



DesignNews

Raspberry Pi 5 Automation Lecture Series

Day 5:

Driving Servos with the Raspberry Pi 5

Sponsored by

DigiKey



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.

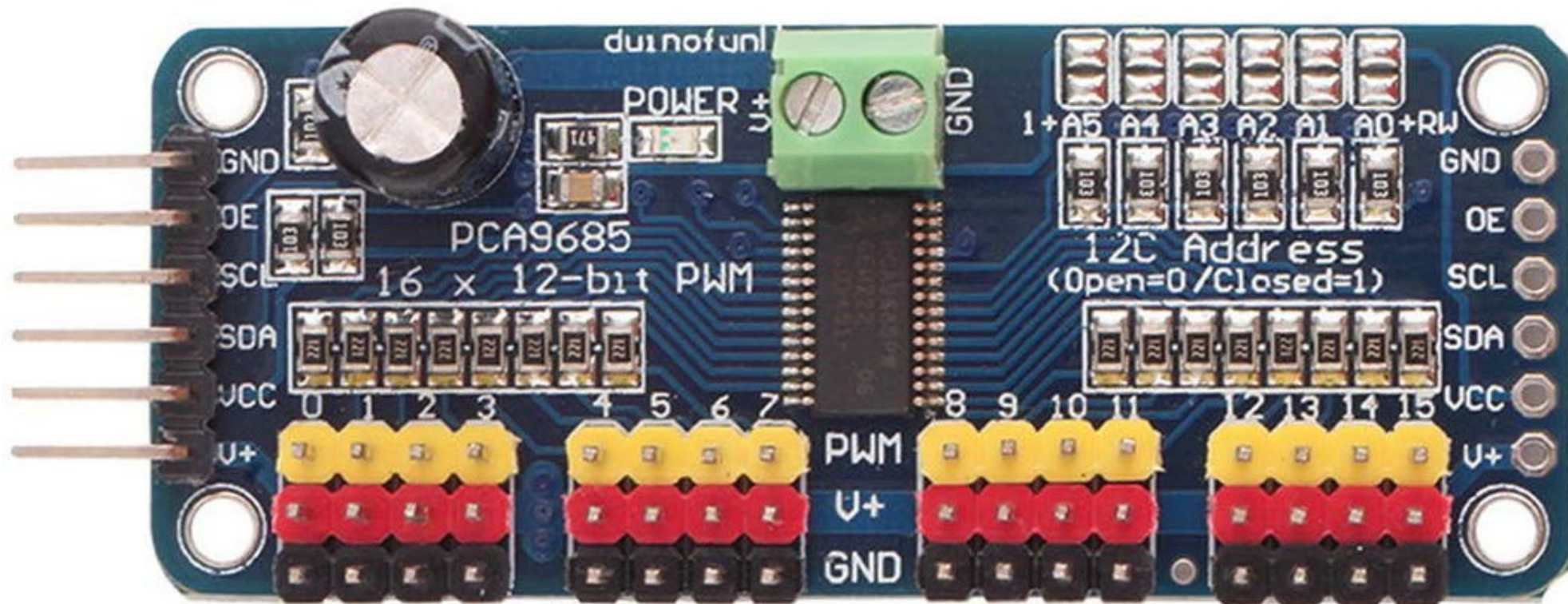


Fred Eady

Visit 'Lecturer Profile' in your console for more details.

AGENDA

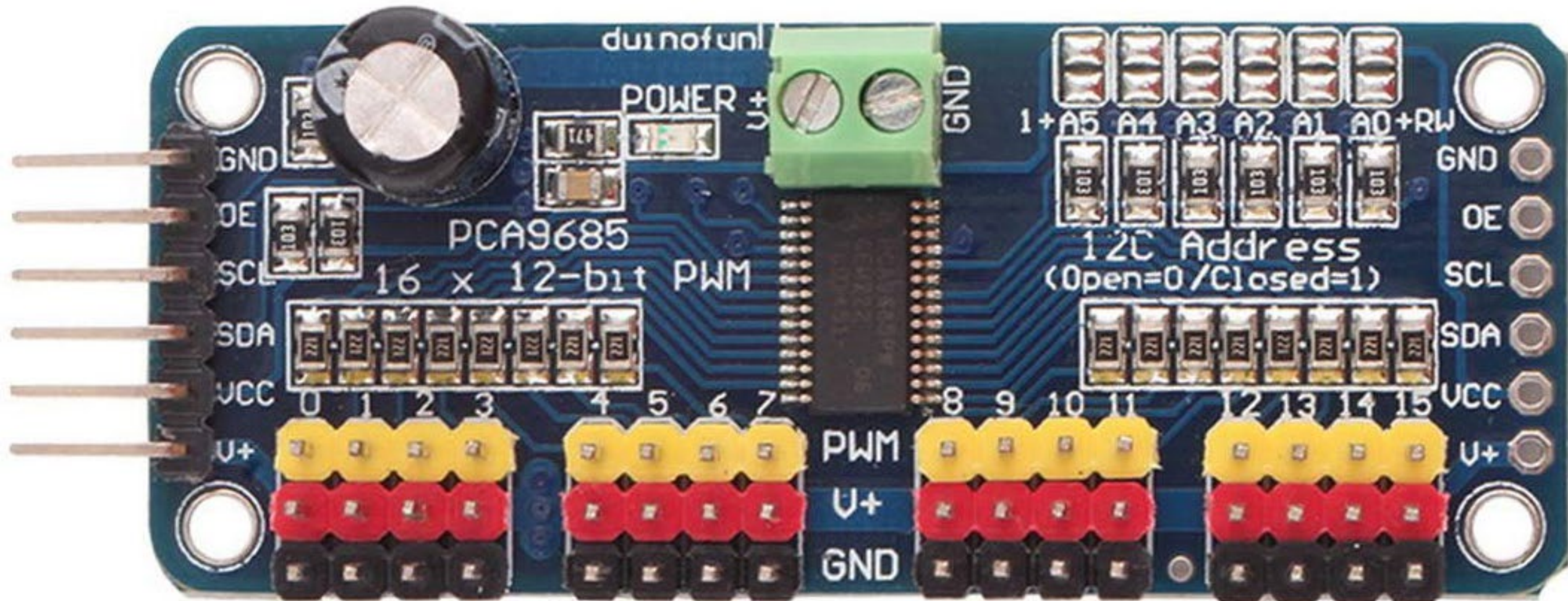
- **Raspberry Pi 5 Servo Drive “Helper” Hardware**
- **Raspberry Pi 5 Servo Drive TCP/IP Client Software**
- **Raspberry Pi 5 Servo Drive “Helper” Firmware**



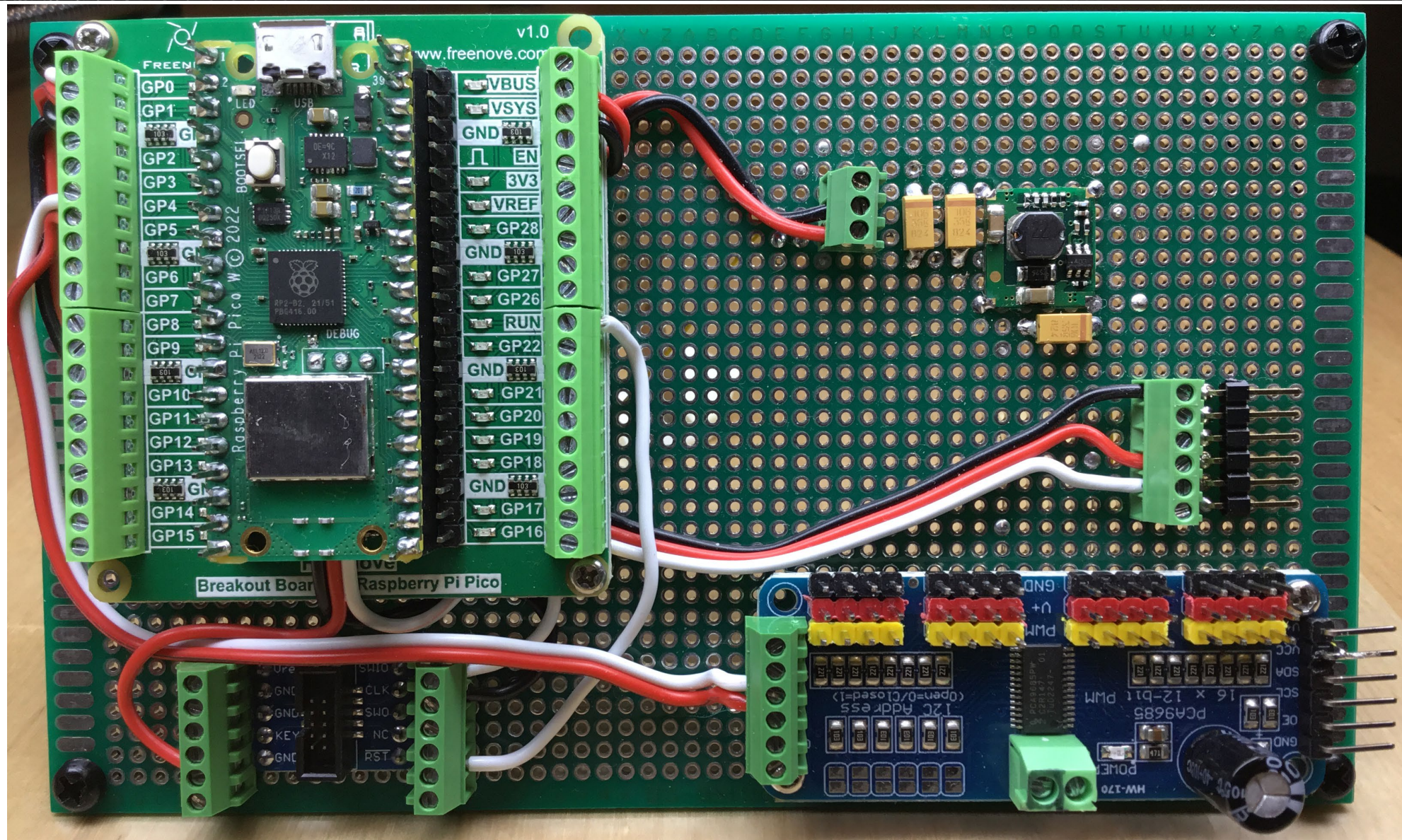
Servo Driver Development Layout



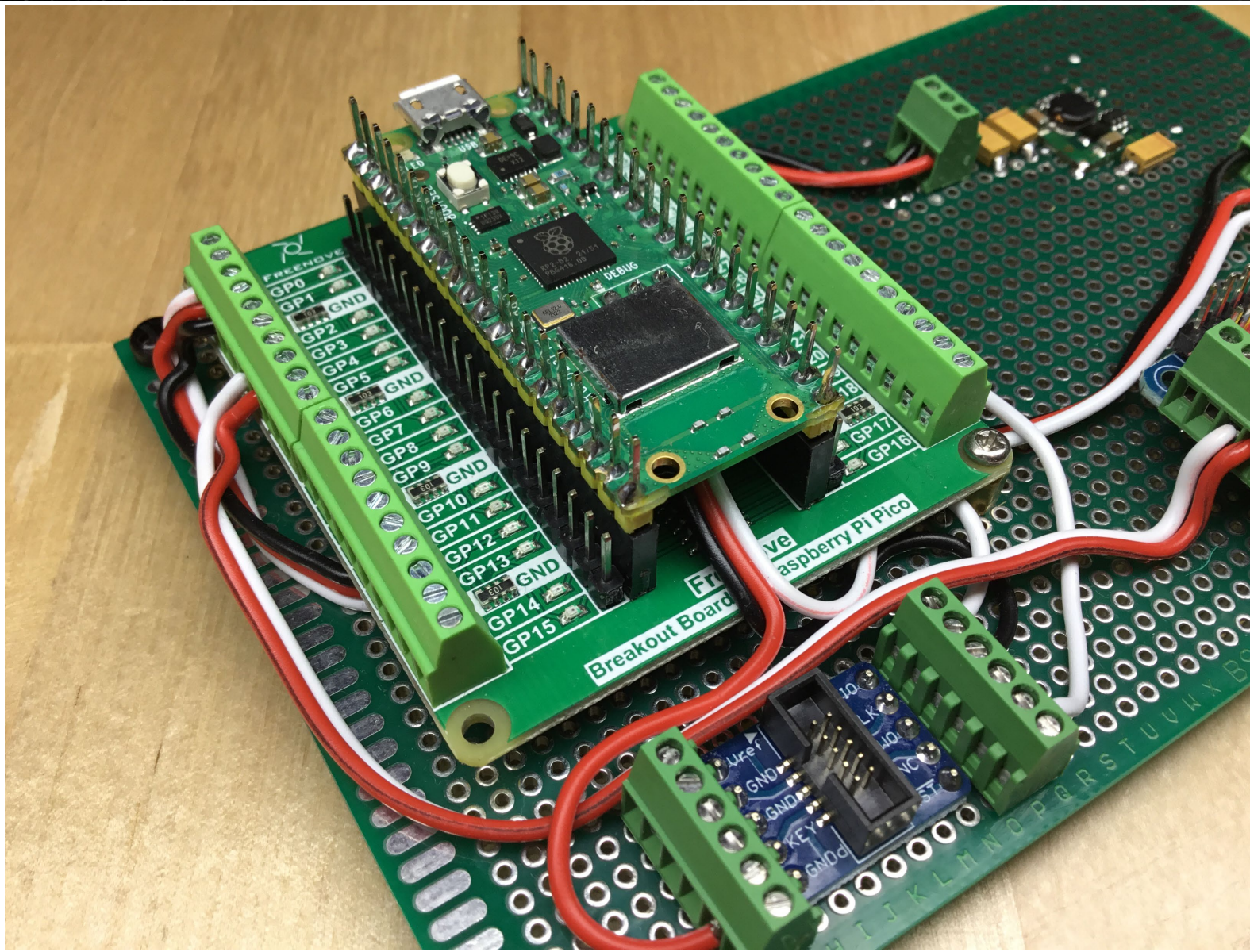
PCA9685



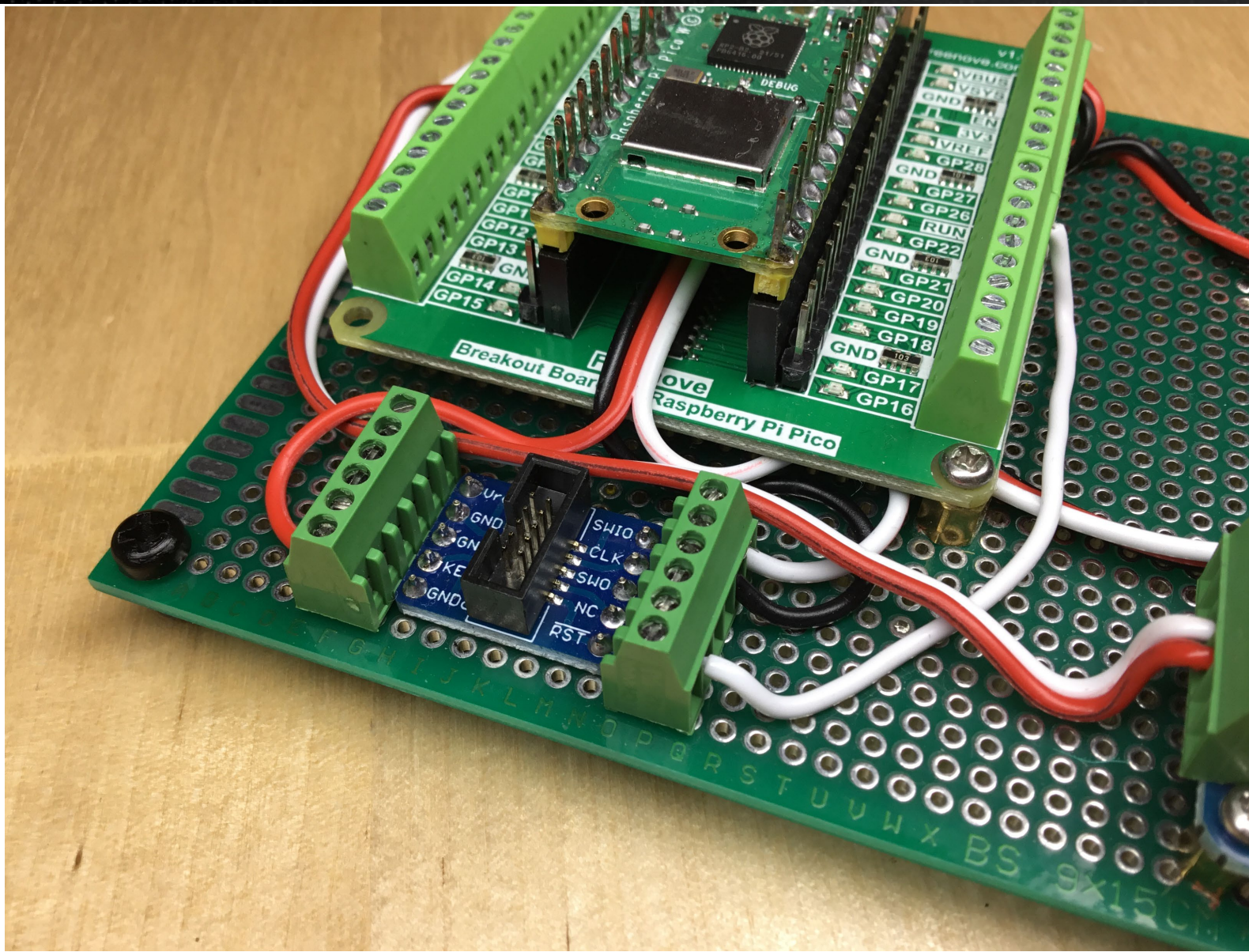
Raspberry Pi PICO W "Helper"



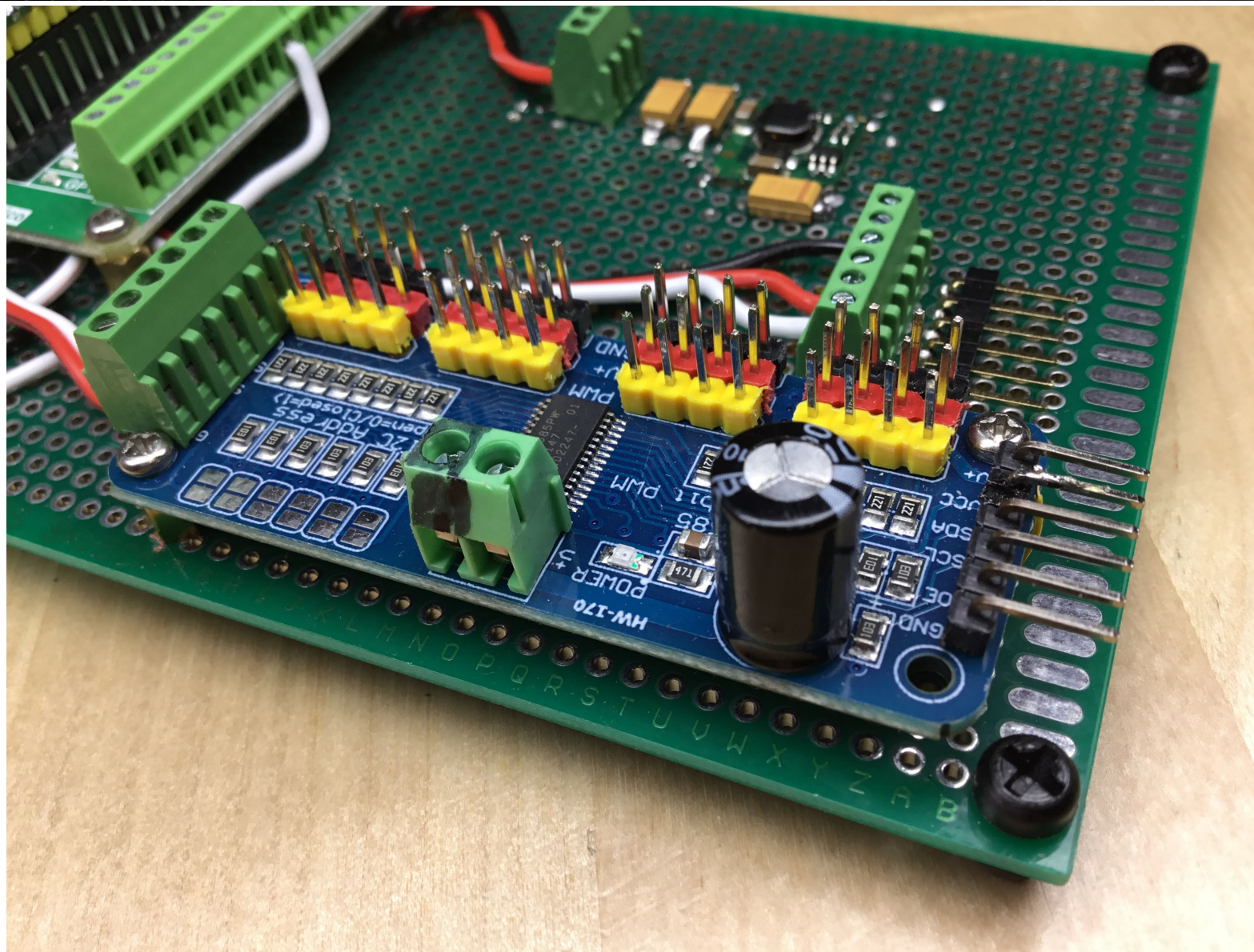
Raspberry Pi PICO W "Helper"



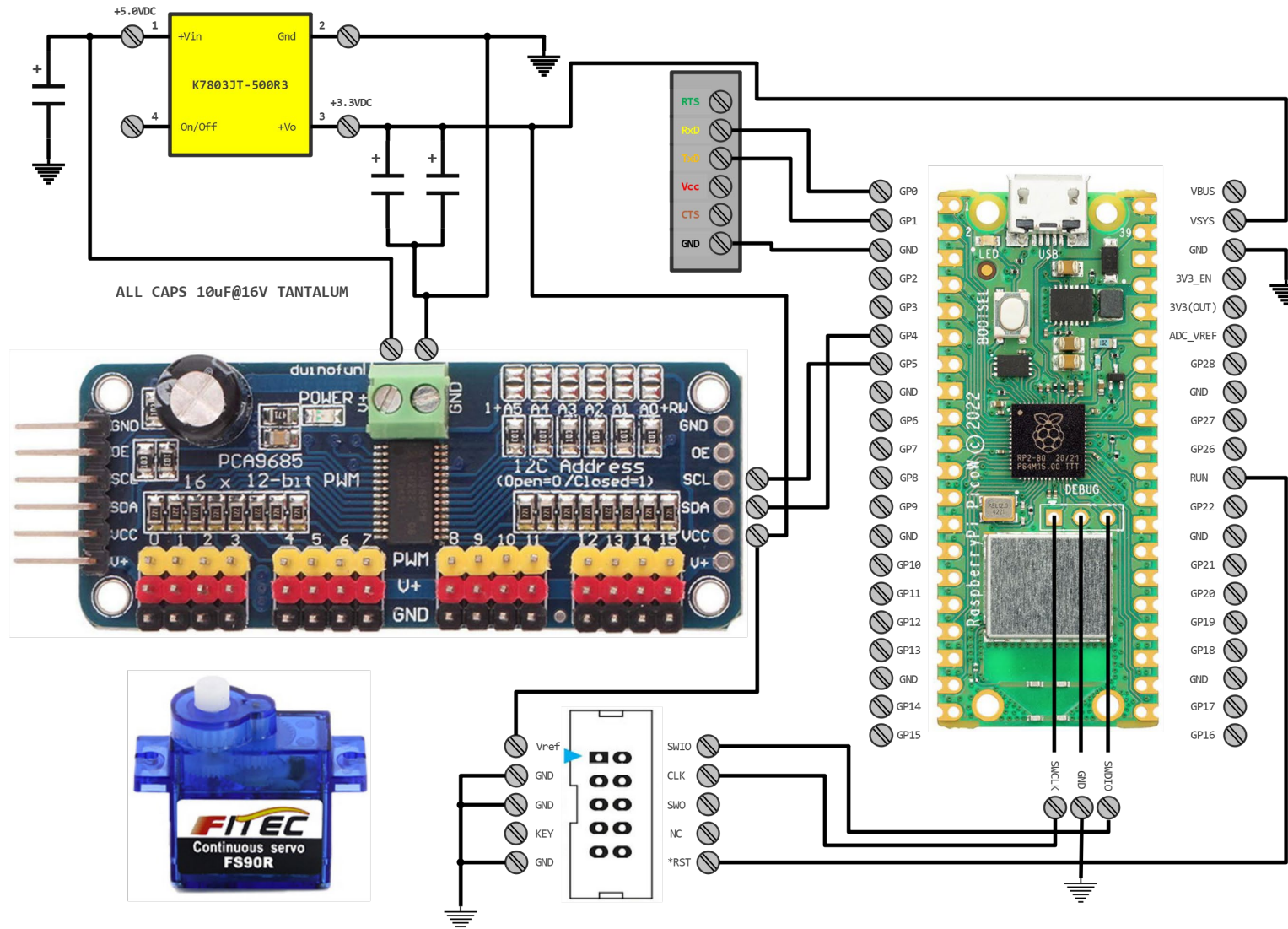
Raspberry Pi PICO W "Helper"



Raspberry Pi PICO W "Helper"



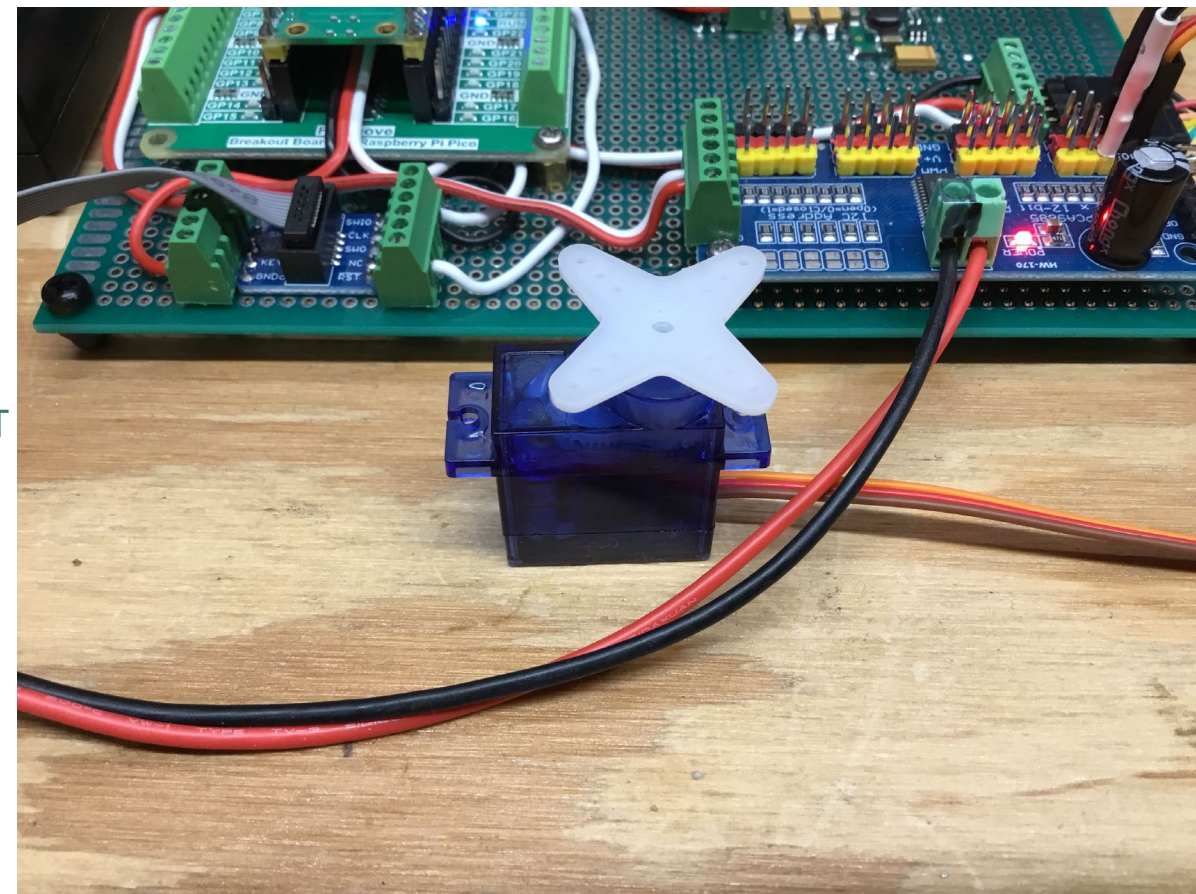
Raspberry Pi PICO W "Helper"



Raspberry Pi 5 Client Servo Driver – #includes

```
/** *****  
/** PCA9685 Servo Driver TCP/IP Client  
/** Written by: Fred Eady  
/** Last Updated: 02-11-2024  
/** Notes:  
/** *****  
/** Continuous Servo Command Settings  
/** Spin CLOCKWISE: 10 = FASTEST - 289 SLOWEST  
/** Spin COUNTERCLOCKWISE: 307 = SLOWEST - 500 = FASTEST  
/** STOP BETWEEN 290 AND 303 - 300 IS IDEAL
```

```
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <stdint.h>  
#include <sys/socket.h>  
#include <arpa/inet.h> //inet_addr  
#include <lgpio.h>
```



Raspberry Pi 5 Client Servo Driver – send_servo_cmd

```

char cmdVal[8];

void send_servo_cmd(char* servocmd)
{
    int socket_servo;
    struct sockaddr_in server;
    char cmdMsg[16];

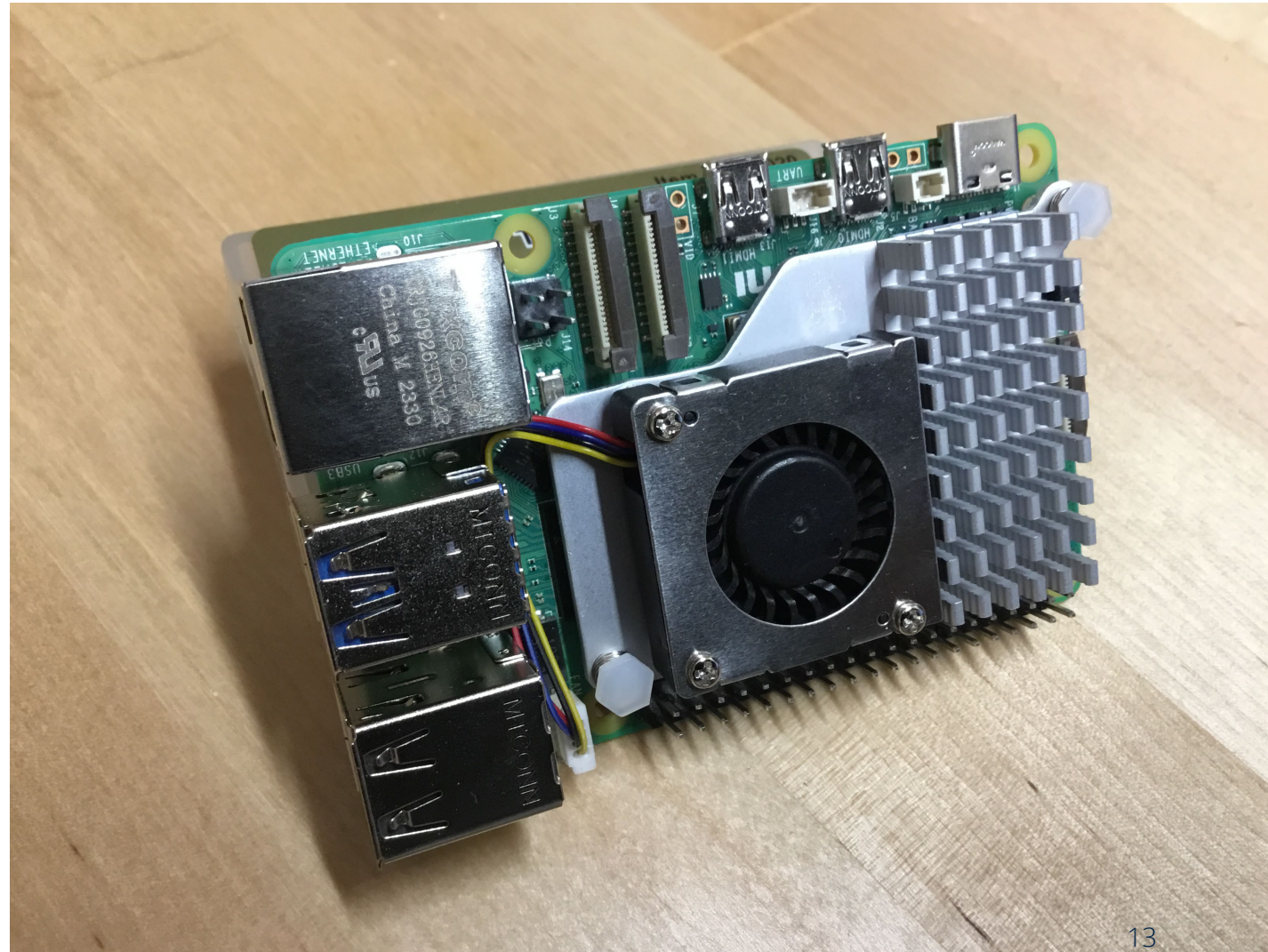
    memset(cmdMsg, 0x00, sizeof(cmdMsg));
    cmdMsg[0] = 'C'; // command packet header
    cmdMsg[1] = 'M';
    cmdMsg[2] = 'D';
    cmdMsg[3] = '.';
    cmdMsg[4] = cmdVal[0]; // servo number (0-1)
    cmdMsg[5] = cmdVal[1]; // servo number (0-5)
    cmdMsg[6] = cmdVal[2]; // servo command data MSByte
    cmdMsg[7] = cmdVal[3];
    cmdMsg[8] = cmdVal[4]; // servo command data LSByte
    cmdMsg[9] = '\r';      // end of command packet
    // Create socket
    socket_servo = socket(AF_INET , SOCK_STREAM , 0);
    if (socket_servo == -1) printf("Could not create socket");

    server.sin_addr.s_addr = inet_addr("192.168.1.132");
    server.sin_family = AF_INET;
    server.sin_port = htons( 8088 );
    // Connect to remote server
    if (connect(socket_servo , (struct sockaddr *)&server , sizeof(server)) < 0){
        puts("connect error");
        return;
    }

    // Send command packet
    if( send(socket_servo , cmdMsg , strlen(cmdMsg), 0) < 0){
        puts("Send failed");
        return;
    }

    close(socket_servo);
}

```



Raspberry Pi 5 Client Servo Driver – Send STOP Value

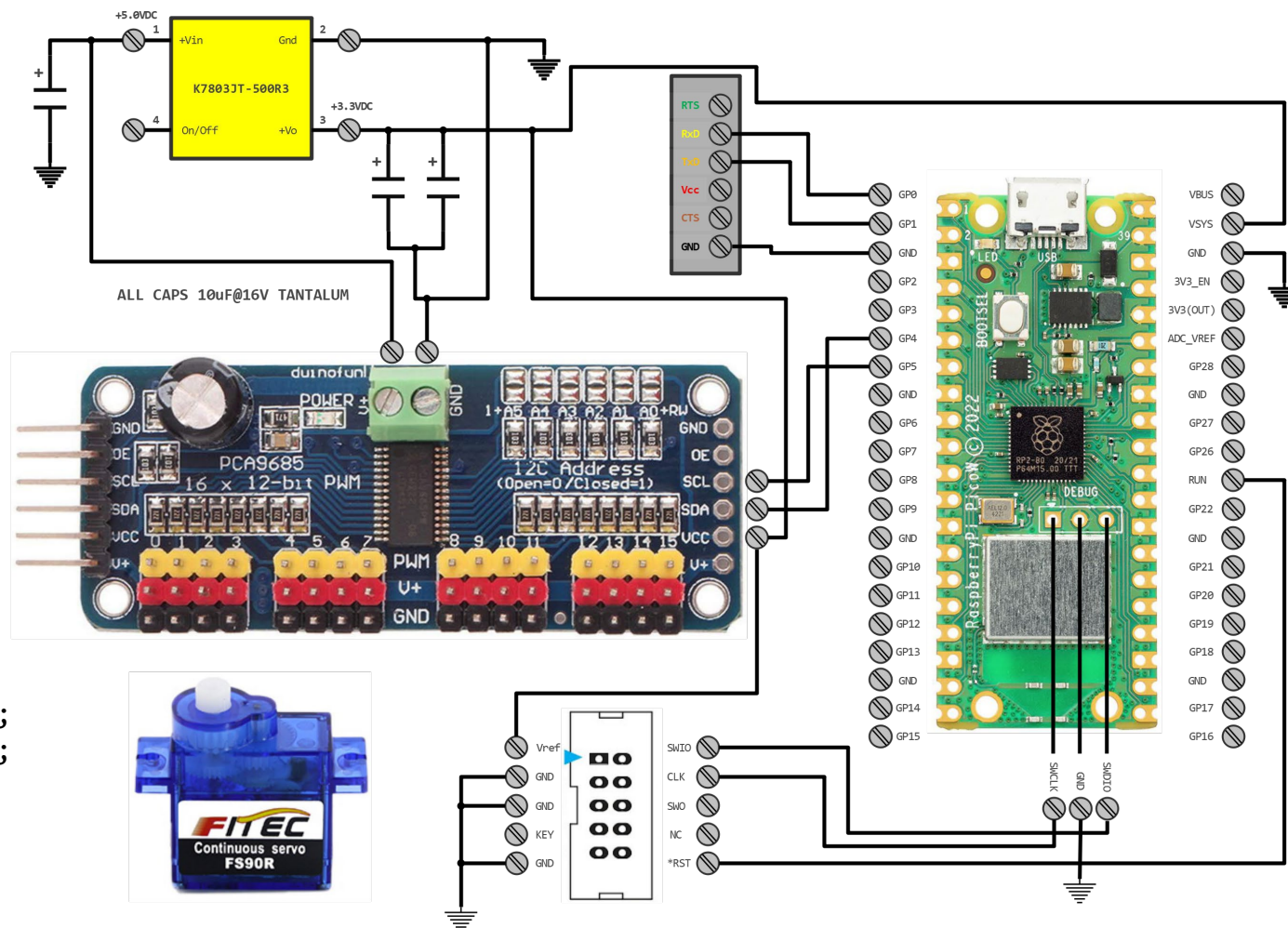
```
int main(int argc, char *argv[])
{
    cmdVal[0] = '0'; // servo number HI (0-1)
    cmdVal[1] = '0'; // servo number LO (0-5)
    cmdVal[2] = '3'; // servo cmd MSByte
    cmdVal[3] = '0';
    cmdVal[4] = '0'; // servo cmd LSByte
    send_servo_cmd(cmdVal);
    return 0;
}
```

Monitor Mode Serial View Mode Text Port /dev/tty

API: 12.2
Data: RaspberryPi.PicoW
Compiler: 1.29.4
ClmImport: 1.47.1
Customization: v5 22/06/24
Creation: 2022-06-24 06:55:08

Connecting to WiFi...
connect status: joining
connect status: no ip
connect status: link up
Connected.
Starting server at 192.168.1.132 on port 8088
43 4D 44 3A 30 30 33 30 30 00 00 00 00 00 00 00
43 4D 44 3A 00 00 00 00 00 00 00 00 00 00 00 00
Servo Command Received
servoNum = 0
cmd = 300

Raspberry Pi PICO W "Helper" – main function

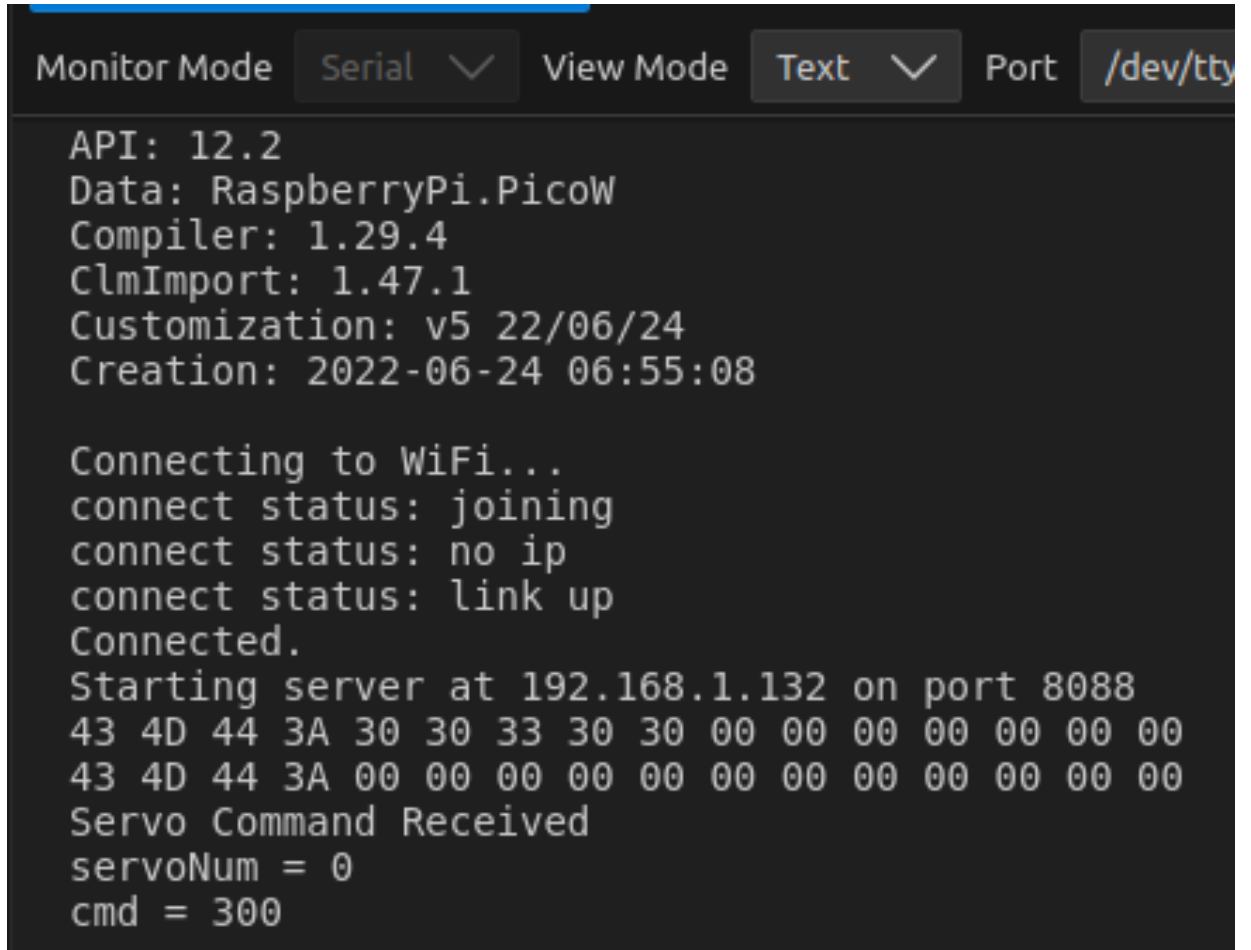


```
int main(void)
{
    stdio_init_all();
    i2c_init(i2c0, 400000);
    gpio_set_function(4, GPIO_FUNC_I2C);
    gpio_set_function(5, GPIO_FUNC_I2C);
    reset();
    setOscFreq(OSC_FREQUENCY);
    setPWMfreq(50);

    TaskHandle_t task;
    xTaskCreate(main_task, "MainThread", configMINIMAL_STACK_SIZE, NULL, MAIN_TASK_PRIORITYhandle_connection_helper, &task);
    s_ConnectionSemaphore = xSemaphoreCreateCounting(kConnectionThreadCount, kConnectionThreadCount);
    vTaskStartScheduler();
}
```


Raspberry Pi PICO W "Helper" – command handler

```
if (!strcmp(bufcmd, "CMD:"))
{
    cyw43_arch_gpio_put(0, true);
    printf("Servo Command Received\r\n");
    servoNum = (buffer[4] - 0x30) * 10;
    servoNum += (buffer[5] - 0x30);
    cmd = (buffer[6] - 0x30) * 100;
    cmd += (buffer[7] - 0x30) * 10;
    cmd += buffer[8] - 0x30;
    printf("servoNum = %u\r\n", servoNum);
    printf("cmd = %u\r\n", cmd);
    // CONTINUOUS SERVO
    setPWM(servoNum, 0, cmd); // Set Pulse Width
    setPWM(1, 0, cmd);       // Saleae Logic 16 Pro
}
else
{
    printf("no match\r\n");
}
break;
```



```
Monitor Mode Serial View Mode Text Port /dev/tty

API: 12.2
Data: RaspberryPi.PicoW
Compiler: 1.29.4
ClmImport: 1.47.1
Customization: v5 22/06/24
Creation: 2022-06-24 06:55:08

Connecting to WiFi...
connect status: joining
connect status: no ip
connect status: link up
Connected.
Starting server at 192.168.1.132 on port 8088
43 4D 44 3A 30 30 33 30 30 00 00 00 00 00 00 00
43 4D 44 3A 00 00 00 00 00 00 00 00 00 00 00 00
Servo Command Received
servoNum = 0
cmd = 300
```


Driving Standard Servos

```
//*****  
/** PCA9685 Servo Driver TCP/IP Client  
/** Written by: Fred Eady  
/** Last Updated: 02-11-2024  
/** Notes:  
//*****  
/** Continuous Servo Command Settings  
/** Spin CLOCKWISE: 10 = FASTEST - 289 SLOWEST  
/** Spin COUNTERCLOCKWISE: 307 = SLOWEST - 500 = FASTEST  
/** STOP BETWEEN 290 AND 303 - 300 IS IDEAL  
/** Standard Servo Command Settings  
/** SERVOMIN = 210 = 1.0mS Pulses  
/** SERVOCTR = 300 = 1.5mS Pulses  
/** SERVOMAX = 400 = 2.0mS Pulses
```

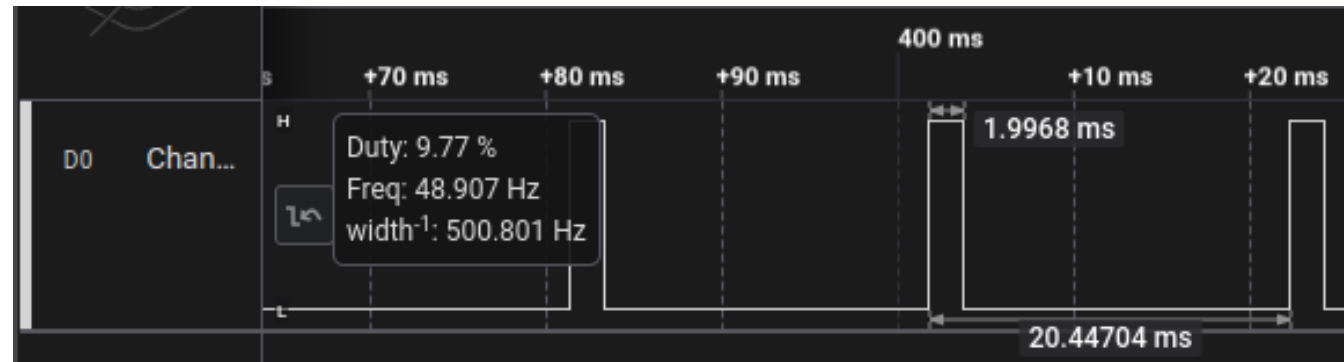
```
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <stdint.h>  
#include <sys/socket.h>  
#include <arpa/inet.h> //inet_addr
```

```
#include <lgpio.h>
```



Driving Standard Servos - SERVOMAX

```
int main(int argc, char *argv[])
{
    cmdVal[0] = '0'; // servo number HI (0-1)
    cmdVal[1] = '0'; // servo number LO (0-5)
    cmdVal[2] = '4'; // servo cmd MSByte
    cmdVal[3] = '0';
    cmdVal[4] = '0'; // servo cmd LSByte
    send_servo_cmd(cmdVal);
    return 0;
}
```



```
Connecting to WiFi...
connect status: joining
connect status: no ip
connect status: link up
Connected.
Starting server at 192.168.1.132 on port 8088
43 4D 44 3A 30 30 34 30 30 00 00 00 00 00 00 00
43 4D 44 3A 00 00 00 00 00 00 00 00 00 00 00 00
Servo Command Received
servoNum = 0
cmd = 400
|
```


Driving Standard Servos - SERVOCR

```
int main(int argc, char *argv[])
{
    cmdVal[0] = '0'; // servo number HI (0-1)
    cmdVal[1] = '0'; // servo number LO (0-5)
    cmdVal[2] = '3'; // servo cmd MSByte
    cmdVal[3] = '0';
    cmdVal[4] = '0'; // servo cmd LSByte
    send_servo_cmd(cmdVal);
    return 0;
}
```



```
Connecting to WiFi...
connect status: joining
connect status: no ip
connect status: link up
Connected.
Starting server at 192.168.1.132 on port 8088
43 4D 44 3A 30 30 33 30 30 00 00 00 00 00 00 00
43 4D 44 3A 00 00 00 00 00 00 00 00 00 00 00 00
Servo Command Received
servoNum = 0
cmd = 300
```

Driving Standard Servos - SERVOMIN

```
int main(int argc, char *argv[])
{
    cmdVal[0] = '0'; // servo number HI (0-1)
    cmdVal[1] = '0'; // servo number LO (0-5)
    cmdVal[2] = '2'; // servo cmd MSByte
    cmdVal[3] = '1';
    cmdVal[4] = '0'; // servo cmd LSByte
    send_servo_cmd(cmdVal);
    return 0;
}
```



```
Connecting to WiFi...
connect status: joining
connect status: no ip
connect status: link up
Connected.
Starting server at 192.168.1.132 on port 8088
43 4D 44 3A 30 30 32 31 30 00 00 00 00 00 00 00
43 4D 44 3A 00 00 00 00 00 00 00 00 00 00 00 00
Servo Command Received
servoNum = 0
cmd = 210
|
```


That's All Folks!!

Thank you for attending!!!

Please consider the resources below:

- **Today's Download Package**
- **raspberrypi.org – Raspberry Pi 5**

LET'S EAT!!





DesignNews

Thank You

Sponsored by

DigiKey

