

Universidad de Costa Rica

Proyecto Final

Dispensador de pastillas

Prof. Marco Villalta

Daniela Ríos Mora B65854

José Eras Saborío B72704

Grupo 01

7 de diciembre de 2023

Índice

1. Introducción	1
2. Objetivos y alcances	2
2.1. Objetivos	2
2.2. Alcances	3
3. Justificación	4
4. Marco teórico	5
4.1. Arduino UNO	5
4.1.1. Componentes Clave	5
4.1.2. Entorno de Desarrollo	5
4.1.3. Programación y Carga de Código	5
4.2. Protoboard	5
4.3. LCD	6
4.4. Diodo LED	7
4.5. Pulsador	8
4.6. Resistor	9
4.7. Servomotor	9
4.8. IoT	10
4.9. ThingBoard	10
5. Desarrollo	11
5.1. Lógica Principal (<code>loop()</code>)	11
5.2. Funciones Adicionales	11
5.3. Internet de las Cosas	13
5.4. Implementación del Circuito	13
5.4.1. Elementos del Circuito:	13
5.4.2. Conexiones:	13
6. Análisis de resultados	15
7. Conclusiones, observaciones y recomendaciones	19
7.1. Conclusiones	19
7.2. Observaciones:	19
7.3. Recomendaciones:	19
Referencias	20

1. Introducción

La familia moderna utiliza las diversas herramientas tecnológicas que faciliten el acompañamiento y seguimiento de los diversos tratamientos médicos a los cuales las personas de la tercera edad se ven sometidos, ya que al adquirirse responsabilidades la capacidad de mantener una vigilancia completa al adulto se ve limitada a tiempos muy cortos o no lo suficientemente permanentes en los cuales se pueda asegurar la correcta ingesta de los medicamentos asignados al paciente.^[6]

El Dispensador de Pastillas Diario es un dispositivo inteligente que utiliza una combinación de componentes electrónicos para garantizar que las personas tomen sus medicamentos a tiempo. El usuario configura la hora en la que desea que el dispensador de pastillas libere su dosis diaria, esto se hace mediante el uso de los pulsadores.

Cuando llega la hora programada, el dispensador de pastillas entra en acción. Avisando al usuario por medio de una alarma que debe tomarse en medicamento en ese momento. Cuando se confirma la hora de dispensación programada un servomotor de rotación entra en acción, girando una cantidad de grados de manera que las pastillas lleguen al lugar de depósito donde serán dispensadas y de fácil acceso al usuario. Después de la dispensación, el dispensador de pastillas diario se restablece automáticamente para programar la próxima dosis diaria. El ciclo se repite cada 24 horas, asegurando una administración regular de la medicación.

Se desea incluir una pantalla LCD que muestre un mensaje de recordatorio, que indica que es hora de tomar las pastillas. Además, el usuario puede ver la hora programada previamente. Si el usuario necesita ajustar la hora de dispensación, puede hacerlo mediante los pulsadores antes de la liberación automática de pastillas.



Figura 1: Dispensadores de pastillas (Productos de Amazon.com)

2. Objetivos y alcances

2.1. Objetivos

El objetivo general del proyecto es:

Desarrollar un Dispensador de Pastillas Diario que facilite la administración oportuna de medicamentos, y documentar su funcionamiento a través de un informe técnico. Los objetivos específicos son:

- Diseñar y construir el hardware del Dispensador de Pastillas Diario, incluyendo componentes electrónicos como servomotores, pulsadores, una pantalla LCD y una alarma.
- Programar el Dispensador de Pastillas Diario que permita la programación de la hora de dispensación, la activación de la alarma y el control del servomotor.
- Desarrollar un reporte técnico que explique la funcionalidad del Dispensador de Pastillas, análisis de resultados y conclusiones.

2.2. Alcances

Los alcances de cada objetivo se presentan a continuación en la tabla 1:

Objetivo	Alcances
Diseñar y construir el hardware del Dispensador de Pastillas Diario, incluyendo componentes electrónicos como servomotores, pulsadores, una pantalla LCD y una alarma.	<ul style="list-style-type: none"> ■ Seleccionar y adquirir los componentes electrónicos necesarios ■ Diseñar un circuito electrónico que controle la liberación de pastillas y la pantalla LCD. ■ Integrar un sistema de alarma audible y visual para recordar al usuario la toma de medicamentos. ■ Implementar un sistema de pulsadores para configurar la hora de dispensación.
Programar el Dispensador de Pastillas Diario que permita la programación de la hora de dispensación, la activación de la alarma y el control del servomotor DC por I/O Pin	<ul style="list-style-type: none"> ■ Programar un microcontrolador Arduino UNO para gestionar la hora de dispensación. ■ Desarrollar un algoritmo que active la alarma a la hora programada. ■ Codificar el control del servomotor para dispensar las pastillas. ■ Implementar una interfaz de usuario en la pantalla LCD para configurar la hora de dispensación.
Desarrollar un reporte técnico que explique la funcionalidad del Dispensador de Pastillas, análisis de resultados y conclusiones.	<ul style="list-style-type: none"> ■ Presentar la funcionalidad del Dispensador de Pastillas Diario. ■ Realizar un análisis de los resultados obtenidos a través de pruebas de funcionamiento.

Tabla 1: Alcances del Proyecto. Creación propia.

3. Justificación

Para la justificación de este proyecto se presentan casos de uso mediante 3 historias de usuario.

- Daniela, paciente con múltiples medicamentos.

Como Daniela, una paciente con múltiples condiciones médicas, necesito un dispensador de pastillas diario para asegurarme de tomar los medicamentos correctos en los momentos adecuados, de tal manera que mejore mi adherencia a los tratamientos y reduzca el riesgo de errores en la medicación.

- Jose, el cuidador preocupado.

Como Jose, el cuidador de una persona mayor, necesito un dispensador de pastillas para garantizar que mi ser querido tome sus medicamentos de manera puntual y segura cuando no puedo estar presente, de tal manera que tenga la tranquilidad de que está siguiendo su tratamiento adecuadamente.

- Pablo, el paciente con discapacidad visual.

Como Pablo, un individuo con discapacidad visual, busco utilizar un dispensador de pastillas diario para tomar mis medicamentos de manera independiente, necesito que el dispensador me proporcione recordatorios auditivos y táctiles, de tal manera que pueda tomar las dosis correctas y mantenga mi salud en buen estado.

Como se puede ver en las historias de usuario, el tener un dispensador de pastillas es muy útil. Este nos ayuda a recordar los medicamentos necesarios sin descuidar nuestras actividades diarias corriendo el riesgo de olvidar el medicamento.

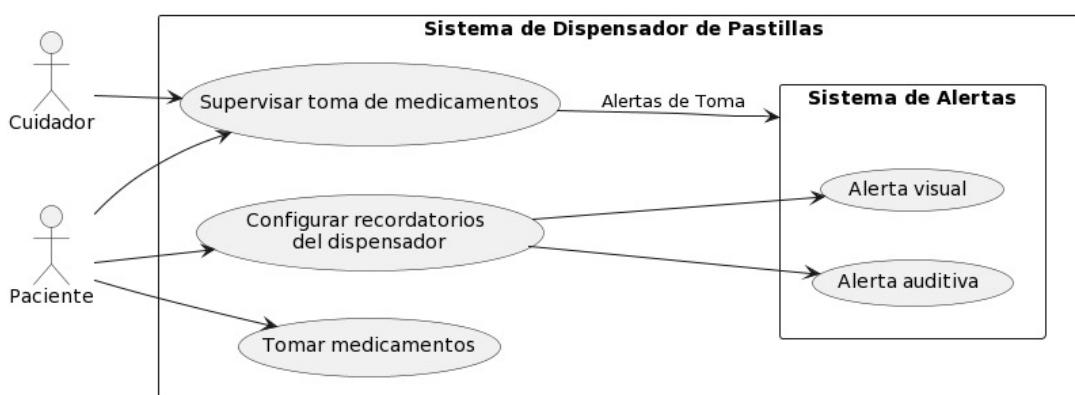


Figura 2: Diagrama UML que representa los casos de uso. [Elaboracion propia]

Este proyecto toma como referencia lo estudiado en el curso IE0624 - Laboratorio de Microcontroladores, en el cual se estará empleando un arduino uno y diversos componentes electrónicos a los cuales se les debe de calcular su respectivo valor para asegurar el buen funcionamiento de este mismo.

4. Marco teórico

4.1. Arduino UNO

Arduino Uno es una plataforma de prototipado electrónica de código abierto que se destaca por su hardware y software flexibles y fáciles de usar. Desarrollado por Arduino LLC, este microcontrolador ha ganado popularidad en la creación de proyectos interactivos y dispositivos electrónicos debido a su simplicidad y versatilidad [3].

4.1.1. Componentes Clave

Algunas características importantes del microcontrolador Arduino UNO se encuentran a continuación [3]:

- **Microcontrolador ATMega328P:** Este componente sirve como el núcleo del Arduino Uno. Es un microcontrolador de 8 bits que cuenta con memoria flash, EEPROM y SRAM incorporadas.
- **Puertos de Entrada/Salida (I/O):** Arduino Uno dispone de pines digitales y analógicos que permiten la interacción con el entorno. Los pines digitales pueden configurarse como entrada o salida, mientras que los analógicos se utilizan para entradas analógicas.
- **Puertos de Comunicación:** Incluye puertos UART, SPI e I2C que facilitan la comunicación con otros dispositivos.
- **USB y Conector de Alimentación:** La presencia de un puerto USB permite la conexión a un ordenador para la programación y la alimentación del Arduino Uno.

4.1.2. Entorno de Desarrollo

- **Arduino IDE (Entorno de Desarrollo Integrado):** Proporciona una interfaz fácil de usar para la escritura y carga de código en la placa Arduino.
- **Lenguaje de Programación:** Arduino utiliza una versión simplificada de C++, lo que facilita la programación, especialmente para aquellos que están comenzando.

4.1.3. Programación y Carga de Código

- **Sketch:** Un programa en Arduino se denomina "sketch". Este consta de dos funciones principales: `setup()` para la configuración inicial y `loop()` para el bucle principal.
- **Carga de Código:** A través del cable USB, el código se carga en la placa Arduino Uno y se ejecuta de inmediato.

4.2. Protoboard

La protoboard es un componente fundamental en el desarrollo de circuitos electrónicos. Consiste en una placa con agujeros conectados eléctricamente entre sí en patrones preestablecidos, permitiendo la conexión y prueba rápida de componentes sin la necesidad de soldadura. Suele dividirse en filas

y columnas donde los componentes electrónicos, como resistores, transistores, LEDs y cables, se insertan para crear conexiones.

En una protoboard, los nodos son los puntos de conexión eléctrica donde los componentes se conectan entre sí. Cada agujero en la protoboard está conectado internamente siguiendo un patrón preestablecido. En general, los agujeros en una misma fila horizontal están conectados eléctricamente, mientras que los agujeros a lo largo de una misma columna vertical no están conectados entre sí.

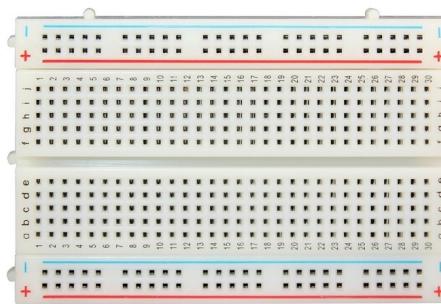


Figura 3: Protoboard [Google]

4.3. LCD

El LCD 2X16, abreviatura de "Liquid Crystal Display" de 2 filas por 16 columnas, es un tipo de pantalla alfanumérica ampliamente utilizada en aplicaciones electrónicas. Está compuesto por un conjunto de cristales líquidos dispuestos en una matriz que puede mostrar hasta 32 caracteres alfanuméricos en un formato de dos líneas, cada una con 16 caracteres de ancho. Estos displays son populares debido a su bajo consumo de energía y su capacidad para mostrar información de manera clara y legible en una amplia gama de condiciones de iluminación.

El funcionamiento básico del LCD 2x16 se basa en la manipulación de la orientación de los cristales líquidos mediante la aplicación de voltajes específicos, lo que permite bloquear o permitir el paso de la luz a través de ellos. Controlado por un microcontrolador o un circuito integrado especializado, el LCD puede recibir datos y comandos para mostrar números, letras y otros caracteres de manera secuencial.

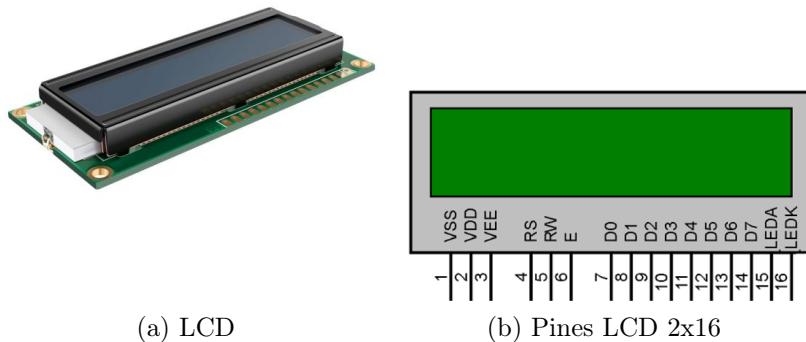


Figura 4: Pantalla LCD 2x16 (elaboración propia)

Siguiendo la información de la figura 5b, podemos armar la tabla 2 la cual nos ayudara a entender de mejor forma el funcionamiento de todos los pines de una pantalla LCD 2X16.

Sr. No	Pin No.	Nombre del PIN	Tipo de PIN	Descripción del PIN	Conexión de pines
1	Pin 1	Ground	Source Pin	Este es un pin de tierra de LCD.	Conectado a tierra del MCU/fuente de alimentación
2	Pin 2	VCC	Source Pin	Este es el pin de voltaje de suministro de la pantalla LCD.	Conectado al pin de suministro de la fuente de alimentación
3	Pin 3	V0/VEE	Control Pin	Ajusta el contraste de la pantalla LCD.	Conectado a un POT variable que puede generar 0-5V
4	Pin 4	Register Select	Control Pin	Alberna entre comando/registro de datos	Conectado a un pin del MCU y obtiene 0 1. 0 ->Modo de comando 1 ->Modo de datos
5	Pin 5	Read/Write	Control Pin	Alberna la pantalla LCD entre operación de lectura/escritura	Conectado a un pin MCU y obtiene 0 o 1. 0 ->Operación de escritura 1->Operación de lectura
6	Pin 6	Enable	Control Pin	Debe mantenerse alto para realizar la operación de lectura/escritura	Conectado al MCU y siempre en alto.
7	Pines 7-14	Data Bits (0-7)	Data/Command Pin	Pines utilizados para enviar comandos o datos a la pantalla LCD.	Pines utilizados para enviar comandos o datos a la pantalla LCD. En modo de 4 cables conectados. Solo 4 pines (0-3) están conectados al MCU. En modo de 8 cables conectados. Todos los 8 pines (0-7) están conectados al MCU.
8	Pin 15	LED positivo	LED Pin	Funcionamiento normal tipo LED para iluminar la pantalla LCD	Conectado a +5V
9	Pin 16	LED negativo	LED Pin	Funcionamiento normal tipo LED para iluminar la pantalla LCD conectada con GND.	Connected to ground

Tabla 2: Funcionamiento pines LCD [Elaboración propia]

4.4. Diodo LED

El diodo emisor de luz (LED por sus siglas en inglés) es un diodo que emite luz visible o invisible (infrarroja) cuando se energiza. Aun cuando la luz no es visible, los LED infrarrojos tienen numerosas aplicaciones donde la luz visible no es un efecto deseable. Éstas incluyen sistemas de seguridad, procesamiento industrial, acoplamiento óptico controles de seguridad como abrepuertas de cochera y centro de entretenimiento domésticos, donde la luz infrarroja del control remoto es el elemento de control. [4]

Color	Materiales	Tensión (V)
Azul	AlInGaP	2.1
Verde	GaN	5.0
Rojo	GaAsP	1.8
Blanco	GaAsP	4.1
Amarillo	AlInGaP	2.1

Tabla 3: Tabla de diodos LED's (elaboración propia)

En la tabla 3, se muestra distintos color de los diodos LED junto a sus materiales implementados en construcción y su respectiva tensión en polarización directa. En la figura 5 se muestra tanto el símbolo esquemático utilizado en diseño de circuitos electrónicos y un diodo LED físico.

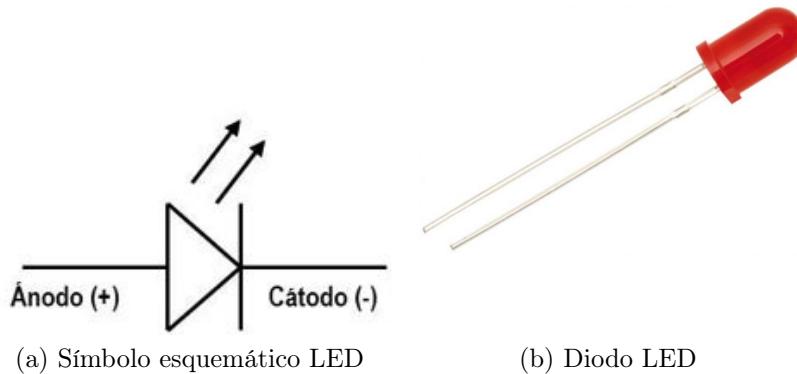


Figura 5: Diodo emisor de luz (elaboración propia)

4.5. Pulsador

Los botones son un tipo de interruptor donde permiten o detienen el flujo de corriente solo cuando están presionados. Existen 2 tipos, Normalmente Cerrado (NC) y Normalmente Abierto). Para el primer tipo, presionar el botón hará que el circuito se abra por un lapso de tiempo y luego vuelva a cerrar. Para el segundo tipo ocurre lo contrario, presionar el botón hará que se cierre el circuito y haya flujo de corriente, después de cierto tiempo se abrirá de nuevo [5]. Los interruptores en general rebotan cuando se abren o se cierran, por lo que se debe tomar en cuenta a la hora de construir el circuito. Una forma de resolverlo es mediante la agregación de un retraso en el código. Otra forma es mediante la adición de un capacitor, donde la capacitancia se calcula con la resistencia y la constante de tiempo deseada.

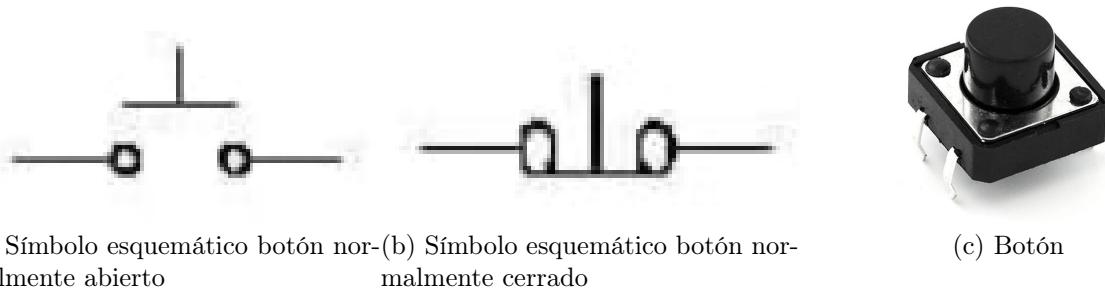


Figura 6: Botón (elaboración propia)

4.6. Resistor

El elemento de circuito que se usa para modelar el comportamiento de resistencia a la corriente de un material es el resistor. Para efectos de fabricación de circuitos, los resistores suelen hacerse de aleaciones metálicas y compuestos de carbono. El símbolo de circuito del resistor se presenta en la figura 7a, donde R significa la resistencia del resistor. El resistor es el elemento pasivo más simple. [2]

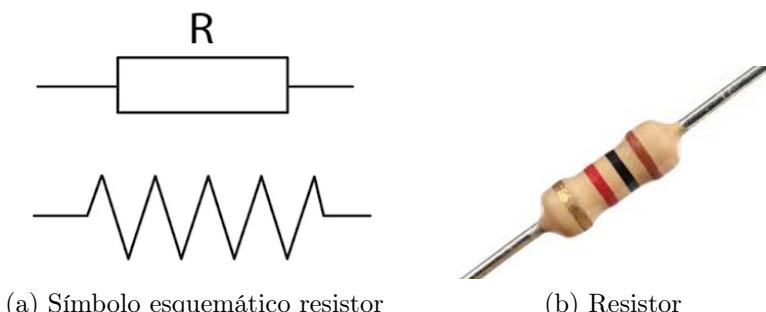


Figura 7: Resistor (elaboración propia)

4.7. Servomotor

Un servomotor es un motor con control, es decir este dispositivo mecánico posee internamente un controlador que posiciona precisamente el rotor del motor en un ángulo especificado por la señal entrante de control. El servomotor se compone de tres cables: GND, Power (5v) y Control. Es decir que los dos primeros se usan como alimentación en cuanto el cable de control es usado para enviar la señal de posicionamiento del motor con una secuencia de pulsos PWM.

Estos se pueden diseñar para obtener un giro continuo que podamos aprovechar, de la misma forma se puede diseñar para que se muevan un ángulo fijo en respuesta a una señal de control, y se mantengan fijos en esa posición. Estos servos o servomotores son muy frecuentes en aeromodelismo y en robótica, por la capacidad que presentan para moverse a un ángulo concreto y mantenerse allí. De hecho se suelen diseñar para que giren un ángulo proporcional a una señal PWM, de forma que su control es muy preciso.

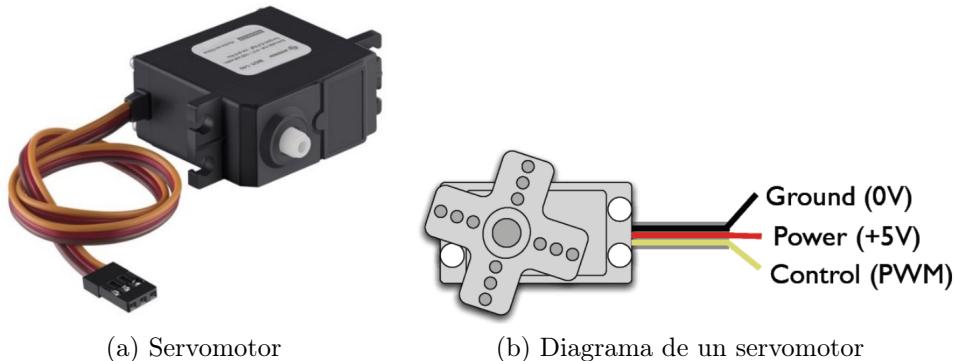


Figura 8: Servomotor (elaboración propia)

4.8. IoT

Internet of Things es un término ampliamente popular y utilizado en medios de comunicación, tanto digitales como analógicos. Los sensores electrónicos forman parte de una gran cantidad de dispositivos que han desarrollado inteligencia y se conectan a redes inalámbricas y a internet. Esto constituye una red global de conectividad total conocida como Internet de las cosas. Un ejemplo son los teléfonos inteligentes, que poseen sensores como giróscopos y acelerómetros que permiten su posicionamiento casi en tiempo real. Además, estos dispositivos electrónicos han disminuido su tamaño y costo, y pueden generar una gran cantidad de información, lo que convierte al mundo actual en un entorno conectado a objetos inteligentes [1].

Los protocolos de comunicación permiten el intercambio de información entre aplicaciones y dispositivos, y los sensores ubicados en cualquier lugar pueden ser accedidos por otros dispositivos y aplicaciones. En consecuencia, el Internet de las cosas constituye un sistema tecnológico en el que personas y objetos pueden conectarse a internet en cualquier momento y lugar para adquirir inteligencia y permitir que los objetos se comuniquen.

Sin duda, Internet y las telecomunicaciones han allanado el camino para esta interconexión masiva. Se proyectaba que para el año 2020 habría alrededor de cincuenta mil millones de dispositivos conectados en el mundo [1].

La definición de este concepto fundamental abarca cuatro componentes esenciales: la conexión, que se refiere a los protocolos de comunicación en estudio; las cosas, que incluyen actuadores, sensores y controladores, con una operación autónoma; y, finalmente, el internet, que implica la seguridad en la comunicación entre estos dispositivos interconectados [1].

4.9. ThingBoard

ThingsBoard es una plataforma de IoT de código abierto que facilita el rápido desarrollo de proyectos relacionados con esta tecnología. Proporciona acceso a una solución local o en la nube, permitiendo el uso de la infraestructura del servidor para las aplicaciones correspondientes [1].

La plataforma es compatible con protocolos estándar de IoT, como MQTT, CoAP y HTTP. Es capaz de ofrecer escalabilidad y tolerancia a fallos, y garantiza un rendimiento eficiente en la captura y procesamiento de datos del dispositivo. Esto se logra mediante un servidor de puerta de enlace que se encarga de la comunicación con los dispositivos conectados a la red, lo que asegura un funcionamiento óptimo y actualizaciones regulares [1].

5. Desarrollo

A partir de la propuesta inicial del dispensador de pastillas, se diseñó la lógica en base de los modos de uso propuestos como se observa en la figura 9. Los casos implementados en el código toman decisiones con respecto a la alarma.

5.1. Lógica Principal (`loop()`)

El bucle principal (`loop()`) inicia con la lectura del pulsador de modo. Dependiendo del modo actual, se toman decisiones específicas:

1. **Modo 0:** El programa se encuentra funcionando en este modo de manera predeterminada. No realiza cambios en el reloj ni en la configuración de la alarma. Es el modo predeterminado del sistema.
2. **Modo 1:** Incrementa horas.
3. **Modo 2:** Incrementa minutos.
4. **Modo 3:** Incrementa horas de la alarma.
5. **Modo 4:** Incrementa minutos de la alarma.
6. **Modo 5:** Alterna estado de la alarma (encendido/apagado).

Cada caso verifica el pulsador de incremento y ejecuta la función correspondiente al modo actual. Después de cada acción, se espera hasta que se suelte el pulsador.

Luego, se verifica si es necesario actualizar el tiempo (`actualizar == true`). Si es así, se llama a la función `actualizarReloj()`. Si la alarma está activada, se comprueba si se ha alcanzado la hora y minutos programados, realizando acciones específicas y activando el servo y el buzzer.

5.2. Funciones Adicionales

Se incluyen varias funciones adicionales que desempeñan roles específicos:

- `controladorTiempo`: Es llamada por el temporizador cada segundo para actualizar el tiempo.
- `actualizarReloj`: Actualiza las horas y minutos del reloj.
- `incrementarHoras`: Incrementa las horas y actualiza el reloj.
- `incrementarMinutos`: Incrementa los minutos y actualiza el reloj.
- `**incrementarHorasAlarma`: Incrementa las horas de la alarma.
- `incrementarMinutosAlarma`: Incrementa los minutos de la alarma.
- `girarServo`: Controla el servo para dispensar pastillas.

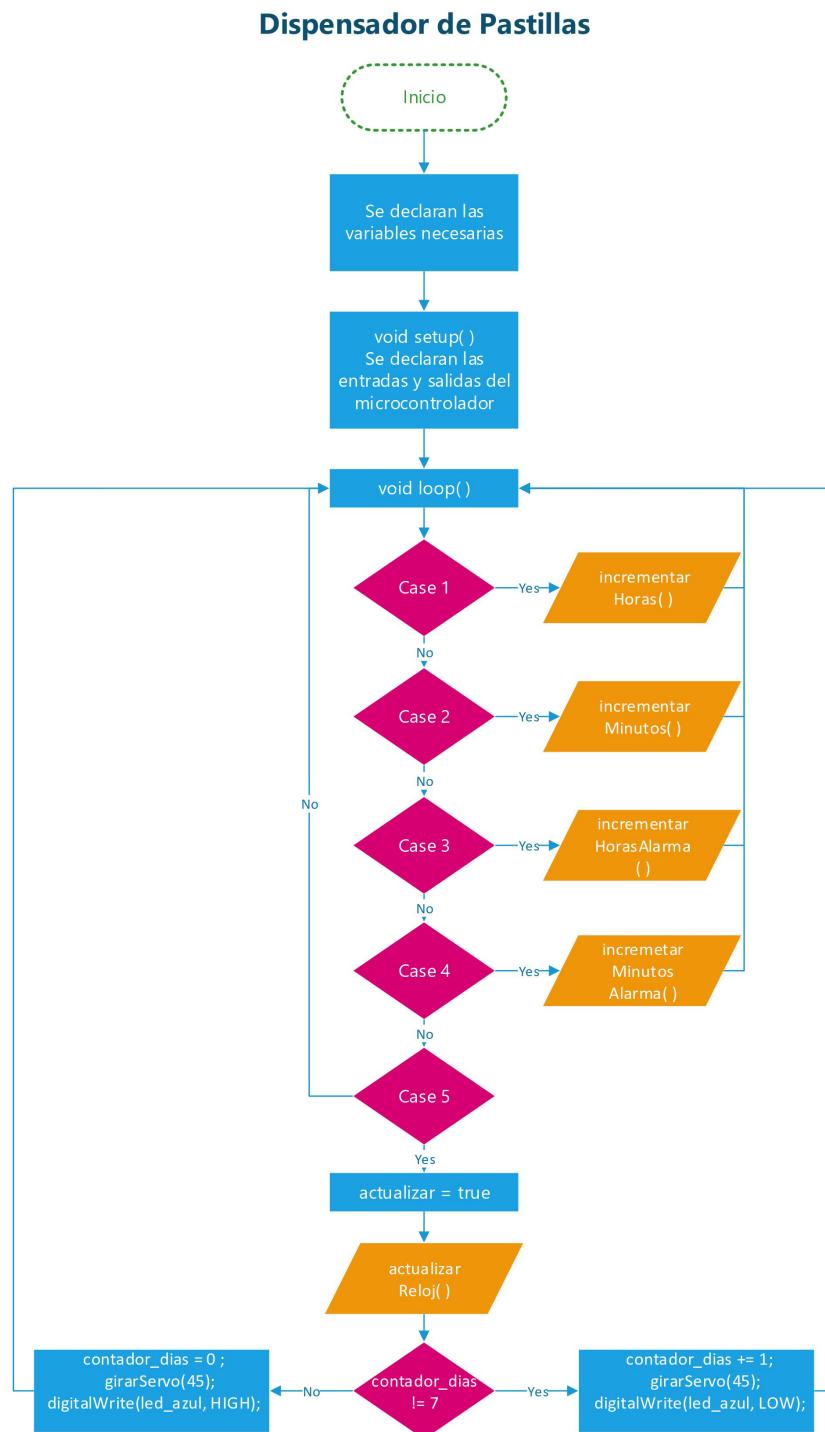


Figura 9: Diagrama de Flujo: Dispensador de Pastillas. Creación propia.

5.3. Internet de las Cosas

Coonectar el proyecto a internet facilita la toma y el seguimiento de medicamentos, tanto un cuidador como un paciente pueden tener acceso a estas alarmas y estar pendientes.

Cada vez que suena el buzzer a la hora en que se definió la alarma, un mensaje llegar diciendo que es hora de tomar la pastilla. Esto permite estar pendiente de la ingesta de medicamentos en todas partes.

5.4. Implementación del Circuito

En la figura 10 se puede observar la simulación del sistema a implementar.

5.4.1. Elementos del Circuito:

- Servomotor de 360 grados.
- Buzzer.
- LED.
- Botones.
- Pantalla LCD.

5.4.2. Conexiones:

- **Servomotor:** Tres cables, conectados al pin 11 (señal), ground (negativo), y 5V (positivo).
- **Buzzer:** Conexión a pin 2 (positivo) y ground (negativo).
- **LED:** Conexión al pin 10 (positivo) y ground (negativo).
- **Botones:** Alimentación de 5 V y GND, con conexiones a los pines 11 y 12 respectivamente.
- **Pantalla LCD:** Múltiples pines, del 9 al 3, con un pin adicional directamente conectado a 5V mediante una resistencia. Dos pines adicionales van a ground y positivo.

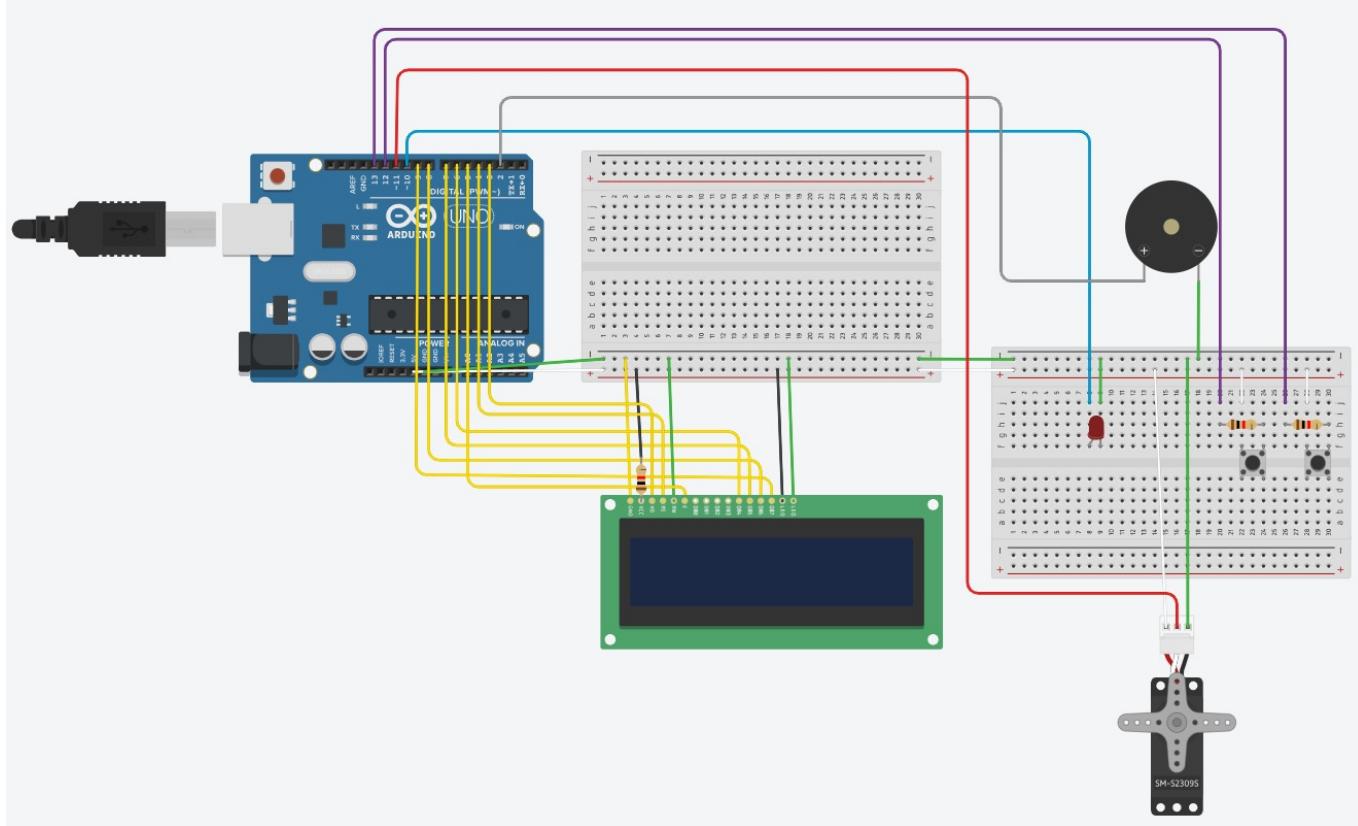


Figura 10: Simulación del Sistema. Creación propia

6. Análisis de resultados

En las figuras 11, 12, 13, 14 y 15 se puede observar el proceso de implementación del proyecto en el hardware.

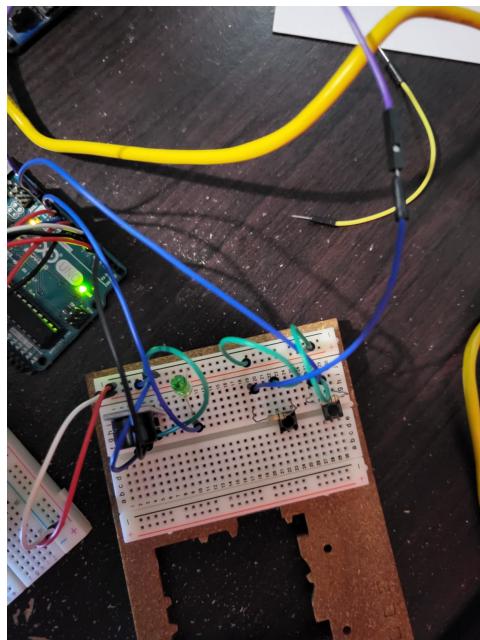


Figura 11: Ejemplo implementación I. Creación propia.

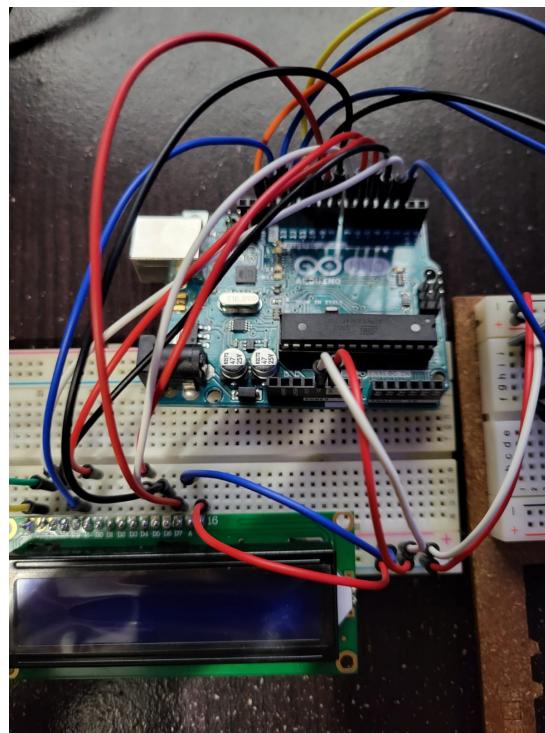


Figura 12: Ejemplo implementación II. Creación propia

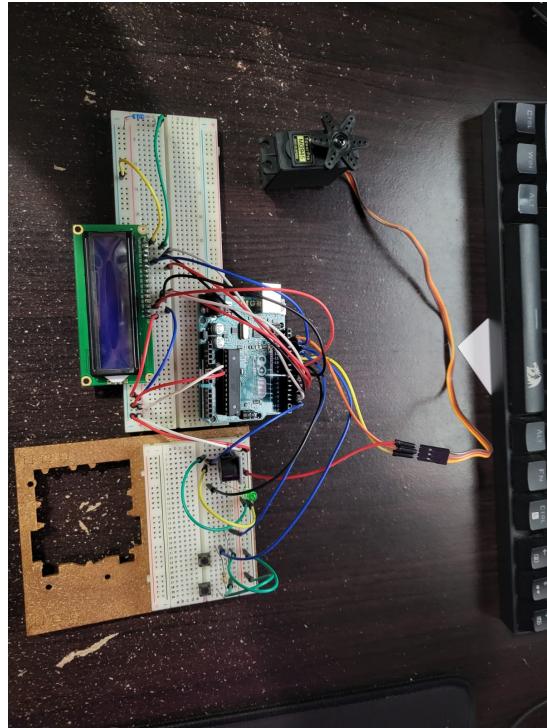


Figura 13: Ejemplo implementación III. Creación propia

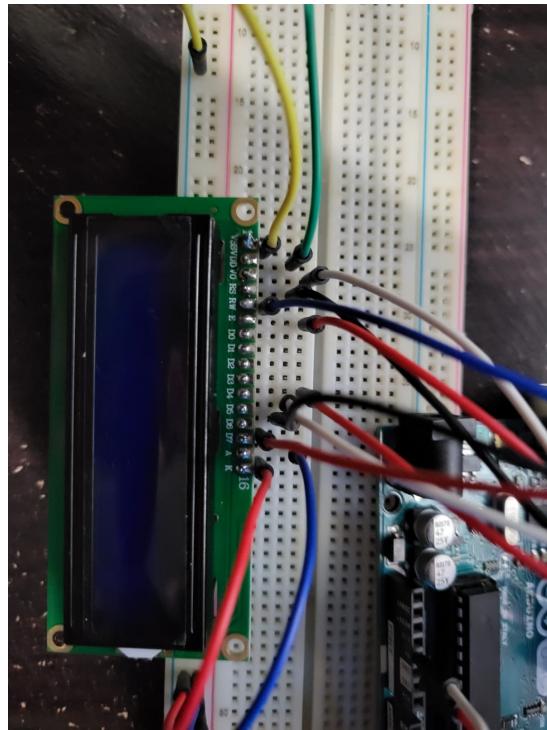


Figura 14: Ejemplo implementación IV. Creación propia

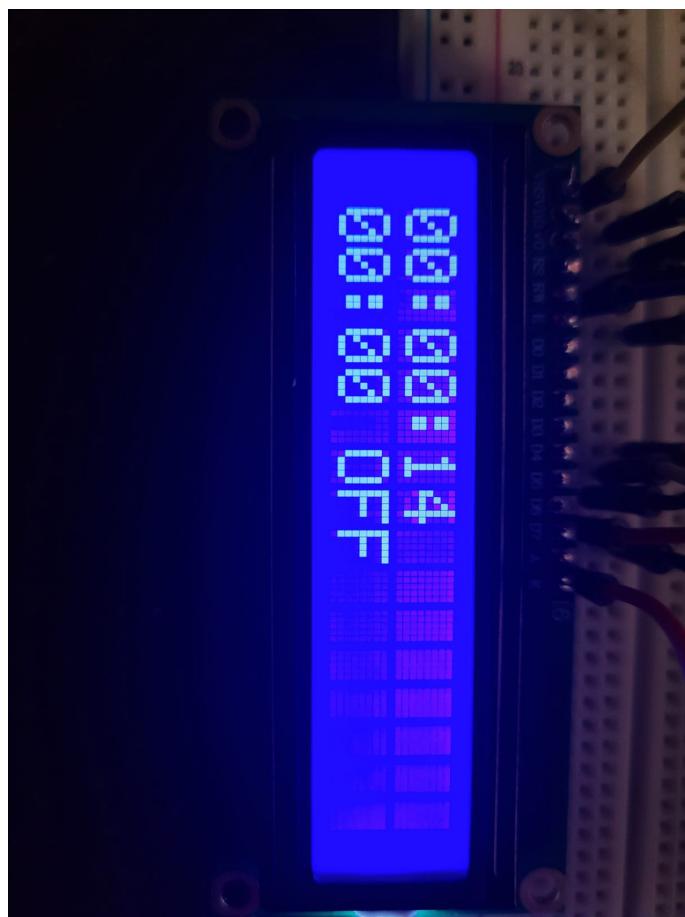


Figura 15: Ejemplo implementación: Pantalla LCD mostrando horas a usuario. Creación propia

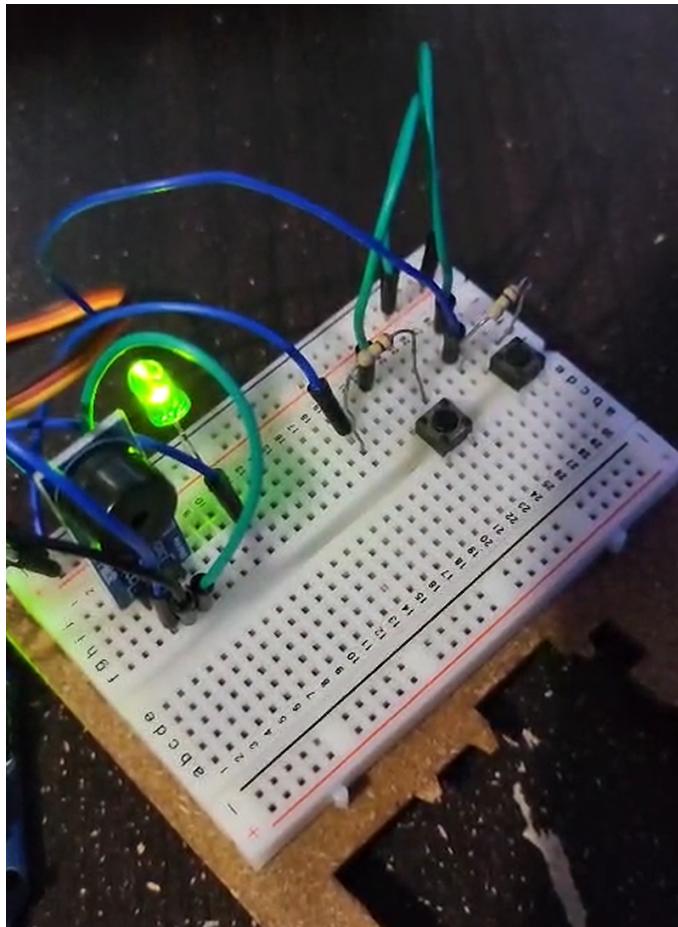


Figura 16: Ejemplo implementación: LED. Creación propia

Además de lo aquí mostrado, en el vídeo demostrativo se puede analizar todo el funcionamiento completo del dispensador de pastillas. Se quiso armar una maqueta para mejorar la demostración sin embargo los cables jumpers no lo permitían debido al largo de estos.

Link de repositorio: <https://github.com/joseeras98/Proyecto-IE0624.git>

Link de video : <https://www.dropbox.com/scl/fi/9a95s43f8y3lncmcyp7cr/Dispensador-.mp4?rlkey=sx365gpiioj52icsjlg71tz2w&dl=0>

7. Conclusiones, observaciones y recomendaciones

7.1. Conclusiones

1. **Dificultades en la adquisición de componentes:** La búsqueda y compra de servos y pantallas LCDs resultaron desafiantes, afectando la disponibilidad de estos elementos clave para el proyecto.
2. **Problemas con los botones:** Los botones presentaron mal funcionamiento, lo cual impactó negativamente en la experiencia de usuario y en la configuración de horas para la dispensación de pastillas.
3. **Desafíos en la implementación de IoT:** La conexión de la aplicación con el dispensador para la supervisión remota y los recordatorios a través de internet presentó obstáculos, afectando la capacidad del dispositivo para enviar mensajes de alarma.

7.2. Observaciones:

1. **Mejora en la adquisición de componentes:** Se recomienda buscar proveedores confiables y gestionar adecuadamente la obtención de los componentes necesarios para evitar demoras y problemas en el desarrollo.
2. **Evaluación y selección de botones de calidad:** La selección de botones de alta calidad es crucial para garantizar un funcionamiento adecuado y evitar problemas de conexión y respuesta.
3. **Revisión de la implementación de IoT:** Se sugiere una revisión profunda de la implementación de IoT, asegurando la correcta conexión entre el dispensador y la plataforma en línea para mejorar la supervisión y los recordatorios remotos.

7.3. Recomendaciones:

1. **Planificación y adquisición anticipada:** Planificar la adquisición de componentes con anticipación y explorar diversas fuentes confiables para garantizar la disponibilidad oportuna de todos los elementos necesarios.
2. **Pruebas exhaustivas de botones:** Realizar pruebas rigurosas de los botones durante el desarrollo para identificar y abordar cualquier problema de funcionamiento antes de la implementación final.
3. **Revisión de la implementación de IoT:** Trabajar en estrecha colaboración con el profesor para mejorar la conectividad y la comunicación remota del dispensador, asegurando una supervisión efectiva y recordatorios precisos.

Referencias

1. Luis Joyanes Aguilar. *Internet de las cosas un futuro hiperconectado: 5G, Inteligencia artificial, Big Data, Cloud, Blockchain, Ciberseguridad.* RA-MA Editorial, 2018.
2. Charles K Alexander and Sadiku Matthew N. O. *Fundamentos de circuitos electronicos.* The McGraw-Hill Companies, 2013.
3. Arduino. *Arduino® UNO R3 Product Reference Manual,* 2023. Modified: 01/12/2023. SKU: A000066.
4. Robert L Boylestad and Louis Nashelsky. *Electrónica: Teoría de circuitos y dispositivos electrónicos.* Pearson, 2009.
5. Shoptronica S.L. Que son los interruptores, pulsadores, commutadores. SHOPTRONICA.com.
6. Cristhian Caminlo Velandia LLanos and Andres Felipe Roncancio Rojas. *PROTOTIPO DISPENSADOR AUTOMÁTICO DE MEDICAMENTOS PARA PERSONAS DE LA TERCER EDAD.* PhD thesis, Universidad el Bosque, Mayo 2020.



Description

The Arduino Uno R3 is the perfect board to get familiar with electronics and coding. This versatile development board is equipped with the well-known ATmega328P and the ATMega 16U2 Processor. This board will give you a great first experience within the world of Arduino.

Target areas:

Maker, introduction, industries



Features

- **ATMega328P Processor**

- **Memory**

- AVR CPU at up to 16 MHz
 - 32KB Flash
 - 2KB SRAM
 - 1KB EEPROM

- **Security**

- Power On Reset (POR)
 - Brown Out Detection (BOD)

- **Peripherals**

- 2x 8-bit Timer/Counter with a dedicated period register and compare channels
 - 1x 16-bit Timer/Counter with a dedicated period register, input capture and compare channels
 - 1x USART with fractional baud rate generator and start-of-frame detection
 - 1x controller/peripheral Serial Peripheral Interface (SPI)
 - 1x Dual mode controller/peripheral I2C
 - 1x Analog Comparator (AC) with a scalable reference input
 - Watchdog Timer with separate on-chip oscillator
 - Six PWM channels
 - Interrupt and wake-up on pin change

- **ATMega16U2 Processor**

- 8-bit AVR® RISC-based microcontroller

- **Memory**

- 16 KB ISP Flash
 - 512B EEPROM
 - 512B SRAM
 - debugWIRE interface for on-chip debugging and programming

- **Power**

- 2.7-5.5 volts



CONTENTS

1 The Board	4
1.1 Application Examples	4
1.2 Related Products	4
2 Ratings	5
2.1 Recommended Operating Conditions	5
2.2 Power Consumption	5
3 Functional Overview	5
3.1 Board Topology	5
3.2 Processor	6
3.3 Power Tree	7
4 Board Operation	8
4.1 Getting Started - IDE	8
4.2 Getting Started - Arduino Web Editor	8
4.3 Sample Sketches	8
4.4 Online Resources	8
5 Connector Pinouts	9
5.1 JANALOG	10
5.2 JDIGITAL	10
5.3 Mechanical Information	11
5.4 Board Outline & Mounting Holes	11
6 Certifications	12
6.1 Declaration of Conformity CE DoC (EU)	12
6.2 Declaration of Conformity to EU RoHS & REACH 211 01/19/2021	12
6.3 Conflict Minerals Declaration	13
7 FCC Caution	13
8 Company Information	14
9 Reference Documentation	14
10 Revision History	14



1 The Board

1.1 Application Examples

The UNO board is the flagship product of Arduino. Regardless if you are new to the world of electronics or will use the UNO as a tool for education purposes or industry-related tasks, the UNO is likely to meet your needs.

First entry to electronics: If this is your first project within coding and electronics, get started with our most used and documented board; Arduino UNO. It is equipped with the well-known ATmega328P processor, 14 digital input/output pins, 6 analog inputs, USB connections, ICSP header and reset button. This board includes everything you will need for a great first experience with Arduino.

Industry-standard development board: Using the Arduino UNO R3 board in industries, there are a range of companies using the UNO board as the brain for their PLC's.

Education purposes: Although the UNO R3 board has been with us for about ten years, it is still widely used for various education purposes and scientific projects. The board's high standard and top quality performance makes it a great resource to capture real time from sensors and to trigger complex laboratory equipment to mention a few examples.

1.2 Related Products

- Starter Kit
- Arduino UNO R4 Minima
- Arduino UNO R4 WiFi
- Tinkerkit Braccio Robot



2 Ratings

2.1 Recommended Operating Conditions

Symbol	Description	Min	Max
	Conservative thermal limits for the whole board:	-40 °C (-40°F)	85 °C (185°F)

NOTE: In extreme temperatures, EEPROM, voltage regulator, and the crystal oscillator, might not work as expected.

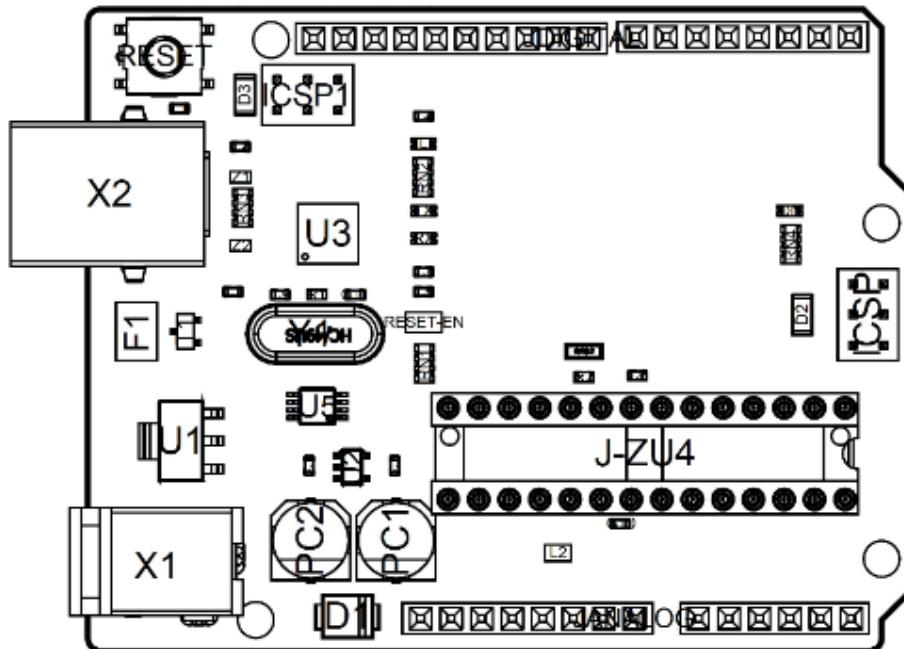
2.2 Power Consumption

Symbol	Description	Min	Typ	Max	Unit
VINMax	Maximum input voltage from VIN pad	6	-	20	V
VUSBMax	Maximum input voltage from USB connector		-	5.5	V
PMax	Maximum Power Consumption	-	-	xx	mA

3 Functional Overview

3.1 Board Topology

Top view



Board topology



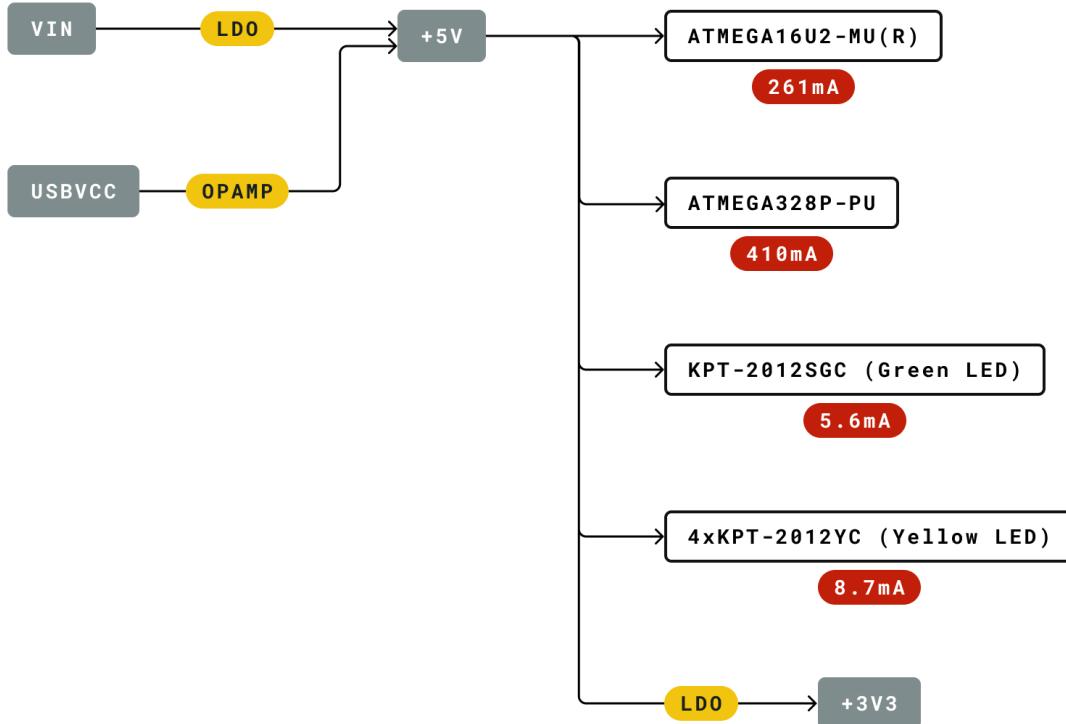
Ref.	Description	Ref.	Description
X1	Power jack 2.1x5.5mm	U1	SPX1117M3-L-5 Regulator
X2	USB B Connector	U3	ATMEGA16U2 Module
PC1	EEE-1EA470WP 25V SMD Capacitor	U5	LMV358LIST-A.9 IC
PC2	EEE-1EA470WP 25V SMD Capacitor	F1	Chip Capacitor, High Density
D1	CGRA4007-G Rectifier	ICSP	Pin header connector (through hole 6)
J-ZU4	ATMEGA328P Module	ICSP1	Pin header connector (through hole 6)
Y1	ECS-160-20-4X-DU Oscillator		

3.2 Processor

The Main Processor is a ATmega328P running at up to 20 MHz. Most of its pins are connected to the external headers, however some are reserved for internal communication with the USB Bridge coprocessor.



3.3 Power Tree



Legend:

- | | | |
|--|--|---|
| <input type="checkbox"/> Component | ● Power I/O | ● Conversion Type |
| ● Max Current | ● Voltage Range | |

Power tree



4 Board Operation

4.1 Getting Started – IDE

If you want to program your Arduino UNO R3 while offline you need to install the Arduino Desktop IDE [1] To connect the Arduino UNO to your computer, you'll need a USB-B cable. This also provides power to the board, as indicated by the LED.

4.2 Getting Started – Arduino Web Editor

All Arduino boards, including this one, work out-of-the-box on the Arduino Web Editor [2], by just installing a simple plugin.

The Arduino Web Editor is hosted online, therefore it will always be up-to-date with the latest features and support for all boards. Follow [3] to start coding on the browser and upload your sketches onto your board.

4.3 Sample Sketches

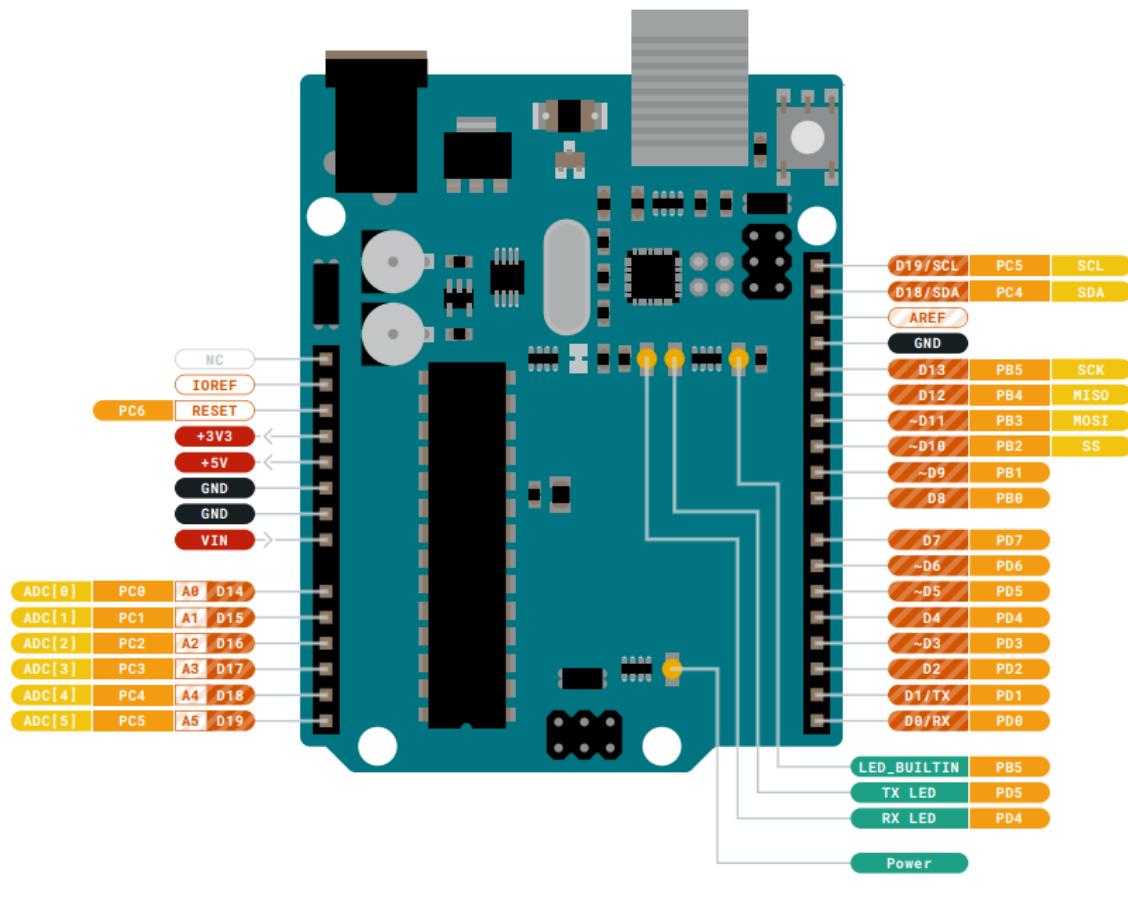
Sample sketches for the Arduino UNO R3 can be found either in the “Examples” menu in the Arduino IDE or in the “Documentation” section of the Arduino website [4]

4.4 Online Resources

Now that you have gone through the basics of what you can do with the board you can explore the endless possibilities it provides by checking exciting projects on Arduino Project Hub [5], the Arduino Library Reference [6] and the online Arduino store [7] where you will be able to complement your board with sensors, actuators and more.



5 Connector Pinouts



Pinout



5.1 JANALOG

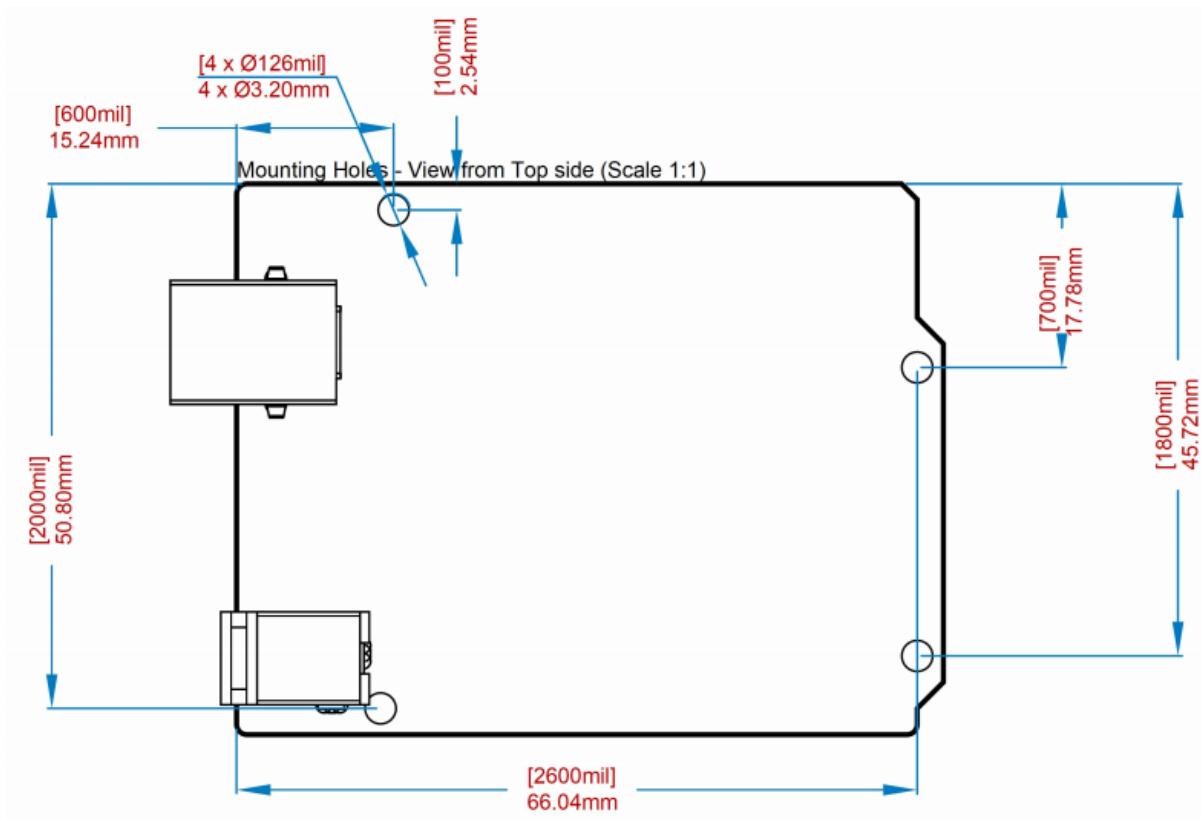
Pin	Function	Type	Description
1	NC	NC	Not connected
2	IOREF	IOREF	Reference for digital logic V - connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	+5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage Input
9	A0	Analog/GPIO	Analog input 0 /GPIO
10	A1	Analog/GPIO	Analog input 1 /GPIO
11	A2	Analog/GPIO	Analog input 2 /GPIO
12	A3	Analog/GPIO	Analog input 3 /GPIO
13	A4/SDA	Analog input/I2C	Analog input 4/I2C Data line
14	A5/SCL	Analog input/I2C	Analog input 5/I2C Clock line

5.2 JDIGITAL

Pin	Function	Type	Description
1	D0	Digital/GPIO	Digital pin 0/GPIO
2	D1	Digital/GPIO	Digital pin 1/GPIO
3	D2	Digital/GPIO	Digital pin 2/GPIO
4	D3	Digital/GPIO	Digital pin 3/GPIO
5	D4	Digital/GPIO	Digital pin 4/GPIO
6	D5	Digital/GPIO	Digital pin 5/GPIO
7	D6	Digital/GPIO	Digital pin 6/GPIO
8	D7	Digital/GPIO	Digital pin 7/GPIO
9	D8	Digital/GPIO	Digital pin 8/GPIO
10	D9	Digital/GPIO	Digital pin 9/GPIO
11	SS	Digital	SPI Chip Select
12	MOSI	Digital	SPI1 Main Out Secondary In
13	MISO	Digital	SPI Main In Secondary Out
14	SCK	Digital	SPI serial clock output
15	GND	Power	Ground
16	AREF	Digital	Analog reference voltage
17	A4/SD4	Digital	Analog input 4/I2C Data line (duplicated)
18	A5/SD5	Digital	Analog input 5/I2C Clock line (duplicated)

5.3 Mechanical Information

5.4 Board Outline & Mounting Holes





6 Certifications

6.1 Declaration of Conformity CE DoC (EU)

We declare under our sole responsibility that the products above are in conformity with the essential requirements of the following EU Directives and therefore qualify for free movement within markets comprising the European Union (EU) and European Economic Area (EEA).

ROHS 2 Directive 2011/65/EU	
Conforms to:	EN50581:2012
Directive 2014/35/EU. (LVD)	
Conforms to:	EN 60950-1:2006/A11:2009/A1:2010/A12:2011/AC:2011
Directive 2004/40/EC & 2008/46/EC & 2013/35/EU, EMF	
Conforms to:	EN 62311:2008

6.2 Declaration of Conformity to EU RoHS & REACH 211 01/19/2021

Arduino boards are in compliance with RoHS 2 Directive 2011/65/EU of the European Parliament and RoHS 3 Directive 2015/863/EU of the Council of 4 June 2015 on the restriction of the use of certain hazardous substances in electrical and electronic equipment.

Substance	Maximum limit (ppm)
Lead (Pb)	1000
Cadmium (Cd)	100
Mercury (Hg)	1000
Hexavalent Chromium (Cr6+)	1000
Poly Brominated Biphenyls (PBB)	1000
Poly Brominated Diphenyl ethers (PBDE)	1000
Bis(2-Ethylhexyl) phthalate (DEHP)	1000
Benzyl butyl phthalate (BBP)	1000
Dibutyl phthalate (DBP)	1000
Diisobutyl phthalate (DIBP)	1000

Exemptions: No exemptions are claimed.

Arduino Boards are fully compliant with the related requirements of European Union Regulation (EC) 1907 /2006 concerning the Registration, Evaluation, Authorization and Restriction of Chemicals (REACH). We declare none of the SVHCs (<https://echa.europa.eu/web/guest/candidate-list-table>), the Candidate List of Substances of Very High Concern for authorization currently released by ECHA, is present in all products (and also package) in quantities totaling in a concentration equal or above 0.1%. To the best of our knowledge, we also declare that our products do not contain any of the substances listed on the "Authorization List" (Annex XIV of the REACH regulations) and Substances of Very High Concern (SVHC) in any significant amounts as specified by the Annex XVII of Candidate list published by ECHA (European Chemical Agency) 1907 /2006/EC.



6.3 Conflict Minerals Declaration

As a global supplier of electronic and electrical components, Arduino is aware of our obligations with regards to laws and regulations regarding Conflict Minerals, specifically the Dodd-Frank Wall Street Reform and Consumer Protection Act, Section 1502. Arduino does not directly source or process conflict minerals such as Tin, Tantalum, Tungsten, or Gold. Conflict minerals are contained in our products in the form of solder, or as a component in metal alloys. As part of our reasonable due diligence Arduino has contacted component suppliers within our supply chain to verify their continued compliance with the regulations. Based on the information received thus far we declare that our products contain Conflict Minerals sourced from conflict-free areas.

7 FCC Caution

Any Changes or modifications not expressly approved by the party responsible for compliance could void the user's authority to operate the equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions:

- (1) This device may not cause harmful interference
- (2) this device must accept any interference received, including interference that may cause undesired operation.

FCC RF Radiation Exposure Statement:

1. This Transmitter must not be co-located or operating in conjunction with any other antenna or transmitter.
2. This equipment complies with RF radiation exposure limits set forth for an uncontrolled environment.
3. This equipment should be installed and operated with minimum distance 20cm between the radiator & your body.

English: User manuals for license-exempt radio apparatus shall contain the following or equivalent notice in a conspicuous location in the user manual or alternatively on the device or both. This device complies with Industry Canada license-exempt RSS standard(s). Operation is subject to the following two conditions:

- (1) this device may not cause interference
- (2) this device must accept any interference, including interference that may cause undesired operation of the device.

French: Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes :

- (1) l'appareil n'effectue pas de brouillage
- (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

IC SAR Warning:

English This equipment should be installed and operated with minimum distance 20 cm between the radiator and your body.



French: Lors de l' installation et de l' exploitation de ce dispositif, la distance entre le radiateur et le corps est d'au moins 20 cm.

Important: The operating temperature of the EUT can't exceed 85°C and shouldn't be lower than -40°C.

Hereby, Arduino S.r.l. declares that this product is in compliance with essential requirements and other relevant provisions of Directive 2014/53/EU. This product is allowed to be used in all EU member states.

8 Company Information

Company name	Arduino S.r.l
Company Address	Via Andrea Appiani 25 20900 MONZA Italy

9 Reference Documentation

Reference	Link
Arduino IDE (Desktop)	https://www.arduino.cc/en/Main/Software
Arduino IDE (Cloud)	https://create.arduino.cc/editor
Cloud IDE Getting Started	https://create.arduino.cc/projecthub/Arduino_Genuino/getting-started-with-arduino-web-editor-4b3e4a
Arduino Website	https://www.arduino.cc/
Project Hub	https://create.arduino.cc/projecthub?by=part&part_id=11332&sort=trending
Library Reference	https://www.arduino.cc/reference/en/
Online Store	https://store.arduino.cc/

10 Revision History

Date	Revision	Changes
26/07/2023	2	General Update
06/2021	1	Datasheet release