**TI Designs**

**Low-Power Wi-Fi Enabled Electronic Paper Display (EPD)**

---

**TI Designs**

TI Designs provide the foundation that you need including methodology, testing and design files to quickly evaluate and customize the system. TI Designs help you accelerate your time to market.

**Design Resources**

- **TIDC-CC3200_EP_Design**
  - Tool Folder Containing Design Files
- **CC3200**
  - Product Folder

**Design Features**

- Low-Power, Battery Powered
- Cloud Connectivity
- Local Control
- Supports 5 Display Sizes
- Supports Custom Graphics
- Broadcast Contents Option

**Featured Applications**

- Electronic Shelf Labels
- Name Tags
- Indoor Signage
- Wearables

---

An IMPORTANT NOTICE at the end of this TI reference design addresses authorized use, intellectual property matters and other important disclaimers and information.
1 Key System Specifications

Table 1 shows the key system specifications.

Table 1. Key System Specifications

<table>
<thead>
<tr>
<th>PARAMETER</th>
<th>SPECIFICATION</th>
<th>DETAILS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating voltage</td>
<td>2.3 V to 3.6 V</td>
<td></td>
</tr>
<tr>
<td>Working environment</td>
<td>Indoor environment. For higher temperature and humidity range, compensation has to be made when writing to the EPD</td>
<td></td>
</tr>
<tr>
<td>Average current</td>
<td>Highly dependant on use-case</td>
<td>See Section 5.3</td>
</tr>
<tr>
<td>Peak current</td>
<td>Achieved during power-ups</td>
<td>CC3200 Datasheet</td>
</tr>
</tbody>
</table>
2 System Description

2.1 Introduction

This design demonstrates the single-chip implementation of a wireless Electronic Paper Display (EPD) using Wi-Fi technology. Technologies that use EPD displays require low-power operation and a method for the user to update the display information. This design provides user control using remote maintenance, an important service in the growing Industrial Internet of things (IoT) market. The design uses the low-power TI SimpleLink Wi-Fi CC3200 wireless MCU, allowing the user to work on two AA batteries for several months, depending on the specific use-case. With an integrated micro-controller and Wi-Fi network processor, this CC3200-based design is a single-chip solution for an EPD display with remote control, locally or over the internet using the MQTT protocol.

3 Design Features

The low-power wireless EPD display allows the user to easily and wirelessly control the contents of any of five different EPD display panels, and demonstrates TI's SimpleLink capabilities of ease of use and low-power. Using either an internal HTTP page embedded in the device, or any MQTT client, the user has the ability to control the contents of the EPD display and configure many parameters. The parameters are then saved and loaded on next power-up. The display accepts text and icons as custom icons, and graphic files may be programmed into the device from a PC.

When remotely controlled, the user has the ability to update any number of EPD panels simultaneously, regardless of their physical location or size by giving them the same name. This use-case is crucial for some electronic shelf label applications, where pricing labels need to be updated at several locations at once, for example.

From a power perspective, this design utilizes one of EPD technology’s main advantages of consuming no power between updates. In between updates, the SimpleLink device is configured to remain connected, allowing for prompt response of the EPD panel, or intermittently connected at a configurable interval. Being connected to the configurable interval allows for an even lower power consumption and longer battery life. CC3200’s Cortex M4 MCU directly drives the EPD display, fast and efficiently, allowing shorter update times compared to other implementations, contributing to the low power consumption of the entire system.

The display is comprised of the following parts:

- A title bar—shows the current operating mode, the device name, and Wi-Fi® reception level.
- An icon, typically 64×64 pixels—either a pre-loaded BMP, or an over-the-air sent graphics.
- Title text—appears next to the icon, typically in a larger font size than the body text. Not available in 1.44” display layout.
- Body text—appears at the bottom. Is typically more than one line long and automatically wraps.
Figure 1 shows the different EPD layouts.

Figure 1. EPD Layout per Display Size
4 Block Diagram

Figure 2 shows a high level block diagram of the system. The hardware is comprised of the standard CC3200 LaunchPad™, the EPD BoosterPack™.

![Figure 2. High Level System Block Diagram](image)

4.1 Highlighted Products

This reference design features the CC3200. The SimpleLink Wi-Fi CC3200 wireless MCU integrates a high-performance Cortex-M4 MCU and peripherals. This design utilizes the CC3200-LAUNCHXL EVM that contains the CC3200 reference design along with emulation and a BoosterPack connector. The CC3200 MCU controls the EPD using an SPI interface. Figure 3 shows the CC3200–LAUNCHXL EVM.

![Figure 3. CC3200–LAUNCHXL EVM](image)
5 System Design Theory

This design introduces two ways of controlling the EPD panel—locally, and remotely.

Locally, the EPD panel is controlled by any connected device on the same network. Remotely, the EPD panel is controlled by any internet connected device, anywhere in the world.

Controlling the EPD panel remotely is more power efficient, as the MQTT broker (server) buffers the data until it is polled by the SimpleLink device. Figure 4 shows the system design theory, case one—remote control and Figure 5 shows the system design theory, case two—local control.

![Figure 4. System Design Theory, Case 1–Remote Control](image1)

![Figure 5. System Design Theory, Case 2–Local Control](image2)
5.1 Hardware

5.1.1 EPD BoosterPack
The EPD BoosterPack is designed to be compatible with the standard BoosterPack connectors of the Launchpad board, utilizing 2 of the 4 rows (to maintain an option for 2×10 connectors instead of the currently 2×20 mounted ones). Table 2 shows the mapping of the signals between the boards.

Table 2. BoosterPack Connector Mapping

<table>
<thead>
<tr>
<th>Pin #</th>
<th>J1 LP Role</th>
<th>J1 BP Role</th>
<th>Description</th>
<th>J1 LP Role</th>
<th>J1 BP Role</th>
<th>Description</th>
<th>J4 BP Role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VBAT</td>
<td>VBAT</td>
<td>GND</td>
<td>GND</td>
<td>GND</td>
<td>For fast charge pumps discharge</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>ANA_IN</td>
<td>BUSY</td>
<td>EPD busy indication</td>
<td>GPIO</td>
<td>DISCHARGE</td>
<td>For fast charge pumps discharge</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>RX</td>
<td>RESET</td>
<td>EPD controller reset</td>
<td>SPI_CS</td>
<td>SPI_CS</td>
<td>SPI interface</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>TX</td>
<td></td>
<td></td>
<td>TEST</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>GPIO</td>
<td>BORDER</td>
<td>For 2.7&quot; displays only</td>
<td>RESET</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>ANA_IN</td>
<td>PWM</td>
<td>For G1 displays only</td>
<td>SPI_MOSI</td>
<td>SPI_MOSI</td>
<td>SPI interface</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>SPI_CLK</td>
<td>SPI_CLK</td>
<td>SPI interface</td>
<td>SPI_MISO</td>
<td>SPI_MISO</td>
<td>SPI interface</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>NWP_LOG</td>
<td>SPI_CLK</td>
<td>SPI interface</td>
<td>SOP2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>SCL</td>
<td></td>
<td>FTDI_TX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>SDA</td>
<td></td>
<td>GND</td>
<td>PANEL_ON</td>
<td></td>
<td>Power switch for EPD</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5.1.2 EPD Panels

The design supports all small display panels from pervasive displays, which are:

• 1.44"
• 1.9"
• 2.0"
• 2.6"
• 2.7"

The supported panel type is the Aurora Mb (V231) G2 by pervasive displays. There is backwards support for Vizplex (V230) or G1 panels using software compilation flags, but it is strongly recommended to only use V231 G2 panels.

EPD panels are sensitive to temperature. The lower the temperature is, the more time is required to drive the image to get similar optical results. This design is aimed for indoor applications, and it is assumed that the EPD panel is operating at room temperature.

In case the temperature drops below room temperature to a point in which there is visible optical degradation, the configurable display quality can be set to optimal quality to compensate for it. The software contains a hook for user temperature sensing function. It is currently statically set for 25 degrees. If required, a function that gets the temperature from the Launchpad’s onboard sensor may be implemented.

NOTE: Compensations for lower temperatures defeat the purpose of using partial update to shorten update cycles.
5.2 Software

The software supplied with this design is based on the SimpleLink CC3200 SDK MQTT example, combined with the pervasive display driver. The software works in a FreeRTOS environment, and has two threads. One for the SimpleLink driver, and the other for the application. Figure 6 shows how the software flow is divided into two main steps—initialization and the main state machine, and Figure 7 shows the software main state machine flow.

![Software Initialization Flow Diagram](image)

Figure 6. Software Initialization Flow
5.2.1 Provisioning

When the system boots (whether from power off or from a hibernate cycle), it will attempt to connect to a known AP. If it fails to find one in a predefined amount of time, the provisioning procedure is activated.

The purpose of this procedure is to connect the system to a new AP and add a profile for later use. When the provisioning procedure starts, a message will be displayed on the screen. Use TI’s provisioning application for iPhone or Android to send the AP credentials to the device.

Once provisioning is complete, a message will be displayed on the screen, and the phone’s provisioning application directs you to the device’s web page. In case provisioning fails or times out, a message is displayed, and the device goes into hibernation to conserve current. The duration of the hibernation is the pre-configured interval, or default value in case it was never configured.
5.2.2 HTTP User Interface

The HTTP page is designed to reflect the currently displayed image on the EPD panel. This is true even when controlled by more than one simultaneously, and updates from one device will propagate to other connected controllers.

The main page allows the user to update the text fields (title and text) and select an icon from a drop-down list. Applying the changes will cause the sign to update.

- Click on the icon sign and load any image if a custom icon is required.

NOTE: The image will be scaled down to 64×64 pixels and is black and white. TI recommends to use an image with the correct size and color pallet to begin with.

- Select the icon number to save the image to the sign, or select volatile in order not to save it when a custom image is uploaded.

Figure 8 shows the HTTP user interface window.

![Figure 8. HTTP User Interface](image)

- Press the three dots icon on the top right corner to open the configuration panel. There you can switch between modes, display types, display qualities, and configure timer intervals. These changes may be saved to the filesystem if required.
5.3 Power

The system is designed to work off 2×AA alkaline batteries, allowing a voltage range of approximately 3-V down to approximately 2.3-V, which is the lowest voltage allowed for the EPD. The system is capable of operating on voltages up to 3.6-V. The EPD driving circuitry is designed to disconnect the EPD (using a FET) whenever it is not in use, and is controlled by the PANEL_ON signal.

The current measurement in this episode are done in two parts (CC3200 and EPD Boosterpack) and are then superimposed.

The detailed analysis of power consumption per mode and element is described in the following subsections. For the lowest power consumption, the intermittently connected profile should be used. With this profile, the longest battery life is achieved.

Table 3 lists the projected battery lifetime for 2xAA alkaline batteries on a 2.7” display (worst-case), for various polling intervals and the percentage of new content available per polling cycle. The breakdown for these figures is detailed in the following chapters.

Table 3. Projected Battery Lifetime for Various Scenarios

<table>
<thead>
<tr>
<th>New Content Available</th>
<th>LIFETIME IN YEARS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Every Update</td>
</tr>
<tr>
<td>Polling every 10 minutes</td>
<td>0.42</td>
</tr>
<tr>
<td>Polling every 15 minutes</td>
<td>0.63</td>
</tr>
<tr>
<td>Polling every 30 minutes</td>
<td>1.23</td>
</tr>
<tr>
<td>Polling every 1 hour</td>
<td>2.36</td>
</tr>
<tr>
<td>Polling every 4 hours</td>
<td>7.5</td>
</tr>
<tr>
<td>Polling twice daily</td>
<td>Over 10</td>
</tr>
<tr>
<td>Polling once daily</td>
<td>Over 10</td>
</tr>
</tbody>
</table>

For example, if the EPD panel is polling the MQTT broker once every 30 minutes. On average, a new message arrives every 2 hours, and the projected battery lifetime would be about 1.7 years.

With a smart scheduler, a combination of polling intervals may be configured. For example, updating every 15 minutes during working ours, and no updates for the rest of the time, with new data coming every second wake-up, will lead to 1.5 years, while still allowing a reasonably short response time.

5.3.1 WLAN Current Consumption

The design incorporates three main working modes, each with its own current consumption profile. The following sections describe these modes in depth.

NOTE: All figures given were measured in a moderately congested open-air environment, on a typical home AP with WPA2 personal security. The current consumption may vary greatly in both directions based on AP type, security type, distance from the AP, air congestion, traffic from peer clients, and more. Please treat these numbers as an estimate.
5.3.1.1 Local Control Mode

In this mode, the control is done locally within the network. This is aimed for home, office network, or maintenance, and is the most power-consuming mode.

To use SimpleLink’s internal HTTP server, the long sleep interval is disabled, and the M4 MCU is kept in low power mode. This is as long as there is no activity except for polling of the device’s status once per second.

Figure 9 shows how loading of the page consumes approximately 49-mA over a period of 1.3 seconds.
An update of the EPD (sending content to the EPD via the local HTTP page, text and icon number only) consumes an average of 9.4-mA over a time of 1563 mSec. Figure 10 shows the WLAN current during an HTTP update.

Figure 10. WLAN Current During an HTTP Update

As seen in Figure 10, these 9.4-mA include the baseline 1.5-mA of the WLAN connection, the transmission and reception of HTTP packets, and the M4 MCU being active to update the EPD. This profile does not include the EPD update itself.
Once the page has been loaded, the browser will poll on the device for any changes to the sign's contents, issued from other devices. Polling occurs once per second and helps keep all browsers in sync with the sign's actual contents. Polling consists of an RX/TX cycle once per second, as well as keeping the M4 MCU active for approximately 100 mSec. The average current in this mode is approximately 12.4-mA for one polling browser. Figure 11 shows the WLAN during HTTP idle polling.

Figure 11. WLAN During HTTP Idle Polling
5.3.1.2 Remote Control Mode or Timer Mode

In this mode, the LSI (Long Sleep Interval) is enabled at an interval of 500-mSec to reduce current consumption while connected to the remote MQTT broker. The HTTP server is disabled in this mode.

While connecting to the remote broker, a keepalive exchange is done every 120 seconds. Averaging the current consumption over 10 minutes shows an average current consumption of slightly less than 1 mA.

An incoming update event from the MQTT broker consumes around 19-mA over a period of 1600 mSec. The update event includes the exchange with the MQTT broker, and the M4 up time for updating the EPD display, but does not include the update of the EPD itself, or the submission of the status update message. Figure 12 shows a single update after 10 minutes of idle connection, showing a summing average of 1.03mA, which gives an estimate for this type of case.

![Figure 12. WLAN Current During an MQTT Update](image-url)
5.3.1.3 Intermittently Connected Mode

This mode consumes the least amount of current. The SimpleLink device remains in hibernate mode between updates for the duration of the configured interval. During this period the EPD is turned off, consumes a negligible amount of current. The WLAN subsystem is also turned off, and the application MCU subsystem is suspended with a timer set for waking it up. The entire system consumes less than 10 μA in this mode (CC3200 + EPD BP). The exact figure varies according to temperature and process, refer to the datasheet for details.

For each hibernate cycle, the MCU wakes up, detects it is out of hibernate mode, begins the WLAN, and waits in power-saving mode until the WLAN is connected to the access point. The MCU then connects to the MQTT broker and waits in power-saving mode for 10 seconds to see if a message is pending. If there are no pending messages, the MCU will disconnect and re-enter hibernate. If there is a pending message, it will be grabbed and the screen will be updated with the new message. According to the settings, the message may be acknowledged (clear retained MQTT message). According to the settings, the message may be acknowledged (clear retained MQTT message). Unacknowledged messages are useful for controlling several clients simultaneously.

If a message was received, a status update will be sent to signal to all sending devices that the display was updated. All current measurements in this design include the acknowledging of the messages (worst-case scenario). Figure 13 shows intermittently connected–polling without an update.

Figure 13. Intermittently Connected–Polling Without an Update

A cycle without an update takes slightly over 10 seconds (as the timer for re-entering hibernate is set for 10 seconds, plus the initialization delay before that timer is set). For such cycles, the EPD panel remains off. Table 4 lists the intermittently connected cycles without an update.

<table>
<thead>
<tr>
<th>EPD TYPE</th>
<th>TIME(S)</th>
<th>AVG. CURRENT (mA)</th>
<th>Q(mC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.7&quot;</td>
<td>10.5</td>
<td>21.7</td>
<td>228</td>
</tr>
</tbody>
</table>

Figure 14 shows the intermittently connected–polling and updating a 2.7” panel.

Figure 14. Intermittently Connected–Polling and Updating a 2.7” Panel
For cycles with an update, the extra current is derived from receiving the message, acknowledging it (when configured to), and sending a status update. The cycles' current figures do not vary much according to display size—only the update time variance between the panel types. The worst case (2.7” panels) measurements are taken for the rest of the current consumption measurements. Table 5 lists the intermittently connected cycles with a 2.7” panel update.

Table 5. Intermittently Connected Cycles With a 2.7” Panel Update

<table>
<thead>
<tr>
<th>EPD TYPE</th>
<th>TIME(S)</th>
<th>AVG. CURRENT (mA)</th>
<th>Q(mC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.7”</td>
<td>10.5</td>
<td>35.5</td>
<td>372.8</td>
</tr>
</tbody>
</table>
5.3.2 EPD Current Consumption

In between updates, the supply rail to the EPD panel is cut by a MOSFET, and consumes a very low amount of power.

To minimize current consumption during updates, a partial update scheme is used, which only changes pixels that require change instead of flashing the entire screen (full update). This allows for a shorter update time, meaning less current wasted by the EPD, as well as by the M4 MCU. The M4 MCU is able to finish the update quicker and go back into power saving mode. The downside of this approach is ghosting of the previous image. The amount of ghosting is derived from the amount of time the update takes. This is user configurable, given in three steps (optimized for power, optimized for quality, or balanced). For each update, the user may force a full refresh to clean previous ghosting when required.

Full refresh is forced during the system initially (since the previous image is not kept), and in intermittently connected mode because the memory is not retained and the previous image is not available for comparison. Using the NVMEM to keep the previous image would save some current, but the system’s lifetime would shorten due to frequent writes to the serial flash.

5.3.2.1 Stage Time Calculations

Stage time is defined as the time it takes to drive the required image. For each iteration, multiple frame drives are required (each one requires a certain amount of frame time). During the frame time, the ink particles are moved toward their destination (black to white, or vice versa). The frame time depends on the display size and the driving MCU.

Pervasive displays recommends driving at least five frames per stage to achieve an acceptable result. The hard coded stage times in the software are set for power optimized values, balanced values and image quality optimized values.

The stage times for full updates are fixed to the values defined by pervasive displays, which are 630-mSec for 2.7” and 2.6” displays, and 480-mSec for the rest of the displays.

For a partial update, the stage time is kept flexible in this design to allow flexibility of display quality v. current consumption. Stage time was measured per display size on CC3200 during partial updates, and different stage times were tried to achieve the stage time per display quality setting and display size. The addition in current per display profile is described in Section 5.2.2.

Table 6 shows the measured stage time per display size. Table 7 shows the hard coded stage times per display size, for each display quality.

<table>
<thead>
<tr>
<th>DISPLAY SIZE</th>
<th>FRAME TIME (mSec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.44”</td>
<td>31</td>
</tr>
<tr>
<td>1.9”</td>
<td>91</td>
</tr>
<tr>
<td>2”</td>
<td>40</td>
</tr>
<tr>
<td>2.6”</td>
<td>117</td>
</tr>
<tr>
<td>2.7”</td>
<td>103</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DISPLAY SIZE</th>
<th>BALANCED</th>
<th>POWER OPTIMIZED</th>
<th>QUALITY OPTIMIZED</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Time (mS)</td>
<td>Frames</td>
<td>Time (mS)</td>
</tr>
<tr>
<td>1.44”</td>
<td>350</td>
<td>12</td>
<td>250</td>
</tr>
<tr>
<td>1.9”</td>
<td>500</td>
<td>6</td>
<td>350</td>
</tr>
<tr>
<td>2.0”</td>
<td>350</td>
<td>9</td>
<td>250</td>
</tr>
<tr>
<td>2.6”</td>
<td>500</td>
<td>5</td>
<td>350</td>
</tr>
<tr>
<td>2.7”</td>
<td>600</td>
<td>6</td>
<td>350</td>
</tr>
</tbody>
</table>
5.3.2.2 **EPD Current Consumption**

EPD boosterpack measurements in this section is completed separately from the CC3200 LaunchPad. Figure 15 shows the 2.7” full update current consumption.

![Figure 15. 2.7” Full Update Current Consumption](image)

Figure 15. 2.7” Full Update Current Consumption

Table 8 lists the full update current consumption per display type.

**Table 8. Full Update Current Consumption per Display Type**

<table>
<thead>
<tr>
<th>EPD TYPE</th>
<th>UPDATE TIME(S)</th>
<th>AVG. CURRENT (mA)</th>
<th>Q(mC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.7&quot;</td>
<td>3.58</td>
<td>1.99</td>
<td>7.12</td>
</tr>
<tr>
<td>2.6&quot;</td>
<td>3.47</td>
<td>2.06</td>
<td>7.14</td>
</tr>
<tr>
<td>2.0&quot;</td>
<td>2.75</td>
<td>2.02</td>
<td>5.55</td>
</tr>
<tr>
<td>1.9&quot;</td>
<td>2.71</td>
<td>2.05</td>
<td>5.55</td>
</tr>
<tr>
<td>1.44&quot;</td>
<td>2.7</td>
<td>2.11</td>
<td>5.69</td>
</tr>
</tbody>
</table>

Table 9 lists the partial update current consumption per display type and display quality.

**Table 9. Partial Update Current Consumption per Display Type and Display Quality**

<table>
<thead>
<tr>
<th>POWER AND QUALITY OPTIMIZED</th>
<th>OPTIMIZED for DISPLAY QUALITY</th>
<th>POWER OPTIMIZED</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Time</td>
<td>Avg Current</td>
</tr>
<tr>
<td>2.7&quot;</td>
<td>1.43</td>
<td>2.1</td>
</tr>
<tr>
<td>2.6&quot;</td>
<td>1.4</td>
<td>2.08</td>
</tr>
<tr>
<td>2.0&quot;</td>
<td>1.15</td>
<td>2.2</td>
</tr>
<tr>
<td>1.9&quot;</td>
<td>1.36</td>
<td>2.08</td>
</tr>
<tr>
<td>1.44&quot;</td>
<td>1.14</td>
<td>2.3</td>
</tr>
</tbody>
</table>
6 Getting Started Hardware

6.1 Hardware
To get started, you are required to have:
• A CC3200 LaunchPad
• An EPD BoosterPack
• One of 5 EPD displays by Pervasive Displays (1.44", 1.9", 2.0", 2.6", or 2.7")

6.2 Power Optimizations
The EPD BoosterPack is designed to be power optimized, thus, no further optimization is required.
The CC3200 Launchpad requires an ECO to make it power optimized. This is not mandatory for running
the application, just for being able to achieve the power figures described here.
The required changes (for LaunchPad Rev 4.1) are:
• Remove R129, R3 and R20
• Remove J12 jumper
• Disconnect the USB cable from the LaunchPad
• Connect the battery positive to J12 pin 1
• Connect the battery negative to J20 pin 3
• Remove R5
• Wire from R5 (the side near the LDO) to Battery +
• Wire J13 (bottom side, near the CC3200 chip) to Battery +
• Remove J6, J7, J8, J9, J10, J11, J2, and J3 jumpers

NOTE: In this configuration, LEDs will not be functional and programming will not be possible. The
reset switch will not work either. This mode is for running the application once its
preprogrammed into the serial flash, and not for debugging.

6.3 Using the Pervasive Displays BoosterPack
This design was made to work with the EPD BoosterPack. If such a board is not available, using the
Pervasive Displays BoosterPack is possible, but some of the features of this design will not be functional.

When using the Pervasive Displays BoosterPack, low-power figures are not achieved, as the BoosterPack
is not optimized for power. In addition, remote modes (remote active and remote low power) may not
function properly, and they are redundant because low power cannot be achieved in this configuration. To
use remote connection, select “always connected” mode instead.

To use the Pervasive Displays BoosterPack, make the following changes to the design software:
1..Undefine #TI_EPD_BOOTERPACK and define #PERVASIVE_DISPLAYS_BOOTERPACK in
   \EPD_Driver\conf_EPD.h and in \EPD_Driver\Pervasive_Displays_Small_EP\EPD_hardware.c
2. In \EPD_Driver\Pervasive_Displays_Small_EP\EPD_hardware.h, change the #define EPD_CS_GPIO
   from 40 to 28
3. In \EPD_Driver\Pervasive_Displays_Small_EP\EPD_hardware.c, empty the functions
   EPD_flash_cs_high and EPD_flash_cs_low (make these functions empty). The design does not use
   the Pervasive Displays BoosterPack’s onboard serial flash to store intermediate images.

In addition, on the LaunchPad, remove jumpers J6, J7, J15 (SoP2), J2, and J3.

Both version 1 and 2 of the Pervasive Displays BoosterPack may be used using this method.
7 Getting Started Firmware

7.1 User Files

The design accepts custom user graphic files. All graphic files must be in Bitmap (.BMP) format, 1 bit (monochrome).

Two user graphic files are supported:

- **Icons**– Load 1 to 10 icons to the design. TI recommends the size of the icons are 64×64 pixels.
- **Splash screen**– This image is displayed when the system initializes from power off, and is removed once it is ready to receive messages. Any resolution up to the limit of the EPD panel is supported, with one restriction. The X aspect must be 32-bits aligned (for example, 32xY, 64xY, 96xY...). This is due to the BMP's format's padding. A small software change may enable 8-bit alignment.

If a missing icon file is requested by the user, a broken image icon (hard coded) will be displayed instead. If a splash screen is missing, a blank image will be displayed instead. Even if no user graphic file is loaded, the application will still run normally. TI recommends to program at least a couple of custom icons to get the best user experience.

Another user file is *display.txt*. This file contains the default display size connected to the system. The contents of this file is a single digit ASCII number of 0 through 4. If this file does not exist, the software will default to 2.7”. This value can be overridden and saved to the file system through the HTTP interface. The user may also switch display sizes while the system is in power-saving mode (between screen updates), since the EPD panel is shut down at those periods. Once the display has been replaced and reconfigured via the HTTP interface, the new display will refresh automatically.

Table 10 lists the display.txt values per display size.

<table>
<thead>
<tr>
<th>DISPLAY TYPE</th>
<th>DISPLAY.TXT VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.44”</td>
<td>0</td>
</tr>
<tr>
<td>2.0”</td>
<td>1</td>
</tr>
<tr>
<td>2.7”</td>
<td>2</td>
</tr>
<tr>
<td>1.9”</td>
<td>3</td>
</tr>
<tr>
<td>2.6”</td>
<td>4</td>
</tr>
</tbody>
</table>

**Warning:** Defining a wrong display size will result in images appearing incorrectly.
7.2 Uniflash

The user files must be added to the existing Uniflash bundle. **Without programming the Uniflash bundle, the application will not work.**

The Uniflash bundle includes the constant icons, application binary, display.txt file, HTTP page and its collaterals, and optional user graphics files—icons and splash screens.

To use, have Uniflash installed on your PC and run it. Load the .ucf file located at the root of the Uniflash files folder. **Figure 16** shows the Uniflash folder example.

![Figure 16. Uniflash Folder Example](image)

1. Determine the COM port number assigned to the LaunchPad by your PC once Uniflash has loaded.
2. Refer to 7.3.1 for details (user logs are output on the same COM port).
3. Make sure the SOP2 jumper on the LaunchPad is assembled to enable programming.
4. Update the COM port field with the correct COM port number if required.
5. If this is the first usage of the LaunchPad, format the serial flash by pressing the format button (approve the default 1-MB serial flash value).
6. Press the reset button on the LaunchPad when prompted.
7. Remove the SOP2 jumper form the LaunchPad once all programming is completed.
8. Press the reset button.
9. The software begins to run.
10. The user is greeted by the splash screen graphics if one was programmed.
Figure 17 shows the Uniflash interface.

![Uniflash Interface](image)

**Figure 17. Uniflash Interface**

### 7.3 MQTT Client

A different build of the MQTT client has to be used with this TI design. It is similar to the one supplied with the SDK of CC3200, but built with some different parameters, to comply with the usage of this application.

Make sure to use this custom build, and not the build from SDK. Otherwise, the application fails in all modes except local control.
7.4 Software

Assuming all required information is present in the SimpleLink’s filesystem (including application binary, display size) upon power-up. The EPD panel displays either the splash screen or clear the screen if the splash screen graphics is not present. Then, the embedded network processor attempts to connect to a pre-configured AP. If none is configured, or the AP is not found, the provisioning procedure starts.

**NOTE:** In case provisioning fails for any reason (such as time out), the application will enter an indefinite hibernate state to conserve power. A message will be displayed on the EPD to instruct the user to reset the device to re-attempt provisioning.

Once AP connection is established, the MCU software enters one of the three supported modes. For each mode, an indication icon is displayed on the top left hand side. The reception indicator is located on the top right side, and the panel’s URN (or name) is displayed in the middle. Table 11 lists the displayed icons’ description.

<table>
<thead>
<tr>
<th>ICON</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>✗</td>
<td>Disconnected from the AP. Provisioning may be required.</td>
</tr>
<tr>
<td>⚫</td>
<td>Local control mode. Control the EPD panel using SimpleLink's internal HTTP server. Fastest response time, highest current consumption. Internet connection is not required in this mode.</td>
</tr>
<tr>
<td>☁</td>
<td>Connected to remote MQTT broker. Fast response time, higher current consumption.</td>
</tr>
<tr>
<td>🌧️</td>
<td>Temporarily always connected. Once the user-configurable timer expires, mode will switch to Intermittently connected. While in this mode, attributes are the same as always connected.</td>
</tr>
<tr>
<td>🌧️</td>
<td>Intermittently connected. Display will update at configured intervals. This mode consumes the least amount of power, but response time is limited to the configured interval time.</td>
</tr>
<tr>
<td>🌪️</td>
<td>Excellent reception (above -45 dB)</td>
</tr>
<tr>
<td>🌪️</td>
<td>Good reception (between -45 dB and 60 dB)</td>
</tr>
<tr>
<td>🌪️</td>
<td>Average reception (between -60 dB and 80 dB)</td>
</tr>
<tr>
<td>🌪️</td>
<td>Poor reception (Below -80 dB)</td>
</tr>
</tbody>
</table>
8 Test Setup

Current measurements in this design were completed in two phases:
1. Measuring CC3200 current
2. Measuring the EPD panel's current

The total current consumption is the superposition of the two.

To measure the CC3200 current, a LaunchPad board is used. A current meter is connected instead of jumper 12. This method isolates the CC3200 (and its serial flash) current consumption from the rest of the board. **Figure 18** shows the measuring current on the CC3200 LaunchPad.

![Figure 18. Measuring Current on the CC3200 LaunchPad](image-url)
For the EPD BoosterPack, a current meter is connected between the LaunchPad's P1.1 (VCC to the BoosterPack) and its counterpart on the BoosterPack. Thus, measuring all current flow to the BoosterPack.

9  Test Data

9.1  **EPD Current Consumption**

EPD current measurements and analyses are described in Section 5.3.2.

9.2  **WLAN Current Consumption**

WLAN current measurements and analyses are described in Section 5.3.1.
10 Design Files

10.1 Schematics
To download the schematics for each board, see the design files at http://www.ti.com/tool/TIDC-CC3200_EPD_DESIGN.
shows the EPD BoosterPack Schematic.

10.2 Bill of Materials
To download the bill of materials (BOM), see the design files at http://www.ti.com/tool/TIDC-CC3200_EPD_DESIGN.

<table>
<thead>
<tr>
<th>ITEM #</th>
<th>DESIGNATOR</th>
<th>QUANTITY</th>
<th>VALUE</th>
<th>PART NUMBER</th>
<th>MANUFACTURER</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>PCB 1</td>
<td>1</td>
<td>EPD_BP</td>
<td></td>
<td>Any</td>
<td>Printed Circuit Board</td>
</tr>
<tr>
<td>2</td>
<td>C1</td>
<td>1</td>
<td>2.2 uF</td>
<td>0805ZD225KAT2A</td>
<td>AVX</td>
<td>CAP, CERM, 2.2 µF, 10 V, +/- 10%, X5R, 0805</td>
</tr>
<tr>
<td>3</td>
<td>C2, C3, C5, C8, C9, C10, C11, C15, C16, C18</td>
<td>10</td>
<td>2.2 uF</td>
<td>0805YD225KAT2A</td>
<td>AVX</td>
<td>CAP, CERM, 2.2 µF, 10 V, +/- 10%, X5R, 0805</td>
</tr>
<tr>
<td>4</td>
<td>C4, C6, C7, C13, C14</td>
<td>5</td>
<td>2.2 uF</td>
<td>C2012X5R1H225K1 25AB</td>
<td>TDK</td>
<td>CAP, CERM, 2.2 µF, 16 V, +/- 10%, X5R, 0805</td>
</tr>
<tr>
<td>5</td>
<td>C12</td>
<td>1</td>
<td>1 uF</td>
<td>08053D105KAT2A</td>
<td>AVX</td>
<td>CAP, CERM, 1 µF, 25 V, +/- 10%, X5R, 0805</td>
</tr>
<tr>
<td>6</td>
<td>C17, C19</td>
<td>2</td>
<td>0.1 uF</td>
<td>06033C104KAT2A</td>
<td>AVX</td>
<td>CAP, CERM, 0.1 µF, 25 V, +/- 10%, X7R, 0603</td>
</tr>
<tr>
<td>7</td>
<td>D1</td>
<td>1</td>
<td>30 V</td>
<td>BAT54SWT1G</td>
<td>Fairchild Semiconductor</td>
<td>Diode, Schottky, 30 V, 0.2 A, SOT-323</td>
</tr>
<tr>
<td>8</td>
<td>FID1, FID2, FID3</td>
<td>3</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Fiducial mark. There is nothing to buy or mount.</td>
</tr>
<tr>
<td>9</td>
<td>J1</td>
<td>1</td>
<td>Right Angle Socket</td>
<td>FH12-40S-0.5SH(55)</td>
<td>Hirose Electric Co. Ltd.</td>
<td>Connector, FFC/FFP Receptacle, 40-Pos, 1 Row, 0.5mm Pitch, SMT</td>
</tr>
<tr>
<td>10</td>
<td>J2, J3</td>
<td>2</td>
<td>SSW-110-22-F-D-VS-K</td>
<td>Samtec</td>
<td>Connector, Receptacle, 100mil, 10-2, Gold plated, SMD</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>Q1, Q2, Q3, Q7</td>
<td>4</td>
<td>60 V</td>
<td>2N7002KW</td>
<td>Fairchild Semiconductor</td>
<td>MOSFET, N-CH, 60 V, 0.31 A, SOT-323</td>
</tr>
<tr>
<td>12</td>
<td>Q4, Q6</td>
<td>2</td>
<td>-50 V</td>
<td>BSS84W-7-F</td>
<td>Diodes Inc.</td>
<td>MOSFET, P-CH, -50 V, -0.13 A, SOT-323</td>
</tr>
<tr>
<td>13</td>
<td>Q5</td>
<td>1</td>
<td>20 V</td>
<td>Si2377EDS-T1-GE3</td>
<td>Vishay-Siliconix</td>
<td>MOSFET, P-CH, 20 V, -4.4 A, SOT-23</td>
</tr>
<tr>
<td>14</td>
<td>R1</td>
<td>1</td>
<td>2.0 k</td>
<td>CRCW08052K00JN EA</td>
<td>Vishay-Dale</td>
<td>RES, 2.0 k, 5%, 0.125 W, 0805</td>
</tr>
<tr>
<td>15</td>
<td>R2, R3, R4</td>
<td>3</td>
<td>100 k</td>
<td>CRCW0805100KJN EA</td>
<td>Vishay-Dale</td>
<td>RES, 100 k, 5%, 0.125 W, 0805</td>
</tr>
</tbody>
</table>
### 10.3 PCB Layout Recommendations

There is no special requirement for layout or routing. The form factor of the board was designed to match that of the LaunchPad beneath it, and the flat-cable connector was placed so that all display sizes will have a mechanical support from the bare-PCB. Mechanical support is also the reason SMT connectors were used.

#### 10.3.1 Layout Prints
To download the layout prints for each board, see the design files at [http://www.ti.com/tool/TIDC-CC3200_EPD_Design](http://www.ti.com/tool/TIDC-CC3200_EPD_Design).

#### 10.4 Altium Project
To download the Altium project files, see the design files at [http://www.ti.com/tool/TIDC-CC3200_EPD_Design](http://www.ti.com/tool/TIDC-CC3200_EPD_Design).

#### 10.5 Assembly Drawings
To download the assembly drawings for each board, see the design files at [http://www.ti.com/tool/TIDC-CC3200_EPD_Design](http://www.ti.com/tool/TIDC-CC3200_EPD_Design).

#### 10.6 Software Files
To download the software files, see the design files at [http://www.ti.com/tool/TIDC-CC3200_EPD_DESIGN](http://www.ti.com/tool/TIDC-CC3200_EPD_DESIGN).

### 11 Terminology

- **WLAN**: Wireless LAN
- **EPD**: Electrophoretic Display
- **AP**: Access Point
About the Author

ASAF CARMELI is a systems engineer at Texas Instruments where he defines hardware modules and software flows for the SimpleLink devices. Asaf brings to this role 10 years in system engineering.
Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

<table>
<thead>
<tr>
<th>Changes from Original (January 2016) to A Revision</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Updated title to Low-Power Wi-Fi Enabled Electronic Paper Display (EPD).</td>
<td>1</td>
</tr>
<tr>
<td>• Updated Introduction section.</td>
<td>3</td>
</tr>
<tr>
<td>• Added Using the Pervasive Displays BoosterPack section.</td>
<td>21</td>
</tr>
</tbody>
</table>
IMPORTANT NOTICE FOR TI REFERENCE DESIGNS

Texas Instruments Incorporated ("TI") reference designs are solely intended to assist designers ("Designer(s)") who are developing systems that incorporate TI products. TI has not conducted any testing other than that specifically described in the published documentation for a particular reference design.

TI's provision of reference designs and any other technical, applications or design advice, quality characterization, reliability data or other information or services does not expand or otherwise alter TI's applicable published warranties or warranty disclaimers for TI products, and no additional obligations or liabilities arise from TI providing such reference designs or other items.

TI reserves the right to make corrections, enhancements, improvements and other changes to its reference designs and other items. Designer understands and agrees that Designer remains responsible for using its independent analysis, evaluation and judgment in designing Designer's systems and products, and has full and exclusive responsibility to assure the safety of its products and compliance of its products (and of all TI products used in or for such Designer's products) with all applicable regulations, laws and other applicable requirements. Designer represents that, with respect to its applications, it has all the necessary expertise to create and implement safeguards that (1) anticipate dangerous consequences of failures, (2) monitor failures and their consequences, and (3) lessen the likelihood of failures that might cause harm and take appropriate actions. Designer agrees that prior to using or distributing any systems that include TI products, Designer will thoroughly test such systems and the functionality of such TI products as used in such systems. Designer may not use any TI products in life-critical medical equipment unless authorized officers of the parties have executed a special contract specifically governing such use. Life-critical medical equipment is medical equipment where failure of such equipment would cause serious bodily injury or death (e.g., life support, pacemakers, defibrillators, heart pumps, neurostimulators, and implantables). Such equipment includes, without limitation, all medical devices identified by the U.S. Food and Drug Administration as Class III devices and equivalent classifications outside the U.S.

Designers are authorized to use, copy and modify any individual TI reference design only in connection with the development of end products that include the TI product(s) identified in that reference design. HOWEVER, NO OTHER LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE TO ANY OTHER TI INTELLECTUAL PROPERTY RIGHT, AND NO LICENSE TO ANY TECHNOLOGY OR INTELLECTUAL PROPERTY RIGHT OF TI OR ANY THIRD PARTY IS GRANTED HEREIN, including but not limited to any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license to use such products or services, or a warranty or endorsement thereof. Use of the reference design or other items described above may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

TI REFERENCE DESIGNS AND OTHER ITEMS DESCRIBED ABOVE ARE PROVIDED "AS IS" AND WITH ALL FAULTS. TI DISCLAIMS ALL OTHER WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING THE REFERENCE DESIGNS OR USE OF THE REFERENCE DESIGNS, INCLUDING BUT NOT LIMITED TO ACCURACY OR COMPLETENESS, TITLE, ANY EPIDEMIC FAILURE WARRANTY AND ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF ANY THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

TI SHALL NOT BE LIABLE FOR AND SHALL NOT DEFEND OR INDEMNIFY DESIGNERS AGAINST ANY CLAIM, INCLUDING BUT NOT LIMITED TO ANY INFRINGEMENT CLAIM THAT RELATES TO OR IS BASED ON ANY COMBINATION OF PRODUCTS AS DESCRIBED IN A TI REFERENCE DESIGN OR OTHERWISE. IN NO EVENT SHALL TI BE LIABLE FOR ANY ACTUAL, DIRECT, SPECIAL, COLLATERAL, INDIRECT, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES IN CONNECTION WITH OR ARISING OUT OF THE REFERENCE DESIGNS OR USE OF THE REFERENCE DESIGNS, AND REGARDLESS OF WHETHER TI HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TI's standard terms of sale for semiconductor products (http://www.ti.com/sc/docs/stdterms.htm) apply to the sale of packaged integrated circuit products. Additional terms may apply to the use or sale of other types of TI products and services. Designer will fully indemnify TI and its representatives against any damages, costs, losses, and/or liabilities arising out of Designer's non-compliance with the terms and provisions of this Notice.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2016, Texas Instruments Incorporated