**PO1\_DGC\_Digital Calculator**

**SRS Document**

**Version 1.5**

**Proposed**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Document Change History** | | |  |
| **Version** | **Author** | **Date** | **Change** | **Status** |
| 1.0 | Mina Helmi | 03/2/2020 | * Initial creation | Draft |
| 1.1 | Hazem Mekawy | 05/2/2020 | * Review | Proposed |
| 1.2 | Mina Helmi | 07/2/2020 | * Removed requirement **Req\_PO1\_DGC\_SRS\_010\_V01**, and appended its useful parts to requirement **Req\_PO1\_DGC\_SRS\_007\_V01** * Added a new requirement **Req\_PO1\_DGC\_SRS\_012\_V01** to indicate that the user input shall be received from the keypad. * Added the desired previous state of the switch in both requirements **Req\_PO1\_DGC\_SRS\_010\_V01** and **Req\_PO1\_DGC\_SRS\_011\_V01** * Changed requirement **Req\_PO1\_DGC\_SRS\_006\_V01** to be more explicit and clear regarding the operator and the operand * Added a new requirement **Req\_PO1\_DGC\_SRS\_013\_V01** to indicate that the software design should follow the described flow chart * Renamed State Machine to Flow Chart | Proposed |
| 1.3 | Mina Helmi | 08/2/2020 | * Corrected flow chart * Corrected required switch state in requirement **Req\_PO1\_DGC\_SRS\_011\_V01** * Corrected input of requirement **Req\_PO1\_DGC\_SRS\_012\_V01** | Proposed |
| 1.4 | Mina Helmi | 14/2/2020 | * Added 2 new requirements **Req\_PO1\_DGC\_SRS\_014\_V01** and **Req\_PO1\_DGC\_SRS\_015\_V01** to indicate the range of the horizontal cursor position and what should happen if they were exceeded * Added 2 new requirements **Req\_PO1\_DGC\_SRS\_016\_V01** and **Req\_PO1\_DGC\_SRS\_017\_V01** to indicate the range of the vertical cursor position and what should happen if they were exceeded * Added project description from the CYRS document (V1.2 Released) | Proposed |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1.5 | Mina Helmi | 18/2/2020 | * Updated the requirements **Req\_PO1\_DGC\_SRS\_004\_V01** and **Req\_PO1\_DGC\_SRS\_005\_V01** to reflect the new requirements of the buzzer tones |  |

|  |  |  |
| --- | --- | --- |
| **Reference Document** | | |
| **Reference document** | **Version** | **Status** |
| CYRS | 1.2 | Released |
| HSI | 1.3 | Released |

**Table of Contents**

[Project Description 7](#_Toc32607013)

[Requirements 8](#_Toc32607014)

[Flow Chart 15](#_Toc32607015)

**Index of Figures:**

[Figure 1: Flow Chart 15](file:///C:\Users\MH\Desktop\SRS.docx#_Toc32607032)

# Project Description

This project is a digital calculator that takes input from user and displays the input and the result on a screen.

The hardware used in the calculator is Keypad which takes input from user, LCD to display the result, buzzer to generate tunes on each key press and micro controller that performs all operations in the system.

# Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_001\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_001\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall accept from the user 2 operands <**Opr1**> and <**Opr2**> and an operator <**Operator**> in the following order:   1. Operand 1 is a number, may contain decimal point character and/or a sign. 2. Operator, could be one of the following ASCII characters: '**+**' '**-**' '**/**' '**\***' 3. Operand 2 is a number, may contain decimal point character and/or a sign.   Operands my be of accuracy up to 2 decimal points.  An OK error state will be reported to <**ErrState**>.  The result of the mathematical operation shall be assigned to <**Result>**. | | |
| **Inputs** | * Opr1 * Opr2 * Operator | **Outputs** | * Result = <Opr1> <Operator> <Opr2> * ErrState = OK |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_002\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_001\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall report a bad error state <**NOK**> if an operand, either <**Opr1**> or <**Opr2**> already contained a decimal point and the user tried to type another decimal point. | | |
| **Inputs** | * Opr1 * Opr2 | **Outputs** | ErrState = NOK |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_003\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_001\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall report a bad error state <**NOK**> if any operand either <**Opr1**> or <**Opr2**> is preceded by more than one sign. | | |
| **Inputs** | * Opr1 * Opr2 | **Outputs** | ErrState = NOK |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_004\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_002\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | The software shall send a signal to the buzzer for **1 second** on each press from the user on the keypad.  The period of the pulse **Tpulse = 2ms**, with a duty cycle = **50%**. | | |
| **Inputs** | Keypad press | **Outputs** | Buzzer tone 1 |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_005\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_002\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall send a signal to the buzzer for **1 second** when a bad error state is reported <**NOK**> at any point during the software execution.  The period of the pulse **Tpulse = 4ms**, with a duty cycle = **50%**. | | |
| **Inputs** | ErrState = NOK | **Outputs** | Buzzer tone 2 |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_006\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_003\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall check the operator <**Operator**> if it’s equal to the ASCII symbol '**/**' and the second operand <**Opr2**> if it’s equal to 0, and report a bad error state <**NOK**> if a division by zero occurred. | | |
| **Inputs** | * Opr2 * Operator | **Outputs** | ErrState = NOK |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_007\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_003\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall display an error message if a bad error state <**NOK**> was reported at any point during the software execution for only 2 seconds on the first line of the LCD exclusively, then the retained user input shall be displayed again and the cursor is set back to its last saved location. | | |
| **Inputs** | ErrState = NOK | **Outputs** | Error message: "**ERR: Wrong Input**" |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_008\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_004\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall display the user operands <**Opr1**> and <**Opr2**> and the operator <**Operator**> on the first line of the LCD, then update the cursor position <**Cur\_x**> and <**Cur\_y**>. | | |
| **Inputs** | * Opr1 * Opr2 * Operator | **Outputs** | * Cur\_x * Cur\_y * Output on the first line of the LCD |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_009\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_004\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall display the result of the mathematical operation <**Result**> on the second line of the LCD, and update the cursor positions <**Cur\_x**> and <**Cur\_y**> correspondingly. | | |
| **Inputs** | Result | **Outputs** | * Cur\_x * Cur\_y * Output on the second line of the LCD |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_010\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_005\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall turn on and clear the display, reset the cursor location <**Cur\_x**> and <**Cur\_y**>, and start a new session if the user pressed the ON/OFF switch, regardless of the previous switch state. | | |
| **Inputs** | ON/OFF switch = ON/OFF | **Outputs** | * LCD on * Clear LCD * Cur\_x * Cur\_y |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_011\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_005\_V01 |
| **Author** | Mina Helmi | **Date** | 03 / 02 / 2020 |
| **Description** | Software shall turn off the display if the user pressed the ON/OFF switch, and the previous switch state was <**ON**> | | |
| **Inputs** | ON/OFF switch = ON | **Outputs** | Turn off LCD |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_012\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_001\_V01 |
| **Author** | Mina Helmi | **Date** | 07 / 02 / 2020 |
| **Description** | Software shall accept the user input from the keypad and store it in either one of the operands <**Opr1**> or <**Opr2**> or in the operator <**Operator**> | | |
| **Inputs** | Keypad press | **Outputs** | * Opr1 * Opr2 * Operator |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_013\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_001\_V01 |
| **Author** | Mina Helmi | **Date** | 07 / 02 / 2020 |
| **Description** | Software shall follow the described flow chart | | |
| **Inputs** | Flow Chart | **Outputs** | Overall software design |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_014\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_004\_V01 |
| **Author** | Mina Helmi | **Date** | 14 / 02 / 2020 |
| **Description** | The software shall make sure that the horizontal cursor position of the LCD stored in <**Cur\_x**> never goes outside the range [0, 15] inclusively. | | |
| **Inputs** | Cur\_x | **Outputs** | Any value between [0, 15] inclusively |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_015\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_004\_V01 |
| **Author** | Mina Helmi | **Date** | 14 / 02 / 2020 |
| **Description** | The software shall reject the user input from the keypad if the horizontal cursor position stored in <**Cur\_x**> went outside the range [0, 15] inclusively. | | |
| **Inputs** | Cur\_x | **Outputs** | Reject user input from the keypad |
|  | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_016\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_004\_V01 |
| **Author** | Mina Helmi | **Date** | 14 / 02 / 2020 |
| **Description** | The software shall make sure that the vertical cursor position of the LCD stored in <**Cur\_y**> never goes outside the range [0, 1] inclusively. | | |
| **Inputs** | Cur\_y | **Outputs** | Any value between [0, 1] inclusively |
|  | | | |
| **Req\_ID** | Req\_PO1\_DGC\_SRS\_017\_V01 | **Covers** | Covers\_ PO1\_DGC \_CYRS\_004\_V01 |
| **Author** | Mina Helmi | **Date** | 14 / 02 / 2020 |
| **Description** | The software shall reject the user input from the keypad if the vertical cursor position stored in <**Cur\_y**> went outside the range [0, 1] inclusively. | | |
| **Inputs** | Cur\_y | **Outputs** | Reject user input from the keypad |
|  | | | |

# Flow Chart

**Yes**

Yes

Figure 1: Flow Chart

No

No

Yes

No

Turn off  
display

Is calculator on?

Result = <Opr1> <Operator> <Opr2>

Trigger  
buzzer

Display error message

Display error message

Trigger  
buzzer

Is operand valid?

Get  
operand 2  
<Opr2>

Get  
operator

Is operand valid?

Turn on  
display

Reset  
cursor

Clear  
display

Get  
operand 1  
<Opr1>