Application Note

Using the Z8 Encore!® MCU as a Frequency Counter

AN018202-0708

Z8 Encore!® Flash Microcontrollers

Abstract

This application note describes how Zilog's Z8 Encore!® microcontroller unit (MCU) can be used as a frequency counter to measure frequencies accurately up to 10 kHz. An input frequency is measured using the MCU's built-in timers and the measured frequency value is displayed on a 5x7 LED Matrix, in Hertz.

The Z8 Encore! Flash MCUs have four built-in timers that can be used as interrupt sources, event counters, and to generate pulse-width modulation (PWM). Using these timers, the Z8 Encore! MCU can be used to measure frequency.

Note: *The source code file associated with this application note,* `AN0182-SC01.zip` *is available for download at* www.zilog.com.

Z8 Encore! Flash MCUs

Zilog's Z8 Encore! products are based on the new eZ8 CPU and introduce Flash memory to Zilog's extensive line of 8-bit MCUs. Flash memory in-circuit programming capability allows for faster development time and program changes in the field. The high-performance register-to-register based architecture of the eZ8 core maintains backward compatibility with Zilog's popular Z8® MCU.

Z8 Encore! MCUs combine a 20 MHz core with Flash memory, linear-register SRAM, and an extensive array of on-chip peripherals. These peripherals make the Z8 Encore! MCU suitable for a variety of applications including motor control, security systems, home appliances, personal electronic devices, and sensors.

Developing the Z8 Encore! Frequency Counter

A typical frequency counter is an event counter that logs the number of events over a fixed time interval. Frequency counters have a wide range of application from simple event counting to complex communications.

This application note describes how the Z8 Encore! MCU can be used as a frequency counter. The signal to be measured is fed as an input to the MCU. Any interrupt ports of the Z8 Encore! MCU can be used to receive the signal. The MCU uses the waveform as a source of interrupt whenever a rising edge is detected.

For every interrupt detected, a counter is incremented. The counter continues to be incremented until the built-in timer generates another interrupt. When the timer reaches its terminal count, the value of the counter is stored and the frequency of the signal is displayed on the 5x7 LED matrix display. The displayed frequency is updated every second. The counter value and the timer resets when updating is completed, and counting is resumed. Figure 1 on page 2 displays the input frequency and timer output frequency.

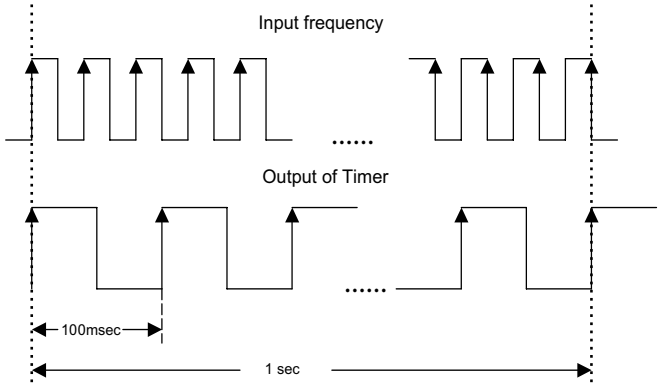

Figure 1. Frequency Counter Input and Output Frequencies

Hardware Architecture

The block diagram of the Z8 Encore!® Frequency Counter is displayed in Figure 2. For details of the hardware implementation of the 5x7 LED Matrix Display, refer to the schematics of the Z8 Encore! Development Board available in the *Z8 Encore!® Flash Microcontroller Development Kit User Manual (UM0146)*.

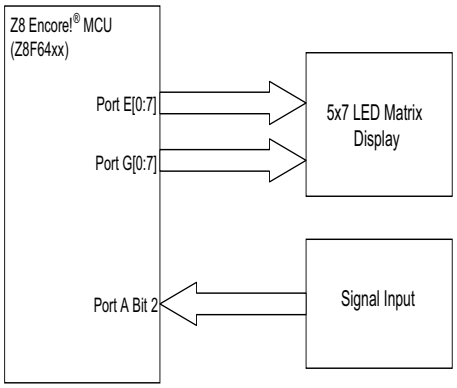

Figure 2. Hardware Architecture of the Z8 Encore! Frequency Counter

To trigger the General-Purpose Input/Output (GPIO) port of the Z8 Encore! MCU, the allowable amplitude for the signal input is from +2.1 V to +3.3 V.

Note: *To avoid damage to the MCU, do not use negative signals.*

Software Implementation

The Z8 Encore! MCU uses two interrupts—an external interrupt from an input signal and an internal interrupt generated by the Timer0 peripheral. The Timer0 interrupt is set to highest priority while the GPIO pin PA2 interrupt (used to sense the input signal) is set to normal priority.

Timer0 is programmed for continuous mode of operation and is configured for a 100 msec time-out. The reload value to generate an output of 100 msec is computed using the following equation.

$$\text{Continuous Mode Time-Out(s)} = \frac{\text{(Reload value x Prescale)}}{\text{System Clock (Hz)}}$$

A Reload value High and Low byte, which equal 38h and 40h respectively, are used with a prescale value of 128 in an 18.432 MHz system clock.

To obtain a 1 sec time-out, a counter is used to count the timer interrupts.

The number of rising edges of the input signal (Port A bit 2) within 1 sec is counted, which is the frequency of the input signal in Hertz. This frequency is displayed on the 5x7 LED Display.

The 5x7 LED Matrix Display board mounted on the Z8 Encore!® Development Board is used to display the measured frequency.

Figure 3 on page 6 displays the flowchart for the main() routine of the Z8 Encore! Frequency Counter. The main() routine initializes the GPIO Port A pin 2 as an input pin to receive the signal input, the GPIO Port E and Port G pins as output pins to the LED display, and initializes Timer0 to generate interrupts. Timer0 is configured to operate in continuous mode. Initially, the LED Display is set to zero; subsequently it displays the updated frequency.

Figure 4 on page 7 displays the flowchart for the Port A interrupt routine. When an interrupt is detected, the counter is incremented until the Timer0 times out to generate an interrupt.

Figure 5 on page 8 displays the flowchart for the Timer0 interrupt routine. The routine waits for the Timer0 interrupt. A delay counter is used to accommodate a 1 sec delay before displaying the measured frequency; the counter is then reset to zero.

Testing

This section provides necessary details for testing the functionality of the Z8 Encore! Frequency Counter. The setup diagram is displayed in Figure 2 on page 2.

For more information on Z8 Encore! Development Kit, refer to *Z8 Encore!® Flash Microcontroller Development Kit User Manual (UM0146)*.

The compiled code, available in the AN0182-SC01.zip file (on www.zilog.com), is downloaded to the Z8 Encore! MCU Flash memory using ZDS II IDE—Z8 Encore! v4.6.1[1].

Equipment Used

The equipment used are as follows:

- The Z8 Encore! Development Kit (Z8ENCORE000ZCO) that includes:
  - Z8 Encore! Development Board (Z8F640x MCU)
  - LED array with four 5x7 LED matrices
  - Smart Cable for PC to Z8 Encore!
  - Universal 5 V DC Power Supply
  - Zilog ZDS II—Z8 Encore! (IDE with ANSI C-Compiler)
- Signal Generator
- Oscilloscope (Agilent 54641D)

Procedure

Follow the steps below to download the code to the Flash memory on the Z8 Encore!® MCU:

1. Set up the connection between the Z8 Encore! Development Board and the Signal Generator as displayed in Figure 2 on page 2. Connect the output of the Signal Generator to Port A bit 2 of the Z8 Encore! MCU.
2. Set the Signal Generator to generate signals at the desired frequency. The frequency must be between 1 Hz to 9999 Hz for near-accurate measurement and display. The signal amplitude must be between +2.1 V to +3.3 V to trig-

1.For details on how to download the code to the Flash memory on the Z8 Encore! MCU, refer to *Zilog Developer Studio II—Z8 Encore!® User Manual (UM0130)*.

ger the Z8 Encore! MCU GPIO pins to generate interrupts.

3. Turn the signal generator ON; the output wave must be a square wave.

Note: *Other waveforms can also be used as inputs.*

4. Read the displayed value on the 5x7 LED Matrix Display on the Z8 Encore! Development Board.

Results

Table 1 lists the actual and measured frequencies for a frequency range of 100 Hz to 10 kHz. The maximum error is 0.05%.

Note: *Frequencies above 10 kHz were not measured because the LED display used for the measurement does not show more than four digits.*

Table 1. Actual and Measured Frequencies of the Z8 Encore! Frequency Counter

Actual Frequency (Hz)	Measured Frequency (Hz)	Percentage Error (%)
100	100	0
1000	1000	0
2000	1999	0.05
3000	2999	0.03
4000	3998	0.05
5000	4998	0.04
6000	5998	0.03
7000	6997	0.04
8000	7996	0.05
9000	8996	0.04
9999	9995	0.04

Summary

This application note demonstrates the capability of the Z8 Encore!® MCU to accurately measure the frequency of input signals up to 10 kHz.

The Z8 Encore! Frequency Counter is simple, and easy to use, with no additional hardware requirements or any calibration procedures. A Z8 Encore! MCU with the code and the signal to be measured is all that is necessary; the signal frequency may be displayed or may be used for further computation if required.

References

The documents associated with Z8 Encore! available on www.zilog.com are provided below:

- Z8 Encore!® Flash Microcontroller Development Kit User Manual (UM0146)
- Z8 Encore! XP® F64XX Series Development Kit User Manual (UM0151)
- Z8 Encore!® F0822 Series Development Kit User Manual (UM0150)
- Z8 Encore! XP® F64XX Series Product Specification (PS0199)
- eZ8 CPU User Manual (UM0128)
- Zilog Developer Studio II—Z8 Encore! User Manual (UM0130)

Appendix A—Flowcharts

This appendix contains the flowcharts associated with the Z8 Encore! Frequency Counter application described in this application note.

The flowchart for the main routine is displayed in Figure 3.

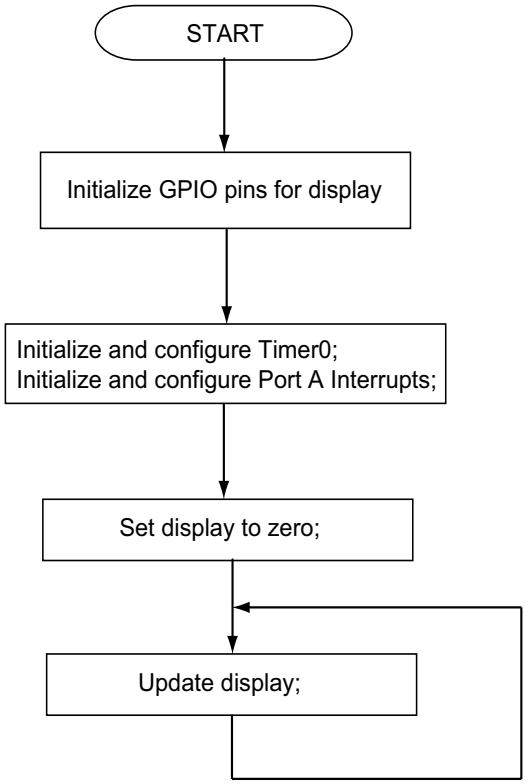

Figure 3. Flowchart for the Main Routine

The flowchart for the Port A interrupt routine is displayed in Figure 4.

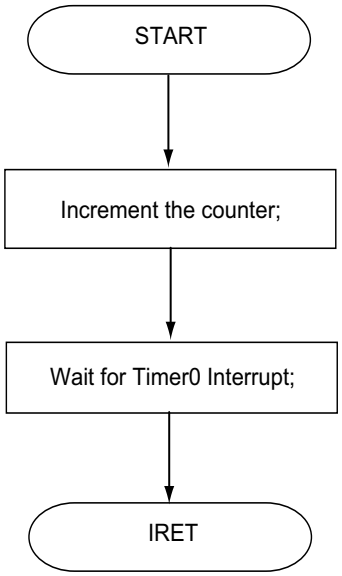

Figure 4. Flowchart for the Port A Interrupt Routine

The flowchart for the Timer0 interrupt routine is displayed in Figure 5.

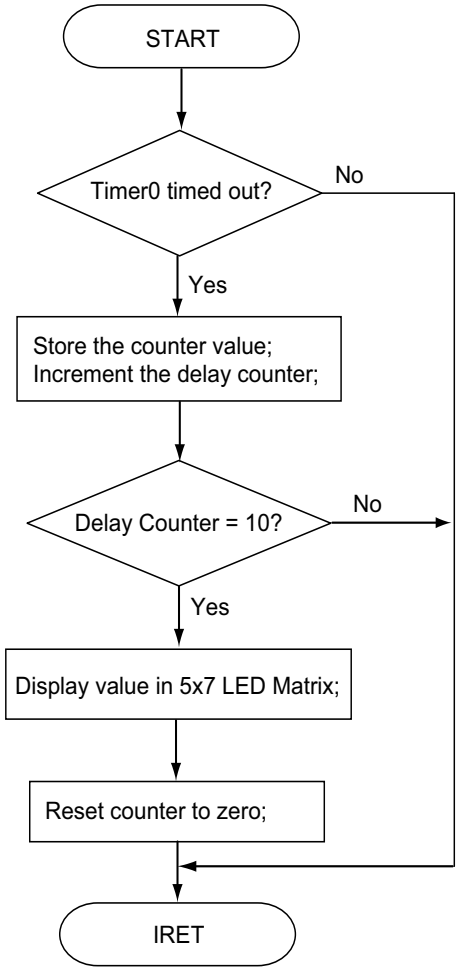

Figure 5. Flowchart for the Timer0 Interrupt Routine

Warning: DO NOT USE IN LIFE SUPPORT

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