



LCD and KEYPAD

Static Design

Bassel Yasser Mahmoud

Contents

LCD	1
APIs	1
void HLCD_vidInit(void);	1
void HLCD_vidWritecmd (Uint8_t u8commandCopy);	1
void HLCD_vidWriteChar (Uint8_t u8CharCopy);	1
void HLCD_ClrDisplay(void);	1
void HLCD_ShiftLeft(void);	2
void HLCD_gotoXY (Uint8_t row, Uint8_t pos);	2
void HLCD_WriteString (Uint8_t* str);	2
void HLCD_WriteInt (Uint32_t number);	2
void HLCD_vidCreatCustomChar (Uint8_t* pu8custom, Uint8_t u8Location);	2
Flowchart	3
void Timer_CallBackFunc (void);	3
void HLCD_vidInit(void);	5
void HLCD_vidWritecmd (Uint8_t u8commandCopy);	6
void HLCD_vidWriteChar (Uint8_t u8CharCopy);	7
void HLCD_ClrDisplay(void);	8
void HLCD_ShiftLeft(void);	9
void HLCD_gotoXY (Uint8_t row, Uint8_t pos);	10
void HLCD_WriteString (Uint8_t* str);	11
void HLCD_vidCreatCustomChar (Uint8_t* pu8custom, Uint8_t u8Location);	12
void HLCD_WriteInt (Uint32_t number);	13
KEYPAD	14
APIs	14
void KEYPAD_vidInit_V2(void);	14
Uint8_t KEYPAD_u8GetPressed_V2(void);	14
Flowchart	15
void KEYPAD_vidInit_V2(void);	15
Uint8_t KEYPAD_u8GetPressed_V2(void);	16

LCD

APIs

```
/*
 * function      : HLCD_vidInit
 * description   : func to set LCD initialization
 * input param  : void
 * return       : void
 * */
```

```
void HLCD_vidInit(void);
```

```
/*
 * function      : HLCD_vidWritecmd
 * description   : func to configure some commands on lcd
 * input param  :
 *
 *                                     u8commandCopy --> take lcd cmd instructions from
instruction table
<https://components101.com/sites/default/files/component\_datasheet/16x2%20LCD%20Datasheet.pdf>
 * return       : void
 * */
```

```
void HLCD_vidWritecmd (Uint8_t u8commandCopy);
```

```
/*
 * function      : HLCD_vidWriteChar
 * description   : func to write char on lcd
 * input param  : u8CharCopy -> take ascii code of char or char address on
CGROM
 * return       : void
 * */
```

```
void HLCD_vidWriteChar (Uint8_t u8CharCopy);
```

```
/*
 * function      : HLCD_ClrDisplay
 * description   : func to clear anything on lcd
 * input param  : void
 * return       : void
 * */
```

```
void HLCD_ClrDisplay(void);
```

LCD

```
/*
 * function      : HLCD_ShiftLeft
 * description    : func to shift the lcd display from right to left
 * input param   : void
 * return        : void
 * */
```

```
void HLCD_ShiftLeft(void);
```

```
/*
 * function      : HLCD_gotoXY
 * description    : func to determine position which char print at this position on
lcd ### NOTE : (2rows x 16coloms)
 * input param   :
 *                row -> take row number 0 or 1
 *                pos -> take colom number from 0 ~ 16
 * return        : void
 * */
```

```
void HLCD_gotoXY (Uint8_t row, Uint8_t pos);
```

```
/*
 * function      : HLCD_WriteString
 * description    : func to write string on lcd
 * input param   : str --> which take string as argument
 * return        : void
 * */
```

```
void HLCD_WriteString (Uint8_t* str);
```

```
/*
 * function      : HLCD_WriteInt
 * description    : func to write integer number on lcd
 * input param   : number --> which take number as argument
 * return        : void
 * */
```

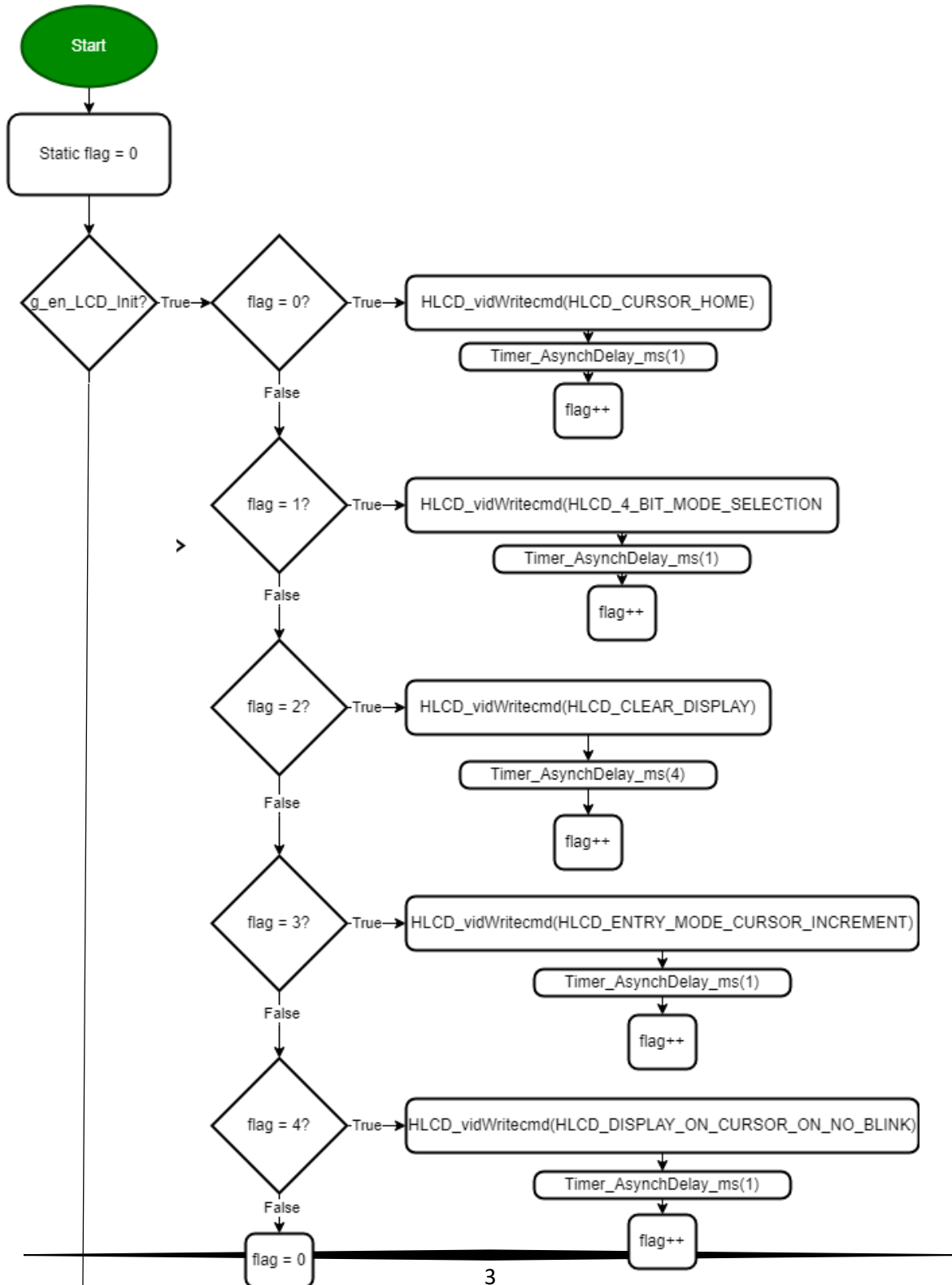
```
void HLCD_WriteInt (Uint32_t number);
```

```
/*
 * function      : HLCD_vidCreatCustomChar
 * description    : func to store new pattern on CGRAM
 * input param   :
 *                pu8custom -> take pointer to array which having LCD
Custom Character Generated data ### take only 8 characters
 *                u8Location -> determine location on CGRAM [0 ~ 8]
 * return        : void
 * */
```

```
void HLCD_vidCreatCustomChar (Uint8_t* pu8custom, Uint8_t u8Location);
```

Flowchart

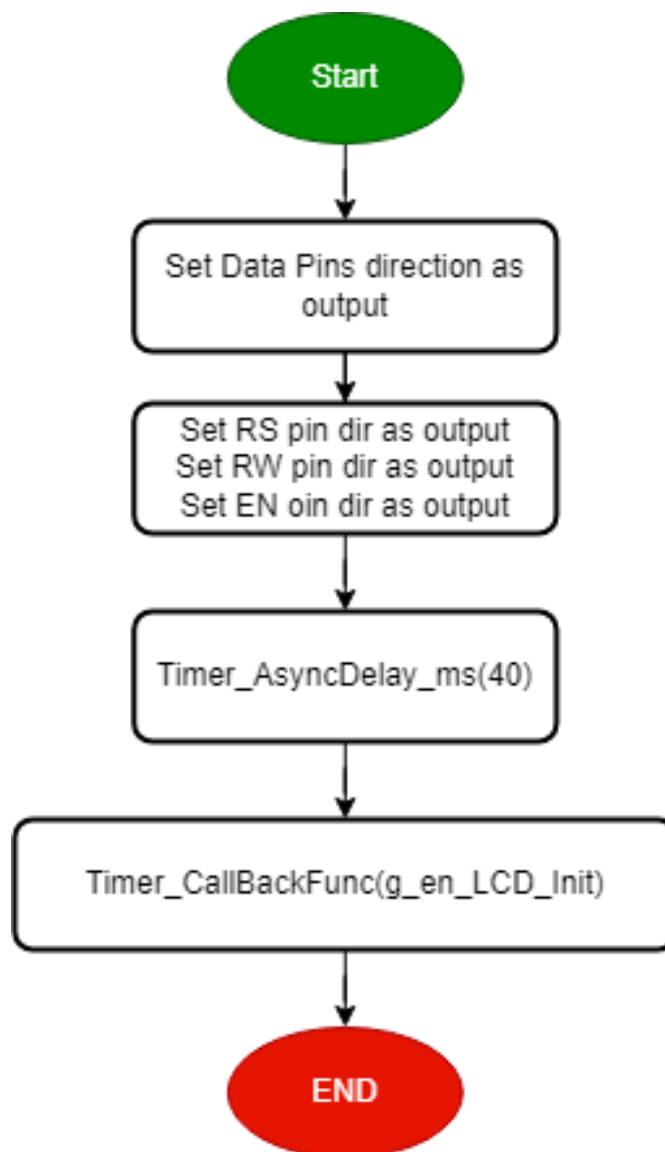
```
void Timer_CallBackFunc (void);
```



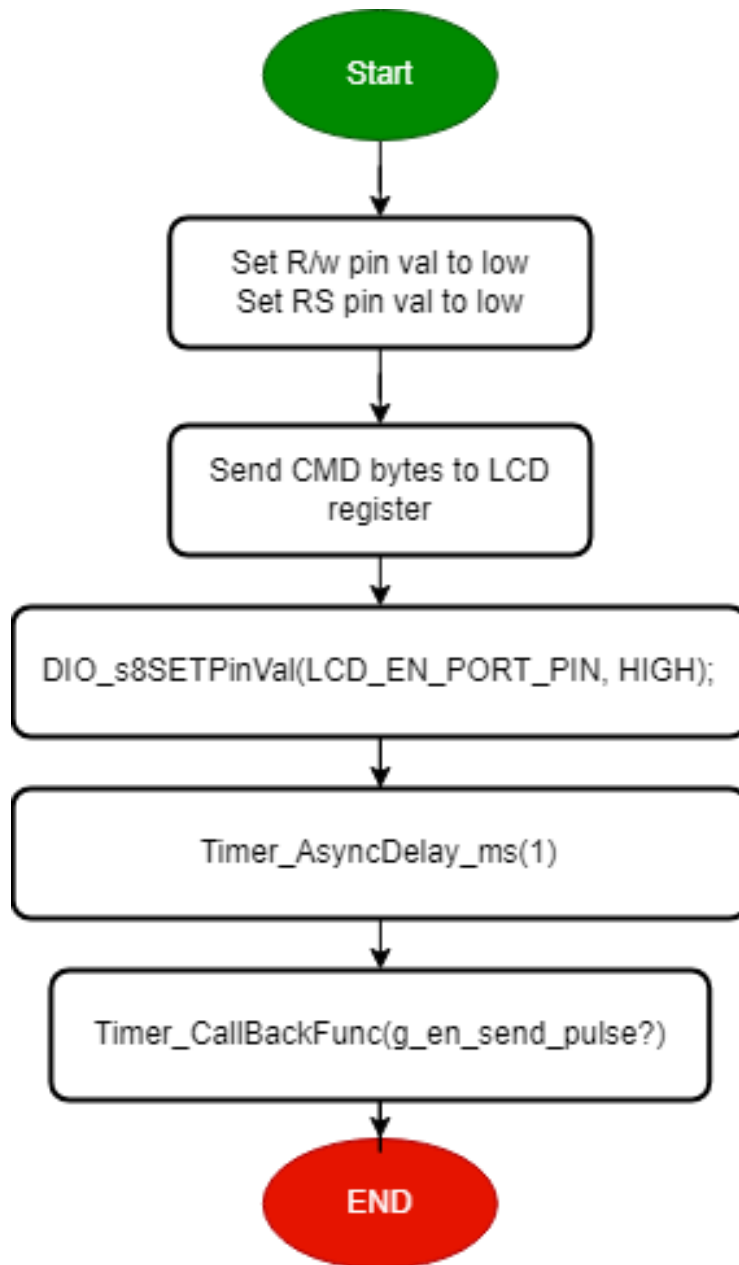


KEYPAD

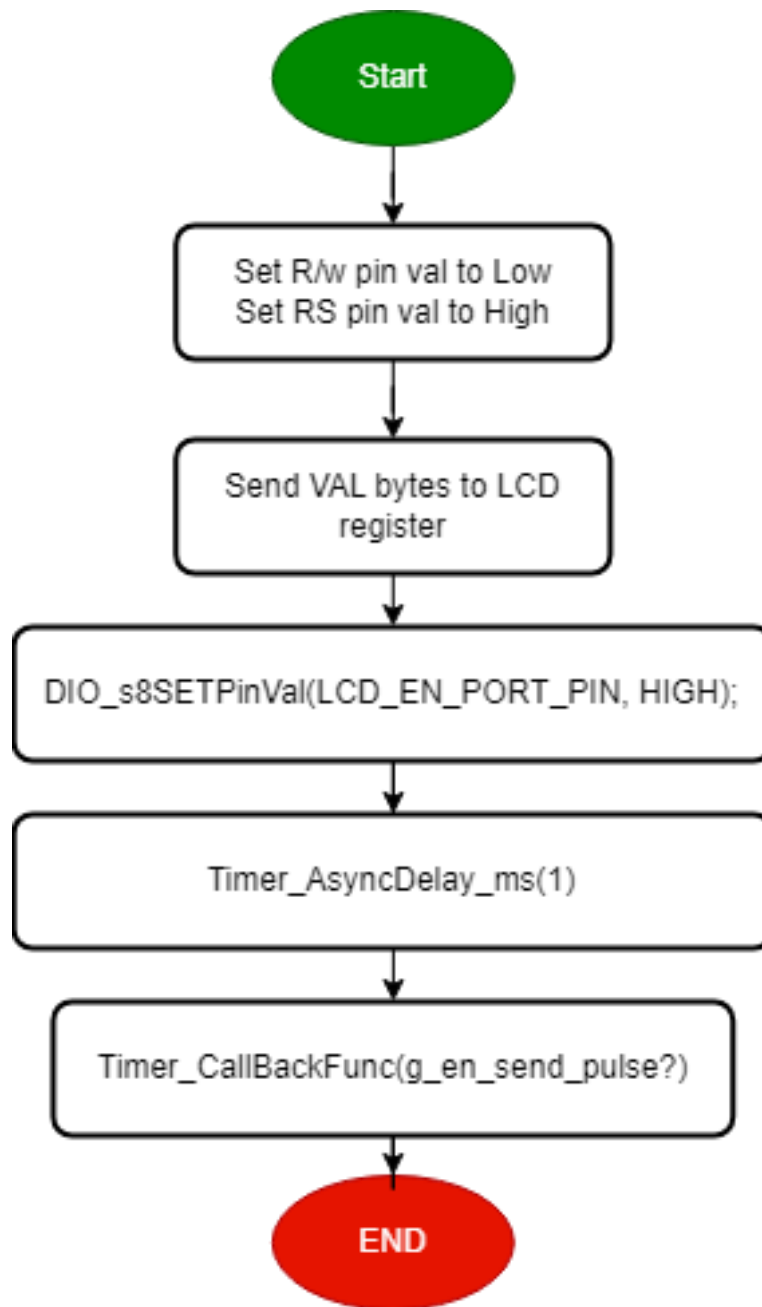
```
void HLCD_vidInit(void);
```



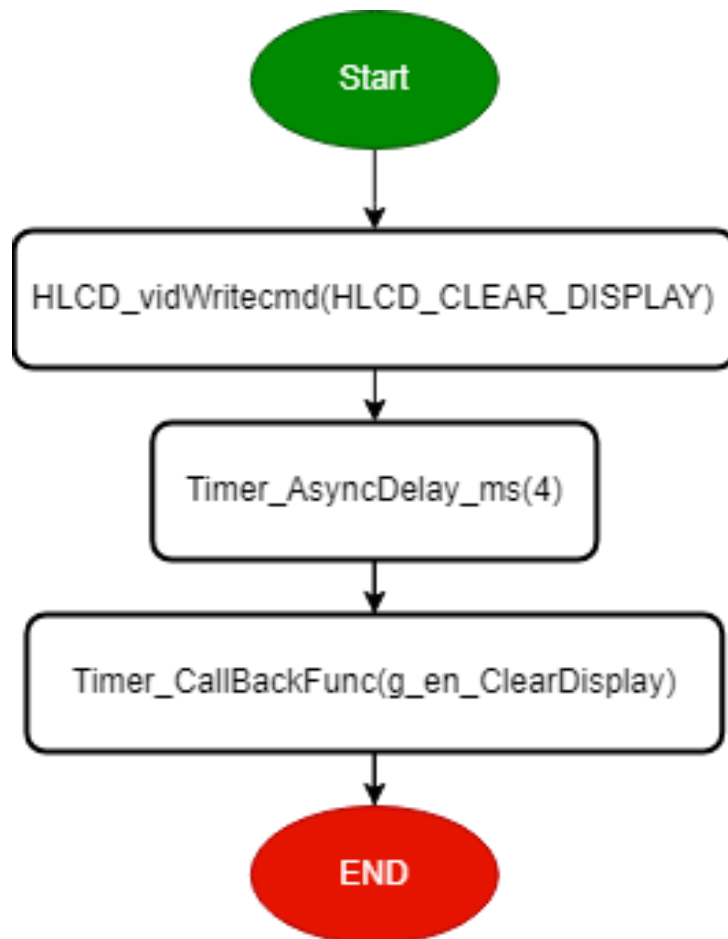
```
void HLCD_vidWritecmd (Uint8_t u8commandCopy);
```




```
void HLCD_vidWriteChar (Uint8_t u8CharCopy);
```

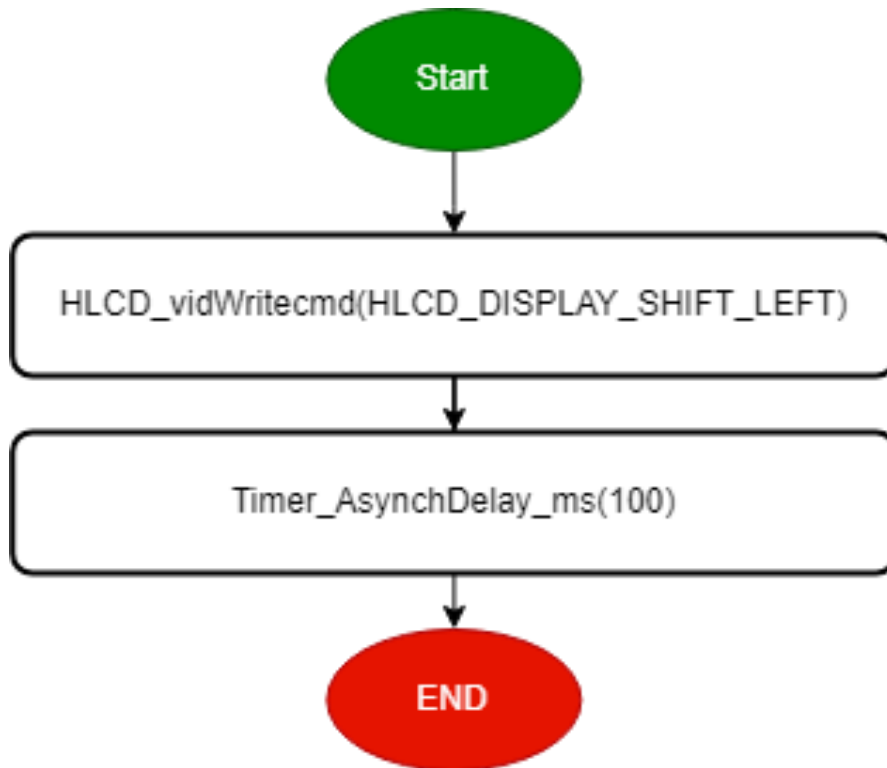


```
void HLCD_ClrDisplay(void);
```



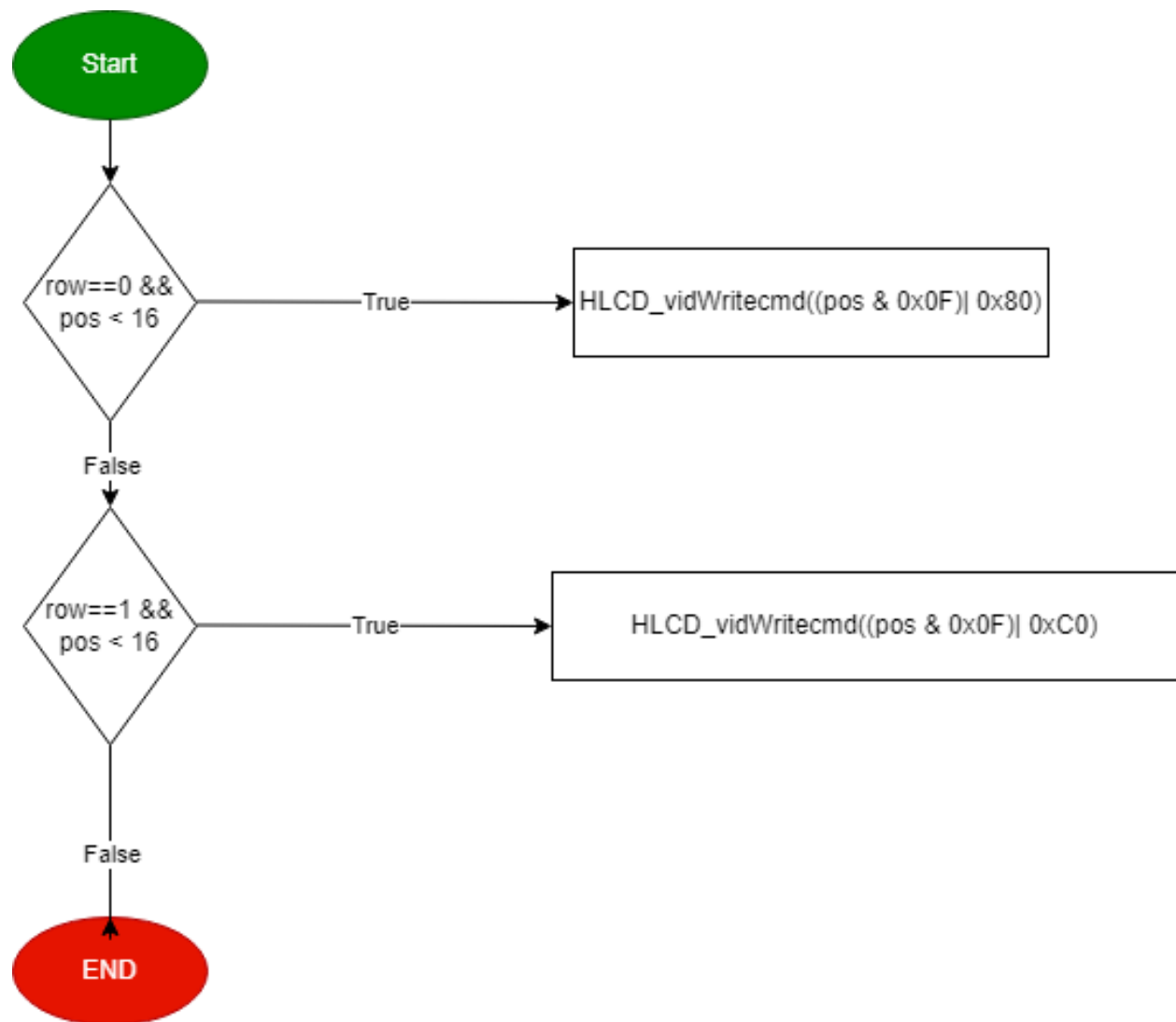
KEYPAD

```
void HLCD_ShiftLeft(void);
```

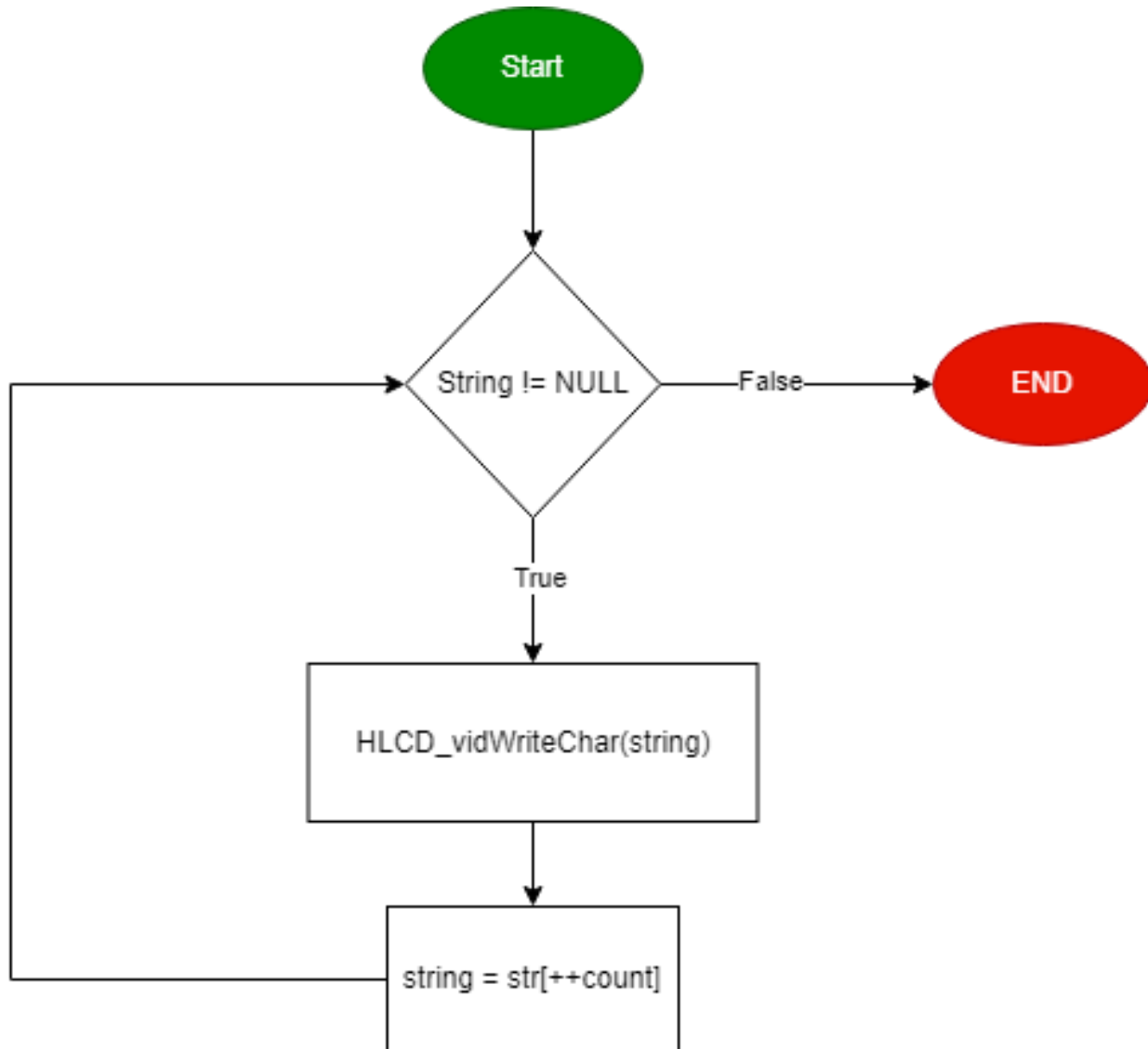


KEYPAD

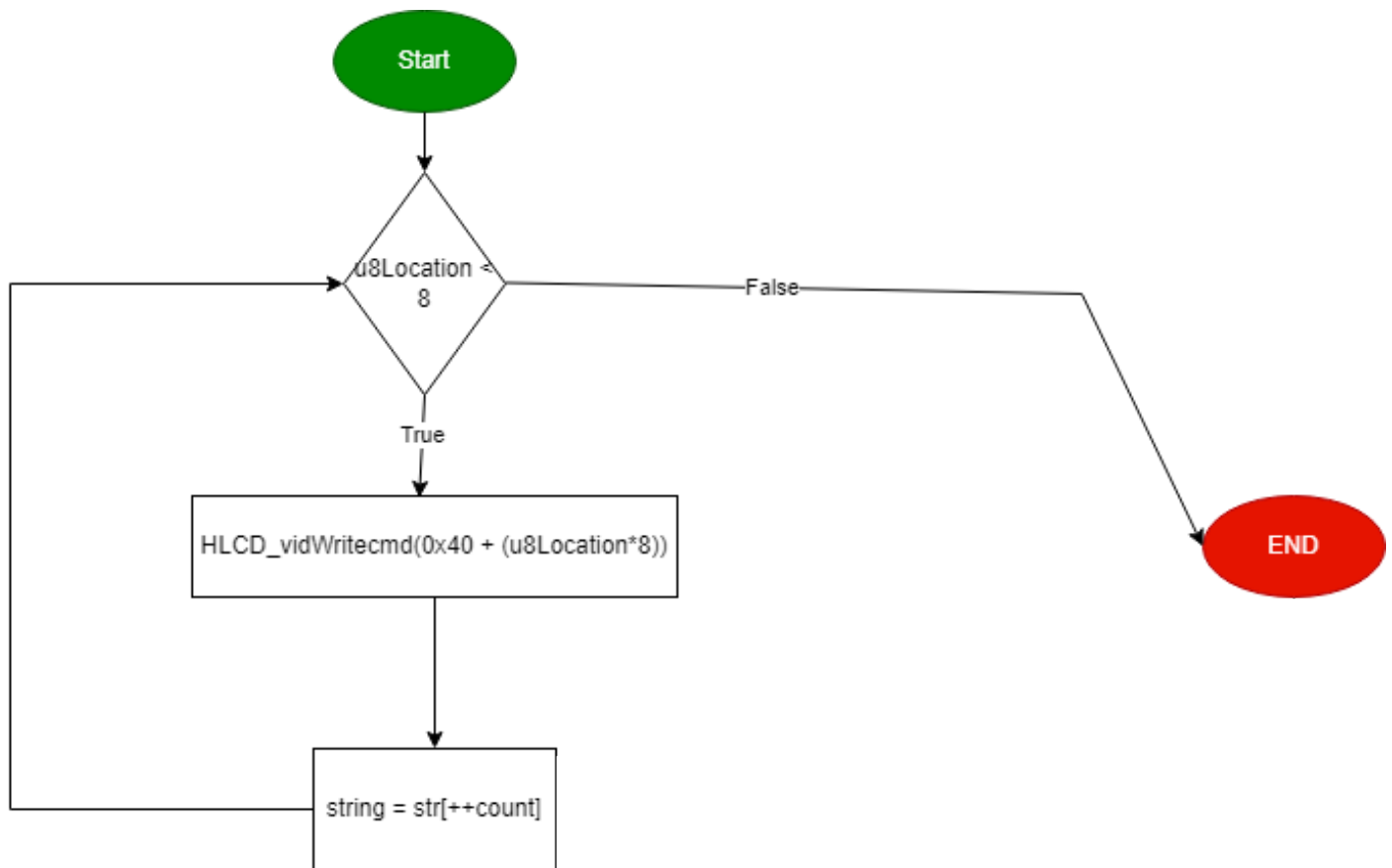
```
void HLCD_gotoXY (Uint8_t row, Uint8_t pos);
```



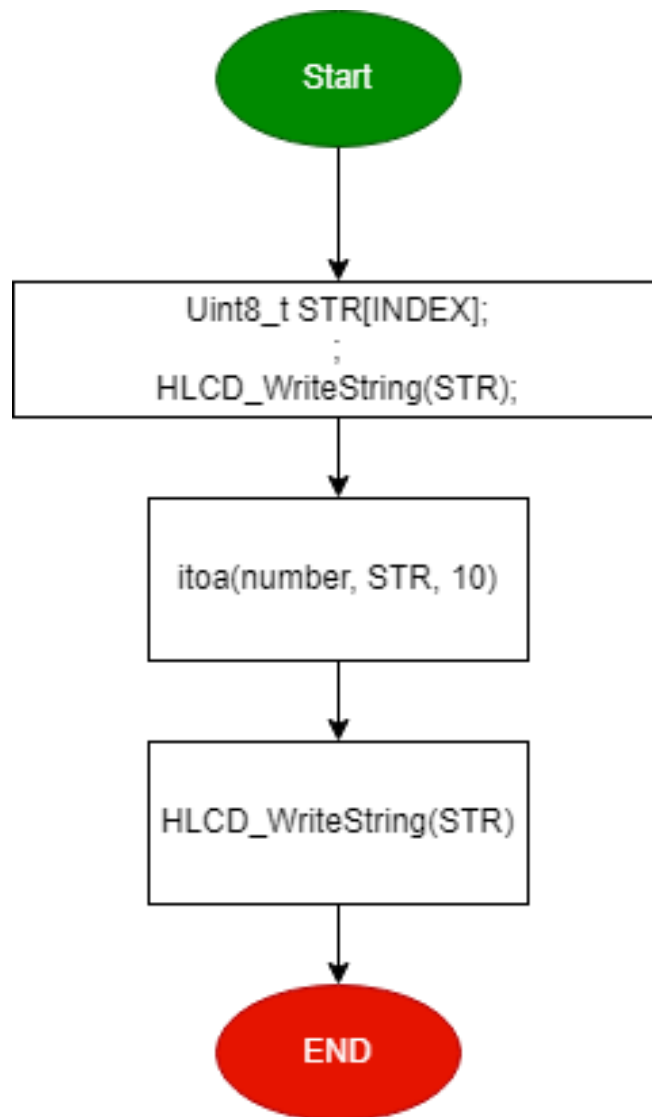
```
void HLCD_WriteString (Uint8_t* str);
```



```
void HLCD_vidCreatCustomChar (Uint8_t* pu8custom, Uint8_t u8Location);
```



```
void HLCD_WriteInt (Uint32_t number);
```



KEYPAD

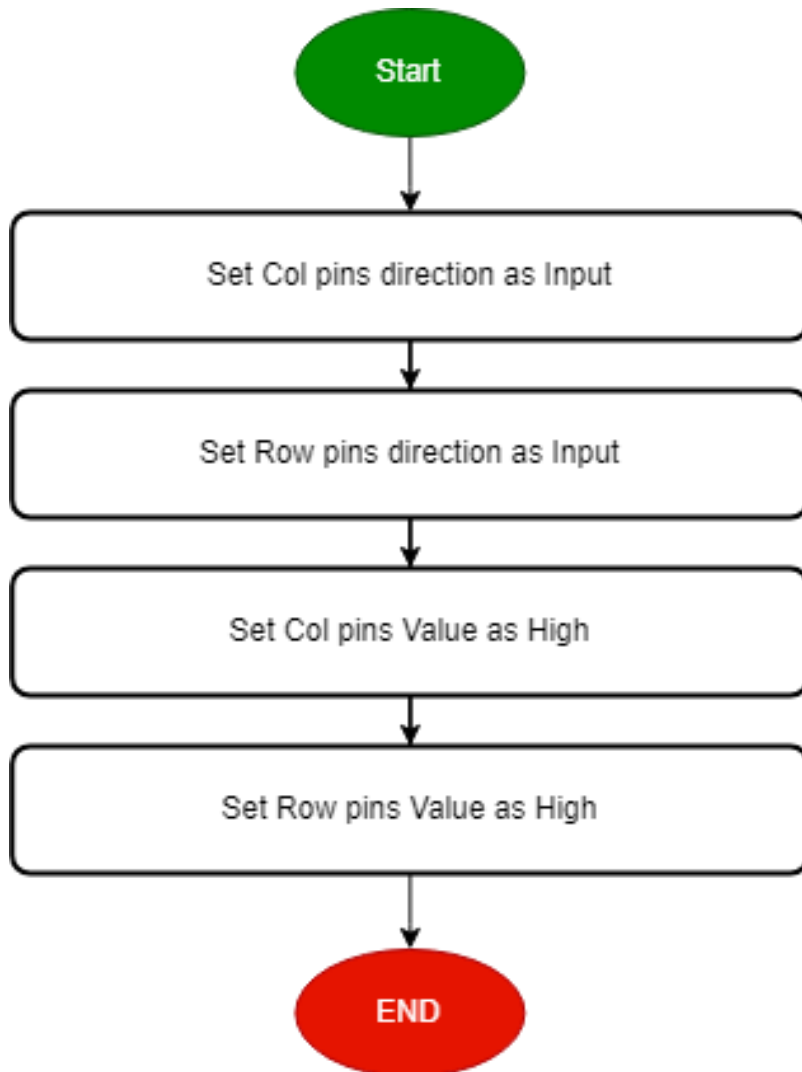
APIs

```
/*
 * Author          : Bassel Yasser Mahmoud
 * Function        : KEYPAD_vidInit_V2
 * Description     : KEYPAD Initialization
 * in[1]           : void
 * Return          : void
 */
void KEYPAD_vidInit_V2(void);

/*
 * Author          : Bassel Yasser Mahmoud
 * Function        : KEYPAD_u8GetPressed_V2
 * Description     : KEYPAD get pin status
 * in[1]           : void
 * Return          : Uint8_t {Pin Status}
 */
Uint8_t KEYPAD_u8GetPressed_V2(void);
```


Flowchart

```
void KEYPAD_vidInit_V2(void);
```



```
uint8_t KEYPAD_u8GetPressed_V2(void);
```

