Introduction

Hello. My name is Alan Hawse. I am Senior Vice President of Technical Staff for Solutions and Software at Cypress Semiconductor. Welcome to Cypress Academy. This is the first video of a series of 40 or so that will provide you short, fundamental lessons in developing ultra-low-power, embedded IoT solutions using our PSoC 6 Microcontroller. PSoC 6 is Cypress’ solution for next generation embedded IoT applications. With its dual core ARM Cortex-M4 and Cortex-M0+ architecture, it delivers the performance needed for these applications; delivers security features critical for connected devices; and enables innovative solutions with its highly flexible and programmable resources—all while built on an ultra-low-power MCU architecture. Look, you’re going to get the best performance, the best connectivity and security—all in one package, and it’s awesome!

But I’m not in marketing, I’m an engineer…so let me show you what PSoC 6 can do. In this video series, I’m going to take you through an example IoT application from soup to nuts and step you through each part of the design. In this video Ill show you what I am going to build, and then in the rest of the videos in this series I will go through each of the building blocks and steps to realize this vision.

So, what are we going to build? Well, because my son, Nicholas, loves robots so much, we’re going to build one…we are going to build a BLE-controlled robotic arm assembly and not just any Robot… it is going to be a rise of the machine terminator style robot. I’m talking about kill switches, BLE wireless control, RTOS’, capacitive sensing, accelerometers, thermistors, RTCs, trace analyzers and debugging tools, secure bootloading, WiFi and before I am done I may even throw in some speech to text using some web-based services that are out there. Why such a complicated system? Well, because I don’t want Nicholas to claim it is boring… but really because it’s going to help you in developing your next IoT application…because every step of the way I’m going to show you how to build each of those building blocks. It’s gonna get messy, we’re both going to learn some stuff together but in the end, it’s going to be fun.

As we go, you might have questions or get stuck somewhere…to help you along the way I’ll add references to app notes and other documentation. If you have any questions along the way, post them in the PSoC 6 community and we’ll help each other out. Or if you have something else or just want to chat, please feel free to email me at [alan\_hawse@cypress.com](mailto:alan_hawse@cypress.com) or tweet me @askiotexpert.

Let’s get started…

For this class I will build all the projects on this devkit…. Called the PSoC 6 BLE Pioneer Kit or CY8CKIT-062-BLE…this kit ships with three boards in the box: an Arduino-Uno compatible base board, this is the main board with the PSoC 6 BLE MCU on it; an E-ink display shield with some sensors on it; and a CySmart BLE 4.2 USB dongle. The PSoC 6 BLE base board has a lot of stuff on it, here’s a few of the highlights:

* PSoC 6 BLE MCU with 1MB of internal flash, dual ARM cores and a PCB antenna
* PSoC 5LP MCU for debugging and programming
* An external 512 mega-bit flash IC
* Arduino-Uno compatible headers for Arduino shields
* EZ-PD CCG3 USB Type-C power delivery system and interface
* CapSense sliders and buttons
* RGB LED, etc

The E-ink display shield has:

* An E-ink display
* A PDM microphone
* Thermistor
* And a 9-axes motion sensor

In addition to this, I’m using a custom h-bridge shield I put together to control this 4-servo robotic arm I bought. The robotic arm, itself, is simply controlled by any MCU via a pulse width modulator (PWM) output. So, we’ll be using PWMs, to control each of the individual servos.

Through this video series we might use some additional hardware, for example a Cypress WiFi module or a second PSoC 6 BLE Pioneer Kit for some of the more advanced topics. I’ll introduce those pieces when we get there.

The class has 4 chapters which I will turn into the 40ish videos. They are:

1. System Architecture (Design flows, FreeRTOS, etc.)
2. Using the MCU peripherals (GPIOs, SmartIOs, PWMs, ADCs, etc.)
3. Building a BLE remote control (BLE, CapSense, E-ink display, motion sensor, etc.)
4. Connecting it all to the Internet with WiFi and cloud services

Again, post your comments and questions in the PSoC 6 community or, as always, feel free to email me at alan\_hawse@cypress.com or tweet me at @askioexpert with your comments, suggestions, criticisms and questions. Most of all let’s have some fun!