

Project Integration

Research Applied Computer Science

Jochem Arends

495637

Group 1

Academic Year: 2023-2024

Contents

Abbreviations	3
Glossary	3
References	3
1 Introduction	4
2 The ESP32	4
3 Espressif	4
4 Lilygo	4
5 Assignment 1	5
5.1 Circuit	5
5.2 Toolchain	5
5.3 Software	6
6 Assignment 2	7
6.1 Toolchain	7
6.2 Software	8
7 Assignment 4	9
7.1 Toolchain	9
7.2 Software	9

Abbreviations

ACS Applied Computer Science.
AIoT Artificial Intelligence of Things.

Glossary

Bluetooth A short-range wireless interconnection of mobile phones, computers, and other electronic devices.

WiFi A wireless networking technology that uses radio waves to provide high-speed Internet access.

References

Espressif. (n.d.). *About Espressif*. <https://www.espressif.com/en/company/about-espressif>.

1 Introduction

As part of Project Integration, Applied Computer Science (ACS) students need to conduct research about the ESP32 microcontroller. Upon completion of this research, a set of assignments have to be completed. First, some introductory assignments have to be done in order to familiarize ourselves with concepts that are important for the project. At the end of the research, each student works on a final assignment that goes more in depth into a specific topic.

2 The ESP32

The ESP32 is a microcontroller developed by Espressif. The ESP32 has integrated WiFi and Bluetooth modules.

3 Espressif

Espressif Systems is a multinational semiconductor company that focusses among other things on developing wireless communication and Artificial Intelligence of Things (AIoT). (Espressif, n.d.)

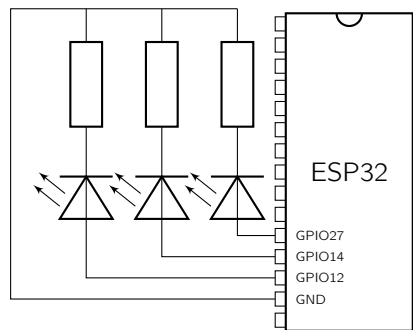
4 Lilygo

5 Assignment 1

Show that the ESP32 can turn three different LEDs on and off separately using an internal loop with delays.

5.1 Circuit

The circuit for this assignment is quite simple. Just connect three LEDs to the ESP32, put these LEDs in series with a resistor, and connect them to a common ground.



5.2 Toolchain

For this assignment, I have used the ESP-IDF toolchain. This assignment was done using the ESP-IDF toolchain.

5.3 Software

```
#include <array>
#include <chrono>
#include <ranges>
#include <thread>

#include "driver/gpio.h"
#include "rom/gpio.h"

extern "C" void app_main() {
    const std::array<gpio_num_t, 3> led_pins{
        GPIO_NUM_12,
        GPIO_NUM_14,
        GPIO_NUM_27,
    };

    // configure the pins for output
    for (auto pin : led_pins) {
        gpio_pad_select_gpio(pin);
        gpio_reset_pin(pin);
        gpio_set_direction(pin, GPIO_MODE_OUTPUT);
    }

    // blink individual LEDs indefinitely
    for (auto pin : std::views::join(std::views::repeat(led_pins))) {
        gpio_set_level(pin, 1);
        std::this_thread::sleep_for(std::chrono::seconds{1});
        gpio_set_level(pin, 0);
    }
}
```

6 Assignment 2

Show that the ESP32 can turn three different LEDs on and off separately by sending commands over the serial interface.

6.1 Toolchain

Just like the previous one, for this assignment I have used the ESP-IDF toolchain. The ESP-IDF toolchain provides good C++ support. From the UART can be read using `std::cin`, but to get the usual behaviour, our program needs to initialize the UART driver first.

6.2 Software

```
#include <iostream>
#include <map>
#include <ranges>

#include "driver/gpio.h"
#include "driver/uart.h"
#include "esp_vfs_dev.h"
#include "rom/gpio.h"

void toggle(gpio_num_t pin) {
    int level = gpio_get_level(pin) ^ 1;
    gpio_set_level(pin, level);
}

extern "C" void app_main() {
    // configure stdin to use blocking mode
    setvbuf(stdin, nullptr, _IONBF, 0);
    constexpr auto uart_num = CONFIG_ESP_CONSOLE_UART_NUM;
    uart_driver_install(static_cast<uart_port_t>(uart_num), 256, 0, 0, nullptr, 0);
    esp_vfs_dev_uart_use_driver(uart_num);

    const std::map<char, gpio_num_t> led_pins{
        {'0', GPIO_NUM_12},
        {'1', GPIO_NUM_14},
        {'2', GPIO_NUM_27},
    };

    // configure the pins for input and output
    for (auto pin : led_pins | std::views::values) {
        gpio_pad_select_gpio(pin);
        gpio_reset_pin(pin);
        gpio_set_direction(pin, GPIO_MODE_INPUT_OUTPUT);
    }

    for (char ch : std::views::istream<char>(std::cin)) {
        if (auto it = led_pins.find(ch); it != led_pins.end()) {
            toggle(it->second);
        }
    }
}
```


7 Assignment 4

Show that the ESP32 can turn three different LEDs on and off separately. Connect the ESP32 to a WiFi access point and host a webserver on the ESP32 to control the LEDs.

7.1 Toolchain

7.2 Software

<https://github.com/jochemarends/project-integration/tree/main/leds-wifi>