

An OLED Interface Using the Z8 Encore! XP[®] Series of MCUs

AN032901-0611

Abstract

This application note demonstrates how to interface a serial organic light emitting diode (OLED) display module with the UART peripheral block on Zilog's Z8 Encore! XP Series of MCUs. The source code associated with this application provides device driver APIs to display text and graphics on the μ OLED-128-G1(SGC) serial OLED display module.

-
- **Note:** The source code file associated with this application note, [AN0329-SC01.zip](#), is available for download on [zilog.com](#). This source code has been tested with version 5.0.0 of ZDSII for Z8 Encore! XP-powered MCUs. Subsequent releases of ZDSII may require you to modify the code supplied with this application note.
-

Z8 Encore! XP UART Features

The Z8 Encore! XP UART peripheral is a full-duplex communication channel capable of handling asynchronous data transfers. Features of the UART include:

- 8-bit asynchronous data transfer
- Selectable even- or odd-parity generation and checking
- Option of one or two stop bits
- Separate transmit and receive interrupts
- Separate transmit and receive enables
- Framing, parity, overrun, and break detection
- 16-bit Baud Rate Generators (BRG)
- Selectable 9-bit MULTIPROCESSOR Mode with three configurable interrupt schemes
- Baud Rate Generator timer mode
- Driver enable output for external bus transceivers

The UART consists of three primary functional blocks: transmitter, receiver, and baud rate generator. The UART's transmitter and receiver each function independently but use the same baud rate and data format. Figure 1 shows the UART architecture.

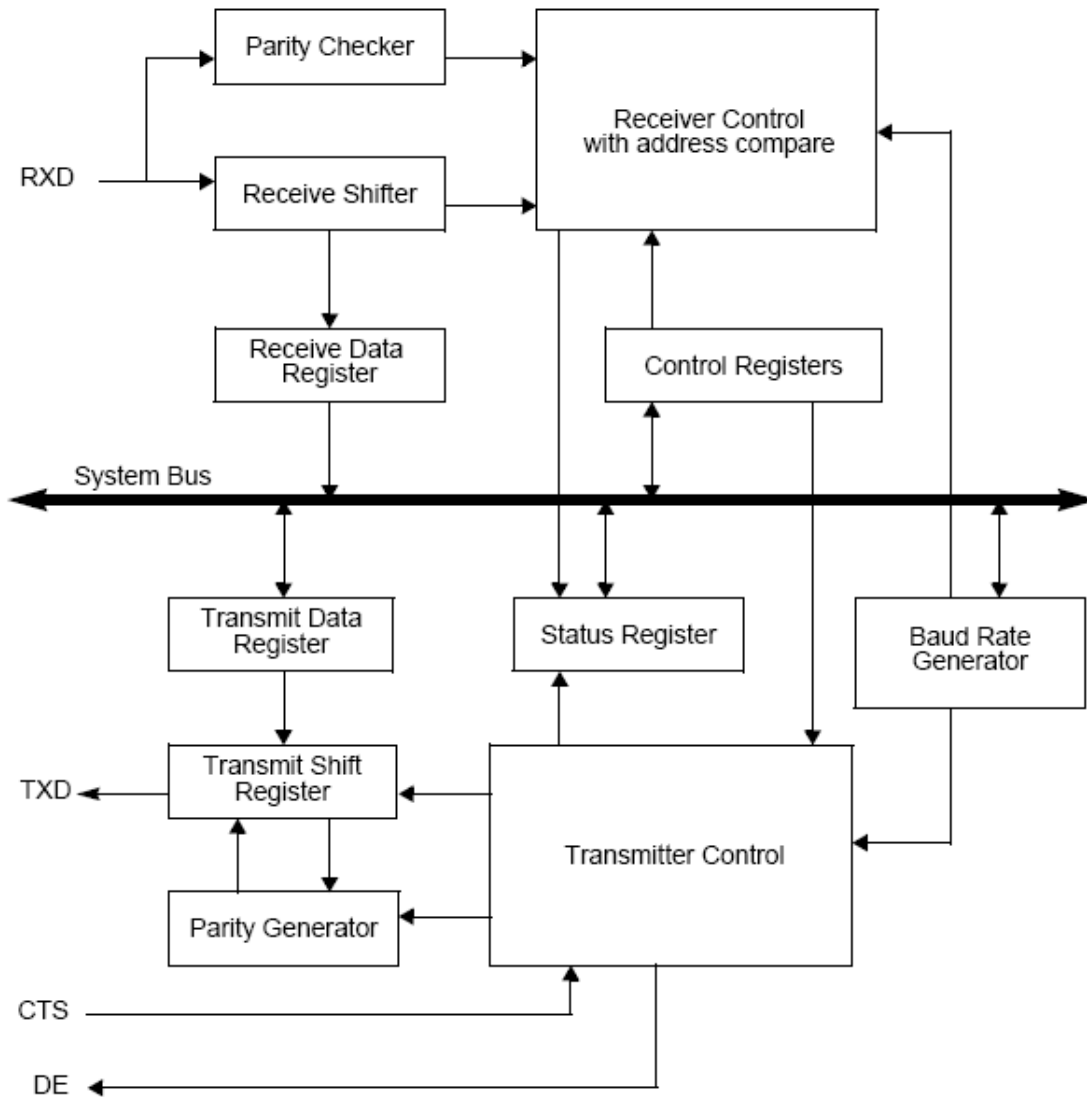


Figure 1. Diagram of the Z8 Encore! XP UART Block

- **Note:** For a description about how to initialize and facilitate data transfer via the Z8 Encore! XP UART function for Zilog's F0830 and F083A devices, please refer to the Application Note titled [An Interrupt-Driven UART for Z8 Encore! XP and Z8 Encore! MC MCUs \(AN0330\)](#); also see the Application Note titled [Software UART for the Z8 Encore! XP MCU \(AN0147\)](#).

OLED Display Module

An organic light emitting diode, or OLED, is an LED composed of organic compounds that emit light when electricity is applied. OLED displays can use either passive-matrix or active-matrix addressing schemes; each of these types has its advantages.

Passive-matrix OLEDs (PMOLED) contains strips of cathode and anode, and organic layers. The anode and cathode strips are arranged perpendicular against each other so that their intersections form the pixels. Active-matrix OLEDs (AMOLED) uses full layers of cathode, anode, and organic layers. The anode layer overlays a thin film transistor array that forms a matrix, which defines the pixels. AMOLEDs consume less power, thus, making it more efficient for large-screen displays such as OLED TVs. While PMOLEDs are cheaper to create but consume more power than AMOLEDs, it makes it more suitable for small-screen displays such as mobile phones, digital cameras and PDAs.

OLED displays have several advantages over LCD displays, some of which are:

- Thinner, lighter, and more flexible than ordinary LEDs or LCDs
- Brighter than LCDs
- Does not require backlighting to make displays viewable, thus, making it consume much less power than LCDs
- Have larger fields of view, reaching up to 170 degrees

OLED graphic displays are usually offered as modules, which can be interfaced using various serial interfaces, such as UART, I²C, and SPI. The μ OLED-128-G1(SGC) device used in this application is a 128x128 PMOLED screen module that functions as a slave peripheral device with a bidirectional serial interface to the Z8 Encore! XP Series of MCUs via its UART peripheral.

Software Implementation

The OLED device used in this application note is the μ OLED-128-G1(SGC) by 4D Systems. For detailed specifications, please refer to <http://www.4dsystems.com.au>. This device features an embedded GOLDELOX-SGC graphics controller to support graphics, text and an image display via a serial interface. Additionally, the OLED module's serial interface features are:

- Full duplex 8-bit data transmission and reception via the Tx and Rx pins
- Data format: 8 bits, no parity, 1 stop bit
- Autobaud feature that supports baud rates ranging from 300bps up to 256kbps

Initialization

Initializing the GOLDELOX-SGC graphics controller requires the following operations:

- Wait for a minimum 500msec delay after power-up or reset for the OLED module to settle. If a μ SD card is inserted, an additional delay time – from approximately 75msec to 1 second – must be considered.
- The Z8 Encore! XP Series MCU transmits the Autobaud command and waits for an acknowledgement (ACK) from the μ OLED device; see Figure 2.

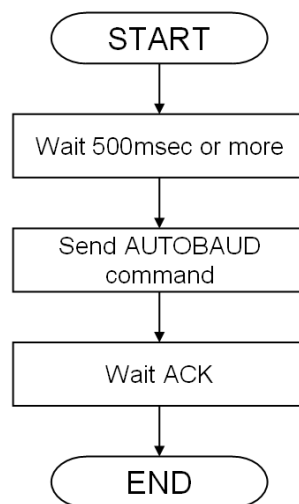


Figure 2. Initialization Flowchart

Command Execution

The GOLDELOX-SGC is a slave device and all communications must be initiated by the Z8 Encore! XP Series MCU.

In response to commands that do not require any specific data reply from the OLED, the OLED returns an ACK or a Not Acknowledge (NACK). In response to commands that are waiting for data from the OLED, the OLED responds with the data instead of an ACK or NACK. Such commands can cause the OLED to take a certain amount of time to deliver a response, depending on the type of command and the type of operation that the command requires the OLED to perform.

Figure 3 offers a graphical representation of the OLED command protocol.

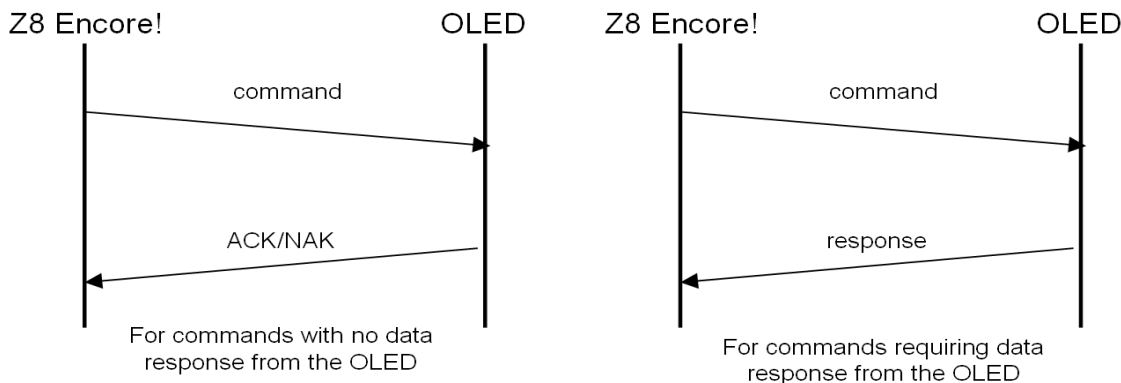


Figure 3. OLED Command Protocol

A set of serial commands is available at the link provided in the References section of this document.

- **Note:** APIs are provided in this document’s companion source code file, [AN0329-SC01.zip](#), to implement certain commands. A list of these APIs is provided in [Appendix A. µOLED Driver APIs](#) on page 7.

Hardware Setup

The OLED device is tested and run using a [Z8 Encore! XP Dual F1680 28-pin Development Kit](#). The APIs for this device can be used on other devices, though the UART will likely require configuration. For examples, please see the Application Note titled [An Interrupt-Driven UART for Z8 Encore! XP and Z8 Encore! MC MCUs \(AN0330\)](#) and/or the [Z8 Encore! XP MCU Application Note \(AN0147\)](#) for UART details.

Figure 4 illustrates the simple UART interface between the Z8 Encore! XP F1680 Development Board and the OLED.

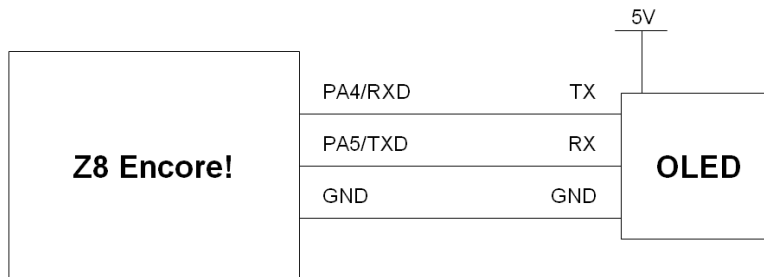


Figure 4. An OLED-UART Interface with the Z8 Encore! XP Series MCU

References

The following documents associated with the Z8 Encore! XP Series of MCUs are available on www.zilog.com.

- [Z8 Encore! XP F1680 Series Product Specification \(PS0250\)](#)
- [Z8 Encore! XP F1680 28-Pin Series Development Kit User Manual \(UM0203\)](#)
- [Software UART for the Z8 Encore! XP MCU Application Note \(AN0147\)](#)
- [An Interrupt-Driven UART for Z8 Encore! XP and Z8 Encore! MC MCUs Application Note \(AN0330\)](#)

The following document can be found on <http://www.4dsystems.com.au>.

- [µOLED Serial Display Module Specifications and Serial Commands List](#)

Appendix A. μ OLED Driver APIs

Table 1 lists the APIs contained in the accompanying AN0329-SC01 source code file. These APIs follow the specifications described in the μ OLED Serial Display Module Specifications and Serial Commands List document published by 4D Systems.

Table 1. μ OLED Driver API Parameters and Descriptions

API	Description	Parameters	Return Value(s)
OLED_SetAutoBaud	Enables the OLED device to adapt to the host baud rate. This command must be the very first command to be sent to the OLED after power-up or reset.	None.	None.
OLED_GetDeviceInfo	Requests all information from the OLED device about its characteristics and capability.	<i>output</i> : output device information to UART only or to UART and OLED display).	<i>mstOLED_Device Info</i> : device information.
OLED_ReplaceBackground	Changes the current background color.	<i>color</i> : 16-bit RGB representation of the desired color.	None.
OLED_ClearScreen	Clears the entire screen using the current background color.	None.	None.
OLED_DrawCircle	Draws a colored circle with characteristics specified in the parameters.	<i>x, y</i> : location of the circle's center. <i>radius</i> : radius of the circle. <i>color</i> : 16-bit RGB representation of the desired color for the circle. <i>fill</i> : true, if circle is solid. Otherwise, circle outline only.	None.

Note:

1. The **vertices* parameter must be included in the OLED_DrawPolygon API.
2. The **string* parameter must be included in the OLED_DrawButtonTxt and OLED_DrawStringTxt APIs.

Table 1. μ OLED Driver API Parameters and Descriptions (Continued)

API	Description	Parameters	Return Value(s)
OLED_DrawTriangle	Draws a colored triangle with characteristics specified in the parameters.	<i>x1, y1</i> : 1st vertex of the triangle. <i>x2, y2</i> : 2nd vertex of the triangle. <i>x3, y3</i> : 3rd vertex of the triangle. <i>color</i> : 16-bit RGB representation of the desired color for the triangle. <i>fill</i> : true, if triangle is solid. Otherwise, triangle outline only.	None.
OLED_DrawLine	Draws a colored line on the screen.	<i>x1, y1</i> : point1 of the line. <i>x2, y2</i> : point2 of the line. <i>color</i> : 16-bit RGB representation of the desired color for the line.	None.
OLED_DrawPixel	Draws a colored pixel at the specified location.	<i>x, y</i> : location of pixel point to draw. <i>color</i> : 16-bit RGB representation of the desired color for the pixel.	None.
OLED_DrawPolygon	Draws a polygon up to a maximum of 7 vertices. ¹	<i>numVertices</i> : number of vertices the polygon should have. <i>color</i> : 16-bit RGB representation of the desired color for the polygon.	None.
OLED_SetFont	Selects one of the available internal fonts. Changes will take effect after the command is sent. Any character drawn on the screen using the previous font set will remain unchanged.	<i>fontsize</i> : defines which font set to use on the next printing of a string or characters.	None.

Note:

1. The **vertices* parameter must be included in the OLED_DrawPolygon API.
2. The **string* parameter must be included in the OLED_DrawButtonTxt and OLED_DrawStringTxt APIs.

Table 1. μ OLED Driver API Parameters and Descriptions (Continued)

API	Description	Parameters	Return Value(s)
OLED_DrawCharTxt	Displays an ASCII character on the screen at the specified position.	<i>col</i> : horizontal position of the character. <i>row</i> : vertical position of the character. <hr/> <i>color</i> : 16-bit RGB representation of the desired color for the pixel. <hr/> <i>data</i> : character data to display.	None.
OLED_DrawButtonTxt	Places a text button on the screen with the specified string written onto it.	<i>state</i> : button state (pressed or not pressed). <hr/> <i>x, y</i> : top left start position of the button. <hr/> <i>bColor</i> : 16-bit RGB representation of the desired color for the button. <hr/> <i>font</i> : selects one of the internal fonts to be used for the text data. <hr/> <i>width</i> : button width. <hr/> <i>height</i> : button height. <hr/> <i>stringsize</i> : length of string.	None.
OLED_DrawStringTxt	Displays a string of ASCII text on the screen at the specified position.	<i>col</i> : horizontal start position of the string. <hr/> <i>row</i> : vertical start position of the string. <hr/> <i>font</i> : selects one of the internal fonts to be used for the string data. <hr/> <i>color</i> : 16-bit RGB representation of the desired color for the string data. <hr/> <i>stringsize</i> : length of string. ²	None.

Note:

1. The **vertices* parameter must be included in the OLED_DrawPolygon API.
2. The **string* parameter must be included in the OLED_DrawButtonTxt and OLED_DrawStringTxt APIs.

Customer Support

To share comments, get your technical questions answered, or report issues you may be experiencing with our products, please visit Zilog's Technical Support page at <http://support.zilog.com>.

To learn more about this product, find additional documentation, or to discover other facts about Zilog product offerings, please visit the Zilog Knowledge Base at <http://zilog.com/kb> or consider participating in the Zilog Forum at <http://zilog.com/forum>.

This publication is subject to replacement by a later edition. To determine whether a later edition exists, please visit the Zilog website at <http://www.zilog.com>.



Warning: DO NOT USE THIS PRODUCT IN LIFE SUPPORT SYSTEMS.

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer

©2011 Zilog, Inc. All rights reserved. Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

Z8, Z8 Encore!, Z8 Encore! XP and ZMOTION are trademarks or registered trademarks of Zilog, Inc. All other product or service names are the property of their respective owners.