Application Note

System Monitor 4-MUX LCD Driver Solution

AN-CM-276

Abstract

This application note describes a simple hardware implementation of a 4-Mux LCD driver using time division multiplexing techniques along with system monitoring. By offloading the LCD driving task to a GreenPAK IC, the MCU frees up its processor from running routine tasks and opens additional GPIO for use in implementing other system requirements.

Contents

Abstract **1**
Contents **2**
Figures **3**
Tables **3**
1 Terms and Definitions **4**
2 References **4**
3 Introduction **5**
4 Creating Drive Signals for LCD Screens **6**
4.1 LCD Driving Techniques: Static Drive and Multiplex Drive 6
4.1.1 Static Drive Technique 6
4.1.2 Multiplex Drive Technique 7
5 GreenPAK Design **9**
5.1 COM Signal Generation 9
5.2 Segment Signal Generation 10
5.2.1 ASM Configuration 10
5.3 2V Voltage Regulator 12
5.4 Additional Features 12
5.5 Modifying the LCD Display 13
6 Testing **13**
6.1 LCD Pattern Display 13
6.2 LCD Reliability 16
7 Design Considerations and Solutions **17**
8 Feature Extension **17**
9 Conclusions **18**
Revision History **19**
Status Definitions **20**
Disclaimer **20**

Figures

Figure 1: LCD Screen 5
Figure 2: Block Diagram 5
Figure 3: Static Drive Technique 6
Figure 4: Voltage Bias Multiplexing Technique 7
Figure 5: Time Division Multiplexing Technique 8
Figure 6: GreenPAK Design 9
Figure 7: COM Signals 10
Figure 8. Segment Waveforms 11
Figure 9: Test Setup with Arduino 14
Figure 10: LCD's Digit and Segment Location 14
Figure 11: DC Offset Measurement 16
Figure 12: Test Setup with DA14585 17

Tables

Table 1: LCD’s COM Signals correspondence with ASM States 10
Table 2: ASM RAM Data Bits 11
Table 3: Example Segment / Common Behavior 12
Table 4: ASM RAM Data 12
Table 5: ASM RAM Registers I2C Addresses 13
Table 6: LCD Pin Map 15
Table 7: ASM and LCD Digits Relation 15
Table 8: Digit 1 Segment Connection with ASM RAM 15
Table 9: Pattern Display 16

1 Terms and Definitions

LCD	Liquid Crystal Display
MUX	Multiplex
I2C	Inter-Integrated Circuit Serial Communication
ASM	Asynchronous State Machine
SEG	LCD's Segment Signals
COM	LCD's Common Signals
IC	Integrated Circuit
GPIO	General Purpose Input Output
MCU	Microcontroller Unit
ACMP	Analog Comparator

2 References

For related documents and software, please visit:

GreenPAK™ Programmable Mixed-Signal Products | Renesas

Download our free GreenPAK Designer software [1] to open the .gp files [2] and view the proposed circuit design. Use the GreenPAK development tools [3] to modify the design into your own customized IC in a matter of minutes. Renesas Electronics provides a complete library of application notes [4] featuring design examples as well as explanations of features and blocks within the Renesas IC.

[1] GreenPAK Designer Software, Software Download and User Guide, Renesas Electronics

[2] AN-CM-276 System Monitor 4-MUX LCD Driver Solution.gp, GreenPAK Design File, Renesas Electronics

[3] GreenPAK Development Tools, GreenPAK Development Tools Webpage, Renesas Electronics

[4] GreenPAK Application Notes, GreenPAK Application Notes Webpage, Renesas Electronics

[5] SLG46537, Datasheet, Renesas Electronics

[6] AN-1092, I2C Controlled State Machine, Application Note, Renesas Electronics

[7] AN1447, Software Driver for 4-Multiplexed LCD with a Standard ST62, Application Note, STMicroelectronics

[8] AN658, LCD Fundamentals and the LCD Driver Module of 8-Bit PIC Microcontrollers, Application Note, Microchip

[9] AN1428, LCD Biasing and Contrast Control Methods, Application Note, Microchip.

[10] Wikipedia Contributors, Liquid-Crystal Display, *Wikipedia The Free Encyclopedia*, Page Version ID: 860435172, 2018

[11] TB1098, Low Power Techniques for LCD Applications, Application Note, Microchip

[12] Segment LCD Displays: Avoiding Excessive DC Component, Journal Article, Focus LCDs

Figure 1: LCD Screen

3 Introduction

Liquid-crystals are commonly used to create simple screens for electronic systems like digital clocks and car audio faceplates.

In an LCD, the liquid-crystal controls whether the segments appear ON or OFF by interacting with the light traveling through the LCD. When a voltage is placed across a segment, the internal molecules align with the electrical field across it and allow light to pass through unobstructed. When no voltage has been placed across the screen, the light travels through the liquid-crystal and rotates 90˚. In both cases, the light travels through front and rear polarized filters that block / pass the light depending on its polarization.

LCDs are unique such that their light sources can be either passive or active. For passive displays, external lights are used to illuminate and view the characters on the display. Active displays, on the other hand, are designed with an internal light source on the back of the screen.

For a detailed description on how LCDs work, please refer to [8]. This article breaks down both the physical components and the scientific theory behind LCDs. It also describes the benefits and drawbacks of various LCD types including reflective, transmissive, and transflective displays.

This application note improves upon the design described in [8] by offloading many of the processing and hardware requirements from the MCU onto a small and inexpensive GreenPAK IC.

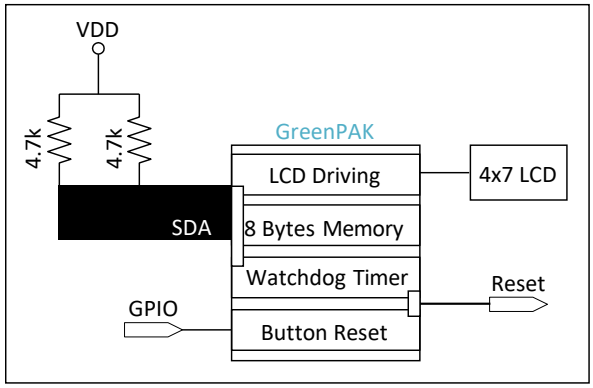

Figure 2: Block Diagram

Figure 2 shows the block diagram of this proposed solution. In addition to the LCD driving circuitry, the GreenPAK solution includes system monitoring features like hardware resets, watchdog timers, and RAM storage.

4 Creating Drive Signals for LCD Screens

LCDs are deceptively similar to LED screens in that they both require a DC voltage to turn on their segments, but the similarities end here. In fact, placing a sustained DC voltage across an LCD segment will damage the liquid-crystal. The following sections detail a few techniques used by designers to maintain an average of zero volts across the individual LCD segments.

4.1 LCD Driving Techniques: Static Drive and Multiplex Drive

When designing drive circuitry for a small LCD like a 7-segment display, there are generally two different categories: Static Drive and Multiplex Drive.

4.1.1 Static Drive Technique

Static Drive LCD segments have two connections: a single common line shared amongst each of the segments and a unique control signal for each segment. To avoid creating a non-zero DC average voltage across a segment, the COM line and the SEGx lines are driven with square waves as shown in Figure 3. The difference between the SEGx lines and the COM line generates a DC voltage across each of the segments without changing the average DC voltage across the segment. To enable the LCD segment, simply invert the square wave driving the specific segment of interest. The average voltage across the LCD segment will still be 0V.

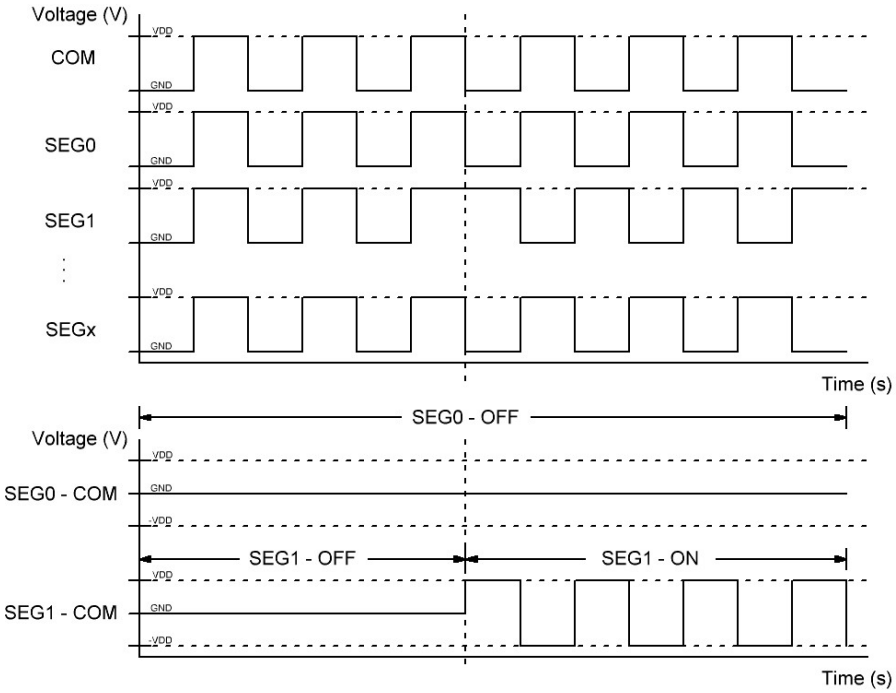

Figure 3: Static Drive Technique

The Static Drive technique is simple and quick for small LCDs. It doesn't require special voltage levels or complicated timing behavior, but the GPIO requirements impact the viability of this technique for high-segment-count displays.

4.1.2 Multiplex Drive Technique

The Multiplex Drive technique reduces the GPIO requirements for driving larger LCDs. The basics of this technique centers around using multiple COM lines for specific SEG lines. Using time domain multiplexing, each segment can be individually enabled on one COM line.

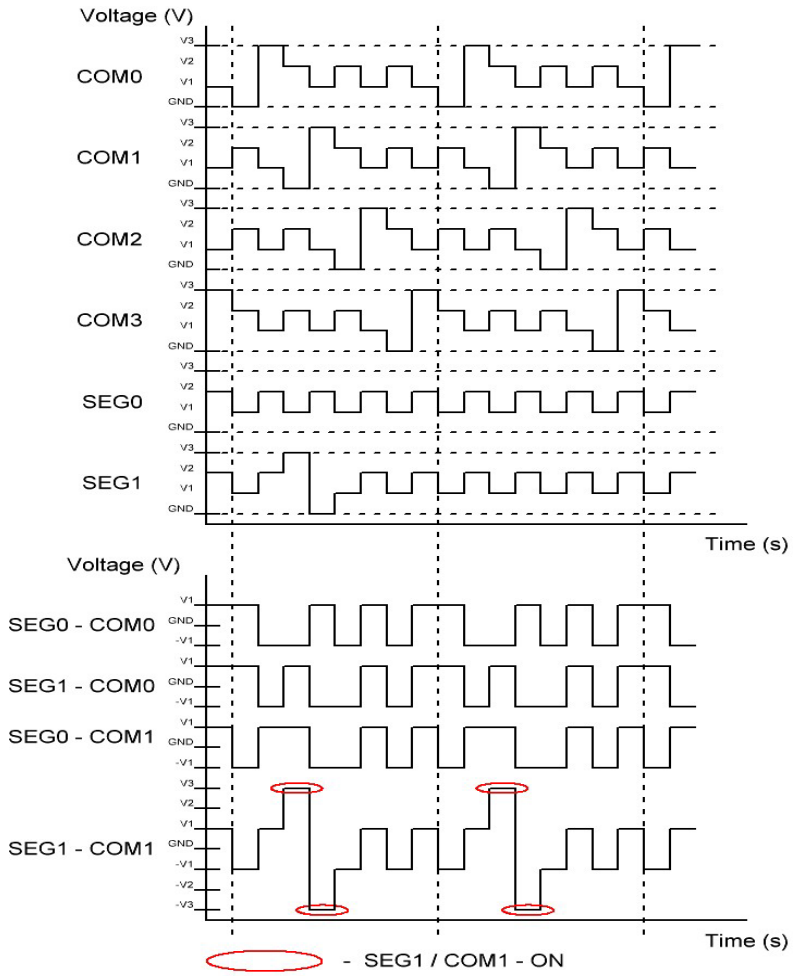

Figure 4: Voltage Bias Multiplexing Technique

Figure 4 shows the first Multiplex Drive technique which uses multiple voltage bias levels to drive each of the SEG and COM lines. Whenever the voltage difference between SEG and COM exceeds V1, that particular SEG/COM pair turns on.

This technique is based on time-division multiplexing of the COM lines. Each of the 4 COM lines are sequentially activated to interact with the SEG lines. During the active period of a COM line, the SEG lines can be inverted and driven to the opposite rails to enable the SEG/COM pair.

When averaging the difference between the SEG and COM lines over one period, the average DC voltage remains at 0 volts. A significant drawback to this technique is that it requires an analog GPO with the capability of producing 4 different output voltage levels.

The second technique, shown in Figure 5, operates off the same time-domain multiplexing principles described in the voltage bias multiplexing architecture. To reduce the number of required output voltage levels, this technique doubles the period to maintain an average of 0V across the LCD segments. Figure 5 shows that SEGx signals in Section B are inverse signals of Section A. Whenever the difference between SEG and COM equals VDD or -VDD, that SEG / COM pair turns on. This technique assumes the magnitude of the segment's turn on voltage is between VDD/2 and VDD.

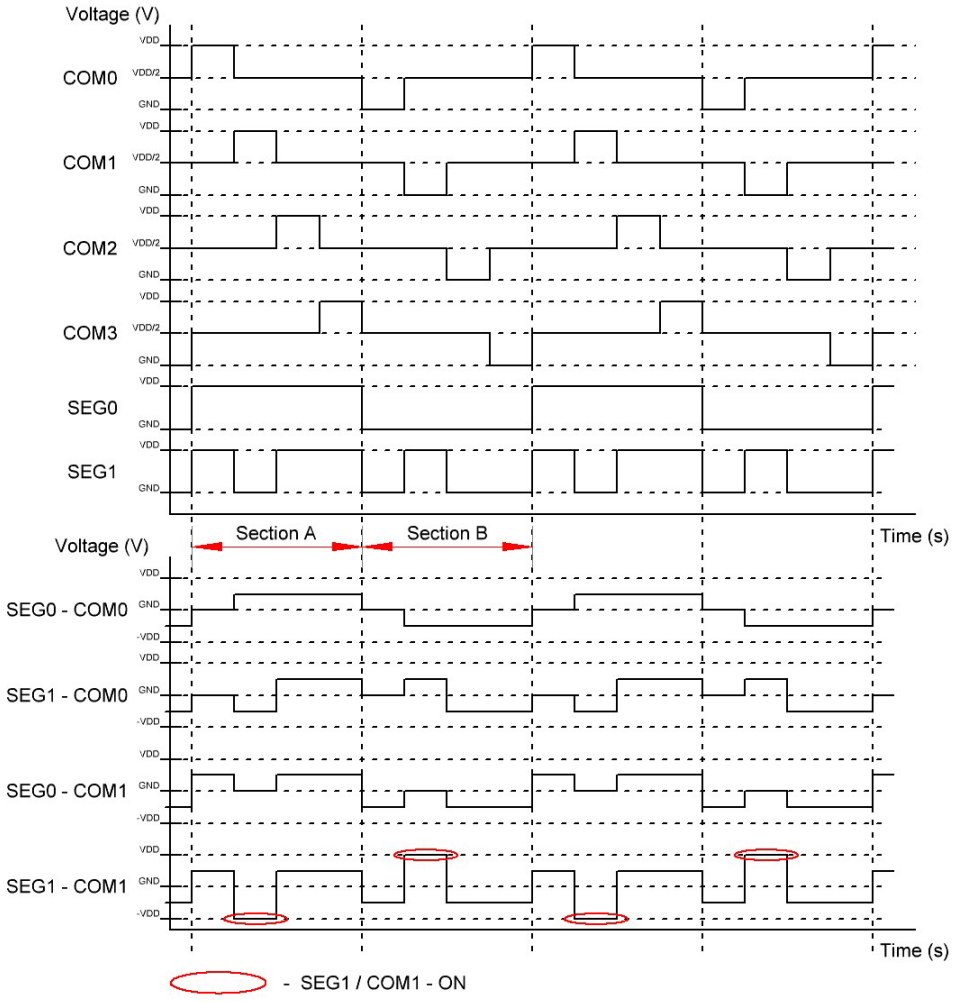

Figure 5: Time Division Multiplexing Technique

This technique takes advantage of the GPIO input structures to generate the VDD/2 voltage level. By connecting the GPIO to a voltage divider from VDD to GND, the GPIO is used to switch between a high-impedance input and a push-pull output to generate the three voltage levels. This technique uses a standard GPIO to drive each of the COM lines.

The GreenPAK design in this application note is based on the second Multiplex Drive technique shown in Figure 5.

5 GreenPAK Design

Figure 6 shows the GreenPAK design which implements the LCD driving architecture along with a 2V regulator, a watchdog timer, a hardware reset, and 8 bytes of RAM into one design. The 2V regulator accomodates for the 1.7V to 2.2V operating voltage range of the LCD.

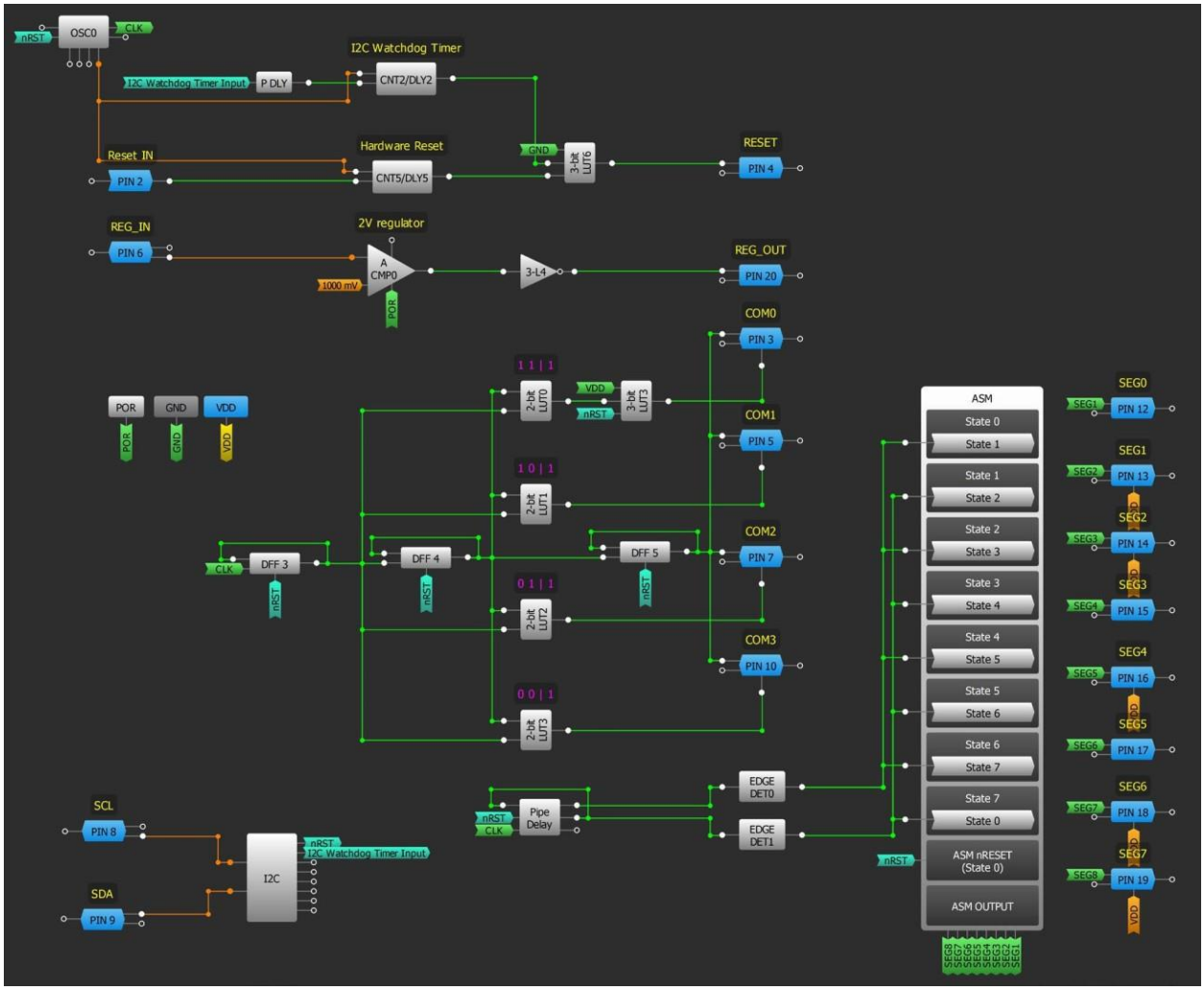

Figure 6: GreenPAK Design

5.1 COM Signal Generation

The COM signals are generated, as previously described, by modifying the GPIO settings and using an external resistive divider set at VDD/2. The GPIO control signals originate from the oscillator-driven DFF chain as shown in Figure 6. These signals determine whether the GPIO is configured as an input or an output and whether the GPIO is HIGH or LOW.

As one can see, the COM signals shown in Figure 7 match the COM signals shown in Figure 5.

Note: VDD = 2.0V, external divider bias set at VDD/2 for High-Z mode

Channel 1 (Yellow) – COM0 (PIN3)
Channel 2 (Green) – COM1 (PIN5)
Channel 3 (Blue) – COM2 (PIN7)
Channel 4 (Magenta) – COM3 (PIN10)

Figure 7: COM Signals

5.2 Segment Signal Generation

5.2.1 ASM Configuration

In this design, the ASM is re-purposed as a fancy pattern generator to drive the SEG outputs. In Figure 6, the Pipe Delay block is used to cycle through the various ASM states.

The GreenPAK ASM consists of 8-states where each LCD COM line corresponds to two ASM states. Table 1 lists the ASM states with their associated COM lines.

Table 1: LCD's COM Signals correspondence with ASM States

COM Signals#	ASM States
COM 0	State 0 and State 4
COM 1	State 1 and State 5
COM 2	State 2 and State 6
COM 3	State 3 and State 7

Each ASM state holds a byte of data in RAM to output to the connection matrix. Table 2 provides a template on how to configure the binary data for each ASM state. Assuming bit a is written to State 0, bit b to State 1, bit c to State 2, and bit d to State 3, the bits in State 4 through State 7 should be inverted to !a, !b, !c, and !d respectively. For example, to turn ON SEG 0 associated with COM 0, write a 0 in the

State 0/SEG 0 location and 1 in the State 4/SEG 0 location. This phenomenon of data inversion correlates with the description in 4.1.2.

Table 2: ASM RAM Data Bits

State Name	Connection Matrix Output RAM							
	SEG 0	SEG 1	SEG 2	SEG 3	SEG 4	SEG 5	SEG 6	SEG 7
State 0	a	a	a	a	a	a	a	a
State 1	b	b	b	b	b	b	b	b
State 2	c	c	c	c	c	c	c	c
State 3	d	d	d	d	d	d	d	d
State 4	!a	!a	!a	!a	!a	!a	!a	!a
State 5	!b	!b	!b	!b	!b	!b	!b	!b
State 6	!c	!c	!c	!c	!c	!c	!c	!c
State 7	!d	!d	!d	!d	!d	!d	!d	!d

Note: VDD = 2.0V, external divider bias set at VDD/2 for High-Z mode

Channel 1 (Yellow) – COM0 (PIN3)
Channel 2 (Green) – COM1 (PIN5)
D0 – SEG0 (PIN12)
D1 – SEG1 (PIN13)
D2 – SEG2 (PIN14)
D3 – SEG3 (PIN15)
D4 – SEG4 (PIN16)
D5 – SEG5 (PIN17)
D6 – SEG6 (PIN18)
D7 – SEG7(PIN19)

Figure 8. Segment Waveforms

An example of SEG/COM behavior is shown in Figure 8. These waveforms show particular COM/SEG pairs enabled in accordance with Table 3. To enable these COM/SEG pairs, the ASM RAM must be configured to the values shown in Table 4 where a “0” in States 0 to 3 and a “1” in States 4 to 7

represents an ON segment. By analyzing Figure 8 one can decipher the state of the LCD segments by subtracting the SEG signals from the COM signals.

Table 3: Example Segment / Common Behavior

	COM0	COM1	COM2	COM3
SEG0	OFF	OFF	OFF	OFF
SEG1	ON	OFF	OFF	OFF
SEG2	OFF	ON	OFF	OFF
SEG3	OFF	OFF	ON	OFF
SEG4	OFF	OFF	OFF	ON
SEG5	OFF	OFF	OFF	OFF
SEG6	OFF	ON	ON	OFF
SEG7	OFF	OFF	OFF	OFF

Table 4: ASM RAM Data

ASM State Name	SEG0	SEG1	SEG2	SEG3	SEG4	SEG5	SEG6	SEG7
State 0	1	0	1	1	1	1	1	1
State 1	1	1	0	1	1	1	0	1
State 2	1	1	1	0	1	1	0	1
State 3	1	1	1	1	0	1	1	1
State 4	0	1	0	0	0	0	0	0
State 5	0	0	1	0	0	0	1	0
State 6	0	0	0	1	0	0	1	0
State 7	0	0	0	0	1	0	0	0

5.3 2V Voltage Regulator

Since the operating voltage range for the LCD is 1.7V to 2.2V, an internal 2V regulator limits the GreenPAK's VDD and determines the logic levels of the COM and SEG signals. This narrow range depends on the specific LCD used in the system. This voltage range, the drive strength of the GreenPAK GPIOs, and the size of the external passive components all impact the contrast of the LCD.

The GreenPAK design uses an ACMP to regulate a 3.3V supply down to 2.0V using a resistive divider with a low side NMOS switch. Passive components limit the current flow and determine the transient behavior of this regulator. ACMP0 shown in Figure 6 regulates the voltage to 2V. Please reference Figure 9 and Figure 12 for a system level implementation of this regulator.

5.4 Additional Features

To synchronize the SEG and COM signals, the I2C virtual input "nRST" will actively reset the design when asserted LOW. When released HIGH, the SEG and COM signals will be synchronized with each other in the time domain as they both operate from the same 25kHz oscillator. For more information on resetting the ASM with I2C, please reference [6].

System monitoring features such as a hardware button reset, a watchdog timer, and RAM storage are also implemented on the GreenPAK solution. The button reset is controlled externally with the help of a GPIO while the watchdog timer and RAM storage are controlled with I2C communication from the MCU.

5.5 Modifying the LCD Display

Enabling and disabling different segments on the LCD is as simple as rewriting the RAM registers within the GreenPAK's ASM. To execute the change properly, we recommend the procedure described below:

1. Assert nRST LOW using the I2C virtual inputs
2. Program the new RAM contents into the ASM
3. Set nRST HIGH to initiate the LCD driving signals

Following the sequence above will guarantee that the COM and SEG pins remain synchronized in the time domain. Table 5 shows the I2C addresses of the ASM states. Data on the LCD is modified by re-writing these ASM RAM registers using I2C.

Table 5: ASM RAM Registers I2C Addresses

ASM State #	I2C Address
State 0	0xD0
State 1	0xD1
State 2	0xD2
State 3	0xD3
State 4	0xD4
State 5	0xD5
State 6	0xD6
State 7	0xD7

6 Testing

A twisted nematic (TN) reflective LCD display (Mouser #: 696-LCD-S401M16KR) is tested using an Arduino Uno. This LCD display is a 4-digit, 7-segment multiplexed display.

6.1 LCD Pattern Display

Figure 9 shows the test schematic used to display different patterns on the multiplexed LCD. The I2C lines (SCL and SDA) are connected to the Arduino's 3.3V output through the R1 and R2 pull-up resistors. Resistors R3 and R4 along with capacitor C2 form the 2V regulator as described in 5.3. R5 through R12 are used to generate bias voltages for the COM signals.

As previously discussed, the LCD contrast depends on both the amplitude and the transient behavior of the LCD drive signals. The sizing of R5 to R12 and the drive strength of the GreenPAK GPIOs impact the transient behavior of the LCD drive signals. Optimal resistance values were determined by iterative bench testing. This external resistor divider network impacts the quiescent current, PCB space, and cost of the system solution. For more information on resistor selection, please refer to [9] and [11].

Different patterns are displayed on the multiplexed LCD by following the steps described in 5.5. The LCD's pin mapping and digit and segment locations are shown in Table 6 and Figure 10 respectively. By correlating Table 6 and Figure 10, it can be determined that pin 5 and pin 6 correspond to Digit 1 of the LCD, pin 7 and pin 8 correspond to Digit 2, pin 9 and pin 10 correspond to Digit 3, and pin 11 and pin 12 correspond to Digit 4. This correlation is shown in Table 7 along with the corresponding ASM outputs.

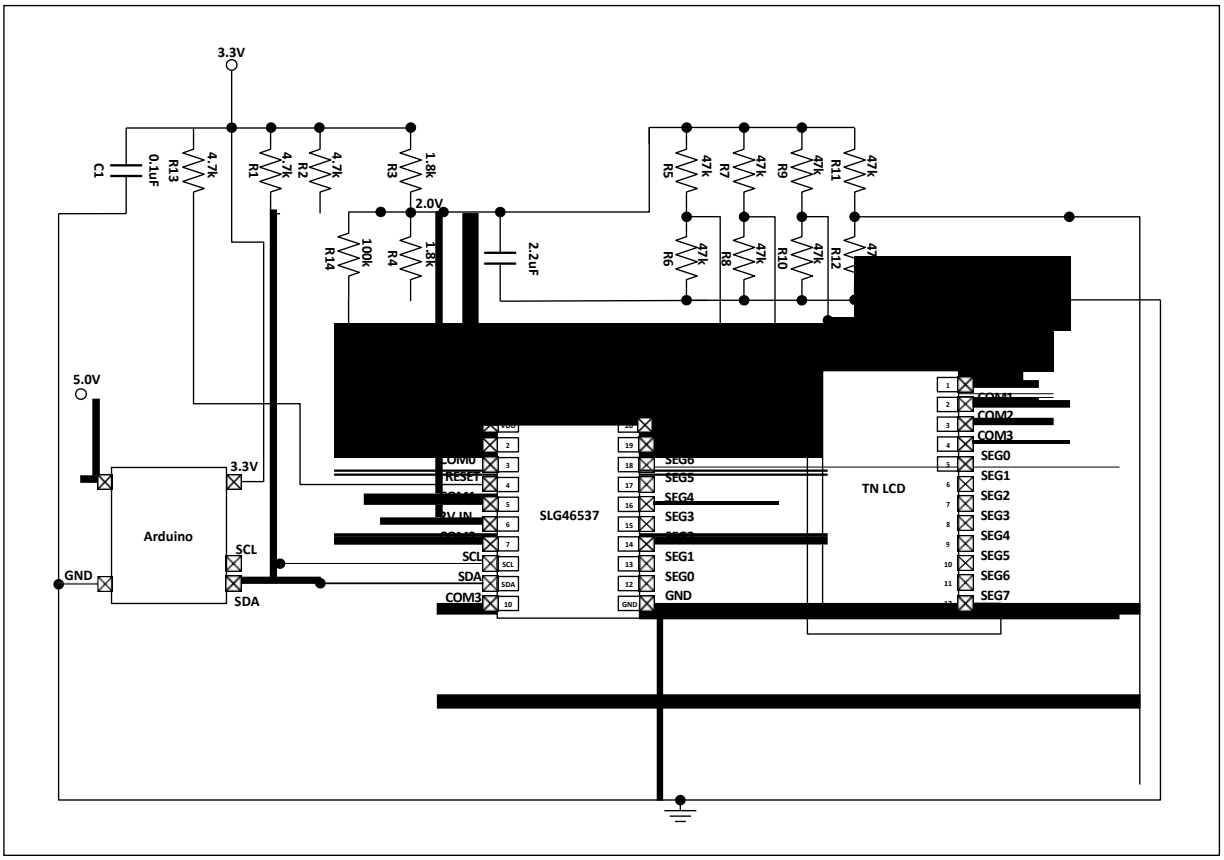

Figure 9: Test Setup with Arduino

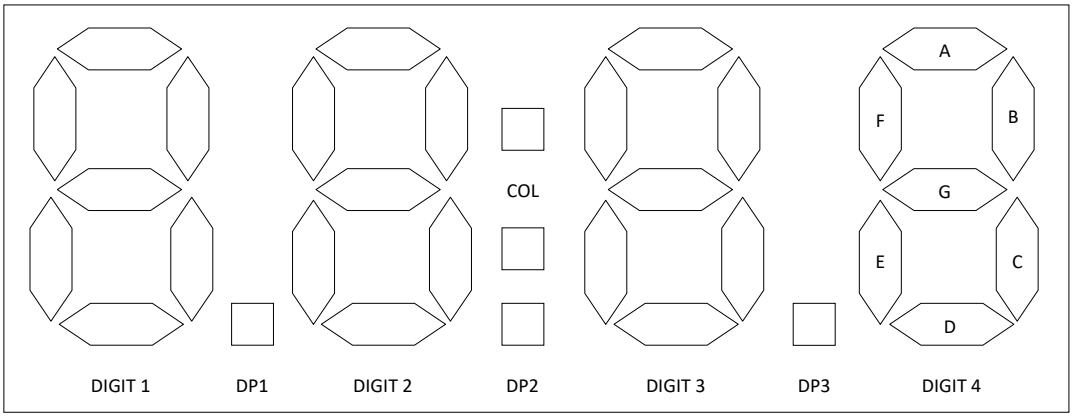

Figure 10: LCD's Digit and Segment Location

Table 6: LCD Pin Map

PIN	1	2	3	4	5	6	7	8	9	10	11	12
COM0	COM0	--	--	--	1D	DP1	2D	DP2	3D	DP3	4D	COL
COM1	--	COM1	--	--	1E	1C	2E	2C	3E	3C	4E	4C
COM2	--	--	COM2	--	1G	1B	2G	2B	3G	3B	4G	4B
COM3	--	--	--	COM3	1F	1A	2F	2A	3F	3A	4F	4A

The SEG 0 and SEG 1 outputs of the ASM control Digit 1 of the LCD. Similarly, SEG 2/3, SEG4/5, and SEG6/7 control Digit 2, 3, and 4 respectively. Table 8 shows the Digit 1 and ASM output connections. Table 1 and Table 8 show that one must write a “0” to State 0/SEG 0 and a “1” to State 4/SEG0 to turn ON segment D on Digit 1. Similarly, other segments can be enabled/disabled by writing respective State x/SEG y locations.

Table 7: ASM and LCD Digits Relation

LCD Digit#	LCD Pin#	ASM Outputs
Digit 1	Pin 5 and Pin 6	SEG 0 and SEG 1
Digit 2	Pin 7 and Pin 8	SEG 2 and SEG 3
Digit 3	Pin 9 and Pin 10	SEG 4 and SEG 5
Digit 4	Pin 11 and Pin 12	SEG 6 and SEG 7

Table 8: Digit 1 Segment Connection with ASM RAM

LCD Digit #	ASM State #	ASM Outputs	
		SEG 0	SEG 1
Digit 1	State 0	D	DP1
	State 1	E	C
	State 2	G	B
	State 3	F	A
	State 4	!D	!DP1
	State 5	!E	!C
	State 6	!G	!B
	State 7	!F	!A

To display a pattern on the LCD, the binary data is first converted into hexadecimal format and then written to the ASM RAM registers. To display “1234” on the LCD, for example, follow the I2C procedure listed in 5.5. The I2C commands for writing data into the ASM RAM registers are as follows:

[Start 0x08 0xF4 0x00 Stop]

[Start 0x08 0xD0 0xEB 0x71 0x01 0x97 0x14 0x8E 0xFE 0x68 Stop]

[Start 0x08 0xF4 0x01 Stop]

Here, 0x08 is the I2C slave address and 0xF4 is the word address for the I2C virtual inputs. There are various I2C slave addresses available in the GreenPAK. For more information on I2C communication please visit [5].

Table 9 shows a few example patterns and the respective hexadecimal bytes required by the ASM RAM.

Some example code for controlling the LCD using an Arduino Uno is included in the ZIP file for this Application Note on Renesas's website.

Table 9: Pattern Display

LCD Pattern	ASM RAM Data (Hex)							
	State 0	State 1	State 2	State 3	State 4	State 5	State 6	State 7
1234	0xEB	0x59	0x01	0x97	0x14	0xA6	0xFE	0x68
95.76	0XB2	0x15	0x98	0x10	0x4D	0xEA	0x67	0xEF
8:07	0x6B	0x43	0x53	0x43	0x94	0xBC	0xAC	0xBC

6.2 LCD Reliability

The LCD reliability is dependent on the average DC offset voltage of the LCD drive signals. The DC offset is measured as the difference between the amplitudes of the COM and SEG signals over time. If the average difference is on the order of 50mV or greater, the DC offset will eventually damage the LCD. For more information on acceptable DC offset specifications, please refer to [12].

Figure 11 displays the test setup used to measure the presence of DC offset in this solution. R13, R14, and C3 create a high impedance measurement interface to monitor the average voltage using a digital multimeter. The resistors chosen for the DC offset measurement are approximately ten times greater than the resistors used in the design to avoid loading the resistive divider network.

The DC offset was measured at about 5mV. As this offset is less than the 50mV offset criteria, the LCD should not be damaged.

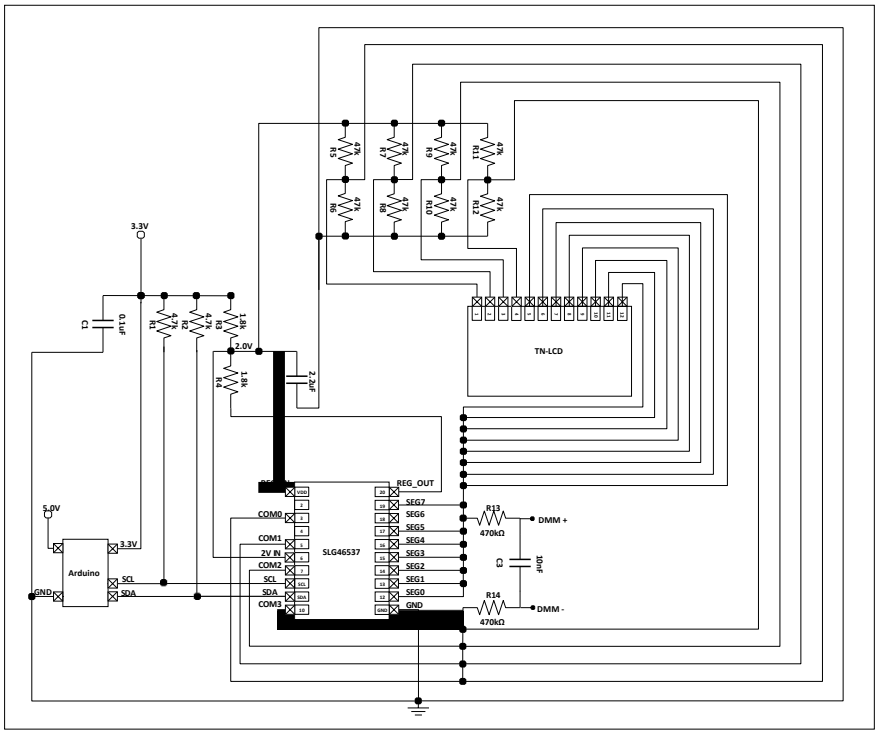

Figure 11: DC Offset Measurement

7 Design Considerations and Solutions

There are certain criteria that should be considered when evaluating this design. These criteria include the operating voltage range of the LCD, the drive strength of the GreenPAK GPIOs, external component selection, and low power optimization.

The operating voltage range for the tested LCD ranges from 1.7V to 2.2V. When operating outside this range, LCD ghosting will occur. This voltage range will vary with the type of LCD.

As shown in Figure 8, the three voltage levels for the COM signals are generated with an external resistor divider network. For optimal performance, we recommend using smaller resistors and stronger GPIO structures to speed up the transient behavior. Many of our GreenPAK GPIOs can be configured as “2x Push-Pull” to increase the drive strength.

There is a trade-off between quiescent current consumption and performance. The smaller external resistors provide better contrast on the LCD but increase the system quiescent current. If current consumption is important for a given application, the user can integrate a low power mode into the GreenPAK design. Using I2C, the MCU can disable the internal oscillator and ACMP when the display isn’t needed. One can also use an external switch to power the voltage regulator. When OFF in low power mode, this switch would guarantee that there is no voltage across the LCD and the resistive divider network. These techniques will limit the overall quiescent current through the device.

8 Feature Extension

The GreenPAK can be controlled with any MCU capable of communicating with I2C commands. DA14585 is a Bluetooth 5.0 SOC-compliant chip capable of interfacing with the GreenPAK and LCD through I2C.

Figure 12 displays a schematic of the test setup with DA14585 chip powered from a 3.0V coin cell battery. P1_0 (Port 1 - Pin 0) and P1_1 (Port 1 – Pin 1) of the DA14585 are I2C pins for SDA and SCL respectively. SDA and SCL lines of the I2C are connected to the 3.0V rail using pull-up resistors R1 and R2. Please follow the description in 5.5 for writing data to the GreenPAK’s ASM RAM registers and I2C virtual inputs.

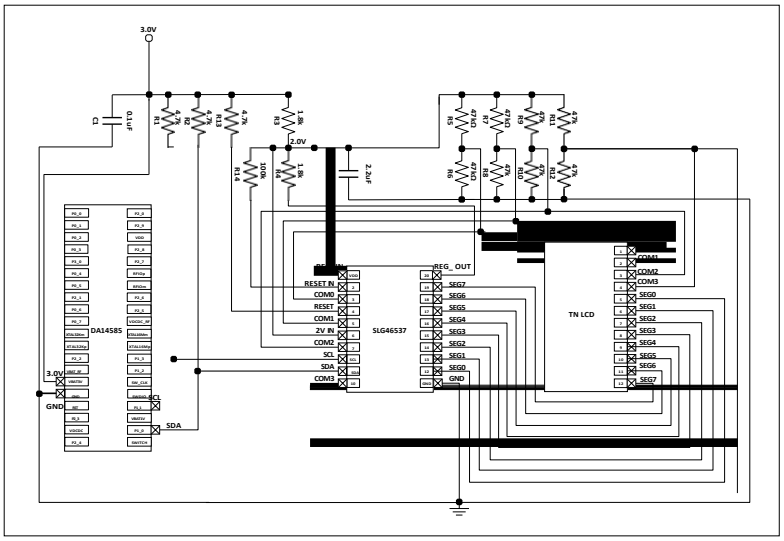

Figure 12: Test Setup with DA14585

Similar system behavior was observed when using the DA14585 chip in place of the Arduino Uno to control the LCD. Some example code for controlling the LCD using the DA14585 is included in the ZIP file for this Application Note on Renesas's website.

9 Conclusions

By using the GreenPAK solution described within this application note, a system designer can decrease MCU pin count, decrease cost, increase system performance, and improve design flexibility. Additionally, battery powered systems might benefit from reduced quiescent current consumption by implementing a low power mode to disable the LCD circuitry.

Often times, MCUs are limited in their functionality based on the number of GPIOs provided to the designer. For smaller LCD applications, offloading LCD driving to the GreenPAK frees up numerous GPIO on the MCU at a low cost for the GreenPAK IC.

Similarly, MCUs are limited in their processing capabilities. Many modern MCUs utilize event-based programming to prioritize tasks in their specific programming environment. The GreenPAK solution allows MCUs to simplify their software requirements by sending a few I2C commands each event. This allows the MCU firmware to either perform other tasks or put the MCU into a low power setting until an event is triggered in the system.

A native benefit of using the GreenPAK IC is the simplicity of testing and modifying designs within the GreenPAK Designer software. At the click of a few buttons, the RAM settings within the ASM and the oscillator clock speed can be changed. Similarly, the watchdog settings and the hardware reset timers can be configured within the IC.

Revision History

Revision	Date	Description
1.0	18-Apr-2019	Initial version.

IMPORTANT NOTICE AND DISCLAIMER

RENESAS ELECTRONICS CORPORATION AND ITS SUBSIDIARIES (“RENESAS”) PROVIDES TECHNICAL SPECIFICATIONS AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT OF THIRD-PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for developers who are designing with Renesas products. You are solely responsible for (1) selecting the appropriate products for your application, (2) designing, validating, and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. Renesas grants you permission to use these resources only to develop an application that uses Renesas products. Other reproduction or use of these resources is strictly prohibited. No license is granted to any other Renesas intellectual property or to any third-party intellectual property. Renesas disclaims responsibility for, and you will fully indemnify Renesas and its representatives against, any claims, damages, costs, losses, or liabilities arising from your use of these resources. Renesas' products are provided only subject to Renesas' Terms and Conditions of Sale or other applicable terms agreed to in writing. No use of any Renesas resources expands or otherwise alters any applicable warranties or warranty disclaimers for these products.

(Disclaimer Rev.1.01 Jan 2024)

Corporate Headquarters

TOYOSU FORESIA, 3-2-24 Toyosu,
Koto-ku, Tokyo 135-0061, Japan
www.renesas.com

Contact Information

For further information on a product, technology, the most up-to-date version of a document, or your nearest sales office, please visit www.renesas.com/contact-us/.

Trademarks

Renesas and the Renesas logo are trademarks of Renesas Electronics Corporation. All trademarks and registered trademarks are the property of their respective owners.