



FACULTAD DE INGENIERÍA Y CIENCIAS AGROPECUARIAS

DISEÑO DEL PROTOTIPO DE UN SISTEMA DE ADQUISICION DE DATOS
PARA EL MONITOREO Y CONTROL DE EXPERIMENTOS QUIMICOS,
MEDIANTE EL USO DE SENSORES Y MICROCONTROLADORES.

Trabajo de Titulación presentado en conformidad con los requisitos
establecidos para optar por el título de Ingeniero en Electrónica y Redes de
Comunicación

Profesor Guía
MSC. Edgar Fernando Solís Acosta

Autor
Añazco Villarreal Luis Alejandro

Año
2015

DECLARACIÓN DEL PROFESOR GUÍA

“Declaro haber dirigido este trabajo a través de reuniones periódicas con el estudiante, orientando sus conocimientos y competencias para un eficiente desarrollo del tema escogido y dando cumplimiento a todas las disposiciones vigentes que regulan los Trabajos de Titulación.”

MSC. Edgar Fernando Solís Acosta
Máster en Evaluación y Auditoría de Sistemas Tecnológicos.
CI: 1803005071

DECLARACIÓN DE AUTORÍA DEL ESTUDIANTE

“Declaro que este trabajo es original, de mi autoría, que se han citado las fuentes correspondientes y que en su ejecución se respetaron las disposiciones legales que protegen los derechos de autor vigentes.”

Luis Alejandro Añezco Villarreal

CI: 2100133426

AGRADECIMIENTOS

Primero a mi abuelito el mejor; a mi madre y padre quienes me hicieron el hombre que soy.

A mi hermano de quien no puedo estar más orgulloso. A mis mejores amigos Nike, Nico, Daya, Pablito, Dimi, Barby y Lis, más que solo compañeros. Al Ing. Solís, quien me guio durante varios semestres en clases y ahora durante este trabajo de titulación, así mismo a mis profesores en especial a Angelito Jaramillo, Sebastián Gómez y Hugo Freire.

DEDICATORIA

A mi hermano, no puedo estar más orgulloso de ti nano y cada día quiero seguir mejorando para que siempre tengas un modelo a seguir.

A mi abuelito quien me dio muchas primeras veces, le dedico esta primera publicación.

RESUMEN

El uso de sensores para monitorear el entorno que nos rodea no es algo nuevo, es una práctica que cada vez invade más cada aspecto de nuestras vidas, desde los complejos sistemas de prevención y estudio de fenómenos naturales usado por científicos de todas las ramas hasta los simples sensores de proximidad instalados para automatizar las luces en una vivienda; los sensores se encuentran cada vez más relacionados con nuestras actividades diarias. La implementación de este prototipo permitirá llevar esta tecnología de uso diario a las aulas escolares, en especial en los laboratorios de experimentación química y física. El trabajo de titulación propuesto será capaz de medir varias magnitudes físicas como la temperatura tanto ambiental como la de sustancias, la cantidad de partículas por millón de una molécula de gas como el CO₂ componente del humo común, las diferentes longitudes de ondas de los colores y el nivel del potencial de hidrogeno de diferentes mezclas.

ABSTRACT

Using sensors to monitor the environment around us is not new, it is a practice that have increasingly invaded every aspect of our lives, from the complex systems of prevention and study of natural phenomena used by scientists of all branches up to simple proximity sensors installed to automate the lights in a home; sensors are increasingly related to our daily activities. Implementing this prototype will allow us to carry this everyday technology to the classrooms, especially in the laboratories of chemistry and physics experimentation. The proposed graduate work will be able to measure various physical quantities as substances and environmental temperature, amount of parts per million of a gas molecule such as CO₂ a component of common smoke, the different wavelengths of the colors and the level hydrogen potential of different mixtures.

INDICE

CAPÍTULO I. INTRODUCCIÓN	1
1.1. Planteamiento del problema	1
1.2. Justificación	2
1.3. Antecedentes.....	3
1.4. Objetivos.....	4
CAPÍTULO II: DISEÑO DEL PROTOTIPO.....	5
2.1. Diseño conceptual	5
2.2. Selección de Equipos	6
2.2.1. Arduino.....	7
2.2.2. Sensor a prueba de agua.....	8
2.2.3. Sensor de Temperatura	8
2.2.4. Probeta de medición de pH.....	9
2.2.5. Circuito de lectura de pH.....	9
2.2.6. Sensor de Gas MQ-7	10
2.2.7. Sensor de Reconocimiento de Luz	10
2.3. Componentes	11
2.3.1. Arduino.....	11
2.3.1.1. Sistema de memoria	13
2.3.1.2. EEPROM Flash.....	13
2.3.1.3. EEPROM Dirección de Bit.....	13
2.3.1.4. SRAM.....	13
2.3.2. One Wire (1-Wire)	14
2.3.3. DS18B20.....	15
2.3.4. TMP36	15
2.3.5. Sensor de reconocimiento de color TCS230/TSC3200.....	17
2.3.6. Probeta de pH PinPoint.....	19
2.3.7. Circuito de medición de pH.	20
2.3.8. MQ7 4P Module	21

2.4.	Diagrama de bloques	21
2.5.	Diseño del Software	26
2.5.1.	Software de Desarrollo.....	28
2.5.1.1.	Arduino Development Environment (ADE).....	28
2.5.1.2.	Visual Studio	31
CAPÍTULO III: IMPLEMENTACION.....		32
3.1.	Diagramas de Cableado.....	32
3.2.	Ensamblaje del Prototipo.....	37
CAPÍTULO IV: Pruebas y Resultados		39
4.1.	Análisis de resultados.....	39
4.1.1.	Nombre del Proyecto	41
4.1.2.	Profesor a Cargo.....	41
4.1.3.	Materiales.....	42
4.1.4.	Botón Guardar.....	42
4.1.5.	Gráficos de temperatura.	42
4.1.6.	Cuadros de texto de valor numérico	42
4.1.7.	Cuadro de color.....	42
4.2.	Reporte de Costos.....	43
CONCLUSIONES Y RECOMENDACIONES		44
REFERENCIAS		46
ANEXOS		48

CAPÍTULO I. INTRODUCCIÓN

1.1. Planteamiento del problema

Al realizar experimentos químicos y físicos a nivel secundario e incluso a nivel superior en el Ecuador, no se cuenta con herramientas que faciliten tanto el aprendizaje como el éxito del experimento, en la mayoría de casos se realizan los experimentos en base a al conocimiento colectivo que se encuentra en libros, estos experimentos son copias de experimentos pasados de los cuales ya se tiene un resultado esperado pero qué pasa cuando el resultado no es el esperado, en el mejor de los casos el profesor a cargo tiene conocimiento de ese resultado y es capaz de dar una explicación heurística y los estudiantes quedan con un vacío tanto en la práctica como en la teoría. Por ello es importante tener un registro de las condiciones que pudieron alterar el resultado.

La automatización es un proceso inminente, y el cual permite no solo facilitar muchos aspectos de la vida diaria pero es una herramienta con la cual se puede controlar, medir y cambiar automáticamente el ambiente al rededor, el avance tecnológico es un proceso continuo que cada día busca solucionar incluso los problemas más simples por medio de dispositivos electrónicos y automáticos, desde la apertura de una puerta de garaje, pasando por el control climático y lumínico de habitaciones, la manufactura en masa, transporte público, operaciones militares hasta incluso viajes al espacio. A niveles científicos la experimentación tanto química como física esta ayudada de diferentes herramientas que van desde simples recipientes hasta elaborados sistemas electrónicos y computacionales que permiten monitorear cada aspecto de los experimentos, estimar resultados basados en estadísticas, controlar con extremada exactitud cantidades, tiempos y proporciones.

La tecnología invade cada aspecto de nuestras vidas, ahora un dispositivo electrónico es indispensable para los estudiantes, el acceso a las redes de

información mundiales facilitan las investigaciones, el manejo de software de texto y hojas de cálculos hace que los trabajos sean cada vez más fáciles y completos. Todo en cuanto a la vida estudiantil se puede mejorar por medio de la tecnología, pero aún existen áreas que no han sido explotadas a la totalidad.

La experimentación a nivel escolar debe ser incentivada en los estudiantes, y para ello se debe contar no solo con los elementos necesarios para realizar el experimento, si no que con las herramientas para medirlo, para definirlo y para observarlo; de esta manera los estudiantes podrán realizar la parte más importante de un experimento, sacar conclusiones basadas en datos reales medidos a través de los sensores.

1.2. Justificación

Al realizar experimentos químicos, se debe considerar variables por encima de lo que se vierte en los matraces, se debe considerar las variables ambientales en las que se desarrolla el experimento, que depende de lo que se realice puede cambiar el resultado del mismo. Un monitoreo eficaz de estas variables, permitiría explicar con una base científica que el resultado no sea el esperado; y en una era en que la tecnología forma parte integral en nuestras vidas, se debe buscar que esta no sea solo para ocio, sino que inunde otros aspectos como lo es el ámbito escolar; tener métodos automatizados en esta área facilitaría en gran medida la tarea del profesor y alentaría a los estudiantes a mostrar más interés, ya que se encuentran en un entorno más amigable y conocido para las nuevas generación, un entorno tecnológico.

La automatización de un proceso estudiantil, no solo permitirá un avance acelerado en la educación a nivel secundario, sino que brindará a los estudiantes herramientas tecnológicas que los ayudará en su entendimiento, elevará sus habilidades tecnológicas y los familiarizara con el entorno que les espera a nivel profesional, la implementación de herramientas como esta desde un nivel secundario aumentará en gran medida las capacidades cognitivas del

estudiante, al tener acceso a tecnologías nuevas se despertará interés y aumentará la atención prestada a las clases recibidas.

1.3. Antecedentes

En la actualidad, se busca automatizar el mayor número de aspectos de nuestras vidas y gracias a los avances tecnológicos de los últimos años, esto se está logrando; con un mundo de información al alcance de todo, se puede aplicar el conocimiento desarrollado en todo el mundo para crear nuevas herramientas que faciliten el desempeño en la vida diaria.

La tecnología invade cada aspecto de la vida humana, ahora un dispositivo electrónico es indispensable para los estudiantes, el acceso a las redes de información mundiales facilitan las investigaciones, el manejo de software de texto y hojas de cálculos hace que los trabajos sean cada vez más fáciles y completos. Todo en cuanto a la vida estudiantil se puede mejorar por medio de la tecnología, pero aún existen áreas que no han sido explotadas a la totalidad.

Los sensores han tenido un gran despliegue y desarrollo, inicialmente a la investigación militar por parte de DARPA, hoy en día sus usos civiles comprende una variedad de campos, desde seguridad en empresas y hogares, investigación científica, prevención de desastres naturales e incluso trivialidades como ambientación del hogar. Sin lugar a dudas es una tecnología en auge y con un sin número de aplicaciones ya que es la base para sistemas mucho más complejos.

La experimentación es una parte fundamental del proceso de aprendizaje, y si bien los experimentos que se realizan en ámbitos escolares, son controlados y de resultados predecibles, existen variables ambientales que de ser distintas afectarían el resultado final del experimento. Las mediciones de los resultados se las realizan de forma empírica en ambientes escolares; lo cual no da resultados exactos y complica la tarea de enseñanza de los profesores.

La automatización como parte fundamental del desarrollo de las tecnologías más cercanas al hombre, es sin duda un foco de desarrollo, cada día vemos nuevos inventos que facilitan la vida de los seres humanos; y el uso de sensores como medios de obtención de los datos de una manera automática hace posible, por ejemplo, un ambiente controlado cien por ciento por un sistema computarizado y totalmente automatizado, dando lugar a que el usuario solo deba poner sus preferencias y el resto lo hace el sistema computacional.

1.4. Objetivos

Ensamblar una herramienta de recolección y procesamiento de datos, didáctica y de fácil uso para los estudiantes en el proceso de experimentación química y física.

Diseñar un prototipo para la recolección de datos de forma automática de variables ambientales comunes en la experimentación científica a nivel escolar.

Automatizar el procesamiento de información recolectada para un análisis didáctico y sistemático.

Programar el software que procese información recolectada y la presente de una forma didáctica, con la cual mejorar el entendimiento de los estudiantes.

Aplicar las recomendaciones, buenas prácticas y estándares de diseño, programación.

CAPÍTULO II: DISEÑO DEL PROTOTIPO

2.1. Diseño conceptual

El prototipo del sistema de recolección y procesamiento de datos, será capaz de recolectar datos de temperatura ambiental continuamente, pH, temperatura en la mezcla y color de la mezcla. Los sensores de cada dato relevante estarán conectados a un microcontrolador utilizando la plataforma Arduino. Dentro de la cual se programara la secuencia en la que deben ser leídos los valores y la transformación del valor leído por el sensor, por lo general en voltios, a digital y posteriormente a lenguaje humano para su fácil comprensión en números enteros.

El microcontrolador transmite por medio de un cable serial al computador los datos serán recogidos por los sensores y procesados por el microcontrolador; posteriormente serán enviados a la computadora a la que se encuentra conectado a la plataforma Arduino. En la computadora correrá el software de visualización de datos, al estar conectados directamente y ser información de poca importancia en términos de seguridad informática no se prevé encriptación. La información de acceso será grabada en el microprocesador y en el software de visualización; se escogió la transmisión vía cable serial sobre la transmisión vía WiFi, en primer lugar por su facilidad de conexión ya que la plataforma Arduino tiene incorporado un puerto serial, en segundo lugar para permitir una configuración sencilla en la terminal sin quitarle el acceso a internet y tercero por que la cantidad y demanda de datos así como la distancia entre el microcontrolador y la computadora del usuario no justifican una transmisión inalámbrica.

El software de visualización crea un proyecto nuevo, en el cual se debe insertar datos sobre el experimento a realizar, luego abre la conexión con el microcontrolador que envía constantemente datos; una vez que el software de visualización en la computadora del usuario recibe los datos los guarda

automáticamente en la memoria del software y los muestra en la pantalla de la terminal en datos numéricos para cada dato medido, los cuales se actualizan constantemente al llegar nuevos datos. De ser requerido el software puede mostrar los datos en un gráfico de dos dimensiones en función de tiempo para cada variable medida.

Antes de cerrar el experimento se podrá exportar los datos adquiridos a un archivo .CSV que es compatible con Microsoft Excel; cuando se cierra el experimento, se cierran las conexiones abiertas.

2.2. Selección de Equipos

Los equipos fueron seleccionados en base a su facilidad de adquisición, uso e instalación. A continuación detallaremos los componentes a ser utilizados.

Componentes:

- Plataforma Arduino UNO R3
- Sensor de Temperatura TMP36
- Sensor de Temperatura a prueba de agua DS18B20
- Sensor de Luz TCS34725FN
- Probeta pH PinPoint
- Circuito para Arduino para pH
- Alarma de humo
- Protoboard
- Cables de cobre varios
- Resistencias varias
- LEDs Varios

Librerías Arduino Extra

- 1-Wire
- DallasTemperature

Software

- Arduino IDE
- Visual Studio Express
- Fritzing

Herramientas

- Peladora de cables
- Multímetro
- Clips
- Lupa
- Iron Pen

2.2.1. Arduino

En este caso se escogió el modelo UNO R3, por motivo de su disponibilidad y amplia bibliografía, ya que es el modelo preferido de muchos autores al momento de usar en sus proyectos electrónicos tiene una mayor facilidad de uso. Existen varios modelos dentro de la familia de plataformas Arduino que nos permitirían realizar este proyecto de titulación con facilidad, pero ya que la principal diferencia entre ellos varía en las dimensiones de sus características, teniendo diferentes procesadores y tamaños de memoria, detalles que se puede apreciar en la tabla 1 a continuación.

Tabla 1. Comparación entre modelos de Arduino.

Comparación de modelos de Arduino		
Características	Arduino UNO R3	Arduino Nano
Procesador	ATmega328	ATmega168
Voltaje de Operación / Voltaje de entrada	5 V / 7 – 12 V	5 V / 7 – 9 V
Velocidad de Procesamiento	16MHz	16MHz

Pines Análogos Entrada/Salida	6/0	8/0
Pines Digitales IO/PWM	14/6	14/6
EEPROM	1KB	1KB
SRAM	2KB	2KB
Flash	32KB	32KB
USB	Regular	No

Tomados de: Arduino, 2015.

2.2.2. Sensor a prueba de agua

En el mercado no existen muchas opciones al momento de elegir sensores a prueba de agua de temperatura, existen sin embargo los sensores DS18B20 de diferentes manufacturas, además el protocolo 1-Wire nos permite una fácil implementación y la posibilidad de crecimiento escalable.

2.2.3. Sensor de Temperatura

Como se puede observar en la tabla 2 a continuación, estos son dos de los más populares sensores para medir temperatura ambiental, sus características son muy similares, incluso se podría llegar a intercambiar ya que ambos tienen una salida de 250mV, una diferencia es que el LM35 necesita una resistencia externa para tener rango total de lectura con precisión.

Tabla 2. Comparación de Sensores de Temperatura

Comparación de Sensores de Temperatura		
Características	TMP36	LM35
Calibración	En Celsius	En Celsius
Escala	10mV/°C	10mV/°C
Rango de operación	-40°C a +125°C	-55°C a +135°C
Bajo auto calentamiento	SI	SI

Tomados de: Datasheet

2.2.4. Probeta de medición de pH

Tabla 3. Comparación entre probetas de medición de pH

Comparación entre probetas de medición de pH		
Características	American PinPoint	Atlas Scientific
Rango pH	0-14	0-14
Temperatura de Operación	1°C – 50°C	1°C – 99°C
Profundidad Máxima	10M	60M
Velocidad de respuesta	1 s	1s
Conector BNC	SI	SI

Tomados de: Atlas Scientific y American Marine Inc.

Las probetas de medición están disponibles en diferentes marcas en las cuales varían la calidad de los componentes y el precio. Las probetas están estandarizadas para poder ser usada con la mayoría de monitores de pH y puedan ser intercambiables fácilmente. Se escogió la probeta de Atlas Scientific.

2.2.5. Circuito de lectura de Ph

Tabla 4. Comparación entre circuitos de lectura de pH.

Comparación entre circuitos de pH		
Características	Ezo™ pH Circuit	5.0 pH Circuit
Rango	0.001 – 14.000	0.01 - 14.00
Lecturas independientes o dependientes de la temperatura	SI	SI
Lecturas únicas o continuas	SI	SI
Consumo de poder en modo de descanso	0.995mA a 3.3V	1.89mA a 3.3V
Voltaje de Operación	3.3V – 5V	3.3V – 5V

Tomados de: Atlas Scientific

Estos módulos integrados de lectura de potencial de hidrogeno reciben la información desde la probeta conectada a través de un conector BNC, ambos son de la misma marca pero su principal diferencia radica en la precisión de cada una y en la eficiencia energética. Se escogió el circuito EZO por su superioridad en precisión.

2.2.6. Sensor de Gas MQ-7

La principal diferencia entre estos sensores es que uno es un módulo integrado el cual ya nos entrega datos tanto analógicos como digitales, y el sensor solo necesita de circuitería extra para calibrar su salida y hacerla entendible para la plataforma Arduino. Se escogió el modulo ya que está listo para ser conectado a la plataforma Arduino.

Tabla 5. Comparación de Sensores de Gas

Comparación de Sensores de Gas		
Características	Módulo SainSmart MQ-7	Sensor MQ-7
Tipo de Detección	CO2	CO2
Rango de Detección	100-1000ppm	100-1000ppm
Datos Digitales	SI	NO
Datos Analógicos	Si	Si
Conexiones Extras	NO	Si

Tomados de: SainSmart

2.2.7. Sensor de Reconocimiento de Luz

Aunque el modelo TCS34725 es ligeramente superior es logísticamente más difícil de adquirir que su contraparte TCS230/TCS2300, sin embargo con el segundo se puede suplir las mismas funciones a un costo menor como se puede observar en la tabla 6 a continuación.

Tabla 6. Comparación de Sensores de Reconocimiento de Luz

Comparación de Sensores de Reconocimiento de Luz		
Características	TCS34725	TCS230/2300
Sensor RGB	SI	Si
Respuesta Rápida	Si	Si
Salida Analógica	Si	SI
Salida Digital	NO	SI
Sensor Luz Clara	SI	SI
Filtro IR	SI	SI

Tomados de: Datasheet

2.3. Componentes

2.3.1. Arduino

Es una plataforma electrónica que permite aprovechar al máximo un microprocesador de la manera más fácil y didáctica posible, este hardware es *open source*, lo que significa que nos he posible usarlo y alterarlo de muchas maneras sin afectar licencias siempre y cuando documentemos nuestro trabajo al igual que se puede encontrar una plétora de documentación.

Arduino también es conocido como “*physical computing*”, ya que se conecta a las computadoras por medio de un cable USB y brinda la posibilidad de utilizar Arduino como una tarjeta de interfaz entre la computadora y el mundo real, por medio de la cual se puede controlar diferentes elementos de nuestro entorno utilizando dispositivos electrónicos conectados a los diferentes pines del Arduino.

La plataforma Arduino que se usara en el presente trabajo de titulación es el Arduino UNO R3, la cual está equipada con un conector USB que permite la programar el procesador desde una computadora a través de un cable serial común y también se puede usar técnicas ISP (*In System Programming*) con un conector ISP de 6 pines.

El procesador central de la plataforma Arduino UNO R3 es microprocesador Atmel Atmega328. Es de 28 pines, microcontrolador de 8 bits. Utiliza la arquitectura RISC por sus siglas en inglés (*Reduced Instruction Set Computer*) Computador de Conjunto de Instrucciones Reducidas, que permite que el procesador pueda completar 20 millones de instrucciones por segundo (MIPS) cuando se opera a 20 MHz.

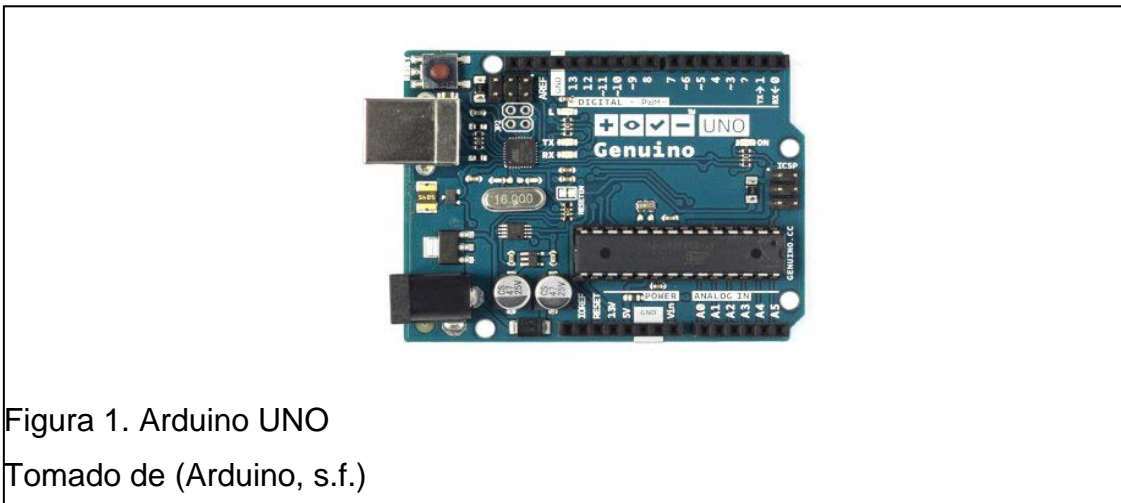


Figura 1. Arduino UNO
Tomado de (Arduino, s.f.)

Este microprocesador en particular cuenta con un número de características clasificadas en los siguientes sistemas:

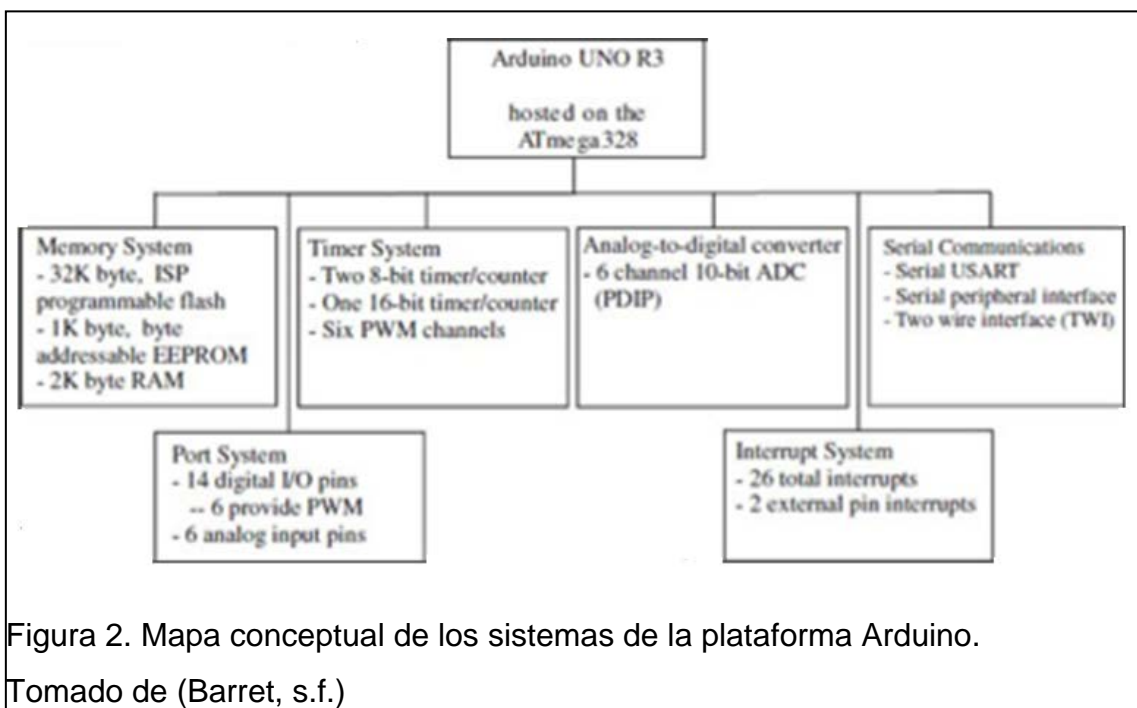


Figura 2. Mapa conceptual de los sistemas de la plataforma Arduino.
Tomado de (Barret, s.f.)

2.3.1.1. Sistema de memoria

El sistema de memoria de la plataforma Arduino UNO R3 tiene tres tipos de memorias EEPROM (*Erasable Programmable Read Only Memory*) de tipo flash, SRAM (*Static Random Access Memory*) y EEPROM de tipo dirección de bit para almacenamiento de datos. (Barret, 2013)

2.3.1.2. EEPROM Flash

Esta memoria es usada para guardar programas, es no volátil lo que significa que continúa guardando la información incluso cuando la plataforma Arduino no está en uso. Puede borrarse y volverse a programar lo que permite hacer cambios en el programa que se usa en Arduino, también es capaz de recibir constantes de programación, en caso de que su número sea muy grande, si se las programe como variables globales. (Barret, 2013)

2.3.1.3. EEPROM Dirección de Bit.

Este tipo de memoria es usada para guardar permanentemente variables durante su ejecución, así como para llamarlas, se usa en especial por ser localizable a nivel de byte en sistemas de logs de fallas durante la ejecución del programa, o almacenando datos que necesitan ser recordados en caso de perder energía eléctrica pero que necesitan ser cambiados periódicamente como combinaciones de candados electrónicos o secuencias de desbloqueo de puertas eléctricas. (Barret, 2013)

2.3.1.4. SRAM

Funciona igual que una memoria estática RAM, es volátil lo cual significa que su información se pierde cuando la plataforma pierde energía eléctrica, se la puede escribir o leer durante la ejecución del programa. Se usa para guardar

variables globales, ayudar en la ubicación dinámica de espacios de memoria para las variables de los programas. (Barret, 2013)

2.3.2. One Wire (1-Wire)

1-Wire™ es una tecnología desarrollada por Dallas Semiconductor™ y en el presente sigue funcionando a través de Maxim Integrated™, los sensores que usan este concepto son muy populares ya que son de fácil uso además de permitir una arquitectura maestro-esclavo, teniendo un sensor maestro y muchos sensores esclavos los cuales se alimentan de energía utilizando el modo parasito. La característica principal de estos sensores es la posibilidad de conectar varios de ellos a un solo pin de un microcontrolador y a la vez conseguir lecturas calibradas digitalmente. En este proyecto se usarán los sensores de temperatura los cuales vienen en una sonda lo que nos permite utilizarlos en líquidos o ambientes poco amigables. Ya que se puede conectar muchos sensores en un mismo pin del microcontrolador, se usa el método *get Temp CByIndex ()*; como solo se usara un sensor el parámetro en este caso particular es 0.

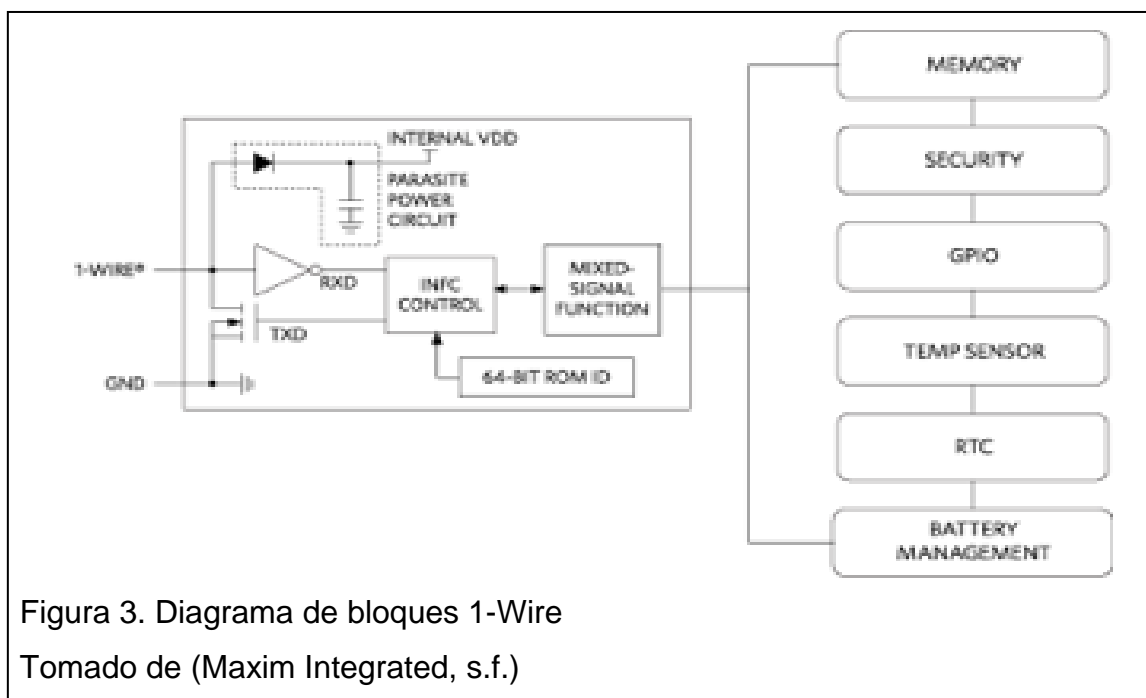


Figura 3. Diagrama de bloques 1-Wire

Tomado de (Maxim Integrated, s.f.)

2.3.3. DS18B20

Este sensor es el más el común de temperatura que usa la tecnología 1-wire, funciona de manera similar a un sensor de temperatura común, con la particularidad que sumergible, lo que permite medir la temperatura de soluciones; tiene un rango de -55°C hasta $+125^{\circ}\text{C}$ con una precisión de $\pm 0.5^{\circ}\text{C}$

La configuración de pines en este sensor en especial es simple, el cable de color negro como en la mayoría de casos es tierra, el cable rojo del sensor el pin de voltaje el cual debe conectarse con una resistencia de $4.7\text{K}\Omega$ y finalmente el cable amarillo es el pin de datos, los cuales son digitales y no necesitan transformación de unidades.



2.3.4. TMP36

Es un sensor analógico que permite lecturas precisas en grados centígrados, el voltaje resultado del sensor es linealmente proporcional a la temperatura en Celsius. La precisión típica de este sensor es de $\pm 1^{\circ}\text{C}$ a temperaturas hasta 25°C y en el rango de -40°C hasta $+125^{\circ}\text{C}$ con una precisión $\pm 2^{\circ}\text{C}$. La disposición de los pines es bastante estándar, el pin del medio es el de datos analógicos; los pines de los extremos son de Vcc y Tierra.

Sensor	Offset Voltage (V)	Output Voltage Scaling (mV/°C)	Output Voltage at 25°C (mV)
TMP35	0	10	250
TMP36	0.5	10	750
TMP37	0	20	500

Figura 6. Tabla de las características de salida de los sensores de la serie TMP3x

Tomado de (Circuitholic, s.f.)

2.3.5. Sensor de reconocimiento de color TCS230/TSC3200

El integrado TCS230/TCS3200 es un sensor de luz capaz de distinguir los tres colores principales del modelo RGB, cada color es medido por separado y da como resultado una onda cuadrada con una frecuencia directamente proporcional a la intensidad de la luz. La intensidad de cada color es recibida por el microcontrolador de manera invertida esto quiere decir que mientras más fuerte es la intensidad recibida, menor es el valor numérico que representa dicho color, para ello en el código del microcontrolador se lo debe invertir con la siguiente fórmula:

$$\text{colorValue} = 255 - \text{colorIntensity} \quad \text{(Ecuación 3)}$$

Donde *colorValue* es el valor numérico de cada color base en el sistema RGB, 255 es el máximo valor que puede tomar cada color; finalmente *colorIntensity* es el valor leído por el pin de entrada en el Arduino.

Este sensor tiene 8 pines, los cuales se pueden observar en la siguiente imagen, además de los 8 pines posee 4 LEDs de luz blanca que ayudan al fotodiodo en el centro detectar el tipo de color dependiendo de la configuración de los pines S2 y S3. Los pines S0 y S1 son para configurar la escala de la frecuencia de salida la cual es leída a través del pin *OUT* y el pin *OE* activa la

frecuencia de salida; finalmente tiene un pin para poder VCC y otro para tierra GND.

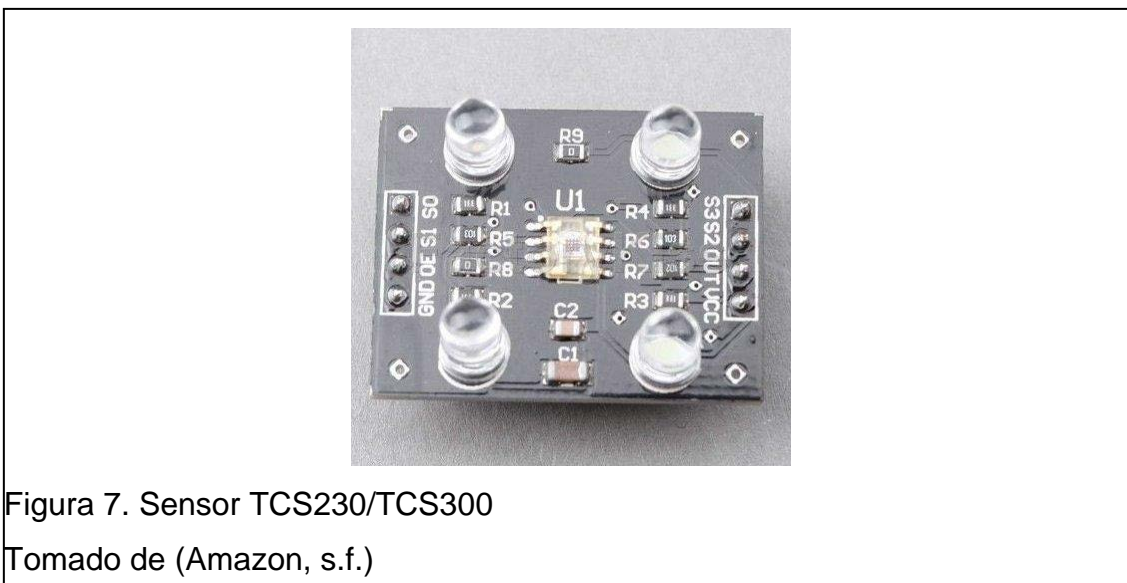


Figura 7. Sensor TCS230/TCS300

Tomado de (Amazon, s.f.)

Como se puede observar en la siguiente tabla, es necesario cambiar el voltaje de salida de los pines del microcontrolador que están conectados a los pines S2 y S3 del sensor; de esta manera se puede escoger la base del color RGB a medir, es decir se puede elegir entre la base roja, azul o verde. Para obtener datos suficientes para representar un color de manera digital es necesario medir al menos las tres bases del sistema RGB.

Tabla 7. Tipo de Lectura del fotodiodo.

TIPO DE LECTURA DEL FOTODIODO		
S2	S3	COLOR A LEER
LOW	LOW	ROJO
LOW	HIGH	AZUL
HIGH	HIGH	LUZ SIN FILTRO
HIGH	HIGH	VERDE

Con las configuraciones de los pines S0 y S1 podemos controlar la frecuencia de salida del sensor para su lectura apropiada por parte del procesador, la tabla 8 a continuación detalla la manera en que se puede usar estos pines.

Tabla 8. Escala de Frecuencia de Salida.

ESCALA DE FRECUENCIA DE SALIDA		
SO	S1	FRECUENCIA DE SALIDA
LOW	LOW	SIN PODER
LOW	HIGH	2%
HIGH	HIGH	20%
HIGH	HIGH	100%

2.3.6. Probeta de pH PinPoint

Es un dispositivo pasivo que contiene cloruro de plata para realizar la medición del potencial de hidrogeno, el cambio de resistencia eléctrica que se produce en el cloruro de plata al estar expuestos a diferentes potenciales de hidrogeno es medida a través de un circuito especializado para obtener un resultado entendible para los humanos. Esta probeta está fabricada en Rugged Epoxi, un polímero termoestable altamente resistente y aislante de electricidad, lo cual la hace virtualmente irrompible. Es de fácil instalación y reemplazo ya que se conecta por medio de un conector BNC al circuito que interpreta los datos medidos y es este circuito es el que le transmite un valor interpretable a la plataforma Arduino para que él los procese.



Figura 8. Probeta de pH Atlas Scientific.

Tomado de(Atlas Scientific, s.f.)

2.3.7. Circuito de medición de pH

Este circuito desarrollado por Atlas Scientific nos permite medir de manera precisa los potenciales de hidrógenos de las sustancias a través de una probeta. La comunicación del circuito de Atlas con el procesador Arduino es hecha utilizando 14 simples comandos de un carácter ASCII. Entre las características del circuito pH esta la medida total del rango de alcalinidad y acidez con una precisión de milésimas o dos unidades significativas, es decir desde 00.01 hasta 14.00. Nos permite tomar mediciones dependientes o independientes de la temperatura ya sean estas mediciones únicas o continuas.

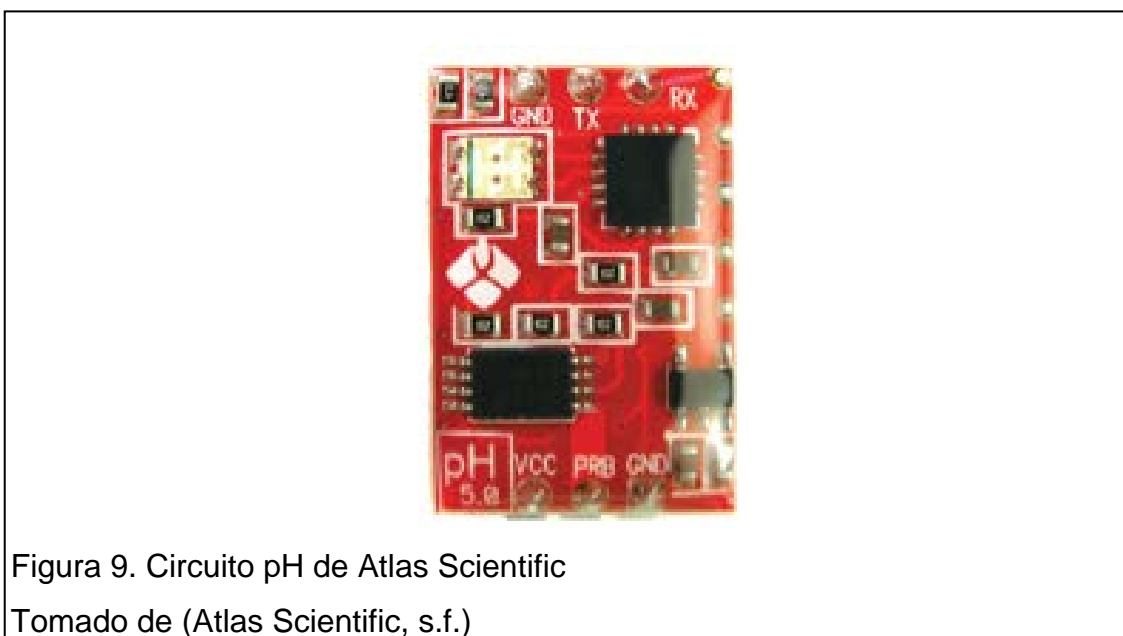


Figura 9. Circuito pH de Atlas Scientific

Tomado de (Atlas Scientific, s.f.)

Solo tiene 6 pines, de los cuales dos son para la probeta, uno para datos que se conecta al cable de datos del conector BNC y uno para la tierra del cable o el aislamiento identificados como PRB y GND el cual se encuentra a lado del PRB para evitar confusiones con la tierra del circuito; 2 pines más para transmisión y recepción de datos con la plataforma Arduino identificados como TX y RX respectivamente. Finalmente se tiene los pines para VCC y GND del sistema los cuales se conectan a sus respectivos pares en la plataforma Arduino. (Atlas Scientific, 2015)

2.3.8. MQ7 4P Module

Este sensor en específico viene con su circuito integrado para la facilidad de su uso, es de la familia de los sensores MQ pero diseñado para usarse con la plataforma Arduino, es un sensor analógico que detecta los cambios de conductividad en la superficie de dióxido de estaño en presencia del monóxido de carbono, presente en el humo. Es capaz de detectar desde 10 hasta 1000 ppm (partes por millón), con un tiempo de respuesta de 1 segundo nos permite tener una detección rápida de la presencia de humo.

Datasheet MQ7, 2015.



Figura 10. Módulo MQ-7 vista posterior

Tomado de (Amazon, s.f.)

2.4. Diagrama de bloques

Mediante un diagrama de bloques se puede apreciar claramente las conexiones que se piensa realizar en este prototipo gracias a una clasificación de colores en el sistema. En el centro del mismo tenemos al Arduino, el cual se conectará los diferentes sensores del prototipo.

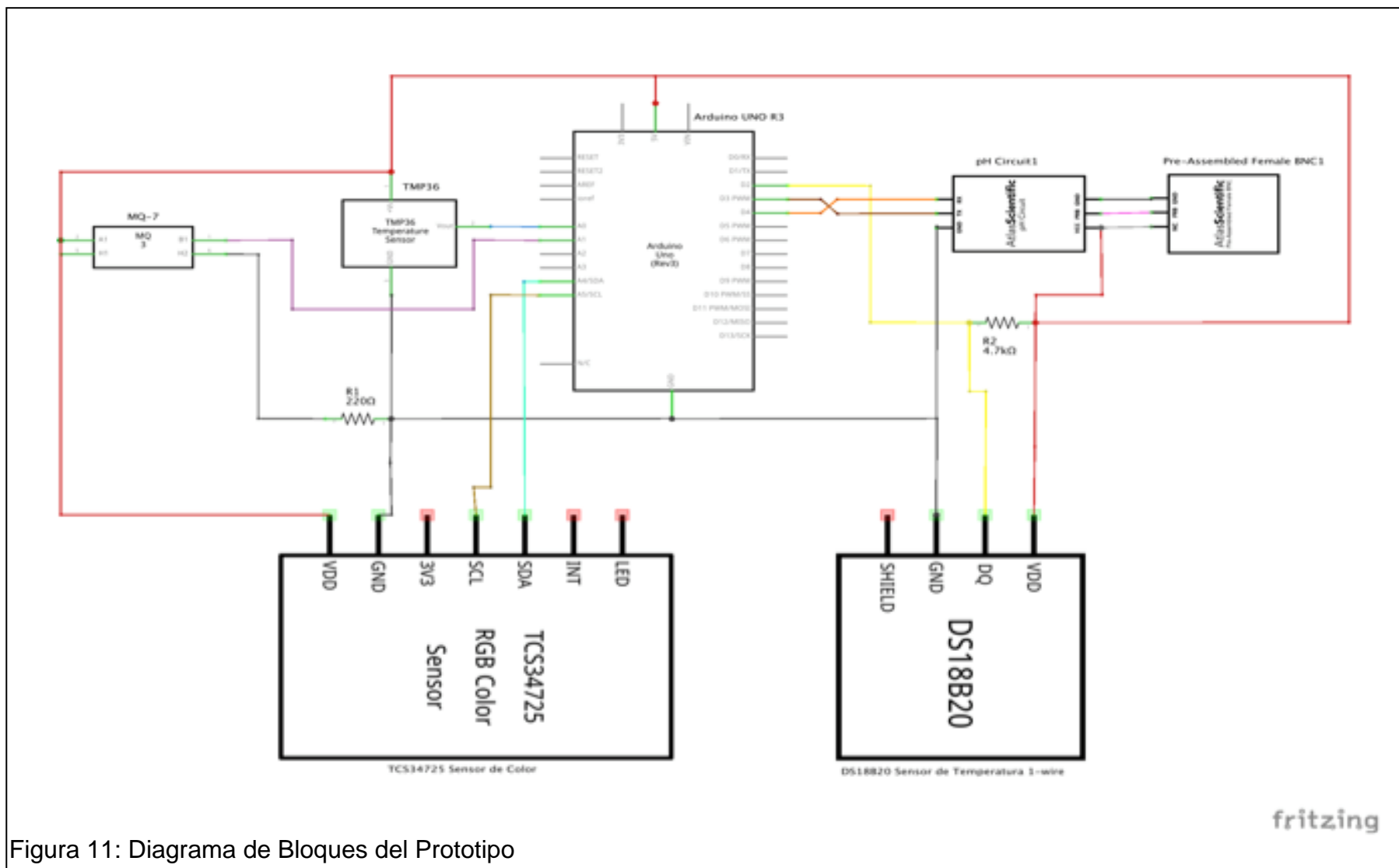


Figura 11: Diagrama de Bloques del Prototipo

La plataforma Arduino es el núcleo funcional del prototipo, desde la cual se recolecta la información del ambiente obtenida por los diferentes sensores. Como se puede apreciar en la figura 11, todos los sensores se conectan al micro procesador a través de diferentes pines tanto analógicos como digitales. Las diferentes conexiones están codificadas con diferentes colores:

- Rojo: 5V.
- Negro: Tierra.
- Azul: Datos Analógicos de Temperatura.
- Purpura: Datos Analógicos del Sensor de Gas.
- Cian: Datos Analógicos Sensor de Luz.
- Ocre: Datos Analógicos Sensor de Luz.
- Amarillo: Datos Digitales Sensor de Temperatura 1-Wire.
- Naranja: Datos Digitales Circuito de pH, Rx-Tx.
- Café: Datos Digitales Circuito de pH, Tx-Rx.

Todos los sensores trabajan constantemente recolectando información del entorno que los rodea y continuamente envían los datos obtenidos como señales analógicas o digitales, estas a su vez deben ser interpretadas por el procesador, algunos como el caso del sensor de temperatura se tiene una interpretación directa o con alguna proporcionalidad. En otros casos como los sensores de potencial de Hidrogeno y de gas CO₂, es necesario un módulo o un circuito integrado capaz de reconocer las señales recibidas por los sensores y transformarlas a datos que puedan ser leídos por el Arduino para que este a su vez los transmita a la computadora y usando el software de interfaz al usuario final en lenguaje humano para su fácil comprensión.

El proceso de medición es una secuencia repetitiva, que sigue su curso mientras el procesador tenga energía, es en términos informáticos un *loop* infinito. Como se muestra en la figura 12 a continuación.

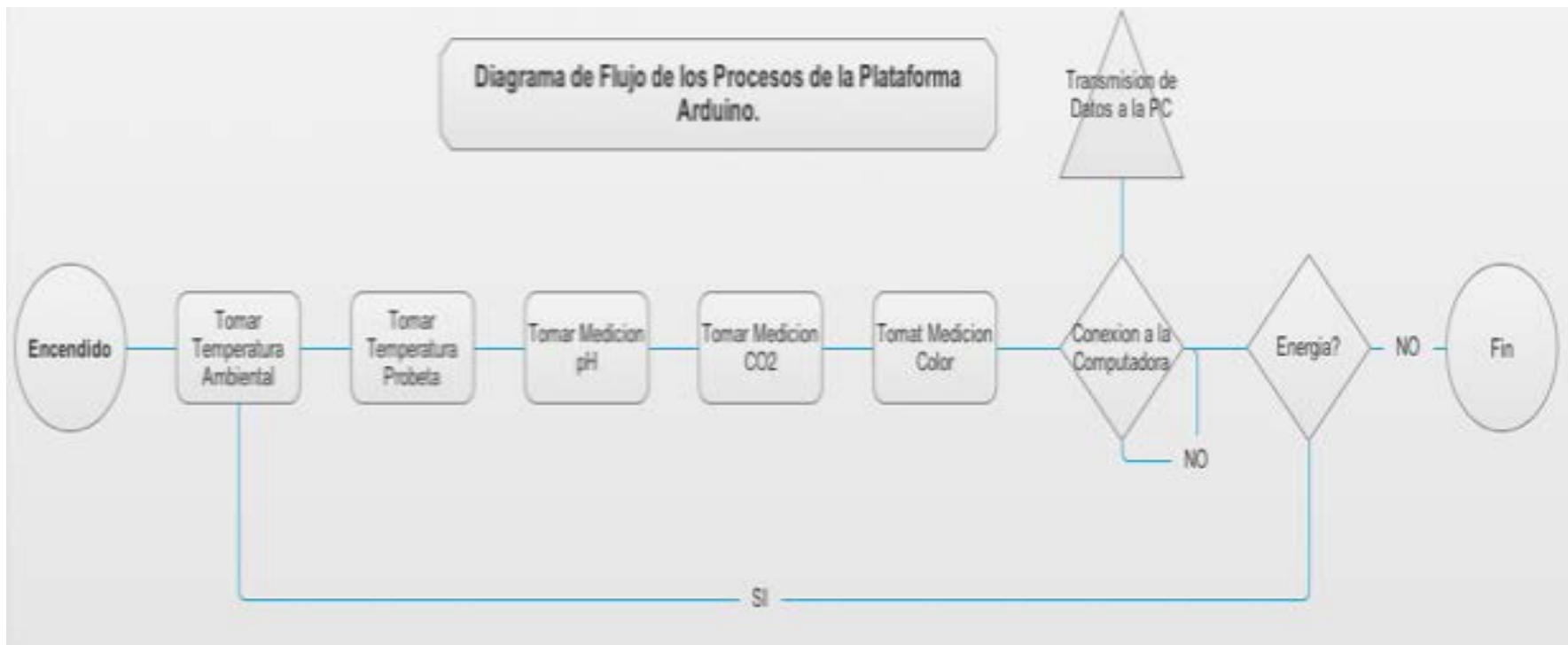


Figura 12: Diagrama de flujo de los procesos de la Plataforma Arduin

En la figura 12 se tiene el diagrama de bloques del prototipo, como todos los programas grabados en esta clase de microcontroladores son secuenciales, esto quiere decir que las ordenes programadas dentro del *loop* del sistema serán ejecutadas una por una en orden mientras la plataforma Arduino tenga energía.

Primero, tomará la temperatura ambiental por medio del sensor TMP35 conectado a un pin analógico de la plataforma Arduino, como es un sensor analógico, el voltaje que recibe el pin se interpreta como un rango de 0-1023 el cual mediante fórmulas matemáticas se transforma en temperatura en grados centígrados. Después se tomara la medida del sensor de temperatura a prueba de agua, el cual al ser un sensor digital transmite los datos en señales de pulso que son interpretadas por el microcontrolador como valores numéricos en grados centígrados.

Para tomar la medición del potencial de Hidrogeno (pH) es necesario dos instrumentos, el primero es una probeta de medición, esta como tal es un sensor como los demás, el pH al que es sometido causa una reacción química dentro del sensor que cambia la conductividad del mismo, este cambio actúa como una resistencia variable cambiando el valor del voltaje de salida. Este sensor debe ser calibrado usando soluciones de potencial de hidrogeno conocido y seteando el valor de voltaje de salida a ese valor específico de pH. El segundo componente que se necesita es un circuito integrado que reciba este valor de voltaje y lo transforme a un valor digital entendible para el Arduino, este circuito cumple otras funciones como la de calibración de la probeta, permite elegir si medir una vez o de manera continua, además permite seleccionar si queremos medidas dependientes o independientes de la temperatura.

Luego tomara la medición de gas, a través del módulo del sensor MQ7, sensor de gas que tiene Oxido de Estaño en su interior que al entrar en contacto con el CO₂ produce una reacción química que cambia la conductividad del sensor y el

valor que entrega como voltaje de salida es interpretado para transformarlo a un valor de partes por millón de moléculas de CO₂.

Finalmente se mide el color usando un sensor de luz, el cual mide el color basándose en los 3 colores base de sistema RGB, tiene que medir ívidamente el valor de cada color en su base roja, verde y azul. De esta manera obtenemos datos digitales de cada base para formar el color completo en el sistema antes mencionado. Una vez terminadas todas las mediciones estas son transmitidas al ordenador para que sean leídas a través del puerto serial por el programa desarrollado en C# y sus datos sean mostrados de una manera amigable al usuario final.

2.5. Diseño del Software

En conjunto con el prototipo del circuito, se desarrollara también el prototipo del software que, por parte del Arduino recolectará y procesará los datos de medidos por los sensores en una especie de CPU para estos sensores que actúan de a modo de periféricos; y por otra parte una aplicación para Windows que recibirá la información procesada por el Arduino en una línea de texto a través del puerto serial, después de lo cual se mostrará al usuario final utilizando componentes gráficos del ambiente de desarrollo *Windows Form*. Finalmente se dará la posibilidad de exportar los datos recolectados y guardados por la aplicación a un archivo CSV el cual puede ser usado en aplicaciones conocidas como *Notepad* y *Excel* ambas de *Microsoft*.

En la figura 13, se puede observar la interacción entre el programa del Arduino y el software de la computadora final.

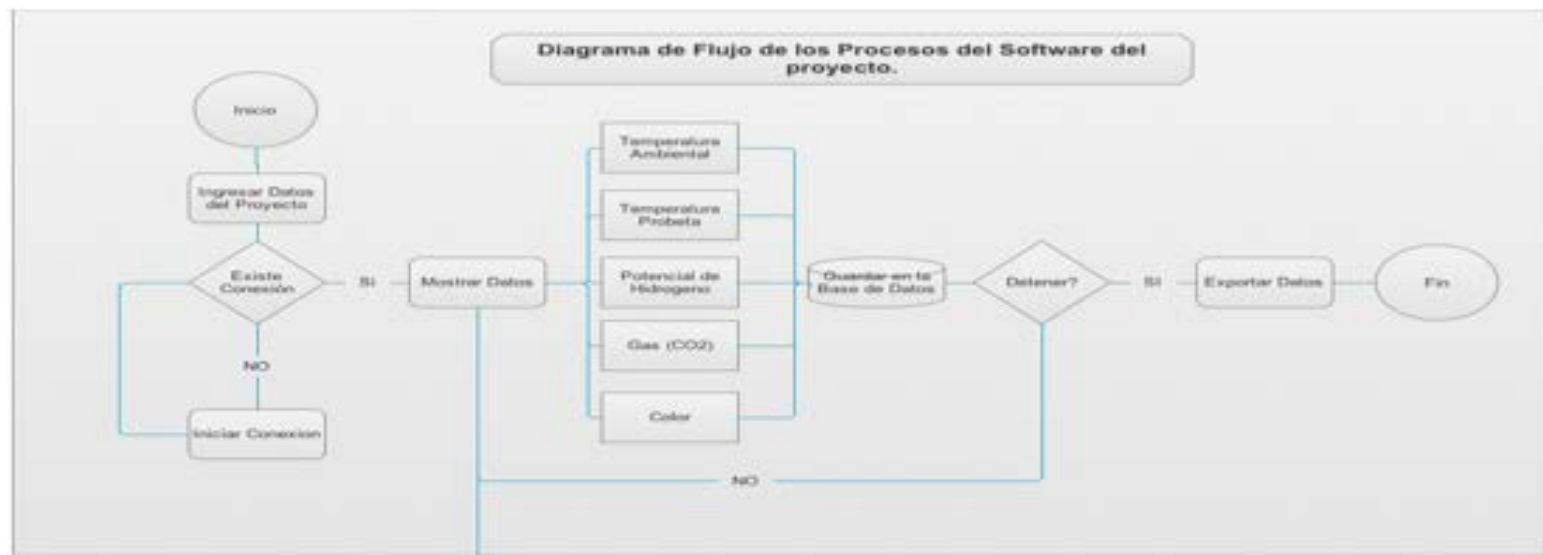


Figura 13: Diagrama de Flujo del sistema de recolección de datos

En el programa del microcontrolador, primero se declararon las variables necesarias, luego uno por uno se codificó los métodos que se necesitan para cada sensor, para finalmente en el LOOP, llamar a cada método y enviar la cadena de texto con la información completa al computador. Se puede revisar el código del software en el CD anexo al presente trabajo de titulación, el mismo que se encuentra comentado para su fácil comprensión.

2.5.1. Software de Desarrollo

Para el desarrollo del presente trabajo de titulación se utilizó dos ambientes de desarrollo. El primero el *ARDUINO DEVELOPMENT ENVIROMENT (ADE)*, es un IDE desarrollado por Arduino que permite funciones básicas de codificación con el sencillo lenguaje de programación de alto nivel propio muy parecido a C++.

Para la aplicación en Windows se utilizó de igual manera software propio de Microsoft, en este caso Visual Studio Express, utilizando C# como lenguaje de programación. Se realizó la aplicación como un proyecto de *Windows Form*, lo cual nos permite agregar controles gráficos de fácil entendimiento para el usuario final.

2.5.1.1. Arduino Development Environment (ADE)

El ambiente de desarrollo Arduino, es una interface amigable que nos permite escribir, verificar sintaxis, compilar y cargar a la plataforma Arduino programas fácilmente, como la mayoría de asistentes de programación para microcontroladores se debe programar en C o en alguna de sus variaciones, y el ADE compila en dos niveles lo que nos permite revisar errores de sintaxis del lenguaje C que sería el más amigable para el programador. En el primer nivel el compilador convierte las líneas de programación en C a ASSEMBLY, transformando el archivo con extensión .c o .h en un archivo .asm; una vez que el programa no tiene ningún error de sintaxis es transformado desde el archivo

con extensión .asm a un archivo con extensión .hex que el código a nivel de maquina el cual es el que se carga en la plataforma Arduino, en el microcontrolador ATmega32.

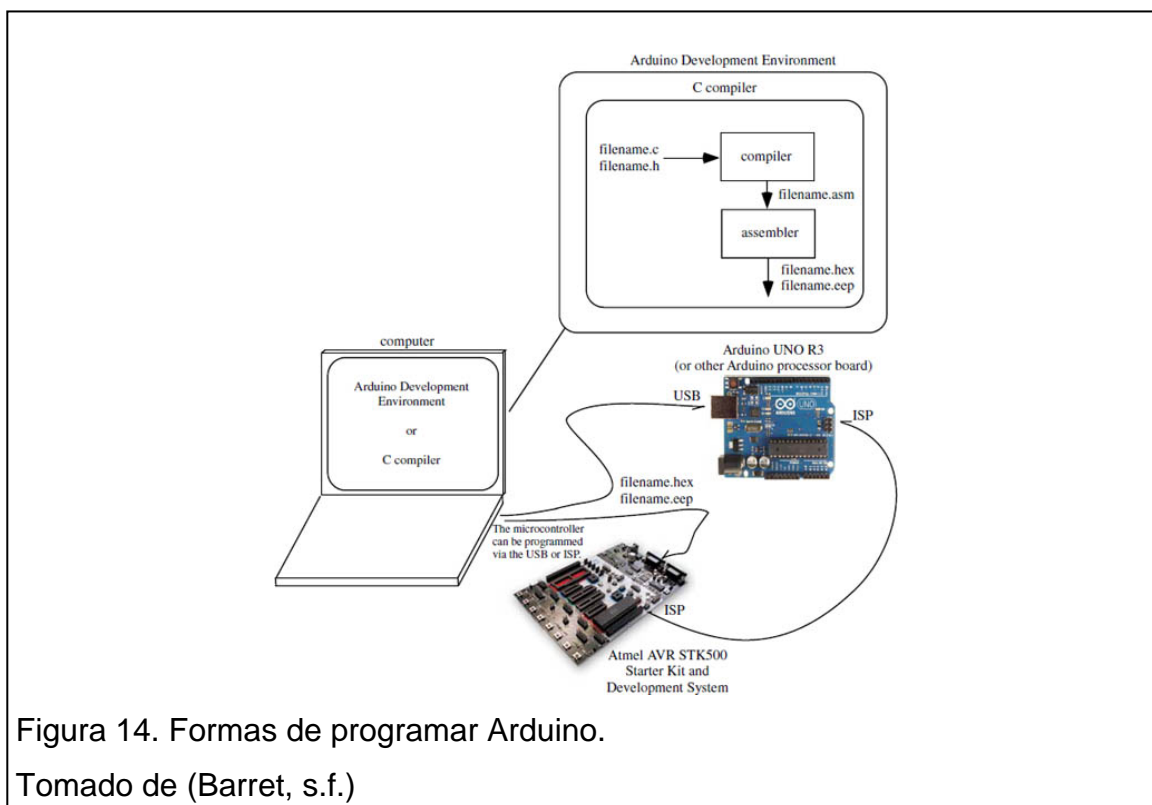


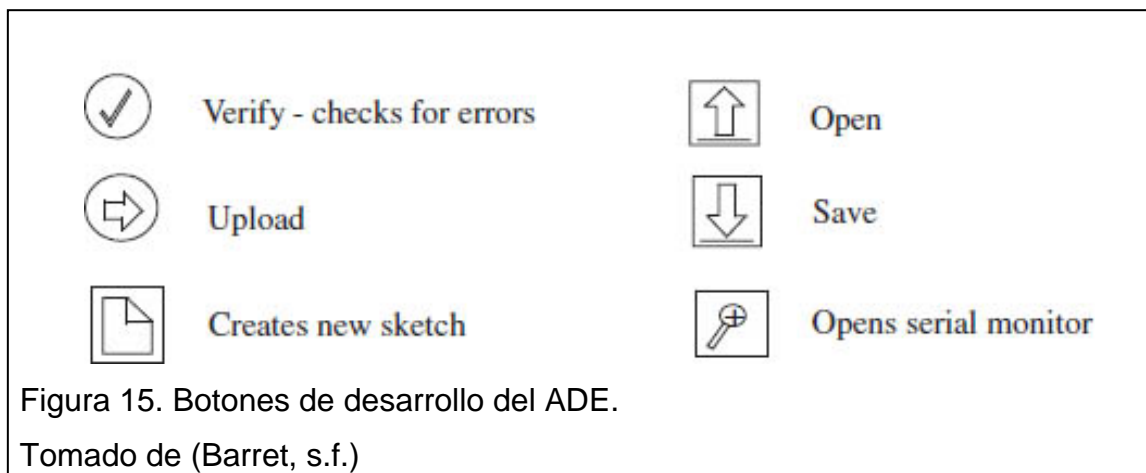
Figura 14. Formas de programar Arduino.

Tomado de (Barret, s.f.)

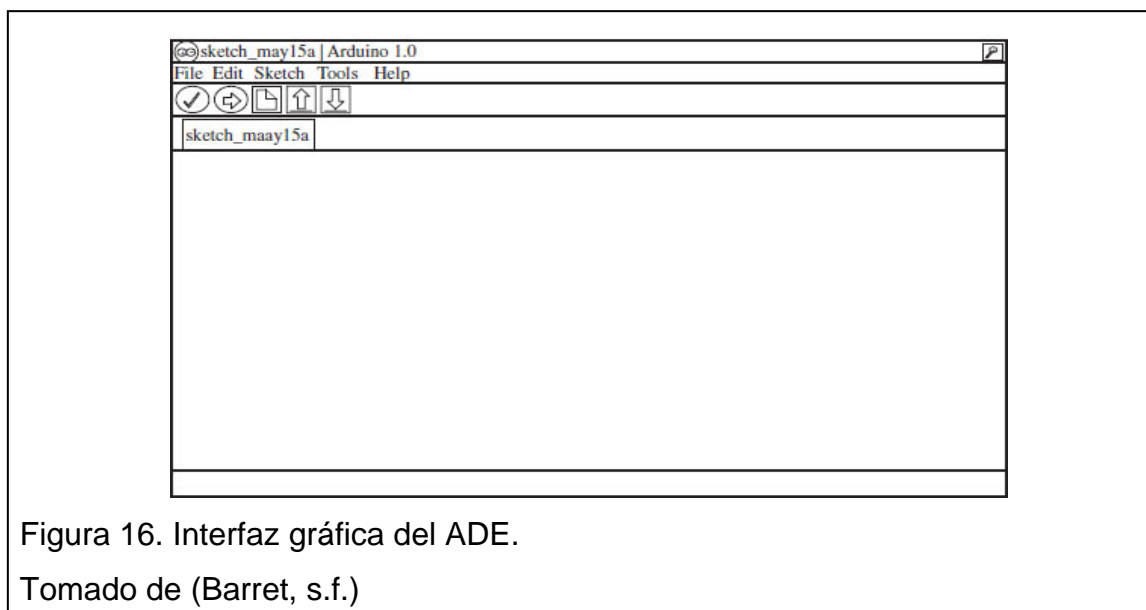
Como se muestra en la imagen existen dos maneras de programar el microcontrolador, la primera que he explicado previo a la imagen se utiliza conectando la computadora con la plataforma Arduino utilizando un cable serial a USB, y la interfaz ADE. La segunda manera que es más utilizada para modificar un programa ya grabado en la plataforma Arduino cuando esta implementada en un sistema más grande, ya que solo necesita la conexión de 6 pines con un ISP (*In-System Programmer*) de preferencia uno de Atmel, el cual graba un programa que ya tengamos listo en código de maquina en el microcontrolador.

La manera de programación es la misma que el lenguaje C, por ende todas sus reglas de sintaxis y lógica se aplican, por lo cual se va explicar un poco el entorno de la interfaz ADE.

La interfaz es bastante sencilla de usar ya que tiene solo los botones realmente necesarios, los cuales son seis: Verificar/Veri, señalado como un visto; Cargar/Upload, señalado como una flecha hacia la izquierda; Nuevo/New, señalado como una hoja de papel; Abrir/Open señalado como una puntos suspensivos con una flecha hacia arriba, Guardar/Save, señalado como puntos suspensivos con una flecha hacia abajo y Abrir Monitor Serial/Open Serial Monitor.



La interfaz del ADE, tiene también un editor texto con pestañas, y los menú comunes de archivo, edición, herramientas y ayuda. Más uno de Sketch, el cual nos da funciones de compilación, acceso a la carpeta donde se guardan los Sketch.



Los Sketch es el nombre que se le da al código escrito para Arduino, es similar a cualquier archivo de programación antes de ser compilado y traducido a lenguaje de máquina.

Existen dos funciones fundamentales que deben existir en todo Sketch, la primera es *setup()* de tipo *void*, en la cual se definen constantes y variables, se configura pines, se puede incluso poner funciones que se van a usar más de una vez en el código. La segunda función fundamental es la función *loop()* de tipo *void* que ejecuta el código dentro ella en secuencia paso a paso, y al terminar la secuencia programada se ejecuta de nuevo desde el inicio y así sucesivamente hasta que el programa sea detenido. (Barret, 2013)

2.5.1.2. Visual Studio

Visual Studio es una suite de herramientas de desarrollo de software basado en componentes y otras tecnologías para la construcción de potentes aplicaciones de alto rendimiento. Además, Visual Studio está optimizado para el diseño basado en el equipo, el desarrollo y despliegue utilizando equipo de servicio de la Fundación o de *Team Foundation Server*. (Microsoft, 2014).

Es un ambiente integrado de desarrollo (IDE, *Integrated Development Enviroment*) el cual fue creado por Microsoft para el desarrollo de programas de computadoras, aplicaciones web y Windows Form. Cuenta con un compilador y *debugger* que trabajan a nivel de máquina y a nivel de código fuente; el editor de código con el que cuenta es capaz de soportar *Intelli Sense* y *Code Refactoring*, así mismo contiene un diseñador.

Microsoft, 2015.

CAPÍTULO III: IMPLEMENTACION

3.1. Diagramas de Cableado

A continuación se va a mostrar el las conexiones que se realizaran en el prototipo del sistema. Las conexiones de los pines tanto en el Arduino como en el Protoboard y entre los diferentes componentes del proyecto.

Como se puede observar en la figura 17, solo se usarán 7 pines del Arduino, 3 digitales y 4 analógicos; el voltaje de operación y la tierra del sistema podría usarse externamente pero debido a que la plataforma Arduino posee pines destinados a proveer de manera confiable tanto VCC como GND, estos serán los utilizados. Además se usarán dos resistencias, la primera de 220ohms se utilizará para ofrecer la carga necesaria para el funcionamiento adecuado del sensor MQ-7, en caso de no tener acceso al módulo de Sain Smart. La segunda resistencia de 4.7k ohms se usa para poder proveer del poder necesario a la conexión de datos entre la plataforma Arduino y el sensor DS18B20 de acuerdo a las recomendaciones de la tecnología 1-Wire.

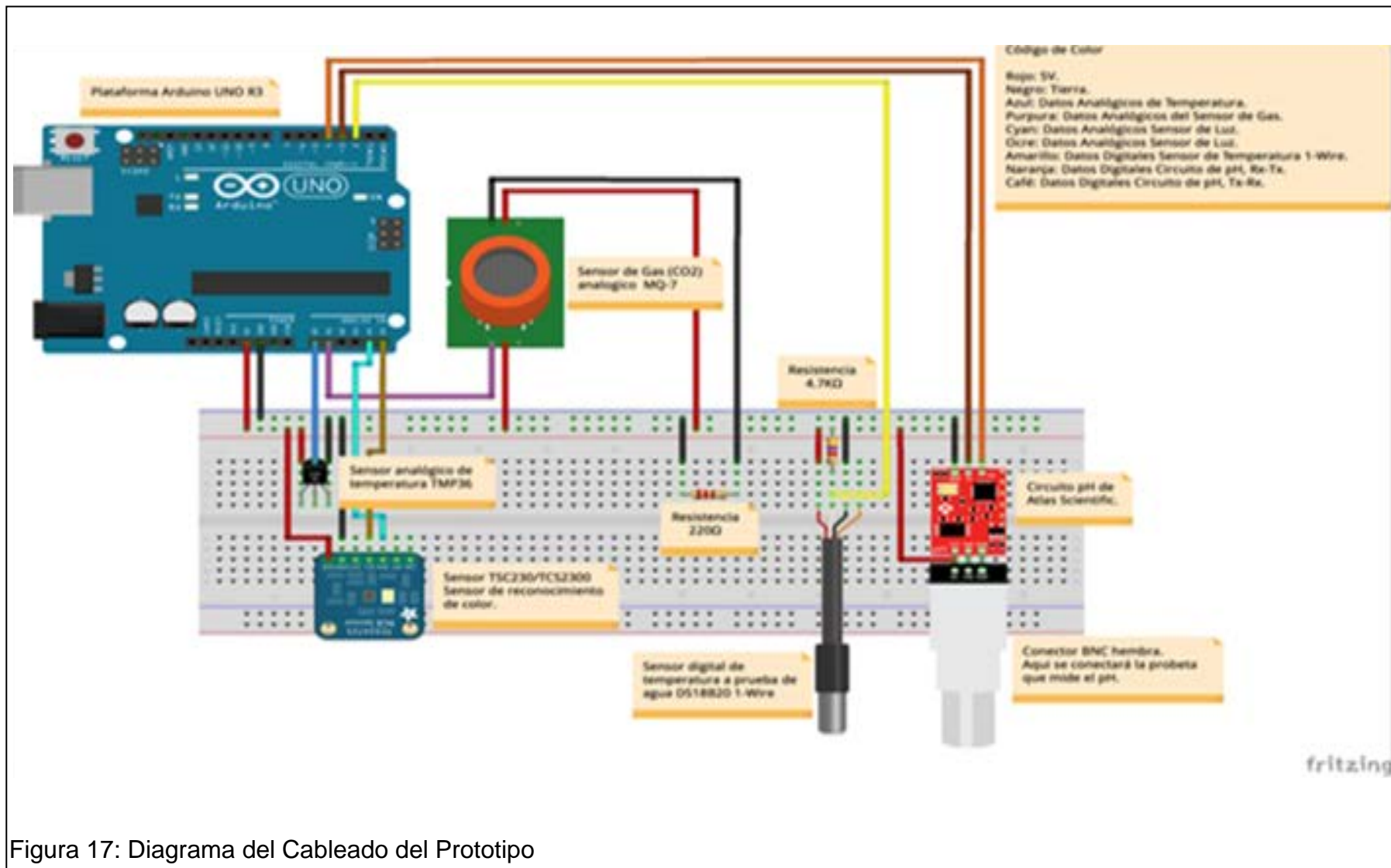


Figura 17: Diagrama del Cableado del Prototipo

En la figura 18 podemos observar la configuración específica del sensor de temperatura ambiental TMP36, para lo cual se utiliza los mismos pines que en el diagrama de cableado general. Como se muestra en la figura el sensor posee 3 pines, el del centro es el pin de salida de voltaje analógico del cual se calcula la temperatura ambiental; el pin a la izquierda del pin de datos en la vista frontal del sensor es el pin de VCC o voltaje de ingreso, y finalmente el pin de la derecha de la misma vista es el pin de tierra o GND.

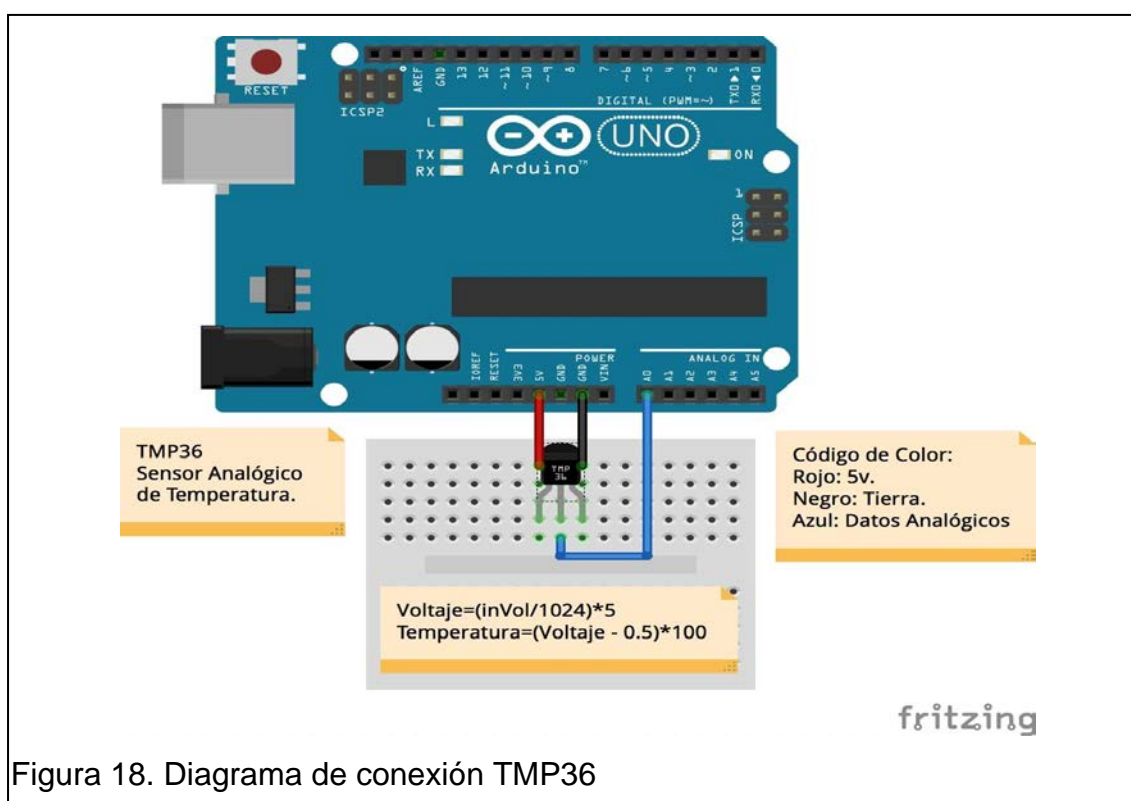


Figura 18. Diagrama de conexión TMP36

En la figura 19 se puede observar la configuración básica de un sensor de gas de la familia MQ-X en este caso el MQ-7 que monitorea los niveles de CO₂ en el ambiente, ya que este tipo de sensores dependen de una reacción química para poder interpretar la cantidad de partículas de CO₂ en el ambiente necesita de dos pines por los cuales se calienta la sustancia dentro del sensor, en este caso Oxido de Estaño, este al reaccionar con el CO₂ cambia su conductividad lo cual se interpreta en el voltaje de salida para poder calcular las partes por millón de CO₂ en el ambiente. Al tener el módulo de *SainSmart*,

este circuito integrado nos da la lectura de manera digital lo que hace mucho más sencillo su uso.

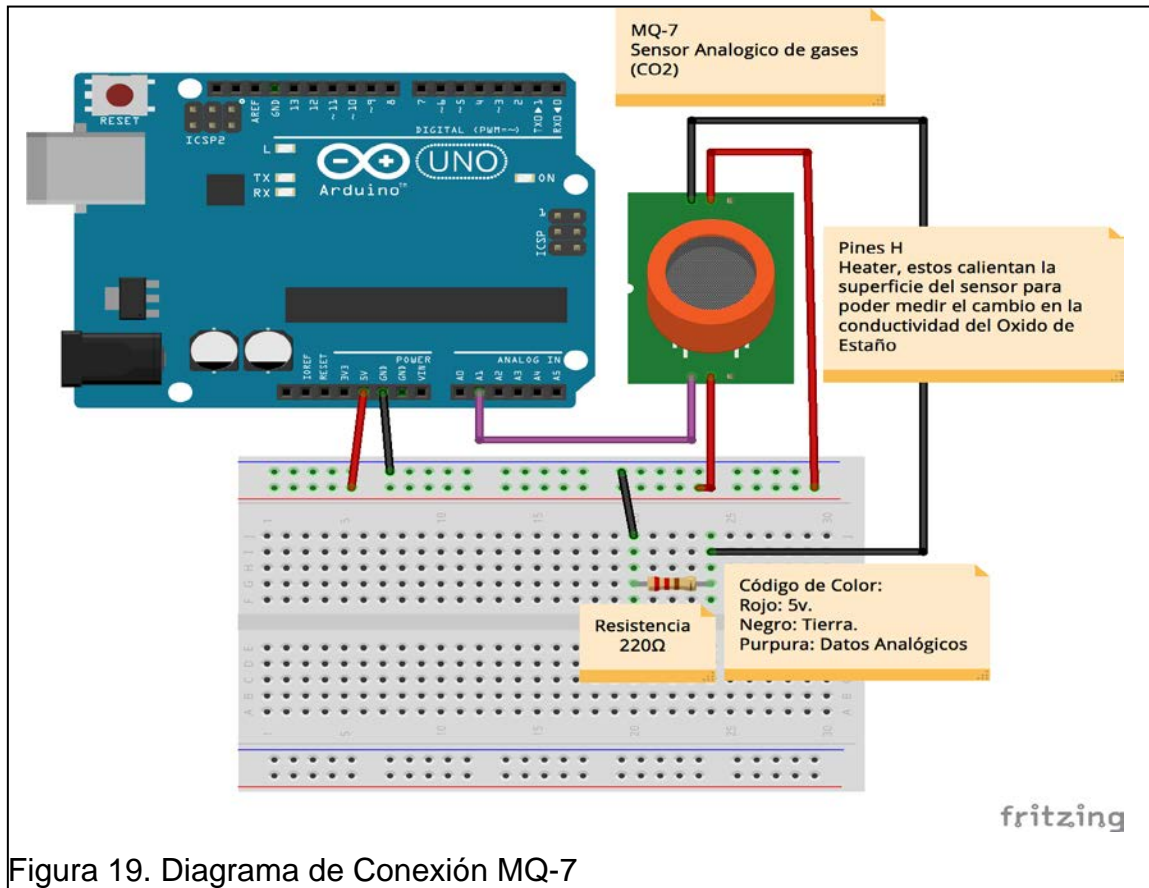


Figura 19. Diagrama de Conexión MQ-7

En la figura 20 se puede observar la conexión del sensor DS18B20, el cual es un sensor de temperatura digital que se encuentra dentro de una probeta para poder ser sumergido en líquidos. Este sensor desarrollado en la tecnología 1-Wire, nos permite conectar más de un sensor en el mismo pin de datos digitales en la plataforma Arduino, lo cual podría permitir hacer mediciones de líquidos múltiples de ser necesario.

Aunque en muchas fuentes se dice que solo es necesario conectar el pin de datos, ya que este proporciona energía al sensor y de ahí la denominación de 1-Wire, la experimentación con el mismo llevo a la conclusión de que los otros pines deben estar conectados a voltaje o VCC y a tierra o GND, siguiendo el estándar de color de cable, rojo a voltaje y negro a tierra.

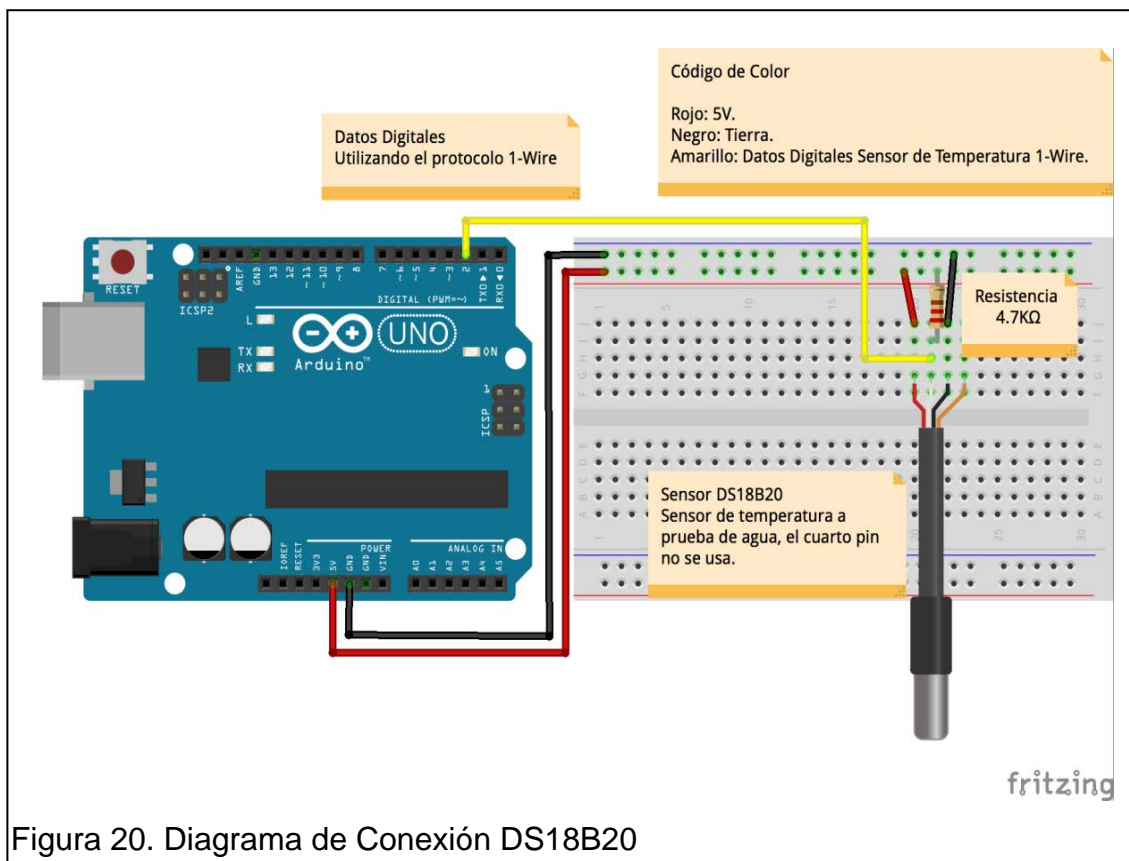


Figura 20. Diagrama de Conexión DS18B20

Finalmente tenemos la medición del potencial de Hidrogeno, el cual se realiza, como se puede observar en la figura 21, a través del circuito integrado de medición de pH de Atlas Scientific, el cual recibe los datos de la probeta por medio de una conexión BNC, y los transforma a digital para su fácil lectura. Además el circuito nos permite calibrar la probeta al sumergirla en sustancias de pH conocido y setear a ese valor por medio de comando en código ASCII.

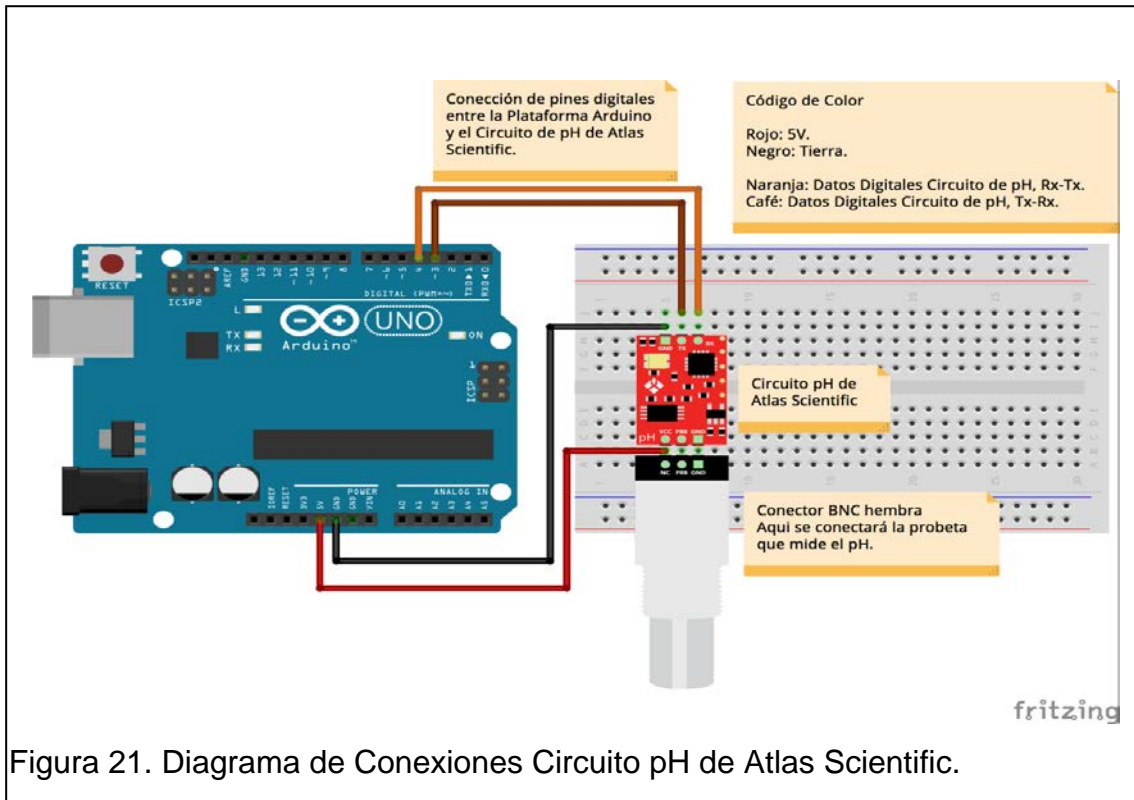


Figura 21. Diagrama de Conexiones Circuito pH de Atlas Scientific.

3.2. Ensamblaje del Prototipo

Debido a las facilidades del diseño que tiene la plataforma Arduino se utilizará un diseño tipo *Shield*, el cual encaja en la parte superior del Arduino; como se puede observar en la figura 22 a continuación, los pines de salida del Arduino son trasladados a la placa PCB y desde los cuales se conectan a través de las conexiones grabadas en la placa cada

Se ha mantenido los mismos pines de salida y entrada que se ha manifestado en las figuras anteriores de los diagramas de cableado, aumentando solamente una alarma sonora de tipo *Buzzer*, que se conecta al pin 13 para su alimentación con energía cuando las partículas por millón de CO_2 sobrepasen las 500, y por su otro extremo el *Buzzer* se conecta a tierra.

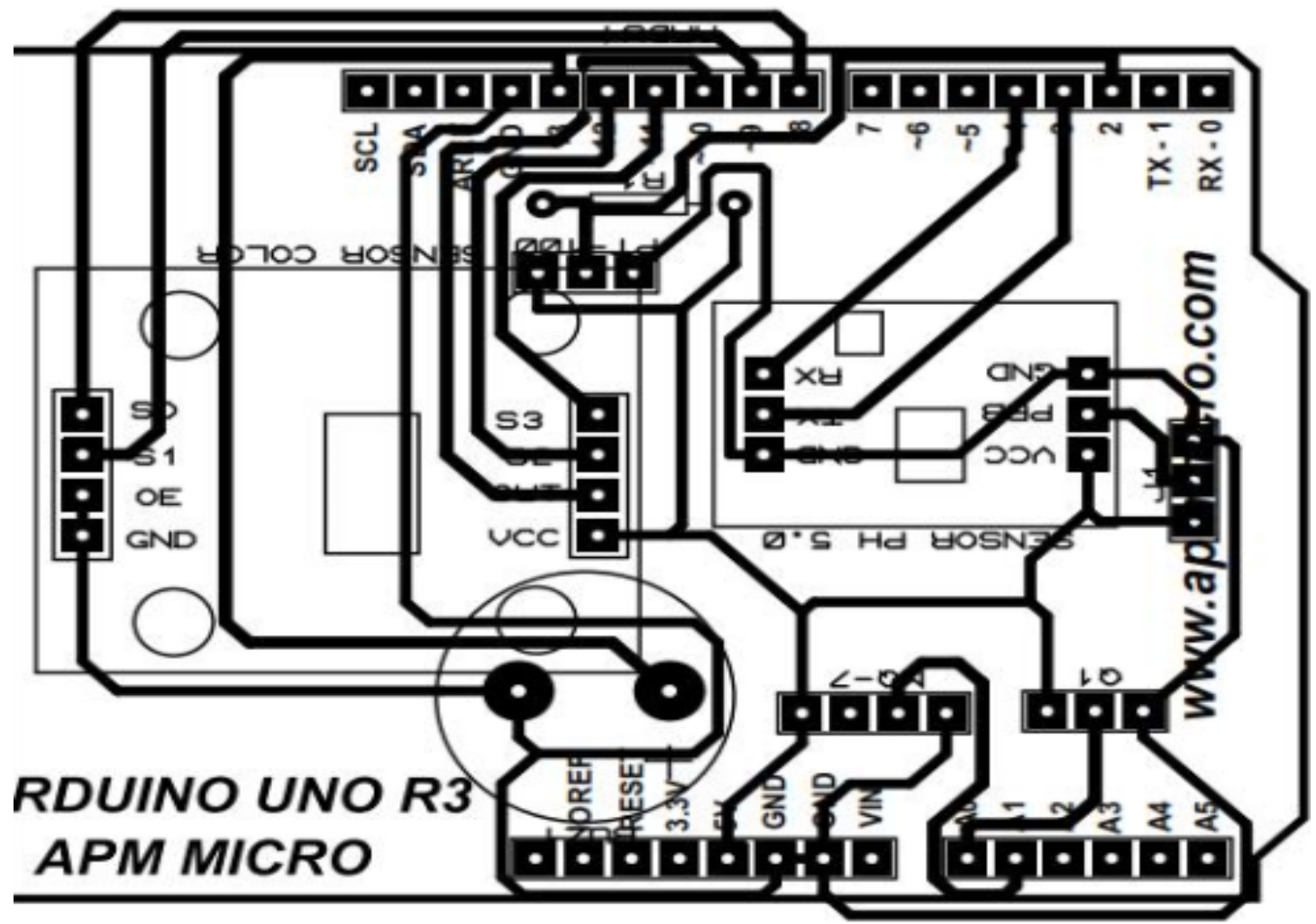


Figura 22: Layout para placa PCB

CAPÍTULO IV: PRUEBAS Y RESULTADOS

4.1. Análisis de resultados

Para probar el funcionamiento del sistema se realizó el experimento químico de creación de cristales usando sulfato de cobre. El procedimiento de este experimento es bastante sencillo, primero se debe moler el sulfato de cobre hasta obtener un polvo fino, luego se lo debe mezclar en un recipiente lleno de agua y calentarlo hasta que se mezclen homogéneamente a una temperatura entre 30 y 50 grados centígrados, después se debe verter la solución en un recipiente seco y dejar que se evapore los remanentes de agua haciendo que el sulfato de cobre se cristalice tomando un color azulado.

Por medio de este prototipo se midió la temperatura a la que se calentó la solución para evitar que esta sobrepasara los rangos que se pedía, al mismo tiempo se midió la temperatura ambiental y como esta variaba durante el periodo del experimento. Durante la misma, ya que no se creaba humo se constató el funcionamiento del sensor de gas. El resultado final fue el esperado se creó unos pequeños cristales de color azulado el cual fue registrado por el sensor de luz.

Varios sensores como los de luz y de temperatura tuvieron problemas de precisión, en caso del sensor de luz el mayor problema fue la dispersión de la luz blanca de los LEDs, para lo cual se concentró la iluminación hacia la parte superior de los LEDs en lugar de permitir la diseminación lumínica. El sensor de temperatura ambiental, al estar cerca de los demás elementos del prototipo tiende a dar medidas incorrectas al estar a bajas temperaturas, ya que el funcionamiento propio de la plataforma mantiene un temperatura entre 25 y 28 grados centígrados.

El sensor DS18B20, que es el sensor de temperatura a prueba de agua, al estar en una probeta metálica empieza con mediciones frías hasta que el metal protector equipara su temperatura con la temperatura del líquido que lo rodea.

El sensor de CO₂, tiende a tener mediciones sensibles, por lo cual cuando se respira cerca del mismo puede dar resultados erróneos y activar de manera errónea la alarma sonora.

A continuación se puede observar la vista final del software en donde están los datos del experimento químico. Como se puede concluir con la figura 18. La meta de monitorear y guardar los datos por un periodo de tiempo fue alcanzada, sin embargo se detectaron pequeños errores de precisión en los sensores de temperatura, en especial en el sensor a prueba de agua el cual al estar recubierto de metal tiene una superficie fría que es tomada como primera lectura y la medición correcta de la temperatura de las sustancias llega cuando esta igual la temperatura del metal protector con la de la sustancia en la que está sumergido. La precisión del sensor de luz tampoco es 100% acertada, si bien el color es bastante parecido al resultado final de los cristales no es exactamente el mismo.

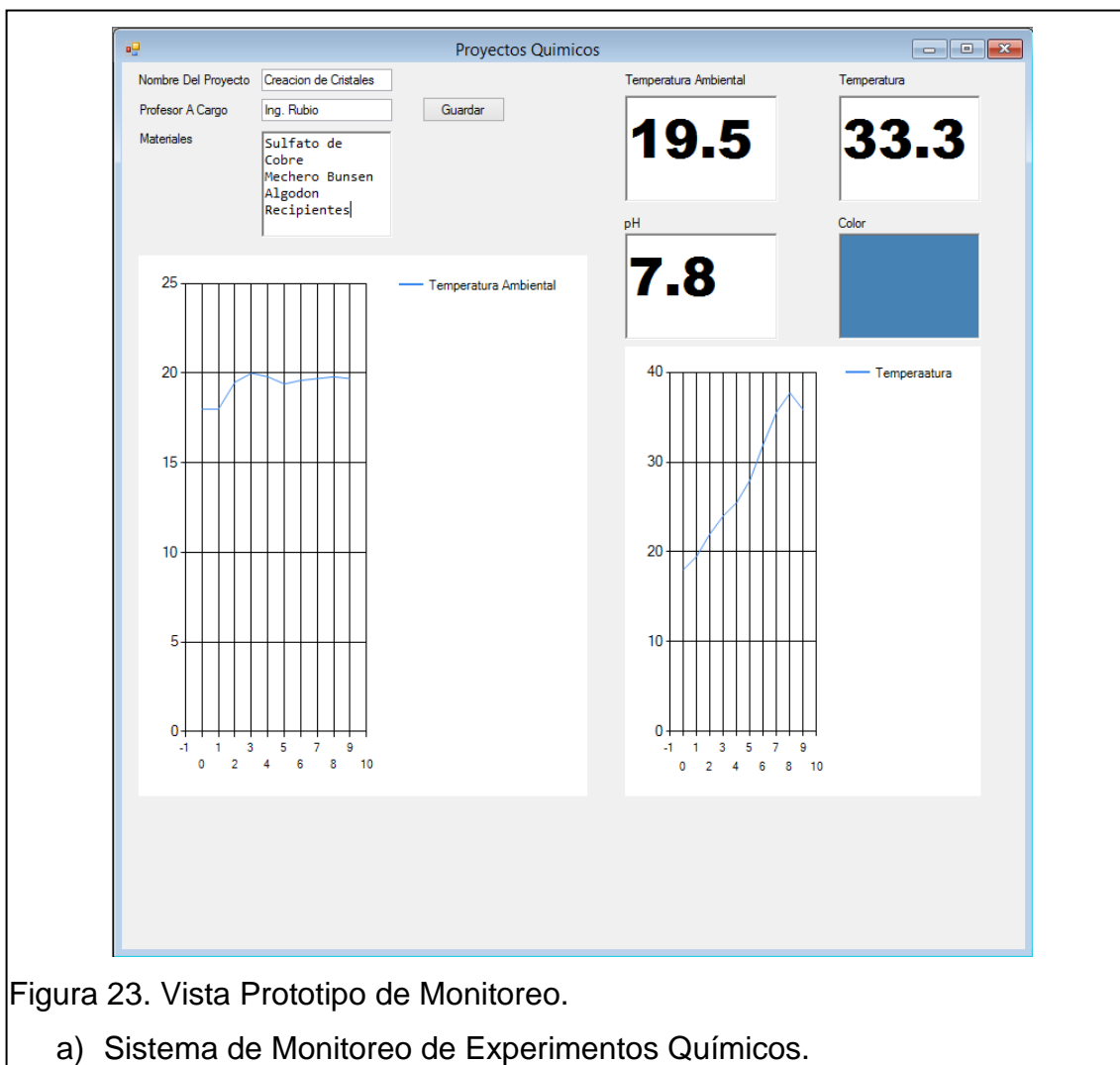


Figura 23. Vista Prototipo de Monitoreo.

a) Sistema de Monitoreo de Experimentos Químicos.

En la figura 23, se puede apreciar la vista de monitoreo de experimentos químicos, en cuales se pueden apreciar los siguientes elementos:

4.1.1. Nombre del Proyecto

En este campo debe ser escrito el nombre del proyecto por parte del usuario final.

4.1.2. Profesor a Cargo

En este campo debe de ser escrito el nombre del profesor a cargo del experimento

4.1.3. Materiales

En este campo se deben escribir los diferentes materiales que se usaran en el proyecto de experimentación química separados por un *enter*.

4.1.4. Botón Guardar

Este botón tiene la función de guardar los campos previamente mencionados en una tabla e inicia el proceso de guardar los datos recibidos por el microprocesador, los mismos datos que serán mostrados en los diferentes componentes como los gráficos de línea para las temperaturas, cuadros de texto para mostrar un valor numérico y el color.

4.1.5. Gráficos de temperatura.

Estos gráficos de líneas, muestran la curva de la variación de la temperatura en las ultimas 8 mediciones, las cuales se actualizan cada que haya 8 mediciones nuevas.

4.1.6. Cuadros de texto de valor numérico

Estos componentes muestran el ultimo valor numérico de las mediciones de las temperaturas y del potencial de hidrogeno.

4.1.7. Cuadro de color

Este componente muestra la última medición de color tomada por el sensor de luz.

4.2. Reporte de Costos

Tabla 9. Costos del proyecto.

Componente A	Precio A	Componente B	Precio
Arduino R3 UNO	\$26.49	Arduino Mega	\$37.03
DS18B20 (1-Wire)	\$3.81	-	\$3.81
TMP36	\$5.95	LM35	\$6.90
PinPoint pH Probe	\$35.95	Atlas pH Probe	\$70,99
pH Circuit	\$28.00	EZO™pH Circuit	\$34.00
TCS230/TSC3200	\$7.50	TCS34735	\$12.32
MQ-7 Module	\$10.20	MQ-2 Module	\$8.99
Conector BNC 2P	\$2.95	-	\$2.95
Sub Total:	\$120.85		\$175.90
Herramientas	±\$40.00		±\$40.00
Shipping	±\$60.00		±\$60.00
Total:	±\$221		±\$276

CONCLUSIONES Y RECOMENDACIONES

Conclusiones

A través del ensamblaje del prototipo y su trabajo en conjunto con el software diseñado en Visual Studio, se logró tener una herramienta de fácil uso, que permite de una manera didáctica la recolección y procesamiento de datos ambientales como temperatura, potencial de hidrogeno, colores y gas en el entorno inmediato.

Para lograr un sistema automático se programó en el microcontrolador él envió constante de datos en una línea del puerto serial, la cual es automáticamente leída y almacenada por el software desarrollado en Visual Studio. Se utilizó estándares en la codificación tanto del microcontrolador como en la del software de interfaz con los usuarios finales.

Los errores en las mediciones de temperatura, se encuentran en los rangos establecidos por el *Datasheet* de cada sensor, por otra parte la medición de colores tiene errores importantes, estos son ocasionados por la luz blanca que se usa para obtener la medición y la interpretación del pulso digital no es del toda precisa, ya que el sensor da medidas parecidas para los dos parámetros más altos y bastante baja para el tercer parámetro.

De este prototipo se puede partir a sistemas más complejos o dinámicamente escalables, lo que quiere decir que se puede utilizar este trabajo de titulación como base para aumentar o cambiar el tipo de sensores, teniendo como fundamento el monitoreo de elementos medibles por medio de sensores. Un ejemplo claro de sistemas parecidos es el de monitoreo de estaciones de geología en pozos petroleros, en los cuales se usa sensores sencillos como sensores de proximidad y sensores más complejos como sensores ultrasónicos para medir distancias y sensores de suma importancia y sensibilidad para detectar no solo gases peligrosos pero también combinaciones de estos que

pueden terminar siendo altamente inflamable; lo cual conllevaría una mayor inversión en sensores de características industriales o de excelente precisión, sin contar con un software más robusto y seguro para dichas aplicaciones.

Recomendaciones

Se recomienda cubrir las partes laterales de los LEDs en el sensor de luz, para enfocar la luz blanca que emite y de esa manera obtener mejores lecturas del mismo sensor. De igual manera es necesario evitar que los sensores se mojen, a excepción de las probetas que son a prueba de agua. Es necesario realizar un chequeo periódico de sensores como el de gas o el de potencial de Hidrogeno ya que estos deben ser calibrados periódicamente. Al igual que muchos otros sensores que podrían ser usados en sistemas basados en este trabajo de titulación.

Al utilizar métodos en la programación del Arduino, se recomienda probar cada uno por separado antes de utilizar todos los métodos en un solo programa, adicional hay que tomar en cuenta que el tiempo de respuesta de unos sensores es distinto al de otros, un ejemplo de este trabajo de titulación es el sensor de pH, cuyo tiempo de respuesta es mayor que el resto de sensores, para solucionar el problema se guardo todos los datos adquiridos en una variable local antes de transmitir en una sola cadena de texto los datos al computador del usuario.

También se puede utilizar este prototipo para el monitoreo y control ambiental de locaciones como hogares o pequeñas industrias, debido a que mucho de estos sensores se pueden usar también en mediciones ambientales de una casa y la Plataforma *Arduino* es capaz de controlar dispositivos activos para el control de ambientes.

REFERENCIAS

- ARDUINO. (s.f.). *The Arduino Playground*. Recuperado el 12 de Enero de 2015 de <http://playground.arduino.cc>
- ARDUINO. (s.f.). Using an Arduino as an AVR ISP (In-System Programmer).
Recuperado el 19 de Diciembre de 2014 de <http://arduino.cc/en/Tutorial/ArduinoISP>
- Atmel. (s.f.) ISP Programming. Recuperado el 19 de Diciembre de 2014 de http://www.atmel.com/webdoc/stk600/stk600.programming_isp.html
- Barret, S. (s.f.). *Arduino Micro controller Processing for Everyone!*. Recuperado el 17 de Diciembre de 2014 <https://www.safaribooksonline.com/library/view/arduino-microcontroller-processing/9781627052535/xhtml/cover.html>.
- Blankenship, J., Mishal, S. (2011). *Interfacing The Arduino with a PC Using RobotBASIC's Protocol*.
- Desjardins, P. (2013). *Visual Studio Condensed*. New York, United States of America: Apress.
- Di Justo, P., Gertz, E. (2012). *Atmospheric Monitoring with Arduino*. Sebastopol, United States of America:Maker Media.
- Di Justo, P., Gertz, E. (2012). *Enviromental Monitoring with Arduino*. Sebastopol, United States of America:Maker Media.
- Margolis, M. (2011). *Arduino Cookbook*. Sebastopol, United States of America:O'Reilly Media.
- MaximIntegrated. (s.f.). 1-WIRE, Recuperado el 25 de Marzo de 2015 de <http://www.maximintegrated.com/en/products/comms/one-wire.html>
- Microsoft. (s.f.). *Visual Studio Express*. Recuperado el 12 de enero de 2015 de <http://www.visualstudio.com/downloads/download-visual-studio-vs>
- Microsoft. (s.f.). *Visual Studio Resources*. Recuperado el 10 de Enero de 2015 de <https://msdn.microsoft.com/en-us/vstudio/cc136611.aspx>
- Monk, S. (2011). *Programming Arduino™: Getting Started with Sketches*. New York, United States of America:McGraw-Hill.

- Monk, S. (2011). *Programming Arduino™ Next Steps: Going Further with Sketches*. New York, United States of America: McGraw-Hill
- Monk, S., Ladyada. (s.f.). *Adafruit learning system*. Recuperado el 12 de enero de 2015 de <http://learn.adafruit.com>
- Sapiensman. (s.f.). *Medición de temperatura*. Recuperado el 12 de Enero de 2013 de http://www.sapiensman.com/medicion_de_temperatura/
- SENSING. (2014). *Sensor de temperatura por infrarrojos*. Recuperado el 12 de Enero de 2013 de http://www.sensores-de-medida.es/sensing_sl/SENSORES-Y-TRANSDUCTORES_35/Sondas-de-temperatura-y-humedad_92/Sensor-de-temperatura-por-infrarrojos_174/
- Shiloh, M., Banzi, M. (2014), *Make: Getting Started with Arduino, 3rd Edition*. Sebastopol, United States of America: Maker Media.
- Rubio, N. (2014). Laboratorio química analítica, cuantitativa y cualitativa de la Universidad Central del Ecuador. *Entrevista realizada por Luis Alejandro Añazco V.* Quito, Ecuador.

ANEXOS