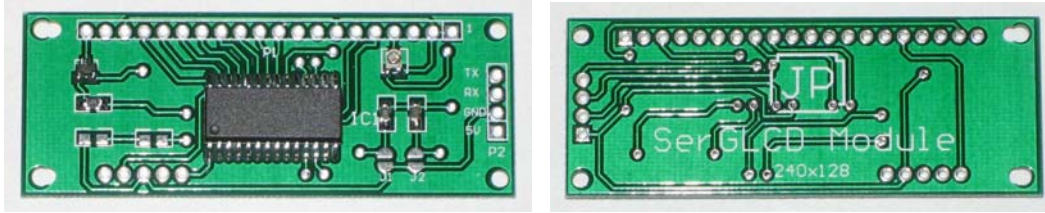


JP Serial 240x128 GLCD Module



JP serial 240x128 module is a simple and cost effective interface controller module. It supports 240x128 Graphic LCD display base on T6963C controller. It integrates T6963C display commands into a PIC18F26K22 and includes an 8x8, 12x16, 13x24, 19x32, 28x48 font library. Also it has 7 segments number library (16x24, 18x32, 28x48, and 34x64) in it. It also works with any microcontroller capable of accepting asynchronous serial data.

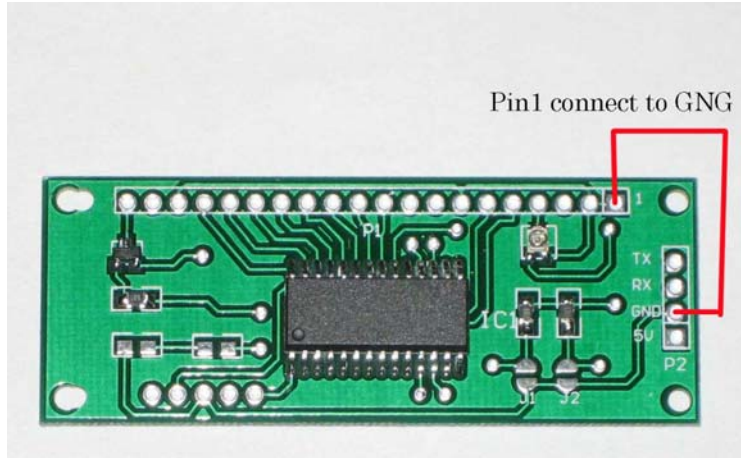
SPECS:

Power: 5V VDC
Size: 65mm x 25mm (2.1" x 1.0")
Speed: 4800, 9600, 19200 and 38400 Baud.

GLCD PIN FUNCTIONS:

Pin1: Vss
Pin2: Vss
Pin3: Vdd
Pin4: Vo
Pin5: E
Pin6: R/W
Pin7: CE
Pin8: RS
Pin9: RST
Pin10: D0
Pin11: D1
Pin12: D2
Pin13: D3
Pin14: D4
Pin15: D5
Pin16: D6
Pin17: D7
Pin18: FS
Pin19: Vee
Pin20: LED+ (Backlight)

Note: Pin1 must connect to ground.



P2 PIN FUNCTION:

Rx: Serial input connection to JP Module. The module allows 4800, 9600, 19200 or 38400 baud based on J1 and J2 settings.

Tx: Serial output connection to JP Module. The module allows 4800, 9600, 19200 or 38400 baud based on J1 and J2 settings.

J1 and J2 Settings:

J1	J2	Baud
-	-	4800
x	-	9600
-	x	19200
x	x	38400

x = connected

- = disconnected

GND: Power supply and serial ground. This MUST also be connected to ground on the device to allow the serial data to be sent to the module.

5V: Supply voltage to module. V_{in} may be 5.0V ($\pm 0.5V$), with 25 milliamps of current.

SERIAL DATA FORMAT

The serial data format is eight data bits, no parity and 1 stop bit (8N1). Characters are sent using standard ASCII values. Baud rate may be 2400, 4800, 9600 and 19200 depending on the settings of J1 and J2.

JP Serial 240x128 Graphic LCD Display Module Operate Commands:

Command(hex)	Command (Dec)	Description	Parameter
\$B0	176	Frame EEPROM Disable (Default)	None
\$B1	177	Frame EEPROM Enable	None
\$B2	178	Enable JP Logo Test (default)	None
\$B3	179	Disable JP Logo Test	None
\$B4	180	Turn off Back LED	None

\$B5	181	Turn on Back LED	None
\$B6	182	decrease Back LED	None
\$B7	183	Increase Back LED	None
\$B8	184	Turn on Cursor	None
\$B9	185	Turn off Cursor	None
\$BA	186	Turn on Cursor Blink	None
\$BB	187	Turn off Cursor Blink	None
\$BC	188	Set Cursor Height	0 - 7
\$BD	189	Clean Graphic panel	None
\$BE	190	Clean Text Panel	None
\$BF	191	Clean Graphic and Text Panel	None
\$43	67	Write Dot	X, y, color
\$46	70	Clean Screen	None
\$48	72	Load Image to Screen	None
\$4A	74	Copy Screen Data to Frame Ram	None
\$4B	75	Send Screen Data to Uart	None
\$4C	76	Turn on Graphic Panel	
\$4D	77	Turn off Graphic Panel	
\$4E	78	Turn on Text Panel	
\$4F	79	Turn off text Panel	
\$50	80	Write Text I Text panel with OR Mode	Text string, x, y
\$51	81	Write Text I Text panel with XOR Mode	Text string, x, y
\$52	82	Write Text I Text panel with AND Mode	Text string, x, y
\$53	83	Write Text I Text panel with TEXT Mode	Text string, x, y
\$60	96	Write 28x48 Text String in Screen	x, y, color, space, Text string
\$61	97	Write 19x3 Text String in Screen	x, y, color, space, Text string
\$62	98	Write 13x24 Text String in Screen	x, y, color, space, Text string
\$63	99	Write 12x16 Text String in Screen	x, y, color, space, Text string
\$64	100	Write 8x8 Text String in Screen	x, y, color, space, Text string
\$70	112	Write 28x48 a char in Screen	x, y, char, color
\$71	113	Write 19x3 a char in Screen	x, y, char, color
\$72	114	Write 13x24 a char in Screen	x, y, char, color
\$73	115	Write 12x16 a char in Screen	x, y, char, color
\$74	116	Write 8x8 a char in Screen	x, y, char, color
\$80	128	Write a 34x64 7 Segment Char in Screen	x, y, char, color
\$81	129	Write a 28x48 7 Segment Char in Screen	x, y, char, color
\$82	130	Write a 18x32 7 Segment Char in Screen	x, y, char, color
\$83	131	Write a 16x24 7 Segment Char in Screen	x, y, char, color

\$90	144	Draw a H Line	x1, x2, y, color
\$91	145	Draw a V Line	x, y1, y2, color
\$92	146	Draw a Line	x1, y1, x2, y2, color
\$93	147	Draw a Rectangle	x, y, w, h, color
\$94	148	Draw a Fill Rectangle	x, y, w, h, color
\$95	149	Draw a Square	x, y, l, color
\$96	150	Draw a Fill Square	x, y, l, color
\$97	151	Draw a Circle	x0, y0, r, color
\$98	152	Draw a Fill Circle	x0, y0, r, color
\$99	153	Draw a Ellipse	x0, y0, xr, yr, color
\$9A	154	Draw a Fill Ellipse	x0, y0, xr, yr, color

Note:

JP Serial 240x128 Graphic Display Module integrates T6963C display commands into a PIC18F26K22.

Because the chip doesn't have enough ram to support a virtual frame, so the frame has 5 bottom lines using EEPROM. If dots are wrote in EEPROM area, it will be slow.

When you send command and data to the module, it needs a pause to process them without missing the next command and data.

Please read T6963C datasheet and Pic18F26K22 Datasheet first before purchasing. (www.microchip.com)

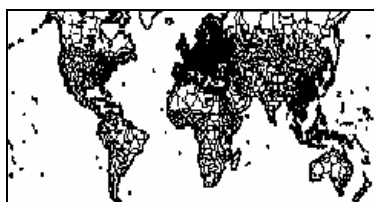
LIABILITY WARNING

This device should be used only for experimental purposes. It has NOT gone through extensive testing and it could erase or corrupt some or all data on media cards that are inside the device. You assume to take your own risk when you purchase this device, and release the responsibility and liability from the manufacturer with no harm.

REGULATORY WARNING

This device is intended solely for experimental purposes; it is not in finished product form and is NOT FCC approved. If you wish to install these modules into non-experimental final finished products, you will be responsible to have the modules approved by the FCC at your own cost.

Test Picture



World Map



Countryside



Cartoon

3, 63, 63, 195, 3, 192, 0, 0, 0, 31, 128, 127, 159, 159, 225, 255, 255, 249, 248, 126, 7, 255, 255, 195, 255, 63, 0, 63, 63, 195, 63, 252, 63, 3, 63, 0, 0, 0, 0, 31, 255, 159, 224, 127, 255, 255, 255, 249, 254, 7, 129, 255, 255, 252, 195, 195, 3, 3, 63, 192, 63, 252, 63, 3, 63, 0, 0, 0, 0, 31, 255, 159, 224, 127, 255, 255, 255, 249, 254, 7, 129, 255, 255, 252, 195, 195, 3, 3, 63, 192, 63, 252, 63, 3, 63, 0, 0, 0, 0, 31, 255, 159, 224, 127, 255, 255, 255, 249, 254, 7, 129, 255, 255, 252, 195, 195, 3, 3, 63, 192, 60, 192, 255, 63, 255, 60, 0, 0, 0, 96, 127, 159, 255, 255, 255, 159, 255, 248, 126, 1, 129, 255, 255, 252, 60, 195, 192, 192, 192, 252, 0, 192, 255, 60, 255, 60, 0, 0, 31, 128, 31, 225, 255, 255, 255, 255, 255, 128, 7, 129, 248, 121, 255, 255, 0, 252, 252, 60, 252, 255, 0, 192, 255, 60, 255, 60, 0, 0, 31, 128, 31, 225, 255, 255, 255, 255, 255, 128, 7, 129, 248, 121, 255, 255, 0, 252, 252, 60, 252, 255, 3, 195, 255, 60, 252, 3, 192, 252, 126, 0, 31, 225, 255, 255, 255, 255, 255, 248, 1, 248, 126, 127, 255, 255, 195, 195, 60, 252, 0, 255, 3, 195, 255, 60, 252, 3, 192, 252, 126, 0, 31, 225, 255, 255, 255, 255, 255, 248, 1, 248, 126, 127, 255, 255, 195, 195, 60, 252, 0, 255, 3, 195, 255, 60, 252, 3, 192, 252, 126, 0, 31, 225, 255, 255, 255, 255, 255, 248, 1, 248, 126, 127, 255, 255, 195, 195, 60, 252, 0, 255, 3, 3, 255, 60, 255, 252, 63, 255, 224, 0, 1, 224, 127, 255, 255, 255, 255, 248, 1, 248, 6, 7, 254, 63, 252, 255, 63, 60, 60, 63, 3, 3, 255, 60, 255, 252, 63, 255, 224, 0, 1, 224, 127, 255, 255, 255, 255, 248, 1, 248, 6, 7, 254, 63, 252, 255, 63, 60, 60, 63, 63, 0, 252, 252, 255, 255, 255, 252, 0, 0, 1, 224, 127, 255, 255, 255, 255, 254, 0, 126, 1, 134, 127, 252, 252, 63, 63, 63, 60, 63, 63, 0, 252, 252, 255, 255, 255, 252, 0, 0, 1, 224, 127, 255, 255, 255, 255, 254, 0, 126, 1, 134, 127, 252, 252, 63, 63, 63, 60, 63, 63, 60, 255, 255, 255, 252, 3, 192, 0, 0, 0, 126, 127, 255, 255, 255, 255, 135, 128, 6, 1, 129, 135, 255, 60, 3, 195, 195, 3, 63, 63, 60, 255, 255, 255, 252, 3, 192, 0, 0, 0, 126, 127, 255, 255, 255, 255, 135, 128, 6, 1, 129, 135, 255, 60, 3, 195, 195, 3, 63, 63, 60, 255, 255, 255, 252, 3, 192, 0, 0, 0, 126, 127, 255, 255, 255, 255, 135, 128, 6, 1, 129, 135, 255, 60, 3, 195, 195, 3, 63, 255, 3, 255, 255, 255, 3, 252, 0, 0, 0, 0, 126, 31, 255, 255, 255, 255, 249, 248, 1, 128, 121, 135, 192, 255, 0, 195, 195, 195, 252, 3, 255, 195, 255, 63, 0, 0, 0, 0, 0, 31, 159, 255, 255, 255, 254, 6, 126, 1, 248, 120, 121, 252, 252, 192, 255, 192, 192, 192, 252, 3, 255, 195, 255, 63, 0, 0, 0, 0, 0, 31, 159, 255, 255, 255, 254, 6, 126, 1, 248, 120, 121, 252, 252, 192, 255, 192, 192, 192, 252, 195, 63, 192, 255, 192, 0, 0, 0, 1, 129, 255, 255, 255, 255, 225, 254, 0, 6, 0, 126, 6, 121, 252, 60, 60, 60, 252, 252, 192, 252, 195, 63, 192, 255, 192, 0, 0, 0, 1, 129, 255, 255, 255, 255, 225, 254, 0, 6, 0, 126, 6, 121, 252, 60, 60, 60, 252, 252, 192, 252, 195, 63, 192, 255, 192, 0, 0, 0, 1, 129, 255, 255, 255, 255, 225, 254, 0, 6, 0, 126, 6, 121, 252, 60, 60, 60, 252, 252, 192, 192, 192, 192, 3, 255, 195, 255, 0, 0, 0, 0, 30, 31, 255, 159, 255, 255, 224, 127, 128, 0, 0, 1, 248, 1, 255, 3, 60, 60, 60, 192, 192, 192, 255, 255, 195, 252, 3, 192, 3, 158, 1, 255, 159, 255, 255, 255, 255, 255, 128, 1, 128, 6, 126, 6, 63, 195, 60, 63, 63, 60, 60, 192, 255, 255, 195, 252, 3, 192, 3, 158, 1, 255, 159, 255, 255, 255, 255, 255, 128, 1, 128, 6, 126, 6, 63, 195, 60, 63, 63, 60, 60, 195, 255, 255, 195, 195, 3, 192, 60, 31, 254, 31, 255, 255, 255, 255, 255, 225, 128, 1, 248, 6, 6, 0, 63, 255, 195, 195, 63, 252, 60, 195, 255, 255, 195, 195, 3, 192, 60, 31, 254, 31, 255, 255, 255, 255, 255, 225, 128, 1, 248, 6, 6, 0, 63, 255, 195, 195, 63, 252, 60, 195, 255, 255, 195, 195, 3, 192, 3, 254, 1, 255, 159, 225, 255, 255, 255, 224, 120, 0, 120, 0, 7, 128, 3, 195, 60, 195, 255, 60, 60, 255, 60, 63, 195, 192, 60, 63, 255, 128, 31, 254, 1, 255, 255, 255, 255, 224, 120, 0, 0, 0, 0, 120, 0, 252, 60, 252, 255, 63, 63, 255, 60, 63, 195, 192, 60, 63, 255, 128, 31, 254, 1, 255, 255, 255, 255, 224, 120, 0, 0, 0, 0, 120, 0, 252, 60, 252, 255, 63, 63, 195, 252, 255, 195, 255, 255, 192, 0, 127, 224, 1, 225, 255, 255, 255, 224, 126, 0, 7, 128, 0, 120, 0, 252, 60, 252, 255, 63, 63, 195, 252, 255, 195, 255, 255, 192, 0, 127, 224, 1, 225, 255, 255, 255, 224, 126, 0, 7, 128, 0, 120, 0, 252, 60, 252, 255, 63, 63, 195, 252, 255, 195, 255, 255, 192, 0, 127, 224, 1, 225, 255, 255, 255, 224, 126, 0, 7, 128, 0, 120, 0, 252, 60, 252, 255, 63, 63,

255, 255, 255, 195, 255, 192, 0, 0, 1, 254, 0, 1, 225, 255, 255, 255, 128, 7, 128, 1, 248, 1, 134, 0, 60, 63, 60, 63, 195, 195,
255, 255, 255, 195, 255, 192, 0, 0, 1, 254, 0, 1, 225, 255, 255, 255, 128, 7, 128, 1, 248, 1, 134, 0, 60, 63, 60, 63, 195, 195,
255, 255, 63, 195, 3, 60, 3, 3, 255, 159, 255, 224, 96, 127, 255, 255, 224, 1, 128, 0, 127, 249, 135, 192, 60, 63, 63, 63, 255, 195,
255, 252, 63, 195, 255, 63, 195, 255, 254, 126, 127, 254, 127, 255, 225, 255, 224, 0, 120, 0, 127, 254, 1, 252, 63, 3, 255, 0, 255, 195,
255, 252, 63, 195, 255, 63, 195, 255, 254, 126, 127, 254, 127, 255, 225, 255, 224, 0, 120, 0, 127, 254, 1, 252, 63, 3, 255, 0, 255, 195,
255, 252, 63, 195, 255, 63, 195, 255, 254, 126, 127, 254, 127, 255, 225, 255, 224, 0, 120, 0, 127, 254, 1, 252, 63, 3, 255, 0, 255, 195,
255, 252, 63, 195, 255, 63, 195, 255, 254, 126, 127, 254, 127, 255, 225, 255, 224, 0, 120, 0, 127, 254, 1, 252, 63, 3, 255, 0, 255, 195,
255, 252, 255, 195, 255, 192, 192, 255, 255, 224, 31, 158, 126, 97, 225, 255, 224, 0, 126, 0, 6, 1, 248, 60, 63, 3, 195, 255, 63, 195,
255, 255, 255, 195, 255, 252, 252, 63, 129, 128, 30, 126, 126, 97, 254, 127, 224, 0, 0, 0, 1, 128, 127, 192, 195, 3, 195, 255, 63, 195,
255, 255, 255, 195, 255, 252, 252, 63, 129, 128, 30, 126, 126, 97, 254, 127, 224, 0, 0, 0, 1, 128, 127, 192, 195, 3, 195, 255, 63, 195,
255, 255, 63, 195, 255, 255, 0, 0, 30, 0, 31, 224, 31, 255, 254, 127, 254, 0, 0, 0, 0, 120, 120, 3, 195, 195, 192, 195, 255, 192,
255, 255, 63, 195, 255, 255, 0, 0, 30, 0, 31, 224, 31, 255, 254, 127, 254, 0, 0, 0, 0, 120, 120, 3, 195, 195, 192, 195, 255, 192,
255, 255, 63, 195, 255, 255, 0, 0, 30, 0, 31, 224, 31, 255, 254, 127, 254, 0, 0, 0, 0, 120, 120, 3, 195, 195, 192, 195, 255, 192,
255, 255, 63, 195, 255, 195, 0, 3, 225, 225, 224, 0, 127, 254, 127, 159, 254, 0, 0, 0, 0, 7, 128, 3, 192, 195, 252, 255, 255, 192,
255, 255, 63, 195, 255, 195, 0, 3, 225, 225, 224, 0, 127, 254, 127, 159, 254, 0, 0, 0, 0, 7, 128, 3, 192, 195, 252, 255, 255, 192,
255, 255, 63, 195, 195, 3, 192, 3, 158, 0, 0, 0, 127, 254, 127, 159, 255, 128, 0, 0, 0, 0, 0, 3, 192, 255, 60, 255, 195, 255,
255, 255, 63, 195, 195, 3, 192, 0, 0, 0, 0, 0, 31, 254, 126, 31, 255, 128, 0, 0, 0, 0, 0, 3, 192, 255, 60, 255, 195, 192,
255, 255, 63, 195, 195, 3, 192, 0, 0, 0, 0, 0, 31, 254, 126, 31, 255, 128, 0, 0, 0, 0, 0, 3, 192, 255, 60, 255, 195, 192,
255, 255, 63, 195, 195, 3, 192, 0, 0, 0, 0, 0, 31, 254, 126, 31, 255, 128, 0, 0, 0, 0, 0, 3, 192, 255, 60, 255, 195, 192,
255, 255, 255, 195, 195, 3, 192, 0, 0, 0, 0, 0, 127, 224, 126, 97, 225, 128, 0, 0, 0, 0, 3, 0, 63, 60, 255, 195, 195,
255, 255, 255, 255, 195, 195, 192, 0, 0, 0, 0, 0, 127, 224, 127, 225, 255, 128, 0, 0, 0, 0, 3, 192, 60, 60, 63, 195, 255,
255, 255, 195, 255, 255, 195, 192, 0, 0, 0, 0, 0, 127, 224, 126, 126, 126, 120, 0, 0, 0, 0, 0, 192, 60, 60, 255, 195, 195,
255, 255, 195, 255, 255, 195, 192, 0, 0, 0, 0, 0, 127, 224, 126, 126, 126, 120, 0, 0, 0, 0, 0, 192, 60, 60, 255, 195, 195,
255, 255, 255, 255, 255, 195, 192, 0, 0, 0, 0, 0, 31, 254, 127, 254, 126, 126, 0, 0, 0, 0, 0, 192, 60, 0, 255, 192, 255,
255, 255, 195, 255, 255, 195, 192, 0, 0, 0, 0, 0, 127, 254, 126, 126, 31, 134, 0, 0, 0, 0, 0, 192, 63, 0, 255, 195, 255,
255, 255, 255, 255, 255, 255, 192, 0, 0, 0, 1, 255, 254, 126, 127, 159, 249, 128, 0, 0, 0, 0, 0, 192, 3, 0, 255, 192, 195,
255, 255, 255, 255, 255, 255, 192, 0, 0, 0, 1, 255, 254, 126, 127, 159, 249, 128, 0, 0, 0, 0, 0, 192, 3, 0, 255, 192, 195,
255, 255, 255, 255, 255, 255, 192, 0, 0, 0, 1, 255, 254, 126, 127, 159, 249, 128, 0, 0, 0, 0, 0, 192, 3, 0, 255, 192, 195,
255, 255, 195, 255, 255, 252, 192, 0, 0, 0, 1, 159, 254, 126, 127, 129, 129, 128, 0, 0, 0, 0, 0, 3, 192, 195, 195, 252,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 30, 31, 254, 127, 255, 129, 134, 120, 0, 0, 0, 0, 60, 3, 192, 255, 192, 192,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 30, 31, 254, 127, 255, 129, 134, 120, 0, 0, 0, 0, 60, 3, 192, 255, 192, 192,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 30, 31, 254, 127, 255, 129, 134, 120, 0, 0, 0, 0, 60, 3, 192, 255, 192, 192,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 30, 31, 254, 127, 255, 129, 134, 120, 0, 0, 0, 0, 60, 3, 192, 255, 192, 192,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 0, 31, 254, 31, 129, 128, 0, 6, 0, 0, 0, 0, 60, 0, 195, 255, 3, 195,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 0, 31, 254, 31, 129, 128, 0, 0, 0, 0, 0, 0, 60, 0, 195, 63, 3, 192,
255, 255, 255, 255, 255, 63, 192, 0, 0, 0, 0, 31, 254, 31, 129, 128, 0, 0, 0, 0, 0, 0, 60, 0, 195, 63, 3, 192,
255, 255, 255, 255, 255, 195, 192, 0, 0, 0, 0, 31, 158, 31, 128, 96, 0, 0, 0, 0, 0, 60, 0, 195, 63, 3, 255,
255, 255, 255, 255, 255, 195, 192, 0, 0, 0, 0, 31, 158, 31, 128, 96, 0, 0, 0, 0, 0, 60, 0, 195, 63, 3, 255,
255, 255, 255, 255, 255, 195, 0, 0, 0, 0, 0, 127, 158, 31, 128, 96, 0, 1, 128, 0, 0, 0, 0, 3, 0, 192, 0, 3, 255,
255, 255, 255, 255, 255, 195, 0, 0, 0, 0, 0, 127, 158, 31, 128, 96, 0, 1, 128, 0, 0, 0, 0, 3, 0, 192, 0, 3, 255,
255, 255, 255, 255, 255, 255, 192, 0, 0, 0, 0, 31, 158, 31, 224, 126, 0, 0, 0, 0, 0, 0, 3, 0, 192, 60, 63, 252,
255, 255, 63, 63, 252, 63, 192, 0, 0, 0, 0, 127, 158, 30, 96, 31, 128, 0, 0, 0, 0, 0, 3, 0, 60, 60, 63, 252,
255, 255, 63, 63, 252, 63, 192, 0, 0, 0, 0, 127, 158, 30, 96, 31, 128, 0, 0, 0, 0, 0, 3, 0, 60, 60, 63, 252,
255, 255, 63, 63, 252, 63, 192, 0, 0, 0, 0, 127, 158, 30, 96, 31, 128, 0, 0, 0, 0, 0, 3, 0, 60, 60, 63, 252,
255, 3, 195, 255, 255, 3, 195, 63, 224, 0, 0, 0, 126, 30, 31, 254, 1, 128, 0, 0, 0, 0, 0, 3, 0, 60, 192, 63, 195,
255, 3, 195, 255, 255, 3, 195, 63, 224, 0, 0, 0, 126, 30, 31, 254, 1, 128, 0, 0, 0, 0, 0, 3, 0, 60, 192, 63, 195,
255, 3, 195, 255, 255, 3, 195, 63, 128, 0, 0, 0, 30, 96, 30, 30, 0, 120, 0, 0, 0, 0, 7, 192, 3, 0, 3, 192, 63, 63,
255, 3, 195, 255, 255, 3, 195, 63, 128, 0, 0, 0, 30, 96, 30, 30, 0, 120, 0, 0, 0, 0, 7, 192, 3, 0, 3, 192, 63, 63,
255, 192, 192, 255, 255, 195, 195, 0, 0, 0, 0, 96, 96, 126, 0, 0, 120, 0, 0, 0, 0, 7, 255, 255, 0, 60, 0, 255, 63,

255, 192, 192, 255, 255, 195, 195, 0, 0, 0, 0, 0, 96, 96, 126, 0, 0, 120, 0, 0, 0, 0, 7, 255, 255, 0, 60, 0, 255, 63,
255, 192, 192, 255, 255, 195, 195, 0, 0, 0, 0, 96, 96, 126, 0, 0, 120, 0, 0, 0, 0, 7, 255, 255, 0, 60, 0, 255, 63,
255, 192, 255, 255, 255, 255, 252, 192, 0, 0, 0, 0, 96, 0, 126, 1, 128, 0, 0, 7, 248, 0, 1, 192, 192, 0, 252, 0, 252, 63,
255, 192, 255, 255, 255, 252, 192, 0, 0, 0, 0, 96, 0, 126, 1, 128, 0, 0, 7, 248, 0, 1, 192, 192, 0, 252, 0, 252, 63,
255, 192, 252, 255, 255, 252, 255, 0, 0, 0, 0, 96, 0, 0, 96, 0, 0, 1, 134, 0, 1, 192, 0, 0, 0, 0, 195, 252,
255, 252, 63, 255, 255, 252, 63, 0, 0, 0, 0, 1, 128, 0, 0, 0, 0, 0, 1, 249, 254, 7, 192, 0, 0, 0, 3, 255, 192,
255, 252, 63, 255, 255, 252, 63, 0, 0, 0, 0, 1, 128, 0, 0, 0, 0, 0, 1, 249, 254, 7, 192, 0, 0, 0, 3, 255, 192,
255, 252, 63, 255, 255, 252, 63, 0, 0, 0, 0, 1, 128, 0, 0, 0, 0, 0, 1, 249, 254, 7, 192, 0, 0, 0, 3, 255, 192,
255, 252, 63, 195, 255, 195, 255, 192, 31, 255, 254, 31, 128, 0, 96, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 0, 63, 63, 195,
255, 252, 63, 195, 255, 195, 255, 192, 31, 255, 254, 31, 128, 0, 96, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 0, 63, 63, 195,
63, 255, 3, 255, 255, 255, 60, 255, 254, 0, 0, 0, 0, 127, 128, 0, 0, 0, 0, 6, 7, 135, 192, 0, 0, 60, 252, 252, 63,
63, 255, 3, 255, 255, 255, 60, 255, 254, 0, 0, 0, 0, 127, 128, 0, 0, 0, 0, 6, 7, 135, 192, 0, 0, 60, 252, 252, 63,
63, 255, 0, 252, 255, 192, 255, 3, 224, 96, 0, 0, 1, 255, 128, 0, 0, 0, 0, 6, 1, 248, 0, 0, 0, 60, 195, 192, 252,
63, 255, 0, 252, 255, 192, 255, 3, 224, 96, 0, 0, 1, 255, 128, 0, 0, 0, 0, 6, 1, 248, 0, 0, 0, 60, 195, 192, 252,
63, 255, 0, 252, 255, 192, 255, 3, 224, 96, 0, 0, 1, 255, 128, 0, 0, 0, 0, 6, 1, 248, 0, 0, 0, 60, 195, 192, 252,
3, 255, 0, 252, 255, 252, 3, 192, 0, 0, 0, 31, 255, 128, 0, 0, 0, 0, 1, 128, 0, 0, 0, 0, 0, 63, 63, 0,
3, 255, 0, 252, 255, 252, 3, 192, 0, 0, 0, 31, 255, 128, 0, 0, 0, 0, 1, 128, 0, 0, 0, 0, 0, 63, 63, 0,
3, 255, 0, 63, 63, 63, 192, 255, 128, 31, 128, 1, 255, 255, 224, 0, 0, 0, 0, 0, 0, 126, 0, 0, 0, 0, 0, 252, 192, 0,
0, 192, 192, 3, 63, 3, 252, 255, 224, 126, 0, 31, 255, 255, 254, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 60, 0, 0, 3,
0, 192, 192, 3, 63, 3, 252, 255, 224, 126, 0, 31, 255, 255, 254, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 60, 0, 0, 3,
0, 192, 192, 3, 63, 3, 252, 255, 224, 126, 0, 31, 255, 255, 254, 0, 0, 0, 0, 0, 6, 0, 0, 0, 0, 60, 0, 0, 3,
0, 252, 252, 3, 195, 192, 60, 60, 127, 224, 0, 31, 255, 255, 158, 0, 0, 0, 0, 7, 128, 6, 0, 0, 0, 0, 0, 0, 255,
0, 252, 252, 3, 195, 192, 60, 60, 127, 224, 0, 31, 255, 255, 158, 0, 0, 0, 0, 7, 128, 6, 0, 0, 0, 0, 0, 0, 255,
0, 252, 255, 0, 195, 255, 0, 3, 254, 0, 0, 31, 255, 255, 158, 0, 0, 0, 0, 1, 249, 254, 0, 0, 0, 0, 0, 3, 252,
0, 252, 255, 0, 195, 255, 0, 3, 254, 0, 0, 31, 255, 255, 158, 0, 0, 0, 0, 1, 249, 254, 0, 0, 0, 0, 0, 3, 252,
0, 63, 63, 0, 252, 255, 192, 0, 126, 126, 0, 31, 255, 254, 31, 128, 0, 0, 0, 6, 127, 248, 0, 0, 0, 0, 0, 63, 3,
0, 63, 63, 0, 252, 255, 192, 0, 126, 126, 0, 31, 255, 254, 31, 128, 0, 0, 0, 6, 127, 248, 0, 0, 0, 0, 0, 63, 3,
0, 63, 63, 0, 252, 255, 192, 0, 126, 126, 0, 31, 255, 254, 31, 128, 0, 0, 0, 6, 127, 248, 0, 0, 0, 0, 0, 63, 3,
0, 63, 3, 192, 60, 60, 252, 0, 30, 96, 0, 31, 255, 254, 127, 128, 0, 0, 0, 6, 126, 0, 0, 0, 0, 0, 63, 60, 3,
0, 3, 0, 252, 3, 60, 0, 0, 30, 96, 0, 31, 255, 224, 96, 96, 0, 0, 0, 6, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 3, 0, 252, 3, 60, 0, 0, 30, 96, 0, 31, 255, 224, 96, 96, 0, 0, 0, 6, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 3, 192, 63, 3, 3, 63, 0, 30, 96, 0, 31, 255, 254, 127, 254, 0, 0, 0, 1, 134, 0, 0, 0, 0, 0, 0, 0, 0,
0, 3, 192, 63, 3, 3, 63, 0, 30, 96, 0, 31, 255, 254, 127, 254, 0, 0, 0, 1, 134, 0, 0, 0, 0, 0, 0, 0, 0,
0, 3, 192, 63, 3, 3, 63, 0, 30, 96, 0, 31, 255, 254, 127, 254, 0, 0, 0, 1, 134, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 192, 3, 3, 3, 0, 0, 30, 0, 0, 1, 255, 224, 0, 31, 128, 0, 0, 0, 6, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 192, 3, 3, 3, 0, 0, 30, 0, 0, 1, 255, 224, 0, 31, 128, 0, 0, 0, 6, 0, 0, 0, 0, 0, 0, 0, 0, 0

//Map Image

```
const byte Image_Array_2[3840] = {  
0, 0, 0, 31, 255, 255, 255, 255, 63, 239, 3, 255, 252, 0, 0, 0, 0, 1, 255, 255, 255, 239, 255, 255, 255, 255, 255, 0,  
0,  
0, 0, 0, 49, 132, 3, 240, 128, 63, 231, 2, 0, 120, 0, 0, 0, 0, 1, 130, 192, 1, 224, 64, 0, 16, 0, 32, 99, 0, 0,  
0, 0, 0, 63, 128, 1, 240, 128, 63, 54, 3, 0, 224, 0, 0, 0, 28, 31, 4, 192, 7, 196, 96, 96, 16, 0, 32, 198, 0, 0,  
0, 0, 0, 120, 72, 65, 224, 128, 31, 51, 3, 1, 192, 0, 0, 0, 62, 15, 4, 192, 4, 4, 96, 96, 16, 0, 65, 140, 0, 0,  
0, 0, 0, 127, 104, 0, 223, 224, 254, 113, 130, 3, 128, 0, 0, 0, 127, 207, 9, 64, 4, 4, 95, 224, 16, 0, 66, 28, 0, 0,  
0, 0, 0, 127, 184, 0, 99, 184, 253, 239, 131, 3, 1, 240, 0, 1, 255, 254, 9, 64, 4, 4, 79, 32, 88, 0, 108, 120, 0, 0,  
0, 0, 0, 253, 248, 32, 192, 39, 251, 207, 1, 130, 1, 240, 0, 1, 218, 124, 5, 32, 4, 4, 0, 49, 248, 0, 248, 248, 0, 0,  
0, 0, 1, 248, 236, 35, 240, 39, 255, 207, 1, 130, 1, 184, 0, 1, 255, 184, 5, 16, 8, 4, 96, 59, 62, 0, 133, 208, 0, 0,  
0, 0, 1, 248, 236, 35, 240, 39, 255, 207, 1, 130, 1, 184, 0, 1, 255, 184, 5, 16, 8, 4, 96, 59, 62, 0, 133, 208, 0, 0,  
0, 0, 7, 0, 112, 252, 128, 76, 0, 124, 0, 198, 0, 96, 0, 3, 255, 216, 7, 236, 12, 4, 31, 228, 16, 241, 151, 0, 0, 0,  
0, 0, 6, 0, 112, 39, 192, 72, 0, 120, 0, 108, 0, 0, 0, 7, 127, 248, 244, 196, 124, 6, 24, 68, 16, 23, 63, 0, 0, 0,  
0, 0, 6, 0, 112, 39, 192, 72, 0, 120, 0, 108, 0, 0, 0, 7, 127, 248, 244, 196, 124, 6, 24, 68, 16, 23, 63, 0, 0, 0,  
0, 0, 240, 0, 112, 0, 135, 236, 0, 0, 0, 56, 0, 0, 0, 27, 255, 251, 132, 228, 127, 247, 184, 35, 112, 9, 239, 128, 0, 0,  
0, 0, 240, 0, 48, 64, 130, 46, 0, 248, 0, 0, 0, 0, 0, 31, 223, 255, 68, 252, 167, 58, 136, 56, 200, 8, 39, 128, 0, 0,  
0, 0, 240, 0, 48, 64, 130, 46, 0, 248, 0, 0, 0, 0, 0, 31, 223, 255, 68, 252, 167, 58, 136, 56, 200, 8, 39, 128, 0, 0,  
0, 0, 192, 0, 48, 30, 128, 15, 1, 249, 128, 0, 0, 0, 112, 31, 255, 255, 62, 127, 227, 137, 252, 124, 191, 12, 39, 128, 0, 0,  
0, 0, 0, 0, 60, 192, 207, 205, 131, 15, 128, 0, 0, 0, 112, 63, 239, 251, 125, 113, 176, 63, 219, 178, 154, 167, 231, 128, 0,  
56,  
0, 0, 0, 0, 60, 192, 207, 205, 131, 15, 128, 0, 0, 0, 112, 63, 239, 251, 125, 113, 176, 63, 219, 178, 154, 167, 231, 128, 0,  
56,  
0, 128, 0, 0, 55, 192, 191, 156, 230, 13, 128, 0, 0, 0, 120, 31, 207, 255, 255, 51, 236, 35, 121, 59, 134, 110, 5, 128, 0, 16,  
1, 224, 0, 0, 50, 33, 135, 198, 56, 1, 192, 0, 0, 0, 248, 15, 255, 255, 247, 191, 30, 239, 248, 154, 112, 248, 4, 240, 0, 0,  
1, 224, 0, 0, 50, 33, 135, 198, 56, 1, 192, 0, 0, 0, 248, 15, 255, 255, 247, 191, 30, 239, 248, 154, 112, 248, 4, 240, 0, 0,  
17, 0, 0, 0, 58, 47, 133, 194, 104, 2, 192, 0, 0, 1, 248, 31, 223, 255, 250, 199, 227, 105, 121, 47, 231, 88, 4, 176, 0, 0,  
120, 0, 0, 0, 28, 240, 63, 128, 56, 4, 112, 0, 0, 3, 254, 63, 255, 255, 253, 255, 254, 119, 243, 182, 120, 112, 2, 192, 0, 0,  
120, 0, 0, 0, 28, 240, 63, 128, 56, 4, 112, 0, 0, 3, 254, 63, 255, 255, 253, 255, 254, 119, 243, 182, 120, 112, 2, 192, 0, 0,  
224, 0, 0, 0, 30, 223, 27, 129, 48, 62, 48, 0, 0, 3, 238, 127, 255, 255, 252, 191, 244, 252, 179, 246, 126, 94, 195, 128, 0,  
0,  
64, 0, 0, 0, 30, 207, 203, 141, 0, 119, 240, 0, 0, 3, 254, 255, 255, 255, 254, 92, 46, 76, 251, 252, 226, 79, 193, 128, 0, 0,
```

0, 0, 0, 0, 31, 221, 126, 76, 176, 128, 240, 0, 0, 1, 127, 255, 255, 255, 249, 252, 59, 246, 190, 239, 223, 235, 225, 128, 0,
0,
0, 0, 0, 0, 62, 239, 251, 221, 253, 63, 224, 0, 0, 0, 127, 255, 255, 255, 253, 255, 125, 247, 191, 110, 79, 59, 225, 128, 0,
0,
0, 0, 0, 0, 55, 169, 83, 127, 39, 127, 240, 0, 0, 0, 127, 255, 255, 255, 247, 45, 215, 218, 221, 46, 101, 63, 225, 128, 0, 0,
0, 0, 0, 0, 61, 255, 127, 255, 36, 254, 184, 0, 0, 0, 127, 255, 255, 255, 254, 105, 16, 139, 117, 190, 79, 246, 241, 128, 0,
0,
0, 0, 0, 0, 125, 155, 223, 255, 255, 249, 216, 0, 0, 0, 63, 255, 255, 255, 255, 251, 127, 173, 127, 238, 76, 103, 241, 0, 0,
0,
0, 0, 0, 0, 119, 240, 202, 223, 223, 251, 248, 0, 0, 0, 63, 255, 255, 255, 223, 179, 255, 255, 223, 23, 254, 181, 241, 128, 0,
0,
0, 0, 0, 0, 116, 112, 107, 191, 222, 159, 120, 0, 0, 0, 31, 255, 255, 255, 213, 239, 191, 243, 219, 253, 235, 153, 241, 0, 0,
0,
0, 0, 0, 0, 108, 122, 127, 255, 255, 255, 16, 0, 0, 0, 31, 255, 255, 255, 255, 127, 241, 255, 60, 253, 113, 207, 251, 0, 0, 0,
0, 0, 0, 0, 124, 215, 212, 127, 255, 254, 0, 0, 0, 7, 159, 255, 255, 255, 254, 254, 154, 49, 62, 250, 115, 47, 251, 0, 0, 0,
0, 0, 0, 0, 127, 175, 101, 127, 255, 184, 0, 0, 0, 7, 255, 255, 255, 209, 254, 151, 253, 57, 251, 143, 191, 59, 155, 0, 0, 0,
0, 0, 0, 0, 62, 173, 119, 119, 255, 0, 0, 0, 0, 7, 255, 255, 255, 205, 250, 193, 223, 127, 243, 8, 207, 239, 158, 0, 0, 0,
0, 0, 0, 0, 62, 249, 93, 245, 255, 0, 0, 0, 0, 15, 255, 255, 255, 223, 251, 119, 240, 103, 251, 232, 158, 255, 158, 0, 0, 0,
0, 0, 0, 0, 59, 55, 243, 223, 255, 0, 0, 1, 0, 14, 255, 255, 255, 247, 255, 63, 181, 65, 234, 255, 245, 127, 190, 0, 0, 0,
0, 0, 0, 0, 59, 55, 243, 223, 255, 0, 0, 1, 0, 14, 255, 255, 255, 247, 255, 63, 181, 65, 234, 255, 245, 127, 190, 0, 0, 0,
0, 0, 0, 0, 61, 251, 255, 251, 248, 0, 0, 1, 128, 31, 252, 55, 255, 255, 255, 177, 254, 122, 127, 95, 219, 252, 24, 0, 0, 0,
0, 0, 0, 0, 31, 74, 127, 255, 240, 0, 0, 0, 0, 31, 255, 59, 255, 183, 255, 176, 158, 79, 253, 243, 251, 252, 28, 0, 0, 0,
0, 0, 0, 0, 31, 74, 127, 255, 240, 0, 0, 0, 0, 31, 255, 59, 255, 183, 255, 176, 158, 79, 253, 243, 251, 252, 28, 0, 0, 0,
0, 0, 0, 0, 15, 87, 247, 223, 240, 0, 0, 0, 0, 15, 255, 240, 239, 255, 185, 255, 47, 172, 135, 199, 255, 188, 28, 0, 0, 0,
0, 0, 0, 0, 7, 218, 125, 255, 224, 0, 0, 0, 0, 7, 255, 251, 231, 255, 215, 251, 61, 168, 135, 111, 127, 254, 60, 0, 0, 0,
0, 0, 0, 0, 7, 218, 125, 255, 224, 0, 0, 0, 0, 7, 255, 251, 231, 255, 215, 251, 61, 168, 135, 111, 127, 254, 60, 0, 0, 0,
0, 0, 0, 0, 7, 254, 238, 239, 192, 0, 0, 0, 0, 3, 255, 251, 199, 79, 254, 189, 97, 252, 143, 255, 255, 158, 252, 0, 0, 0,
0, 0, 0, 0, 7, 141, 229, 255, 0, 24, 0, 0, 0, 7, 223, 127, 153, 195, 223, 231, 99, 255, 254, 63, 255, 157, 248, 0, 0, 0,
0, 0, 0, 0, 7, 141, 229, 255, 0, 24, 0, 0, 0, 7, 223, 127, 153, 195, 223, 231, 99, 255, 254, 63, 255, 157, 248, 0, 0, 0,
0, 0, 0, 0, 7, 255, 214, 254, 0, 24, 0, 0, 0, 15, 248, 31, 62, 7, 223, 188, 118, 228, 203, 207, 255, 219, 240, 0, 0, 0,
0, 0, 0, 0, 1, 238, 255, 143, 0, 0, 0, 0, 13, 128, 55, 255, 255, 167, 239, 136, 220, 207, 127, 255, 227, 192, 0, 0, 48,
0, 0, 0, 0, 1, 238, 255, 143, 0, 0, 0, 0, 13, 128, 55, 255, 255, 167, 239, 136, 220, 207, 127, 255, 227, 192, 0, 0, 48,
0, 0, 0, 0, 1, 237, 240, 15, 64, 0, 0, 0, 0, 93, 4, 44, 255, 255, 159, 186, 153, 255, 231, 251, 255, 225, 128, 0, 128, 16,
24, 0, 0, 0, 1, 247, 112, 7, 224, 0, 0, 0, 0, 254, 4, 47, 255, 254, 79, 203, 255, 31, 220, 47, 255, 224, 129, 192, 192, 0,
24, 0, 0, 0, 0, 251, 48, 7, 112, 0, 0, 0, 0, 250, 7, 240, 255, 254, 99, 250, 79, 62, 221, 175, 255, 227, 0, 192, 192, 0,
0, 0, 0, 0, 0, 29, 112, 2, 240, 0, 0, 0, 0, 55, 4, 232, 131, 254, 63, 190, 110, 253, 255, 255, 255, 247, 1, 128, 0, 0,
0, 0, 128, 0, 0, 12, 224, 64, 248, 0, 0, 0, 0, 121, 136, 24, 2, 191, 32, 255, 246, 222, 255, 247, 255, 254, 1, 192, 0, 0,
0, 0, 240, 0, 0, 13, 123, 255, 144, 0, 0, 0, 0, 209, 40, 20, 126, 191, 209, 235, 15, 77, 14, 127, 255, 176, 0, 0, 0, 0,
0, 0, 240, 0, 0, 13, 123, 255, 144, 0, 0, 0, 0, 209, 40, 20, 126, 191, 209, 235, 15, 77, 14, 127, 255, 176, 0, 0, 0, 0,
0, 0, 48, 0, 0, 7, 223, 95, 224, 0, 0, 0, 0, 209, 48, 38, 123, 251, 153, 27, 7, 255, 254, 255, 255, 0, 0, 0, 0, 0,
0, 0, 16, 0, 0, 1, 219, 192, 254, 64, 0, 0, 0, 225, 12, 65, 98, 47, 238, 18, 3, 160, 99, 255, 248, 16, 0, 48, 0, 0,
0, 0, 16, 0, 0, 1, 219, 192, 254, 64, 0, 0, 0, 225, 12, 65, 98, 47, 238, 18, 3, 160, 99, 255, 248, 16, 0, 48, 0, 0,
0, 0, 0, 0, 0, 254, 249, 255, 224, 0, 0, 0, 193, 6, 193, 62, 63, 104, 118, 0, 254, 227, 255, 184, 56, 0, 48, 0, 0,
0, 0, 0, 0, 0, 0, 54, 249, 255, 224, 0, 0, 0, 224, 3, 1, 228, 55, 168, 236, 0, 166, 193, 247, 240, 56, 0, 48, 0, 192,
0, 0, 0, 0, 0, 7, 248, 8, 96, 0, 0, 12, 192, 191, 0, 4, 47, 191, 56, 0, 231, 129, 253, 224, 120, 0, 48, 14, 240,
0, 0, 0, 0, 0, 1, 216, 56, 112, 0, 0, 12, 251, 179, 254, 12, 63, 214, 112, 0, 255, 1, 255, 224, 120, 0, 48, 4, 248,
0, 0, 0, 0, 0, 0, 88, 252, 112, 0, 0, 0, 191, 254, 131, 24, 54, 247, 192, 0, 110, 0, 191, 240, 56, 0, 112, 0, 240,
0, 16, 0, 0, 0, 0, 126, 255, 240, 0, 0, 0, 239, 221, 255, 255, 127, 255, 128, 0, 102, 3, 63, 240, 60, 0, 32, 0, 240,
0, 48, 0, 0, 0, 0, 63, 183, 240, 0, 0, 0, 255, 227, 127, 223, 255, 254, 28, 0, 54, 3, 63, 240, 62, 0, 0, 0, 88,
0, 0, 0, 0, 0, 0, 31, 249, 112, 0, 0, 0, 255, 119, 97, 248, 61, 123, 220, 0, 62, 2, 31, 240, 55, 0, 0, 0, 24,
0, 0, 0, 0, 0, 0, 31, 179, 176, 0, 0, 0, 111, 253, 247, 216, 45, 127, 204, 0, 62, 2, 27, 240, 127, 0, 0, 0, 62,
0, 0, 0, 0, 0, 0, 3, 255, 28, 0, 0, 0, 61, 236, 207, 184, 75, 189, 192, 0, 62, 2, 57, 224, 111, 1, 0, 7, 54,
0, 0, 0, 0, 0, 0, 7, 245, 191, 128, 0, 0, 55, 46, 220, 231, 233, 254, 192, 0, 31, 0, 57, 192, 239, 3, 0, 7, 16,
0, 0, 0, 0, 0, 0, 15, 101, 155, 192, 0, 0, 31, 175, 119, 162, 47, 182, 128, 0, 31, 0, 28, 128, 207, 129, 31, 192, 0,
0, 0, 128, 0, 0, 0, 31, 31, 155, 192, 0, 0, 13, 127, 249, 161, 36, 253, 128, 0, 3, 0, 78, 1, 239, 128, 127, 192, 0,
0, 0, 192, 0, 0, 0, 29, 143, 31, 224, 0, 0, 7, 255, 251, 191, 191, 253, 128, 0, 3, 0, 239, 1, 227, 0, 69, 16, 0,
0, 0, 64, 0, 0, 0, 0, 30, 227, 3, 248, 0, 0, 3, 240, 254, 99, 237, 247, 0, 0, 0, 0, 255, 7, 224, 128, 0, 16, 32,
0, 0, 64, 0, 0, 0, 0, 30, 227, 3, 248, 0, 0, 3, 240, 254, 99, 237, 247, 0, 0, 0, 0, 255, 7, 224, 128, 0, 16, 32,
0, 0, 48, 0, 0, 0, 0, 28, 126, 245, 71, 128, 0, 0, 0, 63, 255, 238, 62, 0, 0, 0, 63, 31, 255, 200, 0, 0, 32,
0, 0, 48, 0, 0, 0, 0, 63, 243, 229, 65, 224, 1, 0, 0, 59, 177, 63, 60, 0, 0, 0, 31, 191, 127, 223, 0, 0, 0,
0, 0, 48, 0, 0, 0, 0, 63, 243, 229, 65, 224, 1, 0, 0, 59, 177, 63, 60, 0, 0, 0, 31, 191, 127, 223, 0, 0, 0,
0, 0, 0, 0, 0, 0, 57, 160, 138, 40, 88, 0, 0, 0, 54, 65, 46, 240, 0, 0, 0, 15, 154, 255, 255, 249, 112, 0,
0, 0, 0, 0, 0, 0, 13, 89, 10, 56, 88, 0, 0, 0, 58, 225, 123, 224, 0, 0, 0, 15, 155, 253, 255, 252, 48, 0,
0, 0, 0, 0, 0, 0, 13, 89, 10, 56, 88, 0, 0, 0, 58, 225, 123, 224, 0, 0, 0, 15, 155, 253, 255, 252, 48, 0,
128, 0, 0, 0, 0, 0, 14, 166, 9, 47, 240, 0, 0, 0, 25, 187, 243, 224, 0, 0, 0, 6, 159, 253, 239, 254, 120, 0,
192, 48, 0, 0, 0, 0, 2, 255, 143, 196, 96, 0, 0, 0, 15, 148, 179, 224, 12, 0, 0, 0, 3, 254, 56, 0, 247, 204, 0,
192, 48, 0, 0, 0, 0, 2, 255, 143, 196, 96, 0, 0, 0, 15, 148, 179, 224, 12, 0, 0, 0, 3, 254, 56, 0, 247, 204, 0,
224, 0, 0, 0, 0, 0, 2, 221, 211, 134, 192, 0, 0, 0, 7, 95, 182, 224, 4, 0, 0, 0, 1, 255, 148, 0, 207, 142, 0,
48, 0, 0, 0, 0, 0, 1, 246, 48, 130, 128, 0, 0, 0, 6, 137, 88, 64, 0, 0, 96, 0, 0, 15, 255, 136, 51, 195, 0,
48, 0, 0, 0, 0, 0, 1, 246, 48, 130, 128, 0, 0, 0, 6, 137, 88, 64, 0, 0, 96, 0, 0, 15, 255, 136, 51, 195, 0,

```

48, 0, 0, 0, 0, 0, 0, 0, 247, 253, 60, 128, 0, 0, 0, 7, 253, 77, 195, 0, 0, 32, 0, 0, 0, 54, 28, 17, 193, 128,
16, 0, 0, 0, 0, 0, 0, 0, 53, 251, 71, 128, 0, 0, 0, 6, 199, 71, 227, 0, 0, 0, 0, 0, 0, 0, 31, 152, 1, 192,
0, 0, 0, 0, 0, 0, 0, 0, 61, 113, 135, 128, 0, 0, 128, 6, 132, 238, 231, 128, 0, 0, 0, 0, 0, 0, 189, 152, 0, 176,
0, 0, 0, 0, 0, 0, 0, 0, 61, 153, 255, 0, 0, 0, 192, 7, 156, 230, 111, 128, 0, 0, 0, 0, 0, 1, 227, 188, 0, 16,
0, 0, 0, 0, 0, 0, 0, 0, 57, 207, 28, 0, 0, 0, 128, 15, 252, 190, 253, 128, 0, 0, 0, 0, 0, 7, 106, 252, 0, 0,
16, 0, 0, 0, 0, 0, 0, 32, 55, 238, 48, 0, 0, 0, 13, 12, 255, 249, 0, 0, 0, 0, 0, 14, 55, 124, 0, 112,
24, 0, 0, 0, 0, 0, 0, 48, 61, 26, 32, 0, 0, 0, 15, 230, 182, 243, 0, 0, 0, 0, 0, 31, 225, 220, 0, 112,
16, 0, 0, 0, 0, 0, 0, 32, 57, 154, 96, 0, 0, 0, 15, 127, 223, 247, 0, 0, 0, 0, 0, 123, 33, 254, 1, 112,
24, 0, 0, 0, 0, 0, 0, 40, 206, 224, 0, 0, 0, 12, 91, 75, 187, 0, 0, 0, 0, 0, 1, 254, 97, 111, 1, 176,
12, 0, 0, 0, 0, 0, 0, 58, 124, 192, 0, 0, 0, 13, 65, 46, 50, 6, 0, 0, 0, 0, 3, 249, 209, 223, 129, 248,
12, 0, 0, 0, 0, 0, 0, 58, 124, 192, 0, 0, 0, 5, 143, 52, 118, 14, 0, 0, 0, 0, 6, 76, 30, 67, 128, 248,
0, 0, 0, 0, 0, 0, 0, 54, 99, 128, 0, 0, 0, 6, 249, 148, 38, 24, 0, 0, 0, 0, 6, 132, 24, 34, 192, 0,
0, 0, 0, 0, 0, 0, 12, 54, 159, 0, 0, 0, 7, 147, 214, 36, 0, 0, 0, 0, 6, 134, 16, 95, 192, 0,
0, 224, 0, 0, 0, 0, 12, 50, 255, 0, 0, 0, 2, 156, 86, 60, 0, 0, 0, 0, 7, 2, 16, 77, 192, 0,
0, 192, 0, 0, 0, 0, 0, 115, 54, 0, 0, 0, 2, 145, 156, 60, 0, 0, 0, 0, 7, 2, 16, 69, 192, 0,
0, 0, 0, 0, 0, 0, 0, 99, 44, 0, 0, 0, 2, 159, 24, 0, 0, 0, 0, 6, 2, 112, 134, 192, 128,
48, 0, 3, 0, 0, 0, 0, 53, 188, 0, 0, 0, 3, 91, 248, 0, 0, 0, 0, 0, 7, 135, 208, 196, 192, 192,
112, 0, 3, 0, 0, 0, 0, 61, 248, 0, 0, 0, 3, 249, 112, 0, 0, 0, 0, 0, 6, 220, 48, 159, 128, 192,
112, 0, 3, 0, 0, 0, 0, 61, 248, 0, 0, 0, 3, 249, 112, 0, 0, 0, 0, 0, 6, 220, 48, 159, 128, 192,
48, 0, 0, 0, 0, 0, 0, 52, 192, 0, 0, 0, 1, 193, 240, 0, 0, 0, 0, 0, 6, 14, 62, 213, 0, 96,
0, 0, 0, 0, 0, 0, 0, 54, 192, 0, 0, 0, 0, 195, 96, 0, 0, 0, 0, 7, 248, 31, 127, 0, 112,
0, 0, 0, 0, 0, 0, 0, 54, 192, 0, 0, 0, 0, 195, 96, 0, 0, 0, 0, 7, 248, 31, 127, 0, 112,
0, 0, 0, 0, 0, 0, 0, 60, 192, 0, 0, 0, 0, 187, 192, 0, 0, 0, 0, 1, 0, 12, 220, 0, 124,
0, 0, 0, 0, 0, 0, 0, 61, 128, 0, 0, 0, 0, 215, 128, 0, 0, 0, 0, 0, 0, 4, 252, 0, 204,
0, 0, 0, 0, 0, 0, 0, 61, 128, 0, 0, 0, 0, 215, 128, 0, 0, 0, 0, 0, 0, 4, 252, 0, 204,
0, 0, 0, 0, 0, 0, 0, 61, 0, 0, 0, 0, 0, 254, 0, 0, 0, 0, 0, 0, 0, 7, 120, 0, 248,
0, 0, 0, 0, 0, 0, 0, 45, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 224, 3, 224,
0, 0, 0, 0, 0, 0, 0, 45, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 3, 224, 3, 224,
0, 0, 0, 0, 0, 0, 0, 61, 129, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 240, 6, 224,
0, 0, 0, 0, 0, 0, 0, 29, 1, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 240, 31, 128,
0, 0, 0, 0, 0, 0, 0, 29, 1, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 240, 31, 128,
0, 0, 0, 0, 0, 0, 0, 29, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 224, 63, 0,
0, 0, 0, 0, 0, 0, 0, 29, 128, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 54, 0,
0, 0, 0, 0, 0, 0, 0, 15, 192, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 108, 0,
0, 0, 0, 0, 0, 0, 0, 7, 224, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 124, 0,
0, 0, 0, 0, 0, 0, 0, 3, 224, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 56, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0
};

```

```

void setup()
{
  Serial.begin(4800);
  //Serial.begin(9600);
  //Serial.begin(19200);
  //Serial.begin(38400);
  delay(1000);
  //Enable_EE_Frame();
  BackLight_Test();
  Show_Font();
  Show_7_Segments();
  Geom_Test();
  Geom_Test_1();
  TextString_Test();
  Graphic_Text_Panel_Test();
  Load_Image_Test();
  Test_End();
}

void loop(){
}

//===== Test Text Function =====
void xdelay() {
  delay(500);
}

void xdelay1() {
  delay(500);
}

void Enable_EE_Frame() {

```

```

Serial.write(176);          // Disable EEPROM Frame, for fast (Default)
//Serial.write(177);       // Enable EEPROM Frame, it will be slow if dot touch this area
delay(500);                // The delay time needs to change!!!
}
void parameter_2() {
  Serial.write(cmd);       //Command
  Serial.write(x);         // x position
  Serial.write(y);         // y position
}

void parameter_4() {
  Serial.write(cmd);       //Command
  Serial.write(x);         // x position
  Serial.write(y);         // y position
  Serial.write(c1);
  Serial.write(c2);
}

void parameter_5() {
  Serial.write(cmd);       //Command
  Serial.write(x);         // x position
  Serial.write(y);         // y position
  Serial.write(c1);
  Serial.write(c2);
  Serial.write(c3);
}

void BackLight_Test() {
  Serial.write(70);        //Clean Screen
  delay(1000);
  cmd = 98; x = 30; y = 25; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("BackLight Test."); // string
  Serial.write(10);        // end char
  xdelay();
  cmd = 98; x = 60; y = 60; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("On / Off"); // string
  Serial.write(10);        // end char
  xdelay();
  for (int i=0; i <= 5; i++){
    Serial.write(180);      //Command
    delay(500);
    Serial.write(181);      //Command
    delay(500);
  }
  Serial.write(70);        //Clean Screen
  delay(1000);
  cmd = 98; x = 30; y = 25; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("BackLight Test."); // string
  Serial.write(10);        // end char
  xdelay();
  cmd = 98; x = 10; y = 60; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("Dark <--> Bright"); // string
  Serial.write(10);        // end char
  xdelay();
  for (int i=0; i <= 15; i++){
    Serial.write(182);      //Command $A1
    delay(500);
  }
  for (int i=0; i <= 15; i++){
    Serial.write(183);      //Command $A1
    delay(500);
  }
  for (int i=0; i <= 6; i++){
    Serial.write(182);      //Command $A1
    delay(500);
  }
}
}

```

```

void Show_7_Segments() {
  Serial.write(70);          //Clean Screen
  delay(1000);
  cmd = 98; x = 20; y = 20; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("Show 7 Segments."); // string
  Serial.write(10);        // end char
  xdelay();
  for (int i=32; i <= 45; i++){
    cmd = 128; x = 45; y = 55; c1 = i; c2 = 10; //font size = 34x64
    parameter_4();
    xdelay(1);
    cmd = 129; x = 90; y = 68; c1 = i; c2 = 10; //font size = 28x48
    parameter_4();
    xdelay(1);
    cmd = 130; x = 130; y = 80; c1 = i; c2 = 10; //font size = 18x32
    parameter_4();
    xdelay(1);
    cmd = 131; x = 167; y = 85; c1 = i; c2 = 10; //font size = 16x24
    parameter_4();
    xdelay(1);
  }
}

void Show_Font() {
  Serial.write(70);          //Clean Screen
  delay(1000);
  cmd = 98; x = 50; y = 25; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("Show font."); // string
  Serial.write(10);        // end char
  xdelay();
  for (int i=32; i <= 127; i++){
    cmd = 112; x = 25; y = 64; c1 = i; c2 = 10; //font size = 28x48
    parameter_4();
    xdelay(1);
    cmd = 113; x = 70; y = 80; c1 = i; c2 = 10; //font size = 19x32
    parameter_4();
    xdelay(1);
    cmd = 114; x = 110; y = 88; c1 = i; c2 = 10; //font size = 13x24
    parameter_4();
    xdelay(1);
    cmd = 115; x = 150; y = 96; c1 = i; c2 = 10; //font size = 12x16
    parameter_4();
    xdelay(1);
    cmd = 116; x = 190; y = 104; c1 = i; c2 = 10; //font size = 8x8
    parameter_4();
    xdelay();
  }
}

void Geom_Test() {
  Serial.write(70);          //Clean Screen
  delay(1000);
  cmd = 98; x = 30; y = 18; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("Geometry Test."); // string
  Serial.write(10);        // end char
  xdelay(1);
  cmd = 99; x = 25; y = 55; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("1. Circle. 2 Ellips."); // string
  Serial.write(10);        // end char
  xdelay(1);
  cmd = 99; x = 15; y = 75; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("3. Square. 4. Rectangle"); // string
  Serial.write(10);        // end char
  xdelay(1);
  cmd = 99; x = 1; y = 95; c1 = 8; c2 = 2;

```

```

parameter_4();
Serial.write("5.H-Line. 6.V-Line. 7.Line"); // string
Serial.write(10); // end char
delay(2000);
}
void Geom_Test_1() { // Geom Fill for Frame Ram
Serial.write(70); //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 151; x = 120; y = 64; c1 = 10*i; c2 = 10; // Circle for Frame Ram
parameter_4();
xdelay1();
}
delay(1000);
Serial.write(70); //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 152; x = 120; y = 64; c1 = 10*i; c2 = 10; // Circle_Fill for Frame Ram
parameter_4();
xdelay1();
}
delay(1000);
Serial.write(70); //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 153; x = 120; y = 64; c1 = 20*i; c2 = 10*i; c3 = 10; // Ellipse for Frame Ram
parameter_5();
xdelay1();
}
delay(1000);
Serial.write(70); //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 154; x = 120; y = 64; c1 = 20*i; c2 = 10*i; c3 = 10; // Ellipse_Fill for Frame Ram
parameter_5();
xdelay1();
}
delay(1000);
Serial.write(70); //Clean Screen
delay(1000);
for (int i=0; i <= 12; i++){
cmd = 144; x = 0; y = 239; c1 = 10*i; c2 = 10; // H-Line for Frame Ram
parameter_4();
xdelay1();
}
for (int i=0; i <= 12; i++){
cmd = 145; x = 20*i; y = 0; c1 = 127; c2 = 10; // V-Line for Frame Ram
parameter_4();
xdelay1();
}
cmd = 146; x = 0; y = 0; c1 = 239; c2 = 127; c3 = 10; // Line for Frame Ram
parameter_5();
xdelay1();
cmd = 146; x = 0; y = 127; c1 = 239; c2 = 0; c3 = 10; // Line for Frame Ram
parameter_5();
xdelay1();
delay(1000);
Serial.write(70); //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 147; x = 120 - 20 * i; y = 64 - 10 * i; c1 = 40 * i; c2 = 20 * i; c3 = 10; // Rectangle for Frame Ram
parameter_5();
xdelay1();
}
delay(1000);
Serial.write(70); //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 148; x = 120 - 20 * i; y = 64 - 10 * i; c1 = 40 * i; c2 = 20 * i; c3 = 10; // Rectangle_Fill for Frame Ram
parameter_5();
}

```

```

xdelay1();
}
delay(1000);
Serial.write(70);          //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 149; x = 120 - 10 * i; y = 64 - 10 * i; c1 = 20 * i; c2 = 10; // Square for Frame Ram
parameter_4();
xdelay1();
}
delay(1000);
Serial.write(70);          //Clean Screen
delay(1000);
for (int i=1; i <= 6; i++){
cmd = 150; x = 120 - 10 * i; y = 64 - 10 * i; c1 = 20 * i; c2 = 10; // Square_Fill for Frame Ram
parameter_4();
xdelay1();
}
}

void TextString_Test() {          // Text for Frame Ram
Serial.write(70);          //Clean Screen
delay(1000);
cmd = 96; x = 45; y = 10; c1 = 10; c2 = 2; // 28x48 Text for Frame Ram
parameter_4();
Serial.write("28x48");      // string
Serial.write(10);          // end char
delay(1000);
cmd = 96; x = 45; y = 64; c1 = 10; c2 = 4; // 28x48 Text for Frame Ram
parameter_4();
Serial.write("Hello!");    // string
Serial.write(10);          // end char
delay(2000);
Serial.write(70);          //Clean Screen
delay(1000);
cmd = 97; x = 65; y = 30; c1 = 10; c2 = 2; // 19x32 Text for Frame Ram
parameter_4();
Serial.write("19x32");     // string
Serial.write(10);          // end char
delay(1000);
cmd = 97; x = 15; y = 64; c1 = 10; c2 = 3; // 19x32 Text for Frame Ram
parameter_4();
Serial.write("Hello, World!"); // string
Serial.write(10);          // end char
delay(2000);
Serial.write(70);          //Clean Screen
delay(1000);
cmd = 98; x = 80; y = 30; c1 = 10; c2 = 2; // 13x24 Text for Frame Ram
parameter_4();
Serial.write("13x24");     // string
Serial.write(10);          // end char
delay(1000);
cmd = 98; x = 40; y = 64; c1 = 10; c2 = 3; // 13x24 Text for Frame Ram
parameter_4();
Serial.write("Hello, World!"); // string
Serial.write(10);          // end char
delay(2000);
Serial.write(70);          //Clean Screen
delay(1000);
cmd = 99; x = 90; y = 30; c1 = 10; c2 = 2; // 12x16 Text for Frame Ram
parameter_4();
Serial.write("12x16");     // string
Serial.write(10);          // end char
delay(1000);
cmd = 99; x = 55; y = 65; c1 = 10; c2 = 3; // 12x16 Text for Frame Ram
parameter_4();
Serial.write("Hello, World!"); // string
Serial.write(10);          // end char
delay(2000);
Serial.write(70);          //Clean Screen

```



```

delay(1000);
cmd = 100; x = 90; y = 30; c1 = 10; c2 = 2; // 8x8 Text for Frame Ram
parameter_4();
Serial.write("8 x 8"); // string
Serial.write(10); // end char
delay(1000);
cmd = 100; x = 65; y = 65; c1 = 10; c2 = 2; // 8x8 Text for Frame Ram
parameter_4();
Serial.write("Hello, World!"); // string
Serial.write(10); // end char
delay(1000);
}

void Graphic_Text_Panel_Test() {
  Serial.write(70); //Clean Screen
  delay(1000);
  cmd = 82; x = 2; y = 7;
  parameter_2();
  Serial.write("Text panel with mode AND."); // Text on Text panel
  Serial.write(10); // end char
  xdelay();
  cmd = 83; x = 2; y = 8;
  parameter_2();
  Serial.write("Text panel with mode TEXT."); // Text on Text panel
  Serial.write(10); // end char
  xdelay();
  cmd = 80; x = 2; y = 9;
  parameter_2();
  Serial.write("Text panel with mode OR."); // Text on Text panel
  Serial.write(10); // end char
  xdelay();
  cmd = 81; x = 2; y = 10;
  parameter_2();
  Serial.write("Text panel with mode XOR."); // Text on Text panel
  Serial.write(10); // end char
  xdelay();
  cmd = 98; x = 45; y = 25; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("Graphic panel."); // Text on graphic panel
  Serial.write(10); // end char
  xdelay();
  cmd = 148; x = 15; y = 50; c1 = 210; c2 = 45; c3 = 10; // Rectangle_Fill for Frame Ram
  parameter_5();
  xdelay();
  for (int i=1; i <= 6; i++){
    Serial.write(77); // Graphic Panel Off
    delay(1000);
    Serial.write(76); // Graphic Panel On
    delay(1000);
    Serial.write(79); // Text Panel Off
    delay(1000);
    Serial.write(78); // Text Panel On
    delay(1000);
  }
  Serial.write(190); // Clean Text Panel
}

void Load_Image_Test() {
  Serial.write(178); //Enable JP Logo
  delay(1000);
  Serial.write(70); //Clean Screen
  delay(1000);
  cmd = 98; x = 10; y = 40; c1 = 8; c2 = 2;
  parameter_4();
  Serial.write("Image is loading..."); // Text on graphic panel
  Serial.write(10); // end char
  xdelay();
  Serial.write(72); //72 to Screen directly.
  for ( int i=0; i < 3840; i++){
    //Serial.write(Image_Array_0[i]); // Conutryside

```

```
//Serial.write(Image_Array_1[i]); // Cartoon
Serial.write(Image_Array_2[i]); // Map
}
delay(1000);
Serial.write(74); // Copy screen Image to Frame Ram
delay(1000);
Serial.write(189); //Clean Screen
delay(1000);
Serial.write(69); // Load Image to Screen from Ram
}

void Test_End() {
  Serial.write(70); //Clean Screen
  delay(1000);
  cmd = 98; x = 10; y = 30; c1 = 10; c2 = 2; // 13x24 Text for Frame Ram
  parameter_4();
  Serial.write("Function test end."); // string
  Serial.write(10); // end char
  delay(1000);
  cmd = 98; x = 5; y = 70; c1 = 10; c2 = 1; // 13x24 Text for Frame Ram
  parameter_4();
  Serial.write("Thank you for watch."); // string
  Serial.write(10); // end char
  delay(1000);
}
```

Jianping Electronics
jp@jianpingusa.com
www.jianpingusa.com