

[\(/wiki\)](#)[\(/shop\)](#)

Interfacing glcd with 8051

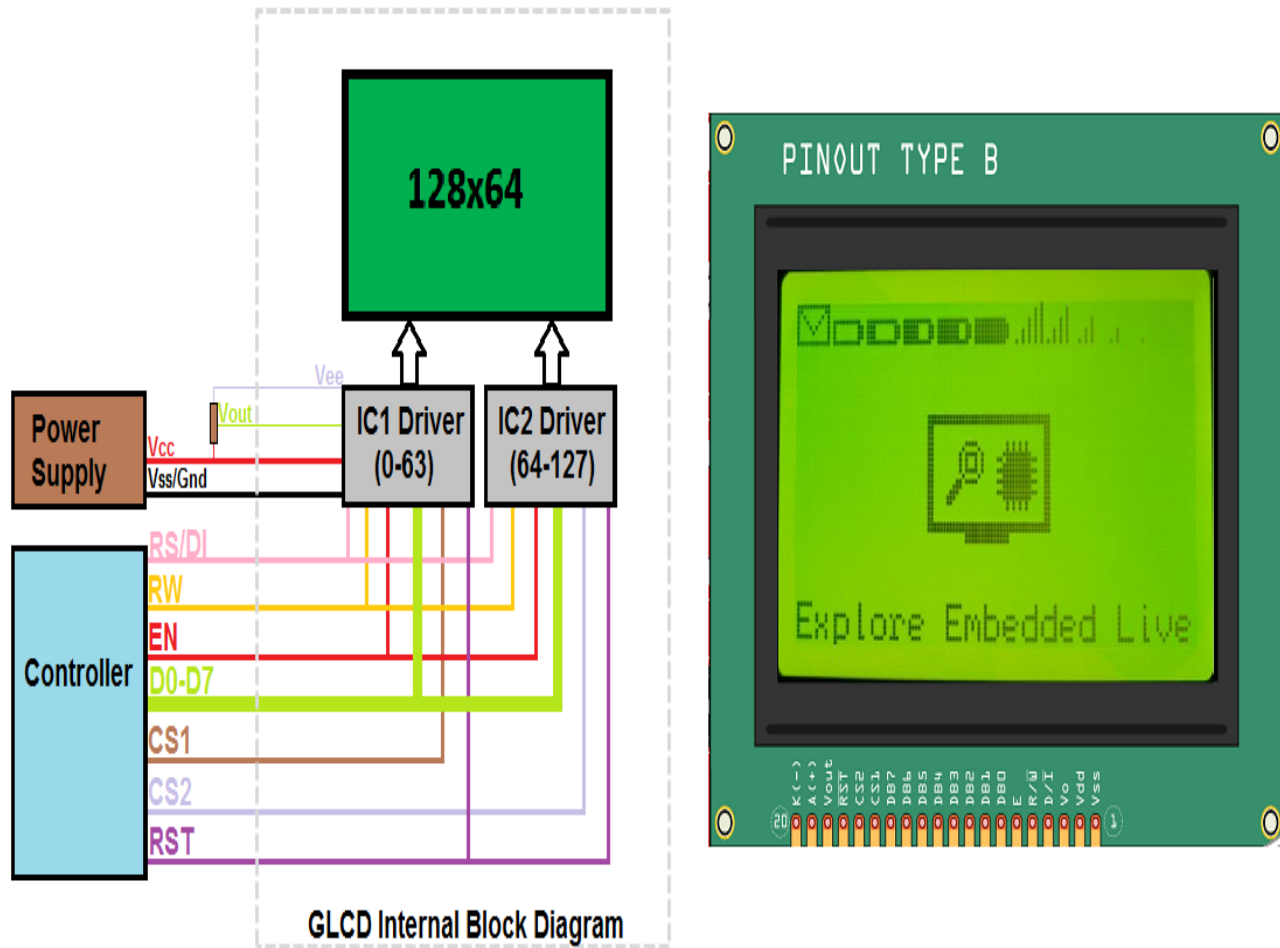
In this tutorial, we will see how to interface and graphical LCD(GLCD) with 8051. We will be interfacing KS0108 controller based JHD12864E display. There are many displays out there based on KS0108 or compatible display controller. They all work the same way but make sure to check the datasheet for the pin diagram before doing the connection.

We will look at the working of the display, the hardware setup and programming with 8051. We have it tested and working on 8051, AVR, PIC and ARM. We have similar tutorials on these MCUs in respective controller tutorial section.

Unlike a 16 x 2 display, this does not have a character map for ASCII values stored in its ROM. However, it allows us the flexibility of creating fonts like Arial, times new roman etc. We could also display bitmap images on it and stretch it little further we can make GUI's and little animation, but that's for another day. So let's get started.

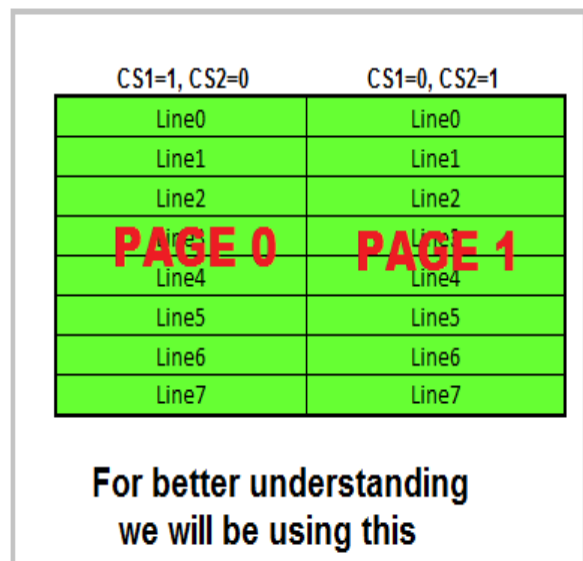
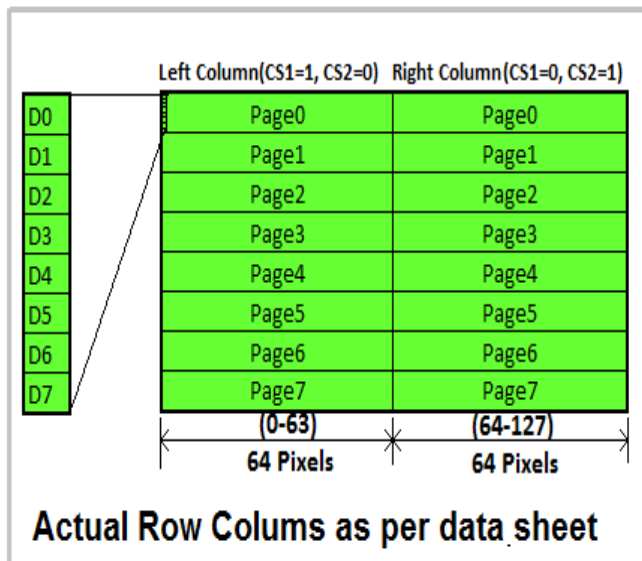
GLCD Internals And Pinout

Below image shows the internal block diagram of 128x64 GLCD along with its pin out.



10	D3	Data Bit 3
11	D4	Data Bit 4
12	D5	Data Bit 5
13	D6	Data Bit 6
14	D7	Data Bit 7/Busy Flag
15	CS1	Chip Select for IC1/PAGE0
16	CS2	Chip Select for IC2/PAGE1
17	RST	Reset the LCD module
18	VEE	Negative voltage used along with Vcc for brightness control
15	A/LED+	Back-light Anode(+)
16	K/LED-	Back-Light Cathode(-)

Page, Line and Cursor Selection



(/wiki/File:GLCD_Pages.png) Lets view the GLCD as a open book with 2pages constisting of 8lines on each page.

Each line has 64 cursors positions to display data/images.

The required page can be selected using CS1,CS2 pins as shown below.

CS2-CS1:Chip Select Lines

00 = None

01 = Page 0

10 = Page 1

11 = Both Pages

Line Selection

To select the lines we need to send the command/line address to GLCD.
The line address starts from 0xb8 and goes till 0xbf as shown below.

7	6	5	4	3	2	1	0
1	0	1	1	1	Y2	Y1	Y0

Y2-Y0:Line Selection

000 = Line0 (Address = 0xB8)

001 = Line1 (Address = 0xB9)

010 = Line2 (Address = 0xBA)

011 = Line3 (Address = 0xBB)

100 = Line4 (Address = 0xBC)

101 = Line5 (Address = 0xBD)

110 = Line6 (Address = 0xBE)

111 = Line7 (Address = 0xBF)

Cursor/Char Position

To set the cursor position(0-63) we need to send its address to GLCD.
The cursor positions address starts from 0x40 and goes till 0x7f as shown below.

7	6	5	4	3	2	1	0
0	1	x5	x4	x3	x2	x1	x0

x5-x0:Line Selection

000000 = Cursor Position 0 (Address = 0x40)

000001 = Cursor Position 1 (Address = 0x41)

000010 = Cursor Position 2 (Address = 0x42)

,

111111 = Cursor Position 63 (Address = 0x7F)

Instruction Set

Below is the complete instruction table.

	Instruction	RS	R/W	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0	Function
	Display ON/OFF	L	L	L	L	H	H	H	H	H	L/H	Controls the display on or off. Internal status and display RAM data is not affected. L:OFF, H:ON
Cursor Position	Set Address (X-Address)	L	L	L	H	X-address (0~63)						Sets the address in the address counter.
Line Numbers	Set Page Line	L	L	H	L	H	H	H		Page Line (0~7)		Sets the address at the address register.
Not Used Dont worry about this	Display Start Line	L	L	H	H	Display start line (0~63)						Indicates the display data RAM displayed at the top of the screen.
	Status Read	L	H	B U S Y	L	O N / O F F	R E S E T	L	L	L	L	Read status. BUSY L: Ready H: In operation ON/OFF L: Display ON H: Display OFF RESET L: Normal H: Reset
	Write Display Data	H	L	Write Data								Writes data (DB0:7) into display data RAM. After writing instruction, Y address is increased by 1 automatically.
	Read Display Data	H	H	Read Data								Reads data (DB0:7) from display data RAM to the data bus.

Modified Instruction Set for better understanding

(/wiki/File:GLCD_InstructionSet.png)

Steps

Steps for Sending Command:

- step1: Send the I/P command to LCD.
- step2: Select the Control Register by making RS low.
- step3: Select Write operation making RW low.
- step4: Send a High-to-Low pulse on Enable PIN with some delay_us.

```
1 /* Function to send the command to LCD */
```

```

2 void Glcd_CmdWrite(char cmd)
3 {
4     GlcdDataBus = cmd;    //Send the Command
5     RS = 0;               // Send LOW pulse on RS pin for selecting Command register
6     RW = 0;               // Send LOW pulse on RW pin for Write operation
7     EN = 1;               // Generate a High-to-low pulse on EN pin
8     delay(100);
9     EN = 0;
10
11     delay(1000);
12 }

```

Bagwan/4b4b84c5fb48f5b7635b7e723570f791/raw/dd635aadcf21589d5d7477f0b8478efb1d2a55b1/8051_glcdCmdWrite.c)
 8051_glcdCmdWrite.c (https://gist.github.com/SaheblalBagwan/4b4b84c5fb48f5b7635b7e723570f791#file-8051_glcdcmdwrite-c) hosted with ❤ by GitHub (<https://github.com>)

Steps for Sending Data:

- step1: Send the character to LCD.
- step2: Select the Data Register by making RS high.
- step3: Select Write operation making RW low.
- step4: Send a High-to-Low pulse on Enable PIN with some delay.

```

1 /* Function to send the data to LCD */
2 void Glcd_DataWrite(char dat)
3 {
4     GlcdDataBus = dat;    //Send the data on DataBus
5     RS = 1;               // Send HIGH pulse on RS pin for selecting data register
6     RW = 0;               // Send LOW pulse on RW pin for Write operation
7     EN = 1;               // Generate a High-to-low pulse on EN pin
8     delay(100);
9     EN = 0;
10
11     delay(1000);
12 }

```

blalBagwan/6f62052188f834afdcbe3b3cefd81cc1/raw/e9d5af4f8534ffe1612eed71bdbfb0326749cb1e/8051_glcdDataWrite.c)
 8051_glcdDataWrite.c (https://gist.github.com/SaheblalBagwan/6f62052188f834afdcbe3b3cefd81cc1#file-8051_glcddatawrite-c) hosted with ❤ by GitHub (<https://github.com>)

Code

Below is the sample code for displaying HELLO WORLD on two different pages of GLCD.

```
1  #include<reg51.h>
2
3  /* Configure the data bus and Control bus as per the hardware connection
4     Dstatus bus is connected to P20:P27 and control bus P00:P04*/
5  #define GlcdDataBus  P2
6  sbit RS  = P0^0;
7  sbit RW  = P0^1;
8  sbit EN  = P0^2;
9  sbit CS1 = P0^3;
10 sbit CS2 = P0^4;
11
12 /* 5x7 Font including 1 space to display HELLO WORLD */
13 char H[]={0x7F, 0x08, 0x08, 0x08, 0x7F, 0x00};
14 char E[]={0x7F, 0x49, 0x49, 0x49, 0x41, 0x00};
15 char L[]={0x7F, 0x40, 0x40, 0x40, 0x40, 0x00};
16 char O[]={0x3E, 0x41, 0x41, 0x41, 0x3E, 0x00};
17
18 char W[]={0x3F, 0x40, 0x38, 0x40, 0x3F, 0x00};
19 char R[]={0x7F, 0x09, 0x19, 0x29, 0x46, 0x00};
20 char D[]={0x7F, 0x41, 0x41, 0x22, 0x1C, 0x00};
21
22
23 /* local function to generate delay */
24 void delay(int cnt)
25 {
26     int i;
27     for(i=0;i<cnt;i++);
28 }
29
30
31 void Glcd_SelectPage0() // CS1=1, CS2=0
32 {
33     CS1 = 1;
34     CS2 = 0;
35 }
36
37 void Glcd_SelectPage1() // CS1=0, CS1=1
38 {
39     CS1 = 0;
40     CS2 = 1;
41 }
```

```
42
43  /* Function to send the command to LCD */
44  void Glcd_CmdWrite(char cmd)
45  {
46      GlcdDataBus = cmd;    //Send the Command
47      RS = 0;               // Send LOW pulse on RS pin for selecting Command register
48      RW = 0;               // Send LOW pulse on RW pin for Write operation
49      EN = 1;               // Generate a High-to-low pulse on EN pin
50      delay(100);
51      EN = 0;
52
53      delay(1000);
54  }
55
56  /* Function to send the data to LCD */
57  void Glcd_DataWrite(char dat)
58  {
59      GlcdDataBus = dat;    //Send the data on DataBus
60      RS = 1;               // Send HIGH pulse on RS pin for selecting data register
61      RW = 0;               // Send LOW pulse on RW pin for Write operation
62      EN = 1;               // Generate a High-to-low pulse on EN pin
63      delay(100);
64      EN = 0;
65
66      delay(1000);
67  }
68
69  void Glcd_DisplayChar(char *ptr_array)
70  {
71      int i;
72      for(i=0;i<6;i++) // 5x7 font, 5 chars + 1 blankspace
73          Glcd_DataWrite(ptr_array[i]);
74  }
75
76
77  int main()
78  {
79      /* Select the Page0/Page1 and Turn on the GLCD */
80      Glcd_SelectPage0();
81      Glcd_CmdWrite(0x3f);
82      Glcd_SelectPage1();
83      Glcd_CmdWrite(0x3f);
84      delay(100);
85
```



```

86      /* Select the Page0/Page1 and Enable the GLCD */
87      Glcd_SelectPage0();
88      Glcd_CmdWrite(0xc0);
89      Glcd_SelectPage1();
90      Glcd_CmdWrite(0xc0);
91      delay(100);
92
93      Glcd_SelectPage0(); // Display HELLO on Page0, Line1
94      Glcd_CmdWrite(0xb8);
95      Glcd_DisplayChar(H);
96      Glcd_DisplayChar(E);
97      Glcd_DisplayChar(L);
98      Glcd_DisplayChar(L);
99      Glcd_DisplayChar(O);
100
101      Glcd_SelectPage1(); // Display WORLD on Page1, Last line
102      Glcd_CmdWrite(0xbf);
103      Glcd_DisplayChar(W);
104      Glcd_DisplayChar(O);
105      Glcd_DisplayChar(R);
106      Glcd_DisplayChar(L);
107      Glcd_DisplayChar(D);
108
109      while(1);
110  }

```

lBagwan/d464c084e66d826c9ded648308909c01/raw/155dc94981f6475ca575363bd1f8a034e5152f88/8051_glcdExample1.c)
 8051_glcdExample1.c (https://gist.github.com/SaheblalBagwan/d464c084e66d826c9ded648308909c01#file-8051_glcdexample1-c) hosted with ❤ by GitHub (<https://github.com>)

Using ExploreEmbedded GLCD Lib

```

1  #include "glcd.h"    //User defined LCD library which contains the lcd routines
2  #include "delay.h"
3
4  /* start the main program */
5  void main()
6  {
7      /* Initialize the glcd before displaying any thing on the lcd */
8      GLCD_Init();
9
10     GLCD_Printf("Interfacing\n\n");
11     GLCD_Printf("  KS108 128x64\n\n");

```

```
12     GLCD_Printf("    With 8051");
13     GLCD_GoToLine(6);
14     GLCD_Printf("    ExploreEmbedded");
15
16     while(1);
17 }
```

lBagwan/35237f505ddde414428854941282d0d6/raw/77ef183748ea12cd0a8711547b833c125867d0ce/8051_glcdExample2.c)
8051_glcdExample2.c (https://gist.github.com/SaheblalBagwan/35237f505ddde414428854941282d0d6#file-8051_glcdexample2-c) hosted with ❤ by GitHub (<https://github.com>)



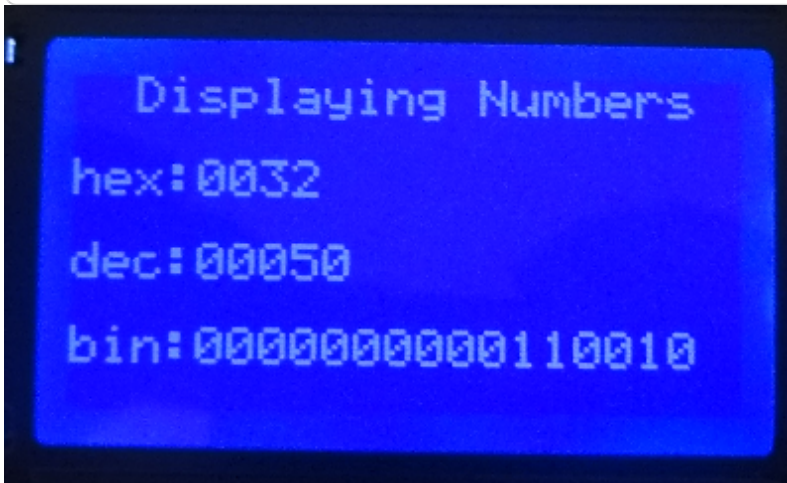
(/wiki/File:0Glcd_CharDisplay8051.png)

Displaying Numbers

```
1  #include "glcd.h"
2  #include "delay.h"
3
4  void main()
5  {
6      int count=0;
7
8      GLCD_Init();
9      GLCD_DisplayString("  Displaying Numbers");
10
11     while(1)
12     {
13         GLCD_GoToLine(2);
14         GLCD_Printf("hex:%4x \n\ndec:%5d \n\nbin:%16b",count,count,count);
15         DELAY_ms(100);
16         count++;
17     }
```

```
18 }
```

bagwan/17b64c4e262e9173b3823d12d1eced4a/raw/a6535d1e378c9e88328bc940fb15ced4f1e97cdc/glcd_DisplayNumbers.c)
glcd_DisplayNumbers.c (https://gist.github.com/SaheblalBagwan/17b64c4e262e9173b3823d12d1eced4a#file-glcd_displaynumbers-c) hosted with ❤️ by GitHub (<https://github.com>)

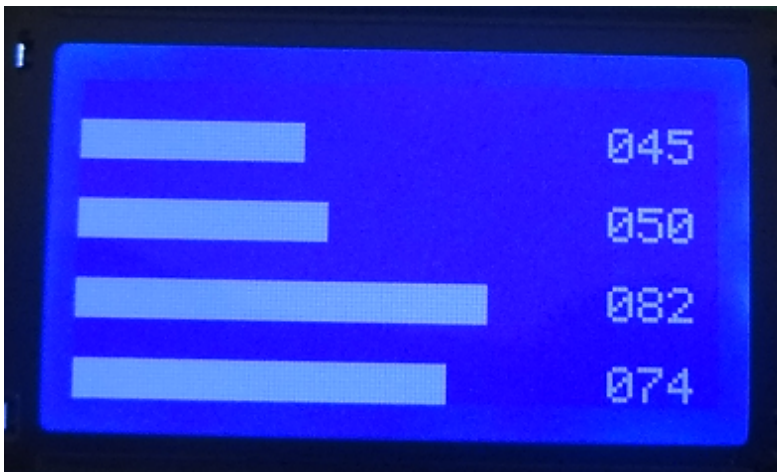


(/wiki/File:Glcd_DisplayNumber.png)

Displaying Bar Graphs

```
1  #include "glcd.h"
2
3  void main()
4  {
5      GLCD_Init();
6
7      GLCD_HorizontalGraph(0,45);
8      GLCD_HorizontalGraph(1,50);
9      GLCD_HorizontalGraph(2,82);
10     GLCD_HorizontalGraph(3,74);
11
12     while(1);
13 }
```

bagwan/49e0ad0b4d5617535430989c6736d8af/raw/eb690cf97c5c2b384b0ed9d9b81b80959a8b6190/glcd_HorizontalGraph.c)
glcd_HorizontalGraph.c (https://gist.github.com/SaheblalBagwan/49e0ad0b4d5617535430989c6736d8af#file-glcd_horizontalgraph-c) hosted with ❤️ by GitHub (<https://github.com>)



(/wiki/File:Glcd_HorizontalGraph.png)

```

1  #include "glcd.h"
2
3  void main()
4  {
5      GLCD_Init();
6
7      GLCD_VerticalGraph(0,45);
8      GLCD_VerticalGraph(1,50);
9      GLCD_VerticalGraph(2,82);
10     GLCD_VerticalGraph(3,74);
11
12     while(1);
13 }

```

lalBagwan/28b1f36fc769b7ab785c414672380eff/raw/d0bea2520e320d2a707fd36a906a8c0e3e2afc9a/glcd_VerticalGraph.c)
 glcd_VerticalGraph.c (https://gist.github.com/SaheblalBagwan/28b1f36fc769b7ab785c414672380eff#file-glcd_verticalgraph-c) hosted with ❤ by GitHub (<https://github.com>)



(/wiki/File:Glcd_VerticalGraph.png)

Downloads

Download the sample code and design files from this link
(https://github.com/ExploreEmbedded/8051_DevelopmentBoard).

Have an opinion, suggestion, question or feedback about the article let it out here!

ALSO ON EXPLOREEMBEDDED.COM/WIKI

Hornbill ...

6 years ago • 1 comment

We have said several times that the ESP32 is feature rich. We will use two of ...

LPC2148: ...

4 years ago • 1 comment

Objective In this tutorial, we will be discussing how to upload the .bin file to ...

AWS IOT ...

6 years ago • 37 comments

In the earlier tutorial we discussed how to setup AWS IOT for an Amazon user ...

Setti

6 years ago

Eclipse develo with r

1 Comment

 Login ▼

Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS 

Name



Share

Best Newest Oldest



Shiva Vishnu

6 years ago

can you please share glcd library?

o o Reply • Share ›

Subscribe

Privacy

Categories (/wiki/Special:Categories): 8051 tutorials (/wiki/Category:8051_tutorials)
| GLCD KS108 (/wiki/Category:GLCD_KS108)

Subscribe to hear about our latest Explorations!

name@example.com

SUBSCRIBE

[Contact \(/contact\)](#) [About \(/about\)](#) [Warranty \(/refund\)](#) [Terms & Conditions \(/terms\)](#) [Reward points \(/rewards\)](#)

<https://twitter.com/exploreembedded> <https://www.facebook.com/ExploreEmbedded/>

<https://www.youtube.com/channel/UCvXGpvPuosEI-ALxvCrSbaA> <https://github.com/ExploreEmbedded>



Now shipping worldwide from India with