

Trabajo Practico N°2: Fundamentos de Programación IoT y Módulos de Desarrollo

Investigación de las plataformas de desarrollo

Arduino



Arduino es una plataforma de hardware de código abierto que consiste en placas de circuito impreso con microcontroladores y un entorno de desarrollo integrado (IDE).

Las placas de Arduino están diseñadas para ser fáciles de usar, incluso para personas sin experiencia en electrónica o programación

Una de las principales características de Arduino es su comunidad activa y su amplio ecosistema de hardware y software. Existen numerosas placas Arduino con diferentes especificaciones y características, lo que permite a los usuarios elegir la que mejor se adapte a sus necesidades.

PlatformIO

PlatformIO es un entorno de desarrollo integrado (IDE) y una plataforma de código abierto para el desarrollo de software embebido. Se utiliza principalmente para programar microcontroladores y microprocesadores en proyectos de electrónica y sistemas embebidos. PlatformIO proporciona un flujo de trabajo unificado para el

desarrollo de firmware, que incluye la creación, compilación, carga y depuración de código para una amplia gama de plataformas de hardware.

Arduino con PlatformIO:

Sintaxis:

- Utiliza un lenguaje de programación basado en C/C++.
- Sintaxis simple y fácil de aprender, especialmente para principiantes en programación.

Entorno de desarrollo:

- Plataforma de desarrollo integrada (IDE) propia, que proporciona un editor de código, compilador y herramientas de carga.
- PlatformIO extiende las capacidades de desarrollo de Arduino integrándose con entornos de desarrollo populares como Visual Studio Code, Atom, y Eclipse, ofreciendo características avanzadas como depuración, gestión de bibliotecas mejorada, y soporte multiplataforma.

Ventajas:

- Amplia compatibilidad con hardware, lo que permite una gran variedad de aplicaciones en IoT.
- Comunidad activa y abundante documentación disponible.
- Ideal para proyectos simples a medianamente complejos en IoT.

Desventajas:

- Aunque PlatformIO mejora la gestión de bibliotecas, el ecosistema de Arduino puede ser menos estructurado y un poco más difícil de navegar en comparación con entornos de desarrollo más avanzados.
- Limitaciones en la gestión de recursos y multitarea en comparación con sistemas operativos completos como RTOS.

Aplicaciones en IoT:

- Control de sensores y actuadores.
- Monitorización ambiental.
- Automatización del hogar.
- Sistemas de alarma y seguridad.

MicroPython con RT-Thread

MicroPython es una implementación del lenguaje de programación Python 3, optimizado para microcontroladores y sistemas embebidos. Permite a los desarrolladores escribir programas en Python y ejecutarlos directamente en dispositivos con recursos limitados, como microcontroladores de bajo costo y placas de desarrollo.

RT-Thread es un sistema operativo en tiempo real (RTOS) de código abierto diseñado para sistemas embebidos y dispositivos IoT (Internet de las cosas). Por otro lado, MicroPython es un lenguaje de programación Python optimizado para microcontroladores y sistemas embebidos

Sintaxis:

- Utiliza Python, un lenguaje de programación de alto nivel y fácil de aprender.
- La sintaxis es clara, legible y concisa, lo que facilita la programación y la depuración.

Entorno de desarrollo:

- MicroPython generalmente se programa en un entorno de desarrollo integrado (IDE) compatible, como Thonny o Visual Studio Code con extensiones adecuadas.
- RT-Thread es un sistema operativo en tiempo real que proporciona un entorno de desarrollo completo con capacidades de multitarea, gestión de recursos y comunicaciones.

Ventajas:

- Facilidad de aprendizaje debido a la sintaxis de Python, ideal para principiantes y programadores no especializados.
- Eficiencia de recursos y soporte para multitarea en RT-Thread, lo que permite aplicaciones más complejas en dispositivos de recursos limitados.
- Flexibilidad para desarrollar prototipos rápidos y aplicaciones interactivas.

Desventajas:

- Menor rendimiento en comparación con lenguajes de bajo nivel como C/C++ en ciertos casos de uso intensivo de recursos.
- Menos bibliotecas y recursos disponibles en comparación con el ecosistema de Arduino.

Aplicaciones en IoT:

- Control remoto y monitoreo de dispositivos.
- Sensores y recolección de datos.

- Automatización industrial.
- Dispositivos portátiles y wearables.

ESP32

El ESP32 es un microcontrolador de bajo costo y alto rendimiento desarrollado por Espressif Systems. Es una versión mejorada del ESP8266, con características adicionales y capacidades mejoradas. El ESP32 es ampliamente utilizado en una variedad de aplicaciones, incluyendo el Internet de las cosas (IoT), sistemas embebidos, dispositivos conectados, automatización del hogar, y más.

Los módulos "shield" para ESP32 son placas de expansión diseñadas para complementar y extender las capacidades del ESP32, un microcontrolador de bajo costo y alto rendimiento popularmente utilizado en proyectos de IoT. Estos shields proporcionan funcionalidades adicionales, como conectividad Wi-Fi, Bluetooth, sensores, actuadores y más.

Aquí hay algunas características clave del ESP32:

1. **Procesador Dual-Core:** El ESP32 incluye dos núcleos de procesamiento Xtensa LX6 de 32 bits, lo que proporciona un rendimiento mejorado y la capacidad de ejecutar múltiples tareas simultáneamente.
2. **Conectividad Inalámbrica:** Es compatible con Wi-Fi 802.11 b/g/n/e/i, así como con Bluetooth v4.2 y v5.0. Esto permite la conexión a redes Wi-Fi y la comunicación inalámbrica con otros dispositivos, lo que lo hace ideal para aplicaciones de IoT.
3. **Bajo Consumo de Energía:** El ESP32 está diseñado para ser altamente eficiente en términos de consumo de energía, lo que lo hace adecuado para dispositivos alimentados por batería y otras aplicaciones donde se requiere una larga vida útil de la batería.
4. **Interfaces Periféricas Abundantes:** El ESP32 incluye una amplia variedad de interfaces periféricas, incluyendo puertos UART, SPI, I2C, ADC, DAC, y más, lo que permite la conexión a una amplia gama de sensores, actuadores y otros dispositivos externos.
5. **Amplia Comunidad y Soporte:** El ESP32 cuenta con una comunidad activa de desarrolladores y una amplia variedad de recursos disponibles en línea, incluyendo documentación oficial, bibliotecas de código abierto, tutoriales y ejemplos de proyectos.