Code example: Controlling the outputs

The Mercury has a high current output on board which you can use to power external devices. Short term it can handle 10-12 amps and continuously 2-3 amps. It does also have a LED attached to the output, so to test the output and flash a LED on the board we can use this example code.

Using Arduino IDE? Our online programmer includes Mercury_Pins.h by default so the pin names work without issue. If you are using Arduino IDE or another programmer, copy the Mercury_Pins.h tab content and paste it into the top of your program.

```
/*
 * Mercury V1 (ESP32-C6) Output example
 * Flashes the output LED and turns the high current output on/off.
 */
#include "Mercury_Pins.h"

void setup() {
   pinMode(OUT1, OUTPUT);     // You need to pullup the BUTTON input, it will be 1 when NOT pressed and 0 when pressed.
   digitalWrite(OUT1, LOW);   // Always turn the output off when starting is good practise
}
void loop() {
   digitalWrite(OUT1, HIGH);
   delay(500);
   digitalWrite(OUT1, LOW);
   delay(500);
}
```

```
#pragma once
/*
 * Mercury (ESP32-C6) Pin Definitions
 * Board-specific GPIO assignments
 */

// ——— Status LED (NeoPixel) ———
#define LEDPOWER      3    // NeoPixel power (drive HIGH to enable)
#define LED           2    // NeoPixel data signal

// ——— I2C Bus ———
#define SDA           21   // I2C data
#define SCL           22   // I2C clock

// ——— Sensor Power ———
#define VACC          20   // Sensor power rail (drive HIGH to enable)

// ——— General Purpose Ports ———
#define GP06          6    // GP06 port
#define GP07          7    // GP07 port

// ——— High Current Output ———
#define OUT1          5    // High current output (e.g. pyro / relay)

// ——— Battery Bar LEDs ———
#define BL1           4    // Battery LED 1 (lowest)
```

```
#define BL2          14   // Battery LED 2
#define BL3          15   // Battery LED 3
#define BL4          18   // Battery LED 4
#define BL5          19   // Battery LED 5 (highest)

// —— Indicators ——
#define DISK         8    // Disk activity LED

// —— Analogue / Detection ——
#define BATIN        0    // Battery voltage (1:1 divider)
#define USBDETECT    1    // USB power detect (HIGH = USB present)
#define BUTTON       9    // BUTTON on the board, boot button but can be used
```