AN10302

Using the Philips LPC2000 Flash utility with the Keil MCB2100 and IAR LPC210x Kickstart evaluation boards

Rev. 04 — 12 October 2006 Application note

Document information

Info	Content
Keywords	LPC2000, Flash utility, Keil MCB2100, IAR LPC210x
Abstract	Application information for the Philips LPC2000 Flash utility with the Keil MCB2100 and IAR LPC210x Kickstart evaluation boards

Revision history

Rev	Date	Description
04	20061012	Fourth version • Updated Figure 2.
03	20040610	Third version
02	20040512	Second version
01	20040430	Initial version

Contact information

For additional information, please visit: **http://www.nxp.com**

For sales office addresses, please send an email to: **salesaddresses@nxp.com**

1. Introduction

In-System programming (ISP) is a method of programming and erasing the on-chip flash or RAM memory using the boot loader software and a serial port. The part may reside in the end-user system. The flash boot loader provides an In-System Programming interface for programming the on-chip flash or RAM memory. This boot loader is located in the upper 8 kB of flash memory, it can be read but not written to or erased.

2. LPC2000 ISP overview

The flash boot loader code is executed every time the part is powered on or reset. The loader can execute the ISP command handler or pass execution to the user application code.

A LOW level, after reset, at the P0.14 pin is considered as the external hardware request to start the ISP command handler. The boot loader samples this pin during reset. Assuming that proper signal is present on X1 pin when the rising edge on RST pin is generated, it may take up to 3 ms before P0.14 is sampled and the decision on whether to continue with user code or ISP handler is made. If P0.14 is sampled LOW and the watchdog overflow flag is set, the external hardware request to start the ISP command handler is ignored. If there is no request for the ISP command handler execution (P0.14 is sampled HIGH after reset), a search is made for a valid user program. If a valid user program is found then the execution control is transferred to it. If a valid user program is not found, the auto-baud routine is invoked.

Pin P0.14 is used as hardware request for ISP requires special attention. Since P0.14 is in high impedance mode after reset, it is important that the user provides external hardware (a pull-up resistor or other device) to put the pin in a defined state. Otherwise unintended entry into ISP mode may occur.

Figure 1 shows the boot sequence of the LPC210x devices.

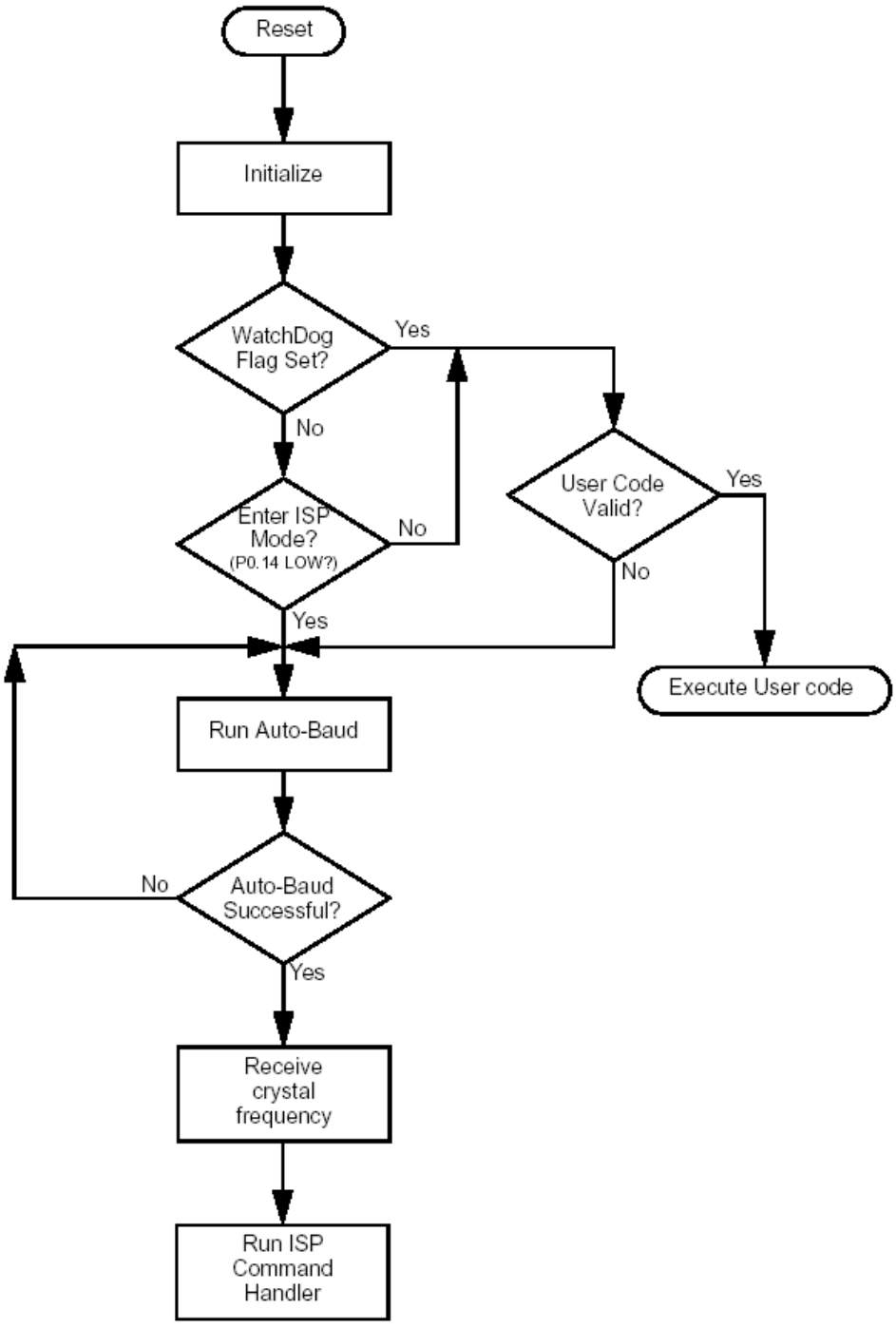

Fig 1. Boot process flowchart.

3. Details of the Philips LPC2000 Flash utility

This flash utility is available for free download from the Philips website. This software, in combination with the hardware described below, allows for hands-off erasure, uploading, and execution of code.

The Philips LPC2000 Flash utility utilizes two, otherwise unused, signals (RTS and DTR) of the PC serial port to control the microcontroller reset and P0.14 pins. The port pin P0.14, if LOW during reset, puts the microcontroller into In System Programming (ISP) mode; this pin has the alternate functions of external interrupt one and general purpose I/O (GPIO). Some details on the associated circuitry will help in understanding how this works.

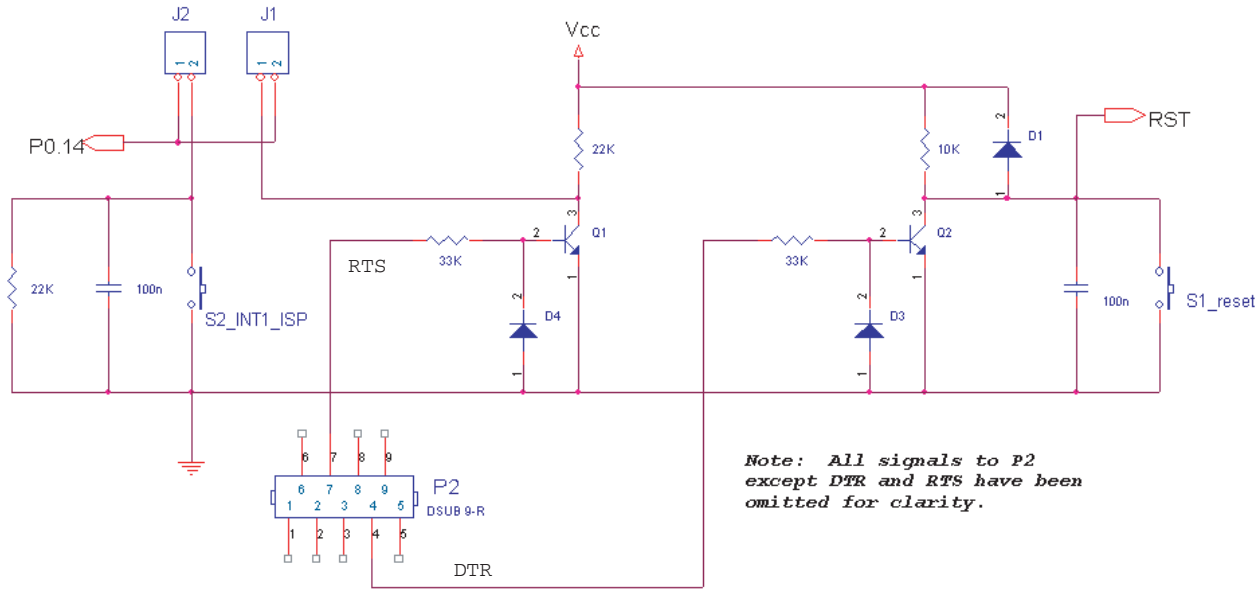

Fig 2. The RTS/DTR control - an example circuit.

3.1 Manual entry into ISP mode

With jumper J1 removed and jumper J2 in place ISP mode will be entered manually by holding S2 while pressing and releasing S1 (reset). This can become cumbersome and so it is advantageous to use RTS/DTR control of these signals.

3.2 ISP mode entry using DTR/RTS

With jumper J1 inserted and jumper J2 removed the reset and P0.14 signals may be controlled by the previously un-used RTS/DTR signals of the PC serial port. In this application both these signals are active HIGH. When RTS is asserted Q2 is turned on and the microcontroller reset is pulled LOW. While the micro is held in reset, DTR is asserted and P0.14 is held LOW. RTS is then brought LOW and so Q2 is turned off. The 10K pull-up resistor releases the RESET signal by pulling it HIGH. The microcontroller is now running in ISP mode.

This sequence of ISP mode entry is performed for every operation offered by the Philips LPC2000 Flash Utility.

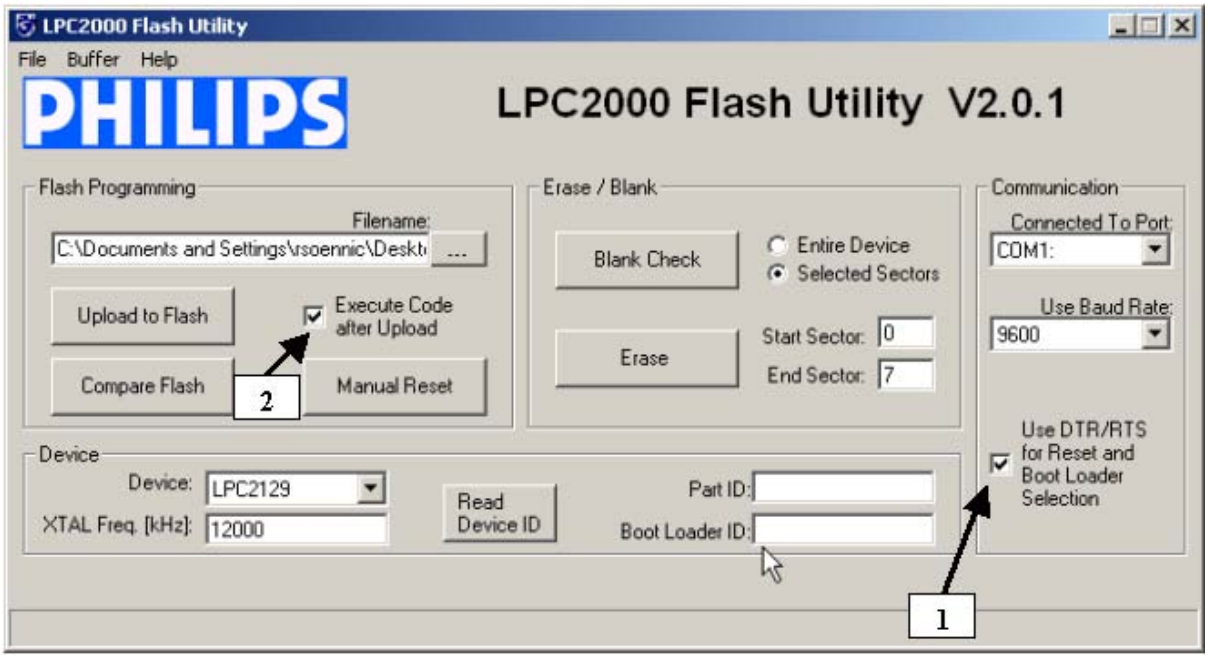

Fig 3. Flash Utility main screen.

The main screen of the Flash Utility provides access to most if its functionality. When the “use DTR/RTS…” box (1) is checked then control of reset and P0.14 is done by the utility as described above. If this box is unchecked then ISP mode must be entered manually. If the “execute code after upload” is checked then, after code is programmed into the flash, an extra reset pulse is sent to the microcontroller to reset the part. Since, at this time, P0.14 will be HIGH, the part will execute code in flash after this reset.

When the utility connects to the MCB2100 it will attempt to connect at the selected baud rate. The highest baud rate achievable will depend mostly on the frequency of the crystal. Using standard baud rate crystals (e.g. 14.7456 MHz) will increase the maximum baud rate achievable.

3.3 Flash buffer operations

The flash buffer operation screen (accessible from the “buffer” pull-down menu) allows functions such as loading a HEX file, downloading from flash, uploading to flash, filling the buffer, saving the HEX file and calculation of the checksum “valid code” vector[1]. There is also the ability to fill the buffer with a particular value[1] and program this buffer to flash.

1. The valid code vector at 0x14 is merely the two's complement of the sum of the vector table. By assigning it this value the checksum for the entire vector table is 0x00 which indicates valid flash code. After reset the bootloader will examine this location and, if the value is correct (an indication of valid user code in flash), will execute code out of flash. If the value is not correct the bootloader will enter ISP mode. The Philips LPC2000 Flash Utility will automatically calculate and program this value during an upload to flash. Alternatively the vector calculation may be performed on the contents of flash buffer as shown in the screen-shot below.

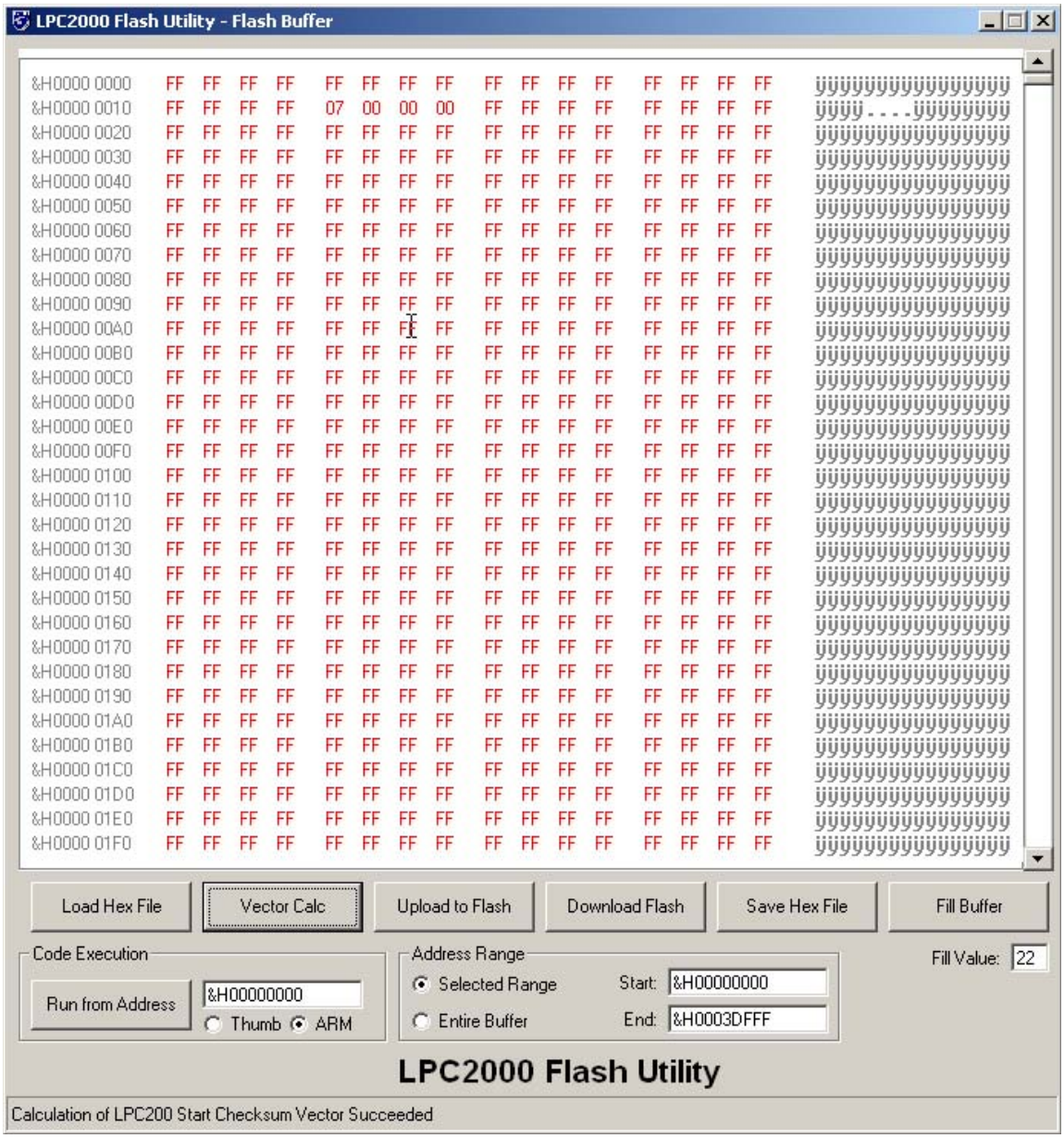

Fig 4. Flash buffer screen.

3.4 RAM buffer operations

Ram buffer operations (accessible from the “buffer” pull-down menu) are similar to flash buffer operations including the uploading of HEX files etc.

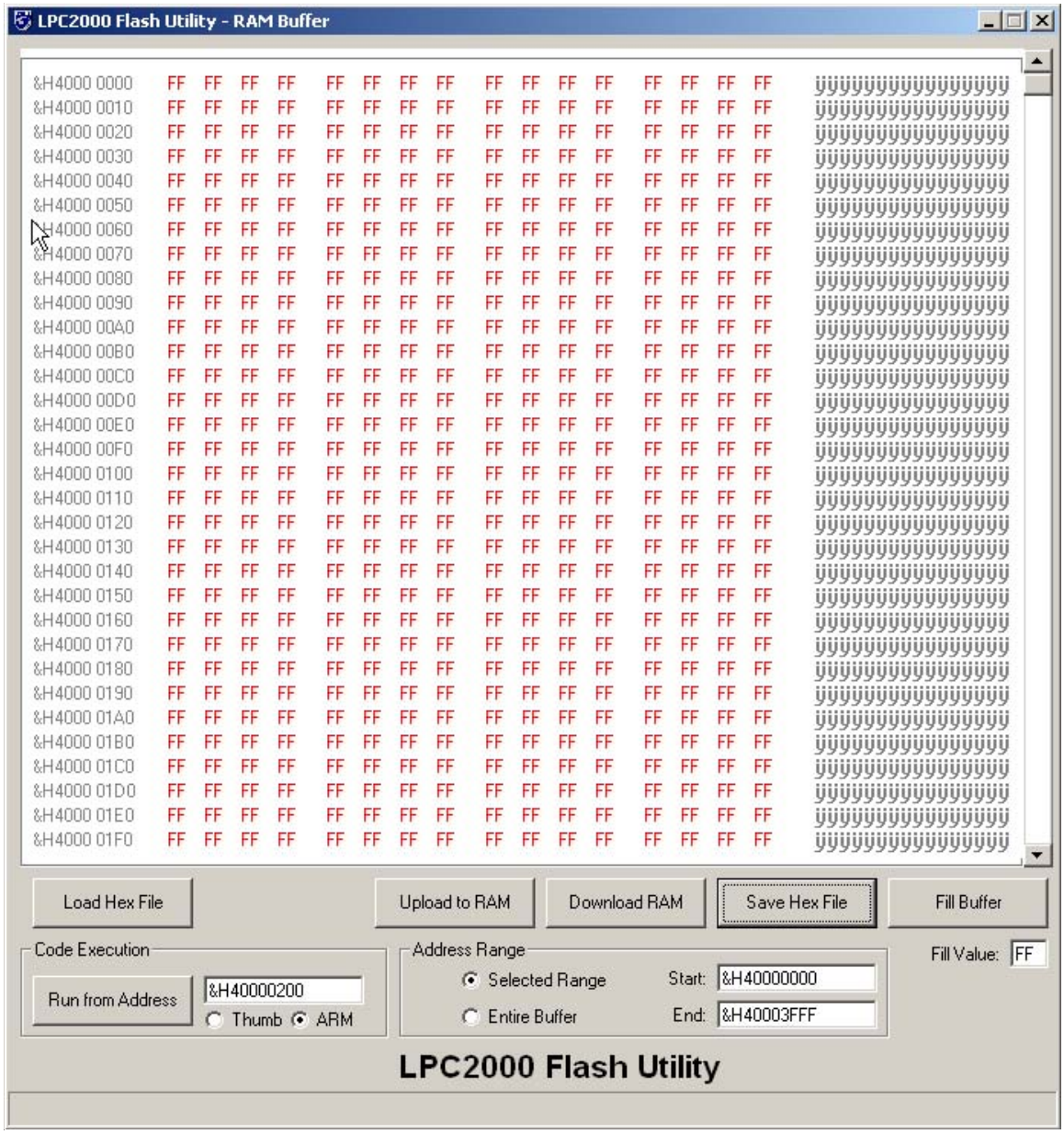

Fig 5. RAM buffer operations.

4. Hardware

4.1 Keil MCB2100 evaluation board

Figure 6 shows an overview of the Keil MCB2100 evaluation board.

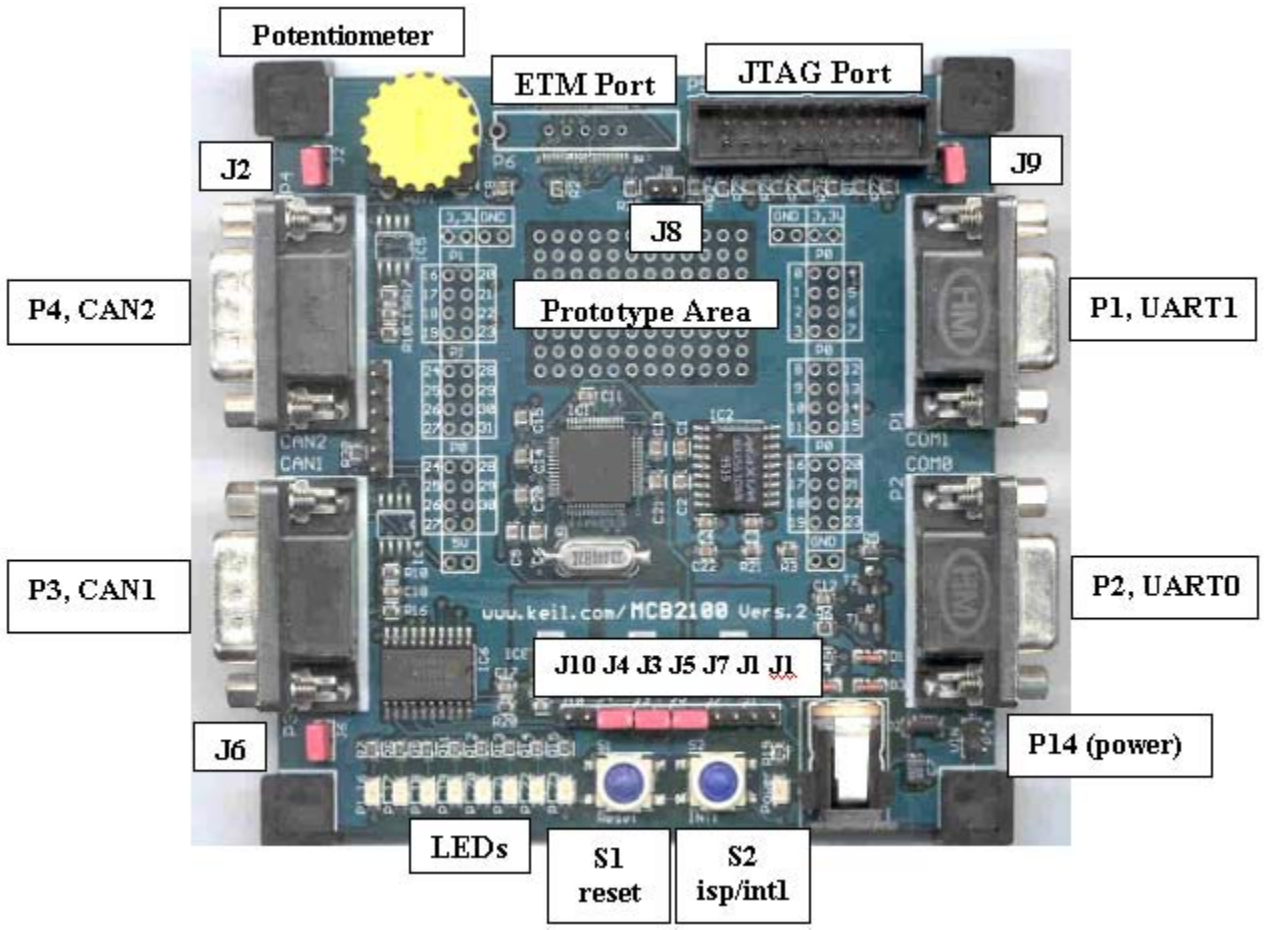

Fig 6. Keil MCB2100 evaluation board overview.

JTAG port — Connection to JTAG emulator (e.g. Keil ULink). This is a standard JTAG port as outlined in ARM documentation.

ETM (Embedded Trace Macrocell) port — Provides interface to emulators with trace capability.

P3 and P4, CAN ports — These provide access to the CAN ports (On boards that feature a microcontroller with CAN interfaces).

P1 and P2, UARTs — Access to UART0 and UART1.

S1 reset — Microcontroller reset.

S2 ISP/INT1 — This button pulls the P0.14 pin of the microcontroller LOW, providing either an external interrupt or manual entry into ISP mode.

LEDs — buffered with a 74LVC octal buffer, enabled by J6.

Potentiometer — Configured as a voltage divider with its output connected to AIN0 via jumper J2.

Table 1. Keil MCB2100 jumper functions

Jumper	Function
J1	Configures P0.14 for DTR/RTS control of ISP (see ISP section below)
J2	Potentiometer/ADC Connect
J3[1]	3.3 V enable
J4[1]	1.8 V enable
J5	3.3 V analog voltage supply enable
J6	LED enable
J7	Configures P0.14 for external interrupt or manual ISP entry
J8	ETM Pins Enable (Pulls TraceSync LOW)
J9	JTAG Debug Pins Enable (Pulls RTCK LOW)
J10[2]	Configures RESET for DTR/RTS control of ISP (see ISP section below)

[1] These jumpers supply the voltages to the microcontroller and must be in for normal operation.

[2] Remove this jumper when not using ISP.

4.1.1 Enabling ISP mode with the MCB2100

The Keil MCB2100 evaluation board was designed to utilize the RTS/DTR control of reset and P0.14 as featured in the Philips LPC2000 Flash utility.

To setup the MCB2100 for ISP programming **set** the jumpers: **J1, J3, J4, J5, J7** and **J10**. Connect the PC serial port to COM0 of the MCB2100 and start the LPC2000 Flash Utility. Check the “Use DTR/RTS……” box and continue.

4.2 The IAR/Philips LPC210x Kickstart card

This evaluation board is populated with an LPC2106 microcontroller and features 2 serial ports, 2 user-defined buttons, 16 fully configurable LEDs, 20-pin JTAG interface connector as well as breakout headers for all pins.

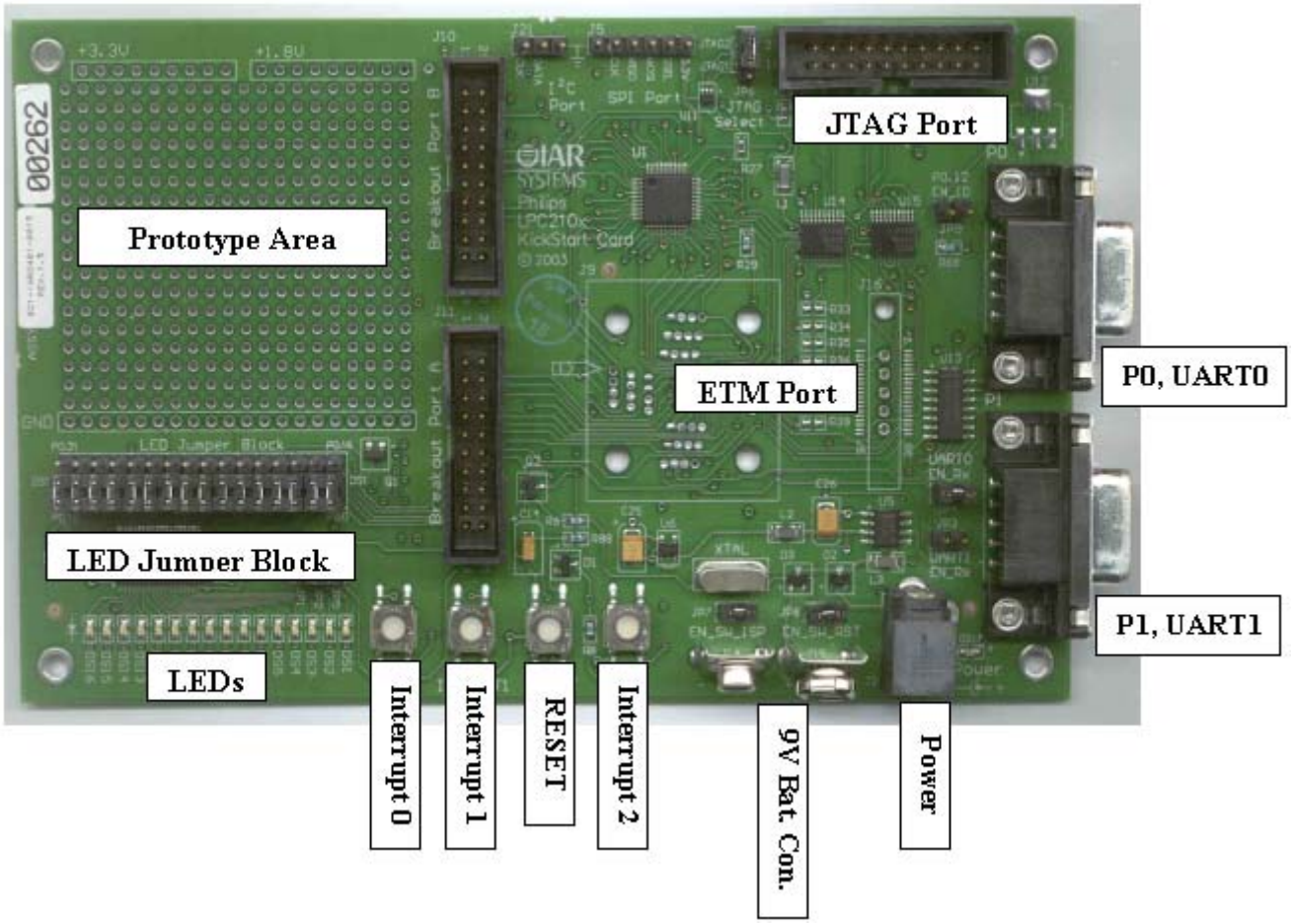

Fig 7. IAR/Philips LPC210x Kickstart card.

JTAG port — Connection to JTAG emulator (e.g. JLink). This is a standard JTAG port as outlined in ARM documentation.

ETM (Embedded Trace Macrocell) port — Provides interface to emulators with trace capability.

P0 and P1, UARTs — Access to UART0 and UART1.

RESET — Microcontroller reset.

Interrupt0 — This button provides a source for interrupt zero.

Interrupt1 — This button pulls the P0.14 pin of the microcontroller LOW, providing either an external interrupt or manual entry into ISP mode.

Interrupt2 — This button provides a source for interrupt two.

LED jumper block — enables/disables individual LEDs.

LEDs — buffered with a LVT16244.

Table 2. IAR/Philips Kickstart card jumper functions

Jumper	Function
JP1	Enables external interrupt zero via the push-button
JP2	Enables ISP and external interrupt one[1]
JP3	Connects P0.9/RxD1 (UART1) to the MAX3232
JP4	Connects P0.1/RxD0 (UART0) to the MAX3232
JP5	Enables external interrupt zero via the push-button
JP6	Primary/Secondary JTAG select[2]
JP7	Enable DTR/RTS control of P0.14
JP8	Enable DTR/RTS control of RESET

[1] P0.14 and external interrupt one share the same pin; therefore this button may also be used for manual entry into ISP mode by pressing it during a reset.

[2] This jumper, when in the JTAG1 position, will cause the microcontroller to enter JTAG debug mode after reset. Therefore, when using ISP, this jumper must be removed or placed in the JTAG2 position.

4.2.1 Enabling ISP mode with the IAR/Philips Kickstart card

The Kickstart Card evaluation board was designed to utilize the RTS/DTR control of reset and P0.14 as featured in the Philips LPC2000 Flash utility.

To setup the Kickstart Card for ISP programming set the jumpers: JP7, JP8, JP2 and JP4. Remove jumper JP6. Connect the PC serial port to P0 (UART0) of the Kickstart Card and start the LPC2000 Flash Utility. Check the “Use DTR/RTS……” box and continue.

5. Legal information

5.1 Definitions

Draft — The document is a draft version only. The content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included herein and shall have no liability for the consequences of use of such information.

5.2 Disclaimers

General — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, nor in applications where failure or malfunction of a NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

5.3 Trademarks

Notice: All referenced brands, product names, service names and trademarks are the property of their respective owners.

6. Contents

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: 12 October 2006
Document identifier: AN10302_4