CEC Continuing Education Center

DesignNews

Scratch Building Raspberry Pi RP2040 IoT Devices

Day 2:

Assembling an RP2040/Pico/Pico W Linux Toolchain

Sponsored by

Digi-Key ELECTRONICS

informa markets

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Attendee Chat’ by maximizing the chat widget in your dock.

Fred Eady

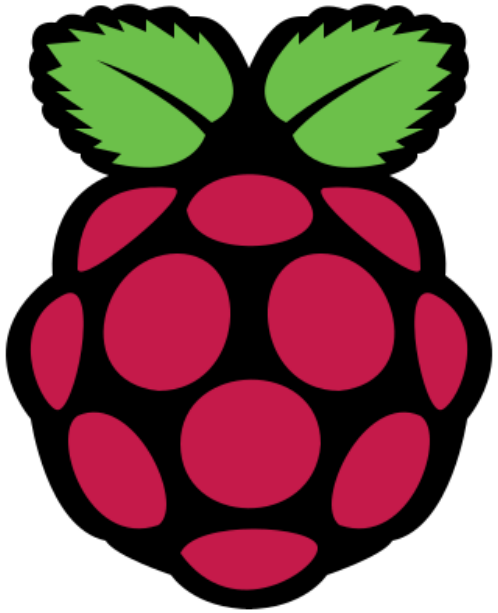
Visit ‘Lecturer Profile’ in your console for more details.

CEC Continuing Education Center

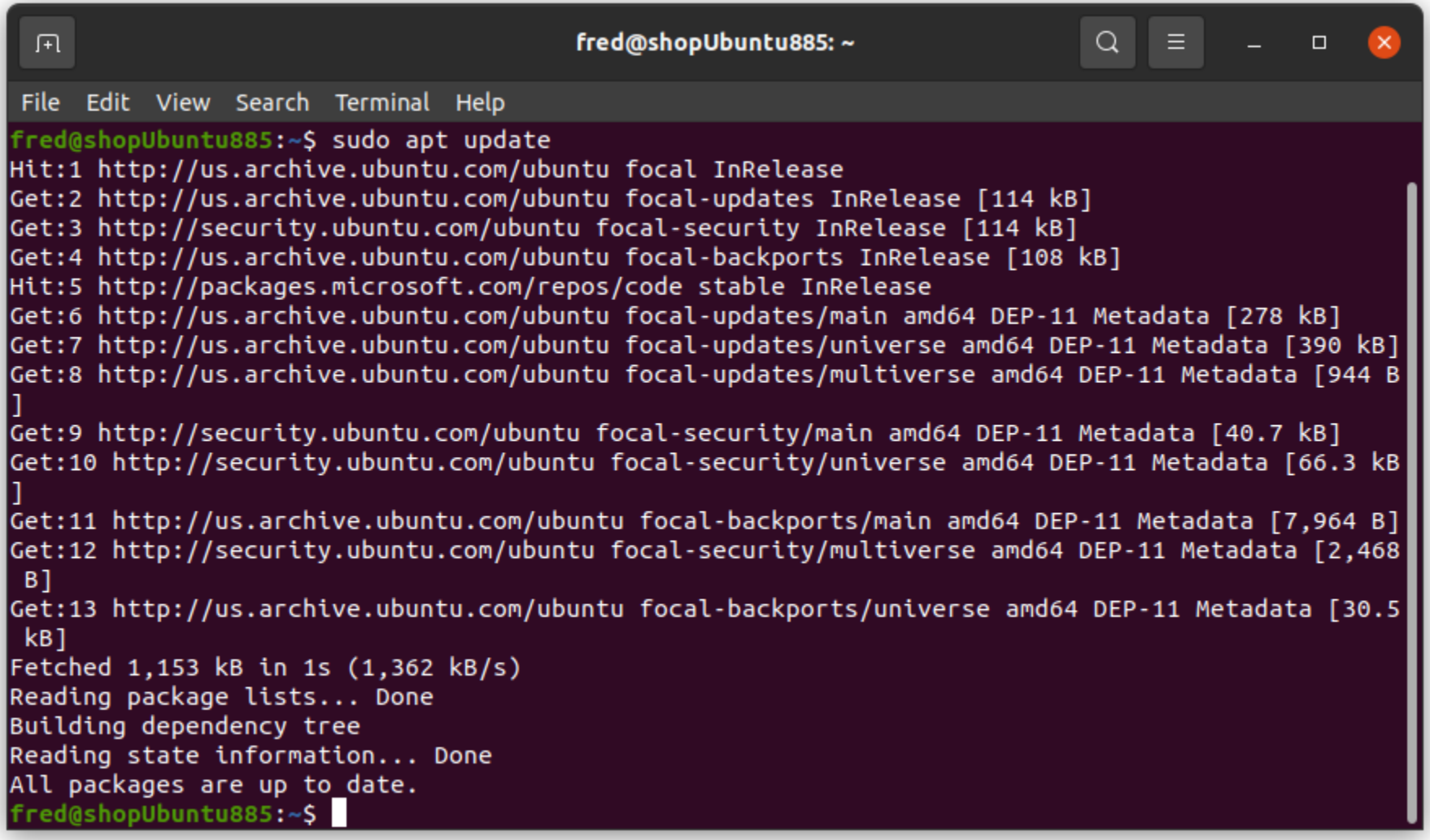
Sponsored By Digi-Key Corporation

AGENDA

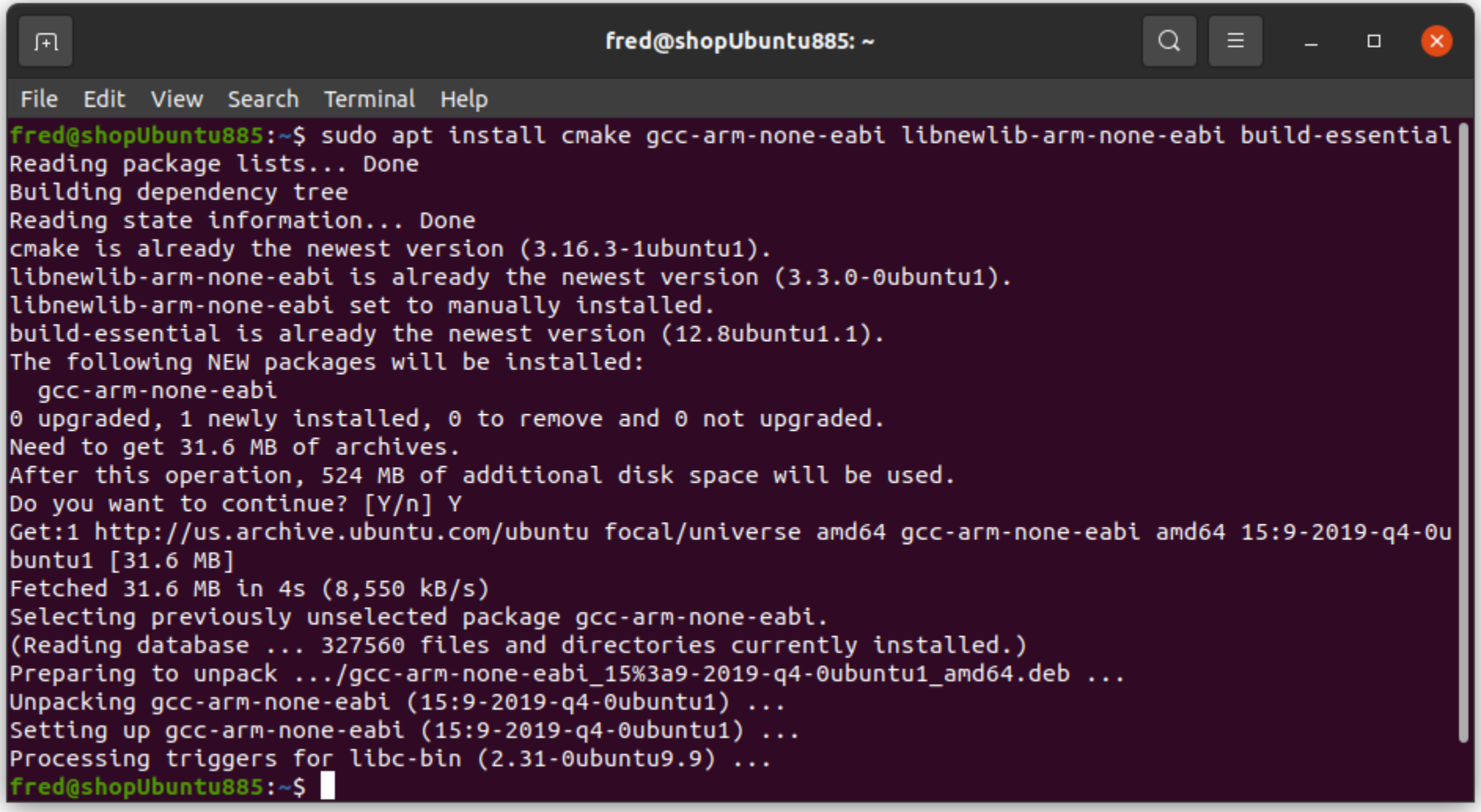
- Install the RP2040 Toolchain
- Install the Raspberry Pi Pico C/C++ SDK
- Install Visual Studio Code for Linux
- Toolchain Installation Verification



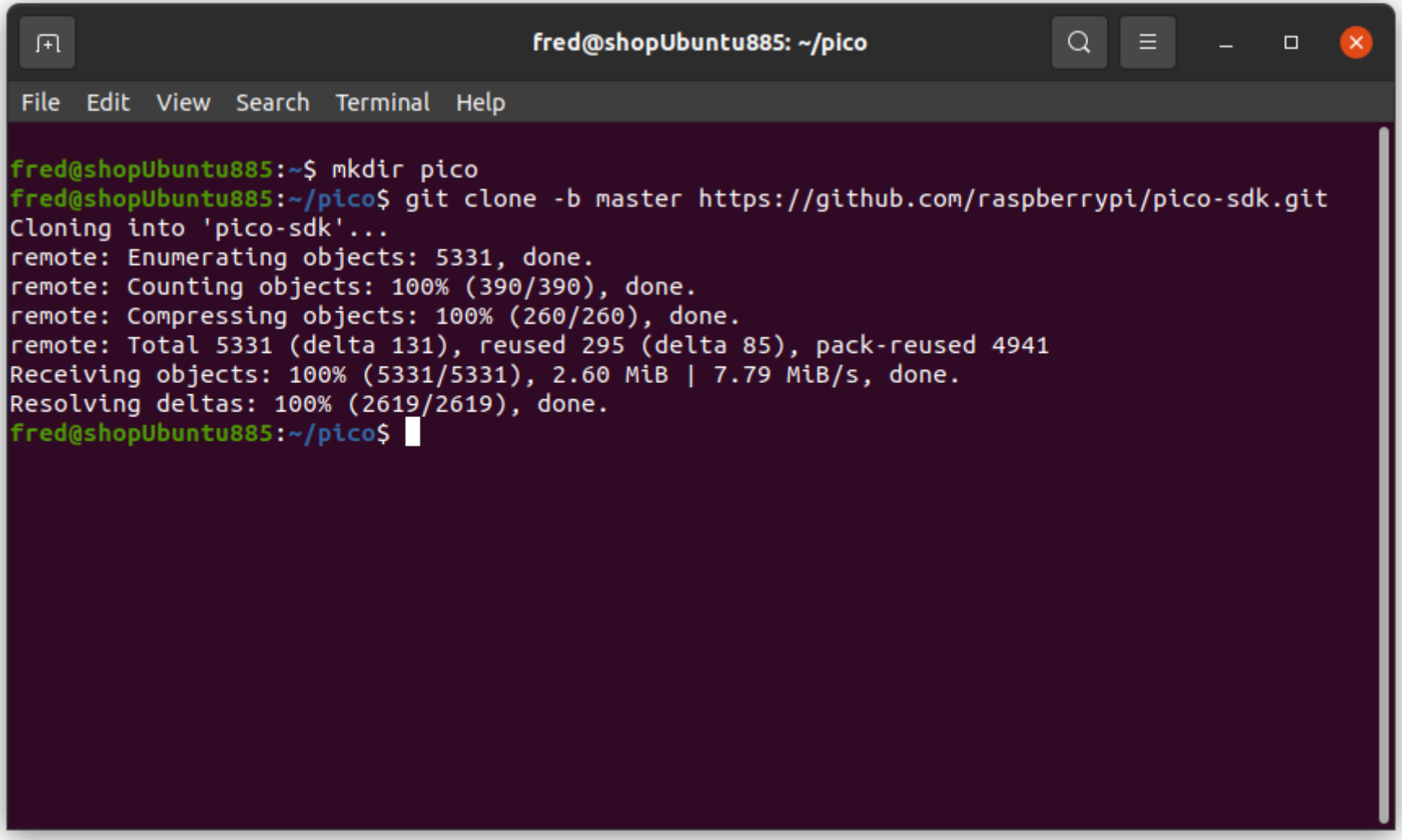
Perform an Ubuntu Update



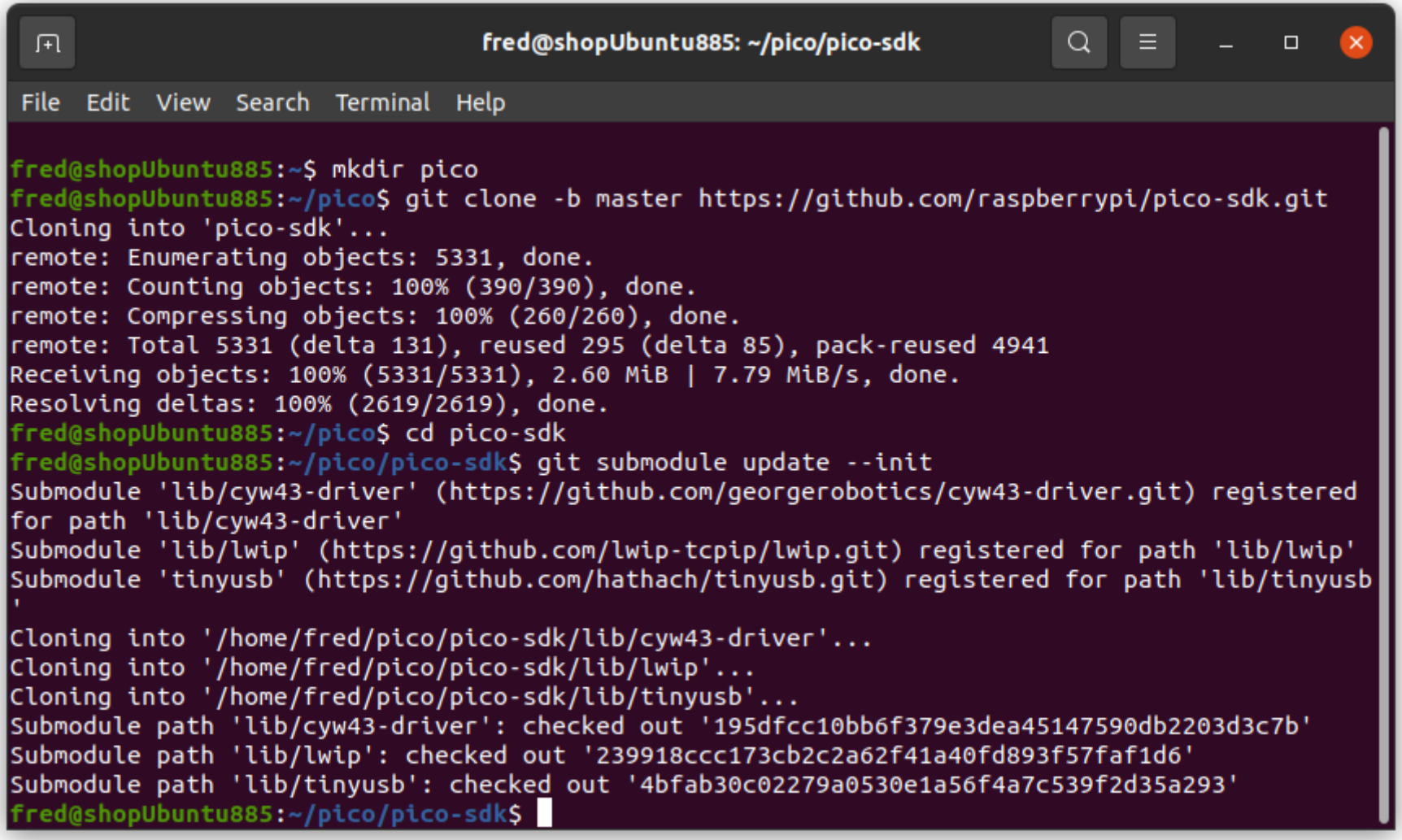
GCC Toolchain Installation



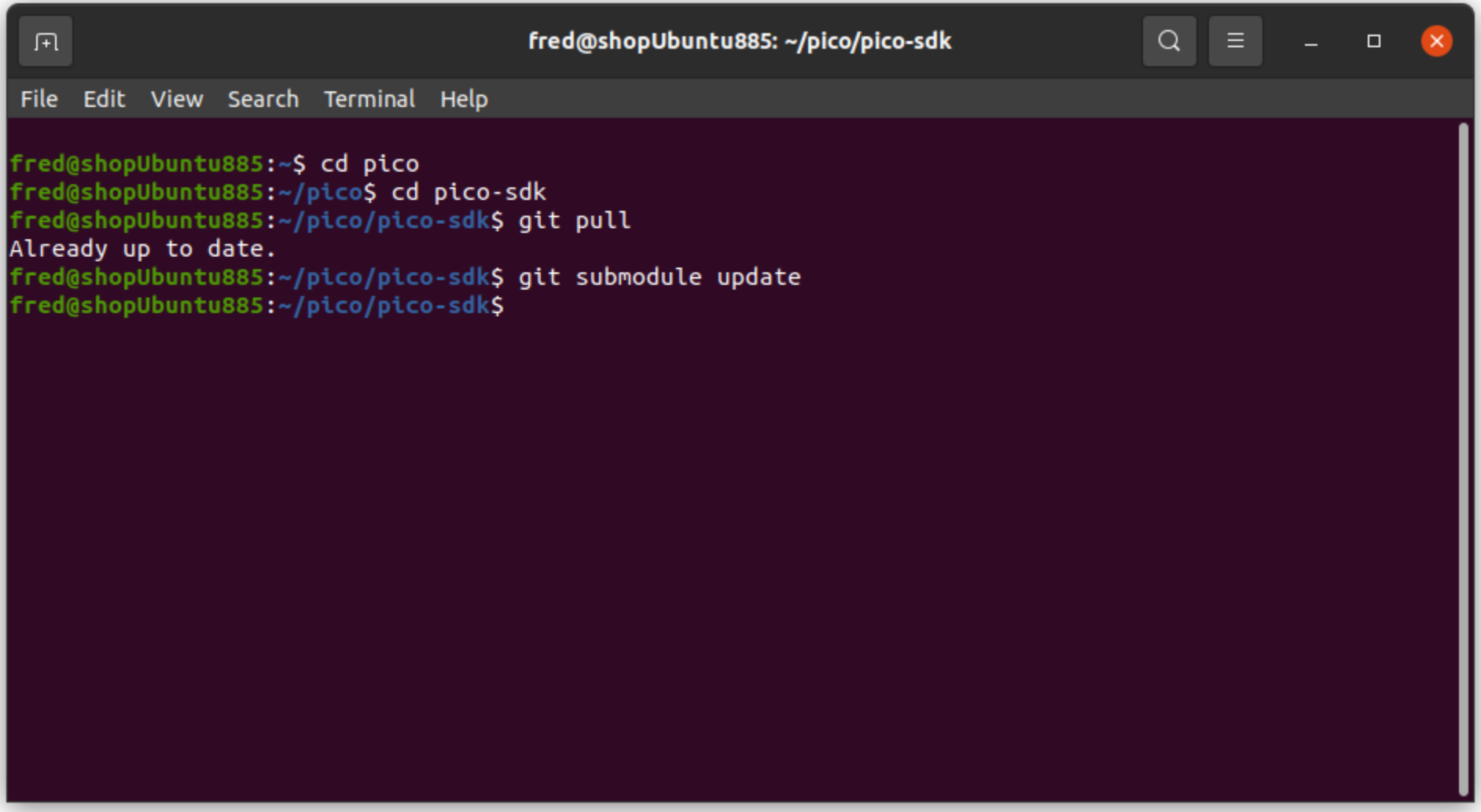
Raspberry Pi Pico C/C++ Installation



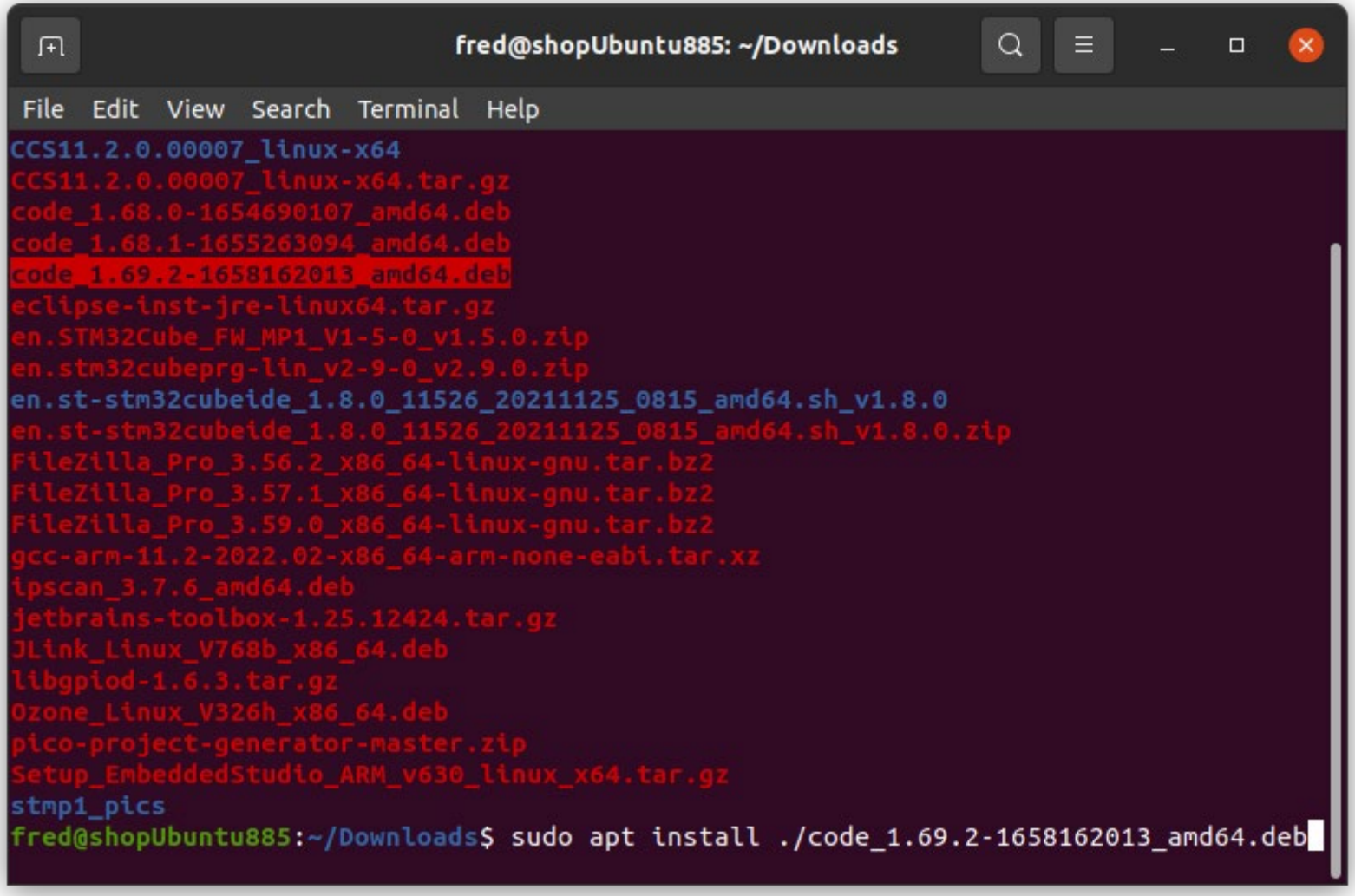
Raspberry Pi Pico C/C++ Installation



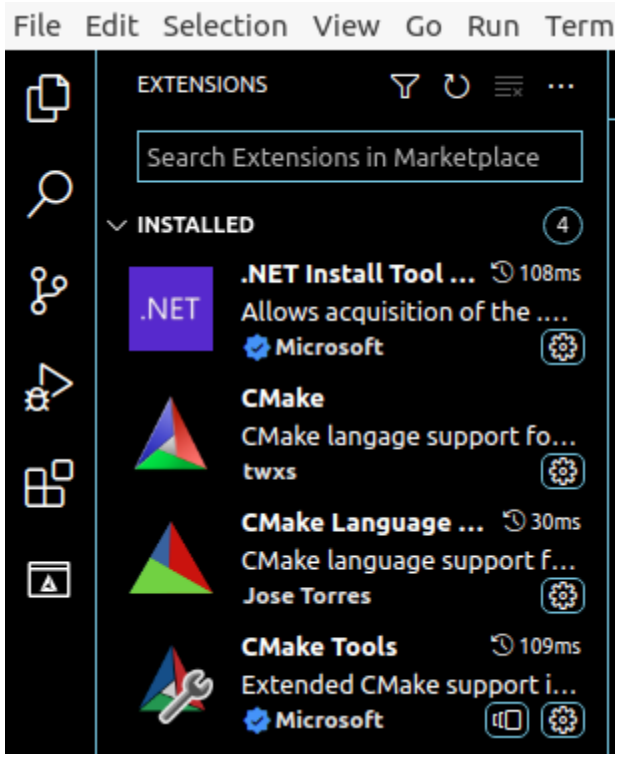
Raspberry Pi Pico C/C++ Installation



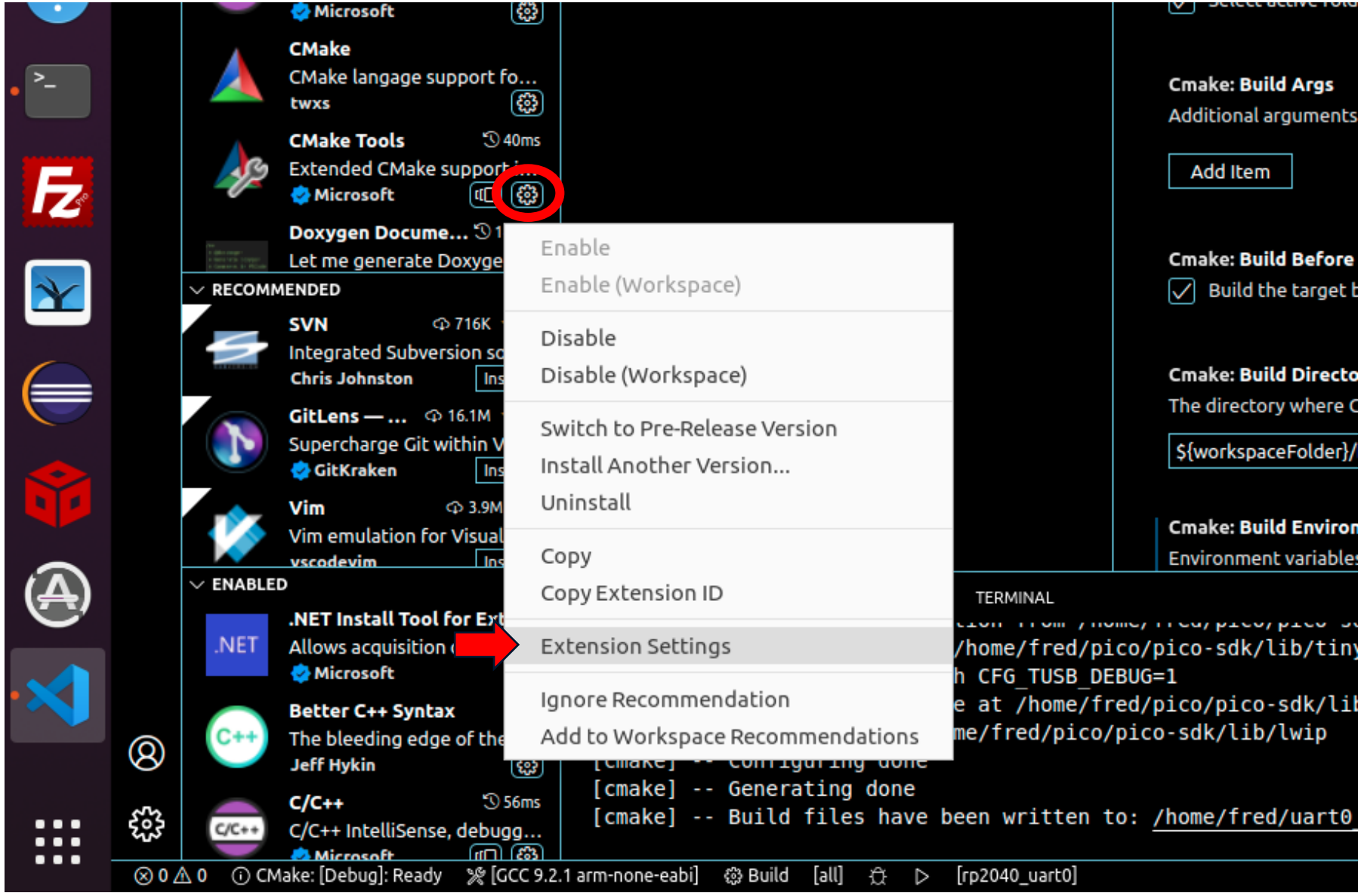
Visual Studio Code Debian Installation



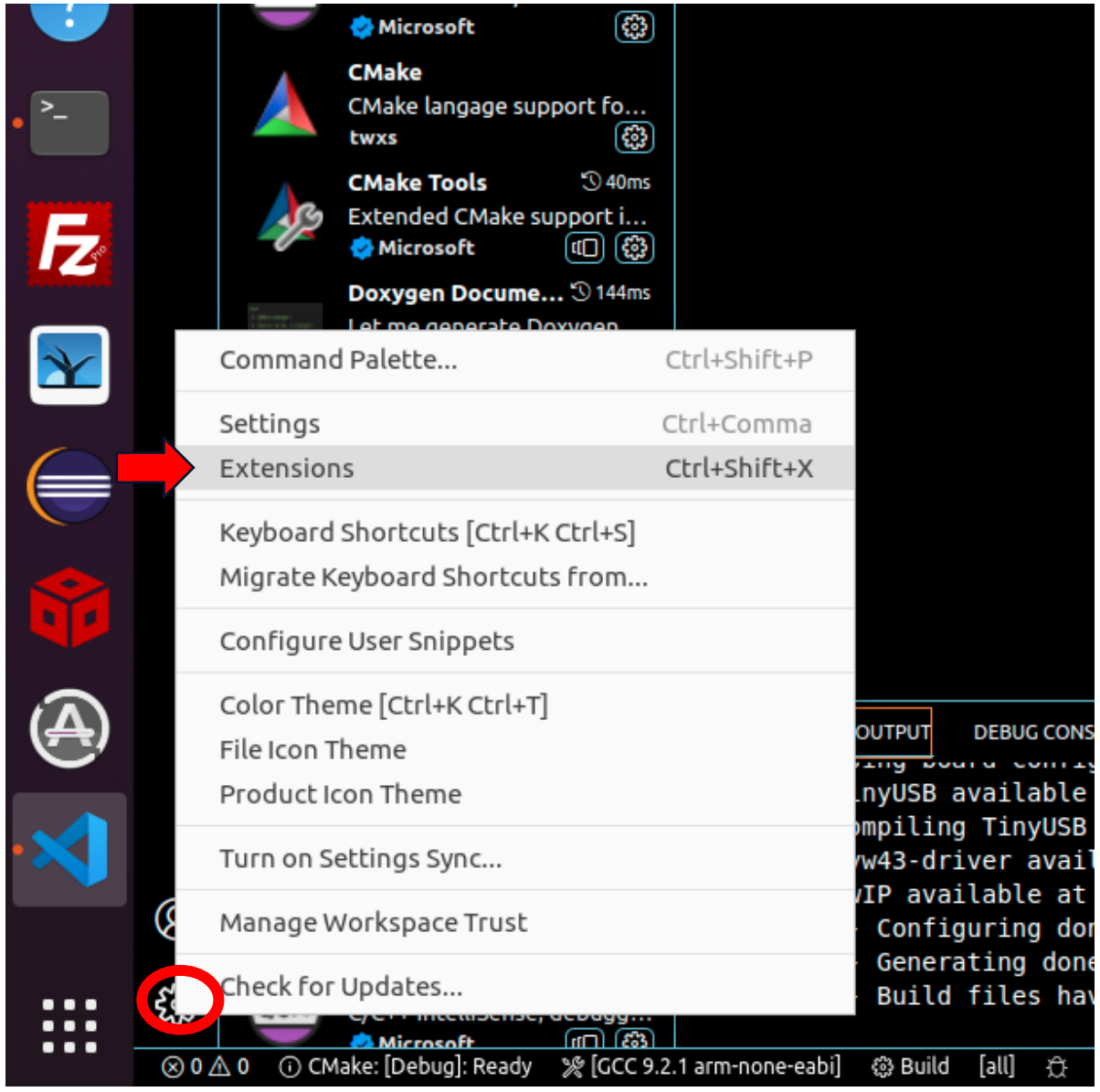
Visual Studio Code Extension Installation



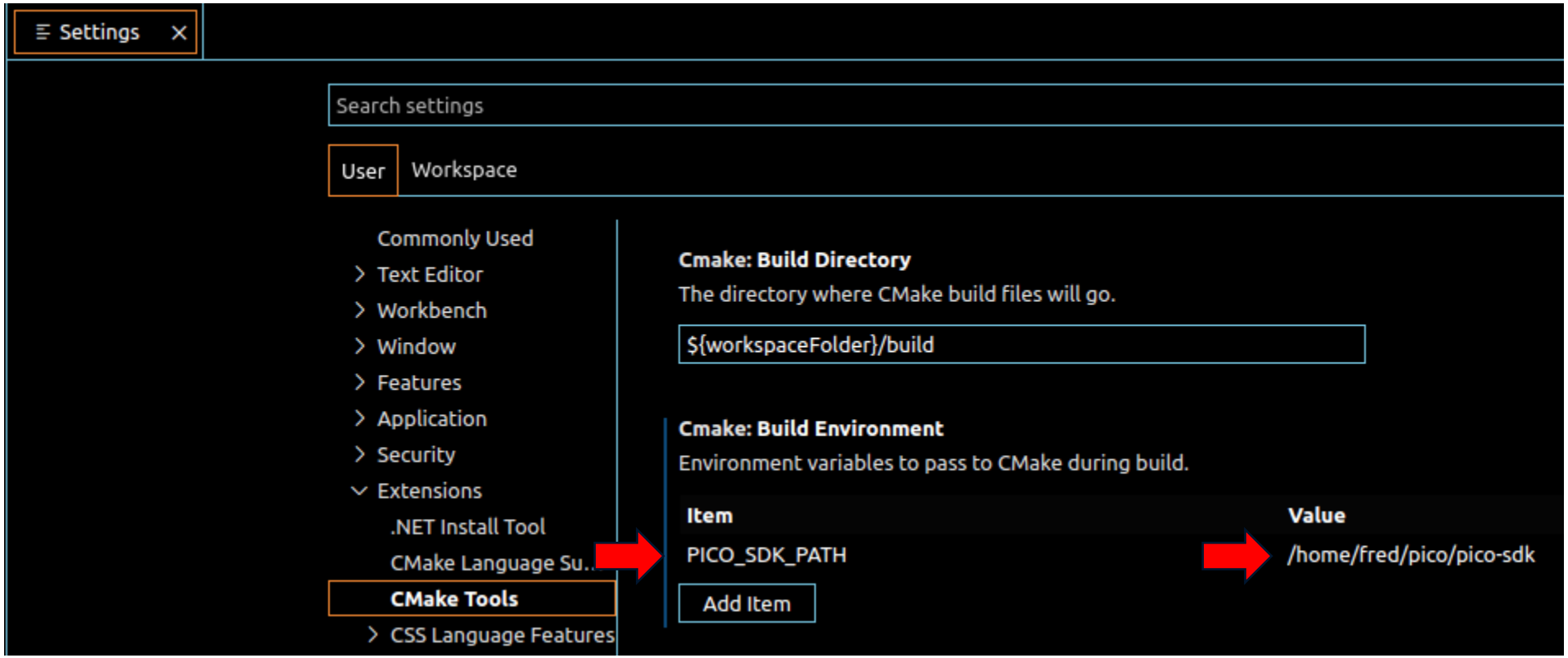
Visual Studio Code Extension Settings



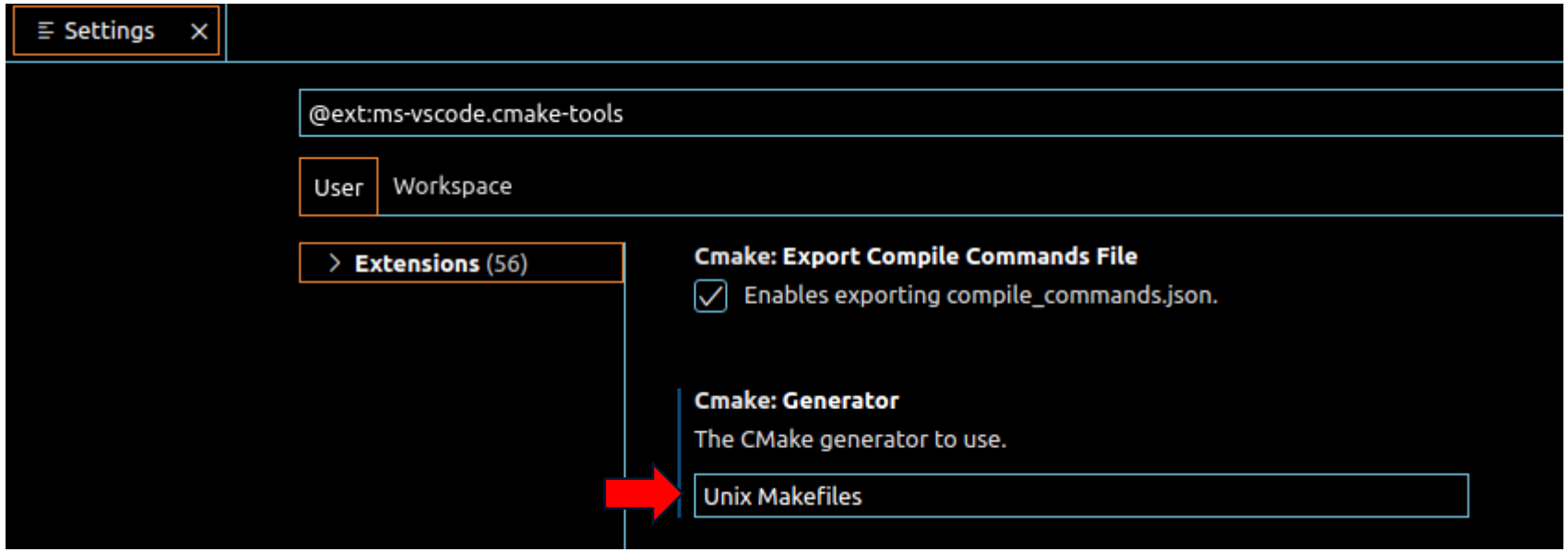
Alternate Path to Extension Settings



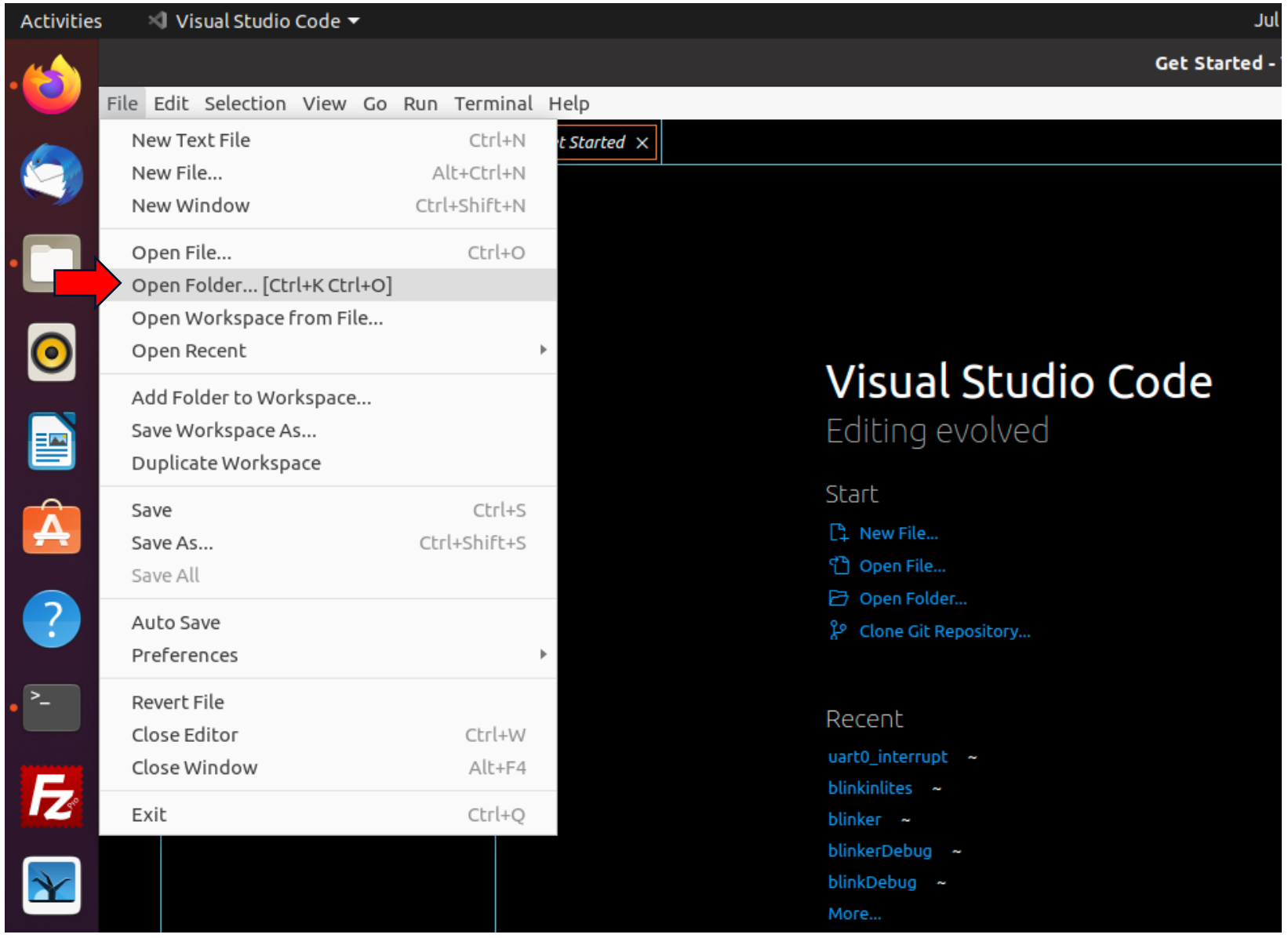
Set Absolute Path to Pico SDK



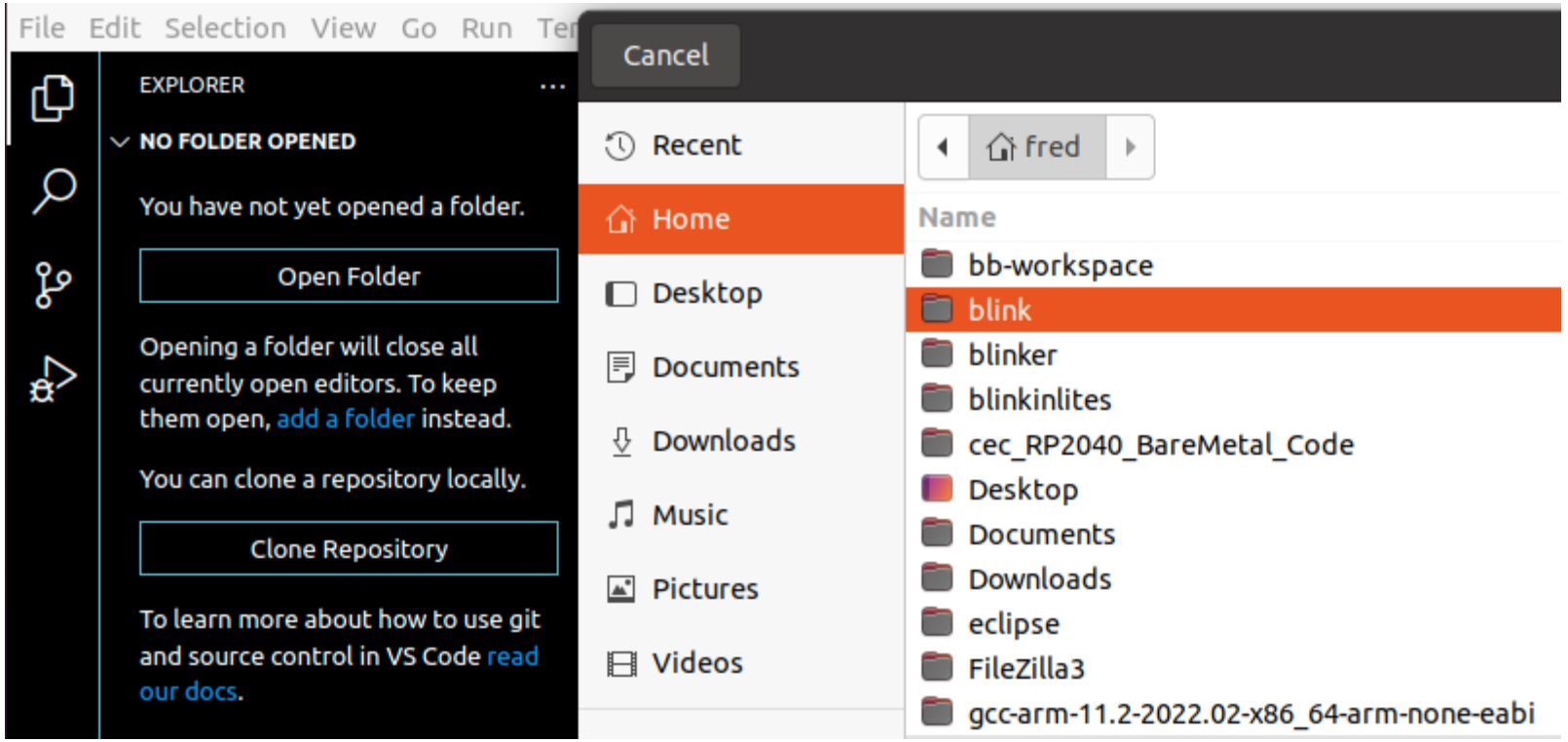
Specify Linux Build System



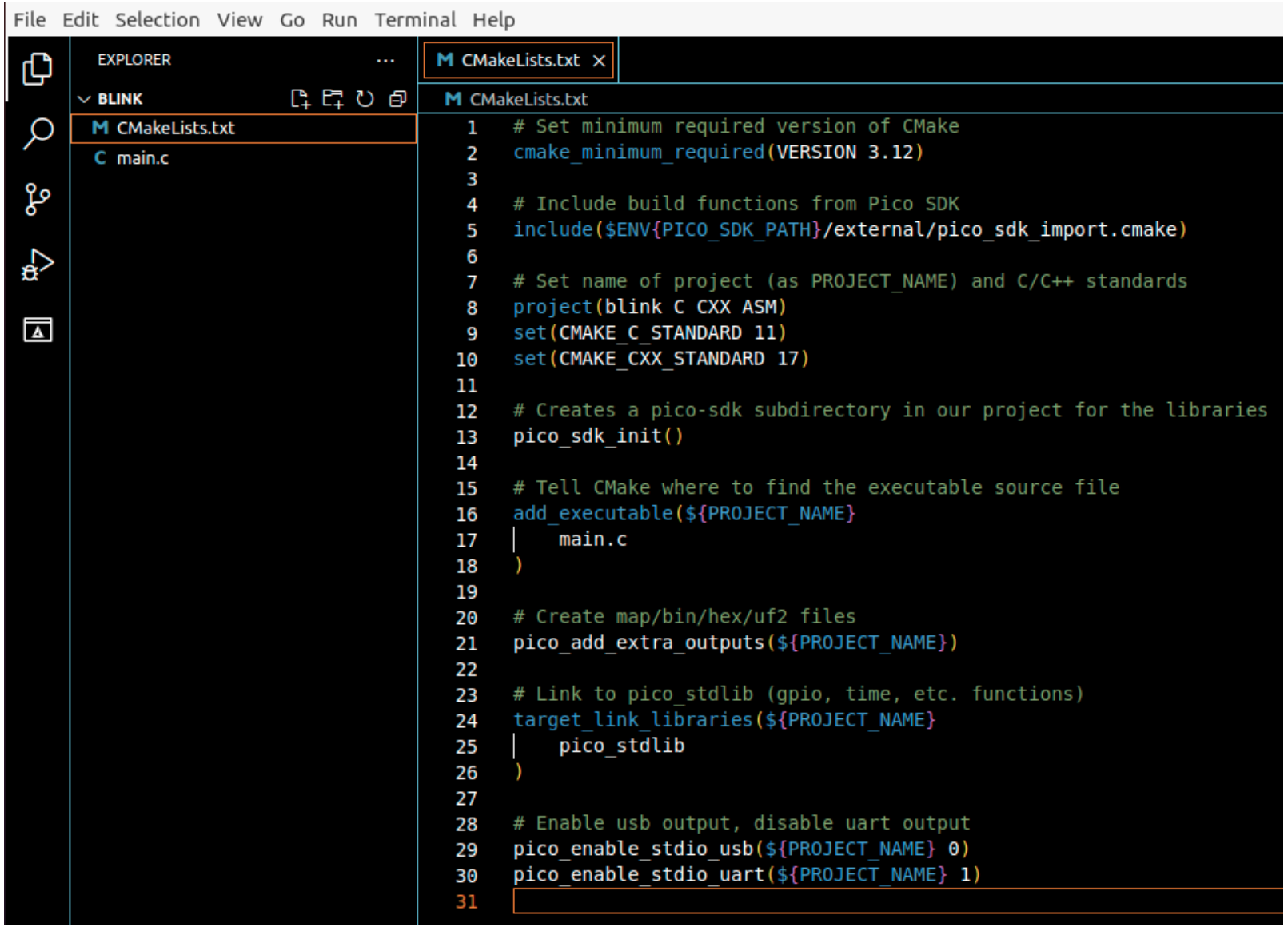
Open Project Folder



Open Project Folder

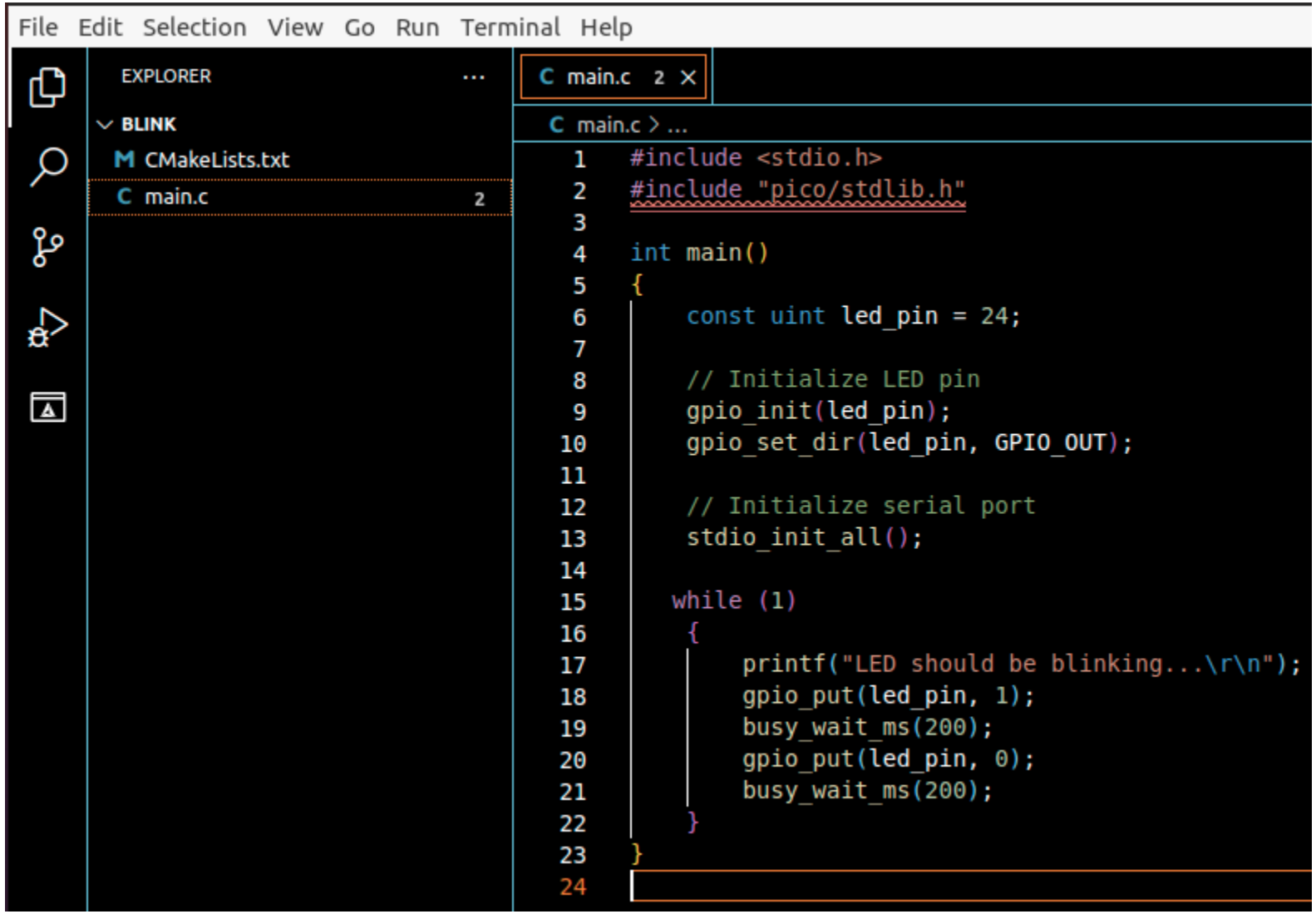


Create CMakeLists.txt

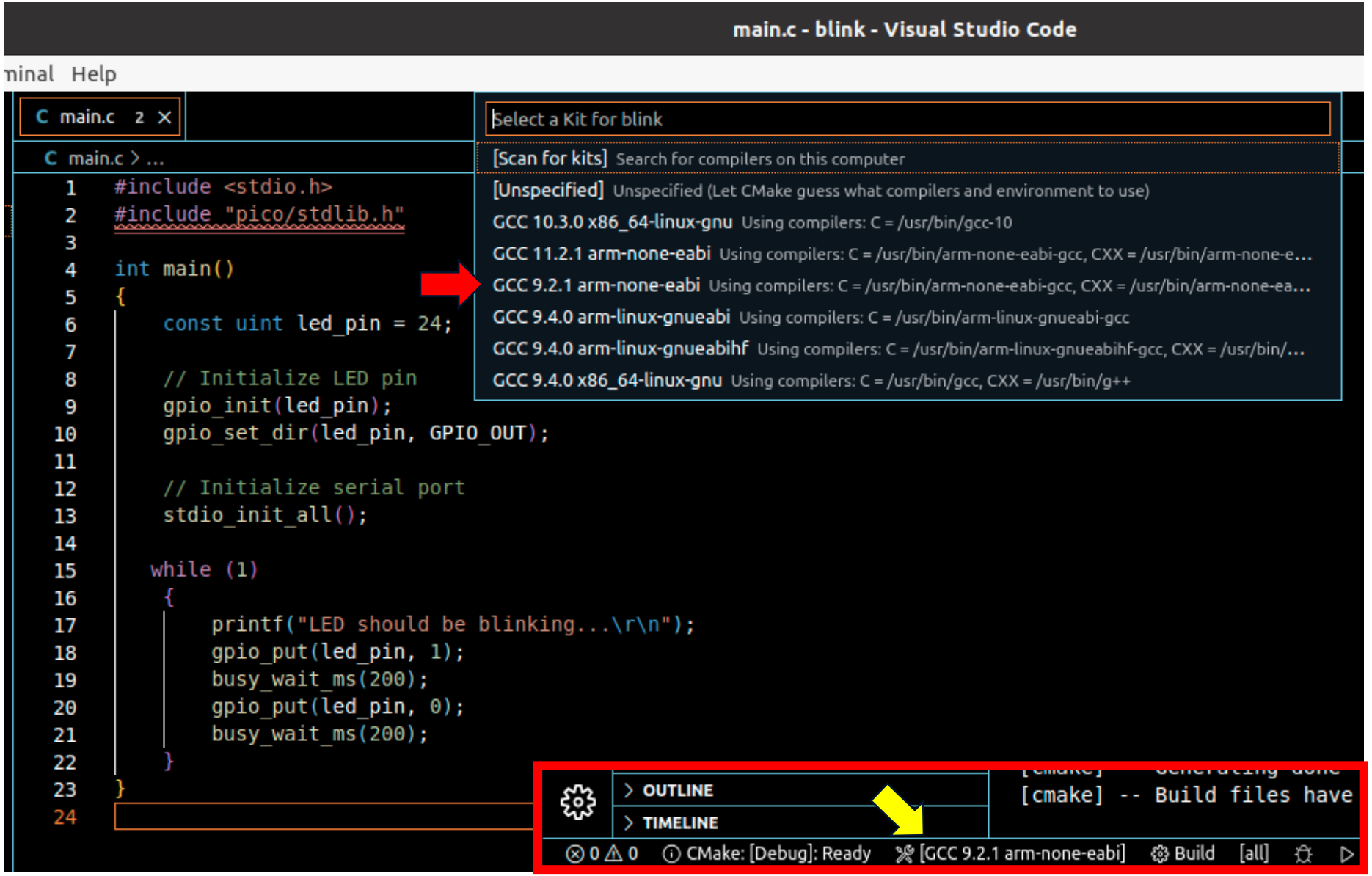


CEC Continuing Education Center
Sponsored By Digi-Key Corporation

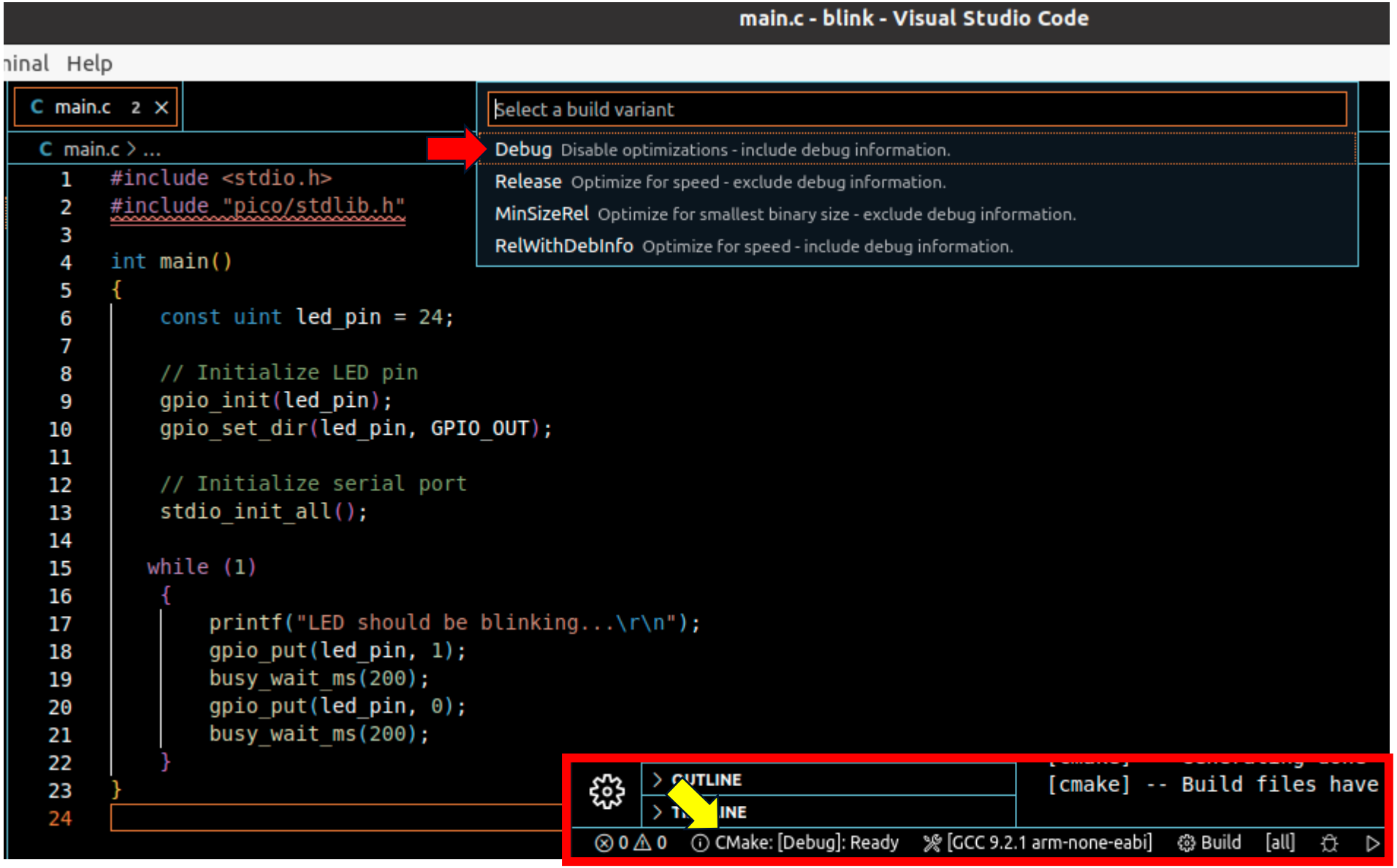
Create main.c



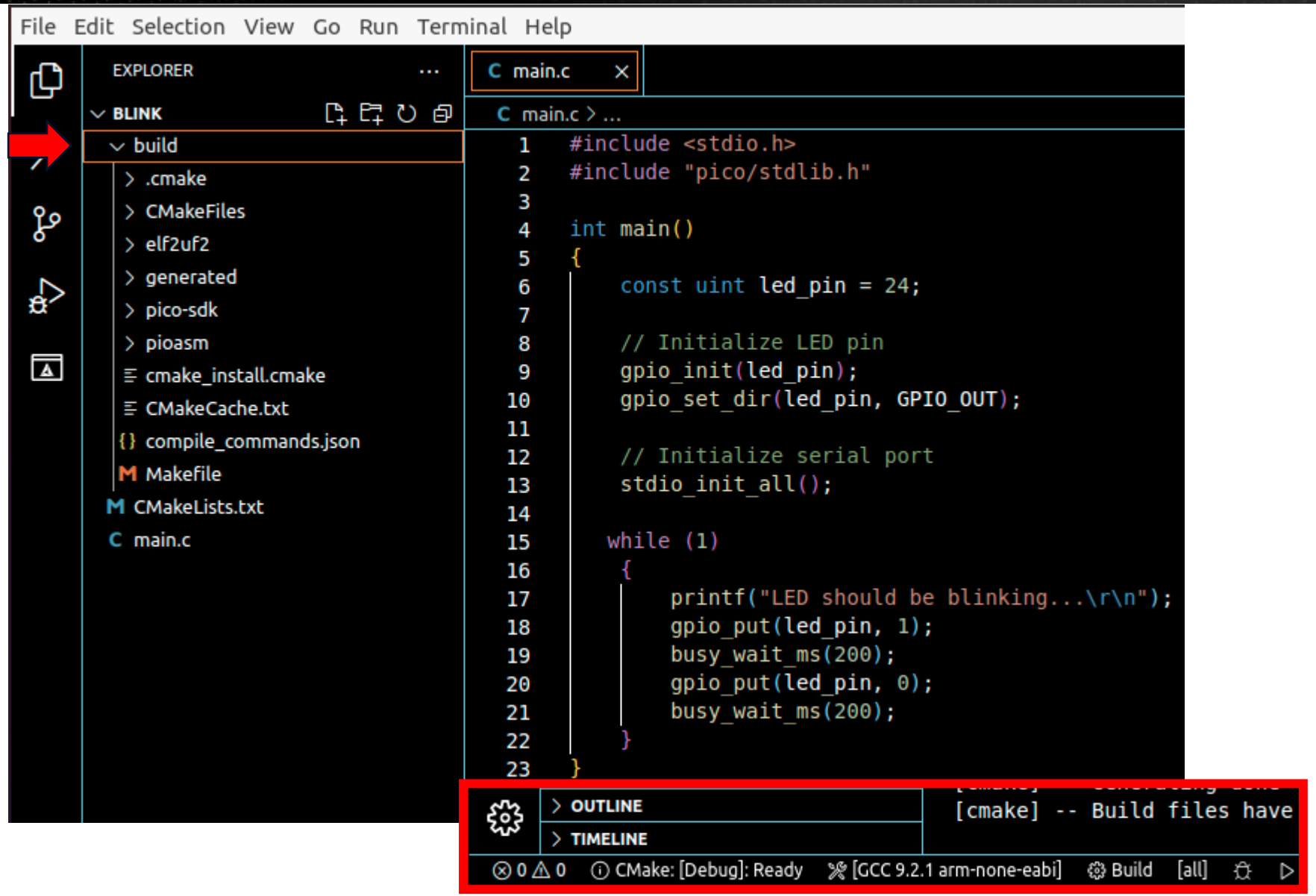
Select the Compiler Kit



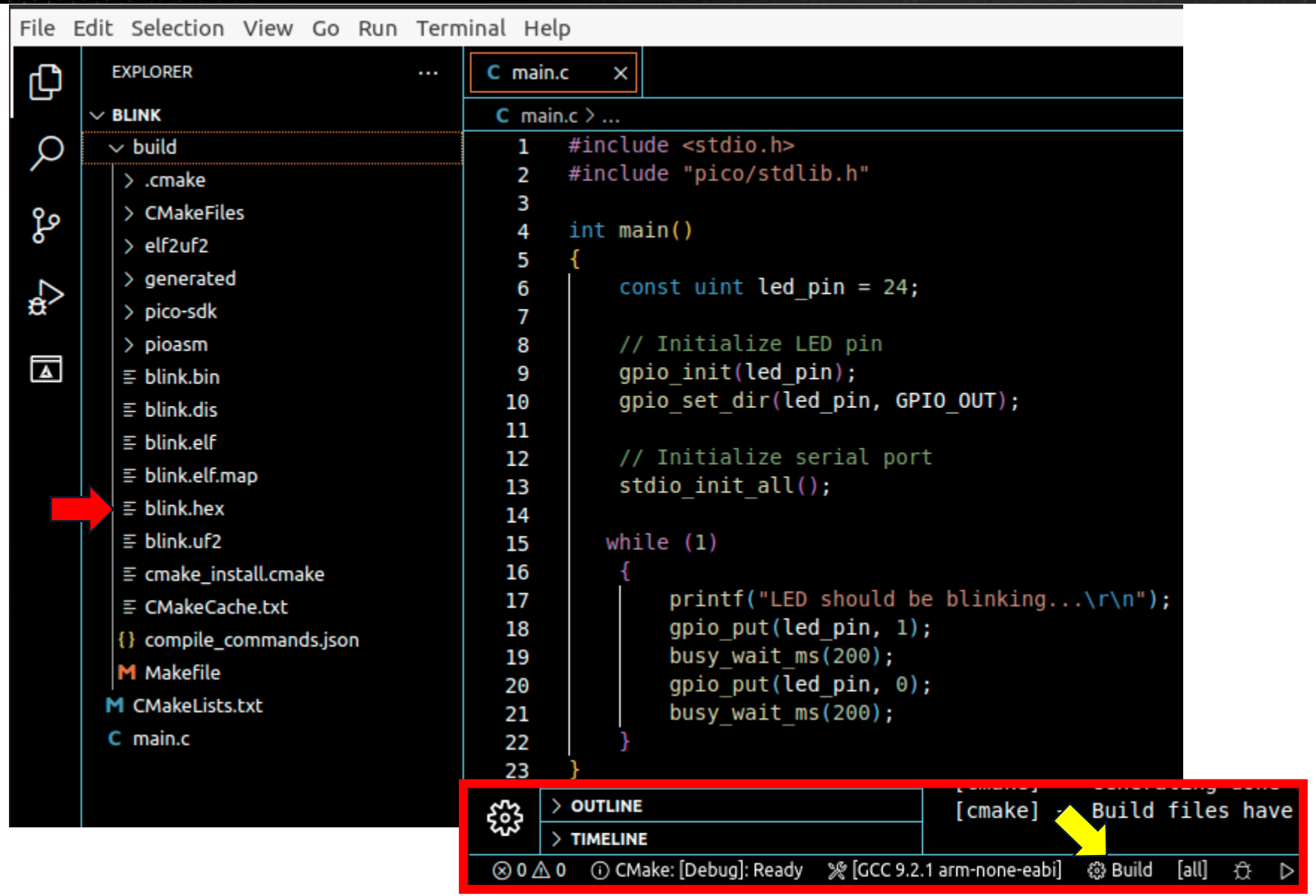
Execute CMake



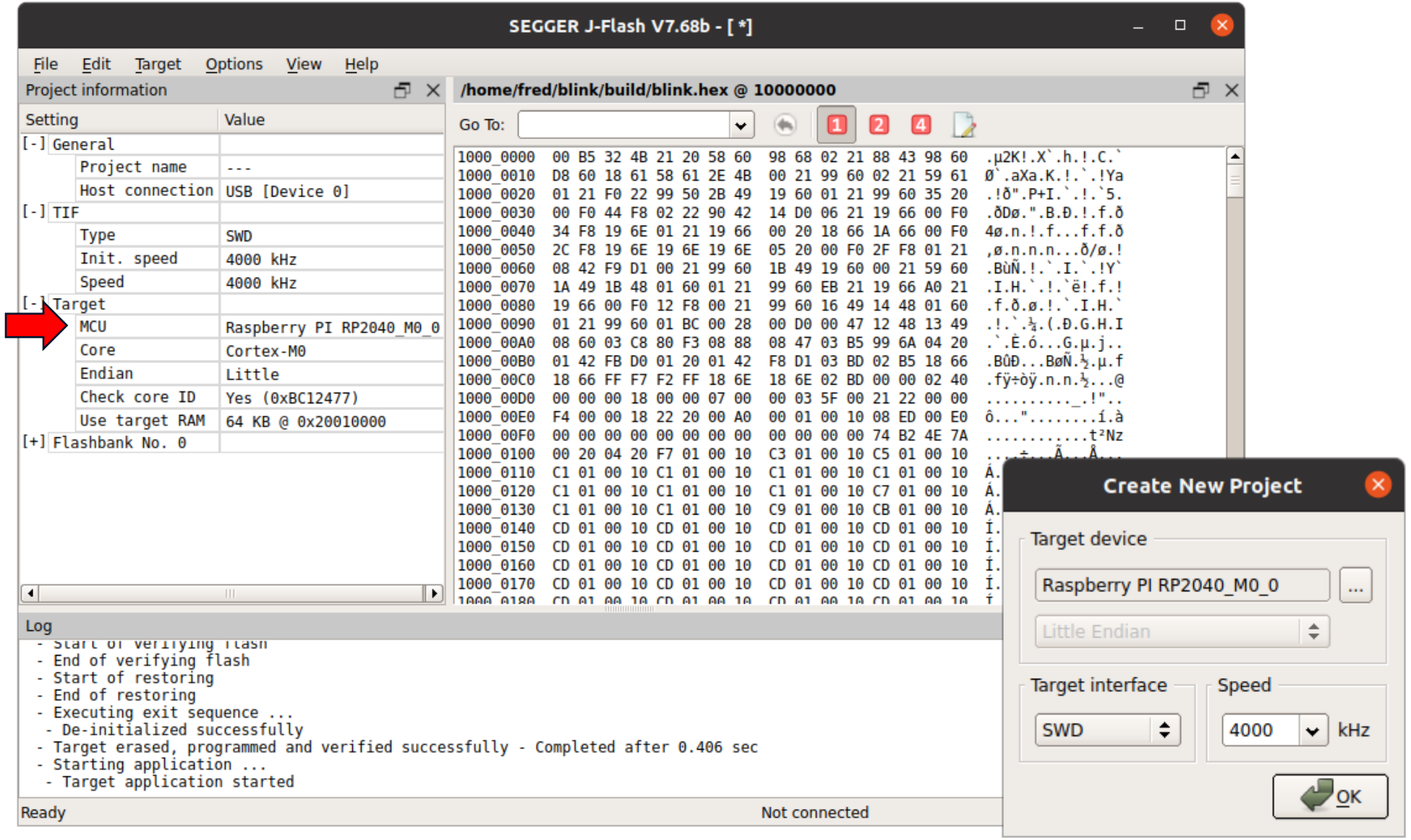
Execute Cmake Results



Execute Build



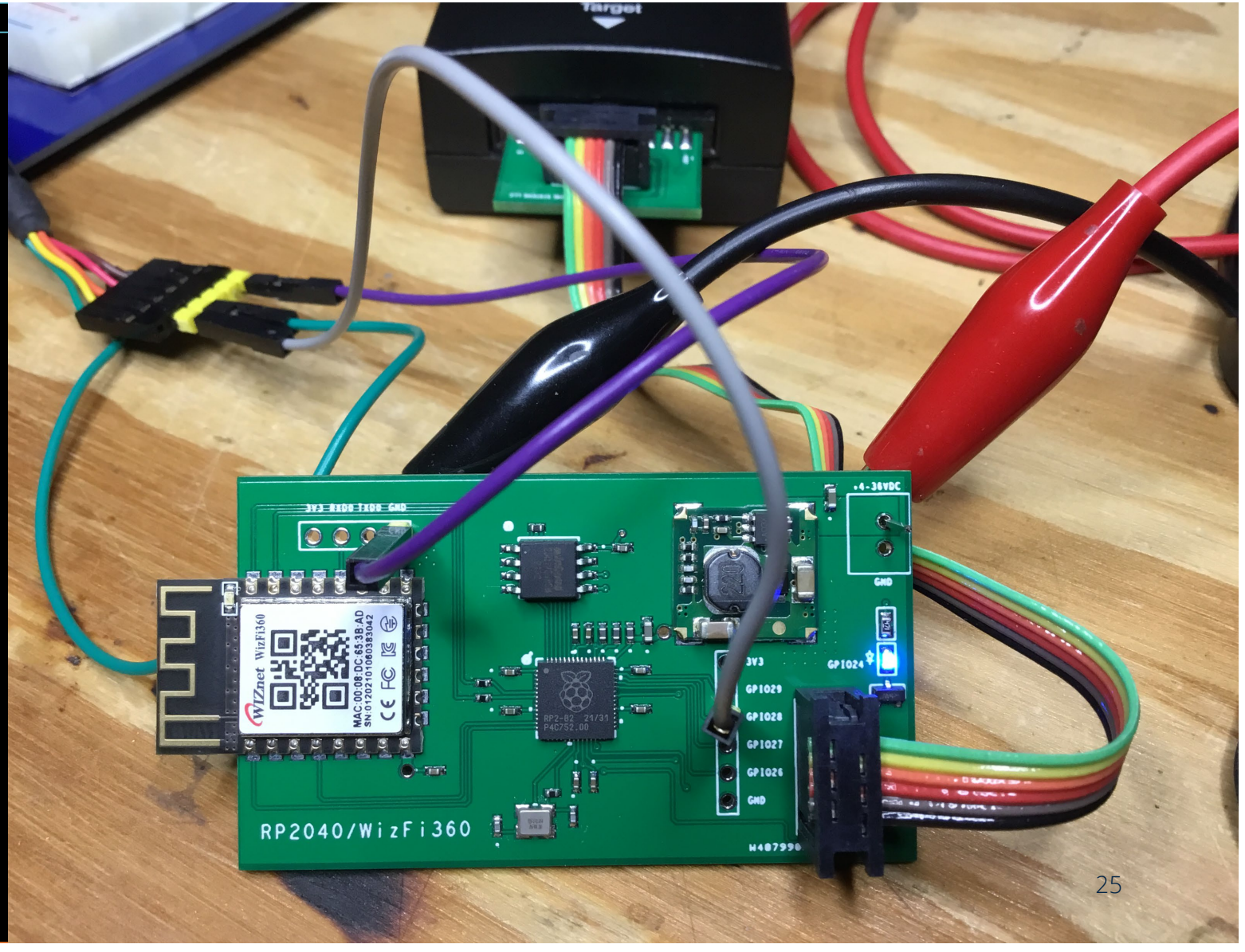
Program the RP2040



Enable Alternate UART0

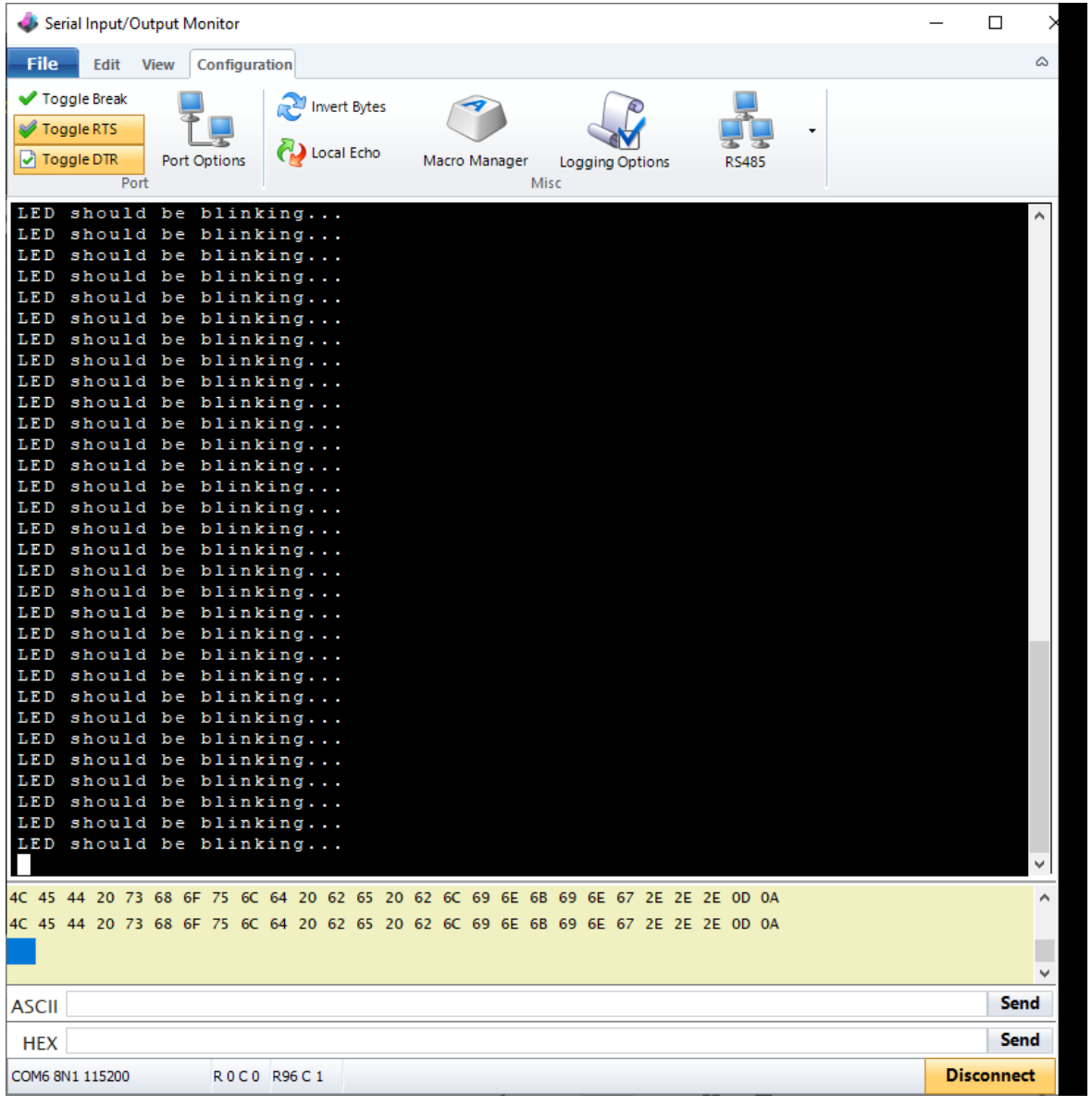
main.c > ...

```c
#include "pico/stdlib.h"

#define UART0_ID  uart0
#define BAUD_RATE 115200
#define DATA_BITS 8
#define STOP_BITS 1
#define PARITY    UART_PARITY_NONE

// define uart pins
#define UART0_TX_PIN 28
#define UART0_RX_PIN 29
#define LED_PIN      24

int main()
{
    // Initialize LED pin
    gpio_init(LED_PIN);
    gpio_set_dir(LED_PIN, GPIO_OUT);

    // Initialize serial port
    gpio_set_function(UART0_TX_PIN, GPIO_FUNC_UART);
    gpio_set_function(UART0_RX_PIN, GPIO_FUNC_UART);
    uart_init(UART0_ID, 115200);
    uart_set_hw_flow(UART0_ID, false, false);
    uart_set_format(UART0_ID, DATA_BITS, STOP_BITS, PARITY);
    uart_set_fifo_enabled(UART0_ID, false);

   while (1)
    {
        uart_puts(UART0_ID,"LED should be blinking...\r\n");
        gpio_put(LED_PIN, 1);
        busy_wait_ms(200);
        gpio_put(LED_PIN, 0);
        busy_wait_ms(200);
    }
}
```



Enable Alternate UART0

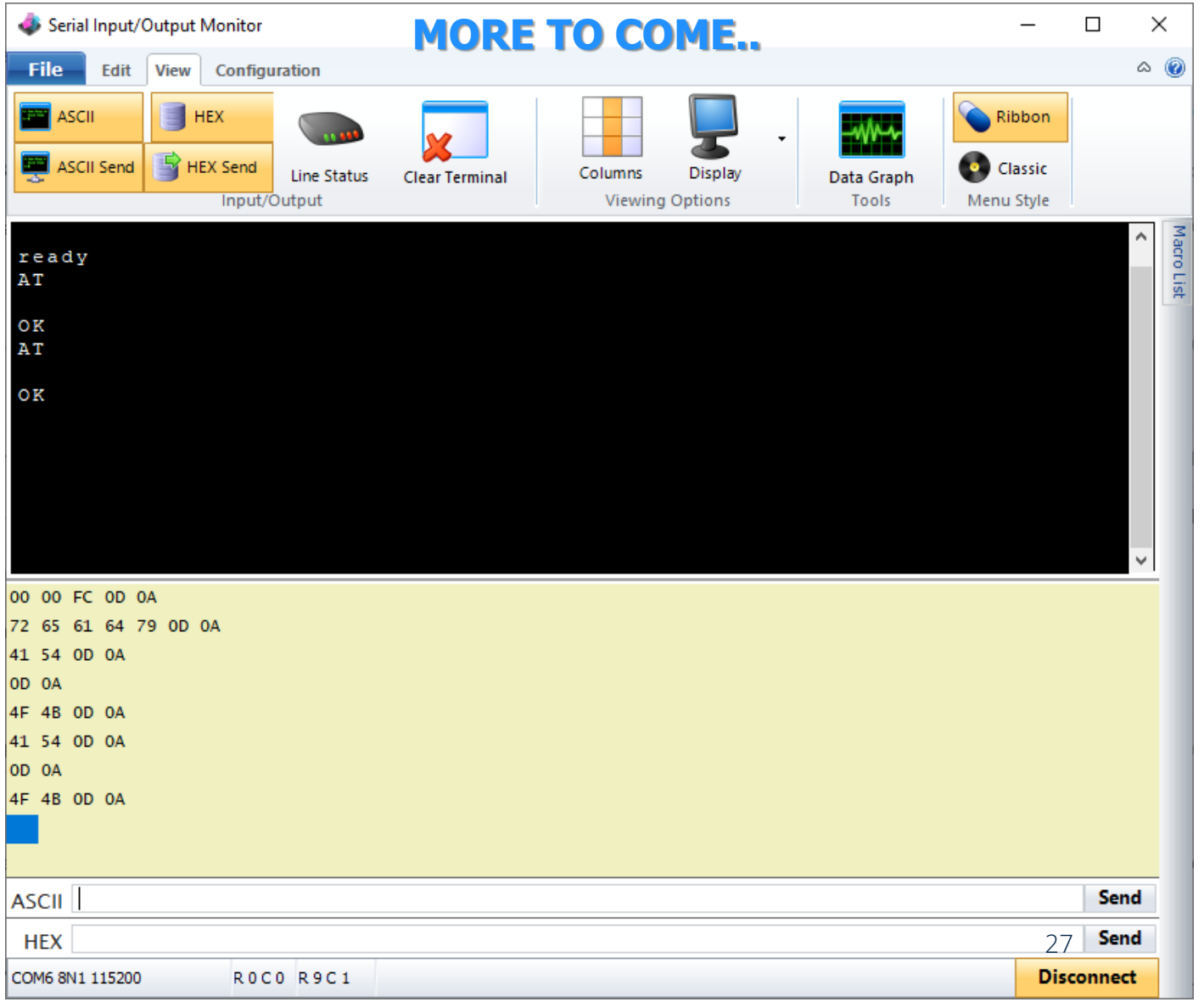
```c
#include "pico/stdlib.h"

#define UART0_ID   uart0
#define BAUD_RATE 115200
#define DATA_BITS 8
#define STOP_BITS 1
#define PARITY    UART_PARITY_NONE

// define uart pins
#define UART0_TX_PIN 28
#define UART0_RX_PIN 29
#define LED_PIN      24

int main()
{
    // Initialize LED pin
    gpio_init(LED_PIN);
    gpio_set_dir(LED_PIN, GPIO_OUT);

    // Initialize serial port
    gpio_set_function(UART0_TX_PIN, GPIO_FUNC_UART);
    gpio_set_function(UART0_RX_PIN, GPIO_FUNC_UART);
    uart_init(UART0_ID, 115200);
    uart_set_hw_flow(UART0_ID, false, false);
    uart_set_format(UART0_ID, DATA_BITS, STOP_BITS, PARITY);
    uart_set_fifo_enabled(UART0_ID, false);

   while (1)
    {
        uart_puts(UART0_ID,"LED should be blinking...\r\n");
        gpio_put(LED_PIN, 1);
        busy_wait_ms(200);
        gpio_put(LED_PIN, 0);
        busy_wait_ms(200);
    }
}
```



Thank you for attending!!!

Please consider the resources below:

- raspberrypi.org
- RP2040 Datasheet
- Raspberry Pi Pico C/C++ SDK
- SEGGER J-Link



CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

Digi-Key ELECTRONICS

informa markets