KENT DISPLAYS INCORPORATED — 320x240x5.6

320x240 Cholesteric Graphic and Character Display/Controller Module, with Standard Bus Interface

Typical Applications

- o Character or Graphic Display Applications
- o Battery Powered, Portable Displays
- o General Purpose Indoor or Outdoor Signage
- o Point of Sale Displays
- o Instrumentation or Avionic Displays
- o Outdoor Instrumentation Display Applications

Product Description

The Kent Displays 320x240 (Quarter VGA, or QVGA) Displays are modular units designed for general-purpose graphic and character display applications. With the onboard controller, the module does not require an external controller module. Intended uses include instrumentation, point of sale and other general-purpose indoor/outdoor hand held display applications.

As with all Kent Display cholesteric (ChLCD) products, the 320x240 modules contain the same optical and power saving advantages over traditional LCD products. After an image is generated on the module, it will remain indefinitely after power is removed, or until a new image is generated.

320x240 Green/Black Display and Controller Module

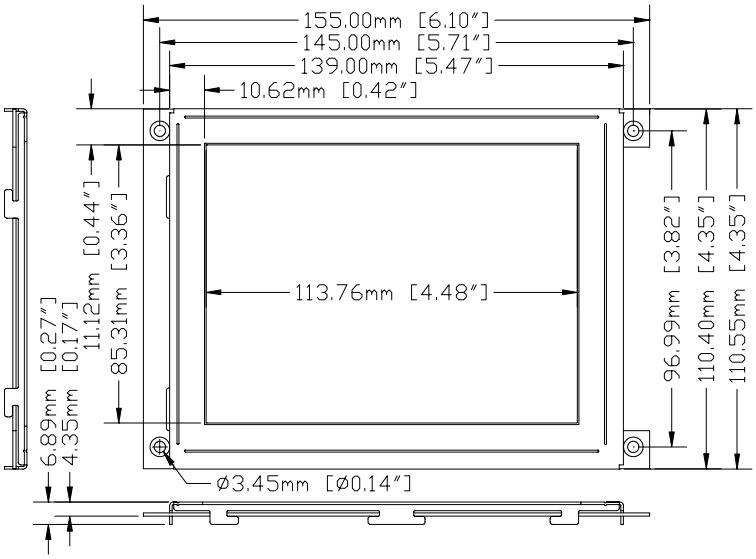
Standard Display Only Features:

- o 320 by 240 Pixels, 5.6" Diagonal, 72 DPI image.
- o Battery powered capability (3-9 VDC).
- o Graphic or character generation capability.
- o Full or Partial Screen Update Capability
- o **Dynamic Update Capability, including:**
  Wipe, Scroll, Rotate, Open, Close, Flashing, Swell and Fade.
- o Indefinite image memory capability.
- o 360 degree unlimited viewing angle.
- o Superior brightness & optical characteristics

Display w/Optional Controller Features:

- o Standard Bus Interface (BUSXV Versions).
- o Up to 31 controllers can be controlled from a host computer (w/Serial Version Controller).
- o Battery powered capability (3-9 VDC Input).
- o Automatic Wake/Idle & Sleep mode capability.
- o Local diagnostics and control.
- o *Stores up to 36 unique messages per display.
- o 160, 512, 960 or 1590 Characters/Display w/ ISO/IEC 8859-1 Compliant Text Generator.
- o 4 unique fonts, and font controls w/ Text Generator.
- o Automatic message generation capability.
- o Bus 3 Volt, Bus 5 Volt, RS-232, TTL and/or RF "wireless" interfaces.
- o 320x240Soft, Windows-Based Software supplied with product, provides the following features:
  Full Text and Graphical Editor.
  Message/ Image Storage & Retrieval.

* Based on serial controller module, storing full screen, and small font text messages.



320x240 Display and Controller Mechanical Dimensions

Kent Displays Inc
343 Portage Blvd.
Kent, OH 44240, USA

Tel (330) 673 8784
Fax (330) 673 4408
sales@kentdisplays.com
website: www.kentdisplays.com

Product Ordering Information:

See detailed Ordering Information & list on next sheet.

320x240 Product Ordering Information:

(Sample Module Part Number) **320x240x5.6 BUS3V YEL**

Catalog Number
(KDI P/N 15297)

Configuration:
- 232 = with RS-232 Port Serial Controller (Standard)
- TTL = with TTL Port Serial Controller
- BUS3V = with Bus Interface Controller for 3 volt logic systems
- BUS5V = with Bus interface Controller for 5 volt logic systems.
- SLV = without Display Controller (Slave interface)

ChLCD/Background Color Options:
- YEL = Yellow/Black
- YLG = Yellow-Green/Black
- WHT = White/Blue
- GRN = Green/Black

(Sample 320x240 Development Kit Part Number) **90024 232 YGN**

ChLCD/Background Color Options:
- YEL = Yellow/Black
- YLG = Yellow-Green/Black
- WHT = White/Blue
- GRN = Green/Black

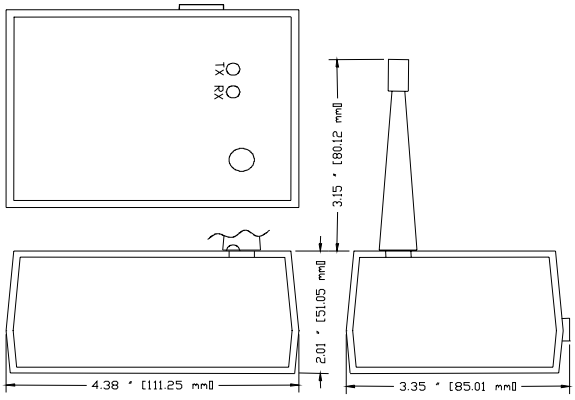
Configuration:
- 232 = with RS-232 Port Serial Controller (Standard)
- BUS5V = with 5 volt Bus Interface (Standard)

Model Number

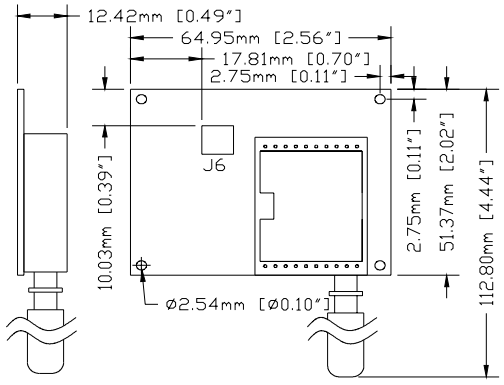
(RF Slave Module Part Number)
10158

(RF Host Box Part Number)
10146

Typical 320x240 Module and Related Product Ordering Examples:	
320x240x5.6-232-YLG	320x240 Yellow-Green/Black Display Module, with RS-232 Port Serial controller (Standard)
320x240x5.6-BUS3V-WHT	320x240 White/Blue Display Module, with 3 Volt logic Bus Interface controller
320x240x5.6-SLV-YEL	320x240 Yellow/Black Display Module, without controller (Slave parallel logic Interface)
90024-232-GRN	320x240 Product Development Kit with Standard Serial controller Green/Black Display module (1 display/controller module, Interface software, COMM & power cables, documentation and technical support are provided).
10158	RF Slave Transceiver Module (1 required per serial controller/display location)
10146	RF Host Transceiver Box Assembly (1 required per Host Computer System)



10146 RF Host Transceiver Mechanical Drawing



10158 RF Slave Mechanical Drawing

Data Sheet Table of Contents

General and Electrical Specifications

General Specifications – 320x240x5.6 Display and Controller Modules

Parameter	Value/Description	Units
Display Panel Resolution and Format	Cholesteric reflective LCD (ChLCD) material with contrasting color background, 240 rows by 320 columns.	
Pixel Pitch	71.4 dots per inch, or 0.3556 (X & Y Axis) between pixel centerlines.	mm
Pixel Size	0.3256 x 0.3256 (0.0128 x 0.0128 inches)	mm
ChLCD Image Area Dimensions	113.76 x 85.31 (142.2 mm, or 5.60" diagonal)	mm
Display Viewing Area Dimensions	114.76 x 86.31 ("X" and "Y" Dimensions)	mm
Bezel Outside Dimensions	139.0 x 110.55 ("X" and "Y" Dimensions) (Refer to drawing on sheet 1)	mm
Display Module Weight	5.3 (150 grams)	oz
*Operating Temperature Range	-20 to 80 (with standard KLC13 ChLCD Material. Low Temperature ChLCD Available)	C^0
**Image Clearing Temperature All Display Configurations	≈90-95 (with standard KLC13 ChLCD Material)	C^0
Storage Temperature Range	-45 to +100**	C^0
Illumination Source	None (reflective technology).	
Full Image Update Rate: Typical all module types	Approximately 1.0 (@ 24 C^0, in slowest, "1x" mode. Refer to Graphs in back of document.)	Sec.
UV Protection Recommendations	98% blocking of 380 nm and lower spectral components is recommended	
Recommended Transparent Plastic Display Cover	Acrylite OP-3-P-99, matte finish, with UV blocker, or equivalent	

*. Displays and controllers have been tested to operate beyond 0-70 C^0, however display electronics can only be guaranteed between 0 and 70 C^o. Image quality may degrade at temperatures above 65 C^0.

** Display will not retain image if the display temperature is greater than specified clearing temperature.

Electrical Characteristics – 320x240x5.6 Display and Controller Modules

Parameter	Min.	Typ.	Max.	Units	Condition
Logic Power Source Voltage (V_{CC}) ("BUS5V" Only)	4.75	5.0	5.25	VDC	
Logic Power Source Voltage (V_{CC}) ("BUS3V" & "SLV" Only)	3.0	3.3	3.6	VDC	
Battery or Power Source Voltage (V_{DD}) ("232" & "TTL" Only)	3.6	6	9	VDC	
Battery or Power Source Voltage (V_{DD}) (All Other versions)	2.5	3.2	18	VDC	
High Level Logic Input Voltage (V_{IH}) ("BUS5V" Only)	4.75			VDC	T = 23 C^o
High Level Logic Input Voltage (V_{IH}) ("BUS3V" & "SLV" Only)	3.0			VDC	T = 23 C^o
Low Level Input Voltage (V_{IL}) (Bus & "SLV" Versions Only)			0.8	VDC	T = 23 C^o
*Average Operating Power, During Image Generation (Display & Controller Consumption)		215		mW	While Driving Image, T= 23 C^0, V_{DD}=5V, Standard KLC13 ChLCD
**Average Operating Power, Controller Active, High Voltage Converter Active, Not Generating Image		127		mW	While Maintaining Previously Generated Image. V_{DD}=5V
**Average Operating Power, Controller Active, High Voltage Converter Inactive, Not Generating Image		79.5		mW	While Maintaining Previously Generated Image. V_{DD}=5V
**Average Operating Power, Controller in Idle mode, High Voltage Converter Inactive, Not Generating Image		35.7		mW	While Maintaining Previously Generated Image. V_{DD}=5V
**Average Power, In Sleep Mode (RS-232 Serial Controller Version)			6	µW	While Maintaining Previously Generated Image. V_{DD}=5V
Power w/ Display Disconnected & Maintaining Image forever.			0	µW	V_{DD}=0V

* Indicates power draw from V_{DD} and V_{CC} of display and controller combined (all types w/Controller). Consumption of Slave module ("SLV") will be significantly less. Max power output is during brief bulk erase (~62 ms @ 25 C^o)

** Indicates power draw from V_{DD} and V_{CC} (if BUS Version) of display and controller combined. Consumption of Slave module will be significantly less.

Note: All measurements taken with DMM, with display and onboard controller combination.

Interface and Pin-Out Information

Interface Connections – 320x240x5.6 Bus Display/Controller ("BUSXV" Versions, J7)

Pin #	Symbol	I/O	Description
4	Batt (V_{DD})	Pwr	Positive Power Source Termination (Battery or system power input). Must be greater than 2.5 VDC for proper operation.
1, 2	GND	Pwr	Ground Return Termination Point
3	V_{CC}	Pwr	+3.3V ("BUS3V"), or 5.0V ("BUS5V") Logic Power Input for Controller module.
11	D0	I/O	Data element 0 of Controller data bus.
12	D1	I/O	Data element 1 of Controller data bus.
13	D2	I/O	Data element 2 of Controller data bus.
14	D3	I/O	Data element 3 of Controller data bus.
15	D4	I/O	Data element 4 of Controller data bus.
16	D5	I/O	Data element 5 of Controller data bus.
17	D6	I/O	Data element 6 of Controller data bus.
18	D7	I/O	Data element 7 of Controller data bus.
4	nWR	I	Data bus write control line (active low)
5	nRD	I	Data bus read control line (active low)
6	nCE	I	Data bus, or module enable ("Chip Enable", active low).
7	C_nD	I	Command (High)/Data (low) select input. With nWR low (active): C_nD high = "Command Write", C_nD low = "Data Write" With nRD low (active): C_nD high = "Status Read", C_nD low = "Data Read"
10	nRST	I	Bus controller reset input (active low).
19	FS1	I	Font Select Input
20	nHalt	I	Controller Halt command input (active low).

Notes: All logic inputs to the device can not exceed typical 0 to 5.0 V levels. Logic Outputs will be 0 to 3.3 V levels.

Interface Connections, Power – 320x240x5.6 Display/Controller ("232" & "TTL", J1)

Pin #	Symbol	Description
1 (Marked with Pad/via)	Batt. (V_{DD})	Positive Power Termination
2 (Marked with O Pad/via)	GND	Negative Power Termination.

Interface Connections, COMM – 320x240x5.6 Display/Controller ("232", J5)

Pin #	Symbol	Description
1	RX_DATA	Module receive data input termination (RS-232). Module "Wake" input (Based on parked low condition).
2	GND	Ground termination point for Communication purposes (RS-232 use only).
3	TX_DATA	Module transmit data output termination (RS-232).

Interface Connections, COMM – 320x240x5.6 Display/Controller ("TTL", J3)

Pin #	Symbol	Description
1	RX_DATA	Module receive data input termination (w/ respect to controller module). Module Wake Input (Based on Parked High condition. Wakes w/ Low input, or when Input is below ~ 0.7 VDC). 0 - 3.3 Volt Logic Input.
2	Ground	Ground termination point.
3	TX_DATA	Controller module transmit data output termination. 0 - 3.3 Volt Logic Output.
4	+3.3 VDC	Power Source input to be used for Finished Product Level Shifters circuitry. Source not provided when Controller is in sleep mode.

Recommended Mating Connector - 320x240x5.6 Display/Controller (All Versions)

Recommended Mating Connectors (RS-232 Serial Controller version)	COMM (J5): 2.0 mm pitched, 1x3 Socketed Plug, Hirose P/N DF3-3S-2C, w/Socketed crimp contacts, Hirose P/N DF3-2428-SC, or equivalent. Power (J1): 2.54 mm pitched, 1x2 via's (plate through holes) in PCB.
Recommended Mating Connector (Bus Controller Versions), J7	2.54 mm pitched (0.1"), 2x10 or 20 conductor Ribbon Cable Socketed Header with Polarizing key, Au Contacts and Strain Relief, KDI P/N 20, CW Industries P/N CKR20G, or equivalent.
Recommended Mating Connectors (TTL Serial Controller version)	COMM (J3): 2.0 mm pitched, 1x4 Socketed Plug, Hirose P/N DF3-4S-2C, w/Socketed crimp contacts, Hirose P/N DF3-2428-SC, or equivalent. Power (J1): 2.54 mm pitched, 1x2 via's (plate through holes) in PCB.
Recommended Mating Connector ("SLV" version), J4	2.0 mm pitched, 2x10 or 20 conductor Socketed Header, KDI P/N 4349, Samtec P/N TLE-110-01-G-DV

Interface Connections – 320x240x5.6 Slave Display Module ("SLV" Version) Interface (J4)

Pin #	Symbol	I/O	Description
1	Batt (V_{DD})	Pwr	Positive Power Source Termination (Battery or system power input). Must be greater than 2.5 VDC for proper operation.
2	GND	Pwr	Ground Return Termination Point
*3	EN_PWR	I	Enable Module Power. Used to turn on/off the display (positive logic level input).
4	D0/ ROW_DATA	I	Data element 0 of module data bus, Row Data Input.
6	D1	I	Data element 1 of module data bus.
8	D2	I	Data element 2 of module data bus.
10	D3	I	Data element 3 of module data bus.
12	D4	I	Data element 4 of module data bus.
14	D5	I	Data element 5 of module data bus.
16	D6	I	Data element 6 of module data bus.
18	D7	I	Data element 7 of module data bus.
5	ENABLE	I	Enable. When disabled (low), all voltage outputs to ChLCD are removed.
7	LATCH	I	Latch. Used to present column image data to ChLCD material, and to zero the column data pointer (triggered at trailing edge).
9	PHASE	I	Phase or Framing. Dictates polarity and frequency of wave form to ChLCD material
11	RCLK	I	Row Clock. Used to clock data present on the row data input, D0 (triggered at trailing edge).
13	C_CLK	I	Column Clock. Used to clock data present on the 8-bit data bus interface (triggered at trailing edge).
15	T_ERASE	I	Transparent Erase. Used to increase image contrast during a transparent erase process.
19	V_{CC}	I	+3.3V Logic Power Input for Display.
20	TEMP	O	Temperature output of ChLCD material. Analog signal representative of ChLCD temperature.
17	N/C	-	No Connect

Notes: 1. All logic inputs to the device (Except EN_PWR) are required to be 0 to 3.3 V levels.
*2. EN_PWR does not need to be 0 to 3.3 levels, and needs only to be greater than 2.2 volts to be activated.

QVGA Display - Serial Controller Versions

Detailed Product Description

The Kent Displays 320x240x5.6 Displays are modular units designed for general-purpose graphic and character display applications. Intended uses include instrumentation, point of sale and other general-purpose indoor/outdoor hand held display applications.

As with all Kent Display cholesteric (ChLCD) products, the 320x240 modules contain the same optical and power saving advantages over traditional LCD products. After an image is generated on the module, it will remain indefinitely after power is removed, or until a new image is generated. The display does not require "power consuming" update signals to retain the image as with traditional LCD technologies. Furthermore, since the technology is completely reflective, no "power robbing" backlight is required to generate a high quality image. The memory and reflective features enable the display to be very energy efficient, enabling battery operation.

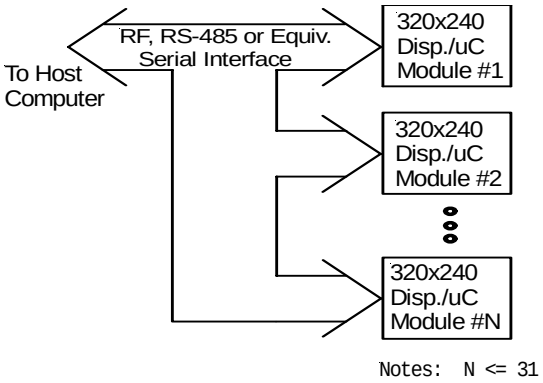
Since a ChLCD does not reflect polarized light, it exhibits a much wider viewing angle over traditional LCD technologies. Its extremely high contrast characteristics are exhibited consistently at viewing angles up to 90^{O} from perpendicular to the display in all directions. The display also exhibits excellent sunlight readability characteristics. The ChLCD technology has superior contrast characteristics than typical LCD display technologies.

The 320x240 display and Controller modules configured as indicated in Application Block Diagram will contain the following features:

- A full duplex asynchronous **RS-232, TTL**, or half-duplex **RF wireless serial interface** can be used to communicate between the host computer and the controller modules. The standard module protocol will support full duplex, half-duplex or simplex operations. Each module is assigned a unique address for multi-drop applications by selecting the module switch setting. 31 unique address selections are possible. Controller modules without the address switches are configured to address #1. Un-addressed modules will ignore the host commands. Broadcast messages are also supported (broadcast messages are assigned address number 0). The module communicates at a standard data rate of 19.2k baud. Other data rates, such as 38.4k, 9600 or other baud rates can be configured per request.

- **Character and graphical** interface **capabilit**y. Full or partial panel graphic or text images can be generated on a given display. The standard Kent Displays serial protocol supports all full or partial display graphical and text generation operational modes. The host computer can change a display image by outputting an ASCII character sequence, or graphic pixel data to the corresponding controller module. The protocol is designed to minimize the communication packet lengths, enable error checking, support multiple address locations, and provide wake-up commands to support a typical "hard wired" configuration (RS-232 or TTL format) or an unlicensed wireless RF communication link. Refer to Kent Displays document 25016 for a detailed communication protocol description.

- When using the **320x240 controller, ISO/IEC 8859-1 Compliant, text generator**, the user can **select** between a **30-row font, a 15-row font, a 10-row font and a 5x7 character font**. Using the 30-row font, up to 20 characters/line and 8 lines of text can be generated on the display. Using the 15-row font, up to 32 characters/line and 16 text lines can be output on a display. Using the 10 row font, up to 40 characters /line and 24 lines of text can be output on the displays. The 5x7 font can be used to output 53 characters/line and 30 lines of text. Using partial screen text images can also alter a text line within a display. The user can also use the windows-based text generator within the system graphic editor to create text based graphical images for output on the displays.



320x240 Display w/ Serial Controller Module to Host Computer Application Block Diagram

- The 320x240 modules have the capability to output text and graphic images in **Dynamic Update mode**. This mode of operation provides the user with the following update features:

  1. Scroll On, Scroll OFF
  2. Wipe ON, Wipe OFF
  3. Open ON, Open OFF
  4. Close ON, Close OFF
  5. Rotate ON
  6. Flashing, Fade and Swell

  Updating display sections in this mode provides an **animation sequence**, allowing the user to **view multiple frames of data in a moving sequence**. The most common application for this type of update method is to display numbers entered on a keypad in a scrolling fashion, frame by frame as the numbers are entered. Different "ON" and "OFF" presentation methods are also possible using the controller protocol and 320x240 Interface software provided with the products. The user simply has to enter the text, or draw the graphic image (Using the 320x240 Interface Graphic editor) to output in the static frame, and the 320x240 controller will create the frame sequences automatically for the user! The user can also **make the image section large**r (for instance, a 10 row font can be output using 20, 30, ... 80 rows of the display, essentially **"Stretching" the image vertically**.) by selecting the corresponding height adjustment feature on the user interface software. For static or single frames of data, the user can also **"stretch" the image horizontally by "2x"** using the software horizontal stretch option. This feature will limit image distortion when the characters are also stretched vertically using the "2x – 8x" feature described previously. Parameters on flashing images, such as the Invert and non-invert image duration's, and the number flashes for the respective image can also be controller by the user. The 320x240 Interface software provides a nice windows-based human interface to control these features.

Code Map, Serial Controller ISO/IEC 8859-1 Compliant 10 Row font

- The Kent Displays automatic "**Wake/Idle/Sleep mode**" is a standard feature, which **supports extended life battery operation**. If no messages or data is received from the host computer after a pre-determined duration, or no local diagnostic/control activity is sensed by the controller logic, the module will reduce power consumption by activating an "Idle" mode operation automatically. If the 320x240 controller senses no further activity, it will remove the load from the 3-9 volt power source. In "**Sleep** mode", the display module will contain the image generated, and power consumption will be less than 6 microWatts (power required to maintain the sleep control circuitry, RS-232 controller versions). The module will "wake-up" and resume normal operation after it receives the host wake-up signal. The standard sleep mode feature can be disabled using the protocol, for applications where power consumption is not important.

- Automatic temperature compensation circuitry to **extend the module operating temperature range to -20 and 80 C^0** (standard module electronics can only be guaranteed from 0 to 70 C^0)

- Diagnostic switches and an LED indication to support **local operation and contro**l.

General Specifications – 320x240x5.6 Display Modules with Serial Controller

Parameter	Value/Description	Units
30 Row Character Pixel Configuration/ Characters per display/ Lines per Display.	16 pixels wide by 30 pixels high/ 160 Characters/ 8 Text Lines	--
15 Row Character Pixel Configuration/ Characters per display/ Lines per Display	10 pixel wide by 15 pixels high/ 512 Characters/ 16 Text Lines	--
10 Row Character Pixel Configuration/ Characters per display/ Lines per Display	8 pixel wide by 10 pixels high/ 960 Characters/ 24 Text Lines	--
5x7 Character Pixel Configuration/ Characters per display/ Lines per Display	6 pixel wide by 8 pixels high/ 1590 Characters/ 30 Text Lines	--
Text Generator Character Set Capability	30 Row: **Modified Letter Gothic Bold 22 point. 15 Row: **Modified Fixed Distance 11 point. 10 Row: **Modified Fixed Distance 9 point. 5x7: Standard 5x7 characters.	--
Character Set Controls	Underline.	--
# Display Modules/ Controller	1	--
Message Storage Capability	36 Full Screen small font Text, 110 Full Screen large font Text, 6 Full Screen binary Graphic, 857 Partial Screen Text (Dyn. Update, Flashing or Normal), 14 Partial Screen binary Graphic, or 91 Partial Screen Dynamic Update Messages per controller.	--
Operating Temperature Range	*-20 to 80 (with standard KLC13 ChLCD Material. Low Temperature ChLCD Available)	C^o
Storage Temperature Range	-45 to +100 C^o.	C^o.
Possible Controller Address Selections	32 (standard, more are available per request)	
Standard Total Wakeup Duration without communication or local activity	20 (Typical)	Sec.
Standard Active Wakeup Duration Before implementing Idle Mode	10 (Typical)	Sec.
Serial Communication Format	RS-232: Asynchronous, full duplex, 8 bits/byte, 1 Stop, no parity. TTL: Asynchronous, full duplex, 8 bits/byte, 1 Stop, no parity.	--
Standard Baud Rate:	19.2k Baud (other rates, such as 38.4k, 9600, 4800, 2400 & 1200 are available per request).	--
Standard Protocol Format	Standard Kent Displays Character/Graphic Serial Protocol (refer to Kent Displays document 25016 for details).	--

* Indicates display electronics can only be guaranteed between 0 and 70^o ambient. Update rates will be slower at lower temperature extremes. Image quality may degrade at operating temperatures above 65 C^0.

** Indicates ISO/IEC 8859-1 compliant font, enabling expanded character generation beyond ASCCI codes 7F hexadecimal (127 decimal).

QVGA Display – Bus Controller Versions

Detailed Product Description

The Kent Displays 320x240x5.6 Displays are modular units designed for general-purpose graphic and character display applications. Intended uses include instrumentation, point of sale and other general-purpose indoor/outdoor hand held display applications.

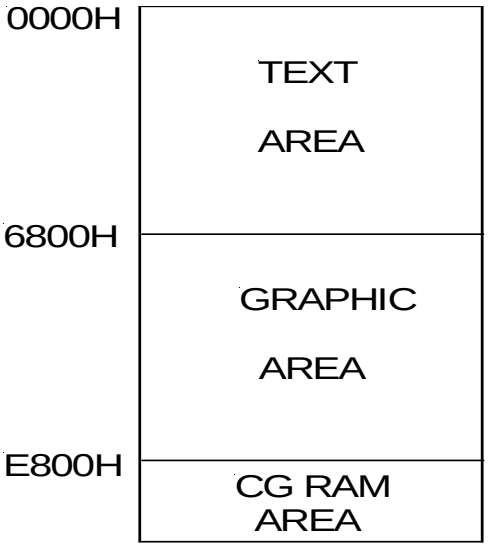
As with all Kent Display cholesteric (ChLCD) products, the 320x240 modules contain the same optical and power saving advantages over traditional LCD products. After an image is generated on the module, it will remain indefinitely after power is removed, or until a new image is generated. The display does not require "power consuming" update signals to retain the image as with traditional LCD technologies. Furthermore, since the technology is completely reflective, no "power robbing" backlight is required to generate a high quality image. The memory and reflective features enable the display to be very energy efficient, enabling battery operation.

Since a ChLCD does not reflect polarized light, it exhibits a much wider viewing angle over traditional LCD technologies. Its extremely high contrast characteristics are exhibited consistently at viewing angles up to 90^{o} from perpendicular to the display in all directions. The display also exhibits excellent sunlight readability characteristics. The ChLCD technology has superior contrast characteristics than typical LCD display technologies.

The interface for the 320x240 display with the Bus controller is modeled after the Toshiba T6963C LCD controller interface. The module J7 bus interface, outlined in the pin-out table on sheet 4 of this document, is intended for direct connection to a processor system bus. The following subsections outline the proper command interface to the bus controller.

RAM Interface

The bus controller provides 64k of RAM storage for data (text, graphic or character generated data) accessible by the interface. The following figure provides a memory map example outlining storage locations where data can reside in the 64k area. Data boundaries are dictated by the register home address location settings, which are explained later.



Memory Map Example for Text, Graphic and Character Generated Data in the Interface 64k RAM Bank.

Communication with the Bus Controller

The bus controller provides 64k of RAM storage for data (text, graphic or character generated data) accessible by interface. The following illustration provides a memory map example outlining storage locations where data can reside in the 64k area. Data boundaries are dictated by the register home address locations settings, which are explained later.

Reading Status Register

Before reading data, or writing data or controls to the Bus controller, the status register must be check before each byte access. To read the status register, the bus control lines must be in the following states:

nRD	Low (active)
nWR	High (inactive)
nCE	Low (active)
C_nD	High (Command access)
D0 – D7	Status Word Data will be presented by Display Bus Controller

The format for the status word data is as follows:

STA7 D7	STA6 D6	STA5 D5	STA4 D4	STA3 D3	STA2 D2	STA1 D1	STA0 D0

STA0	Command execution enable bit.	0 = Disable, 1 = Enable
STA1	Data Read/Write enable bit.	0 = Disable, 1 = Enable
STA2	Auto Mode data read enable bit.	0 = Disable, 1 = Enable
STA3	Auto Mode data write enable bit.	0 = Disable, 1 = Enable
STA4	Not Used.	0 = Disable, 1 = Enable
*STA5	Controller Operation bit.	0 = Disable, 1 = Enable
*STA6	Error Flag bit (used for Screen Peek and Screen copy commands)	0 = No Error, 1 = Error
*STA7	Blink condition bit.	0 = Display Off, 1 = Normal.

* Indicates bit function presently not supported by bus controller logic at the time of printing.

Note 1: STA0 and STA1 should be checked at the same time. In most modes of operation, the STA0 & STA1 bits are used for a status check.
Note 2: STA2 and STA3 are valid in when auto mode (typical mode used for loading data). STA0 and STA1 are not used in auto mode.

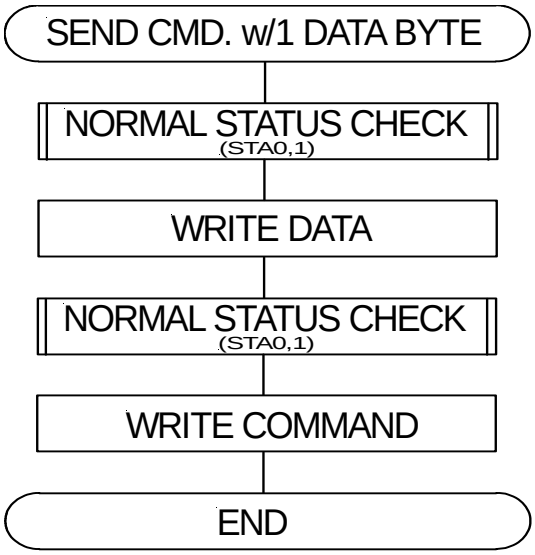
To access the status register, the following flow diagrams should be obeyed:

Note 3: If the interface does not obey the flow diagrams outlined above before each command byte or data byte access, then proper operation of the display bus controller cannot be guaranteed. The controlling software accessing the bus interface should not implement “delays” in place of the “Status check” logic outlined above. Inconsistent display operation will occur if the logic above is not implemented.

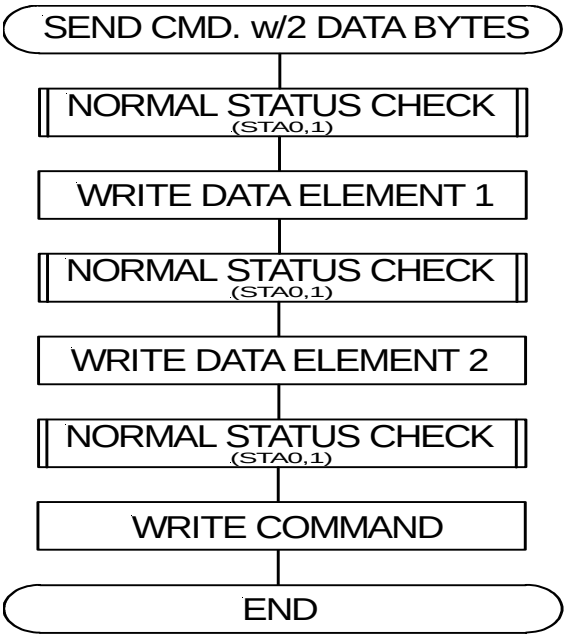
Sending Data and Commands to the Bus controller

When sending commands to the Bus controller, the data elements must proceed, or be sent before the corresponding command byte. The following flow diagram should be implemented.

a. Sequence for Commands with 1 data byte:



b. Sequence for Commands with 2 data bytes:



Commands Definitions:

Command Types	Codes	Data Element 1	Data Element 2	Function
REGISTER SETTINGS	00100001	X address	Y address	Set Cursor Pointer
	00100010	Data	00H	*Set Offset Register
	00100100	Low Address	High Address	Set Address Pointer
SET CONTROL WORD	01000000	Low Address	High Address	Set Text Home Address
	01000001	Columns	00H	Set Text Area
	01000010	Low Address	High Address	Set Graphic Home Address
	01000011	Columns	00H	Set Graphic Area
MODE SET	1000X000	-	-	*OR mode
	1000X000	-	-	*EXOR mode
	1000X000	-	-	*AND mode
	1000X000	-	-	*Text Attribute mode
	10000XXX	-	-	Internal CG ROM mode
	10001XXX	-	-	*External CG RAM mode
DISPLAY MODE	10010000	-	-	Display off
	1001XX10	-	-	Cursor ON, Blink OFF
	1001XX11	-	-	Cursor ON, Blink ON
	100101XX	-	-	Text ON, Graphic OFF
	100110XX	-	-	*Text OFF, graphic ON
	100111XX	-	-	*Text ON, graphic ON
DATA AUTO READ / WRITE	10110000	-	-	Set to Data Auto Write mode
	10110001	-	-	*Set to Data Auto Read mode
	10110010	-	-	Reset Auto Mode
*SCREEN PEEK	11100000	-	-	*Screen Peek
*SCREEN COPY	11101000			*Screen Copy

* Indicates Command is not presently supported by QVGA Bus Controller at the time of printing.

Setting the Cursor Pointer

Setting the cursor position is independent upon other data sent or read from the QVGA Bus controller. The X and Y-axis address locations indicated in the command definition table corresponding to below the corresponding character index position on the display. The QVGA bus controller will only output a cursor below the corresponding character position.

For instance, the cursor is to be moved to below the 5 character, on the 7th text line, when the display is set to 8x20 mode (8 text lines of 20 possible characters each), the following X and Y addresses should be sent:

X address: 4
Y address: 6

Setting the Text Home Address

The text home address is the address in the QVGA controller RAM for text display applications corresponding to the top left hand corner, character position of the display. Setting this address by the corresponding command is required before writing character data to the display RAM.

Setting the Text Area

The text area is defined in software by the number of text character columns in the display (characters per text line). Typically it should equate to the number columns set in hardware. Setting of this register by the corresponding command is typically required before writing character data to the display RAM.

The relationship between external RAM address locations, the text home address and the position on the display correspond to the following tables:

Right Most Address Location	...	Left Most Address Location	Text Line #
TH	...	TH + TA – 1	1
TH + TA	...	TH + 2TA – 1	2
TH + 2TA	...	TH + 3TA – 1	3
TH + 3TA	...	TH + 4TA – 1	4

Where: TH = Text Home Address.
TA = Text Area.

The following example applies:

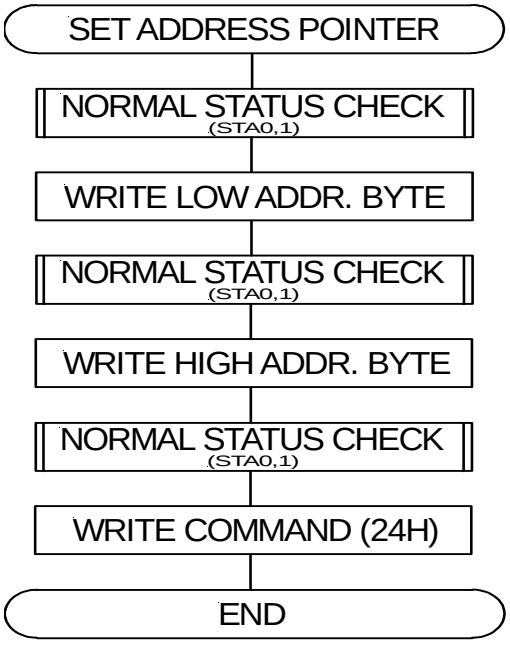
Text Home Address: 0500H
Text Area: 14H (20 Characters / Line)

Right Most Address Location				Left Most Address Location	Text Line #
0500H	0501H	....	0512H	0513H	1
0514H	0515H	...	0526H	0527H	2
0528H	0529H	...	053AH	053BH	3
053CH	053DH	...	054EH	054FH	4

Setting the Address Pointer

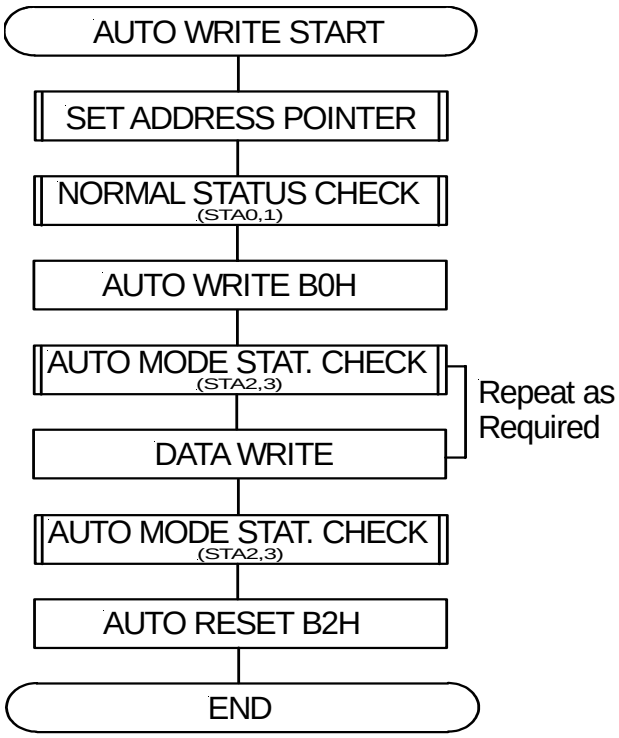
The "Set Address Pointer" command is used to determine the start address for writing data to, or reading data from the QVGA Bus controller RAM. This command typically proceeds a "Set to Data Auto Write" or "Set to Data Auto Read" mode command.

The following logic diagram applies to setting the address pointer. The same logic is applied to setting the text home address and text area commands.



Setting to Data Auto Read or Write Mode

The "Set to Data Auto Write mode" and the "Reset Auto Mode" are very useful when sending a full screen of data, or a full line of data to the QVGA Bus controller RAM. The "Set to Data Auto Write mode" command must be sent after the "Set Address Pointer" command is sent to the QVGA Bus controller. After the Bus controller is set to Auto Write mode, the address pointer will be automatically incremented after each data write to (or read from) the Bus controller RAM. When the Bus Controller is in Auto mode, the controller cannot accept other commands. Hence, the "Reset Auto Mode" will need to be sent to the bus controller after the RAM data is sent (or received), to enable other commands to be properly received by the bus Controller. The command sequence obeys the following the chart.



Character Code Map (Bus Controller Versions)

The following illustration outlines the 5x7 font bit-maps created by the Bus controller, when the internal ROM text generator is used:

Sample Bus Controller Interface Source Code

```
/*****************************************************************************
*
*               QVGA Bus Controller Sample Interface Program, Version 1.0
*                           ("C" Source Code Version)
*
*                       Display Size:  20 Column x 8 Lines
*
*****************************************************************************/
#include "25046aBus_Interface_Src.h"
#include <stdarg.h>
#include <stdio.h>



#define LCD_ADR                         0x0000                /* Base Address of LCD*/
                    /* Address on Data Bus for LCD Commands (Memory Mapped) */
#define LCD_CMD                         LCD_ADR + 0x0001
                    /* Address on Data Bus for LCD Data (Memory Mapped) */
#define LCD_DATA                        LCD_ADR + 0x0000

/* Register set */
#define Set_Cursor_Pointer              0x21
#define Set_Offset_Register             0x22
#define Set_Address_Poiner              0x24

#define Set_Text_Home_Address           0x40
#define Set_Text_Area                   0x41
#define Set_Graphic_Home_Address        0x42
#define Set_Graphic_Area                0x43

#define OR_Mode                         0x80
#define EXOR_Mode                       0x81
#define AND_Mode                        0x83
#define Text_Attribute_Mode             0x84
#define Internal_CG_ROM_mode            0x80
#define External_CG_ROM_mode            0x88

#define Display_Off                     0x90
#define Cursor_ON_Blink_OFF             0x92
#define Cursor_ON_Blink_ON              0x93
#define Text_ON_Graphic_OFF             0x94
#define Text_OFF_Graphic_ON             0x98
#define Text_ON_Graphic_ON              0x9C

#define Set_Data_Auto_Write             0xB0
#define Set_Data_Auto_Read              0xB1
#define Auto_Reset                      0xB2

#define Text_Home_Address               0x0000
#define Text_Home_Address_Low           0x00
#define Text_Home_Address_High          0x00
#define Graphic_Home_Address            0x6800
#define Graphic_Home_Address_Low        0x00
#define Graphic_Home_Address_High       0x68
#define Offsetreg                       0x02    /* 0x00 = 0x0000 - 07FF; 0x01 = 0x0800 - 0x0FFF;
                                                   0x02 = 0x1000 - 0x17FF ....*/
#define Text_Area                       0x14
#define Graphic_Area                    0x14
#define TextRows                        0x08

const unsigned char Sample_Text_Image[]  = {
/*           12345678901234567890 */
            "  This is a Sample\n"              /* Text Line #1 */
            "  Text Based Image\n"              /*           #2 */
            "   output on the\n"                /* .....     #3 */
            "    QVGA Display\n"
            "       in a\n"
            "     8 Line by\n"
            "    20 Character\n"
            "   Configuration\n"                /* Text Line #8 */
}; /* End Sample_Text_Image */
```

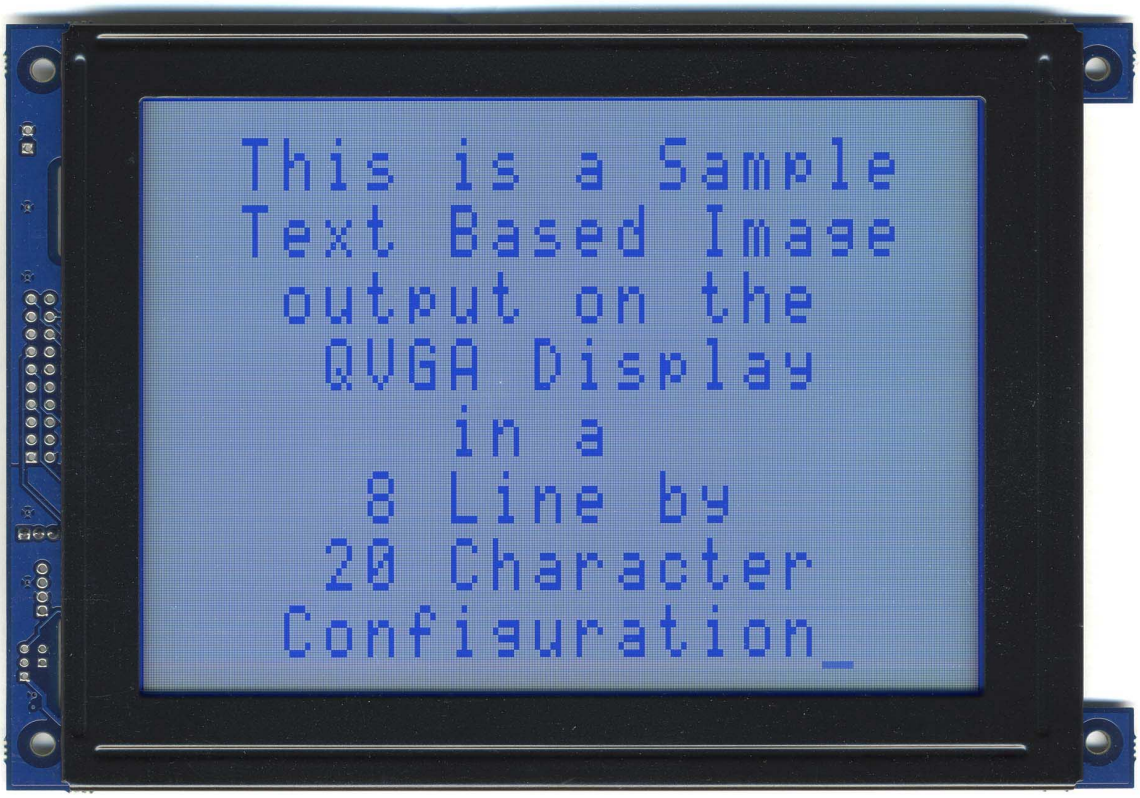
```
/*************************************************************************/
/*************************      MAIN PROGRAM      *************************/
/*************************************************************************/
void main ( void )                                          /* main entry for program      */
{

  unsigned char *ROM_Data, Data_To_LCD, Cursor_Y_Indx, Cursor_X_Indx;

/* Init LCD Registers & Fill Text & Graphic Home areas w/0's */
  LCD_init();

  LCD_2Data_cmd(Set_Address_Poiner, Text_Home_Address_Low, Text_Home_Address_High);
  LCD_cmd(Set_Data_Auto_Write);                                  /* Initialize for Data Load in 0,0
Position */

  ROM_Data = Sample_Text_Image;                                 /* Initialize Data Pointer */
  Cursor_Y_Indx = 0;                            /* Initialize Cursor Position to present state */
  Cursor_X_Indx = 0;
  while (*ROM_Data != '\0')
  {
    while (*ROM_Data != '\n')
    {                             /* Subtract Space, to convert to ASSCCI for LCD Txt. Gen. */
       Data_To_LCD = *ROM_Data - 0x20;
       Write_Data_AutoMode(Data_To_LCD);                  /* Write Data to LCD RAM & Incr. Ptr.*/
       Cursor_X_Indx++;
       ROM_Data++;
    }
    ROM_Data++;                                            /* Point pass end of Line Terminator */
    LCD_cmd(Auto_Reset);                          /* Auto Reset, to LCD, to load into LCD RAM*/
                                                  /* Move Cursor to end of current Line */
    LCD_2Data_cmd(Set_Cursor_Pointer, Cursor_X_Indx, Cursor_Y_Indx);
    if (*ROM_Data != '\0')
    {
      Cursor_Y_Indx++;
                                                  /* Move LCD Address Ptr. to start of Next Line */
      LCD_2Data_cmd(Set_Address_Poiner, Text_Home_Address_Low, (Cursor_Y_Indx * Text_Area));
      LCD_cmd(Set_Data_Auto_Write);               /* Start LCD RAM Data Write Loading Process */
      Cursor_X_Indx = 0;                          /* Initialize X-Axis Cursor Index */
    }
  }
} /* End main */

/*****************************************************************************
*   Function:  LCD_init
*
*   This function will init the display controller registers to the correct values
*   & load 0's (corresponding to space character) into the text & graphic home RAM
*   sections.
*
*****************************************************************************/
void LCD_init(void)
{
    unsigned char x,y;

    LCD_cmd(Display_Off);                                    /* Turn OFF Display (Blank it) */

                                                             /*Set text area home address*/
    LCD_2Data_cmd(Set_Text_Home_Address, Text_Home_Address_Low, Text_Home_Address_High);

/*Set text area set -- set to number of bytes per line to save ram or larger to make the calculation
of addresses easier*/
    LCD_2Data_cmd(Set_Text_Area, Text_Area, 0x00);

                                                             /*Set graphics area home address*/
    LCD_2Data_cmd(Set_Graphic_Home_Address, Graphic_Home_Address_Low, Graphic_Home_Address_High);

/*Set graphics area set -- set to number of bytes per line to save ram or larger to make the
calculation of addresses easier*/
    LCD_2Data_cmd(Set_Graphic_Area, Graphic_Area, 0x00);


                                                             /*Set CG RAM offset regester*/
    LCD_2Data_cmd(Set_Offset_Register, Offsetreg, 0x00);
```

```
/*                                      fill display Text Home Address Section with 0x00 (Space
Characters)
*/
    LCD_2Data_cmd(Set_Address_Poiner, Text_Home_Address_Low, Text_Home_Address_High);
    LCD_cmd(Set_Data_Auto_Write);
    for(x=0;x<TextRows;x++)
    {
      for(y=0;y<Text_Area;y++) {
      Write_Data_AutoMode(0x00);
      }
    }
    LCD_cmd(Auto_Reset);

/*  fill display Text Home Address Section with 0x00 (Space Characters)
*/
    LCD_2Data_cmd(Set_Address_Poiner, Graphic_Home_Address_Low, Graphic_Home_Address_High);
    LCD_cmd(Set_Data_Auto_Write);
    for(x=0;x<TextRows;x++)
    {
      for(y=0;y<Text_Area;y++)
      {
        Write_Data_AutoMode(0x00);
      }
    }
    LCD_cmd(Auto_Reset);
                                        /*Set Cursor Pointer to Top-Left Position on Display*/
    LCD_2Data_cmd(Set_Cursor_Pointer, 0x00, 0x00);

/*Set Display Mode (Turn ON display w/Blank All spaces Image, Internal CG Text Mode, & Turn ON
Blinking Cursor) */

    LCD_cmd(Text_ON_Graphic_OFF | Cursor_ON_Blink_ON);

} /* end LCD_init */

/*********************************************************************************
*   Function:      LCD_cmd
*
*   This function will output a 1 byte command (1 bytes total) to the LCD
*   Bus Controller, using the input parameter passed by the calling context.
*
*********************************************************************************/
void LCD_cmd(unsigned char cmdcode)
{
  LCD_status_check();
  LCD_write_control(cmdcode);
} /* End LCD_cmd */
/*********************************************************************************
*       Function: LCD_2Data_cmd
*
*       This function will output a 2 data byte command (3 bytes total) to the LCD
*       Bus Controller, using the 3 input parameters passed by the calling context.
*
*********************************************************************************/
void LCD_2Data_cmd(unsigned char cmdcode, unsigned char Data1, unsigned char Data2)
{
  LCD_status_check();
  LCD_write_data(Data1);
  LCD_status_check();
  LCD_write_data(Data2);
  LCD_status_check();
  LCD_write_control(cmdcode);
}  /* End LCD_2Data_cmd */

/*********************************************************************************
*       Function: LCD_status_check
*
*       This function will implement a normal mode status check of the LCD Bus interface
*       status register.  Execution will not return to the calling context, unless the
*       2 least significant bits of the LCD controller status register are set (indicating
*       it is OK write or read data to the data bus).
*
*********************************************************************************/
void LCD_status_check(void)
```

```
{
  int i=0;
  while ((LCD_read_control() & 0x03) != 0x03)    /* wait for status check STA0,1 = 3 */
  {
        /*      Possible Loop limit check which can be added
        i++;
        if (i>1000)
        {
                break;
        }
        */
  }
}   /* End LCD_status_check */

/*****************************************************************************
*       Function:  LCD_read_control
*
*       This function will return to the calling context (as an output parameter) the contents
*       of the bus interface status register (Data from a memory Read, from the LCD_CMD address
*       location).
*
*****************************************************************************/
unsigned char LCD_read_control(void)
{
  volatile unsigned char *base = (unsigned char *)(LCD_CMD);
  unsigned char a;
  a = *base;
  return (a);
} /* end LCD_read_control */

/*****************************************************************************
*       Function:       LCD_write_control
*
*       This function will output a single command byte (provided as as input parameters by
*       the calling context), to the LCD Bus interface command address.
*
*****************************************************************************/
void LCD_write_control(unsigned char a)
{
  volatile unsigned char *base = (unsigned char *)(LCD_CMD);
  *base = a;
}/* end LCD_write_control */

/*****************************************************************************
*       Function:       LCD_write_data
*
*       This function will output a single data byte (provided as as input parameters by
*       the calling context), to the LCD Bus interface data address.
*
*****************************************************************************/
void LCD_write_data(unsigned char a)
{
  volatile unsigned char *base = (unsigned char *)(LCD_DATA);
  *base = a;
}/* end LCD_write_data */

/*****************************************************************************
*       Function:       Write_Data_AutoMode
*
*       This function will output a single data byte (provided as as input parameters by
*       the calling context) to the LCD Bus Interface RAM, and is to be used after the Bus
*       controller is configured into Auto Data Load mode.
*
*****************************************************************************/
void Write_Data_AutoMode(unsigned char Data1)
{
  LCD_status_AutoMode_Write();
  LCD_write_data(Data1);
}/* end Write_Data_AutoMode */

/*****************************************************************************
*       Function:       LCD_status_AutoMode_Write
*
*       This function will implement a Auto Mode Write status check of the LCD Bus interface
```

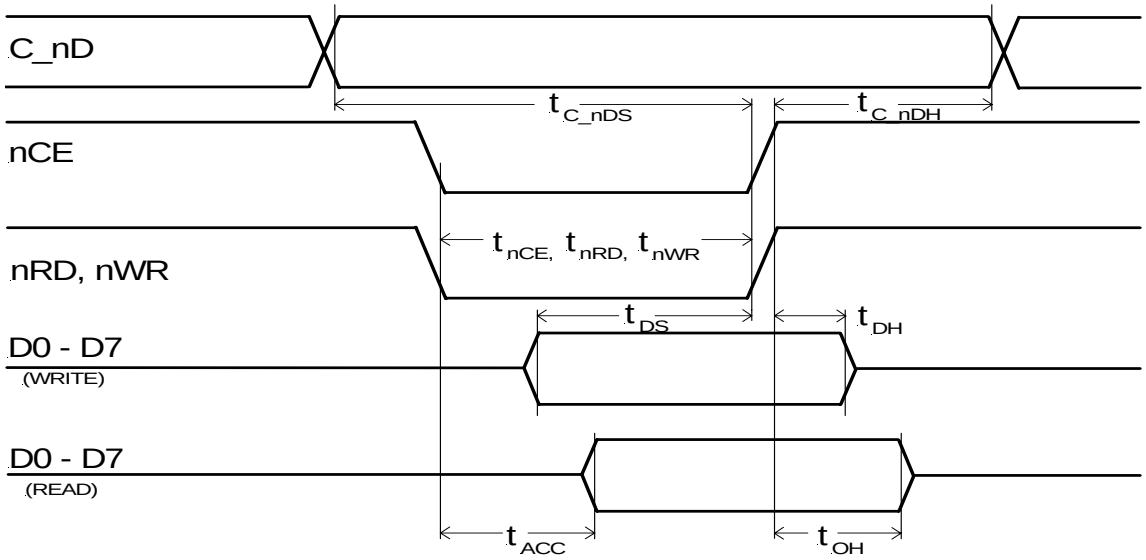
```
*       status register.  Execution will not return to the calling context, unless the
*       3rd bit (D3) of the LCD controller status register are set (indicating
*       it is OK write Auto Mode data to the data bus).
*
*********************************************************************************/
void LCD_status_AutoMode_Write (void)
{
  while ((LCD_read_control() & 0x08) != 0x08)    /* check auto write mode STA2,3 = 8 */
  {
        /*  Possible Loop limit check which can be added
        i++;
        if (i>1000)
        {
                break;
        }
        */
  }
} /* End LCD_status_AutoMode_Write */

/* 25046aBus_Interface_Src.h */
void LCD_init(void);
void LCD_cmd(unsigned char);
void LCD_2Data_cmd(unsigned char, unsigned char, unsigned char);
void LCD_status_check(void);
unsigned char LCD_read_control(void);
void LCD_write_data(unsigned char);
void LCD_write_control(unsigned char);
void Write_Data_AutoMode(unsigned char);
void LCD_status_AutoMode_Write (void);
const unsigned char Sample_Text_Image[];
/*  */
/* End 25046aBus_Interface_Src.h
```

Resultant QVGA Image and Cursor Position from Sample Bus Interface Program



(Sample Program Image on White/Blue QVGA BUS Display Configured to 8x20 Character Mode)

Bus Interface Timing Diagram and Specifications

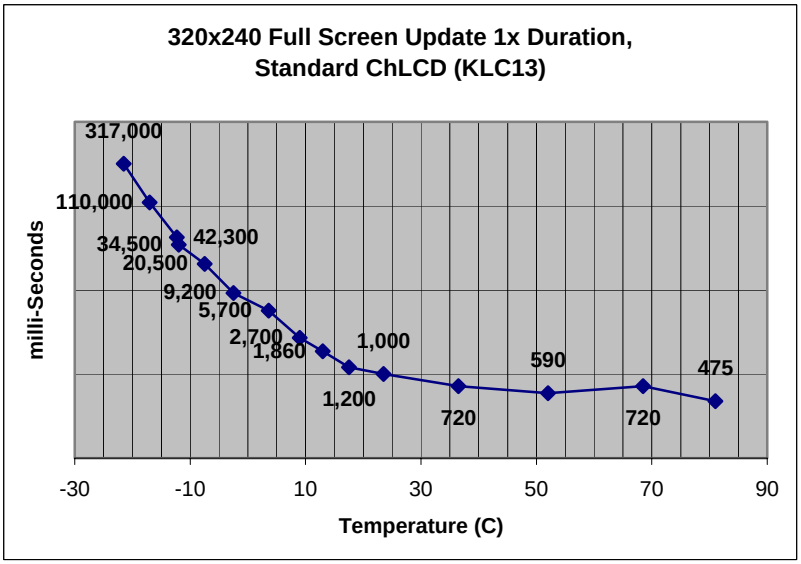


Item	Figure Symbol	Test Conditions	Min	Max	Unit
C_nD Setup Time Duration	t_{C_nDS}	-	*100	-	nS
C_nD Hold Time Duration	t_{C_nDH}	-	*10	-	nS
nCE, nRD, nWR Pulse Widths	t_{nCE}, t_{nRD}, t_{nWR}	-	*80	-	nS
Data Setup Time Duration	t_{DS}	-	*80	-	nS
Data Hold Time Duration	t_{DH}	-	TBD	-	nS
Access Time Duration	t_{ACC}	-	-	*150	nS
Output Hold Time Duration	t_{OH}	-	TBD	TBD	nS

* Indicates item specified is a conservative number. Actual QVGA parameter should be much better than value specified.

Update Speed Information - QVGA Display

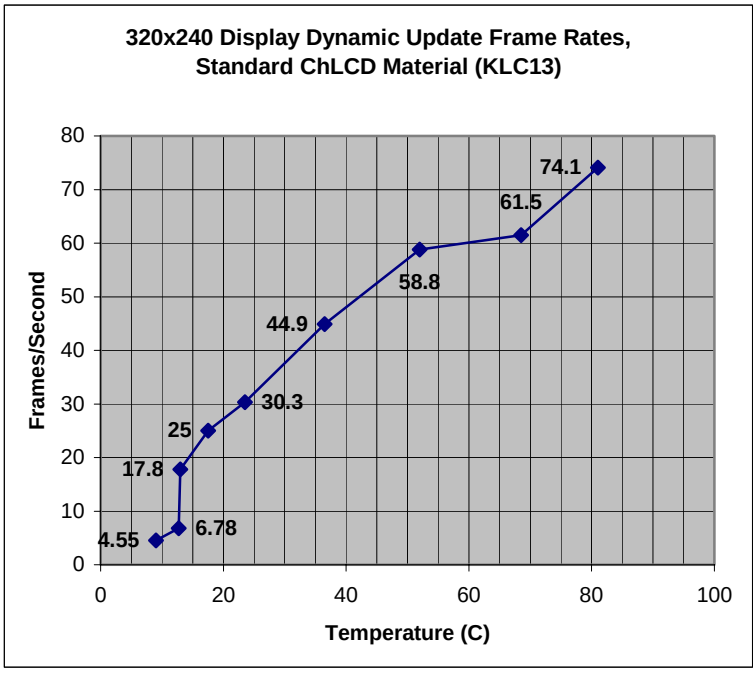
Full Screen Update Rate Information, Standard ChLCD Material (KLC13)



The chart to the left illustrates measured full screen (all 240 rows) update durations, with respect to temperature for a 320x240x5.6 display and controller module. Each value indicated on the chart represents the duration required, in milli-seconds to generate a full screen image on all 240 rows of the display module, when generating the image in the slowest, 1x mode. The durations indicated are approximately ½ the values stated when 2x mode is used, approximately 1/3 when 3x mode is used, ... As indicated on the chart, typical update duration of 1.0 second (1,000 mS) can be expected at typical room temperatures (23.5 C).

Full screen update durations reflect measurements taken on 320x240x5.6 Yellow-Green/Black display at the ambient temperatures indicated, when generating an image in the slowest 1x mode. A slightly slower update duration can be expected for a Green/Black color combinations. As indicated, ***typical update duration of 1,000 mS (1.0 second) can be expected at room temperatures*** *(23.5 C^0).*

Dynamic Update Information (Cumulative Drive™), Standard ChLCD Material (KLC13)



Frame rates illustrated reflect measurements taken of the dynamic update of a 10 row (or 20, 30, .. 80 row if "2x-8x" feature implemented) 320x240x5.6-232-YGN display section during the presentation "ON" of a particular image. A slightly faster frame rate will occur when measuring the presentation "OFF" of an image. A slightly slower frame rate can be expected with a Green/Black color combination. As indicated, ***a typical frames rate of approximately 30 frames/second can be expected at room temperature****s (23.5 C^0).*

The chart to the left illustrates the maximum frame rates possible with the 320x240-display with Serial controller combination. The following additional features are provided and can be programmed for each dynamic update message loaded into the controller RAM:

1. **Scroll, Wipe, Open, Close** and **Rotate** presentation methods.
2. Different ON and OFF presentation methods are available for each message (i.e. Rotate ON, Pause, then Scroll OFF).
3. Pause Duration's to allow the requested static frame to remain on the display between ON & OFF presentation methods are programmable from 0.1 to 25.5 seconds, in 0.1-second increments.
4. "2x – 8x" feature – Graphic or text data can be doubled, tripled, ... or eight times in size vertically to create a larger image update area.
5. Image invert or non-invert characteristics.

The controller module is also capable of generating **flashing** text messages, allowing the user to dictate the invert pause duration, non-invert duration (each can be set up to 25.5 seconds, in 0.1 second increments), and the number of flashing cycles to implement.

Static frames of data can also be output in a "Swell" and "Fade" modes, which enhance image appearances or aesthetics during the update process.

Static or Single frame images can also be updated in **"2x" horizontal stretch mode**, where the characters are twice as wide as specified previously. Using this feature will limit image distortion when the characters are also stretched vertically using the "2x – 8x" feature described in the bullet above.

Module Installation and Local Control Descriptions

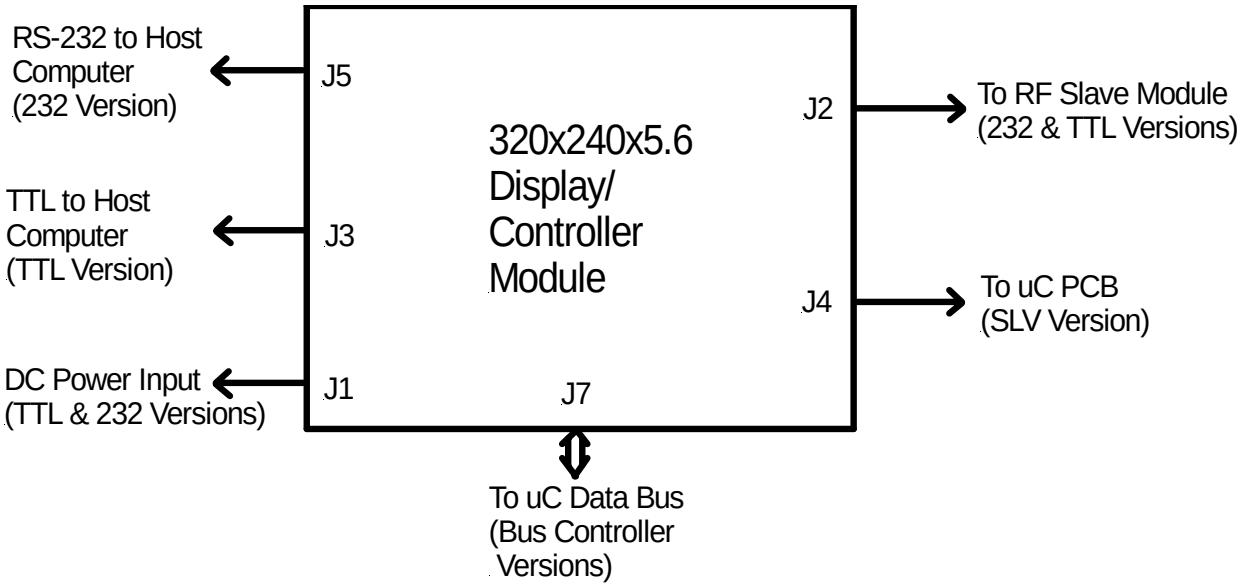
The following interconnect rules apply when installing only the 320x240x5.6 display module into the finished product:

1. Mount the display module into the finished product using the four mounting holes on the module printed circuit board.

2. In order to preserve the quality and life of the display, the finished product design should incorporate a transparent protective cover to protect the viewing area of the display. Use a material that can block UV light, has anti-glare properties and provides protection from user applied pressure points.

3. Mount the cover as close as possible to the face of the display. Use Acrylite OP-3-P-99, matte finish, with UV blocker, or equivalent material.

4. Connect the display/controller module headers to the finished product electronics, as indicated in the "Interface and Pin-Out Information section of this document, and as illustrated in the following figure.

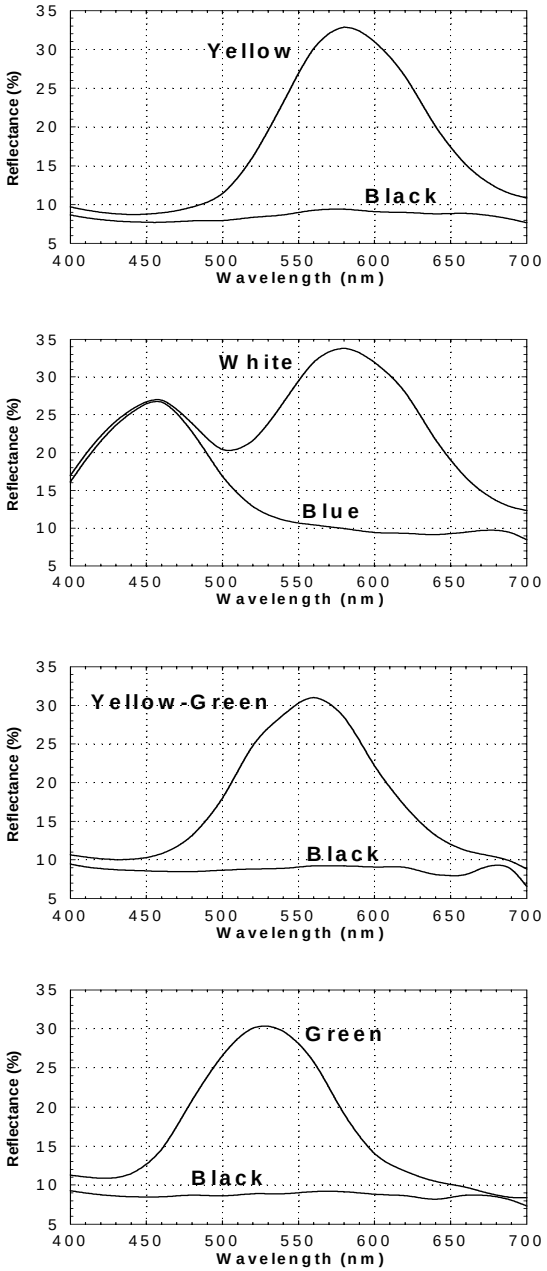


Typical Finished Product Electronics to QVGA Display/Controller Interconnect Diagram

Local Control/Diagnostic Descriptions

Symbol	Description
"RESET" (SW2)	Controller Reset momentary switch.
"TEST" (SW4)	Controller Diagnostic momentary switch (For local control).
D8	LED Diagnostic Output (For local control feedback).
"WAKE" (SW1)	Module Local "Wakeup" momentary switch (Serial Controller Versions Only).
Address, Bit 0 (R20, or SW3, pins 1-12)	Module Address Bit0 (= 1) Shorting Pads, or SW3 Bit0 position (Serial Controller Versions Only).
Address, Bit 1 (SW3, pins 2-11)	Module Address Bit1 (= 2) Shorting Pads, or SW3 Bit1 position (Serial Controller Versions Only).
Address, Bit 2 (SW3, pins 3-10)	Module Address Bit2 (= 4) Shorting Pads, or SW3 Bit2 position (Serial Controller Versions Only).
Address, Bit 3 (SW3, pins 4-9)	Module Address Bit3 (= 8) Shorting Pads, or SW3 Bit3 position (Serial Controller Versions Only).
Address, Bit 4 (SW3, pins 5-8)	Module Address Bit4 (= 16) Shorting Pads, or SW3 Bit4 position (Serial Controller Versions Only).

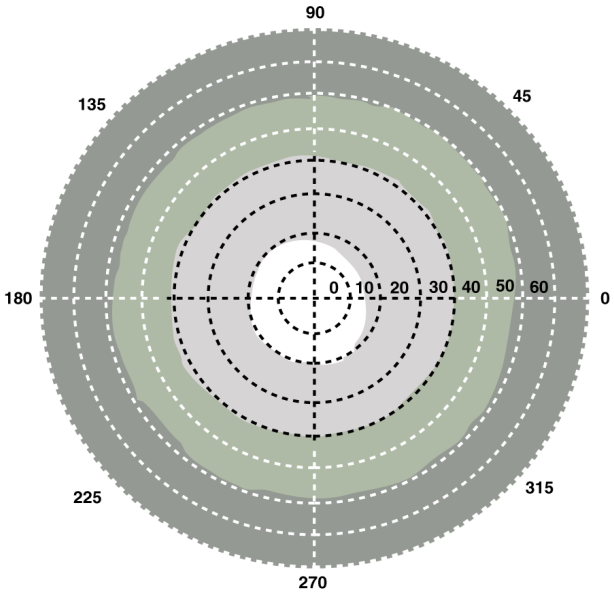
Optical Characteristics for Typical Standard ChLCD (KLC13), Color Configurations:



The above reflectance curves reflect measurements taken from a single pixel. Actual reflectance will depend on display resolution, aperture ratio and other factors.

Typical ChLCD Spectral Reflectance Characteristics

The graphs to the left outline the spectral reflectance characteristics for a given display pixel when switched either of the two possible stable states; reflective planar or transparent focal conic. The top line in each chart outlines the reflective characteristics for the planar state. The bottom line outlines the reflective characteristics for the transparent focal conic state. Graphs for the 4 standard color combinations are illustrated.



Contrast Ratio Polar Representation

As illustrated in the polar graph above, all Kent Displays ChLCD products have a 360-degree viewing cone. **Contrast** at near normal viewing angles **is as high as 25:1 and reflectivity up to 35%** of incident light. Contrast reduces with increased viewing angle, but is still excellent at 11:1 when viewed at the edge of the display. Since no polarizers are used, display contrast reduces uniformly in all directions when the viewing angle is increased.