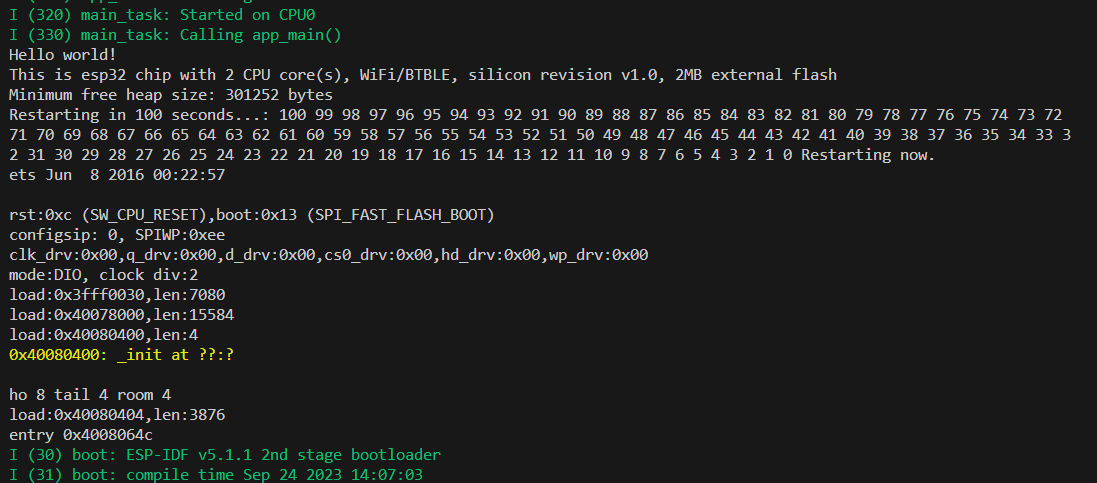
# Practica 1

## Entregable 1

Modifica el período de suspensión de la tarea para que sea mayor o menor, y comprueba que efectivamente esto modifica el comportamiento del firmware cargado.

**Acciones realizadas**:

1. Se crea la constante TIME\_WAIT con valor 100 modificando el bucle dentro de app\_main y al volcar a la placa examinamos el output en el monitor verificando el funcionamiento.

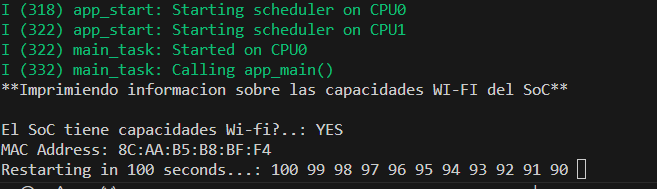


## Entregable 2

Modifica el programa para que se compruebe debidamente si el SoC tiene capacidades WiFi y muestre la información correspondiente por la salida estándar (para ello, puedes consultar [la siguiente página](https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/system/system.html#_CPPv415esp_chip_info_t)).

**Acciones realizadas**:

1. Para verificar si el chip soporta wifi realizamos las siguientes acciones:
   1. Recuperamos la información del chip llamando a la función esp\_chip\_info con una estructura de tipo esp\_chip\_info\_t.
   2. Realizamos una operación AND entre la sección features de la estructura obtenida y la MACRO CHIP\_FEATURE\_WIFI\_BGN.
2. Adicionalmente mostramos información de la MAC del chip WIFI:
   1. Obtenemos la dirección base de la MAC programada de fabrica con la función esp\_efuse\_mac\_get\_default
   2. Leemos la dirección MAC con la función esp\_read\_mac
   3. Mostramos el array obtenido con la función print\_mac

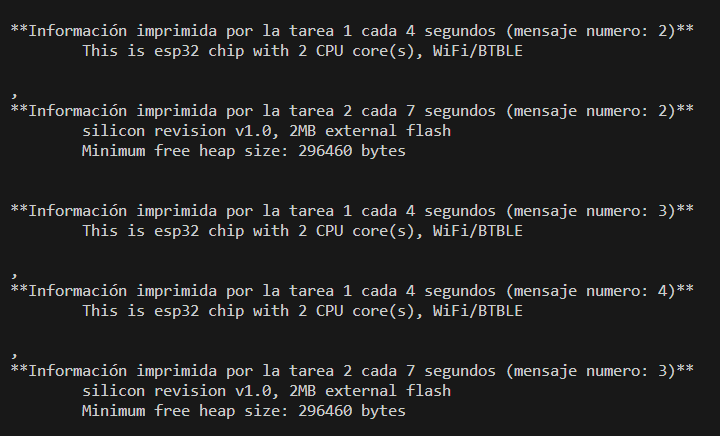


## Entregable 3

Implementa una modificación del programa hello\_world que implemente y planifique dos tareas independientes con distinta funcionalidad (en este caso, es suficiente con mostrar por pantalla algún mensaje) y distintos tiempos de suspensión. Comprueba que, efectivamente, ambas tareas se ejecutan concurrentemente.

**Acciones realizadas**:

1. Se crean dos tareas:
   1. Tarea 1 muestra cada 4 segundos la información de los cores y capacidades del chip
   2. Tarea 2 muestra cada 7 información del chip y su versión, capacidad y tipo de la memoria flash y la disponibilidad de memoria de pila.
   3. En ambas tareas, la información se obtiene localmente usando la función esp\_chip\_info.

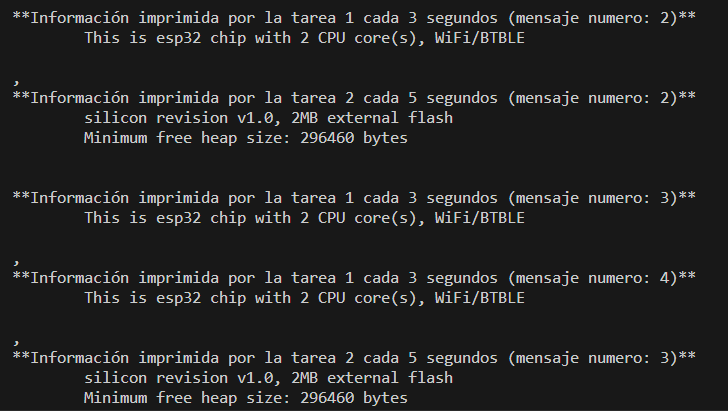


## Entregable 4

Modifica el proyecto hello\_world para que defina dos opciones de configuración que permitan definir el tiempo de espera de cada una de las dos tareas que hayas definido en tu anterior solución. Haz uso de ellas en tu código y comprueba que efectivamente su modificación a través del sistema de menús permite una personalización del comportamiento de tus códigos.

**Acciones realizadas**:

1. Dentro del fichero Kconfig.projbuild se han definido las entradas TIME\_WAIT\_TASK1 & TIME\_WAIT\_TASK2 obteniendo su valor a partir de CONFIG\_TIME\_WAIT\_TASK1 y CONFIG\_TIME\_WAIT\_TASK1 respectivamente.
2. Se muestra la misma información que en el entregable 3.



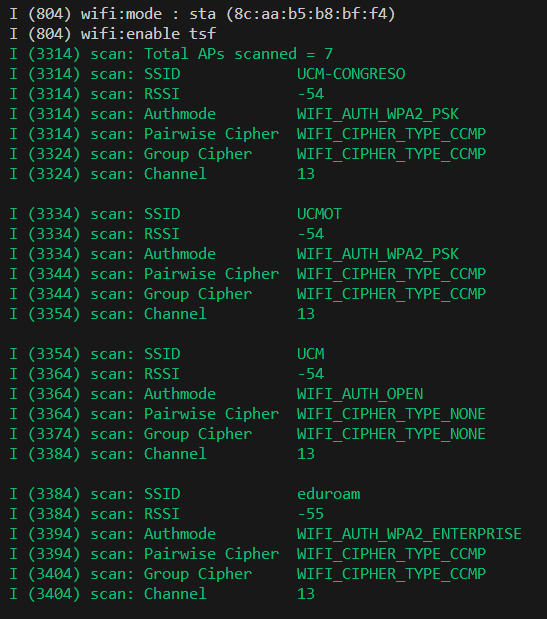
## Entregable 5 - Escaneado de redes WiFi

Compila, flashea y monitoriza el ejemplo scan situado en el directorio examples/wifi/scan. Crea un nuevo proyecto a partir de este ejemplo y amplia el número máximo de redes a escanear a 20 a través del menú de configuración del ejemplo. Crea un punto de acceso WiFi con tu teléfono móvil y observa que, efectivamente, es escaneado por el ejemplo.

Analiza el código de la función wifi\_scan (tarea principal). Céntrate especialmente en las líneas que permiten activar y configurar el escaneado de redes. Intenta entender el funcionamiento general del programa, consultando y apuntando el cometido de cada línea, con especial interés a aquellas funciones con prefijo esp\_wifi\_\*. Si tienes dudas puede consultar la documentación oficial de ESP-IDF.

**Acciones realizadas**:

1. Se compila, flashea y monitoriza el ejemplo scan.
2. Comentadas las llamadas a las funciones dentro de wifi\_scan.

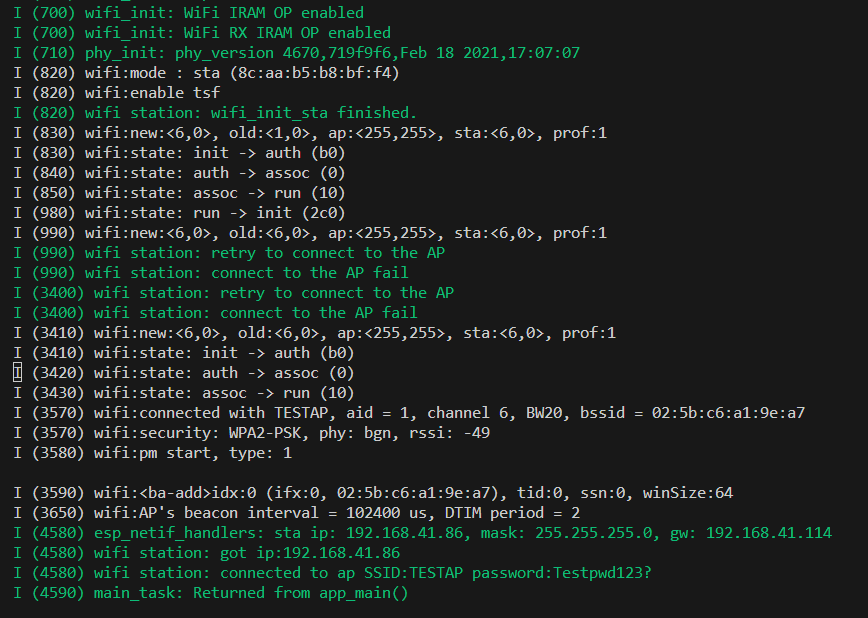


## Entregable 6 - Gestión de eventos de red

Crea un proyecto a partir del ejemplo station situado en el directorio examples/wifi/getting\_started. Compilalo, flashealo y monitoriza su salida estándar. Acuérdate de modificar el SSID de la red al que conectará, así como la contraseña elegida a través del sistema de menús de configuración.

**Acciones realizadas**:

1. Parametrizada la wifi del móvil con valores para testing
2. Al generar el proyecto desde examples/wifi/getting\_started station, se configuran parámetros de entrada desde menuconfig: Wifi SSID / Wifi Password.



Responde a la siguiente pregunta de forma razonada: ¿Qué eventos se asocian a la ejecución de qué función en el firmware que estás estudiando?

Los eventos WIFI\_EVENT e IP\_EVENT se asocian a la función event\_handler.

* Ante un evento WIFI\_EVENT dentro de la función se intenta la conexión llamando a la función esp\_wifi\_connect();
* En caso de un evento IP\_EVENT se muestra la ip obtenida al conectarse a la red wifi.

# Entregable 7

Modifica el firmware para que el handler de tratamiento de la obtención de una dirección IP sea independiente del tratamiento del resto de eventos del sistema WiFi que ya se están considerando. Comprueba que, efectivamente sigue observándose la salida asociada a dicho evento, aun cuando ambas funciones sean independientes. Entrega el código modificado.

**Acciones realizadas**:

1. Se generan dos funciones distintas para cada evento: event\_handler\_WIFIevent y event\_handler\_IPevent.
2. Se verifica la información en el monitor.

