Component Design Document

For

Kenovo Electric-Blender

Document ID: HLD\_4

**Table of Contents**

*1. Introduction* 2

1.1 Objective 2

1.2 Context Diagram 2

2. *External Interface* 3

2.1 Files 3

2.1.1 Types 3

2.1.2 Interface 3

2.1.3 Const 4

2.1.4 Symbol Define .4.

3. *Static Design* 4

3.1 Files 4

3.2 Types 4

3.3 Symbol Define > 4

3.4 Const 5

3.5 Interface (Services) 5

*4.* *Dynamic Design* 6

4.1 Mode Management 6

4.2 Sequence Diagram 6

*5.* *Shared Resources* 6

5.1 Analysis 6

5.2 Protection 7

6. *Configuration Parameters* 7

6.1 Pre-compile time 7

6.2 Link time 7

6.3 Post-build 8

*7.* *Configuration Constrains* 8

*8.* *Integration Constrains* 8

*9.* *History* 8

*1. Introduction*

## Objective

The purpose of this module is to utilize and control button hardware

## Context Diagram

**SW\_GetHoldTime**()

**SW\_GetState**()

Button Driver

**SW\_SwTask()**

# *External Interface*

DIO\_ReadPin()

## Files

STD\_TYPES.h

DIO\_int.h

### Types

|  |  |
| --- | --- |
| Data Type | Description |
| u8 | unsigned int 8 bits |
| u16 | unsigned int 16 bits |
| u32 | unsigned int 32 bits |

### Interface

|  |  |
| --- | --- |
| Function | Description |
| u8 DIO\_u8ReadPin(u8 pinName) | reads the DIO pin value |

### Const

N/A

### Symbol

N/A

# *Static Design*

## Files

### Used Files

|  |  |
| --- | --- |
| FILE NAME | Description |
| sw.c | Module source file contains the module logic |
| sw.h | Module header file includes provided interfaces |
| Sw\_cfg.h | Module header file includes precompile configuration |

### Files Inclusion

STD\_TYPES.h

Sw\_cfg.h

sw.h

sw.c

DIO\_int.h

\*\*Note: X 🡪Y = Y includes X

## Types

N/A

## Symbol Define

## 3.3.1

|  |  |
| --- | --- |
| Req ID | BUTTON\_00 |
| Covers | HLD\_004 |
| Name | ACTIVE\_HIGH |
| Type | u8 |
| Range | 0-255 |
| Description | One of the button connection modes, ie the button at press the microcontroller pin reads logic one |

**3.3.2**

|  |  |
| --- | --- |
| Req ID | BUTTON\_01 |
| Covers | HLD\_004 |
| Name | ACTIVE\_LOW |
| Type | u8 |
| Range | 0-255 |
| Description | One of the button connection modes, ie the button at press the microcontroller pin reads logic zero |

## Const

N/A

## Interface

### SW\_u8GetState

|  |  |  |  |
| --- | --- | --- | --- |
| Req ID | BUTTON\_03 | | |
| Covers | HLD\_4 | | |
| Name/protoTypes | u8 **SW\_u8GetState**(void) | | |
| Service ID | 0x00 | | |
| Re-entrant / Non re- entrant | None Re-entrant | | |
| Synchronous/Asynchronous | Synchronous | | |
| Return Value | | 0  1 | Button is released  Button is pressed |
| Input parameter | | -- | |
| Output parameter | | -- | |
| Input /Output Parameter | | -- | |
| Description | | returns the current button state (pressed or released) | |

### SW\_GetHoldTime

|  |  |  |  |
| --- | --- | --- | --- |
| Req ID | BUTTON\_04 | | |
| Covers | HLD\_4 | | |
| Name/protoTypes | u8 **SW\_u8GetHoldTime**(void) | | |
| Service ID | 0x01 | | |
| Re-entrant / Non re- entrant | None Re-entrant | | |
| Synchronous/Asynchronous | Synchronous | | |
| Return Value | | u32 holdTime | Time in ms the switch has been preesed |
| Input parameter | | -- | |
| Output parameter | | -- | |
| Input /Output Parameter | | -- | |
| Description | | Returns how much time have passed since the button has pressed till this function call.  \*if the button released(not pressed) function will return 0 | |

### SW\_voidTask

|  |  |  |
| --- | --- | --- |
| Req ID | BUTTON\_05 | |
| Covers | HLD\_4 | |
| Name/protoTypes | **void** **SW\_voidSwTask**(**void**) | |
| Service ID | 0x02 | |
| Re-entrant / Non re- entrant | None Re-entrant | |
| Synchronous/Asynchronous | Synchronous | |
| Return Value | | -- |
| Input parameter | | -- |
| Output parameter | | -- |
| Input /Output Parameter | | -- |
| Description | | Periodic task called by OS |

# *Dynamic Design*

## Mode Management

N/A

## Sequence Diagram

N/A

# *Shared Resources*

## Analysis

5.1.1 BUTTON\_06

|  |  |
| --- | --- |
| Name | u8swState |
| Type | u8 |
| Scope | Static to this module only |
| Shared among | |  |  | | --- | --- | | **SW\_voidSwTask**, | **Updates it's value** | | **SW\_u8GetState** | **Reads from it** | |
| Description | This variable holds the current button state(pressed/released) |

5.1.2 BUTTON\_07

|  |  |
| --- | --- |
| Name | u16wHoldTime |
| Type | u16 |
| Scope | Static to this module only |
| Shared among | |  |  | | --- | --- | | **SW\_voidSwTask**, | **Updates it's value** | | **SW\_u8GetHoldTime** | **Reads from it** | |
| Description | This variable holds the much time have passed since the button has pressed till this function call  \*if the button released(not pressed) function will return 0 |

## Protection

### as we use a none preemptive operating system so any task or function call run to completion without any interruption from other tasks running on the system

# *Configuration Parameters*

## Pre-compile time

**6.1.1** BUTTON\_08

|  |  |
| --- | --- |
| Name | SW\_MODE |
| Type | u8 |
| Range | ACTIV\_HIGH/ACTIV\_LOW |
| Description | SW\_MODE defines the physical button hardware connection mode(active high/active low) |

**6.1.2** BUTTON\_09

|  |  |
| --- | --- |
| Name | SAMPLE\_TIME\_MS |
| Type | u8 |
| Range | 5-10 |
| Description | The periodic time button sampling task should run to check and update the button state |

## Link time

N/A

## Post-build

N/A

# *Configuration Constrains*

N/A

# *Integration Constrains*

N/A

# *History*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Abd-Elrahman Mousa | 16th April 2018 | Create Button module CDD | 1.0 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |