**(GDD)**

LED String Animation

Contents

[**1.** **Document History** 3](#_Toc33138863)

[**1.1.** **Revision History Table** 3](#_Toc33138864)

[**1.2.** **Current** **Document** **Status** 3](#_Toc33138865)

[**2.** **Project Description** 3](#_Toc33138866)

[**3.** **Block Diagram** 4](#_Toc33138867)

[**4.** **Feature Description** 5](#_Toc33138868)

[**4.1.** **Start Up Requirements** 5](#_Toc33138869)

[**4.2.** **Tail Function Requirements** 8](#_Toc33138870)

[**4.3. Turn Indicator (TI) Requirements** 10](#_Toc33138871)

[**5.** **Reference Documents Table** 13](#_Toc33138872)

# **Document History**

## **Current Document Status**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Status | Author | **Date** |
| 1.0 | Draft | Caroline - Mark | 28/02/2020 |
| 1.1 | Draft | Mahmoud Hamdy | 29/02/2020 |
| 1.2 | Draft | Caroline | 29/02/2020 |
| 1.3 | Draft | Mirna | 29/02/2020 |
| 1.4 | Draft | Hesham | 29/02/2020 |
| 1.5 | Draft | Mark | 29/02/2020 |

## **Revision History Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Date | Change |
| 1.0 | Caroline - Mark | 28/02/2020 | Initial Creation |
| 1.1 | Mahmoud Hamdy | 29/02/2020 | Added timer component to architecture and to APIs |
| 1.2 | Caroline | 29/02/2020 | Added DIO APIs |
| 1.3 | Mirna | 29/02/2020 | Added LED\_Animation\_voidSetLedON & LED\_Animation\_voidSetLedOFF |
| 1.4 | Hesham | 29/02/2020 | Added Description to the APIs of the SWITCH Driver Component in HAL layer. |
| 1.5 | Mark | 29/02/2020 | Units, Block Diagrams, Input Arguments and Return Types Modified |

# **Reference Documents Table**

|  |  |  |  |
| --- | --- | --- | --- |
| Ref. number | Doc. Name | Version | Status |
| 1 | LED\_STRING\_ANIMATION\_CYRS | 1.6 | Proposed |
| 2 | LED\_String\_HSI | 1.6 | Proposed |
| 3 | LED\_String\_SRS | 1.1 | proposed |

# **Project Description**

The project is composed of 3 sets of LED Strings Simulating the animation of LEDs in a Car, One set is named “Tail” and it simulates the animation of car’s back LEDs while the other 2 sets are named “Left TI” and “Right TI” is simulating the animation of left and right turn indicator in a car.

Each one of the 3 functions operates based on input signals coming from 3 switches named “Tail Switch”, “Left TI” and “Right TI” respectively in addition to “Welcome Mode” which shall operates one of 2 different modes based on the status of the mode switch. System layout is as shown in **Figure 1** below.



Figure 1: layout of the system

# **Software Context Diagram**

LEDs Manager

Switches Status

Handler

LEDs Status

LED Handler

Switch Manager

# **Input Output Signals**

* Mode Signal

Mode

Handler

LED Animation Algorithm

Signal\_Mode

Signal\_LEDsData

LEDs

Switch\_Pressed

Mode

Switch

|  |  |  |
| --- | --- | --- |
| Signal | Range | Unit |
| Signal\_Mode | 0-1 | Mode |
| Signal\_LEDsData | - | - |

* TIR Signal

Signal\_LEDsData

Signal\_TIR\_Mode

LED Animation Algorithm

LEDs

TIR

Handler

Switch\_Pressed

TIR

Switch

|  |  |  |
| --- | --- | --- |
| Signal | Range | Unit |
| Signal\_TIR\_Mode | 0-1 | Mode |
| Signal\_LEDsData | - | - |

* TIL Signal

Signal\_LEDsData

Signal\_TIL\_Mode

LED Animation Algorithm

LEDs

TIL

Handler

Switch\_Pressed

TIL

Switch

|  |  |  |
| --- | --- | --- |
| Signal | Range | Unit |
| Signal\_TIL\_Mode | 0-1 | Mode |
| Signal\_LEDsData | - | - |

* Tail Signal

Signal\_LEDsData

Signal\_Tail\_Mode

LED Animation Algorithm

LEDs

Tail

Handler

Switch\_Pressed

Tail

Switch

|  |  |  |
| --- | --- | --- |
| Signal | Range | Unit |
| Signal\_Tail\_Mode | 0-1 | Mode |
| Signal\_LEDsData | - | - |

# **Software Features**

LEDs Control Features

Input Feature

Output Feature

Switches\_Status\_Signal

Signal\_Mode

Signal\_Tail\_Mode

Signal\_TIR\_Mode

Signal\_TIL\_Mode

Microcontroller Feature

# **Static Architecture**

APPLICATION

LIB

Std\_Types

LED\_Animation

HAL

Switch

LED

Bit\_Man

MCAL

DIO

Timer

# **APIs**

* LED\_Animation
* Types

|  |  |
| --- | --- |
| Name | LEDString\_t |
| Type | Structure |
| Range | - |
| Description | Data structure containing the set of configuration parameters required for setting the leds status |

|  |  |
| --- | --- |
| Name | LED\_Animation\_Tail\_State |
| Type | Enumeration |
| Range | LED\_ANIMATION\_TAIL\_ON |
| LED\_ANIMATION\_TAIL\_OFF |
| Description | The state of the tail LEDs |

|  |  |
| --- | --- |
| Name | LED\_Animation\_Running\_Mode |
| Type | Enumeration |
| Range | LED\_ANIMATION\_MODE\_WELCOME\_1 |
| LED\_ANIMATION\_MODE\_WELCOME \_2 |
| LED\_ANIMATION\_MODE\_TI\_1 |
| LED\_ANIMATION\_MODE\_TI\_2 |
| LED\_ANIMATION\_MODE\_NONE |
| Description | The state of the tail LEDs |

* Function definitions

|  |  |
| --- | --- |
| Name | Timer\_voidInit |
| Parameters | Timer Overflow Value (u32) |
| Return Value | Void |
| Description | Initializes the timer by enabling its interrupt and setting the value at which the timer will overflow |

|  |  |
| --- | --- |
| Name | Timer\_voidStart |
| Parameters | Void |
| Return Value | Void |
| Description | Starts the timer at the specified instant |

|  |  |
| --- | --- |
| Name | Timer\_f64GetTimeMillis |
| Parameters | Void |
| Return Value | Time in milliseconds (Variable will be of "double" type) |
| Description | Returns the time elapsed since the start of the timer in milliseconds |

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidInit |
| Parameters | Void |
| Return Value | Void |
| Description | Initializes all the LEDs and Switches required for the application |

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidSetFlags |
| Parameters | Void |
| Return Value | Void |
| Description | Reads all the switches status and sets all the flags that represents the LEDs animation |

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidStartAnimationMode |
| Parameters | Void |
| Return Value | Void |
| Description | Reads all the flags status and starts the suitable animation mode according to switches status |

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_03\_V01]**

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_04\_V01]**

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidSetTailLeds |
| Parameters | Void |
| Return Value | Void |
| Description | Sets the tail leds on and off according to the tail flag state |

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_01\_V01]**

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidRunModeOne |
| Parameters | Void |
| Return Value | Void |
| Description | Runs the animation of mode 1 which is described in the LED\_STRING\_ANIMATION\_CYRS |

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_02\_V01]**

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidRunModeTwo |
| Parameters | Void |
| Return Value | Void |
| Description | Runs the animation of mode 2 which is described in the LED\_STRING\_ANIMATION\_CYRS |

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_05\_V02]**

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_06\_V01]**

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidRunTI\_Right |
| Parameters | Void |
| Return Value | Void |
| Description | Runs the animation of TI right mode which is described in the LED\_STRING\_ANIMATION\_CYRS until the TI right switch is released |

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_07\_V02]**

**[Req\_ PO5\_LSAN\_ LED STRING ANIMATION\_08\_V01]**

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidRunTI\_Left |
| Parameters | Void |
| Return Value | Void |
| Description | Runs the animation of TI left mode which is described in the LED\_STRING\_ANIMATION\_CYRS until the TI left switch is released |

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidSetLedON |
| Parameters | LEDString\_t |
| Return Value | Void |
| Description | Turn a specific LED ON |

|  |  |
| --- | --- |
| Name | LED\_Animation\_voidSetLedOFF |
| Parameters | LEDString\_t |
| Return Value | Void |
| Description | Turn a specific LED OFF |

|  |  |
| --- | --- |
| Name | DIO\_SetPinDir |
| Parameters | Port\_No , Pin\_No , Direction |
| Return Value | ErrorStatus |
| Description | Configures the Pin Direction Input/Output |

|  |  |
| --- | --- |
| Name | DIO\_SetPinVal |
| Parameters | Port\_No , Pin\_No , Value |
| Return Value | ErrorStatus |
| Description | Configures the Pin Value High/ Low |

|  |  |
| --- | --- |
| Name | DIO\_GetPinVal |
| Parameters | \*PTR\_Value |
| Return Value | ErrorStatus |
| Description | Reads the Pin Value High/Low |

* Switch
* User-defined Data Types

|  |  |
| --- | --- |
| Name | SWITCH\_t |
| Type | Structure |
| Attributes | Uint8\_t PinNum |
| Uint8\_t PortName |
| Uint8\_t Mode |
| Description | This User-defined data structure shall hold the configurations of the switch pin. |

|  |  |
| --- | --- |
| Name | ERROR\_t |
| Type | Enumeration |
| Range | OK |
| NOK |
| Description | This User-defined data structure shall hold the Status of the returned error level form each API indicating success or failure of function completion and it should be either OK or Not OK. |

* Functions’ Description

|  |  |
| --- | --- |
| Name | Switch\_errorConfig |
| Input Arguments | Pointer to structure of the type SWITCH\_t and shall take the following format:-  **SWITCH\_t\* SwitchName** |
| Return Type | **ERROR\_t** |
| Description | This API shall take a pointer to the structure of the type SWITCH\_t of a certain switch and initialize DIO pins according to the data included in that structure like the pin number and port name as well as the mode of switch connection either pull-up or pull-down configurations. This API should return OK in case of successful pin configuration and NOK in case of invalid input configurations. |

|  |  |
| --- | --- |
| Name | Switch\_errorRead |
| Input Arguments | This API shall take 2 input parameters of the type Uint8\_t and I of the type Uint8\_t\* and they shall take the following formats:-  **Uint8\_t PinNum**  **Uint8\_t PortName**  **Uint8\_t\* RetStatus** |
| Return Type | **ERROR\_t** |
| Description | This API shall read the status of a certain switch that is connected to the pin number and port name taken as input arguments and assign its status (High/Low) to the pointer taken and return indication of Success/Failure of the reading operation or the validity of inputs. |