CEC Continuing Education Center

DesignNews

Raspberry Pi Pico/Pico2 Development Using Visual Studio Code

Day 5:

Raspberry Pi Pico 2 Coprocessor

Sponsored by

DigiKey

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.

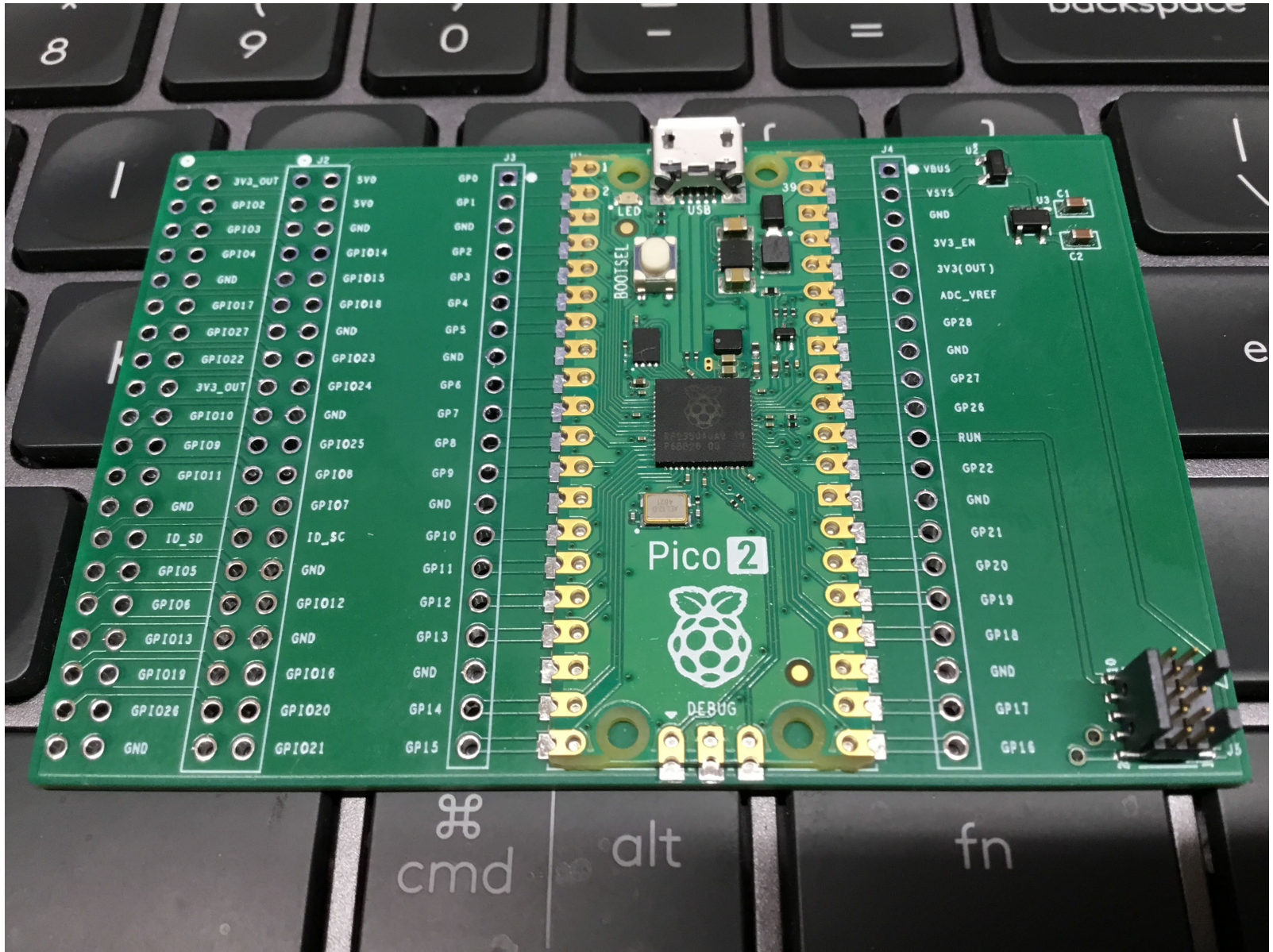
Fred Eady

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

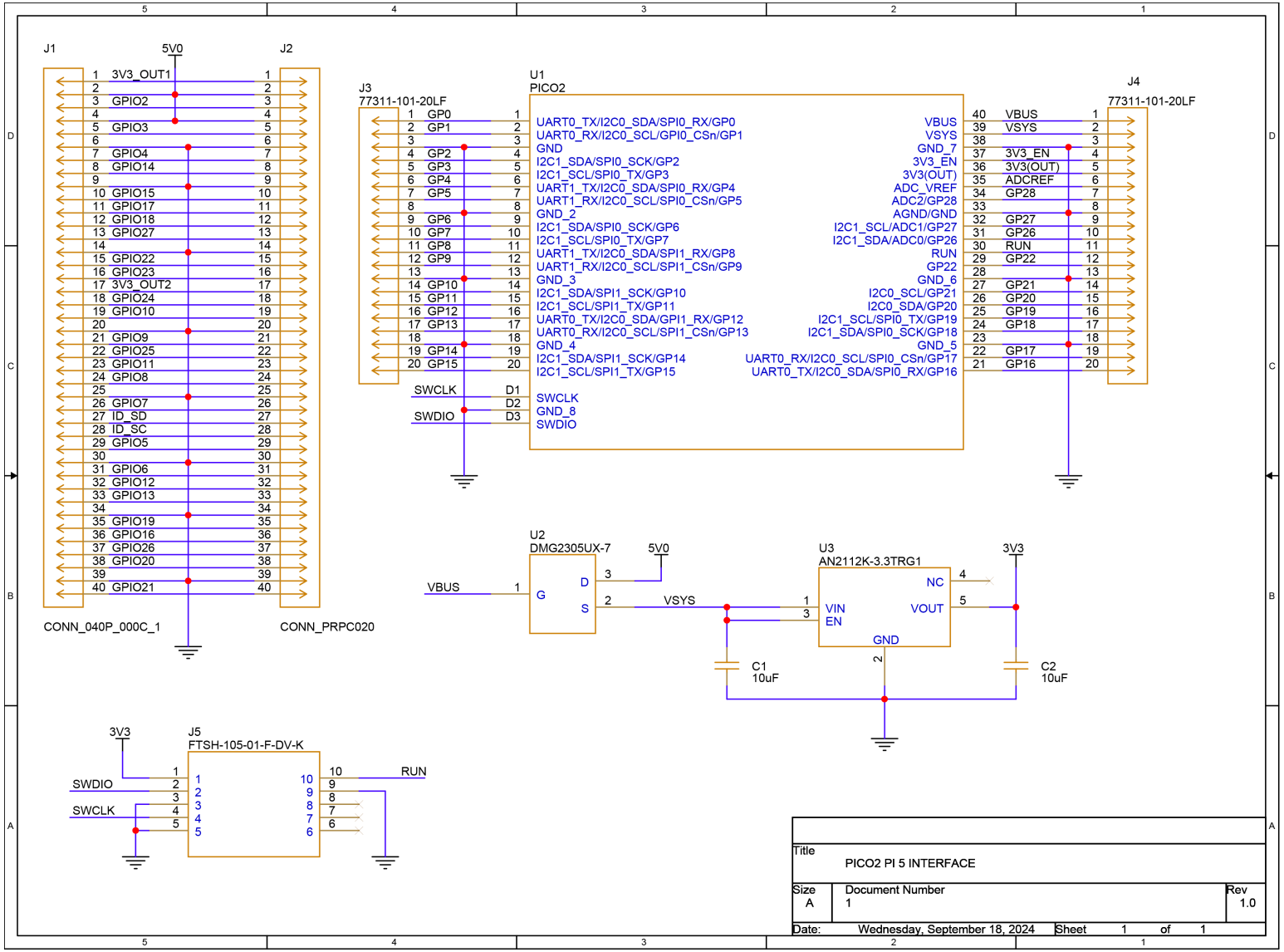
AGENDA

- Pico 2 Coprocessor Hardware
- Connect the Raspberry Pi Debug Probe
- Create a Pico 2 ADC Application
- Integrate the Pico 2 Coprocessor and Raspberry Pi 5

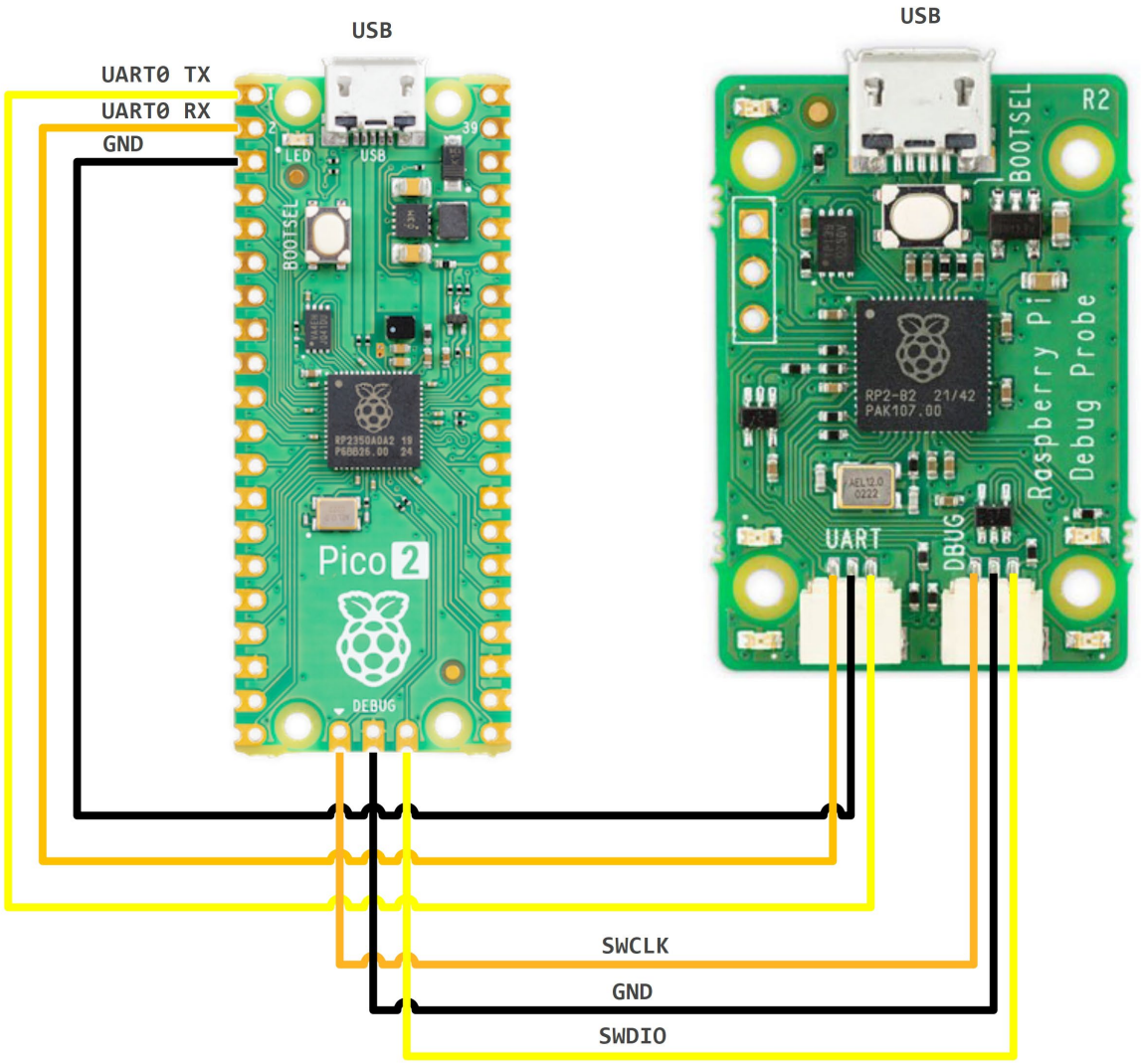
Pico 2 Coprocessor Board

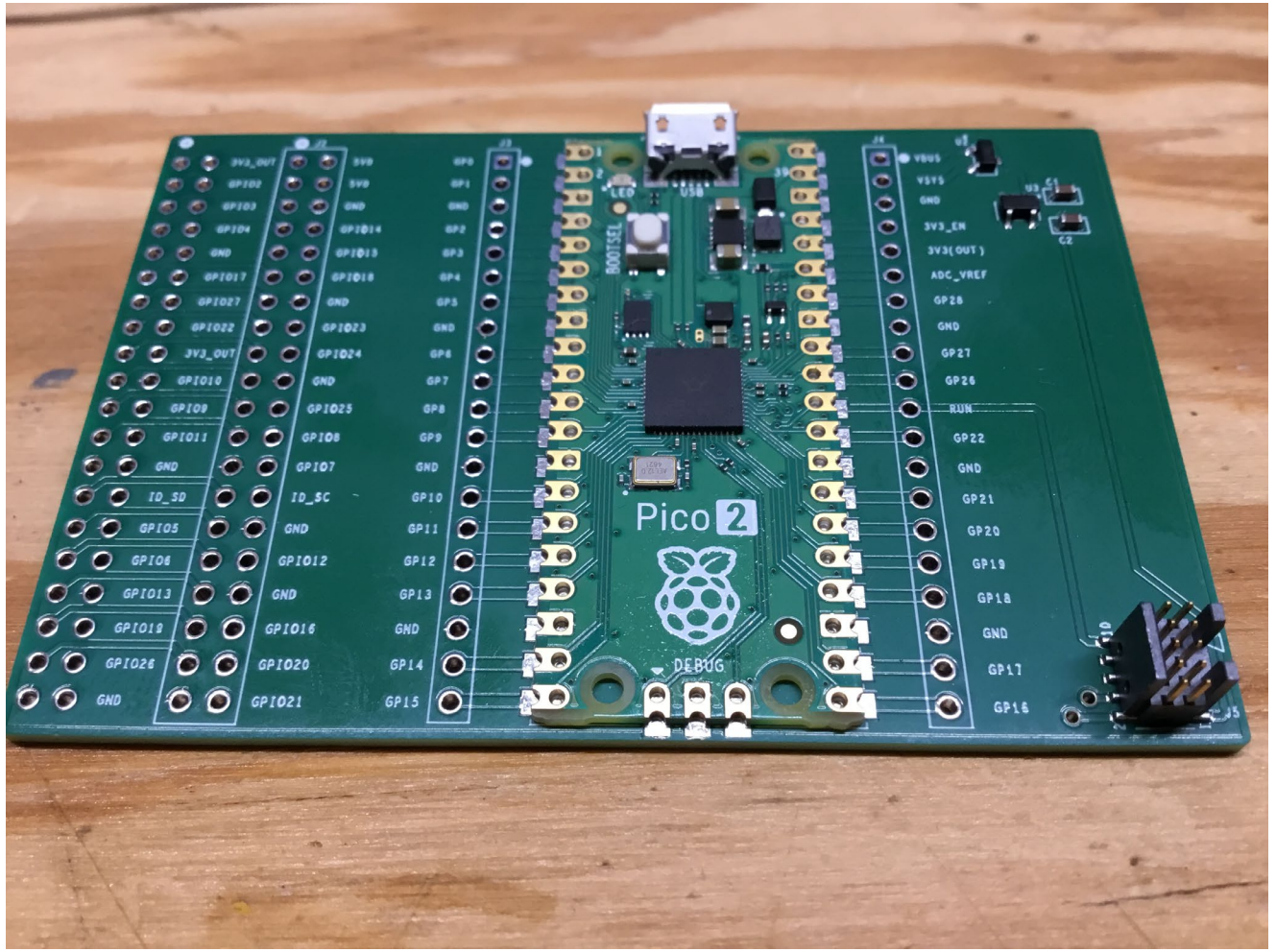


Pico 2 Coprocessor Board

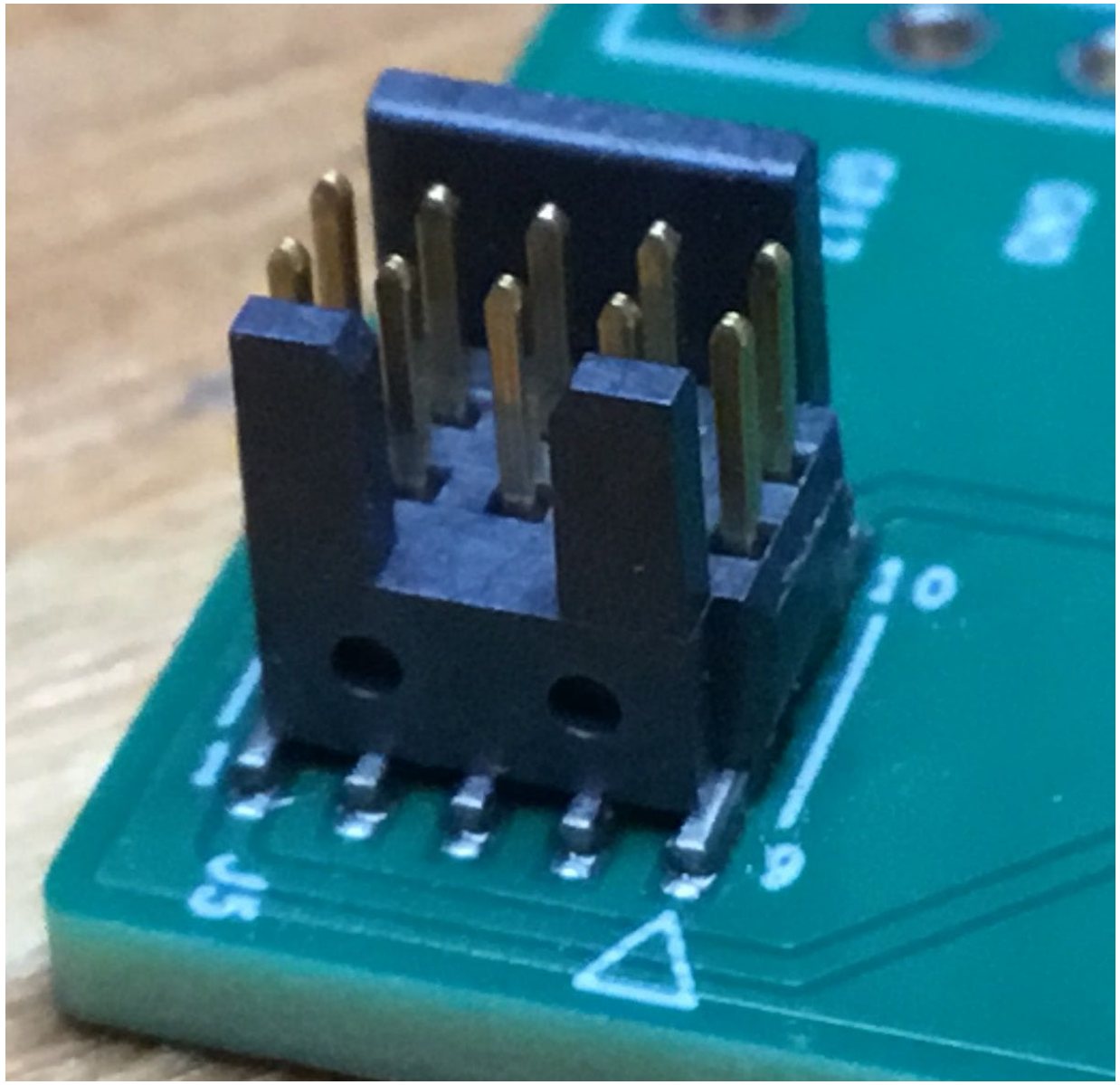


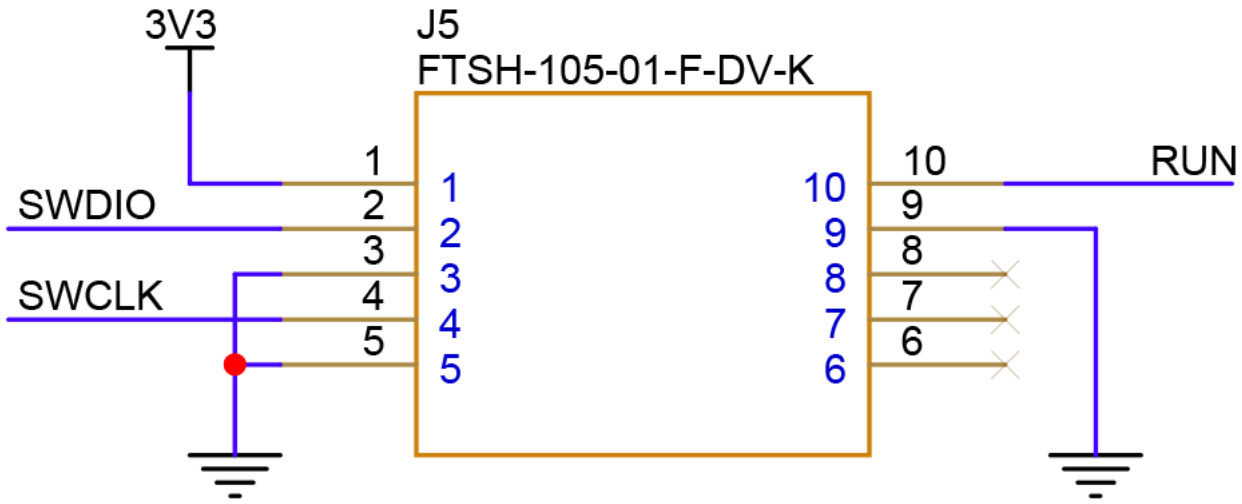
Debug Probe Interface Wiring



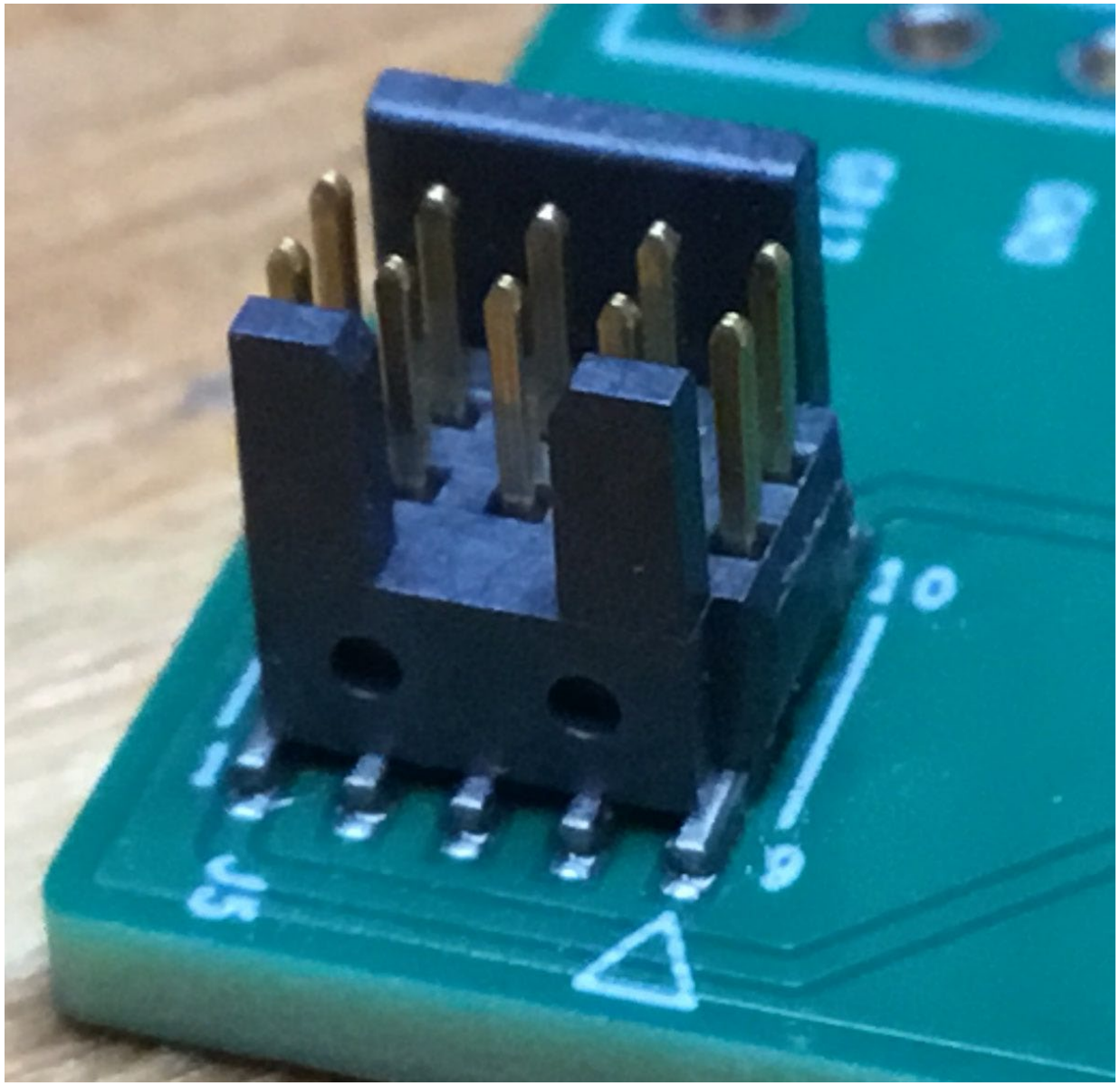


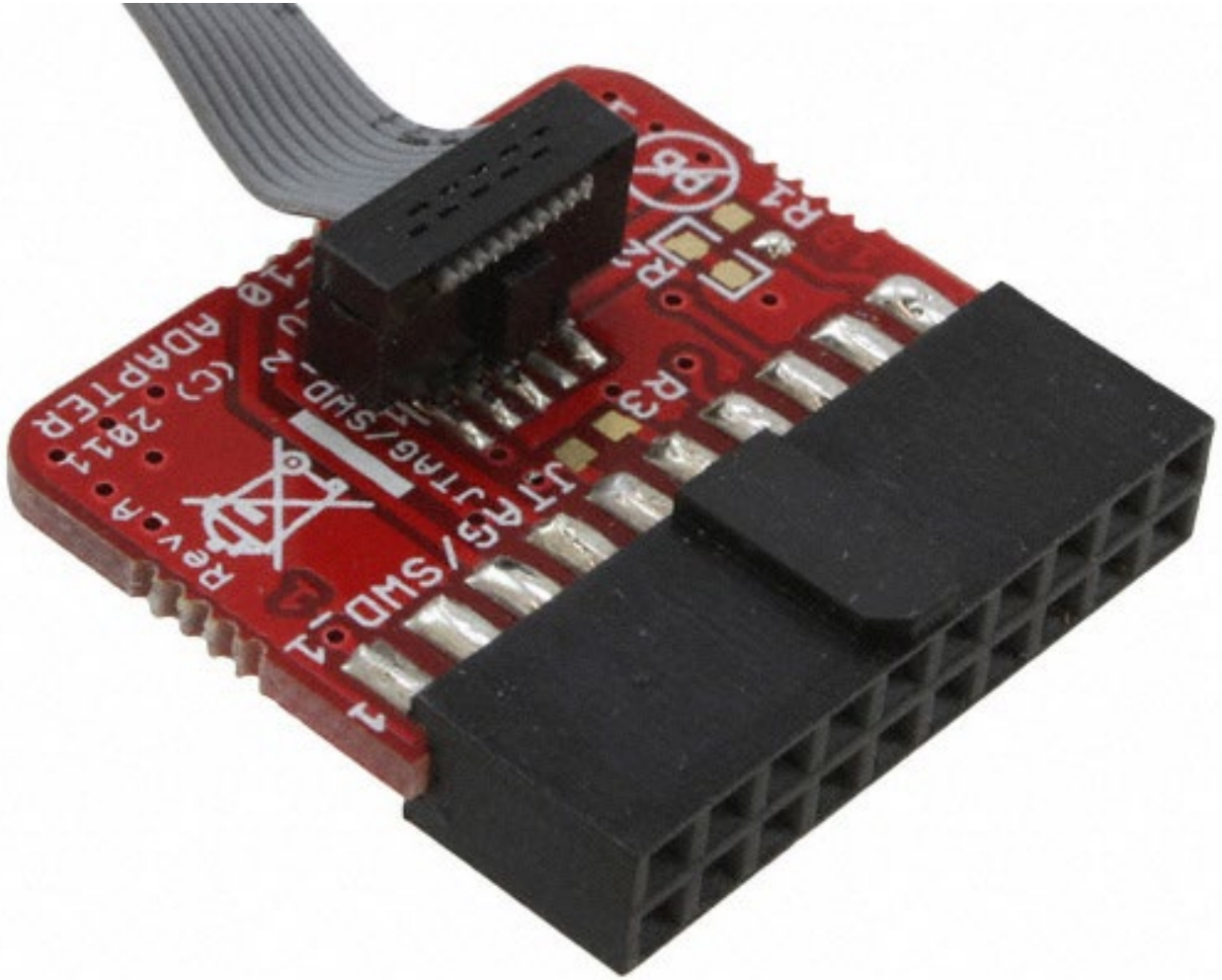
10-Pin SWD Programming/Debugging Interface



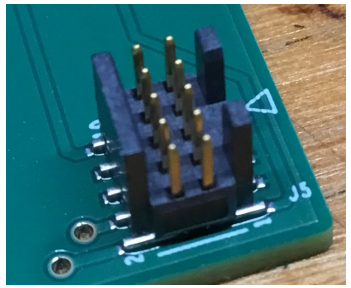


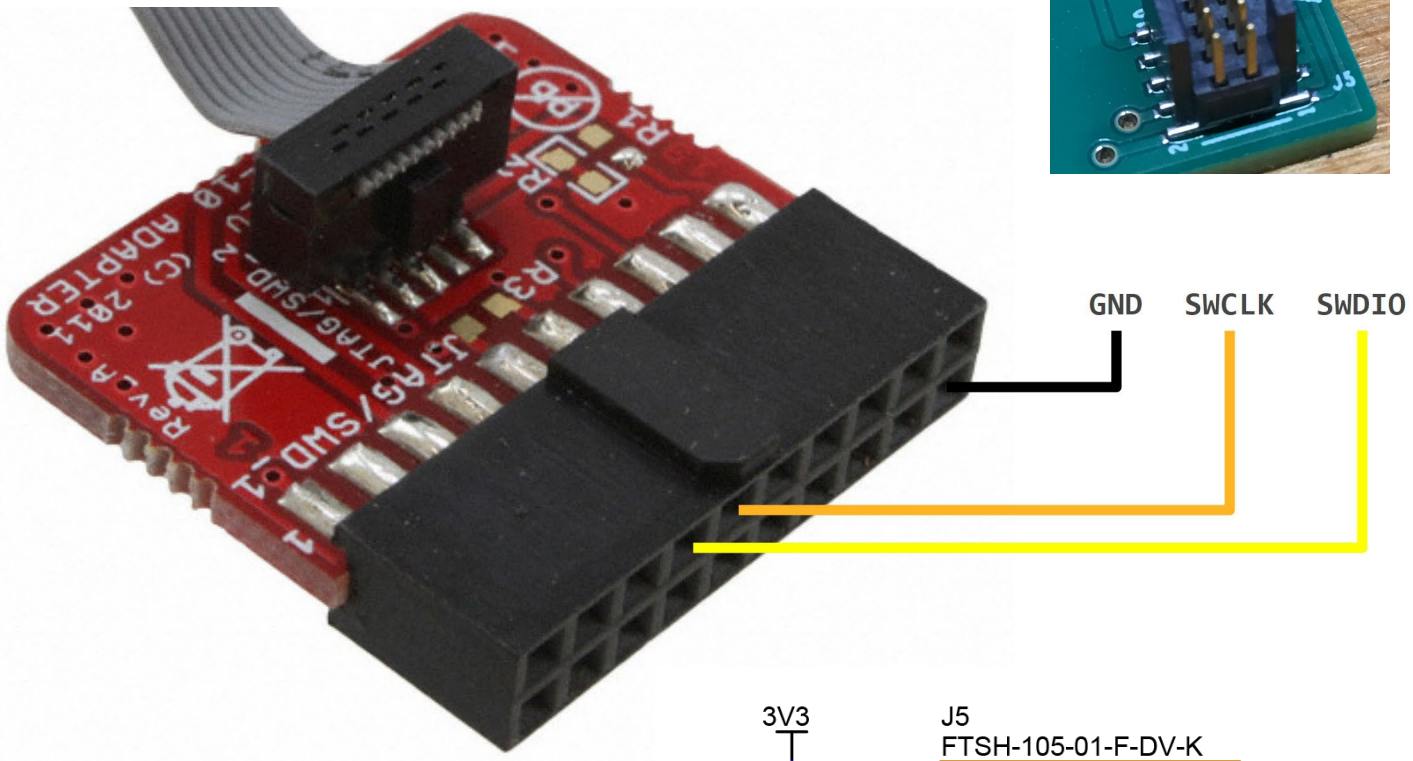
10-Pin SWD Programming/Debugging Interface

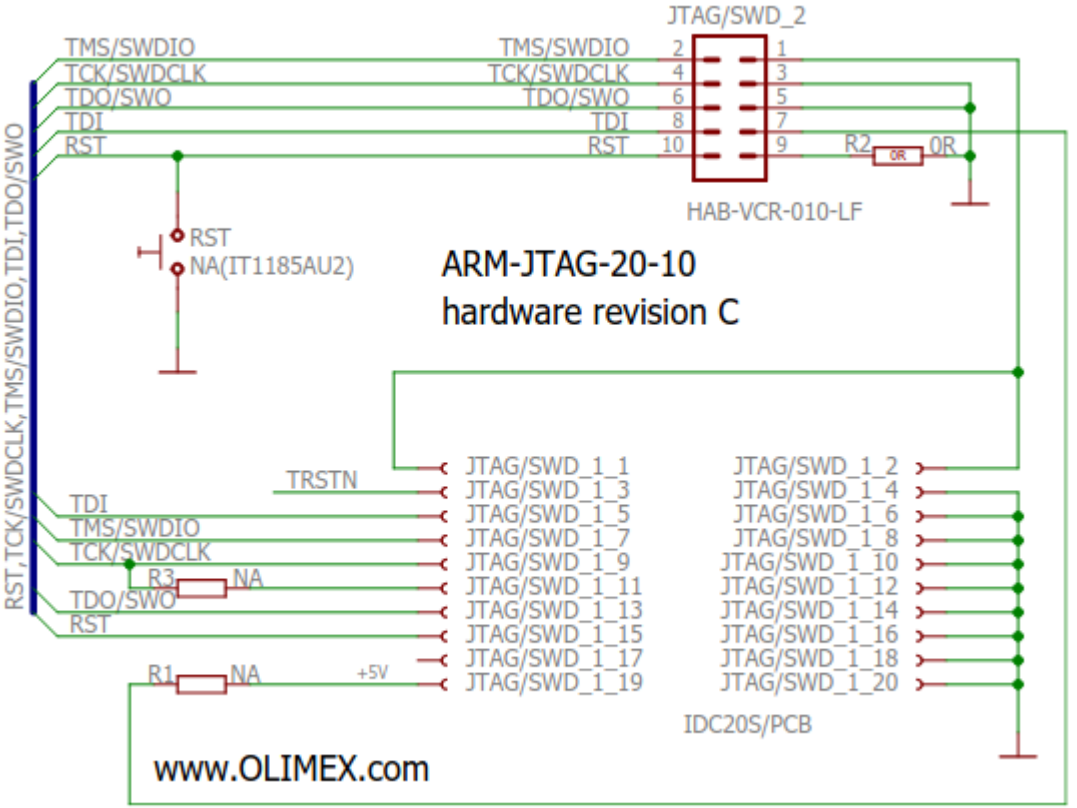


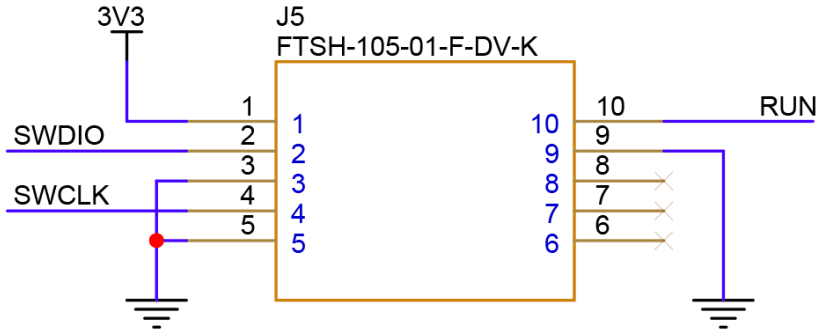


JTAG/SWD Interface

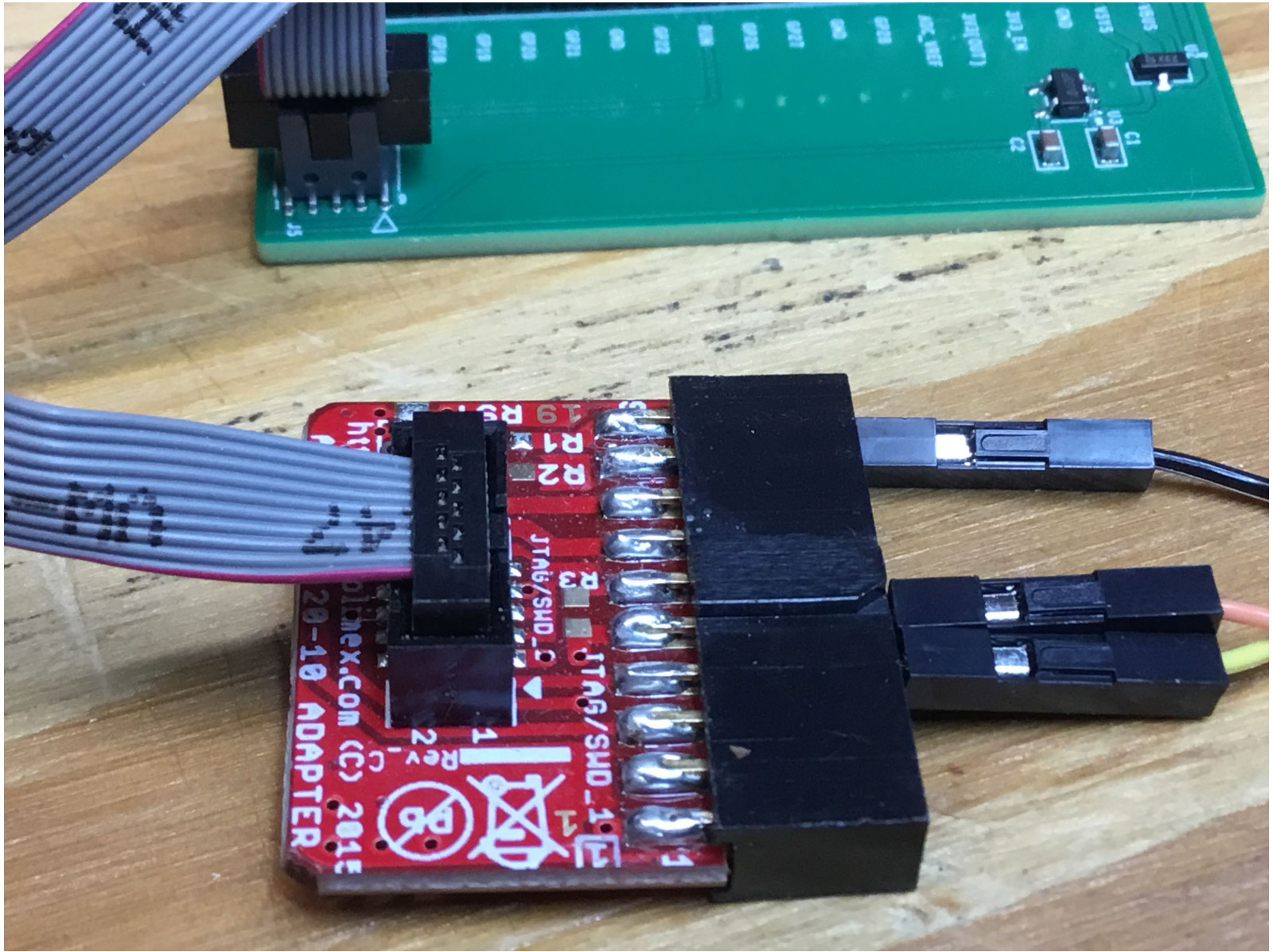




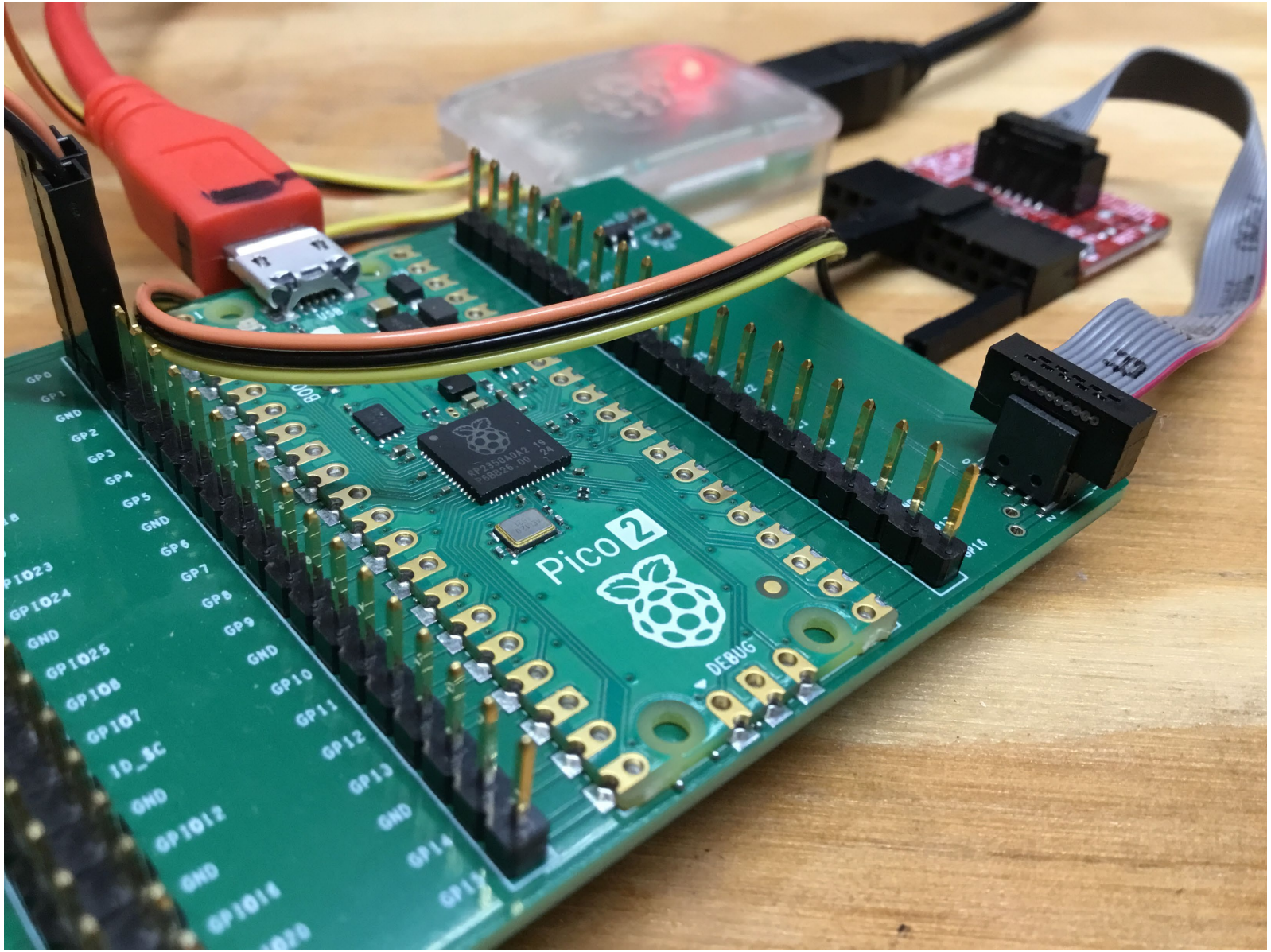




10-Pin SWD Programming/Debugging Interface



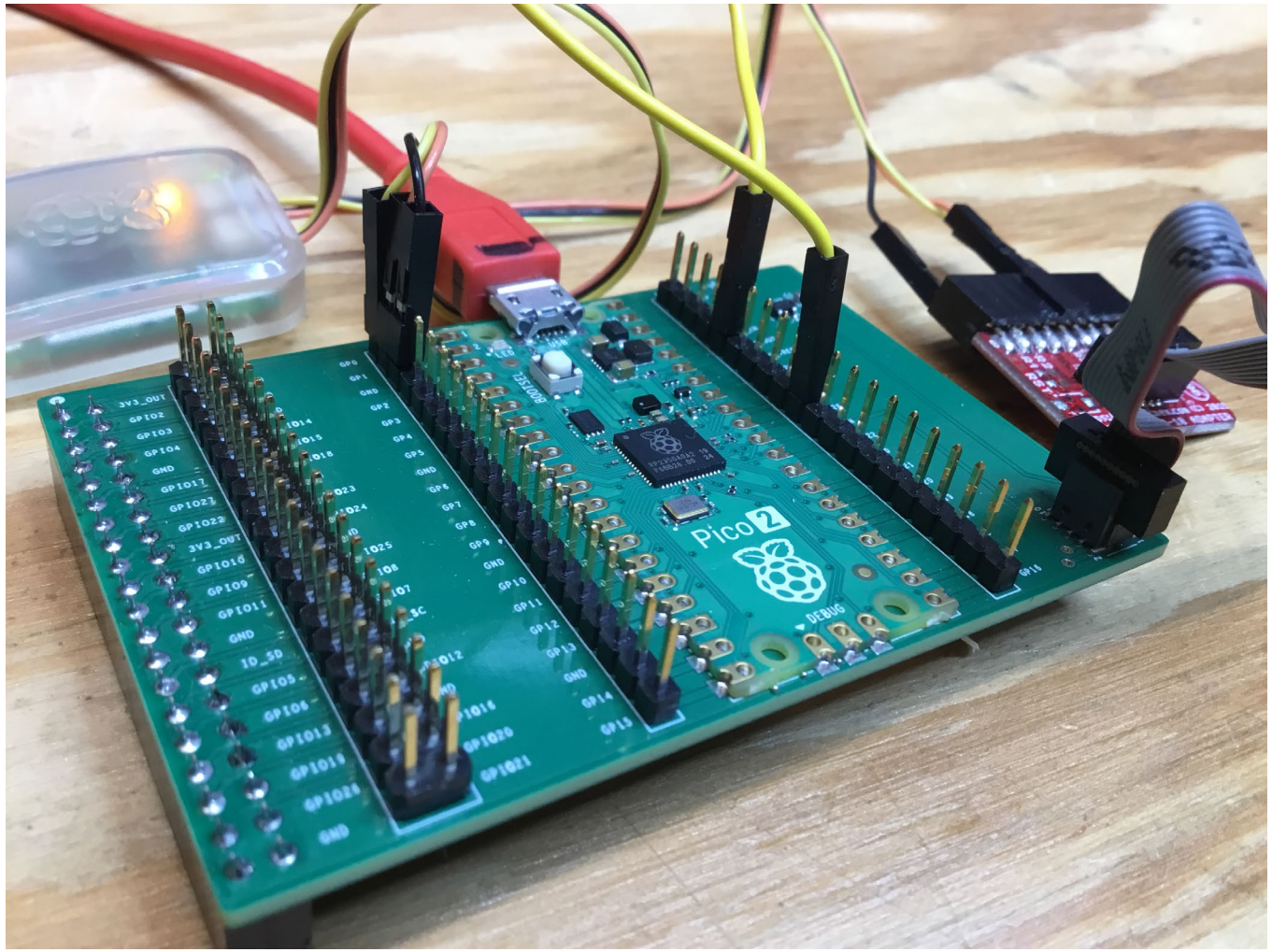
10-Pin SWD Programming/Debugging Interface



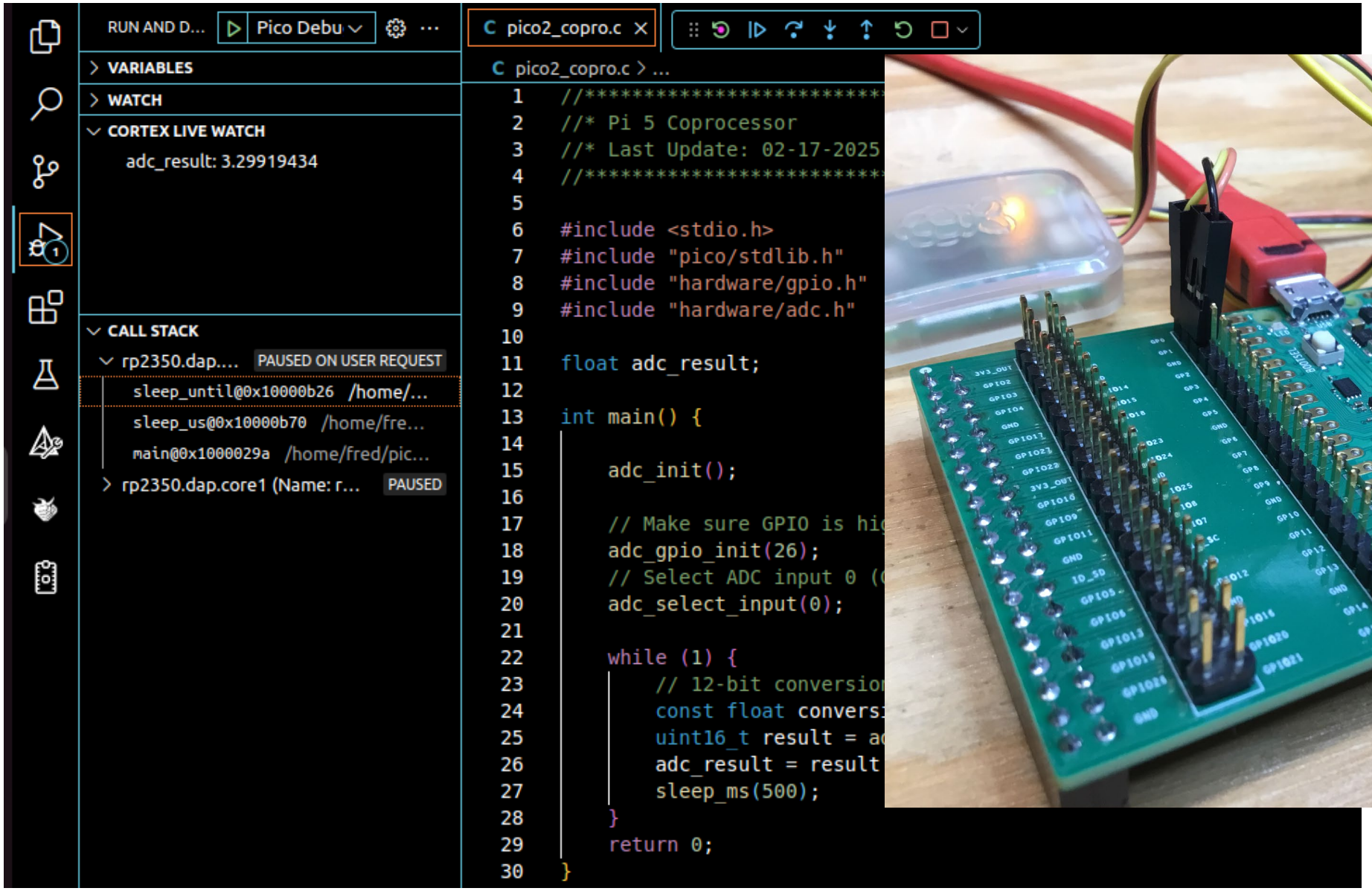
Code the Pico 2 ADC Application

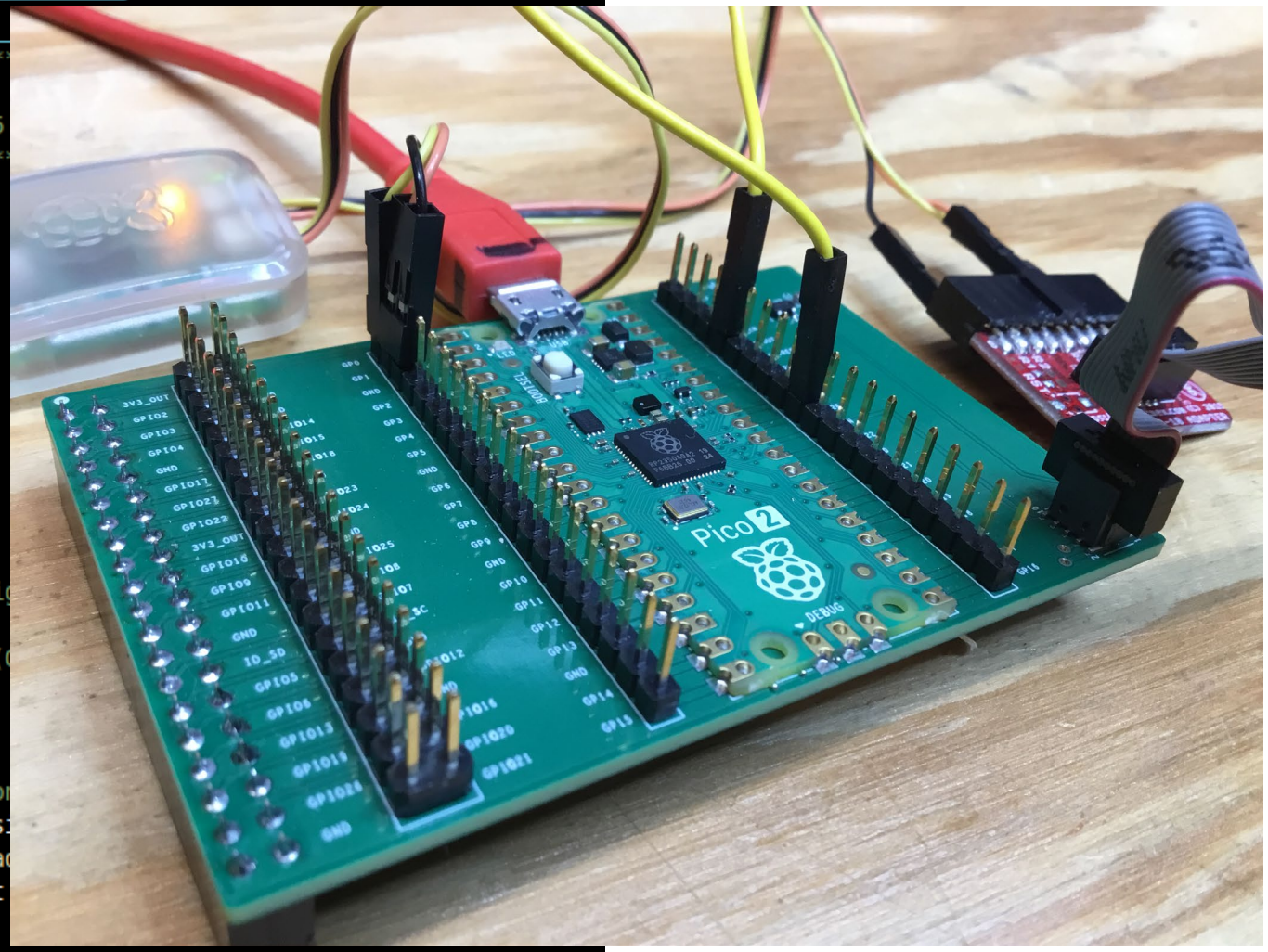
pico2_copro.c

```c
//*****************************************************************
//* Pi 5 Coprocessor
//* Last Update: 02-17-2025
//*****************************************************************

#include <stdio.h>
#include "pico/stdlib.h"
#include "hardware/gpio.h"
#include "hardware/adc.h"

float adc_result;

int main() {

    adc_init();

    // Make sure GPIO is high-impedance, no pullups etc
    adc_gpio_init(26);
    // Select ADC input 0 (GPIO26)
    adc_select_input(0);

    while (1) {
        // 12-bit conversion, assume max value == ADC_VREF == 3.3 V
        const float conversion_factor = 3.3f / (1 << 12);
        uint16_t result = adc_read();
        adc_result = result * conversion_factor;
        sleep_ms(500);
    }
    return 0;
}
```



Debug the Pico 2 ADC Application

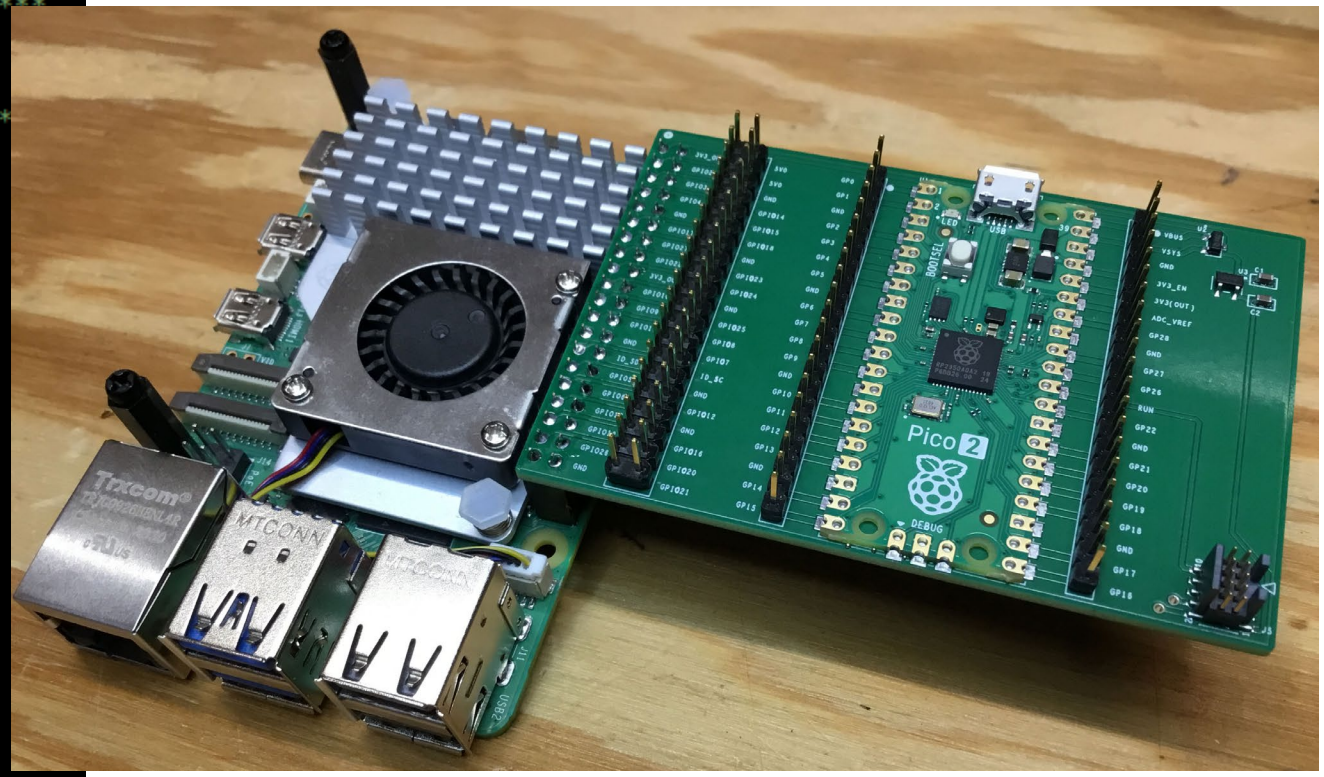




Code the Pico 2 Coprocessor Application

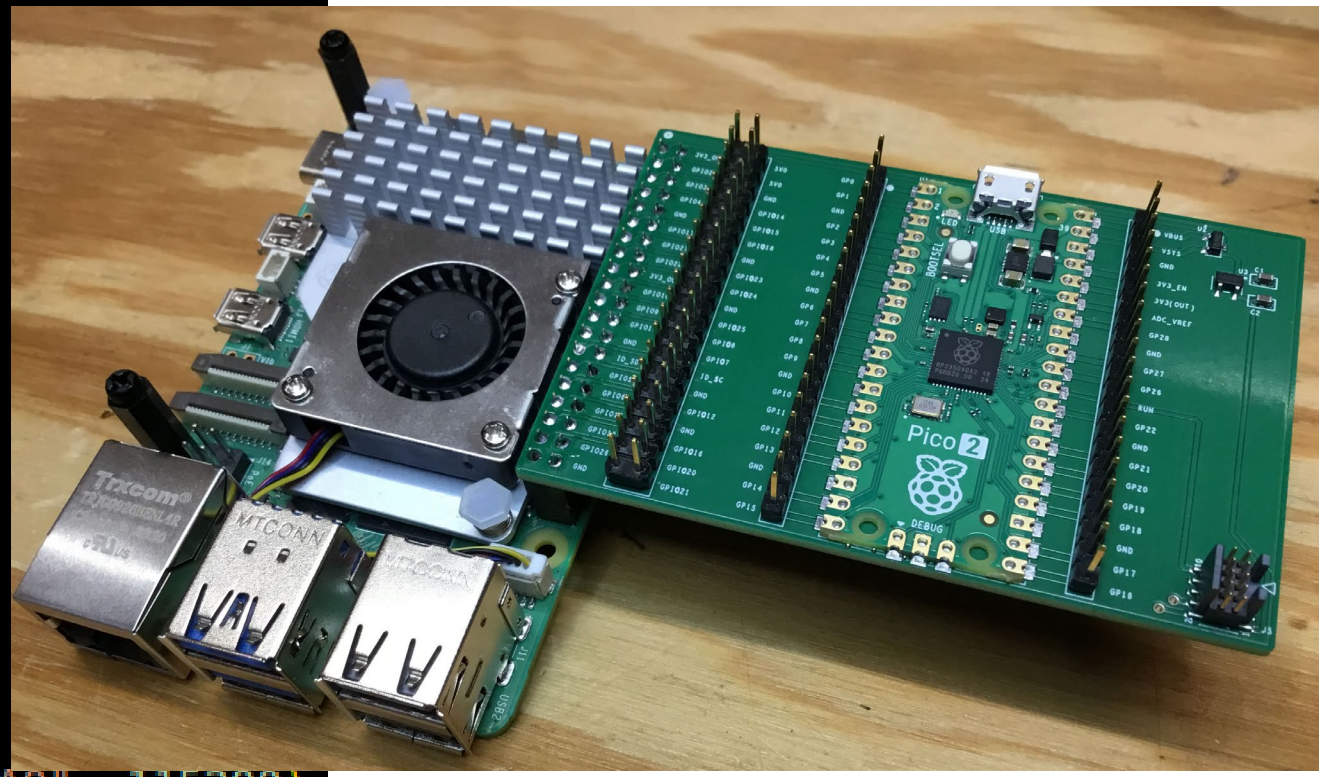
pico2_copro.c

```c
//*****************************************************************
//* Pi 5 Coprocessor
//* Last Update: 02-17-2025
//*****************************************************************

#include <stdio.h>
#include "pico/stdlib.h"
#include "hardware/gpio.h"
#include "hardware/adc.h"

int main() {
    stdio_init_all();

    adc_init();

    // Make sure GPIO is high-impedance, no pullups etc
    adc_gpio_init(26);
    // Select ADC input 0 (GPIO26)
    adc_select_input(0);

    while (1) {
        // 12-bit conversion, assume max value == ADC_VREF == 3.3 V
        const float conversion_factor = 3.3f / (1 << 12);
        uint16_t result = adc_read();
        printf("%1.3f", result * conversion_factor);
        sleep_ms(500);
    }
}
```

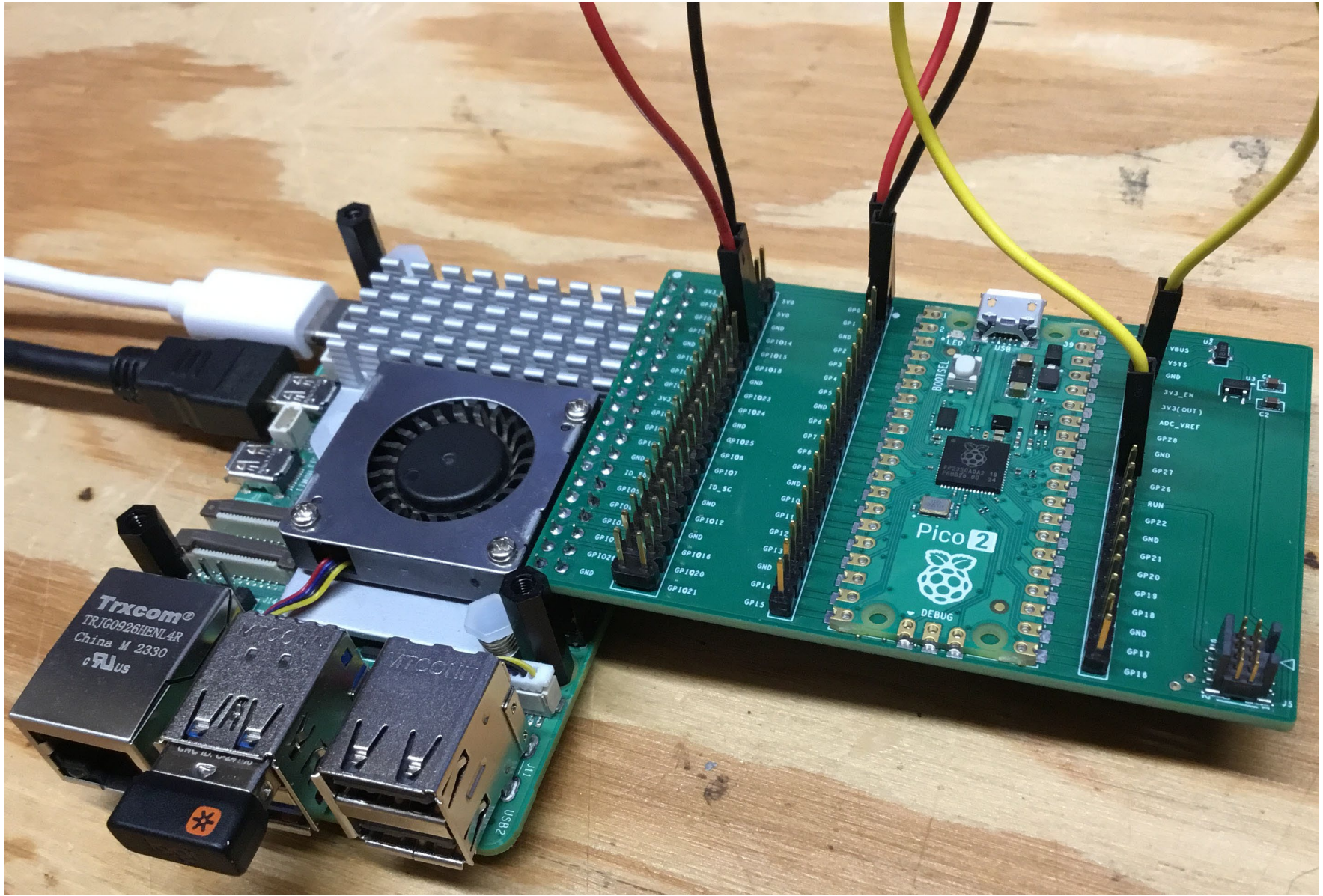


Code the Raspberry Pi 5 Coprocessor Application

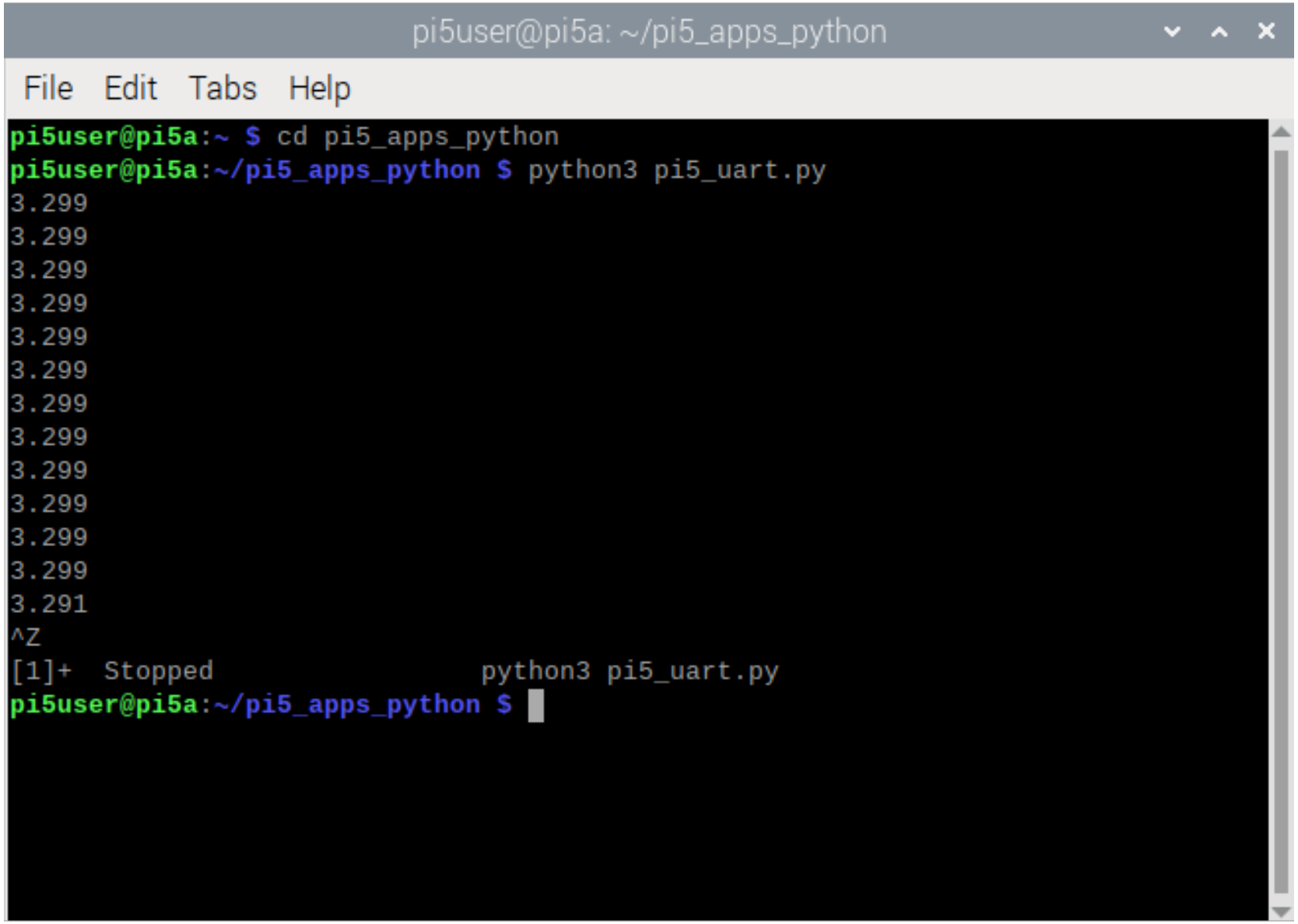
```
1   #**************************************
2   # Raspberry Pi 5 Coprocessor
3   # Written by: Fred Eady
4   # Last Update: 09-19-2024
5   # Notes:
6   #**************************************
7   # TXD = GPIO 14
8   # RXD = GPIO 15
9   #**************************************
10
11  import serial
12  import time
13
14
15  # Open the Pi 5 serial port
16  serialPort = serial.Serial("/dev/ttyAMA0", 115200)
17
18  while True:
19      adcVal = serialPort.read(5)
20      print(adcVal.decode('ASCII'))
```



Configure the Pico 2 Coprocessor Hardware



Run the Pico 2 and Raspberry Pi Coprocessor Applications



Thank you for attending!!!

Please consider the resources below:

- Today's Download Package
- raspberrypi.org
- segger.com

LET'S EAT..

CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

DigiKey

informa markets