SmartICE04

René König

05.06.2012

Table of Contents

1. Introduction

1.1 DISCLAIMER

THIS PRODUCT IS PROVIDED "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS PRODUCT, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

1.2 Feedback

Feedback of any kind is welcome. You can reach me at renkoe@tbkoenig.de.

1.3 What is SmartICE04?

SmartICE04 is a programmer/debug adapter for TI’s RF- SoCs, consisting of some hardware, a COM object managing the communication with the hardware (COM in sense of Microsoft’s Component Object Model, not the serial port) and an AGDI Debug Driver. The name “*SmartICE04*” itself is based on the original SmartRF04EB provided by Texas, and the hardware is actually able to run the original firmware coming with the SmartRF Flash Programmer.

The COM object gives you more or less direct access to the target device’s debug interface. Additionally, it makes you able to program your target from within your own software or from small scripts (sample scripts can be found in the samples folder).

The AGDI Driver, sitting on top of the COM object, gives you the ability to debug your target with KEIL’s uVision debugger.

1.4 Devices supported

The devices supported by SmartICE04 are the CC1110 and CC1111. I assume that it will work with the CC2510 and CC2511 as well, but I did not have the opportunity to test it so far.

2 Hardware

2.1 The Drawing Files

The hardware has been designed with EAGLE. If you don't have EAGLE on your computer yet, go to http://www.cadsoft.de/ and download the latest version. This is a self-extracting executable, so there is no need to go through the setup process. Just open the download with your favorite packer and extract the bundled files. After that, you can execute eagle.exe found in the bin directory. Choose the freeware option shown on the first startup and you're ready to open the files. You can now print the files, produce the gerbers, the bom and whatever data needed.

2.2 The Circuit

The Circuit is basically the SmartRF04EB provided by TI with all things not needed removed. The resistors connected to the port pins of the controller are used to provide proper levels. This is necessary because of the missing hardware mounted onto the SmartRF04EB.

Some modifications have been made: The RC reset circuit has been replaced because of the reasons described in Design Note DN 300, paragraph 3.6: "*..., but we have seen rare cases where the bootloader firmware has been erased or damaged unintentionally.*" This is a well-known problem on the C8051F320 and has been addressed by IC2 and R5.

The second change applies to the level shifter. The ST2738E has been replaced by a 74AVC4T245. This is what they do at Texas with the "*System-on-Chip Debug Plug-In board*" as described in swru134a.pdf, paragraph 9.3.

Resistors R15 and R16 divide the voltage attached to the target. The C8051F320 is able to read the voltage level. This is used to determine if there is a target attached or not. The target's voltage should not exceed 3.7 V.

2.3 The Mechanics

The PCB is designed to fit in an enclosure made by BOPLA, namely EG-1030. The PCB could be much smaller, but I got some enclosures from Reichelt Elektronik before and decided to use them. ☺

3 Software

3.1 Programming the firmware

In order to program the firmware, you need either a SiLabs USB Debug Adaptor or a Serial Adaptor. Connect the adaptor to K2 and attach a USB cable (SmartICE04 gets powered via the USB link).

The PC-Side software to program the hex-file can be found in SmartICE04's bin folder, look for SmartICE04Prog.exe. With this tool, you can either program the original SmartRF04EB hex files (you need to install TI's SmartRF Flash Programmer to get these files) or the SmartICE04 hex file located in the Firmware directory. However, if you need KEIL uVision debugging support, you need to program the latter file.

Please use SmartRF04Prog for both types of hex files for the following reasons: In case of the TI files, you need to mark the device as bus powered (the original evaluation board is self-powered). Additionally, bit 7 in the bmAttributes field in the device's configuration descriptor got accidentally cleared. However, this bit is reserved and **must** be set to one (see USB specification 2.0, paragraph 9.6.3). SmartICE04Prog addresses this and corrects both bits.

In case of the SmartICE04 hex file, SmartICE04Prog sets the device's serial number. This is important, because this is how multiple devices are differentiated by the operating system and by the software found in this download.

3.2 Attaching the hardware to the PC and installing the drivers

If you already programmed the firmware, the device is already attached to the PC. As you found out, the device needs some sort of operating voltage. ☺

If all went well, the PC detects new hardware now. Fortunately there is no need to install special drivers, because SmartICE04 follows the HID protocol. The operating system should do all the things necessary automatically. It does not matter whether you use a 32 bit or a 64 bit platform.

However, if you decided to use the hex-files provided by Texas, a driver is needed. You already installed the driver files together with the SmartRF Flash Programmer, so this should work without any problems as well.

3.3 Installing the Software

3.3.1 Using the installer packages

There are two different installation packages within the setup folder, an x86 version and an x64 version. When using a 64 bit OS, however, you also have to install the x86 package if you like to use the uVison debug driver (uVision is a 32 bit application). The 64 bit version is useful only if you need to use the COM component from within a 64 bit application.

The installer copies the SmartICE04.dll into %CommonProgramFilesFolder%\TBK\SmartICE04, registers the COM components and integrates into your uVision installation by adding the appropriate entries to your TOOLS.INI file.

3.3.2 Installing the software by hand

If you dislike using msi packages, you have to do all the steps required yourself. Fortunately, this is a very easy task: First of all, open a command prompt and navigate to the directory containing SmartICE04.dll (x86 version, even when running a 64 bit OS) and type:

```
Regsvr32 SmartICE04.dll
```

If you're using a 64 bit OS, repeat this step with the x64 version of the DLL.

Now, go to your KEIL installation directory and open the TOOLS.INI file with a text editor and add the following line:

```
TDRV10=<x86_PathToDll>\SmartICE04.dll ("TBK SmartICE04 Driver")
```

If TDRV10 is already used, then pick a different number.

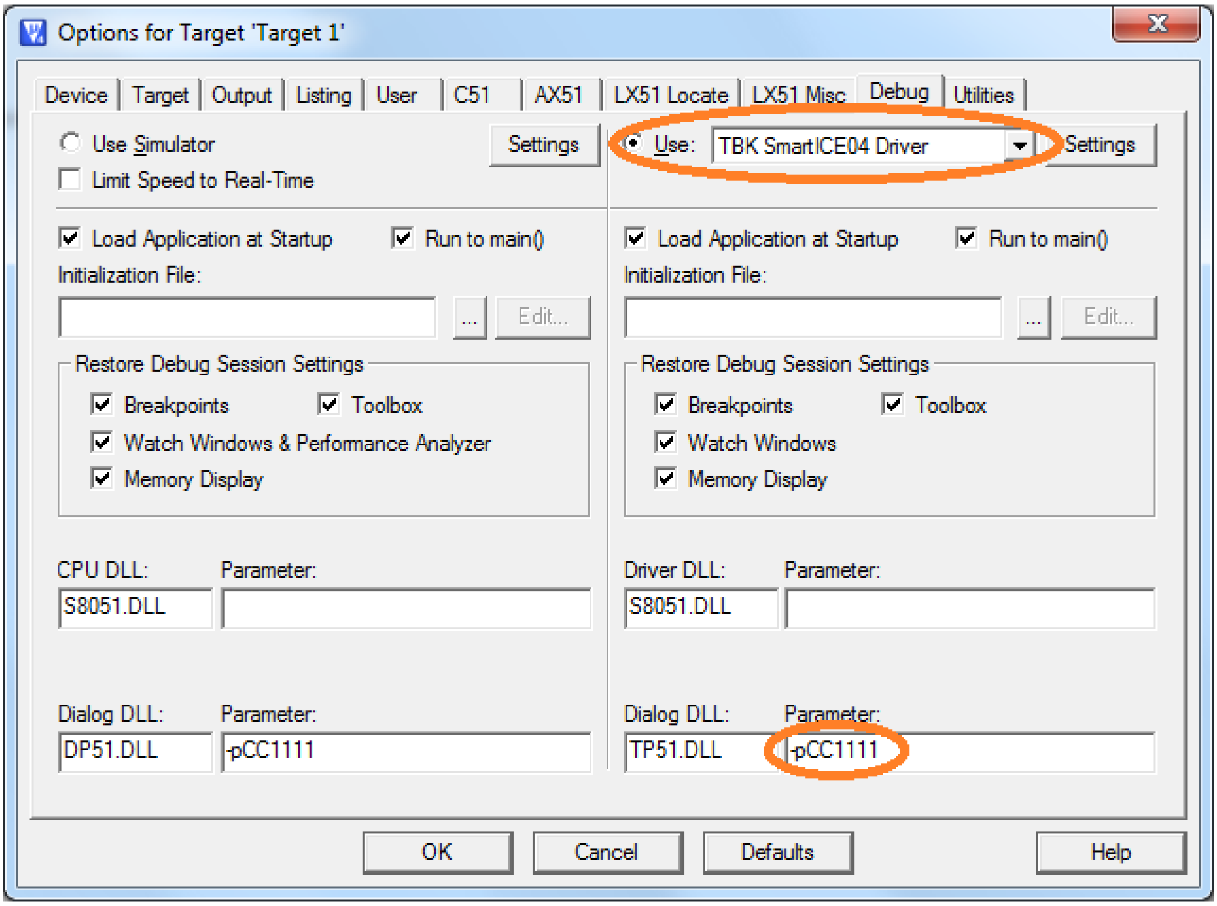
3.4 Debugging with the KEIL uVision Debugger

In order to use SmartICE04 for debugging, you need to open the debug options tab. Choose the "TBK SmartICE04 Driver" and, very **important**, specify the target device. An example setup for the CC1111 can be found in the screenshot on the following page.

If there is a (powered) SoC attached to the debug adapter, you should be able to start the debug session now.

Please keep in mind, that the maximum number of breakpoints is limited within the currently supported devices. If you aren't able to set breakpoints, have a look onto the debug output window. If you reached the limit, you'll find that info here.

Figure 1, Debug Setup Example



○ Use Simulator

☐ Limit Speed to Real-Time

☑ Load Application at Startup ☑ Run to main()

Initialization File:

Restore Debug Session Settings

☑ Breakpoints ☑ Toolbox

☑ Watch Windows & Performance Analyzer

☑ Memory Display

CPU DLL: S8051.DLL Parameter:

Dialog DLL: DP51.DLL Parameter: -pCC1111

● Use: TBK SmartICE04 Driver

☑ Load Application at Startup ☑ Run to main()

Initialization File:

Restore Debug Session Settings

☑ Breakpoints ☑ Toolbox

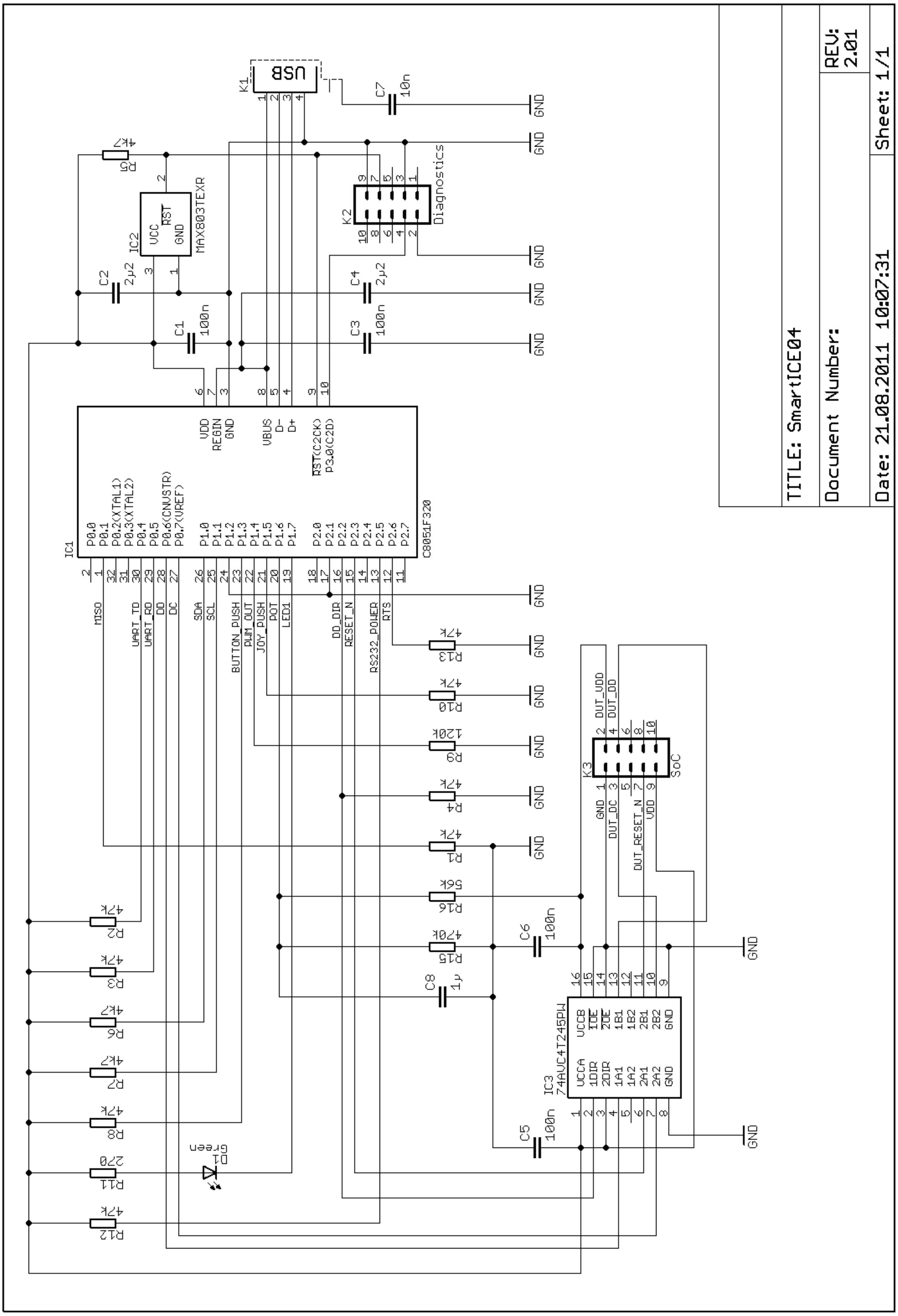
☑ Watch Windows

☑ Memory Display

Driver DLL: S8051.DLL Parameter:

Dialog DLL: TP51.DLL Parameter: -pCC1111

4. Schematic


TITLE: SmartICE04
Document Number:
REV: 2.01
Date: 21.08.2011 10:07:31
Sheet: 1/1
IC1
C8051F320
IC2
MAX803TEXR
IC3
74AVC4T245PW
K1
USB
K2
Diagnostics
K3
SoC
VDD
REGIN
GND
VBUS
D-
D+
RST(C2CK)
P3.0(C2D)
P0.0
P0.1
P0.2(XTAL1)
P0.3(XTAL2)
P0.4
P0.5
P0.6(CNVSTR)
P0.7(VREF)
P1.0
P1.1
P1.2
P1.3
P1.4
P1.5
P1.6
P1.7
P2.0
P2.1
P2.2
P2.3
P2.4
P2.5
P2.6
P2.7
MISO
UART_TD
UART_RD
DD
DC
SDA
SCL
BUTTON_PUSH
PWM_OUT
JOY_PUSH
POT
LED1
DD_DIR
RESET_N
RS232_POWER
RTS
VCCA
1DIR
2DIR
1A1
1A2
2A1
2A2
VCCB
1OE
2OE
1B1
1B2
2B1
2B2
GND 1
DUT_DC 3
DUT_RESET_N 7
VDD 9
DUT_VDD
DUT_DD
C1 100n
C2 2µ2
C3 100n
C4 2µ2
C5 100n
C6 100n
C7 10n
C8 1µ
R1 47K
R2 47K
R3 47K
R4 47K
R5 4K7
R6 4K7
R7 4K7
R8 47K
R9 120K
R10 47K
R11 270
R12 47K
R13 47K
R15 470K
R16 56K
D1 Green