviar información a cada una de las partes del dispositivo: el entorno gráfico, manipulación, etc.

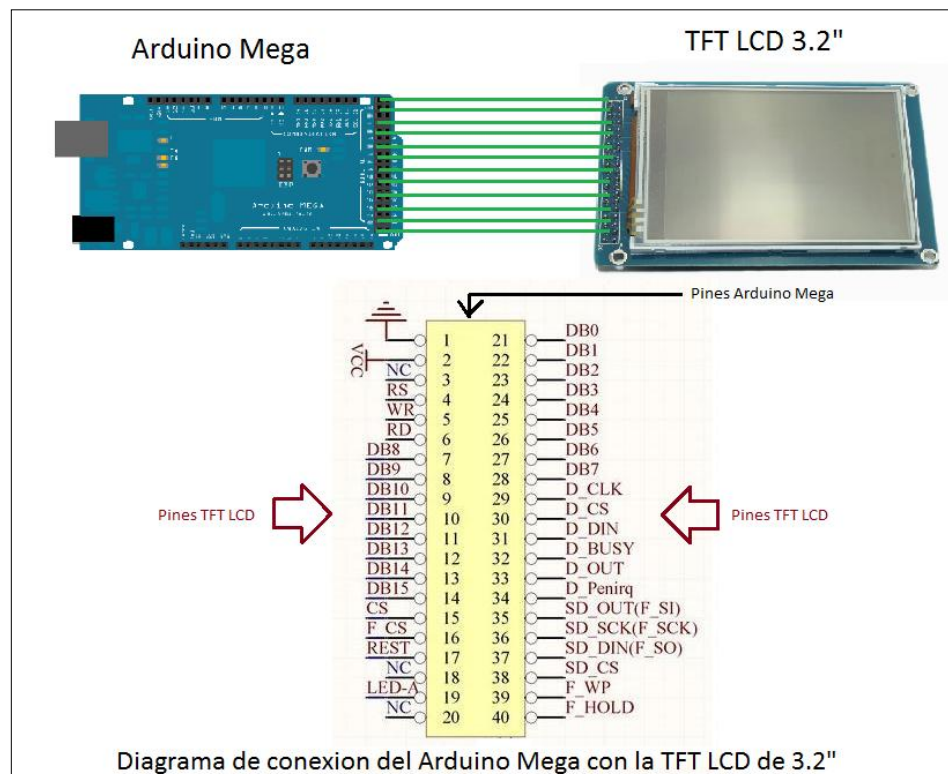
El Arduino Mega 2560 es programado por el *software* abierto propio de Arduino. Para utilizar los diferentes módulos (placas) con los que cuenta Arduino: LCD, *Bluetooth*, sensores, reles, etc. Es necesario instalar librerías para cada uno de los módulos.

Para cada módulo se describirá la librería de Arduino utilizada, también, una descripción de las sentencias utilizadas en el *software*.

#### 4.2.2. Entorno gráfico y manipulación del dispositivo

Estas dos partes del dispositivo se describen juntas debido al módulo de pantalla LCD TFT táctil de 3,2 pulgadas. No es necesario colocar un teclado matricial para manipular el dispositivo, en su lugar, se utilizará la pantalla táctil que es más amigable con el usuario y hace el dispositivo más compacto.

Figura 19. Diagrama de conexión de Arduino mega con la TFT LCD 3.2"



Fuente: elaboración propia.

La pantalla tiene una resolución de 320 x 240 pixeles, para poder iniciar a programar los botones táctiles en el *software* de Arduino es necesario instalar las librerías UTFT y UTouch, para poder utilizar sentencias específicas: dibujar

botones de menú, colocar letras o números a los botones, entre otras opciones disponibles.

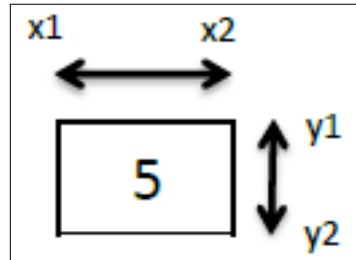
Las siguientes sentencias describen como dibujar una fila de 5 botones cuadrados en la pantalla LCD.

```
#include <UTFT.h>
#include <UTouch.h>
for (x=0; x<5; x++)
{
  myGLCD.setColor(0, 255, 0);
  myGLCD.fillRect (10+(x*60), 60, 60+(x*60), 110);
  myGLCD.setColor(255, 255, 255);
  myGLCD.drawRoundRect (10+(x*60), 60, 60+(x*60), 110);
  myGLCD.setColor(255, 255, 255);
  myGLCD.printNum1(x+1, 27+(x*60), 77);
}
```

Las sentencias que contienen un símbolo numeral al inicio seguidas por include son utilizadas para declarar las librerías a utilizar, en este caso: UTFT.h y la UTouch.h; posteriormente, se observa un ciclo o bucle for que va de aumento de 0 a 4 de uno en uno el valor de la variable x, la sentencia myGLCD.setColor es utilizada para colocarle al botón o al marco del mismo el color deseado, por ejemplo; El (0, 255, 0) es el código para el color verde. Los botones tienen un color verde en su interior y un marco blanco.

Para dibujar la forma de los botones es utilizada la sentencia myGLCD.fillRect(x1,y1,x2,y2) le da una forma cuadrada a los botones como lo muestra la figura 20.

Figura 20. **Ejemplo de botón dibujado en la pantalla LCD TFT**

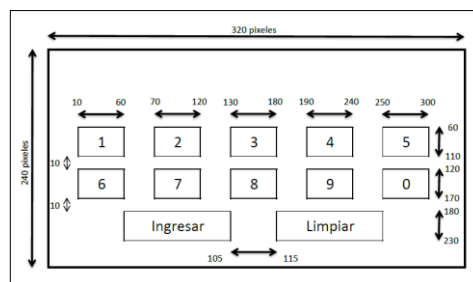


Fuente: elaboración propia.

En lugar de dibujar botón por botón, es posible crear funciones como la utilizada anteriormente, utilizando un ciclo for se creó la primera fila de 5 botones. Para dibujar la línea de marco blanco en el botón se utiliza la sentencia `myGLCD.drawRoundRect(x1,y1,x2,y2)` y por medio de una función se reduce la cantidad de líneas del programa para crear 5 botones.

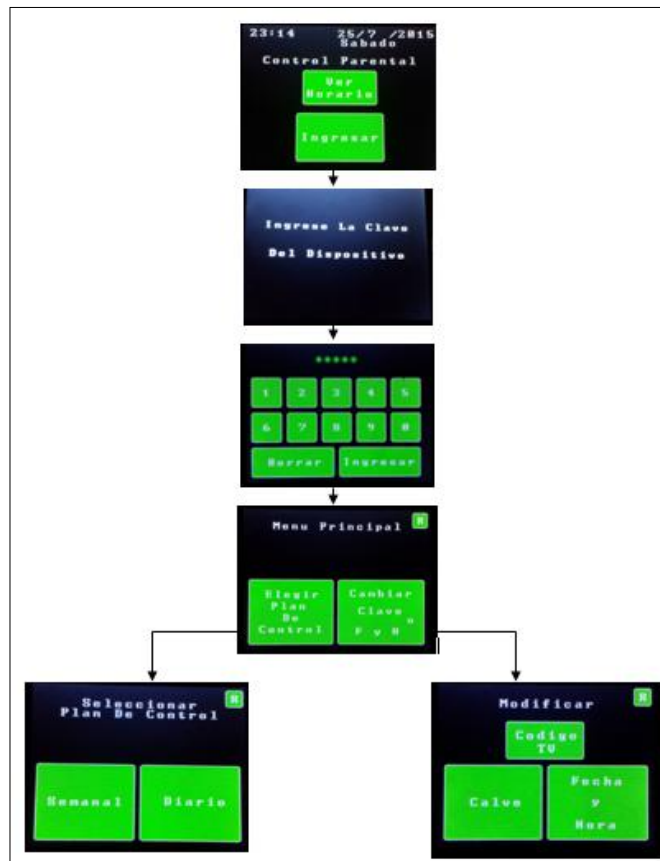
Por último, para colocar el número al centro del botón se utiliza la sentencia `myGLCD.printNumI(5,x,y)` donde 5 es el número a imprimir x y y las coordenadas de pixeles en donde se va a dibujar el número. Los botones se dibujarán con las coordenadas que se muestran en la figura 21.

Figura 21. **Coordenadas de botones en pantalla TFT LCD 3.2"**



Fuente: elaboración propia.

Figura 22. Diagrama de escalabilidad del menú del dispositivo



Fuente: elaboración propia.

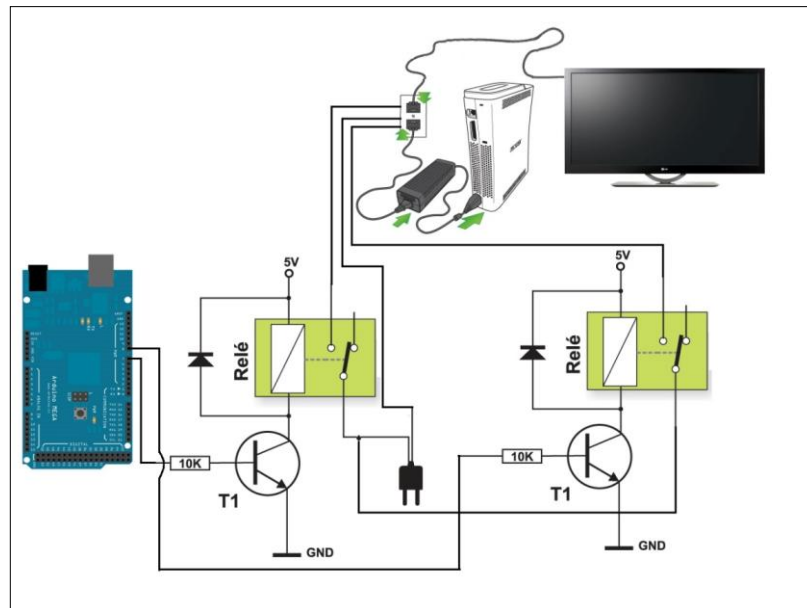
#### 4.2.3. Control de encendido/apagado de TV y consola de videojuego

El encendido/apagado de la TV es controlado por un diodo infrarrojo, simulando el control remoto de la TV. Esta parte del dispositivo, también, controla cuando se habilita la alimentación de 120V AC de la TV, como se puede observar en la figura 23. Con respecto a la consola, solo se controla la alimentación de voltaje 120V AC, para permitirle al usuario guardar los avances de sus juegos; pero como el dispositivo apaga la TV y le corta la alimentación,



solo el padre de familia puede habilitarle más tiempo para que pueda guardar sus partidas; esto solo es en caso de emergencia debido a que para evitar esto, el dispositivo alerta por medio de un *buzzer* que el tiempo de juego está por terminar.

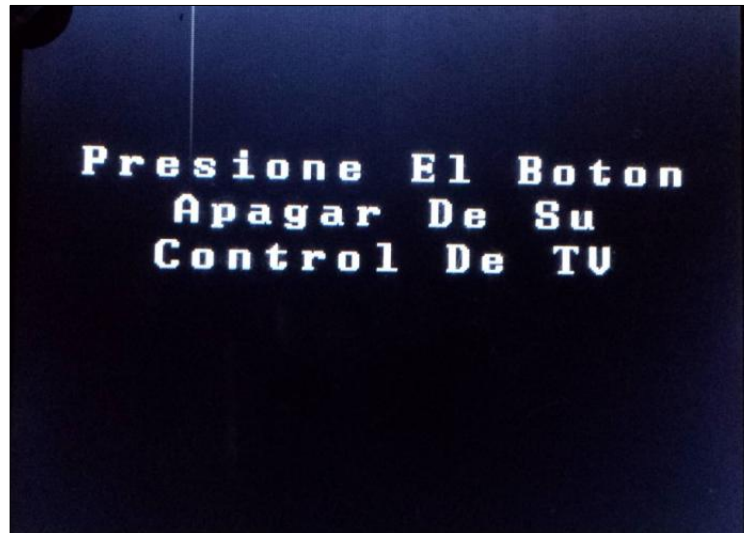
Figura 23. **Diagrama de circuito de control de alimentación de TV y consola de videojuego**



Fuente: elaboración propia.

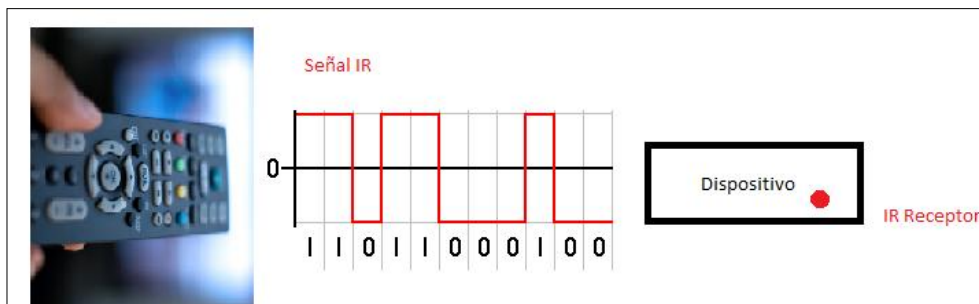
Cuando se utiliza el dispositivo por primera vez, este mostrará el mensaje, ver figura 24, que solicita que se presione el botón de apagado del control remoto de su TV dirigiendo el control remoto hacia el dispositivo, como se puede observar en la figura 25.

Figura 24. **Mensaje presionar botón de apagado de TV**



Fuente: elaboración propia.

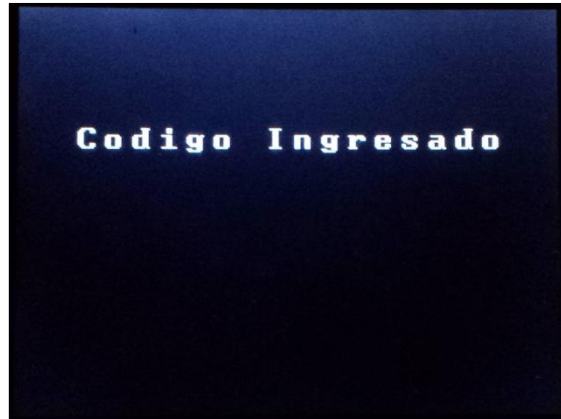
Figura 25. **Señal IR de control remoto de TV**



Fuente: elaboración propia.

El dispositivo recibirá el código enviado por el control remoto y mostrará el mensaje código ingresado como se observa en la figura 26.

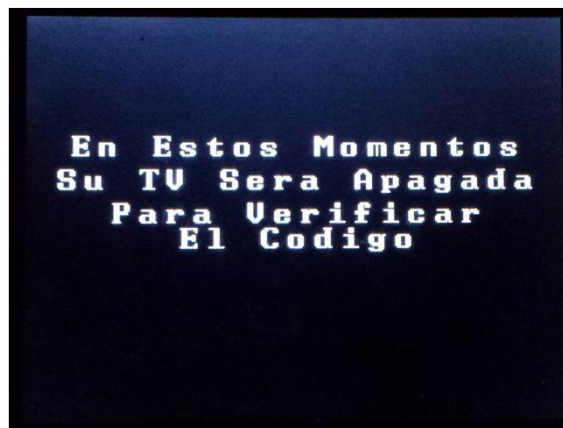
Figura 26. **Código ingresado**



Fuente: elaboración propia.

Para asegurarse de que el dispositivo realizó la decodificación correctamente, este realiza una prueba: envía el código de apagar la TV por el IR del dispositivo, antes, mostrará el mensaje como se ve en la figura 27.

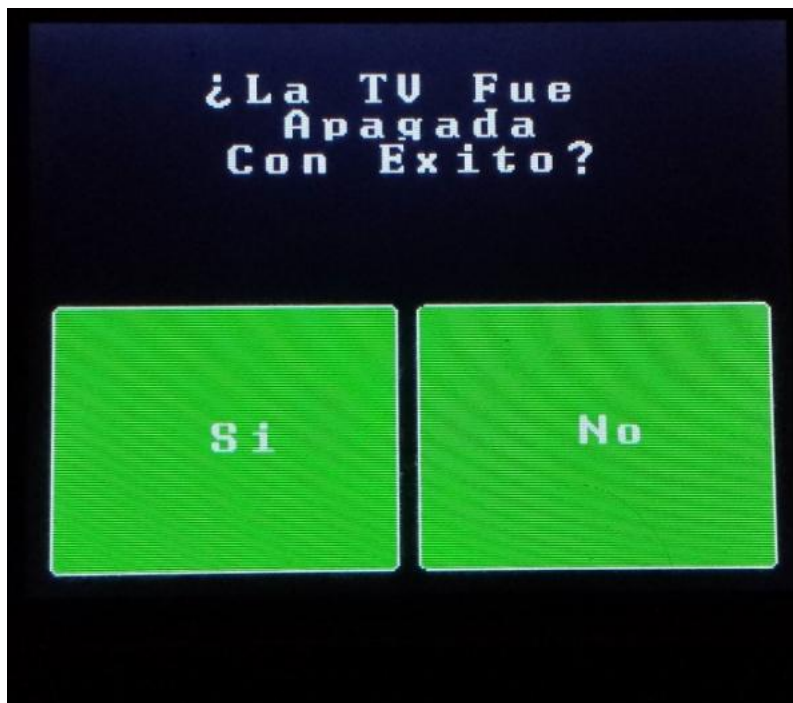
Figura 27. **Verificación del código de TV**



Fuente: elaboración propia.

Una vez se haya enviado el código mostrará el siguiente mensaje: ¿El dispositivo fue apagado con éxito?, como lo muestra la figura 28; si la TV no se apaga, el dispositivo; solicitará que se vuelva a presionar el botón de apagado del control remoto de la TV dirigiéndolo hacia el dispositivo, este proceso se realizará hasta que el dispositivo logre apagar la TV.

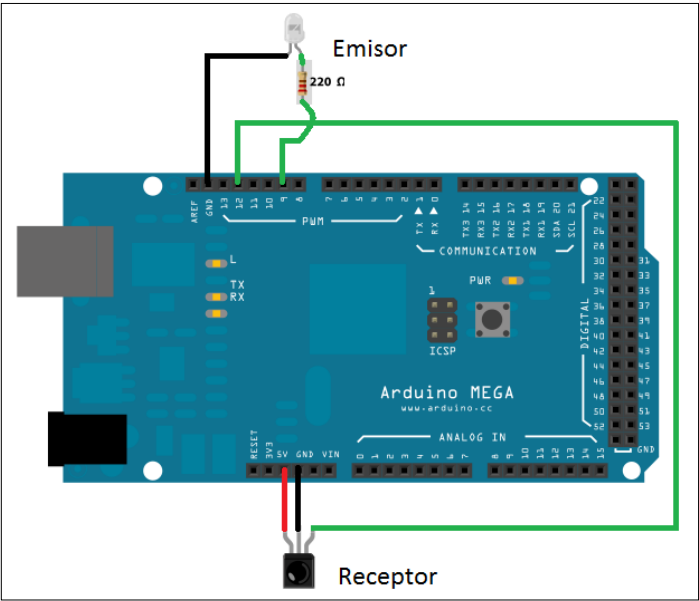
Figura 28. **Verificación de TV apagado con código ingresado**



Fuente: elaboración propia.

Una vez se logre apagar la TV, el código es guardado en la memoria EEPROM del Arduino mega. En la figura 29 se puede observar la conexión de los infrarrojos del receptor y del emisor.

Figura 29. Diagrama conexión de infrarrojo emisor y receptor.



Fuente: elaboración propia.

Para la decodificación y transmisión del código de apagado de la TV se utilizó la librería de Arduino IRremote.h. En la tabla VII se pueden observar las sentencias y variables necesarias para utilizar la librería.

Tabla VII. Declaración de variables y comandos para la librería IRremote.h

Núm.	Sentencia	Descripción
1	int RECV_PIN = 12;	Pin al cual está conectado el infrarrojo receptor al Arduino mega.
2	IRrecv irrecv(RECV_PIN);	Asignarle a la librería el pin del infrarrojo receptor.
3	decode_results codigo;	Guardar el código recibido.
4	IRsend IR_Transmisor;	Habilitar la transmisión IR en el pin 9 del Arduino mega.

Fuente: elaboración propia.

El siguiente código es el utilizado para decodificar el código recibido del control remoto de la TV.

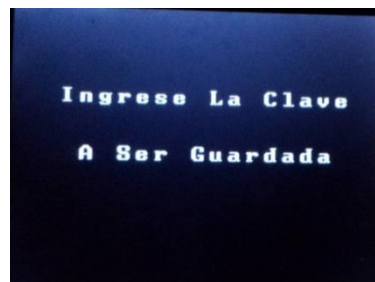
```
Tipo_Codigo = codigo.decode_type;//Decodificar el tipo de codigo
T_codigo = codigo.rawlen;
if (Tipo_Codigo == UNKNOWN)//El tipo de código es desconocido
{
    T_codigo--;
    // Necesitamos convertir de un momento a microsegundos
    for (int i = 1; i <= T_codigo; i++)
    {
        if (i % 2)
        { // Marka
            irBuffer[i-1] = codigo.rawbuf[i]*USECPERTICK - MARK_EXCESS;
        }
        else
        { // Spacio
            irBuffer[i-1] = codigo.rawbuf[i]*USECPERTICK + MARK_EXCESS;
        }
    }
}
else
{
    if (Tipo_Codigo == NEC) // Si el tipo de código es NEC
    { Valor_Codigo = codigo.value;//El valor del código se le asigna a la
variable Valor código.
    T_codigo = codigo.bits;// En la varibale T_codigo si asigna el código en
bits
    }
}
```

#### 4.2.4. Seguridad

Con el fin de que solo el padre de familia pueda manipular el dispositivo, este cuenta con seguridad para limitar el acceso al menú de configuración.

Al momento de utilizar el dispositivo por primera vez, éste solicitará que se ingrese la contraseña de 5 dígitos numéricos a ser utilizada para la seguridad del dispositivo. Como lo muestra la figura 30.

Figura 30. **Ingresar la clave a ser guardada**



Fuente: elaboración propia.

Figura 31. **Seguridad al ingresar los dígitos**



Fuente: elaboración propia.

Por seguridad, como lo muestra la figura 31, al momento de ingresar un dígito, este solo se podrá ver por 2 segundos y posteriormente se rellenará con un asterisco. Esto es para evitar que al momento que se ingrese la contraseña el niño pueda verla fácilmente; al momento de haber ingresado los 5 dígitos, el dispositivo se verá como lo muestra la figura 32.

Figura 32. **Ejemplo de 5 dígitos de contraseña ingresada**



Fuente: elaboración propia.

El dispositivo solicitará ingresar la contraseña nuevamente para confirmar que no se cometió ningún error la primera vez que fue ingresada, como lo muestra la figura 33.

Figura 33. **Confirmar contraseña**

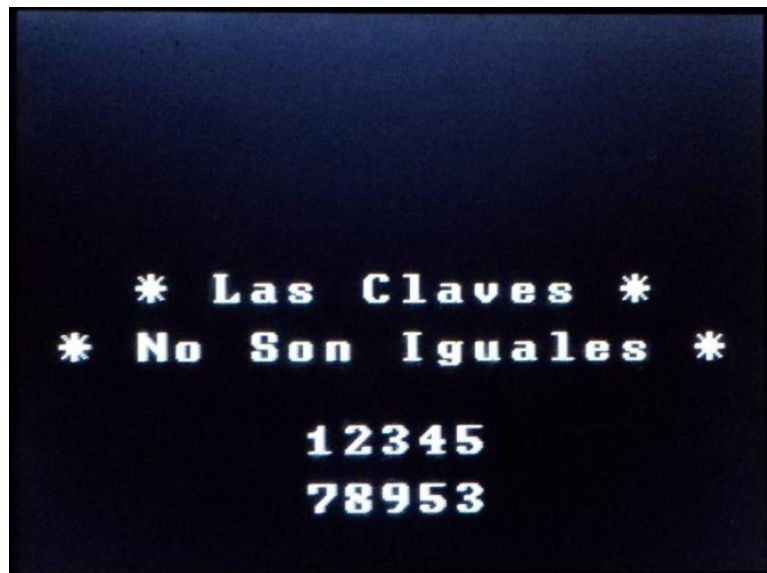


Fuete: elaboración propia.



Si las dos contraseñas no son iguales, el dispositivo solicitará que ingrese nuevamente la contraseña y mostrará las dos contraseñas para que se pueda verificar en donde se cometió el error, como se puede observar en la figura 34.

Figura 34. **Contraseñas no son iguales**

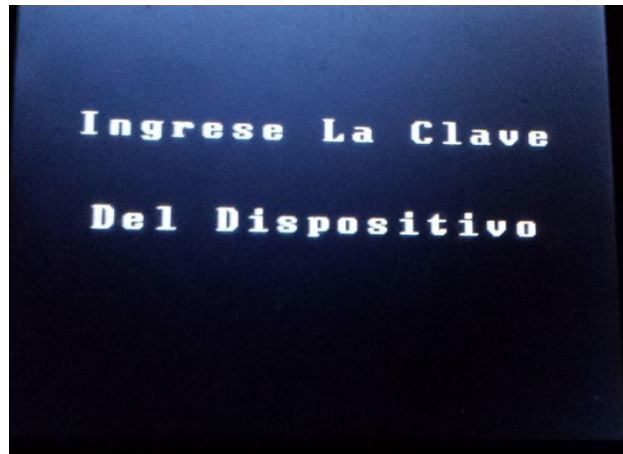


Fuente: elaboración propia.

La contraseña es guardada momentáneamente en una variable para luego ser guardada en la memoria EEPROM del Arduino Mega, esto es debido a que si no se guarda en la memoria EEPROM, la contraseña solo quedará guardada en la memoria RAM del Arduino mega y al momento de apagar el dispositivo.

Al momento de manipular el dispositivo, presionando el botón ingresar, este solicitará la contraseña guardada, como lo muestra la figura 35.

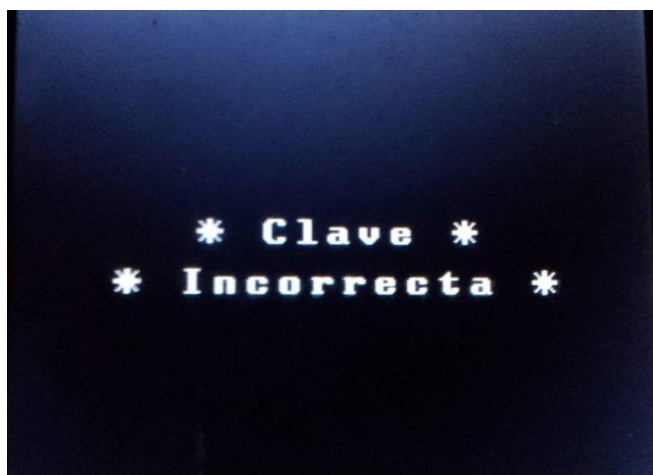
Figura 35. **Solicitud de contraseña del dispositivo**



Fuente: elaboración propia.

Se cuenta con un máximo de 3 intentos, por cada intento fallido, como lo muestra la figura 36, el dispositivo mostrará un mensaje de clave incorrecta.

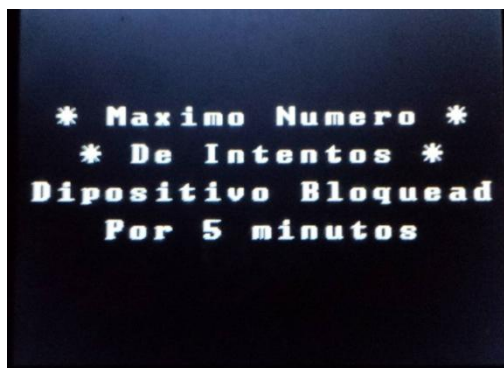
Figura 36. **Clave incorrecta**



Fuente: elaboración propia.

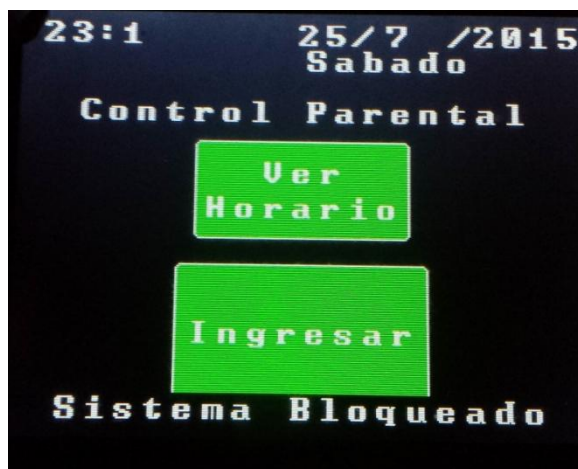
Una vez se alcancen los 3 intentos, el dispositivo mostrará un mensaje como el de la figura 37. Se podrá intentar de nuevo ingresar la clave cuando hayan transcurridos 5 minutos; en el menú de inicio del dispositivo se mostrará el mensaje de sistema bloqueado, como se observa en la figura 38.

Figura 37. **Máximo número de intentos para ingresar al dispositivo**



Fuente: elaboración propia.

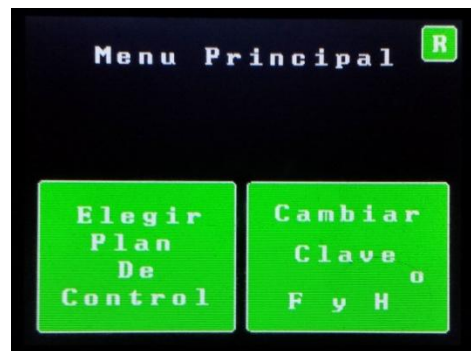
Figura 38. **Sistema Bloqueado**



Fuente: elaboración propia.

Si se desea modificar la contraseña, es necesario dirigirse al botón en el menú principal Cambiar clave o F y H como lo muestra la figura 39.

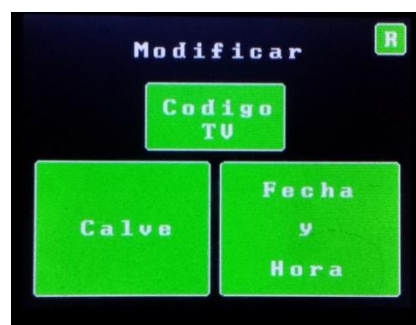
Figura 39. **Cambiar clave o fecha y hora**



Fuente: elaboración propia.

Una vez que se haya ingresado, se selecciona el botón Clave, como lo muestra la figura 39. A continuación se modifica la contraseña. La palabra Clave fue utilizada en lugar de Contraseña, debido a la cantidad de caracteres que ocupa en la pantalla LCD.

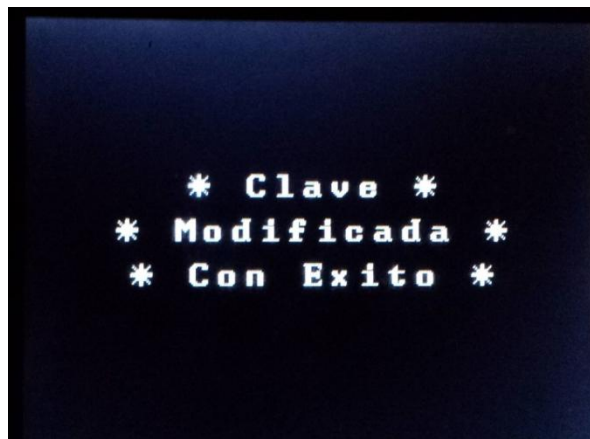
Figura 40. **Modificar clave**



Fuente: elaboración propia.

El dispositivo seguirá los mismos pasos mostrados entre las figuras 30 y 34, cuando se haya finalizado de modificar la clave, se mostrará el mensaje como lo muestra la figura 41.

Figura 41. **Clave modificada con éxito**



Fuente: elaboración propia.

Para el almacenamiento de la contraseña del dispositivo se utilizó la librería <EEPROM.h>. Para definir la librería se utiliza la sentencia #include <EEPROM.h>.

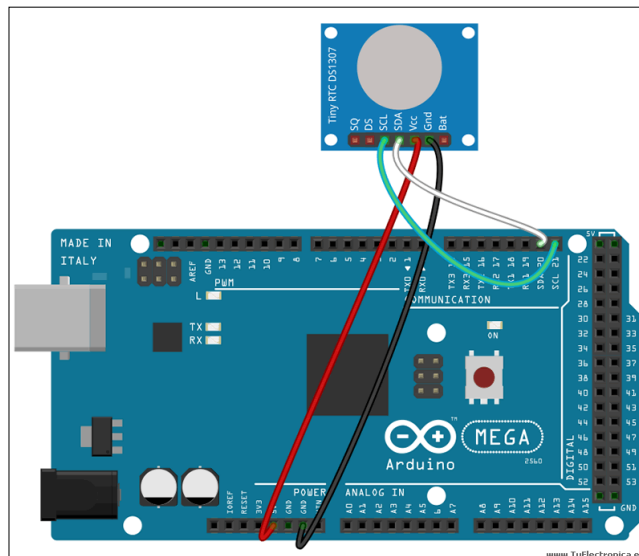
Para escribir o leer una posición específica en la memoria EEPROM del Arduino mega se utilizan las siguientes sentencias: EEPROM.write(Posición en memoria, valor a ser guardado en la posición), y Variable =EEPROM.read(posición en memoria).

#### 4.2.5. Control de tiempo y fecha

El reloj en tiempo real (RTC por sus siglas en inglés), modelo DS1307, es el encargado de enviarle al controlador principal la hora y fecha actual; el reloj es alimentado por una batería de 3V con el fin de que al momento que el Arduino mega deje de alimentar el reloj, este no pierda el conteo de la hora y fecha actual y al enviarlo el dispositivo vuelva a alimentar al reloj, la hora y fecha que envíe sea precisa.

Para la comunicación entre el reloj y el controlador principal se utilizan los puertos SCL (*serial clock line*) y SDA (*serial data line*), por sus siglas en inglés, SCL es para el reloj de sincronización, para el envío de los datos del puerto SDA; por cada ciclo de reloj, el SDA envía datos de forma serial.

Figura 42. Diagrama de conexión del módulo RTC con el Arduino Mega



Fuente: Módulo RTC DS1307 con Arduino. <http://www.tuelectronica.es/tutoriales/Arduino/reloj-rtc-i2c-con-Arduino.html>. Consulta: 1 de abril de 2015.

La figura 42 muestra el diagrama de conexión del módulo RTC a la placa Arduino Mega 2560. La alimentación del módulo debe ser de 5V.

Para la programación, el controlador principal (Arduino mega 2560) utiliza la librería RTCLib la cual puede ser descargada de la página oficial de Arduino.

Para poder iniciar a utilizar sentencias de la librería RTCLib en el software de programación para el Arduino es necesario colocar al inicio del programa la sentencia `#include RTCLib.h`. Después de esto es necesario colocar `RTC_DS1307 rtc;` para describir el nombre del integrado del módulo RTC.

Las primeras sentencias necesarias para iniciar a utilizar el reloj RTC con comentarios que la describen a cada una de ellas después de “//”, son las siguientes:

```
rtc.begin(); //inicializa la comunicación con el reloj RTC
if (! rtc.isrunning()) { //Condicional, si el reloj RTC está trabajando se cumple.
  rtc.adjust(DateTime(__DATE__, __TIME__)); // Al ingrearle el programa al
  Arduino mega, esta sentencia toma la hora y fecha de la PC y la sincroniza con
  el reloj RTC.
}
```

Las sentencias para obtener la hora, minutos, segundos, día, mes y año del reloj RTC son las que se muestran en la tabla VII.

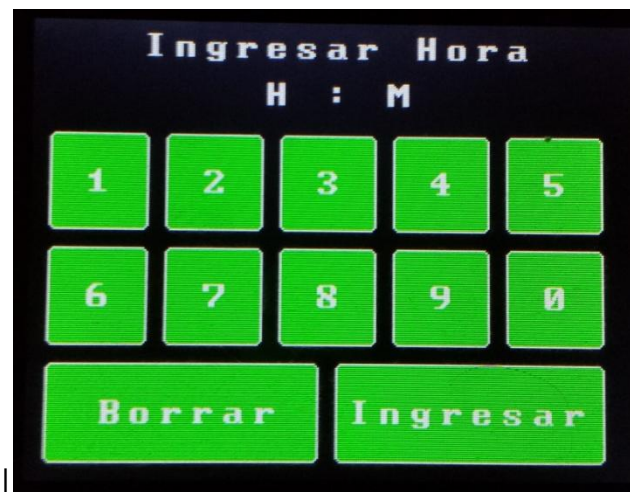
**Tabla VIII. Sentencias para obtener hora y fecha del reloj RTC**

Núm.	Sentencia	Descripción
1	<code>DateTime now= rtc.now();</code>	Para iniciar a solicitar datos al reloj RTC, es necesario colocar esta sentencia al inicio.
2	<code>Variable = now.year();</code>	Obtener el año actual y guardarlo en una variable.
3	<code>Variable = now.month();</code>	Obtener el mes actual y guardarlo en una variable.
4	<code>Variable = now.day();</code>	Obtener el día actual y guardarlo en una variable.
5	<code>Variable = now.hour();</code>	Obtener la hora actual y guardarla en una variable.
6	<code>Variable = now.minute();</code>	Obtener los minutos actuales y guardarlos en una variable.
7	<code>Variable = now.second();</code>	Obtener los segundos actuales y guardarlos en una variable.

Fuente: elaboración propia.

En el dispositivo, la primera vez que sea utilizado, este solicitará que se ingrese la hora actual que se utilizará en el dispositivo, como lo muestran las figuras 43 y 44.

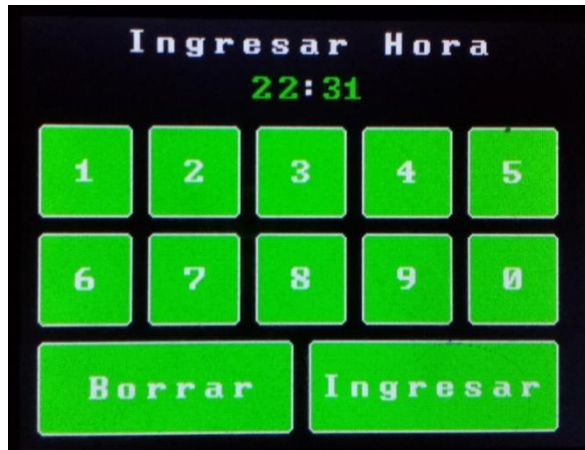
**Figura 43. Ingresar hora del dispositivo**



Fuente: elaboración propia.



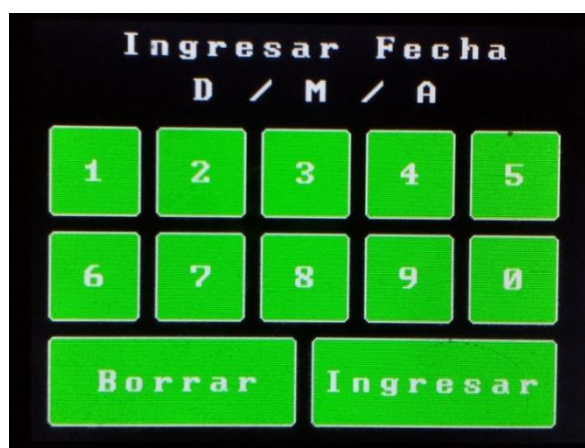
Figura 44. **Ejemplo de hora ingresada del dispositivo**



Fuente: elaboración propia.

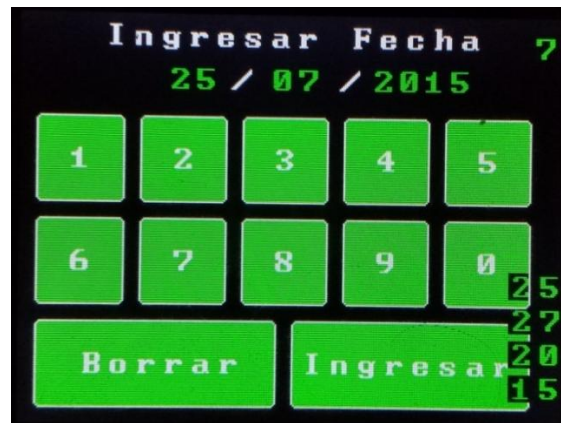
Una vez ingresada la hora, el dispositivo solicitará ingresar la fecha actual, como lo muestra las figuras 45 y 46.

Figura 45. **Ingresar fecha del dispositivo**



Fuente: elaboración propia.

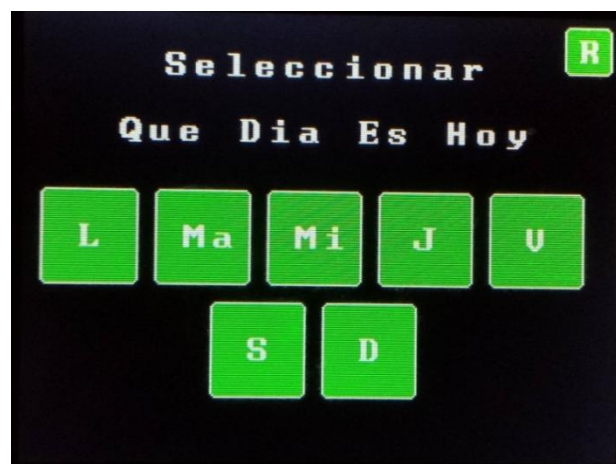
Figura 46. **Ejemplo de fecha ingresada al sistema**



Fuente: elaboración propia.

Por último, el dispositivo solicitará que se seleccione el día de la semana actual, como lo muestra la figura 47, para que el dispositivo lo tome como referencia y saber qué día de la semana se encuentra el dispositivo.

Figura 47. **Seleccionar día actual de la semana**

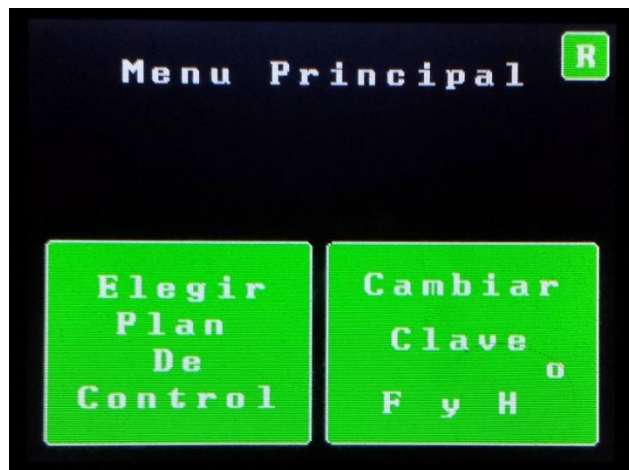


Fuente: elaboración propia.

#### 4.2.5.1. Control Parental

El usuario tiene la opción de seleccionar dos tipos de plan de control: semanal o diario; para lo cual primero es necesario dirigirse al menú principal, como lo muestra la figura 48; en el menú principal se selecciona la opción Elegir plan de control.

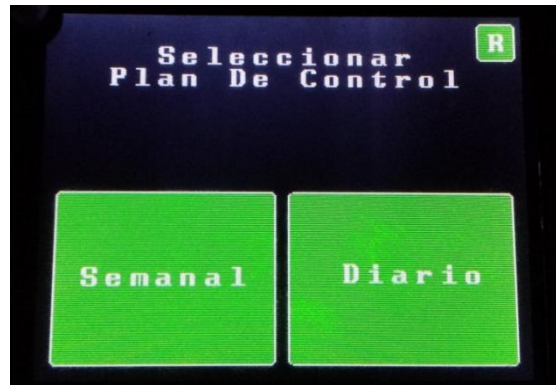
Figura 48. Elegir plan de control



Fuente: elaboración propia.

Una vez presionado el botón, mostrará el menú de plan de control, como lo muestra la figura 49, con las opciones semanal o diario.

Figura 49. **Menú plan de control semanal o diario**



Fuente: elaboración propia.

La opción de plan semanal ingresará el mismo horario para los 7 días de la semana. Primero, solicitará que se ingrese el horario inicial como lo muestra la figura 50, en que el dispositivo encenderá la TV con el código IR de encendido y habilitará la alimentación de la consola.

Figura 50. **Hora inicial de juego**



Fuente: elaboración propia.

Una vez ingresada la hora inicial, el dispositivo solicitará que se ingrese la hora final de juego, como se observa en la figura 51; una vez se alcance un minuto antes del horario final, 30 segundos antes el buzzer estará alertando que el tiempo límite de juego está por finalizar; las oscilaciones del buzzer aumentarán conforme el tiempo se esté agotando.

Figura 51. **Hora final de juego**



Fuente: elaboración propia.

Una vez ingresadas las horas de inicio y final, el dispositivo mostrará cómo fue configurado el horario para la semana completa, como se observa en la figura 52. Este horario, a menos que el usuario lo modifique, será aplicado para todas las semanas.

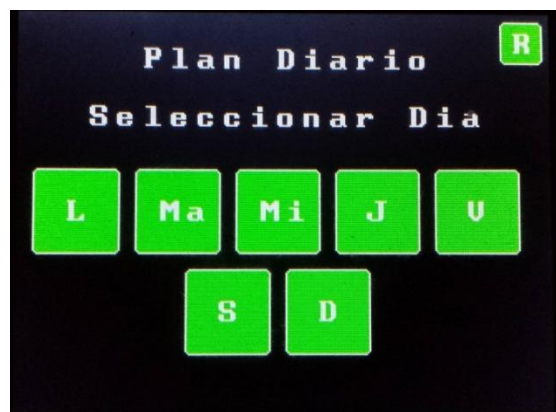
Figura 52. Horario semanal

Horia		
	I	F
Lunes	12:34	14:32
Martes	12:35	14:55
Miercoles	12:35	14:55
Jueves	12:35	14:55
Viernes	12:35	14:55
Sabado	12:35	14:55
Dimingo	12:35	14:55

Fuente: elaboración propia.

En el caso del plan de control diario, el dispositivo solicitará que se seleccione el día de la semana cuando se desea ingresar el horario, como se muestra en la figura 53.

Figura 53. Seleccionar día de plan de control diario



Fuente: elaboración propia.

Seleccionado el día, el proceso es el mismo que se siguió en las figuras 50 y 51. Cuando se ingresa el horario final, el dispositivo muestra el horario semanal completo en donde se podrá observar el día que se ingresó el horario de plan de control diario, como lo muestra la figura 52.





## **CONCLUSIONES**

1. El método de almacenamiento utilizado para guardar la información de horarios de juegos cuando los niños tienen permitido utilizar la consola de videojuegos, es la memoria EEPROM del Arduino Mega.
2. El dispositivo apaga la TV de forma segura con LED infrarrojos y, posteriormente, corta la alimentación de la TV que evita dañarla.
3. El dispositivo cuenta con una alarma audible con el fin de proporcionar al usuario un indicador para alertar que el tiempo de juego está por terminar; conforme el tiempo está por terminar la frecuencia del tono aumenta.



## RECOMENDACIONES

- 1 El dispositivo controla la alimentación de la TV y la consola de videojuegos por medio de dos cables por los cuales se envía los pulsos de activación o desactivación; es recomendable enviar los pulsos de forma inalámbrica utilizando módulos RF.
- 2 Agregar al dispositivo un módulo Ethernet con el fin de permitir al usuario poder configurar el dispositivo, por una aplicación móvil o una página *web*, desde cualquier lugar que cuente con una conexión a Internet. Le permite al usuario poder visualizar el estado del dispositivo, si la TV y la consola están disponibles para ser utilizadas.
- 3 Mejorar el aspecto de los íconos utilizados para los botones en la interfaz gráfica del dispositivo.



## BIBLIOGRAFÍA

1. *Arduino comparación de productos*. [En línea]. <<https://www.arduino.cc/en/Products/Compare>>. [Consulta: 21 de mayo de 2014].
2. *Arduino Mega*. [En línea]. <<https://www.arduino.cc/en/Main/ArduinoBoardMega>>. [Consulta: 21 de mayo de 2014].
3. *Contraseña*. [En línea]. <<https://es.wikipedia.org/wiki/Contrase%C3%B1a>>. [Consulta: 22 de julio de 2014].
4. *LCD 16X2*. [En línea]. <<https://www.scribd.com/doc/44252680/LCD-16X2>>. [Consulta: 29 de enero de 2015].
5. *Library DS1307 real time clock*. [En línea]. <<http://forum.arduino.cc/index.php/topic,37821.0.html>>. [Consulta: 19 de julio de 2014].
6. *Library UTFT*. [En línea]. <<http://www.rinkydinkelectronics.com/library.php?id=52>>. [Consulta: 20 de julio de 2014].
7. *Memoria EPROM*. [En línea]. <[https://es.wikipedia.org/wiki/Memoria\\_EPROM](https://es.wikipedia.org/wiki/Memoria_EPROM)>. [Consulta: 15 de agosto de 2014].
8. *Pantalla TFT LCD Arduino*. [En línea]. <[http://www.openhardware.pe/wpcontent/uploads/2013/03/3inch2\\_Arduino\\_LCD\\_Shield\\_UG.pdf](http://www.openhardware.pe/wpcontent/uploads/2013/03/3inch2_Arduino_LCD_Shield_UG.pdf)>. [Consulta: 19 de julio de 2014].

9. *Referencia lenguaje de programación Arduino*. [En línea]. <<https://www.arduino.cc/en/Reference/HomePage>>. [Consulta: 20 de octubre de 2014].
10. *Reloj RTC I2C con Arduino*. [En línea]. <<http://forum.arduino.cc/index.php?topic=177297.0>>. [Consulta: 8 de agosto de 2014].
11. *Videojuegos*. [En línea]. <[http://www.profeco.gob.mx/encuesta/brujula/bruj\\_2010/bol170\\_videojuegos.asp](http://www.profeco.gob.mx/encuesta/brujula/bruj_2010/bol170_videojuegos.asp)>. [Consulta: 19 de mayo de 2014].