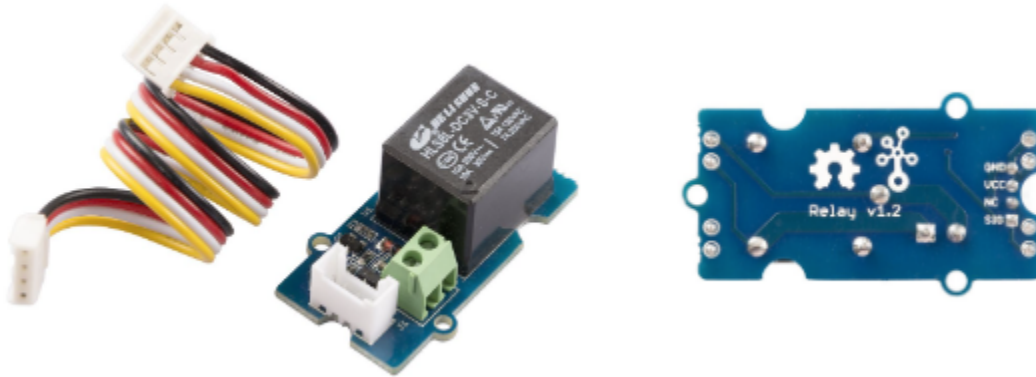


# CMPT 433 How to Guide

## Grove: Relay w/Pump

CMPT 433 Spring 2023

By: Dylan Feng & Sarah Li



Note: Tested on BeagleBone Green & Zen Green (V1.0)

## Table of Contents

|  |    |
|--|----|
| 1. Introduction.....                             | 2  |
| 2. Connecting the Relay to the BeagleBone.....   | 2  |
| 3. Toggling the Relay in the Linux Terminal..... | 3  |
| 4. Toggling the Relay in C.....                  | 4  |
| 5. Wiring the Pump to the Relay.....             | 6  |
| 1. Equipment needed:.....                        | 6  |
| 2. Wiring the pump:.....                         | 6  |
| 3. Controlling the pump:.....                    | 9  |
| 6. Troubleshooting.....                          | 9  |
| • Pump is not working.....                       | 9  |
| • Relay is not working.....                      | 10 |
| References.....                                  | 10 |

## Formatting

1. Host (desktop) commands starting with (host) \$ are Linux console commands  
(host) \$ **echo "Hello world"**
2. Target (board) commands start with (bbg) \$  
(bbg) \$ **echo "On embedded board"**

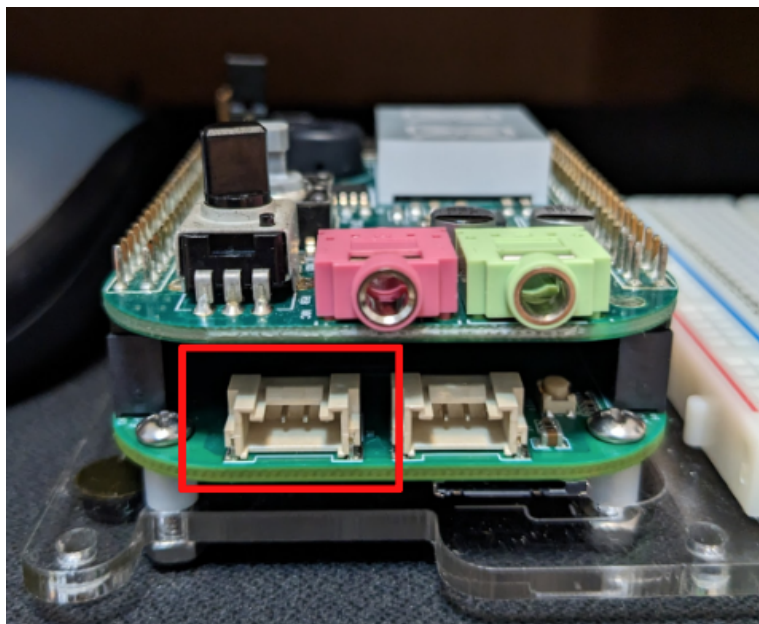
## 1. Introduction

The Grove Relay is a switch that is controllable electrically to open and close a circuit. It uses a Grove connector, which is a 4 pin cable for easier wiring. This guide will demonstrate how to connect the relay to the BeagleBone and use it in a program that uses a 12 volt pump.

| Grove Relay Specifications <a href="#">[1]</a> |               |                       |                    |                       |              |              |                 |                   |
|--|---------------|-----------------------|--------------------|-----------------------|--------------|--------------|-----------------|-------------------|
| Operate Voltage                                | Input Current | Rated Load            | Contact Resistance | Insulation Resistance | Operate Time | Release Time | Input Interface | Type              |
| 3.3V-5V  | 100mA         | 5A@250VAC<br>5A@30VDC | 50mΩ<br>@6VDC 1A   | 100MΩ                 | 10ms Max.    | 5ms Max.     | Digital         | Electromechanical |

## 2. Connecting the Relay to the BeagleBone

Below shows a picture of the two Grove connectors that the BeagleBone has. The leftmost Grove connector is I<sup>2</sup>C (in the red box), and the other Grove connector is UART. We will be using the I<sup>2</sup>C Grove connector, so the white connector on the relay will be plugged into the leftmost Grove pin on the BeagleBone. The white connector should only be able to fit into the Grove connector in one orientation.



To remove the wire, press down firmly on the groove on top of the pin and pull the wire while pressing the groove down. Depending on the wire, some don't need the groove to be pressed down and can be removed simply by pulling.

### 3. Toggling the Relay in the Linux Terminal

When the relay is initially plugged into the BeagleBone, the switch will be closed. When the switch is closed, electricity will flow through the switch and a red LED will light up on the relay. When opening/closing the switch, a short click noise should be audible to indicate that the relay was triggered as well.

| Head_pin | \$PINS | ADDR/OFFSET | Name       | GPIO NO. | Mode7     |
|----------|--------|-------------|------------|----------|-----------|
| P9_01    |        |             | GND        |          |           |
| P9_02    |        |             | GND        |          |           |
| P9_03    |        |             | DC_3.3V    |          |           |
| P9_04    |        |             | DC_3.3V    |          |           |
| P9_05    |        |             | VDD_5V     |          |           |
| P9_06    |        |             | VDD_5V     |          |           |
| P9_07    |        |             | SYS_5V     |          |           |
| P9_08    |        |             | SYS_5V     |          |           |
| P9_09    |        |             | PWR_BUT    |          |           |
| P9_10    |        |             | SYS_RESETn |          |           |
| P9_11    | 28     | 0x870/070   | UART4_RXD  | 30       | gpio0[30] |
| P9_12    | 30     | 0x878/078   | GPIO1_28   | 60       | gpio1[28] |
| P9_13    | 29     | 0x874/074   | UART4_TXD  | 31       | gpio0[31] |
| P9_14    | 18     | 0x848/048   | EHRPWM1A   | 50       | gpio1[18] |
| P9_15    | 16     | 0x840/040   | GPIO1_16   | 48       | gpio1[16] |
| P9_16    | 19     | 0x84c/04c   | EHRPWM1B   | 51       | gpio1[19] |
| P9_17    | 87     | 0x95c/15c   | I2C1_SCL   | 5        | gpio0[5]  |
| P9_18    | 86     | 0x958/158   | I2C1_SDA   | 4        | gpio0[4]  |
| P9_19    | 95     | 0x97c/17c   | I2C2_SCL   | 13       | gpio0[13] |
| P9_20    | 94     | 0x978/178   | I2C2_SDA   | 12       | gpio0[12] |
| P9_21    | 85     | 0x954/154   | UART2_TXD  | 3        | gpio0[3]  |
| P9_22    | 84     | 0x950/150   | UART2_RXD  | 2        | gpio0[2]  |
| P9_23    | 17     | 0x844/044   | GPIO1_17   | 49       | gpio1[17] |

[\[2\]](#)

The I<sup>2</sup>C Grove connector corresponds to I<sup>2</sup>C2 SCL and I<sup>2</sup>C2 SDA on pins P9#19 and P9#20 respectively. To control the relay, we will be using GPIO.

1. Configure the P9#19 & P9#20 pins to GPIO
  - \$ (bbg) **config-pin P9\_19 GPIO**
  - \$ (bbg) **config-pin P9\_20 GPIO**
2. Change into the GPIO directory
  - \$ (bbg) **cd /sys/class/gpio**
3. Export pins if needed (P9#19 maps to GPIO#13 and P9#20 maps to GPIO#12)
  - Check if GPIO#12 and GPIO#13 are available
    - \$ (bbg) **ls**
  - Export them if not listed:
    - \$ (bbg) **echo 13 > export**
    - \$ (bbg) **echo 12 > export**
4. Change into GPIO#13 directory. Only GPIO#13 matters for controlling the relay.
  - \$ (bbg) **cd gpio13**
5. Configure the direction of the GPIO pin to output

- `$ (bbg) echo out > direction`
- 6. To close and open the relay, echo 1 and 0 respectively
  - Close the switch, allowing electricity to flow:
    - `$ (bbg) echo 1 > value`
  - Open the switch, stopping the flow of electricity. The red LED will turn off:
    - `$ (bbg) echo 0 > value`

## 4. Toggling the Relay in C

The steps to toggle the relay in C are the same as the steps as toggling the relay in the terminal.

We will be using these helper functions given by Dr. Brian Fraser to write into a file and use shell commands.

```
static void writeToFile(char* fileLocation, char* message)
{
    FILE* file = fopen(fileLocation, "w");
    if(!file) {
        printf("ERROR OPENING FILE %s\n", fileLocation);
        fclose(file);
        exit(1);
    }

    if(fprintf(file, "%s", message) <= 0) {
        printf("ERROR PRINTING TO FILE %s\n", fileLocation);
        exit(1);
    }

    fclose(file);
}

static void runCommand(char* command)
{
    // Execute the shell command (output into pipe)
    FILE *pipe = popen(command, "r");
    // Ignore output of the command; but consume it
    // So we don't get an error when closing the pipe.
    char buffer[1024];
    while (!feof(pipe) && !ferror(pipe)) {
        if (fgets(buffer, sizeof(buffer), pipe) == NULL)
            break;
        // printf("--> %s", buffer); // Uncomment for debugging
    }
}
```



```

    }

    // Get the exit code from the pipe; non-zero is an error:
    int exitCode = WEXITSTATUS(pclose(pipe));
    if (exitCode != 0) {
        perror("Unable to execute command:");
        printf(" command: %s\n", command);
        printf(" exit code: %d\n", exitCode);
    }
}

```

- Initialize pins P9#19 & P9#20 to GPIO and set the direction to output:

```

// Configure pins to GPIO
runCommand("config-pin P9_19 gpio");
runCommand("config-pin P9_20 gpio");

// Set up GPIO values
writeToFile("/sys/class/gpio/gpio13/direction", "out");
writeToFile("/sys/class/gpio/gpio13/value", "0");

```

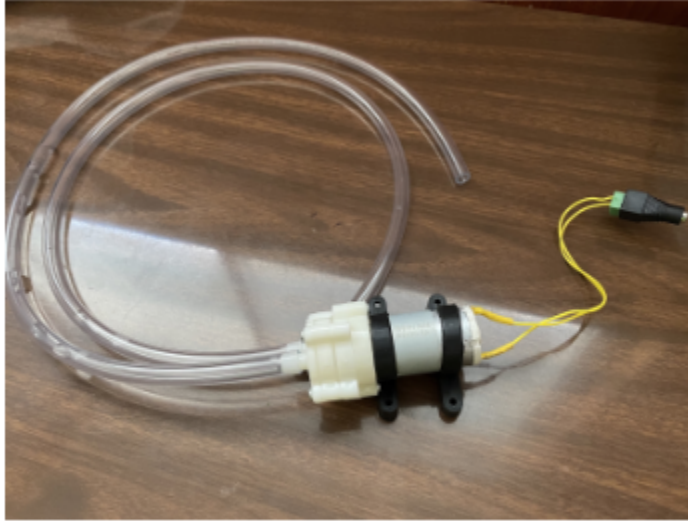
- Close the switch:

```
writeToFile("/sys/class/gpio/gpio13/value", "1");
```

- Open the switch:

```
writeToFile("/sys/class/gpio/gpio13/value", "0");
```

## 5. Wiring the Pump to the Relay



12V Pump



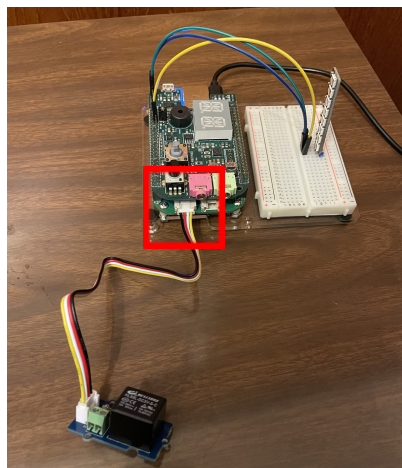
Extra wire

### 1. Equipment needed:

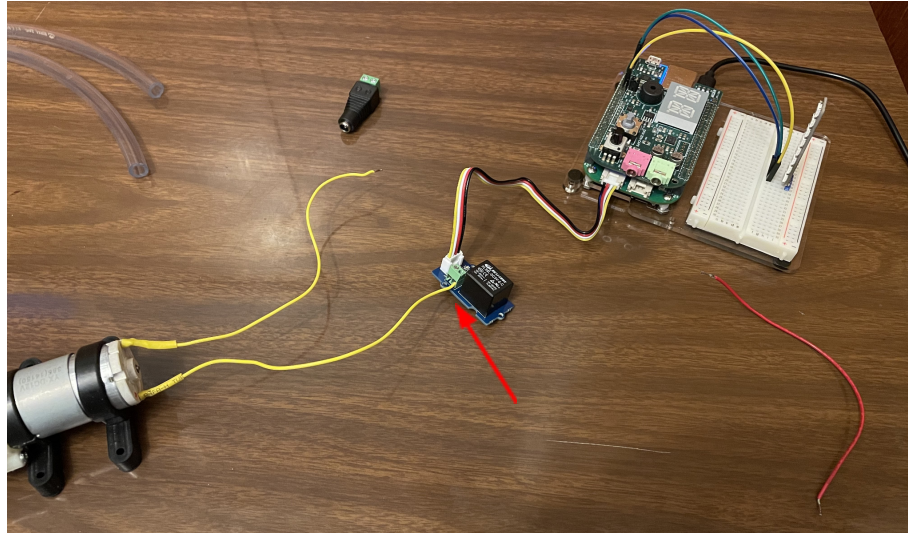
- An extra wire
- Pump
- Relay
- Screwdriver to tighten the clamp on the wires
- External power adapter to power the pump

### 2. Wiring the pump:

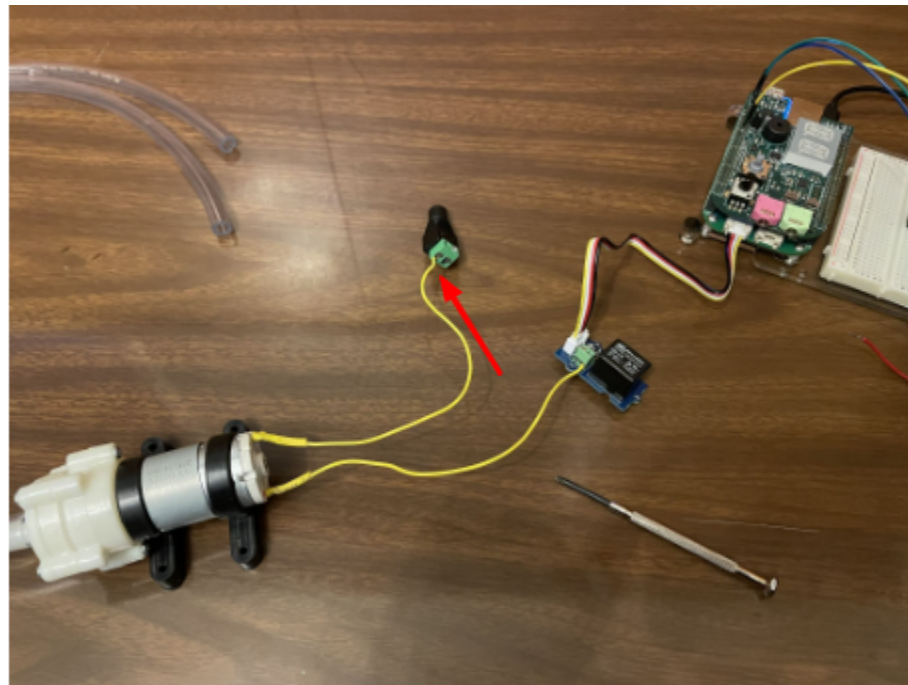
1. Connect the relay to the I<sup>2</sup>C2 Grove connector on the BeagleBone.



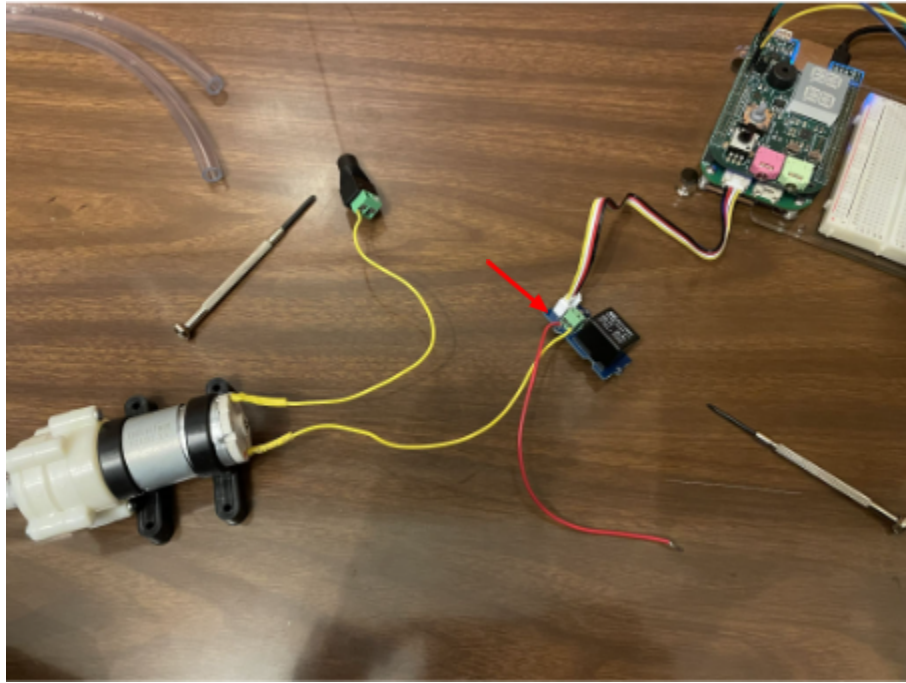
2. Connect one wire from the pump to the relay and tighten the clamp with the screwdriver.



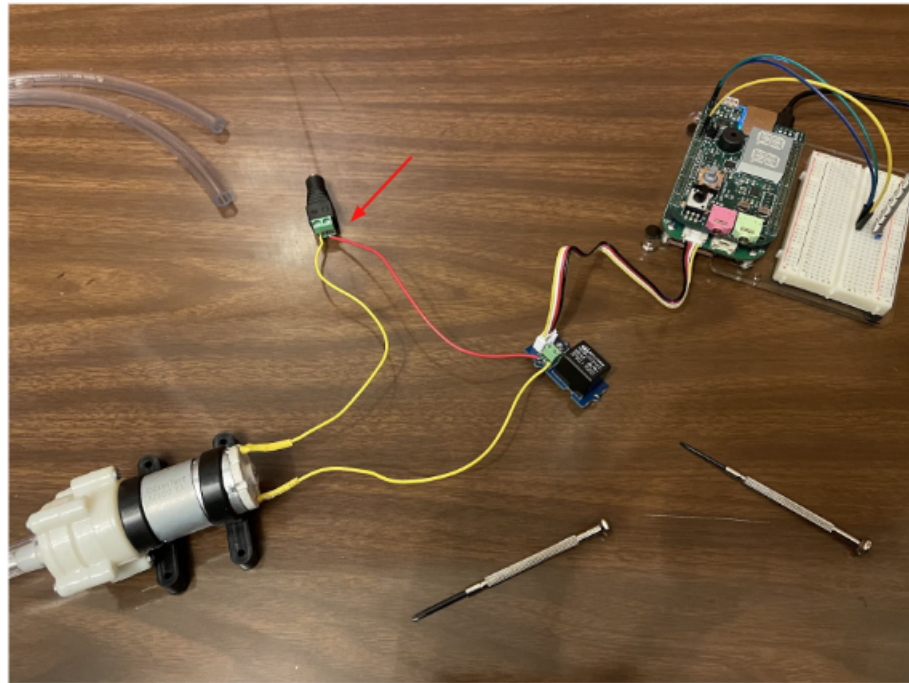
3. Connect the other wire of the pump to a female DC power adapter and tighten the clamp.



4. Use the extra wire and connect one end of it to the relay.

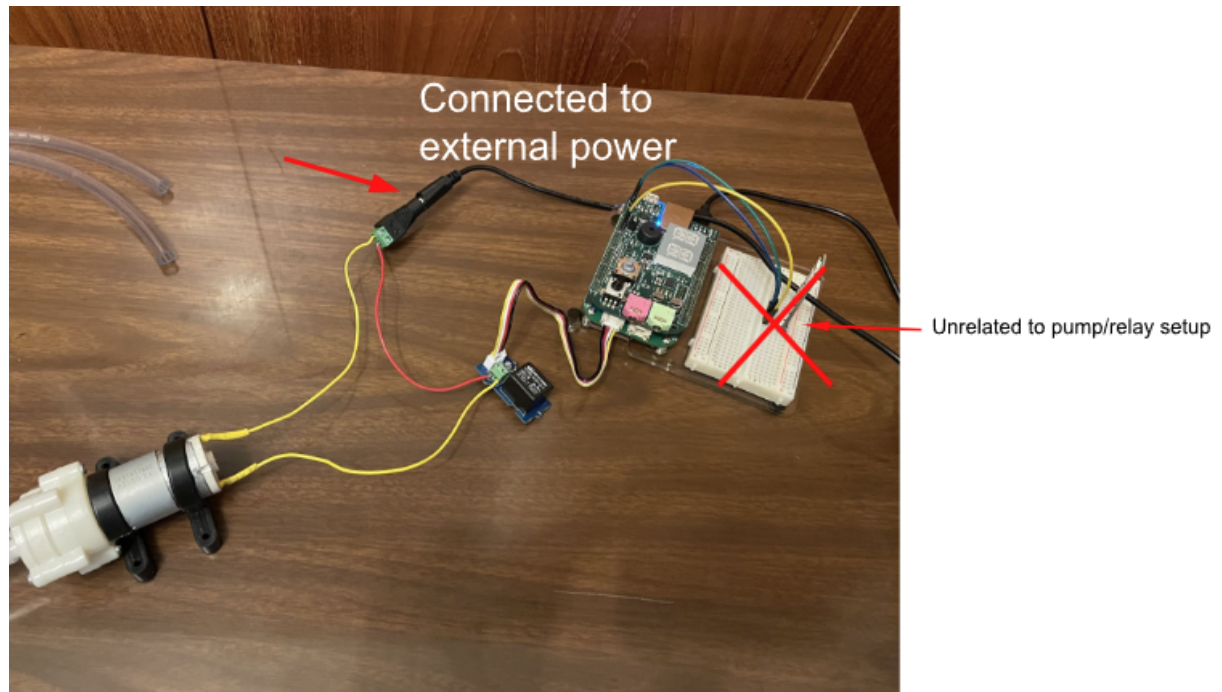


5. Connect the other end of the wire to the female DC power adapter.





6. Connect the female DC power adapter to external power.



**Note:** Make sure all wires are clamped tightly enough to ensure connections are made.

### 3. Controlling the pump:

1. Close the relay to complete the circuit and turn the pump on.
2. Open the relay to break the circuit and turn the pump off.

## 6. Troubleshooting

### ● Pump is not working

- If the relay switches on and off but the pump doesn't work, try connecting the pump to a wall outlet power source directly (without the relay). It should immediately make noise to indicate it is on, otherwise the pump itself may not be working.
- Make sure that all wires are clamped tightly enough to ensure a secure connection (wire connection to the female DC power adapter, wire connection to the relay)

- **Relay is not working**

- If the relay doesn't switch on and off via the command line, check that P9#19 and P9#20 have been configured to GPIO

- `$ (bbg) config-pin -q p9.19`

- `$ (bbg) config-pin -q p9.20`

## References

[1] Seeed Relay Information: [https://wiki.seeedstudio.com/Seeed\\_Relay\\_Page/](https://wiki.seeedstudio.com/Seeed_Relay_Page/)

[2] Beaglebone Black P9 Header Table:

[https://opencoursehub.cs.sfu.ca/bfraser/grav-cms/cmpt433/guides/files/bbg\\_docs/BeagleboneBlackP9HeaderTable.pdf](https://opencoursehub.cs.sfu.ca/bfraser/grav-cms/cmpt433/guides/files/bbg_docs/BeagleboneBlackP9HeaderTable.pdf)