



INSTITUTO TECNOLÓGICO DE CHIHUAHUA II

INGENIERÍA EN SISTEMAS COMPUTACIONALES

SISTEMAS PROGRAMABLES

SEMESTRE 6: ENERO-JUNIO



UNIDAD V: PUERTOS Y BUSES DE UN MICROCONTROLADOR

INVESTIGACIÓN

INTEGRANTES:

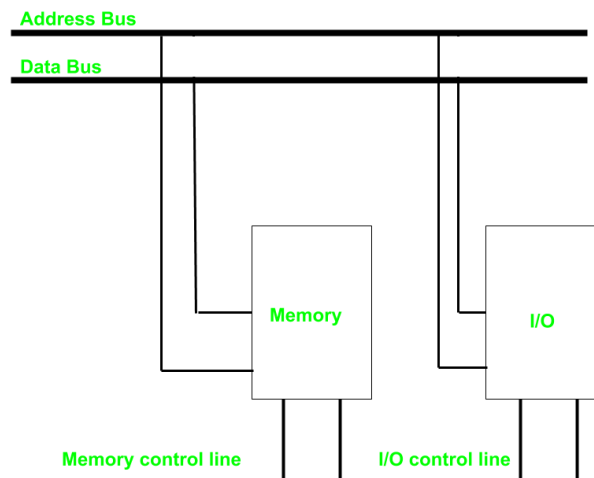
| | |
|-----------------------------------|----------|
| ROCÍO VANESA GARDEA HERNÁNDEZ | 21550330 |
| HÉCTOR ALEJANDRO RODRIGUEZ BARRÓN | 21550353 |
| JOSÉ SEBASTIAN LÓPEZ IBARRA | 21550362 |
| ANDRÉS SAÉNZ OLIVAS | 21550390 |
| JORGE EDUARDO ESCOBAR BUGARINI | 21550317 |
| EDGAR GERARDO DELGADO CERRILLO | 21550297 |

1. ¿Cuáles son los puertos de entrada/salida aislada?

Son los puertos que tienen asignado un espacio de direcciones separado del espacio de direcciones de los registros externos que componen la memoria.

Para que un microprocesador pueda implementar el modo E/S aislada son indispensables las siguientes condiciones:

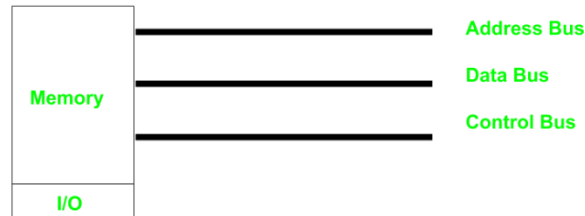
- El microprocesador debe proporcionar señales de control que permitan distinguir entre una operación con un puerto y una referencia a memoria.
- El código de instrucciones debe tener instrucciones especiales con las que se pueda leer (entrada) o escribir (salida) en los puertos.



2. ¿Cuáles son los puertos de entrada/salida mapeada?

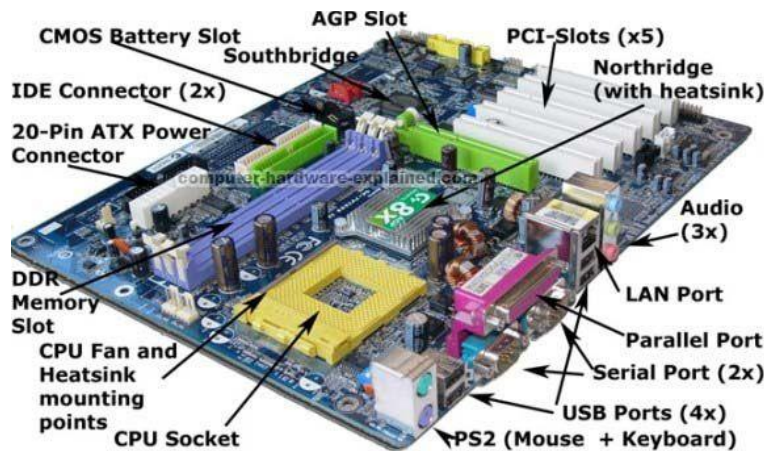
El modo de E/S mapeada a memoria (Memory mapped I/O) se basa en que tanto las localidades de memoria como los puertos de E/S se consideran como registros externos desde el punto de vista del microprocesador. Entonces, las instrucciones que hacen referencia a la memoria también pueden transferir datos entre un dispositivo periférico y el microprocesador, siempre y cuando el puerto de E/S que los interconecta se encuentre dentro del espacio de direccionamiento de memoria, es decir, controlado por las señales de control para memoria. De esta forma, el

registro asociado con el puerto de E/S es tratado simplemente como una localidad de memoria más.



3. ¿Cuáles son los puertos de un microprocesador genérico?

- Puertos de comunicación serie (UART/USART).
- Puertos de comunicación paralela (GPIO).
- Puertos de comunicación SPI (Serial Peripheral Interface).
- Puertos de comunicación I2C (Inter-Integrated Circuit).
- Puertos de comunicación PCIe (Peripheral Component Interconnect Express).

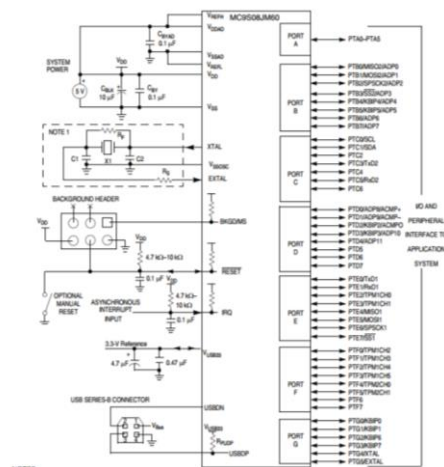


4. ¿Cuáles son los puertos de un microcontrolador genérico?

- **Registro de datos (PTxD):** se muestra la configuración general de un registro de datos de un puerto x.
- **PTxDn:** Establece el estado de cada pin I/O del correspondiente puerto. Al salir del estado de RESET, los pins son configurados como entradas en alta

impedancia pull-ups deshabilitados. Los pins de salida se actualizan al valor de este registro, una vez se haya definido su dirección.

- **Registro de dirección de pin (PTxDD):** muestra la configuración general de un registro de dirección de pins de un puerto x.
- **PTxDDn:** establece la dirección de cada pin I/O del correspondiente puerto.
- **Registro de habilitación de pull-ups (PTxPE):** se muestra la configuración general de un registro de habilitación de pull-ups para los pins de entrada de un puerto x.
- **PTxPEn:** establece una resistencia de pull-up en el pin de entrada indicado.
- **Registro de habilitación del slew rate (PTxSE):** se muestra la configuración general de un registro de habilitación del slew rate para los pins de salida de un puerto x.
- **PTxSEn:** limita la tasa de ascenso de la señal eléctrica de un pin de salida. El efecto es blindar el pin contra fenómenos del tipo EMI (ElectroMagnetic Interferences, Interferencia ElectroMagnetica).
- **Registro de habilitación del drive strength (PTxDS):** se muestra la configuración general de un registro de habilitación del drive strength para los pins de salida de un puerto x.
- **PTxDSn:** habilita el drenaje o suministro de altas corrientes para un pin de salida. El efecto sobre la EMC (ElectroMagnetic Compliance, Compatibilidad ElectroMagnetica, contrario a la EMI) es negativo y se debe ser cuidadoso en su manejo.

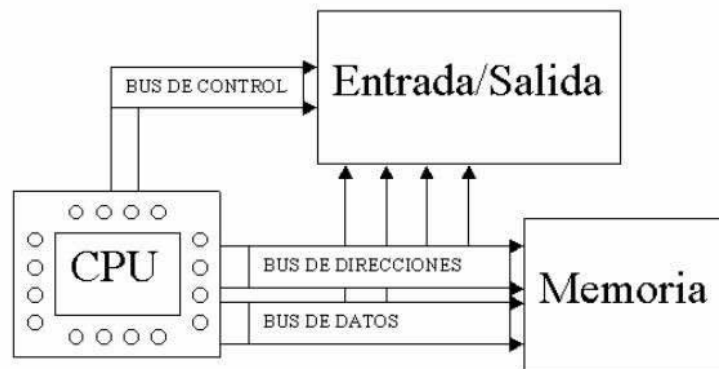


5. ¿Cuáles son los buses incluidos en un microprocesador genérico?

Bus de datos: es el camino por el cual se transfieren los datos entre el microprocesador y la memoria, así como entre el microprocesador y otros dispositivos periféricos. La cantidad de líneas en el bus de datos determina la cantidad de bits que se pueden transferir simultáneamente.

Bus de direcciones: este bus lleva la información de dirección desde el microprocesador hasta la memoria o los dispositivos periféricos. Indica a qué ubicación de memoria se está accediendo o desde dónde se están leyendo o escribiendo datos.

Bus de control: este bus lleva señales de control que determinan las operaciones que se están realizando en el microprocesador, como leer datos de la memoria, escribir datos en la memoria, enviar datos a los periféricos, etc.



6. ¿Cuáles son los buses de un microcontrolador genérico?

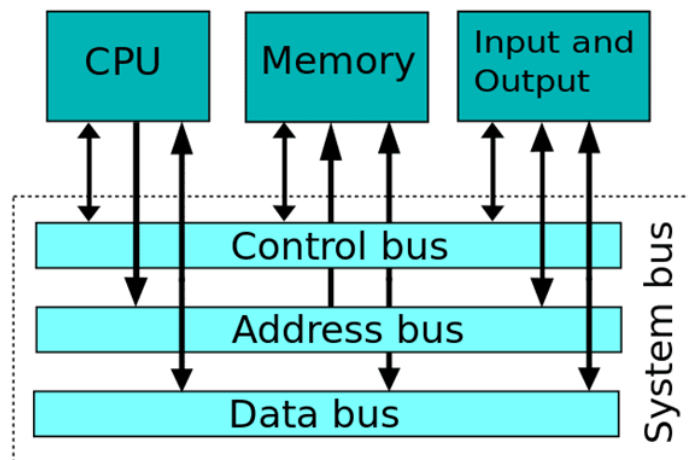
Bus de datos: similar al microprocesador, el bus de datos en un microcontrolador es donde se transfieren los datos entre el microcontrolador y la memoria, así como entre el microcontrolador y los periféricos externos. La anchura del bus de datos determina cuántos bits de datos pueden transferirse simultáneamente.

Bus de direcciones: al igual que en el microprocesador, el bus de direcciones en un microcontrolador se utiliza para especificar la ubicación de memoria o de los registros de los periféricos a los que se está accediendo.

Bus de control: este bus lleva señales de control que determinan las operaciones que se están realizando en el microcontrolador, como la lectura o escritura de datos en la memoria, la activación de periféricos, la gestión de interrupciones, etc.

Bus de expansión: en una primera aproximación creamos un bus de expansión específicamente creado para el microcontrolador. Esto tiene importantes desventajas, entre ellas:

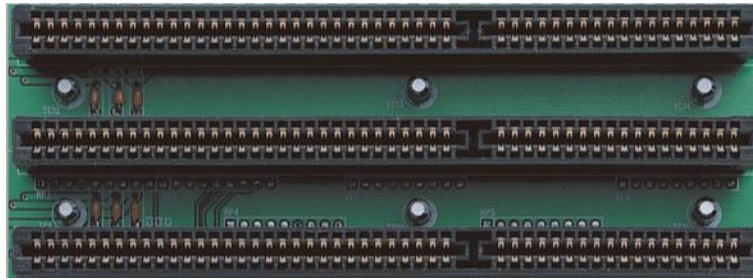
- Para conectar periféricos creados por otros grupos de trabajo es necesario adaptarlos a la señalización del bus en cuestión.
- Periféricos diseñados para ese bus no servían para ser usados en otros diseños con buses de otro tamaño. Por ejemplo: no servían para un bus de 16 o 32 bits sin ser adaptados.



7. ¿A que se refiere cuando se menciona “bus de expansión”?

En una primera aproximación creamos un bus de expansión específicamente creado para el microcontrolador. Esto tiene importantes desventajas, entre ellas:

- Para conectar periféricos creados por otros grupos de trabajo es necesario adaptarlos a la señalización del bus en cuestión.
- Periféricos diseñados para ese bus no servían para ser usados en otros diseños con buses de otro tamaño. Por ejemplo: no servían para un bus de 16 o 32 bits sin ser adaptados.

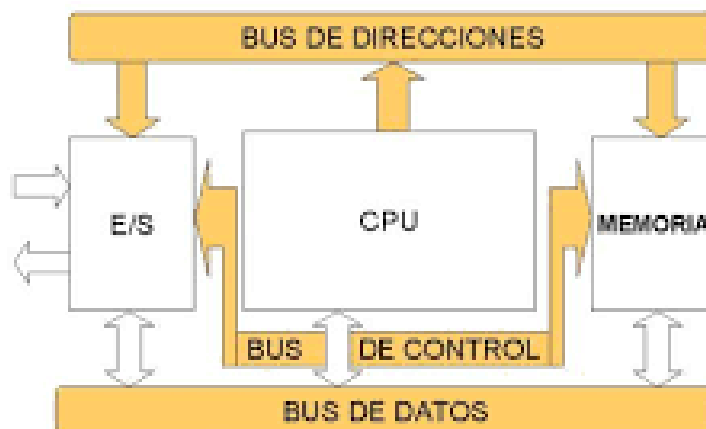


8. ¿Cuáles son las principales aplicaciones de los buses?

Bus de direcciones: la CPU sola puede colocar niveles lógicos en las n líneas de dirección, con la cual se genera 2^n posibles direcciones diferentes.

Bus de datos: en algunos microprocesadores, el bus de datos se usa para transmitir otra información además de los datos (por ejemplo, bits de dirección o información de condiciones). Es decir, el bus de datos es compartido en el tiempo o multiplexado.

Bus de control: este conjunto de señales se usa para sincronizar las actividades y transacciones con los periféricos del sistema. Algunas de estas señales, como R/W, son señales que la CPU envía para indicar que tipo de operación se espera en ese momento. Los periféricos también pueden remitir señales de control a la CPU, como son INT, RESET, BUS RQ.



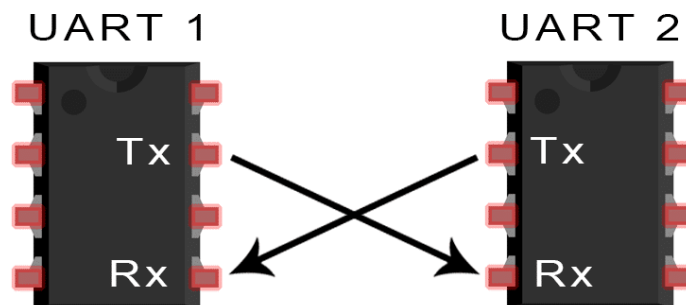
9. ¿Cuáles son las principales aplicaciones de los puertos de E/S?

Estos son los que se utilizan para conectar generalmente periféricos externos como pueden ser teclados, impresoras, ratones, auriculares, su uso es por tanto de propósito general.



10. ¿Qué es y para que se utiliza un “UART”?

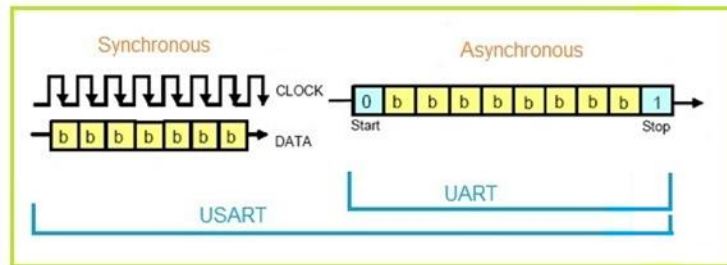
Un UART es un tipo de circuito integrado que se usa para enviar y recibir datos a través de un puerto serie en un equipo o dispositivo periférico. Los UART son ampliamente utilizados y conocidos por su sencillez. Sin embargo, a diferencia de SPI e I2C, los UART no admiten múltiples dispositivos subordinados.



11. ¿Qué es y para que se utiliza un “USART”?

USART (Universal Synchronous/Asynchronous Receiver/Transmitter) es una interfaz de comunicación serie utilizada para la transmisión y recepción de datos en sistemas electrónicos. Es una de las interfaces de comunicación más comunes en

la electrónica, y se utiliza en una amplia variedad de dispositivos, desde microcontroladores hasta módems.



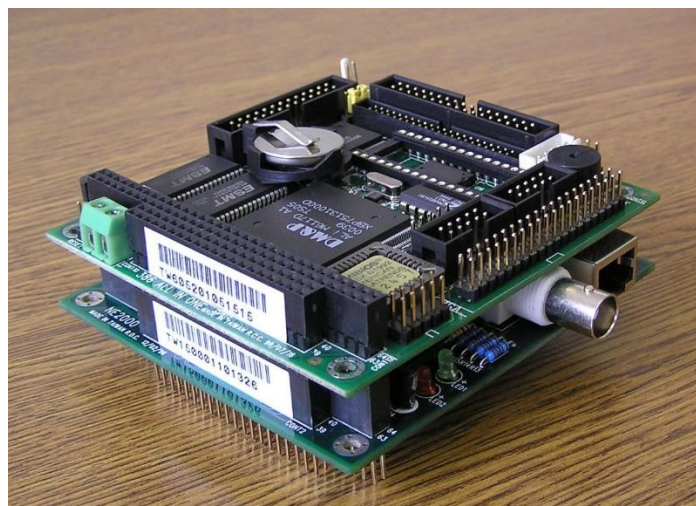
12. ¿Dónde se debe utilizar un compilador?

Los compiladores se utilizan para crear programas que son más eficientes y rápidos que los programas escritos en lenguaje de máquina directamente.

Además, permiten a los programadores escribir programas en lenguajes de alto nivel más legibles y mantenibles, en lugar de tener que trabajar con el código de máquina directamente.

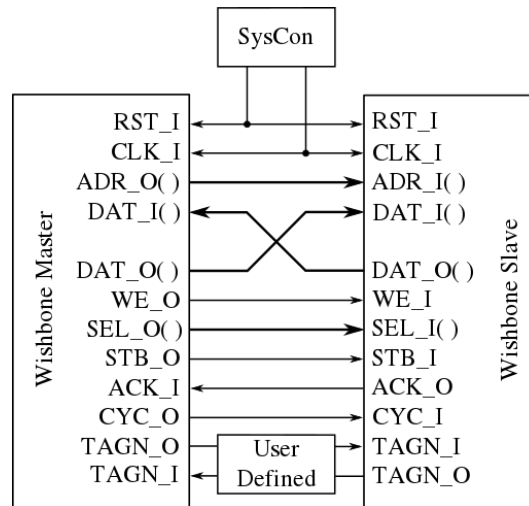
13. ¿Cuáles son los estándares manejados por los buses?

Más de 100 fabricantes de tarjetas respaldan actualmente el estándar PC/104 y ofrecen una solución para casi cada aplicación. En comparación con otras soluciones, el estándar PC/104 es un estándar real y de ese modo permite el intercambio seguro de tarjetas de diferentes fabricantes, asegurando una disponibilidad a largo tiempo.



14. ¿A que se refiere el estándar de interconexión “wishbone”?

Wishbone es una interfaz de bus abierta y libremente disponible diseñada para sistemas integrados. Fue desarrollada originalmente por Silicore Corporation y más tarde se liberó como parte del proyecto OpenCores. La interfaz Wishbone es muy utilizada en la industria y en proyectos de código abierto debido a su simplicidad, flexibilidad y modularidad.



15. Describe cuales son los puertos del Arduino Uno y cuál es la función que desempeña cada uno.

Puerto de alimentación (Power Jack): Este es el puerto donde se conecta la fuente de alimentación externa, que puede ser un adaptador de corriente de 9V o una batería.

Puerto USB: Este puerto se utiliza para conectar el Arduino Uno a la computadora. Se puede utilizar para cargar programas en el Arduino y para la comunicación serie con la computadora.

Pines de E/S digitales (Digital I/O Pins): El Arduino Uno tiene 14 pines digitales que se pueden configurar como entradas o salidas. Estos pines se pueden utilizar para conectar interruptores, LEDs, motores, y otros dispositivos digitales.

Pines de E/S analógicas (Analog Input Pins): El Arduino Uno tiene 6 pines analógicos que se pueden utilizar para medir voltajes analógicos. Estos pines se

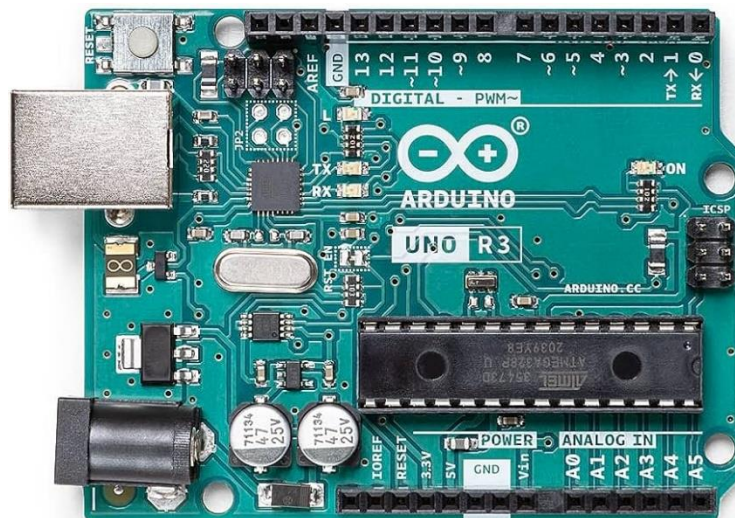
pueden utilizar para leer sensores analógicos como potenciómetros, fotocélulas y termistores.

Puertos de comunicación serial: El Arduino Uno tiene varios puertos de comunicación serie que se pueden utilizar para comunicarse con otros dispositivos, como el puerto UART (Universal Asynchronous Receiver/Transmitter) que se utiliza para comunicarse con dispositivos serie como sensores y pantallas.

Puerto de alimentación 5V: Este puerto proporciona una salida de 5V que se puede utilizar para alimentar circuitos externos.

Puerto de alimentación 3.3V: Este puerto proporciona una salida de 3.3V que se puede utilizar para alimentar dispositivos que funcionan a ese voltaje.

Puertos GND (Ground): El Arduino Uno tiene varios pines GND que se utilizan como referencia de tierra para circuitos externos.



16. Describe cuales son los puertos del ESP32 y cuál es la función que desempeña cada uno.

Puerto USB: Similar al Arduino Uno, el ESP32 tiene un puerto USB que se utiliza para la programación y la comunicación serie con la computadora.

Pines de E/S digitales (GPIO): El ESP32 cuenta con un gran número de pines de E/S digitales que pueden configurarse como entradas o salidas. Estos pines son

versátiles y se pueden utilizar para conectar una variedad de dispositivos y sensores digitales.

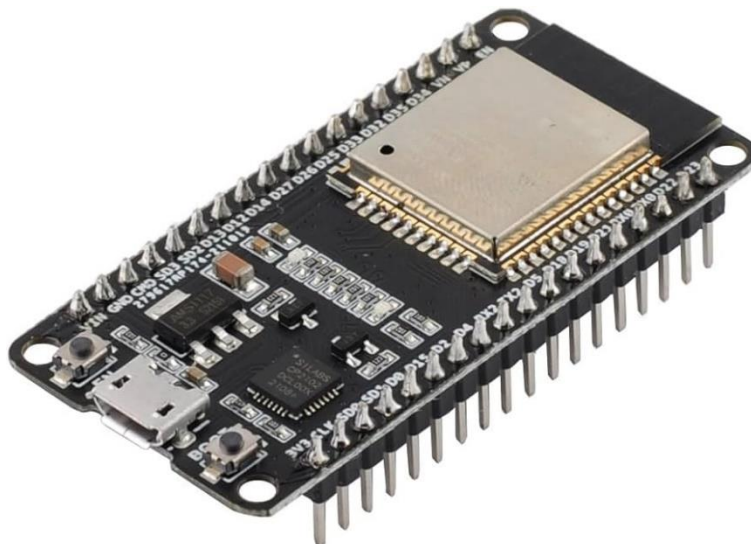
Pines de E/S analógicas (ADC): Al igual que el Arduino Uno, el ESP32 también tiene pines de entrada analógica que se utilizan para medir voltajes analógicos. Esto permite conectar sensores como potenciómetros, fotocélulas y termistores.

Puertos de comunicación serie: El ESP32 tiene múltiples puertos de comunicación serie, incluyendo UART, SPI y I2C. Estos puertos permiten la comunicación con una amplia variedad de dispositivos externos, como sensores, pantallas y módulos de comunicación.

Puertos de comunicación inalámbrica: Una de las características destacadas del ESP32 es su soporte para Wi-Fi y Bluetooth. El ESP32 tiene pines dedicados para la antena Wi-Fi y Bluetooth, así como soporte para otros protocolos de comunicación inalámbrica.

Puertos de alimentación: El ESP32 puede ser alimentado a través de un puerto USB o mediante un voltaje de alimentación externo. También tiene pines dedicados para la entrada de alimentación y pines de tierra (GND).

Otros pines especiales: Además de los pines de E/S estándar, el ESP32 también tiene pines dedicados para funciones específicas, como el control de energía, la gestión de baterías y la programación en modo de arranque.



17. Describe cuales son los puertos del Raspberry y cuál es la función que desempeña cada uno.

Puerto HDMI: Utilizado para conectar el Raspberry Pi a un monitor o televisor mediante un cable HDMI, lo que permite la visualización de la interfaz gráfica de usuario y el contenido multimedia.

Puerto de alimentación microUSB o USB-C: Este puerto se utiliza para suministrar energía al Raspberry Pi. Dependiendo del modelo específico del Raspberry Pi, puede tener un puerto microUSB o USB-C para la alimentación.

Puerto Ethernet (RJ45): Proporciona conectividad de red por cable Ethernet para la conexión a Internet y la comunicación en red local.

Puertos USB: El Raspberry Pi generalmente tiene múltiples puertos USB que permiten conectar dispositivos periféricos como teclados, ratones, cámaras, discos duros externos, entre otros.

Puerto de audio: Permite la conexión de auriculares, altavoces u otros dispositivos de audio al Raspberry Pi para la reproducción de sonido.

Puerto GPIO (General Purpose Input/Output): Estos son pines de propósito general que permiten la conexión de dispositivos y componentes electrónicos externos para interactuar con el Raspberry Pi. Se pueden utilizar para controlar LEDs, leer sensores, controlar motores y mucho más.

Puerto de cámara (CSI): Utilizado para conectar una cámara al Raspberry Pi para capturar imágenes y videos.

Puerto de pantalla táctil (DSI): Permite la conexión de una pantalla táctil al Raspberry Pi para la visualización de contenido interactivo.

Puerto de tarjeta microSD: Utilizado para insertar una tarjeta microSD que contiene el sistema operativo y los archivos del Raspberry Pi.



18. Describe cuales son los puertos del ESP8266 y cuál es la función que desempeña cada uno.

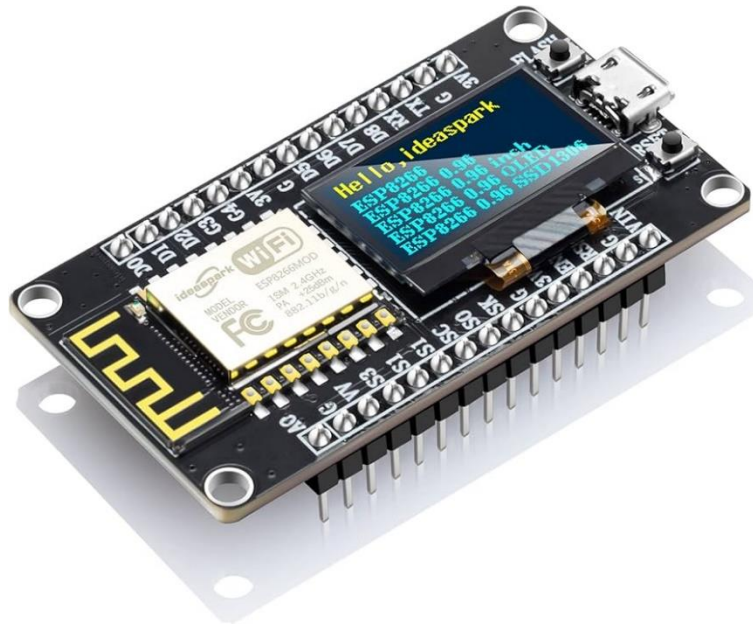
Pines GPIO (General Purpose Input/Output): El ESP8266 generalmente tiene varios pines GPIO que pueden ser configurados como entradas o salidas digitales. Estos pines son utilizados para conectar y controlar dispositivos externos como sensores, LEDs, relés, etc.

Puerto UART (Universal Asynchronous Receiver/Transmitter): El ESP8266 cuenta con un puerto UART que es utilizado para la comunicación serial. Este puerto es comúnmente utilizado para la programación y depuración del ESP8266, así como para la comunicación con otros dispositivos serie.

Puerto SPI (Serial Peripheral Interface): Algunos modelos de ESP8266 tienen soporte para SPI, un protocolo de comunicación serial utilizado para la transferencia de datos entre dispositivos digitales. Este puerto es útil para la comunicación con otros dispositivos como sensores, pantallas, y tarjetas de memoria.

Puerto I2C (Inter-Integrated Circuit): Al igual que el puerto SPI, algunos modelos de ESP8266 también tienen soporte para I2C, otro protocolo de comunicación serial utilizado para conectar dispositivos de forma síncrona con solo dos cables: SDA (datos) y SCL (reloj). Esto permite la conexión de varios dispositivos, como sensores y pantallas, al ESP8266.

Puertos de alimentación: El ESP8266 suele ser alimentado con un voltaje de 3.3V. Hay un pin dedicado para la entrada de alimentación, así como pines de tierra (GND) para conectar la alimentación y proporcionar una referencia de tierra.



Referencias:

<https://alejandra0robleromorales.blogspot.com/2017/05/51-tipos-de-puertos.html>

<http://www.paginaspersonales.unam.mx/files/5190/Asignaturas/1415/Archivo3.2375.pdf>

<https://hardzone.es/tutoriales/componentes/puertos-entrada-salida/>

<https://alejandra0robleromorales.blogspot.com/2017/05/56-aplicaciones-de-buses.html>

<https://learn.microsoft.com/es-es/azure-sphere/app-development/uart?view=azure-sphere-integrated>

<https://polaridad.es/que-es-usart-y-como-funciona-en-la-comunicacion-de-datos/>

<https://immune.institute/blog/que-es-un-compilador/#:~:text=Los%20compiladores%20se%20utilizan%20para,en%20lenguaje%20de%20m%C3%A1quina%20directamente.>

<https://alejandra0robleromorales.blogspot.com/2017/05/54-estandares-de-buses.html>