



SW Architecture Design Description (HLD)

Multi-Color LED Pattern Control SW



Table of Contents

Introduction	3
Architecture Design Description	3
2.1 Architecture Block diagram	3
2.1.1 UI Handler	3
2.1.2 Communication Handler	4
2.1.3 Pattern Executer	4
2.1.4 UART Driver	4
2.1.5 I2C Driver	4
2.2 Use Case Realization	4
2.2.1 Use Case: Configure MLC	5
2.2.2 Use Case: Execute LED Pattern	6
2.3 User Interface	7
2.3.1 Master - Home Screen	7
2.3.2 Slave - Home Screen	8
2.3.3 Slave - (Master Not Found Screen)	9



1 Introduction

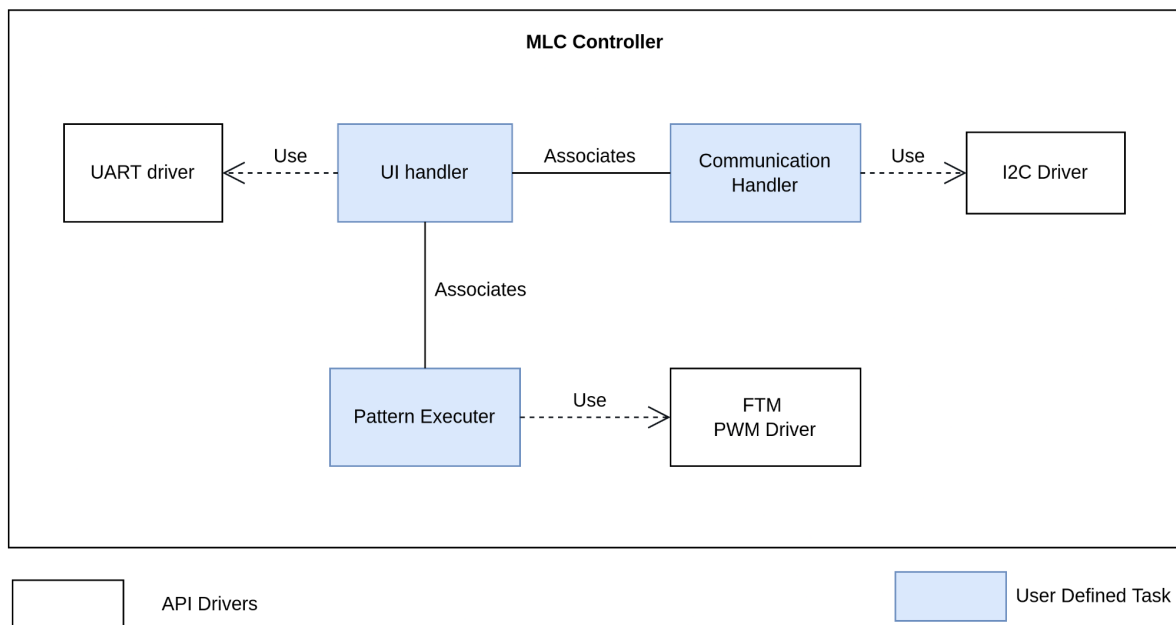
This document describes the SW Design Description (High Level Design) of the case study exercise “Multi Colour LED controller”. (Refer PRD for product requirement description)

2 Architecture Design Description

This subsection describes overall SW architecture through a high level SW block diagram and sequence diagrams, representing interaction between major blocks for the realisation of all use cases.

2.1 Architecture Block diagram

Figure below illustrates the high level block diagram of the SW.



2.1.1 UI Handler

UI Handler is user defined task which have the following responsibilities

- Receive Input from user
- Validate the data
- Display the configuration
- ASCII -> Number Conversion
- Update the configuration



UI Handler uses SDK APIs such as UART_Read(), UART_Write(), to read data from the console and also to display the same.

2.1.2 Communication Handler

Communication Handler is a user defined task which uses I²C protocol to communicate with MLC Master & MLC Slave. The responsibilities of the communication handler are as follows

- Check the presence of slave **(For Master)**
- Check the presence of master **(For Slave)**
- Transfer configuration to slave

2.1.3 Pattern Executer

Pattern Executer is a user defined task in which the entered RGB LED pattern will be executed in the LED. Responsibilities of pattern executer as follows.

- Get LED configuration from UI Handler
- Configure each PWM channel (red, blue, green) according to the RGB Value
- Execute LED pattern

2.1.4 UART Driver

Since we are developing this prototype in FRDM K64, we must configure UART0 of K64F for displaying UI, which is the UI Handler.

2.1.5 I²C Driver

We are using SDK I²C driver for the communication between MLC master and MLC slave.

2.2 Use Case Realization

Refer PRD for Use Case. As per the design document we have two use cases.

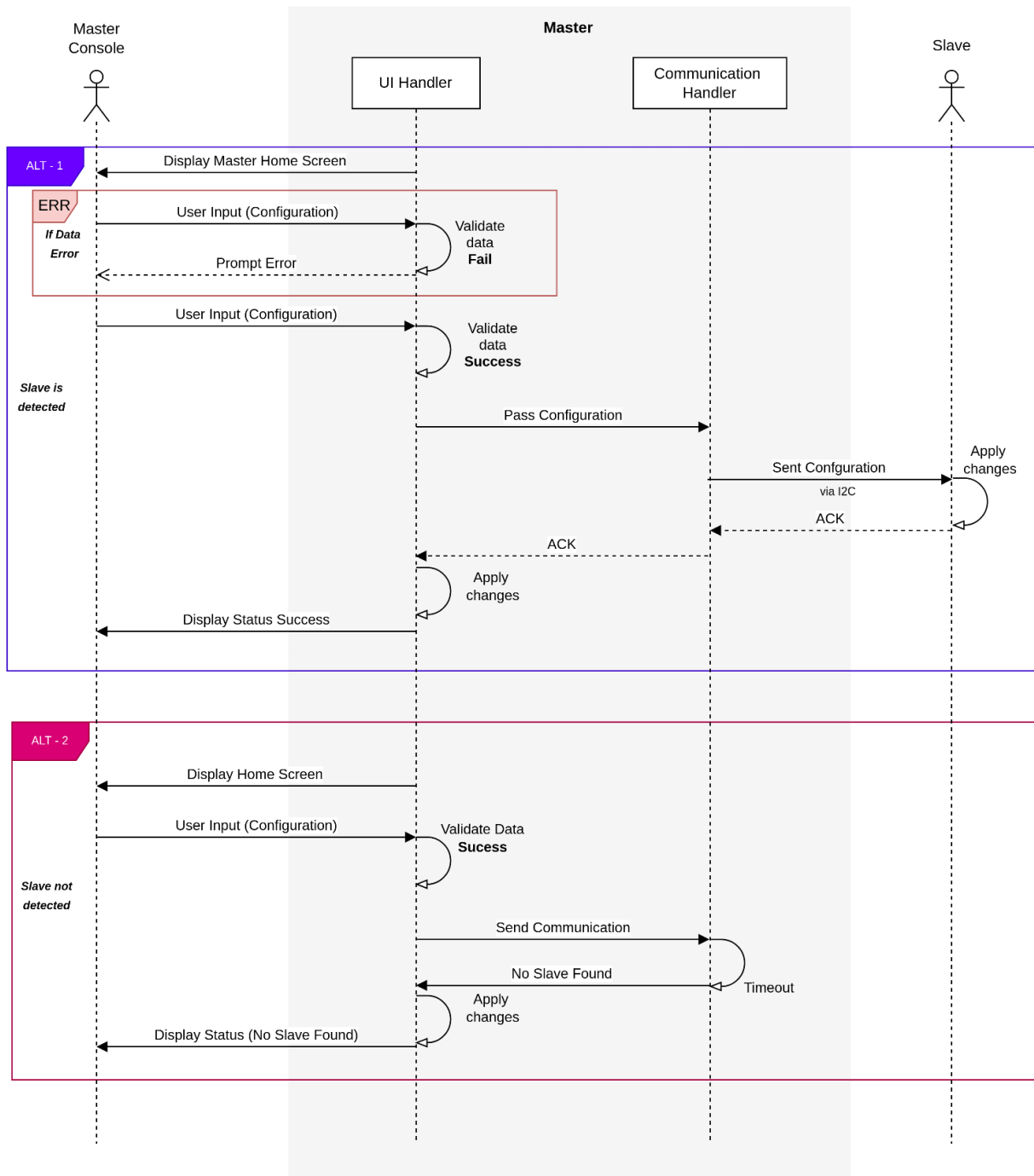


2.2.1 Use Case: Configure MLC

Use Case : Configuring MLC (Detailed)

Scenerios

1. Master with slave connected
2. Master with slave not found



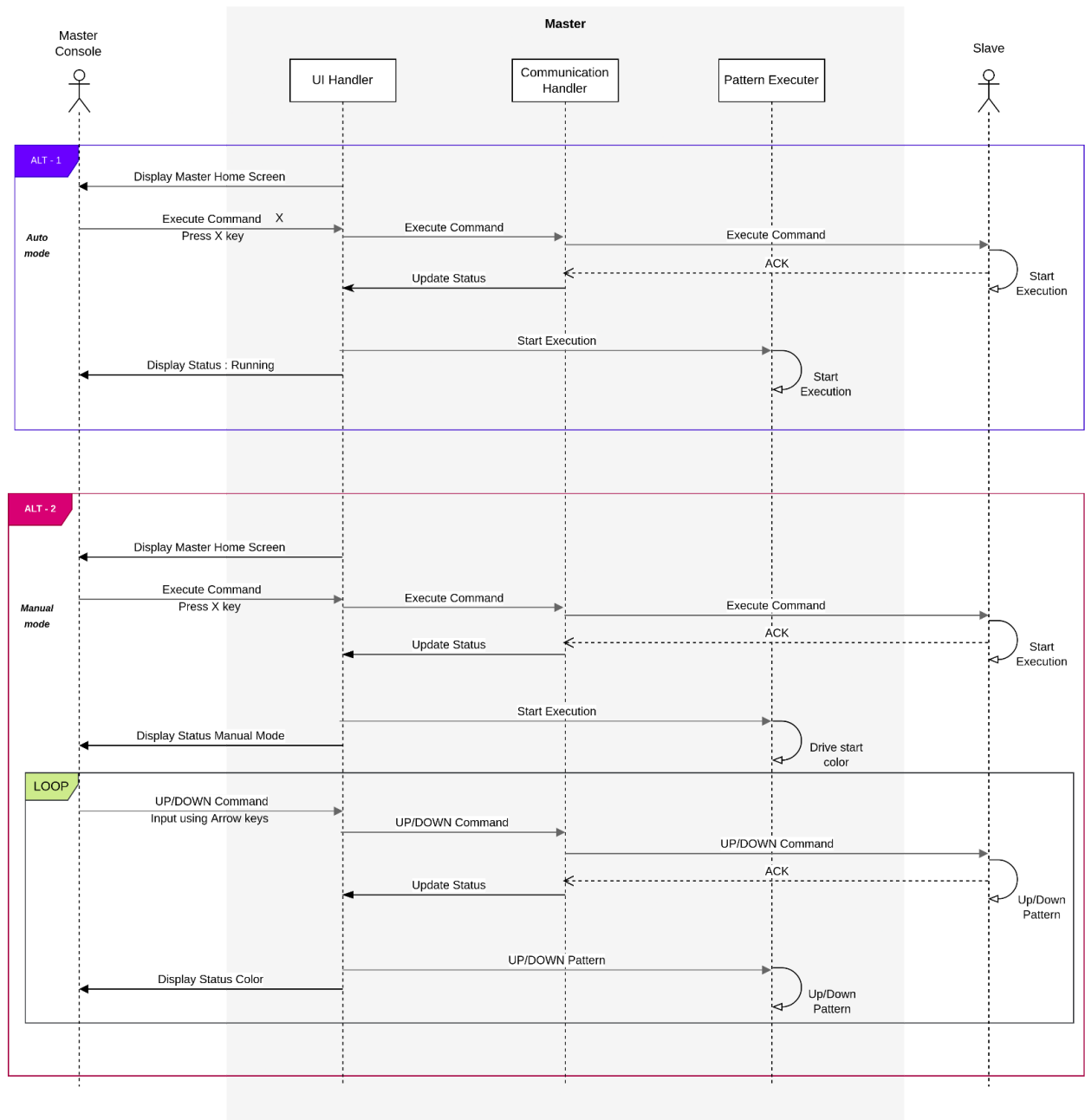


2.2.2 Use Case: Execute LED Pattern

Use Case : Execute LED
Pattern (Detailed)
Master's Perspective

Scenerios

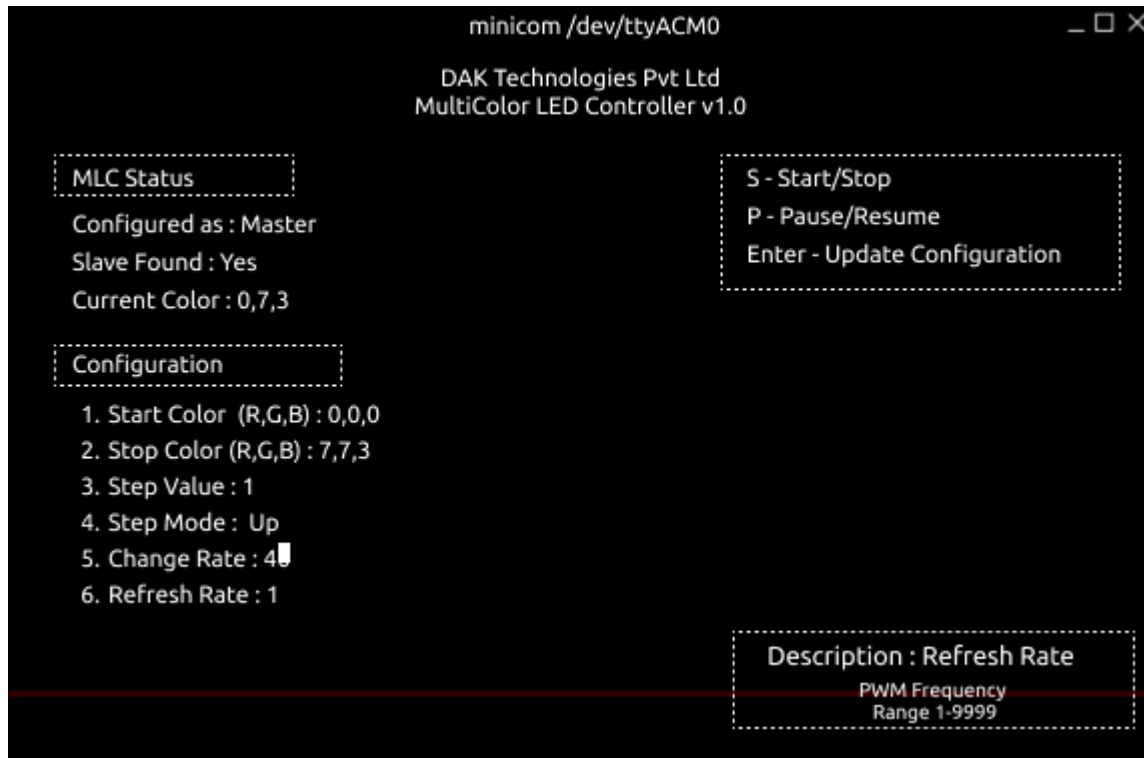
1. Auto Mode
2. Manual Mode





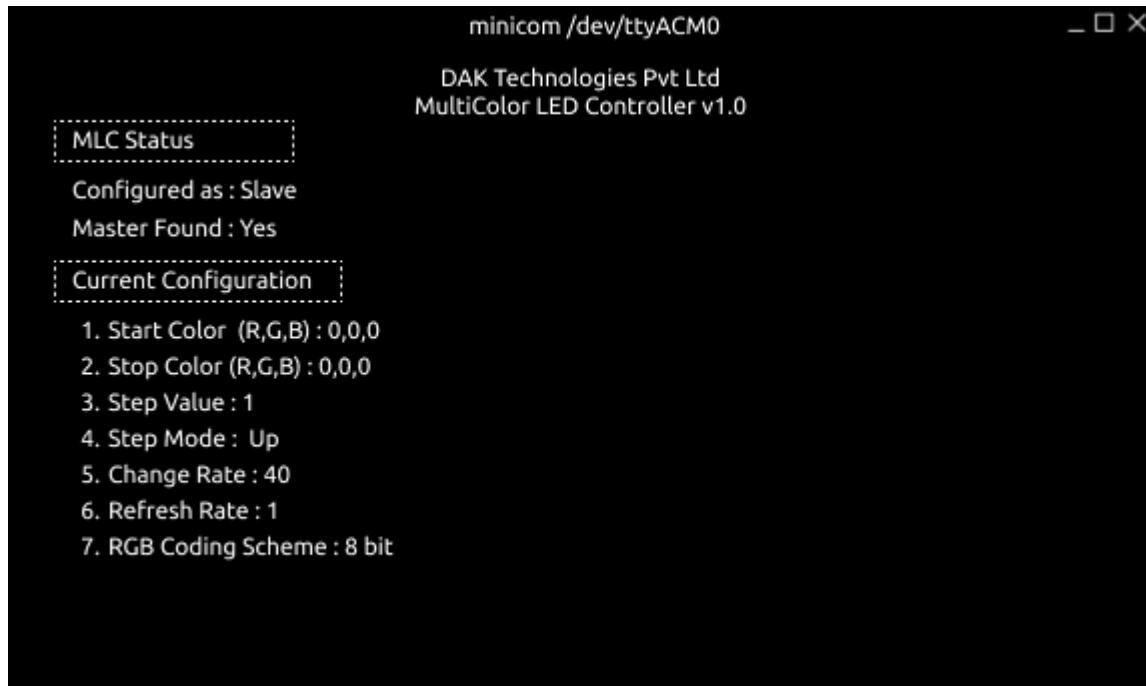
2.3 User Interface

2.3.1 Master - Home Screen





2.3.2 Slave - Home Screen



2.3.3 Slave - (Master Not Found Screen)

