TI Designs: TIDEP-01005
Human Machine Interface (HMI) for Smart Thermostat Reference Design

Description

This processor-based reference design facilitates a quicker time to market and helps customers design cost-effective, human machine interface (HMI) solutions for Smart Thermostat. The AM335x family of processors provide scalable alternatives with processing speeds from 300 MHz to 1 GHz, graphics acceleration, software to satisfy low- to high-end applications, and ample connectivity with the key peripherals required for Smart Thermostat HMI, such as universal asynchronous receiver/transmitter (UART), SDIO, USB, and so forth.

This Smart Thermostat HMI reference design showcases a two-dimensional (2-D) Qt graphical user interface (GUI), along with TI processor capabilities for hardware- accelerated rendered graphics.

Resources

TIDEP-01005	Design Folder
AM335x	Product Folder
AM335x Starter Kit	Tool Folder
AM335x Processor SDK	Software Folder

TI E2E™ Community

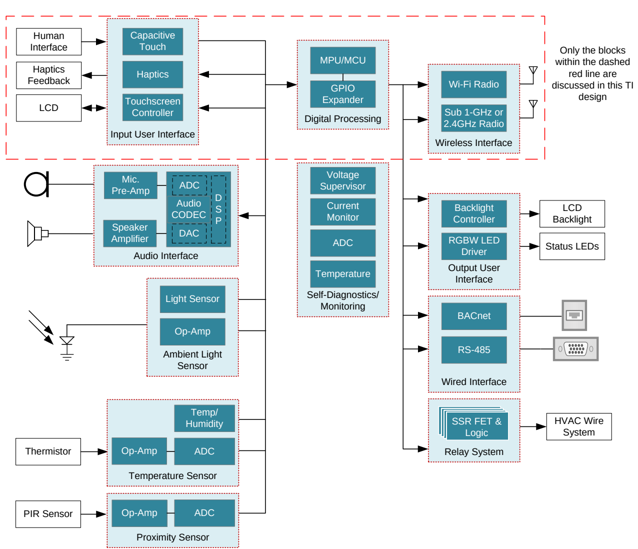
ASK Our E2E Experts

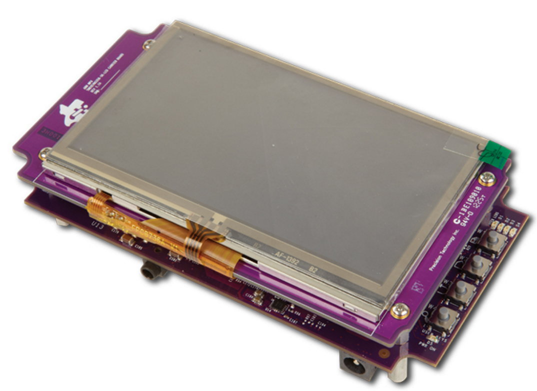
Features

- Demonstrates Smart Thermostat using Hardware-Accelerated Graphics and Qt
- Resistive or Capacitive Touchscreen supported
- 480 × 272 Resolution on Included 4.3-in LCD; Scalable up to 2048 × 2048 on Other Displays
- Built on TI's Processor SDK-Linux for Scalability to Other Sitara™ Processors

Applications

- Smart Thermostat
- HVAC Gateway
- Intrusion Control Panel





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

1 System Description

Network connectivity in the home and in businesses has created the opportunity for all manner of “smart” devices to communicate not only with the Internet, but with each other. A great example of this is the smart thermostat. Increased connectivity has allowed for some very useful features, such as remote programmability through mobile apps, and communication with remote sensors throughout the building. Programmable thermostats are not new, but the ability to program them using an efficient HMI, such as a touch screen or voice commands, rather than with a handful of buttons and arrow icons, has made the experience much easier. Modern thermostats built with more advanced microprocessors also allow for complex customization and the ability to learn the habits of a building’s occupants to conserve energy.

The Sitara™ AM335x processor’s easy-to-use programming tools and portfolio of on-chip capabilities give designers a head start on Smart Thermostat development projects.

The TIDEP-01005 reference design is a quick starting point for customers who want to design a Smart Thermostat HMI module. The solution is also applicable to similar end equipment, such as HVAC Gateway systems and Intrusion Control Panel systems.

The TI AM335x high-performance processors are based on the Arm® Cortex®-A8 core (see Figure 1). These enhanced processors have rich peripheral set and an advanced display capability, including 2-D and 3-D acceleration to help customers design cost-effective Smart Thermostat HMIs. The devices support high-level operating systems (HLOS) such as Linux, which is available free of charge from TI. The AM335x family offers a performance upgrade to systems that are based on lower performance MCUs or processors, and provides enhanced I/O capabilities.

Figure 1. Sitara™ AM335x Chip

The AM335x supports 24-bit, liquid-crystal display (LCD) controllers with resolutions up to 2048 × 2048, which provides scalability from low- to mid-end displays.

The Qt framework is used to develop the GUI for the Smart Thermostat HMI application software. Qt is a cross-platform application framework written in C++. Learn more about Qt at https://www.qt.io.

2 System Overview

2.1 Block Diagram

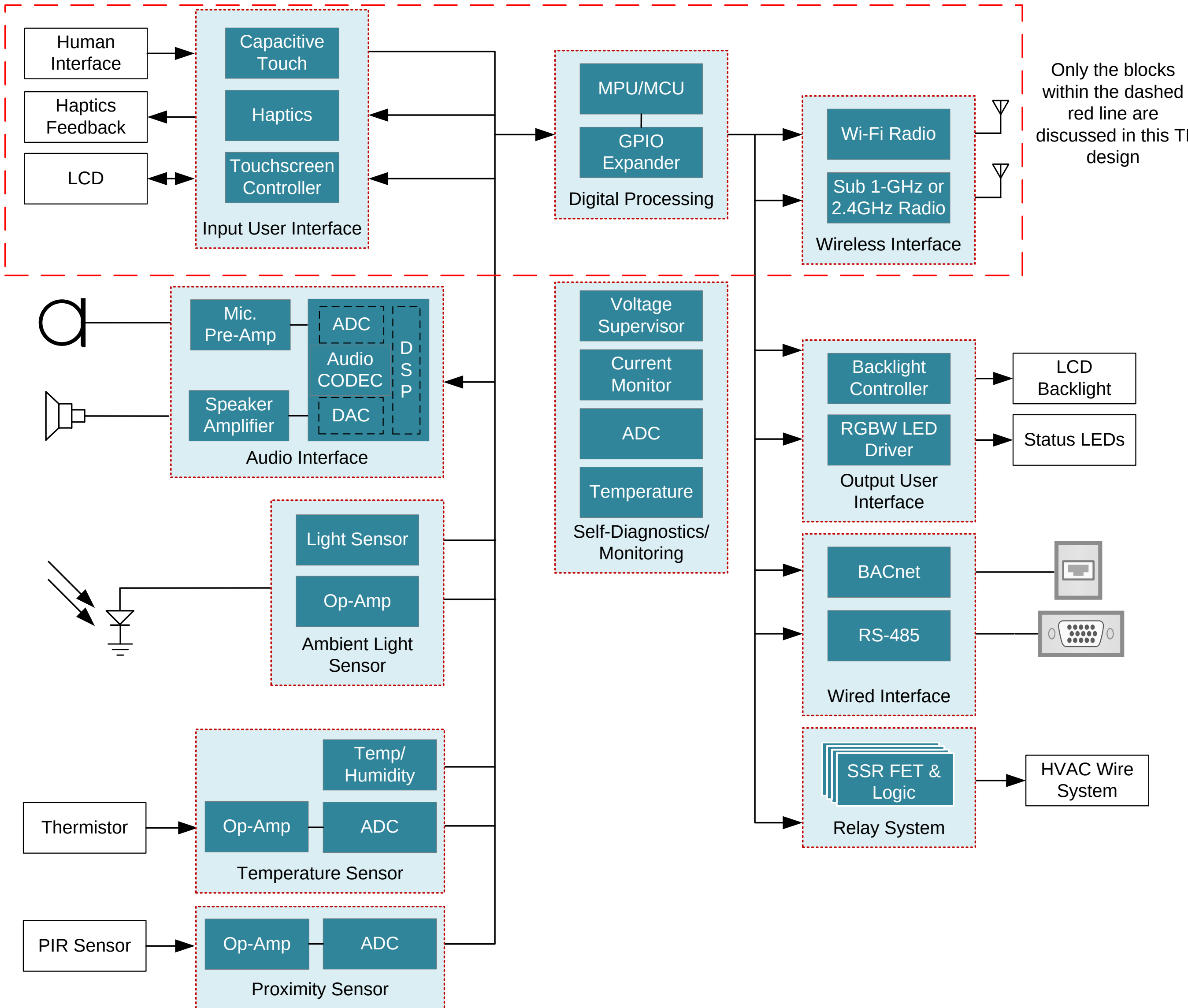


Figure 2. Smart Thermostat HMI Block Diagram

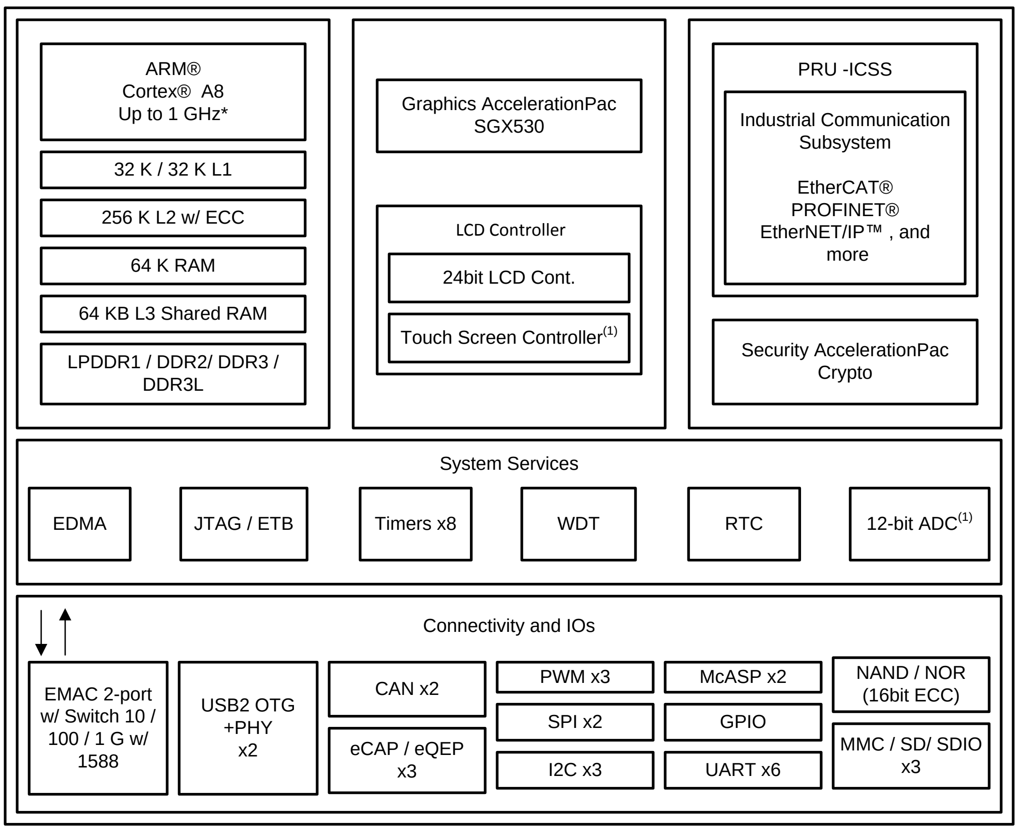
2.2 Highlighted Products

2.2.1 AM335x

The AM335x processors, based on the Arm Cortex-A8 core, are enhanced with image, graphics processing, peripherals, and industrial interface options, such as EtherCAT® and PROFIBUS®.

These devices support HLOS, such as Linux. The AM335x processors contain the subsystems in Figure 3: the microprocessor unit (MPU) subsystem, which is based on the Arm Cortex-A8 core, and the PowerVR SGX530™, which provides 3-D graphics acceleration to support display and gaming effects.

The Programmable Real-Time Unit Subsystem and Industrial Communication SubSystem (PRU-ICSS) is separate from the Arm core and allows independent operation and clocking for greater efficiency and flexibility. The PRU-ICSS enables additional peripheral interfaces and real-time protocols, such as EtherCAT, PROFINET®, EtherNet/IP™, PROFIBUS, Ethernet Powerlink™, Sercos™, and others.



00 MHz / 1 GHz only available on 15 x 15 package. 13 x 13 support up to 600 MHz. Use of TSC will limit available ADC channels.

Figure 3. AM335x Block Diagram

Additionally, the programmable nature of the PRU-ICSS, along with its access to pins, events, and all system-on-chip (SoC) resources, provides flexibility in implementing fast, real-time responses, specialized data handling operations, custom peripheral interfaces, and in offloading tasks from the other processor cores of an SoC.

3 Hardware, Software Requirements, and Demonstration

3.1 Required Hardware and Software

3.1.1 Hardware

The AM335x Starter Kit EVM is required to run the Smart Thermostat HMI demonstration application. The AM335x Starter EVM is a stand-alone test, development, and evaluation module system that enables developers to write software and develop hardware around an AM335x processor subsystem.

See the *AM335x Starter Kit Hardware User's Guide* for instructions on getting started and details on the hardware architecture of the AM335x Starter Kit EVM.

3.1.2 Software

The AM335x Processor SDK for Linux (Processor-SDK-Linux) provides a fundamental software platform for development, deployment, and execution of Linux-based applications. The Smart Thermostat HMI demonstration application source code is integrated in the Processor-SDK-Linux. The example application can be played through the Matrix GUI application launcher. More information on Qt application development using Processor-SDK-Linux can be found in Qt Training: Multipage Resizable Graphical User Interfaces Containing Media and Sitara Linux Training: Hands on with QT.

The Processor-SDK Linux package contains a software user’s guide and additional documentation for setting up and running the Smart Thermostat HMI demonstration application. Download the package from www.ti.com/tool/PROCESSOR-SDK-AM335X.

For the purposes of this design guide, use a Linux host machine for the following instructions. With the required hardware, program the SD card with the Linux processor SDK image using the following steps:

1. Download the SDK installer ti-processor-sdk-am335x-evm-xx.xx.xx.xx-Linux-x86-Install.bin from TI.com (where *xx.xx.xx.xx* is the version number of the latest Linux processor SDK).
2. Create the SD card with default images using the *SDK Create SD Card Script* or see the user's guide.
3. Boot the Linux kernel and file system using the created SD card.
4. The Smart Thermostat demo is found in the Matrix GUI application launcher, as described in the next section.

3.2 Demonstration

3.2.1 Demonstration Setup

This subsection provides details of the test setup with the required hardware and software to run the TI Smart Thermostat HMI software application.

1. Insert the micro SD card created in Section 3.1.2 in the location shown in Figure 4.
2. Insert the 5-V power supply and press the power button shown in Figure 4.

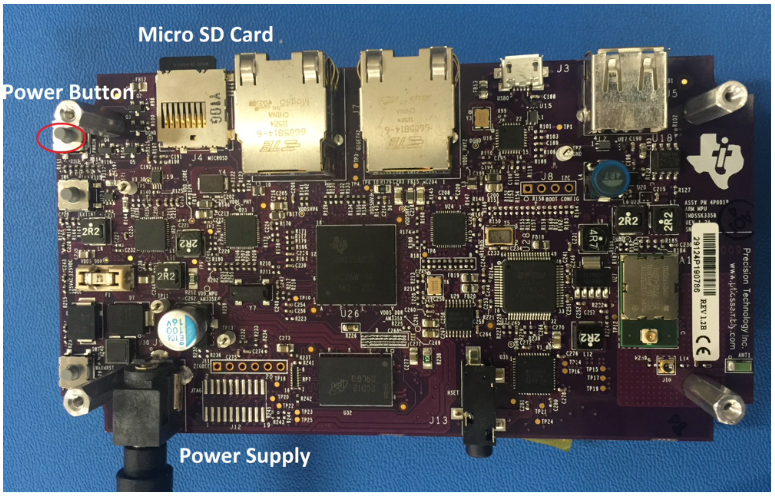


Figure 4. AM335x Starter Kit Setup (LCD Face Down)

3. The board boots up, the Matrix GUI application launches, and the interface shows on the LCD screen (see Figure 5).

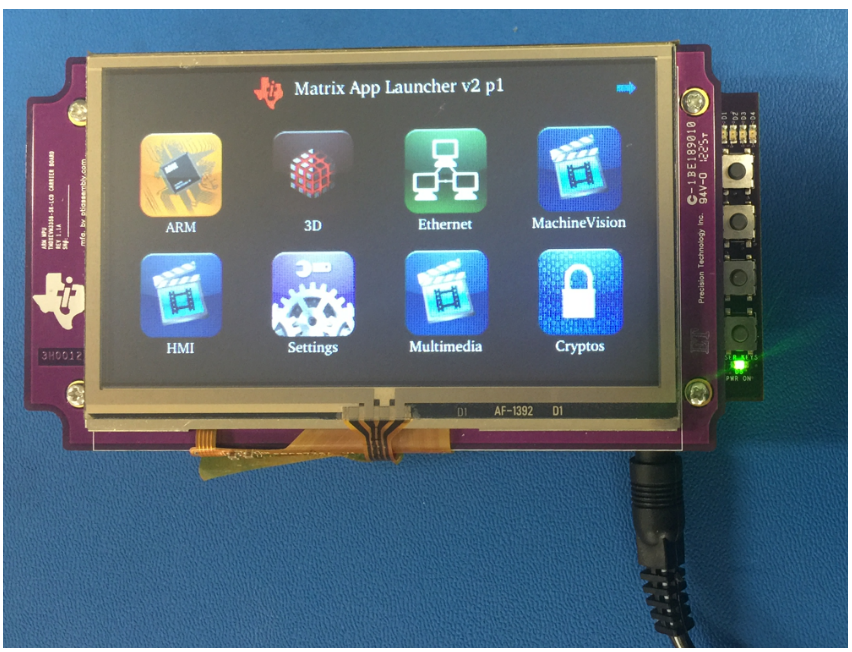


Figure 5. AM335x Starter Kit Setup (LCD Face Up)

3.2.2 Demonstration Procedure

The demonstration procedure is as follows:

1. Click the arrow icon to go to the second page of the Matrix GUI, as shown in Figure 6. Click on the Qt5 icon to navigate to the Qt5-based demonstrations, including the Smart Thermostat HMI.

Figure 6. AM335x Matrix GUI Page 2

2. Figure 7 shows the Qt5 Submenu, which is the next screen that appears. Click on the *Thermostat Demo* icon.

Figure 7. Qt5 Submenu

3. Figure 8 appears on the screen. Click the *Run* button to launch the Smart Thermostat HMI demonstration GUI.

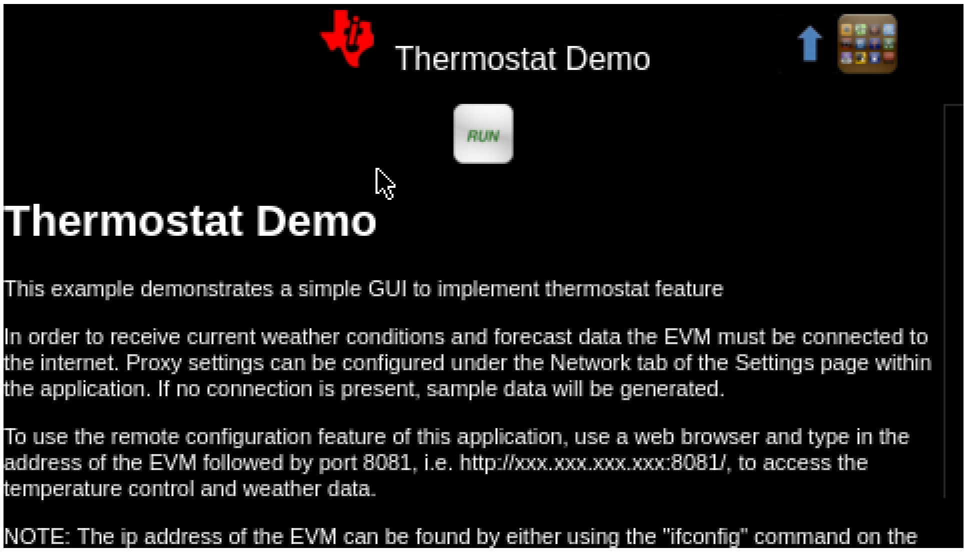


Figure 8. Smart Thermostat HMI Description and Run Screen

4. The Smart Thermostat demonstration GUI launches and the *main* screen shows (see Figure 9).
   - The current temperature, weather, and the three-day weather forecast of the selected city
   - The room temperature and temperature adjustment buttons
   - Three control buttons on the upper left enter the corresponding control screen
     – Setting icon: Various Configuration Options
     – Schedule icon: Weekly Schedule
     – Airplane icon: Energy Saving Mode
   - The purple *Exit* button on the upper right exits out of the demonstration and returns to the Matrix GUI in the previous Figure 5.

NOTE: The *Exit* button on this screen is the only way to exit out of the demonstration GUI and return to the Matrix GUI.

Figure 9. Smart Thermostat Demo GUI: Main Screen

5. The Airplane icon in Figure 9 leads to the energy saving mode screen shown in Figure 10.
   - The return icon in the upper-right corner directs the user back to the *Home* screen in Figure 5.

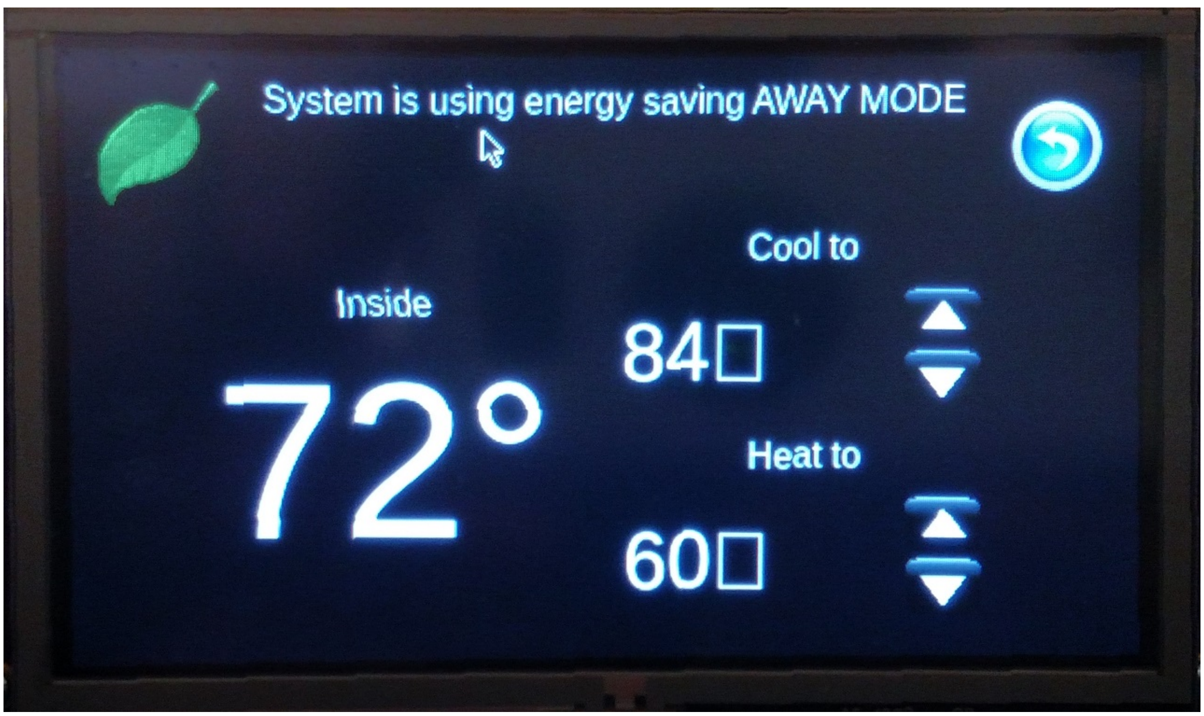


Figure 10. Smart Thermostat Demo GUI: Energy Saving Screen

6. The Scheduler icon in Figure 9 leads to the weekly schedule screen shown in Figure 11.
   - The return icon in the upper-right corner directs the user back to the *Home* screen in Figure 5.

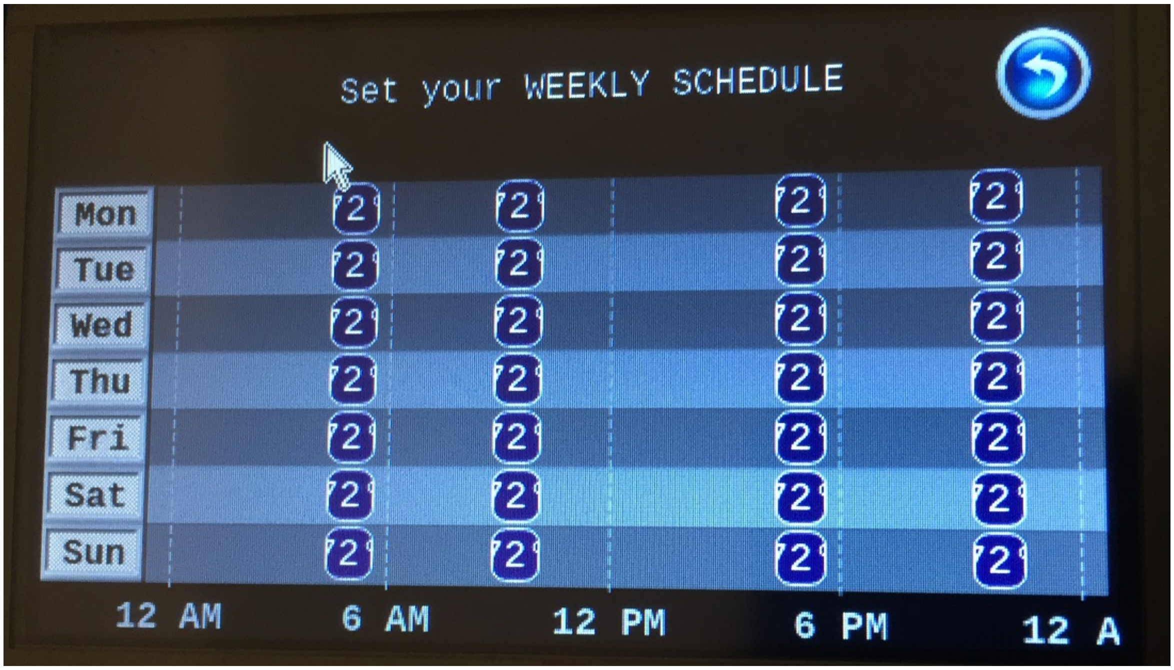


Figure 11. Smart Thermostat Demo GUI: Weekly Schedule Screen

7. The *Click to* Settings icon in Figure 9 leads to the city select screen shown in Figure 12.
   - Use the pull-down menu to select the desired city.
   - The setting options (General/Format/Network) appear on the left.
   - The return icon in the upper-right corner directs the user back to the *Home* screen in Figure 5.

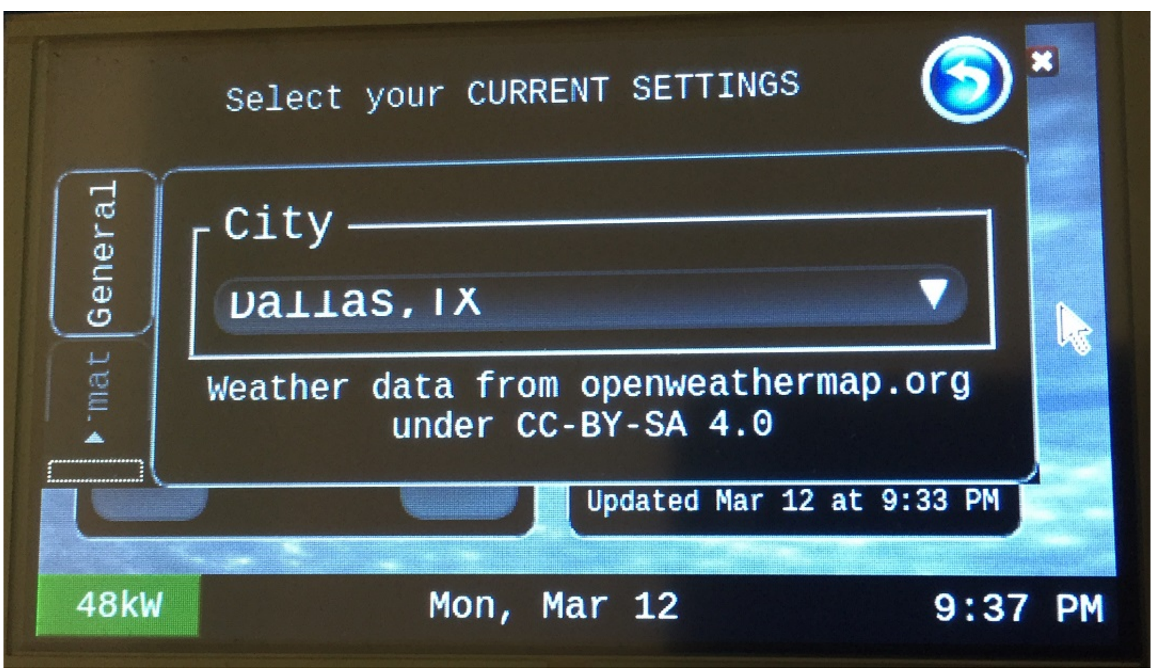


Figure 12. Smart Thermostat Demo GUI: City Select Screen

8. The Format option in Figure 12 leads to the temperature and time format select screen shown in Figure 13.
   - Click the desired format to select.
   - The setting options (General/Format/Network) appear on the left.
   - The return icon on the upper-right corner directs the user back to the *Home* screen in Figure 5.

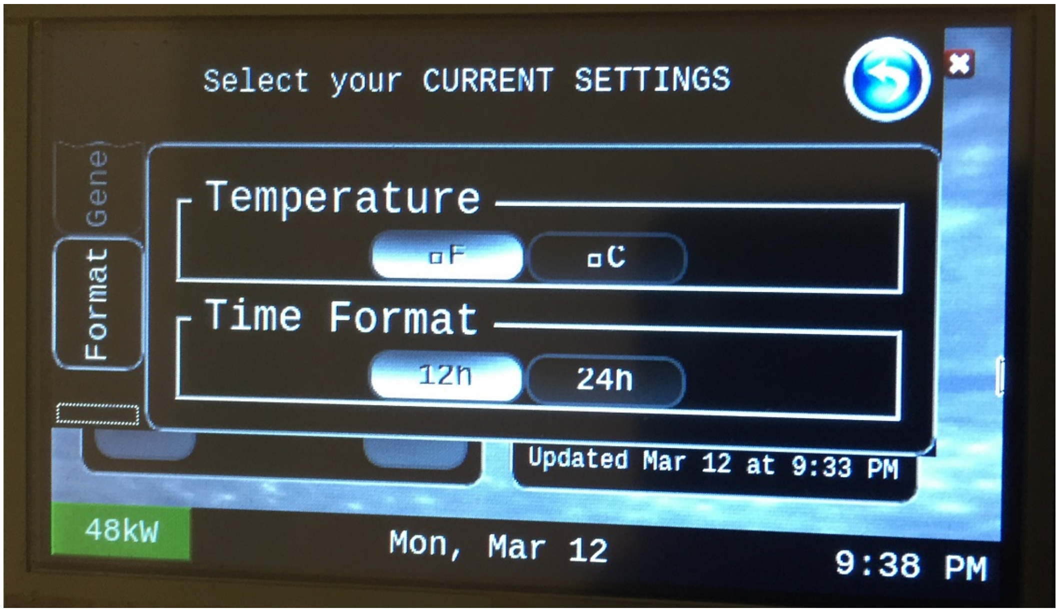


Figure 13. Smart Thermostat Demo GUI: Temperature and Time Format Setting Screen

9. The Network option in Figure 12 leads to the proxy settings screen shown in Figure 14.
   - Enter the proxy host and port number
   - The setting options (General/Format/Network) appear on the left
   - The return icon in the upper-right corner directs the user back to the *Home* screen in Figure 5.

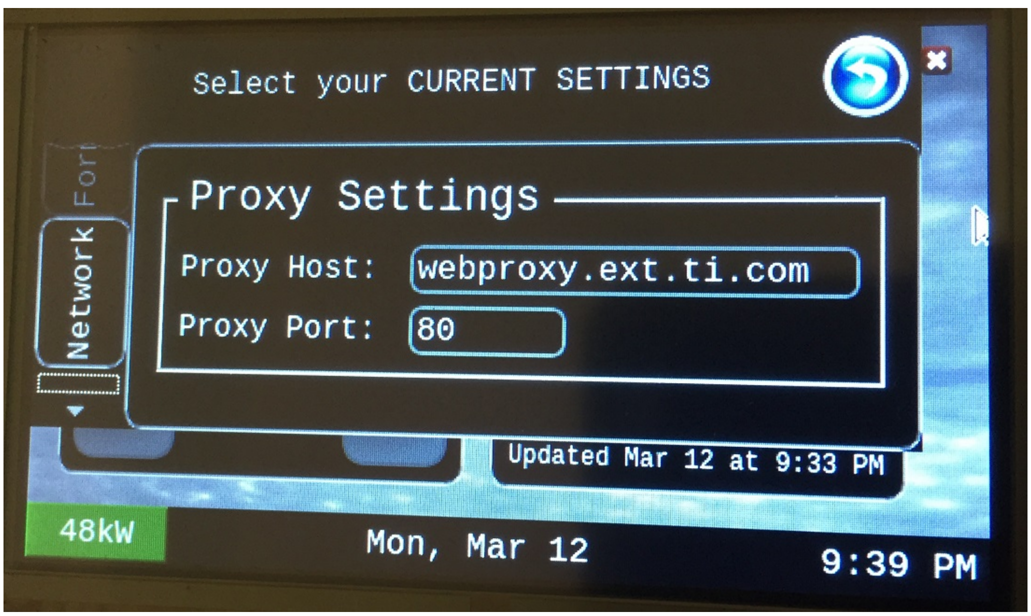


Figure 14. Smart Thermostat Demo GUI: Network Proxy Setting Screen

4 Design Files

To download the hardware design files for the AM335x Starter Kit, see the design files at TMDSSK3358.

5 Software Files

Download the Processor SDK Linux for AM335x from the AM335x software product page.

6 Related Documentation

1. Texas Instruments, *AM335x Starter Kit Hardware User's Guide* , AM335x Wiki Page
2. Texas Instruments, *Qt Training: Multipage Resizable Graphical User Interfaces containing Media* , Application Report (SPRACB2)
3. Texas Instruments, *Sitara Linux Training: Hands on with QT* , Wiki Page
4. Texas Instruments, *Processor SDK Linux Software Developer's Guide* , Wiki Page
5. Texas Instruments, *Processor Linux SDK Graphics and Display* , Wiki Page

6.1 Trademarks

7 Terminology

EVM— Evaluation module

GUI— Graphical user interface

HLOS— High-level operating systems

HMI— Human machine interface

PRU-ICSS— Programmable Real-Time Unit Subsystem and Industrial Communication SubSystem

SDK— Software development kit

SoC— System-on-chip

UART— Universal asynchronous transmitter/receiver

8 About the Author

ERIC RUEI is part of the Software Design team in the Catalog Processors BU. He has been with TI for more than 17 years and has worked on VoIP, Keystone and Sitara platforms. He has been a key player for various projects such as VoIP High-Density Solution and the PRU_based network and security offload engine present on Keystone SoCs. He serves as the lead and domain expert for the Graphics and Display subsystem on Sitara devices now. Eric earned his Master of Science in Electrical Engineering from UMD, College Park, MD