Oak Sensors
Firmware Developing

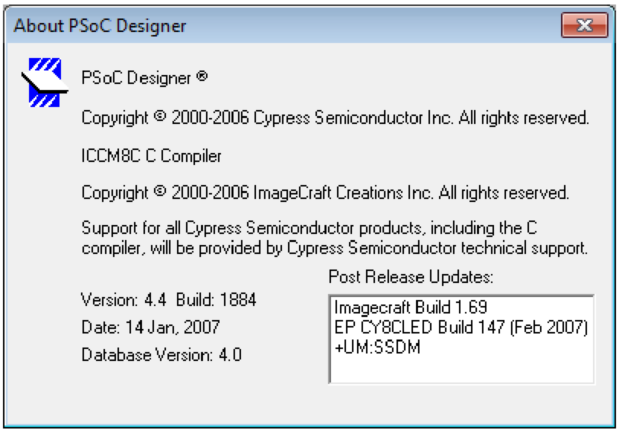
Contents

1. Introduction

This manual describes how to setup the tools to develop the Firmware for the Oak Sensor. Also a brief overview about the firmware structure is given with some hints.

2. Developer Tool “PSOC Designer”

The firmware of the Oak Sensors is developed with the “PSOC Designer”. The following version was used:



About PSoC Designer

PSoC Designer ®

Copyright © 2000-2006 Cypress Semiconductor Inc. All rights reserved.

ICCM8C C Compiler

Copyright © 2000-2006 ImageCraft Creations Inc. All rights reserved.

Support for all Cypress Semiconductor products, including the C compiler, will be provided by Cypress Semiconductor technical support.

Version: 4.4 Build: 1884
Date: 14 Jan, 2007
Database Version: 4.0

Post Release Updates:

Imagecraft Build 1.69
EP CY8CLED Build 147 (Feb 2007)
+UM:SSDM

You can download the “PSOC Designer” at http://www.cypress.com/.

2.1. PSOC Designer draft Overview

- Cypress offers a lot of “ready to run” modules for their Microprocessor.
  - The device editor (“View” -> “Device Editor”) allows selecting and configuring modules for the project.
  - Out of the selected module some of the source code is generated automatically.
- The Application Editor (“View” -> “Application Editor”) allows creating and editing additional source files. These source files use functions and definition generated out of the used modules (see above).

2.2. Update the Developer Tool

Because a lot of the source files are generated by the “PSOC Designer” automatically (see chapter 2.1) with every update some of these generated source files can change.

It is advisable to save the original version and compare it with the new version of the source files.

2.2.1 Boot.tpl

The files “boot.tpl” is one of these files (others are possible too).

After generating the the firmware please check in the file “boot.tpl” if the “Interrupt Vector Table” has at place 64h the correct entry (see below).

Part of the file “boot.tpl”

```
;-----------------------------------------------------------------------------
; Interrupt Vector Table
;-----------------------------------------------------------------------------
;
; Interrupt vector table entries are 4 bytes long.  Each one contains
; a jump instruction to an ISR (Interrupt Service Routine), although
; very short ISRs could be encoded within the table itself. Normally,
; vector jump targets are modified automatically according to the user
; modules selected. This occurs when the 'Generate Application' opera-
; tion is run causing PSoC Designer to create boot.asm and the other
; configuration files. If you need to hard code a vector, update the
; file boot.tpl, not boot.asm. See the banner comment at the beginning
; of this file.
;-----------------------------------------------------------------------------

    AREA TOP (ROM, ABS, CON)

    org   0                        ;Reset Interrupt Vector
    jmp   __Start                  ;First instruction executed following a Reset

    org   04h                      ;Supply Monitor Interrupt Vector
    halt                           ;Stop execution if power falls too low

    org   08h                      ;Analog Column 0 Interrupt Vector
    `@INTERRUPT_2`
    reti

    org   0Ch                      ;Analog Column 1 Interrupt Vector
    `@INTERRUPT_3`
    reti

          :
          :
          :

    org   64h                      ;Sleep Timer Interrupt Vector
    ljmp  _SleepTimer_ISR   <-- check this
    reti
```

If not modify the file “boot.tpl” and regenerate it. After the generation the new file “boot.asm” should contain this interrupt vector.

3. Modify the Firmware

In the source directory of the firmware exists the file “Oak_xxx.SOC”.

A double click on that file starts the “PSOC Designer” and loads all the source files.

A running “PSOC Designer can load the firmware with “File” -> “Open Project”.

4. Structure of the Firmware

4.1. Structure of the Oak Firmware

To work in the Source Code these hints may be useful:

- Most of the Oak Sensors use the “USB” and “Flash” module
- Most of the source files of all Oka Sensors deals with the USB communication and Flash memory. Therefore the firmware of most of the Oak Sensor types contains the same files. Only the the file for the peripheral are different (see next point).
- The whole handling of the used peripherals (sensor chip etc.) is done in the two files “sensor.c” and “sensor.h”. Beside that, additional feature reports can be defined in these files too.
- To setup a new type of Oka Sensor only these two files must be modified

5. Debugging

It is not possible to debug a firmware running on an Oak Sensor.

Because the base for an adaption is an already running firmware it is possible to do some change, compile and link it to generate an new HEX File (in the directory “../source/output”).

This HEX file can be programed and run on the Oak sensors.

If the necessary changes are done in small steps then it’s possible to do a firmware adaption without the use of a real time debugger. I little help could be the use of the LED to signal some situation.

5.1. Real time Debugging

Cypress offers a Real Time Debugger. This Debugger works with a special version of the Cypress chip witch have some additional pins for the debugger. An evaluation board contains this chip.

To run the Oak Firmware on such evaluation boards the peripheral side (sensor chip etc.) must be connected to this evaluations board.

6. Serial Number

The Toradex serial number is an 8 byte string containing only numeric characters.

In case of avoiding conflicts with existing Toradex serial numbers you must use an alpha numeric string which contains at least one at alpha character for your own serial numbers.

Disclaimer:

All data is for information purposes only and not guaranteed for legal purposes. Information has been carefully checked and is believed to be accurate; however, no responsibility is assumed for inaccuracies.

Brand and product names are trademarks or registered trademarks of their respective owners.

Specifications are subject to change without notice.