Application Note

Segment LCD with Contrast Control using GPIOs on the SimpleLink CC2340R5 MCU

TEXAS INSTRUMENTS

ABSTRACT

Seven-segment Liquid Crystal Displays (LCDs) are very popular in ultra-low power metering applications such as water and heat meters, and can also be found in several home automation-capable products such as kitchen appliances. They are typically controlled by a dedicated on-chip LCD controller module in the MCU, however this application report explores the capability of adding a resistor circuit to independently control the LCD through software alone. The material covered in this document will prove how the SimpleLink CC2340R5 MCU is capable of achieving this task while also incorporating its on-chip battery monitor to provide real-time contrast control.

The solution provided by this document uses the LP-EM-CC2340R5 EVM which is available for purchase on TI.com and firmware freely provided on the SimpleLink Low Power F3 Demos GitHub. Connection wires, resistors, and an externally sourced LCD will be required to replicate the example. Additional information about operating conditions is provided so that readers may modify the source project in order to suit their specific LCD requirements and further develop the application.

Table of Contents

Trademarks

All trademarks are the property of their respective owners.

1 Introduction

The SimpleLink™ CC2340R5 is a powerful and low-cost Microcontroller Unit (MCU) with 512 kB of flash and 36 or 64 kB of SRAM, featuring a Arm® Cortex®-M0+ and 2.4 GHz radio. Given this feature set it is capable of achieving a multitude of end applications for a variety of radio protocols in a single-chip design. This application report will highlight a single instance to prove the wider possibilities capable with this device.

Liquid Crystal Displays (LCDs) are a type of display using liquid crystals and an optional backlight to create images by modulating light to control individual pixels. Many low-cost home and industrial applications require seven-segment LCD functionality, due to the very low-power, low cost and small size of these displays. All these LCD-based applications can either have the LCD always on or can tolerate the temporarily deactivation of the LCD to save power.

1.1 CC2340R5

The CC2340R family is part of the SimpleLink™ MCU platform, which consists of Wi-Fi®, Bluetooth Low Energy, Thread, Zigbee, Sub-1GHz MCUs, and host MCUs that all share a common, easy-to-use development environment with a single-core software development kit (SDK) and rich tool set. These devices are optimized for low-power wireless communication with Over the Air Download (OAD) support in Building automation (wireless sensors, lighting control, beacons), asset tracking, medical, retail EPOS (electronic point of sale), ESL (electronic shelf), and Personal electronics (toys, HID, stylus pens) markets.

The LP-EM-CC2340R5 development kit is used to speed up development with the SimpleLink Bluetooth Low Energy MCU with support for Bluetooth 5 Low Energy, and 2.4-GHz proprietary protocols. Software support is provided by the SimpleLink Low Power F3 software development kit (SDK) which can be built using the freely offered Code Composer Studio (CCS) IDE. Features include access to all I/O signals with the BoosterPack™ plug-in module connectors and connecting the LaunchPad development kit to your smartphone using TI SimpleLink Connect. The LP-XDS110ET or LP-XDS110 debugger (sold separately) is required for programming, debugging, and RF evaluation.

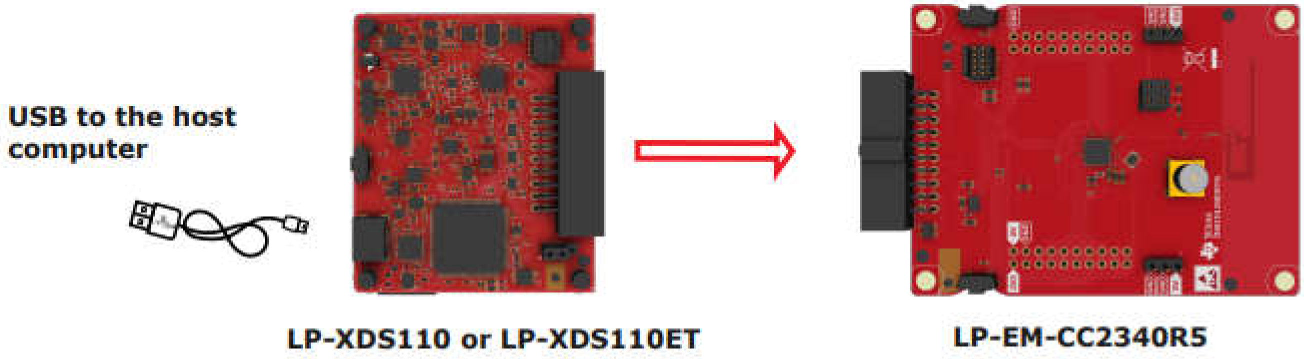
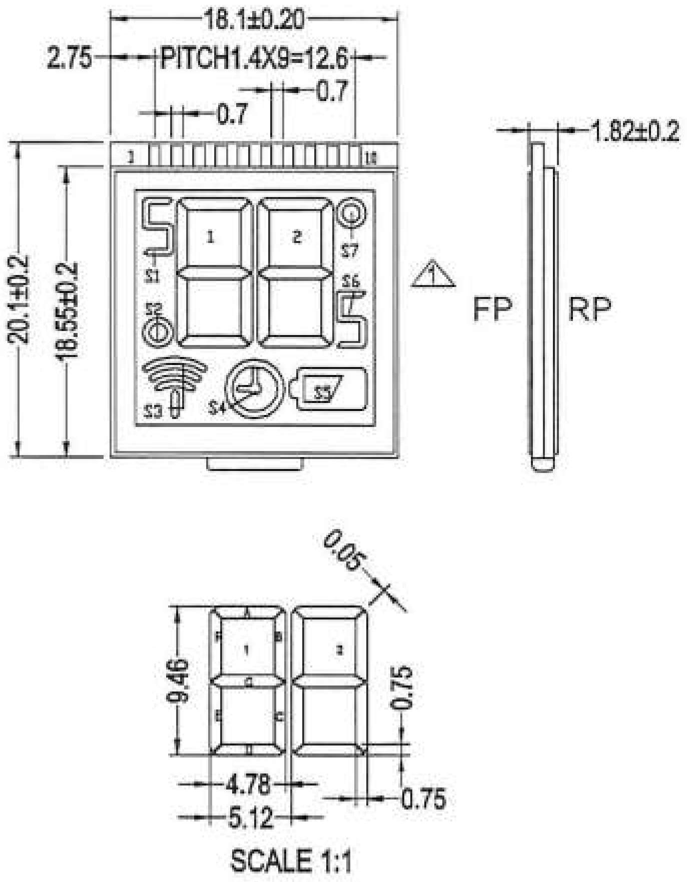


Figure 1-1. LP-XDS110ET and LP-EM-CC2340R5 Connection

1.2 Seven Segment LCD

A 1/3 duty, 1/3 bias LCD with three COM and seven segment lines was selected as the demonstration for this document. As such, there are 21 individual LCD segments which can be controlled on this device including two seven-segment characters and seven single-segment ancillary symbols. Two additional control pins are required for the resistive network, and an additional pin is needed to drive the backlight. Thus the total number of GPIOs for controlling this LCD totals 13, and more would be needed for further LCD segment expansion.



PIN	1	2	3	4	5	6	7	8	9	10
COM1	1F	1A	1B	2F	2A	2B	S2	COM1	/	/
COM2	1E	1G	1C	2E	2G	2C	S4	/	COM2	/
COM3	S1	1D	S7	S6	2D	S5	S3	/	/	COM3

Figure 1-2. Seven-Segment LCD Pin Array

2 Hardware Setup

The following sections will describe the resistive network hardware and CC2340R5 GPIO connections which must be formed so that the LCD may be properly controlled.

2.1 LCD Schematic

The following figure depicts the resistive network that must be created in order to drive the LCD. This can be accomplished on a breadboard, protoboard, or on a PCB design. Not shown is the optional backlight which greatly helps in viewing the segments on the LCD. The multipliers (i.e. “x3” and “x7”) represent the number of times that component must be repeated for the particular “COM” or “SEG” pin.

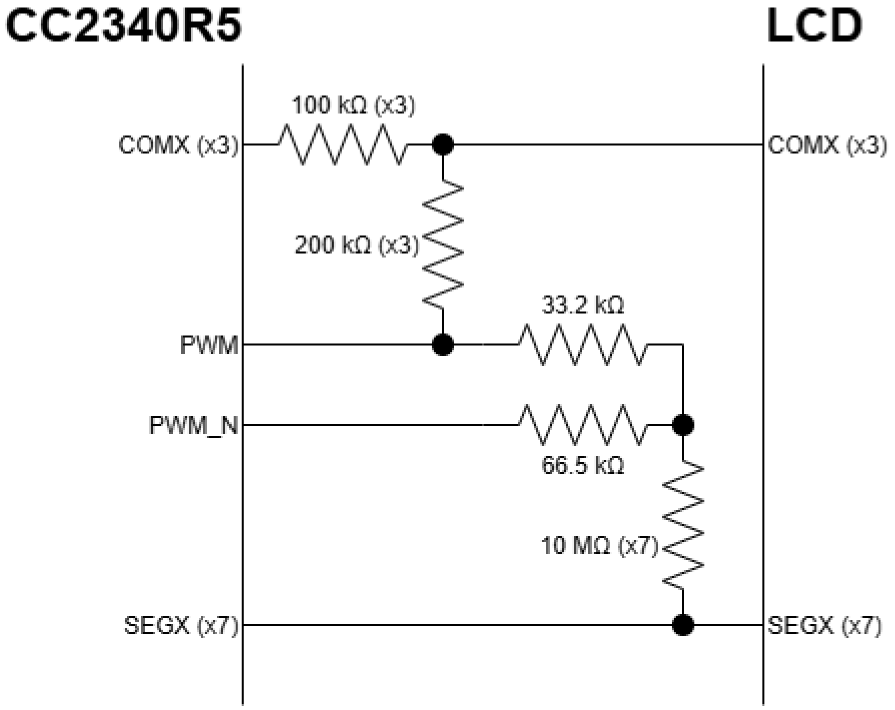


Figure 2-1. DRV8329A Hardware Setup

2.2 CC2340R5 Connection Diagram

The following table shows the connection between the CC2340R5 and the LCD circuit. Each CC2340R5 DIO acts solely as a digital output and can be modified from inside the firmware project SysConfig file’s GPIO module.

Table 2-1. Connections between the CC2340R5 and LCD

CC2340R5 pin	LCD Connection	SysConfig Name
DIO8	COM1	CONFIG_GPIO_COM1
DIO19	COM2	CONFIG_GPIO_COM2
DIO21	COM3	CONFIG_GPIO_COM3
DIO23	SEG1	CONFIG_GPIO_SEG1
DIO25	SEG2	CONFIG_GPIO_SEG2
DIO7	SEG3	CONFIG_GPIO_SEG3
DIO14	SEG4	CONFIG_GPIO_SEG4
DIO6	SEG5	CONFIG_GPIO_SEG5
DIO12	SEG6	CONFIG_GPIO_SEG6
DIO5	SEG7	CONFIG_GPIO_SEG7
DIO11	PWM	CONFIG_GPIO_PWM

Table 2-1. Connections between the CC2340R5 and LCD (continued)

CC2340R5 pin	LCD Connection	SysConfig Name
DIO13	PWM_N	CONFIG_GPIO_PWM_N
DIO15	LED backlight	CONFIG_GPIO_LED_BACKLIGHT

The end result looks similar to the following figure. Notice that the LED header jumpers have been removed from the LP-EM-CC2340R5 as these GPIOs are used for the LCD. The XDS110 is capable of supplying enough power at 3.3V for both the LCD and CC2340R5, however a bench power supply is needed to test the contrast adjustment across the voltage supply range. Two resistors in series with LEDs are used to drive the backlight from a single CC2340R5 GPIO.

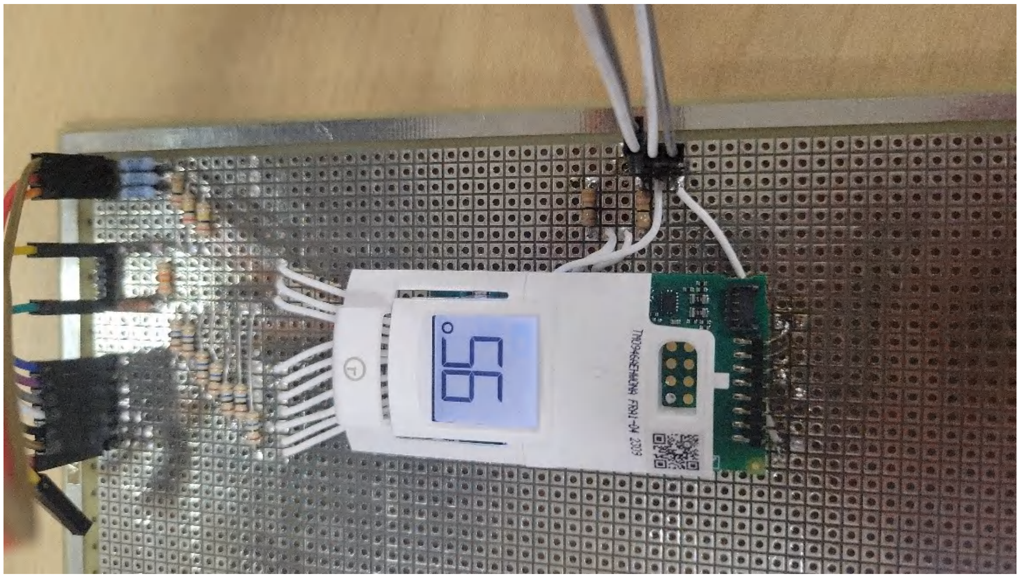


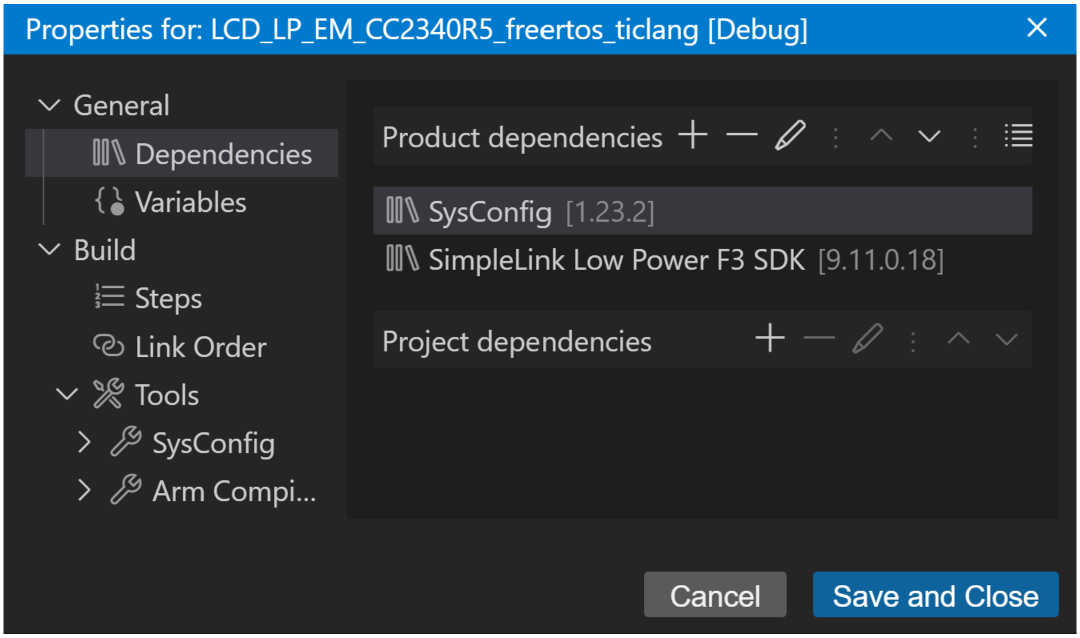
Figure 2-2. Physical Hardware Setup

3 Running the Example

The following sections will discuss the LP-EM-CC2340R5 firmware project dependencies, along with importing the project in CCS, building code, and loading images onto the CC2340R5 device.

3.1 Dependencies

The code project supplied on the SimpleLink Low Power F3 Demos GitHub uses SimpleLink F3 SDK v9.11.0.18, SysConfig v1.23.2, and the TI CLANG v4.0.3 compiler. Ensure that all of these dependencies are installed on your machine before attempting to import the project into Code Composer Studio (CCS) v20 or later. For more examples for setting up your environment, refer to these SimpleLink Academy Labs. Note that users are responsible for migrating and supporting any dependency versions not listed above.



Properties for: LCD_LP_EM_CC2340R5_freertos_ticlang [Debug]

Product dependencies

- SysConfig [1.23.2]
- SimpleLink Low Power F3 SDK [9.11.0.18]

Project dependencies

Figure 3-1. CCS Properties

3.2 Loading Firmware

Projects built inside of CCS can be loaded either directly in this IDE by selecting Run -> Flash Project (Ctrl + F5) or Debug Project (F5). It is recommended to exit Debug Mode to allow free-running if the project is not actively being debugged. Also consider using the Uniflash software tool to load binary images.

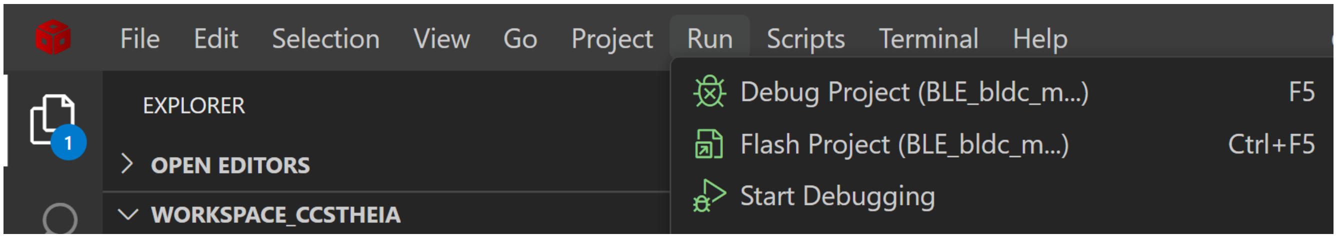


Figure 3-2. CCS Load Options

4 Firmware Design

Now the firmware itself will be further analyzed, pertaining to the CC2340R5 on-chip peripherals and code definitions which are used to modify interactions with the LCD.

4.1 Code Description

All code functions in the example project are implemented in the *lcd.c* file, from which there are the configurable definitions below. The *lcd.syscfg* file is also important for configuring the Device Configuration, TI Drivers, and FREERTOS settings.

Table 4-1. LCD Application Definitions

Definition	Default	Units	Function
LIGHT_STRENGTH	2	Integer	Backlight intensity ranging from zero (0%) to four (100%) in increments of 25%
LIGHT_FREQUENCY	1000	Hz	Backlight frequency
REFRESH_RATE	120	Hz	LCD refresh rate
THRESHOLD_*	200	mV	Voltage threshold for battery monitoring used by the contrast control
NUM_SEGMENTS	7	Integer	Number of segments to display
NUM_DIGITS	10	Integer	Number of digits to display
TEMP_UPDATE	10000000	µs	Time interval for which a temperature measurement will be taken from the on-chip temperature sensor
USE_TEMP	N/A	Boolean	Remove if temperature readings every TEMP_UPDATE are not desired

The main function will initialize all TI drivers, including the on-chip temperature sensor, battery monitor, and GPIOs, as well as the RTOS semaphores and timers necessary for the LCD example to operate. Then further operations pend upon the following software interrupt and hardware callback service routines, all of which will be described in detail through the following sections.

- lightUpdateHandler: backlight PWM control for the LCD
- lcdUpdateHandler: LCD segment and contract control
- tempUpdateHandler: temperature sensor reading
- deltaNotificationFxn: battery monitor threshold updates
- gpioButtonFxn[0/1]: LaunchPad Button

4.2 Timer ISRs

These are the software Interrupt Service Routines (ISRs) which are called upon expiration of a periodic ClockP timer.

4.2.1 lightUpdateHandler

With a frequency of LIGHT_FREQUENCY Hz, this function evaluates the LIGHT_STRENGTH and determines whether to drive CONFIG_GPIO_LED_BACKLIGHT high or low. This is more friendly towards power consumption as compared to a General Purpose Timer (LGPT) Pulse Width Modulation (PWM) which would require the high frequency clock to remain on, thus disallowing standby low-power mode.

4.2.2 lcdUpdateHandler

Taking inspiration from the TIDA-00848 Reference Design, this function uses eight case statements to drive the LCD at a frequency of near REFRESH_RATE Hz (slightly lower due to the the contrast dead time). For each case statement, pins PWM, PWM_N, COM1, COM2, and COM3 are driven in accordance with the LCD waveforms described in the TIDA-00848 Design Guide. Unless adding or removing case statement pairs in accordance with the number of COM lines of your LCD, it is recommended to not changes the write commands of these pins.

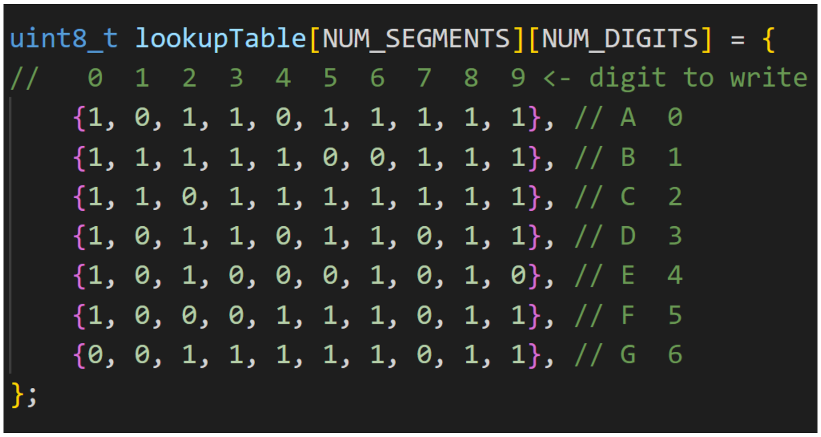
Case 0/1 is used as a “dead time” to drive each LCD segment high for an allotted period of time, which is based on the voltage supply level of the CC2340R5. The purpose of this is to provide contrast control to the LCD across the 2.2 to 3.8 V supply range. At lower voltages, the dead time must be longer to provide extra drive strength to the warranted LCD segments while the voltage is low, but at high voltages the dead time will be minimal as to not overdrive the undesired LCD segments.

Case 2/3 controls the segments driven by COM1, case 4/5 controls COM2, and case 6/7 controls COM3. The pin diagram is provided again for easy reference in Table 3. SEGX pins are driven during the even cases (0, 2, 4, and 6) and off during the odd cases (1, 3, 5, and 7).

Table 4-2. LCD Pin Diagram

PIN	1	2	3	4	5	6	7	8	9	10
COM1	1F	1A	1B	2F	2A	2B	S2	COM1	N/A	N/A
COM2	1E	1G	1C	2E	2G	2C	S4	N/A	COM2	N/A
COM3	S1	1D	S7	S6	2D	S5	S3	N/A	N/A	COM3

The LCD segments labeled “1X” segments control the left seven-segment display, while the “2X” segments control the right seven-segment display. The “SX” segments refer to the ancillary symbols. For purposes of the code demonstration, a look-up table is used to convert a two-digit decimal number into the correct segments used to drive the two seven-segment displays. The ° symbol is also driven on for demonstrative purposes of the on-chip temperature sensor. The look-up table is specific to the LCD being used and will need to be modified in order to accommodate different LCD pin-to-segment mapping configurations.



```
uint8_t lookupTable[NUM_SEGMENTS][NUM_DIGITS] = {
//   0  1  2  3  4  5  6  7  8  9 <- digit to write
    {1, 0, 1, 1, 0, 1, 1, 1, 1, 1}, // A  0
    {1, 1, 1, 1, 1, 0, 0, 1, 1, 1}, // B  1
    {1, 1, 0, 1, 1, 1, 1, 1, 1, 1}, // C  2
    {1, 0, 1, 1, 0, 1, 1, 0, 1, 1}, // D  3
    {1, 0, 1, 0, 0, 0, 1, 0, 1, 0}, // E  4
    {1, 0, 0, 0, 1, 1, 1, 0, 1, 1}, // F  5
    {0, 0, 1, 1, 1, 1, 1, 0, 1, 1}, // G  6
};
```

Figure 4-1. LCD Look-up Table

4.2.3 tempUpdateHandler

If USE_TEMP remains defined, this function will be called every TEMP_UPDATE µs to read the on-chip temperature sensor and return the result in °C, which is then assigned to the two seven-segment displays.

4.3 Hardware Callbacks

These are the callbacks which operate upon a change in one of the CC2340R5’s hardware peripherals.

4.3.1 deltaNotificationFxn

When the batter monitor notices that the voltage supply of the CC2340R5 has increased or decreased by THRESHOLD_DELTA_MILLIVOLT mV since the last reading, this function is used to evaluate the current voltage level, adjust the contrast dead-time of the LCD, and re-register the voltage range for the next notification.

4.3.2 gpioButtonFxn

The LaunchPad’s push buttons are configured to increment the two-digit value of the two seven segment displays on the LCD. BTN_LEFT uses DIO10 and is assigned to gpioButtonFxn0 to increment the “tens” seven segment, while BTN_RIGHT uses DIO9 and is assigned to gpioButtonFxn1 to increment the “ones” seven segment. Rollovers are implemented in the way that would be expected of a two-digit decimal number. Keep in mind that if USE_TEMP remains defined then the display will regardless update every TEMP_UPDATE µs with the current on-chip temperature in °C.

5 Tests and Results

The power consumption of the CC2340R5 LCD solution was measured at 3V using a power analyzer bench tool, and the results are shown in the following table. It goes to follow that different LCD configurations and backlight supplies will result in different power results, and thus users are encouraged to make their own measurements once they have established a working environment.

Table 5-1. LCD and Backlight Power Consumption

LCD (60 Hz)	Backlight (50%)	CC2340R5 Power (µA)
Off	Off	<1
On	Off	3550
Off	On	5750
On	On	7300

A power supply bench tool was also used to observe the contrast adjustment across the acceptable voltage supply range of the CC2340R5, including the minimum of 2.2 V.

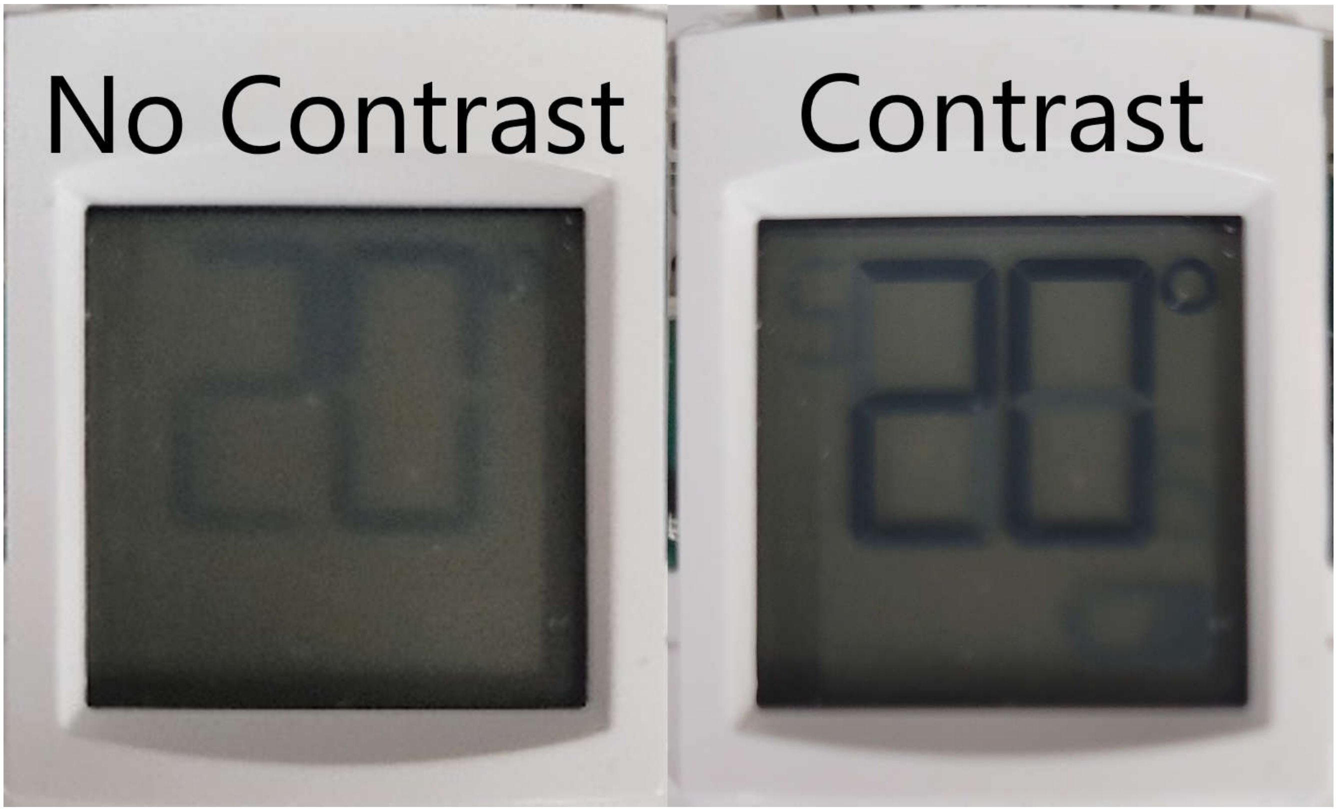


Figure 5-1. LCD Visibility with CC2340R5 Voltage Supply at 2.2V

As before, results will vary by LCD selection and developers are enabled to adjust contrast settings within *lcd.c* to achieve the desired results.

6 Summary

This application report has fully defined a GPIO-driven seven-segment LCD solution with contract adjustment using the SimpleLink CC2340R5. The necessary resistor network, hardware connections and MCU operational code have been described so that users are empowered to operate the out-of-box demonstration. Test results have been provided to confirm the power consumption and contrast output of the implementation. The source code is freely accessible on GitHub and the code flow has been detailed so that developers are familiar with how the project works and are enabled to further modify the project to fit their unique application requirements. Readers are encouraged to post to the E2E forum concerning any additional questions or support needs as it pertains to these resources provided.

7 References

1. Texas Instruments, CC2340R5 Datasheet
2. Texas Instruments, LP-EM-CC2340R5 Quick Start Guide
3. Texas Instruments, Code Composer Studio
4. Texas Instruments, SimpleLink Academy Lab
5. Texas Instruments, TIDA-00848 Reference Design
6. Texas Instruments, SimpleLink Low Power F3 Demos GitHub

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL 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 AND 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 skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you fully indemnify TI and its representatives against any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale, TI’s General Quality Guidelines, or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Last updated 10/2025