API Specification for Common Components

1 Introduction

This document provides the API specifications for the common components used in the project. These components are located in the `Common` directory and include the following modules:

- ADC (Analog-to-Digital Converter)
- I2C (Inter-Integrated Circuit)
- LCD (Liquid Crystal Display)
- UART (Universal Asynchronous Receiver-Transmitter)

2 ADC (Analog-to-Digital Converter)

2.1 Header File: `adc.h`

2.1.1 Functions

- `void adc_init()`
  - **Description**: Initializes the ADC module.
  - **Parameters**: None
  - **Returns**: None
- `uint16_t adc_read(uint8_t channel)`
  - **Description**: Reads the analog value from the specified ADC channel.
  - **Parameters**:
    - `uint8_t channel`: The ADC channel to read from (0-7).
  - **Returns**: The 10-bit analog value read from the specified channel.

3 I2C (Inter-Integrated Circuit)

3.1 Header File: `i2c.h`

3.1.1 Functions

- `void i2c_init()`
  - **Description**: Initializes the I2C interface.
  - **Parameters**: None
  - **Returns**: None
- `void i2c_start()`
  - **Description**: Sends a start condition on the I2C bus.
  - **Parameters**: None
  - **Returns**: None
- `void i2c_stop()`
  - **Description**: Sends a stop condition on the I2C bus.
  - **Parameters**: None
  - **Returns**: None
- `void i2c_write(uint8_t data)`
  - **Description**: Writes a byte of data to the I2C bus.
  - **Parameters**:
    - `uint8_t data`: The data byte to be written.
  - **Returns**: None

4 LCD (Liquid Crystal Display)

4.1 Header File: `lcd.h`

4.1.1 Functions

- `void lcd_send(uint8_t value, uint8_t mode)`
  - **Description**: Sends data or commands to the LCD via I2C in 4-bit mode.
  - **Parameters**:
    - `uint8_t value`: The value to be sent.
    - `uint8_t mode`: The mode (command or data).
  - **Returns**: None

- `void lcd_write_nibble(uint8_t nibble, uint8_t mode)`
  - **Description**: Writes 4 bits to the LCD with the backlight enabled.
  - **Parameters**:
    - `uint8_t nibble`: The 4-bit value to be written.
    - `uint8_t mode`: The mode (command or data).
  - **Returns**: None
- `void lcd_enable_pulse(uint8_t data)`
  - **Description**: Generates an enable pulse to latch data into the LCD.
  - **Parameters**:
    - `uint8_t data`: The data to be latched.
  - **Returns**: None
- `void lcd_init()`
  - **Description**: Initializes the LCD.
  - **Parameters**: None
  - **Returns**: None
- `void lcd_print(const char *str)`
  - **Description**: Prints a string to the LCD.
  - **Parameters**:
    - `const char *str`: The string to be printed.
  - **Returns**: None
- `void lcd_print_row(uint8_t row, const char *str)`
  - **Description**: Prints a string to a specific row on the LCD.
  - **Parameters**:
    - `uint8_t row`: The row number (0 or 1).
    - `const char *str`: The string to be printed.
  - **Returns**: None
- `void lcd_backlight_on()`
  - **Description**: Turns on the LCD backlight.
  - **Parameters**: None
  - **Returns**: None
- `void lcd_clear()`
  - **Description**: Clears the LCD screen.

- **Parameters**: None
- **Returns**: None

- `void lcd_set_cursor(uint8_t row, uint8_t col)`
  - **Description**: Sets the cursor position on the LCD.
  - **Parameters**:
    - `uint8_t row`: The row number (0 or 1).
    - `uint8_t col`: The column number (0-15).
  - **Returns**: None

- `void lcd_show_cursor()`
  - **Description**: Shows the cursor on the LCD.
  - **Parameters**: None
  - **Returns**: None

- `void lcd_hide_cursor()`
  - **Description**: Hides the cursor on the LCD.
  - **Parameters**: None
  - **Returns**: None

- `uint8_t lcd_get_cursor_row()`
  - **Description**: Gets the current cursor row.
  - **Parameters**: None
  - **Returns**: The current cursor row (0 or 1).

- `uint8_t lcd_get_cursor_col()`
  - **Description**: Gets the current cursor column.
  - **Parameters**: None
  - **Returns**: The current cursor column (0-15).

5 UART (Universal Asynchronous Receiver-Transmitter)

5.1 Header File: `uart.h`

5.1.1 Functions

- `void uart_init(unsigned int ubrr)`
  - **Description**: Initializes the UART with the specified baud rate.
  - **Parameters**:

    - `unsigned int ubrr`: The baud rate register value.
  - **Returns**: None

- `void uart_putchar(char c)`
  - **Description**: Sends a character via UART.
  - **Parameters**:
    - `char c`: The character to be sent.
  - **Returns**: None

- `void uart_println(const char *str, ...)`
  - **Description**: Sends a formatted string followed by a newline via UART.
  - **Parameters**:
    - `const char *str`: The format string.
    - `...`: The values to be formatted and sent.
  - **Returns**: None