# Table of Contents

1. Introduction to MINI2440 development board .................................................................6
  1.1 Brief introduction ...........................................................................................................6
      1.1.1 Physical appearance of MINI2440 .........................................................................6
      1.1.2 Hardware Features ..................................................................................................6
      1.1.3 CD Contents ...........................................................................................................8
  1.2 Hardware Resources ......................................................................................................9
      1.2.1 Address Space Layout and Chip Selection signal definition ...................................9
      1.2.2 Jumper ..................................................................................................................11
      1.2.3 Interfaces .............................................................................................................11
  1.3 Linux Features .............................................................................................................11
  1.4 WindowsCE Features ..................................................................................................12

2. Operating Instruction ......................................................................................................13
  2.1 Settings and Connection on the Board ........................................................................13
      2.1.1 Selecting Boot Mode .............................................................................................13
      2.1.2 Connecting Interfaces ..........................................................................................14
      2.1.3 Setting the Hyper-terminal ..................................................................................14
  2.2 BIOS Functions and Operating Instruction ...................................................................17
      2.2.1 Entering BIOS mode .............................................................................................17
      2.2.2 Installing USB Driver ............................................................................................19
      2.2.3 Function Menu .......................................................................................................22
      2.2.4 Sub-menu Functions ..............................................................................................26
      2.2.5 Setting the Linux boot parameters .......................................................................29
  2.3 Testing Peripheral Resources ......................................................................................32
      2.3.1 Downloading and Running Test Program .............................................................32
      2.3.2 Peripheral Resource Testings ................................................................................35
  2.4 Linux Operations ..........................................................................................................44
      2.4.1 Playing MP3 files ....................................................................................................44
      2.4.2 Aborting running programs ....................................................................................44
      2.4.3 Using U-disk/mobile disk .......................................................................................45
      2.4.4 Using SD card .........................................................................................................45
      2.4.5 Using USB Camera .................................................................................................46
      2.4.6 Serial Port Intercommunication ............................................................................47
      2.4.7 Controlling LEDs ....................................................................................................49
      2.4.8 Testing Buttons .......................................................................................................50
      2.4.9 Testing the serial ports 2 and 3 .............................................................................51
      2.4.10 Testing the buzzer .................................................................................................52
      2.4.11 Testing I2C Storage Device ..................................................................................52
      2.4.12 Visiting BBS via Telnet .........................................................................................54
      2.4.13 Visiting Internet .....................................................................................................55
      2.4.14 Setting up MAC address .......................................................................................58
      2.4.15 Remotely Visiting the Development Board via Telnet .......................................61
      2.4.16 Transmitting Files via FTP ....................................................................................61
      2.4.17 Mounting and Using NFS (Network Filesystem) ..................................................62
      2.4.18 Setting and Saving RTC Time ..............................................................................63
      2.4.19 Saving Data to the Flash when Power Fails .........................................................63
      2.4.20 Setting Self-running Programs ............................................................................63
2.4.21 Using Commands to Capture the Screen .........................................................65

2.5 WinCE Functions and Peripheral Resources Testing ........................................65
2.5.1 Testing LED .................................................................................................66
2.5.2 Rotating the Screen ....................................................................................66
2.5.3 Testing Serial Port Communication ...........................................................67
2.5.4 Using U-disk ...............................................................................................69
2.5.5 Using SD/MMC Card ..................................................................................70
2.5.6 Playing mp3 with Windows Media Player ..................................................70
2.5.7 Playing Mpeg4 files with Superplayer .......................................................71
2.5.8 Testing Ethernet Function ..........................................................................71
2.5.9 Visiting the board via telnet ........................................................................73
2.5.10 Transmitting files to the board via FTP ....................................................73
2.5.11 Testing web server ....................................................................................74
2.5.12 Calibrating the Touch Screen ...................................................................75
2.5.13 Using ActiveSync for USB Synchronous Communication ....................76
2.5.14 Testing Wifi ...............................................................................................77

2.6 Using H-JTAG to burn BIOS ..........................................................................80
2.6.1 Introduction to H-JTAG ..............................................................................80
2.6.2 Installing and Setting H-JTAG ....................................................................80
2.6.3 Setting the Flash Model Number and Burning BIOS .................................85

3. Backing up and Installing/Updating the System ..............................................92
3.1 Backing up and Restoring the system ...............................................................92
3.1.1 Backing up the system ...............................................................................92
3.1.2 Restoring System with the Backup File .....................................................97

3.2 Installing Linux System ..................................................................................100
3.2.1 Partitioning Nand Flash ..........................................................................100
3.2.2 Installing bootloader ...............................................................................102
3.2.3 Installing Linux Kernel ...........................................................................104
3.2.4 Installing Filesystem ...............................................................................106

3.3 Installing WinCE OS ......................................................................................109
3.3.1 Partitioning ...............................................................................................109
3.3.2 Installing Bootloader ...............................................................................111
3.3.3 Installing Eboot .......................................................................................114
3.3.4 Installing WinCE Kernel Image .................................................................115

4. Integrated Development Environment .........................................................119
4.1 Creating LED Project in ADS .......................................................................119
4.1.1 Creating a Project ....................................................................................119
4.1.2 Compiling and Linking Project .................................................................124

4.2 Debugging with H-JTAG .................................................................................133
4.2.1 Configuring AXD Debugger for H-JTAG ...............................................133
4.2.2 H-JTAG Debugging in ADS1.2 Environment ........................................136

4.3 Compiling and Burning the 2440test Program .............................................136
4.3.1 Compiling and Debugging 2440test .........................................................136
4.3.2 Downloading the 2440test program via USB .........................................141
4.3.3 Burning 2440test Program to Nand Flash ..............................................145

4.4 Compiling and Burning uCos2 ......................................................................148
4.4.1 Compiling uCos2 ....................................................................................148
4.4.2 Downloading uCos2 to SDRAM ...............................................................150
4.4.3 Burning uCos2 to the Nand Flash .................................................................153
5. Building Linux Development Environment .................................................156
  5.1 Building Development Environment Based on Redhat Linux 9.0 .....156
    5.1.1 Installing Redhat 9.0 .................................................................156
    5.1.2 Building Cross Compiling Environment .................................168
    5.1.3 Configuring NFS Service ..........................................................170
    5.1.4 Booting System via NFS ..........................................................172
    5.1.5 Configuring FTP Service on the PC ........................................174
    5.1.6 Configuring telnet Service on the PC ....................................174
    5.1.7 Creating New User in Redhat ..................................................175
6. Embedded Linux Development .................................................................177
  6.1 Hello, World! .........................................................................................177
    6.1.1 Source Code ...........................................................................177
    6.1.2 Compiling Hello, World .........................................................177
    6.1.3 Downloading the "Hello, World to the Object Board" .............177
  6.2 Embedded Linux Development ..............................................................180
    6.2.1 LED Test Program ..................................................................180
    6.2.2 Testing Buttons .......................................................................181
    6.2.3 UDP Network Programming ..................................................183
    6.2.4 Calling Math Function Library ..............................................188
    6.2.5 Thread Programming ...............................................................189
    6.2.6 Pipe Application Programming .............................................191
    6.2.7 "Hello, World" in C++ ..............................................................195
  6.3 A Simple Embedded Linux Driver Module ............................................196
    6.3.1 "Hello, Module" Source Code ...............................................196
    6.3.2 Adding the "Hello, Module" Program into the kernel code tree and Compiling it... 197
    6.3.3 Downloading & Running the "Hello, Module" Program ..........200
  6.4 Simple Linux Driver Examples ...............................................................201
    6.4.1 LED Driver ............................................................................201
    6.4.2 Button Driver ........................................................................205
  6.5 Porting Embedded Linux Programs .......................................................210
    6.5.1 Porting the MP3 player "madplay" ..........................................210
7. Configuring and Compiling Common Boot Loaders .....................................222
  7.1 Configuring and Compiling vivi .............................................................222
    7.1.1 Compiling vivi Using Default Configurations .......................222
    7.1.2 Making vivi boot from Nor Flash ...........................................225
  7.2 Configuring and Compiling U-boot ........................................................225
    7.2.1 Configuring and Compiling U-boot .......................................226
    7.2.2 Burning U-boot to the development board .........................227
8. Configuring and Compiling Linux Kernel ..................................................230
  8.1 Compiling the kernel with default configuration file .........................230
    8.1.1 Compiling the kernel .............................................................230
    8.1.2 Location of Linux Drivers .....................................................233
  8.2 Customizing Linux Kernel ....................................................................234
    8.2.1 Configuring CPU options ......................................................234
    8.2.2 Configuring Driver support for LCDs ....................................236
    8.2.3 Configuring Touch Screen .....................................................239
    8.2.4 Configuring USB mouse and keyboard ..................................240
8.2.5 Configuring Support for U-disk .................................................................241
8.2.6 Configuring USB Cameras ........................................................................243
8.2.7 Configuring driver for the CS8900 network card .....................................245
8.2.8 Configuring Sound Card Driver .................................................................249
8.2.9 Configuring SD/MMC card Driver ..............................................................252
8.2.10 Configuring LED and Button Drivers ......................................................252
8.2.11 Configuring Serial Port Driver .................................................................253
8.2.12 Configuring RTC Driver ...........................................................................254
8.2.13 Configuring yaffs filesystem support .......................................................254
8.2.14 Configuring EXT2/VFAT/NFS Filesystems .............................................256

8.3 Making yaffs Root Filesystem ........................................................................259

9. WinCE Development .........................................................................................261
9.1 Development Environment based on WinCE5.0 ............................................261
  9.1.1 Installing Platform Builder 5.0 (the 2007 latest patch included) .................261
  9.1.2 Importing and installing BSP .....................................................................274
  9.1.3 Installing wireless network card driver .....................................................277
  9.1.4 Example for Compiling Kernel Project ......................................................280
  9.1.5 Exporting SDK .........................................................................................282
  9.1.6 Installing Embedded Visual C++ (EVC) ....................................................289
  9.1.7 Installing EVC Patch and SDK .................................................................295
  9.1.8 Customizing WinCE Kernel .......................................................................304

9.2 Synchronous Communication with PC ..........................................................318
  9.2.1 Installing ActiveSync ................................................................................318
  9.2.2 Installing USB Driver for Synchronous Communication .........................323
  9.2.3 Copying files with ActiveSync ..................................................................324
  9.2.4 Using ActiveSync for Communication and Screen Shooting ..................327
  9.2.5 Remote Registry Editing ...........................................................................334

9.3 Creating "Hello,World" Program in EVC ......................................................335

9.4 Creating VS2005/2008 Application ...............................................................342
  9.4.1 Creating Project .......................................................................................343
  9.4.2 Setting Connected Development Board ..................................................345
  9.4.3 Compiling and Downloading Programs to the Development Board ...........348

9.5 LED Driver ......................................................................................................350
  9.5.1 Learning the hardware .............................................................................350
  9.5.2 Writing LED Streaming Driver .................................................................351
  9.5.3 Adding the LED Driver to the BSP and Compiling ....................................356
  9.5.4 Writing and Compiling LED Application ..................................................358
  9.5.5 Adding the LED testing program to the Kernel, and create a desktop shortcut..361

Conclusion: ..........................................................................................................362

Appendix IQt Embedded Graphic Development ..............................................363
Appendix II Refreshing System in the Command Line of BIOS ......................373
Appendix III Burning BIOS with SJF2440 ........................................................387

http://www.developmentboard.net  tel: 86-898-66761781  E-mail: info@developmentboard.net
Chapter 1 Introduction to MINI2440 development board

1.1 Brief introduction

The MINI2440 is an efficient ARM9 development board with a reasonable price, it characterizes simple method and high performance-price ratio. Based on the Samsung S3C2440 microprocessor, it embodies professional stable CPU core power source chip and reset chip to ensure the stability of the system operation. The PCB on the MINI2440 board is designed to be 4-layers board, adopting the ENIG technology and professional equal-length wiring to ensure the completeness of the signals of the key signal wire; and manufactured and released under stringent quality control plans. With the help of this detailed manual, users are supposed to become proficient in the development process of embedded Linux and WinCE operating system, they are supposed to get the foundation, so long as they have obtained the basic and necessary knowledge about the C language, in two weeks.

1.1.1 physical appearance of MINI2440
1.1.2 Hardware Features

- **CPU**
  - Samsung S3C2440A, 400MHz Main frequency, 533Mhz Peak Frequency

- **SDRAM**
  - 64M SDRAM on board
  - 32 bit data bus
  - SDRAM clock frequency can reach up to 100MHz

- **Flash Memory**
  - 64M Nand Flash on board, nonvolatile
  - 2M Nor Flash on board, nonvolatile

- **LCD control**
  - STN LCD Displays:
    - A 4-wire resistive touch screen interface is integrated on the board, via which the 4-wire resistive touch screen can be connected.
    - Supports 3 types of LCD panels: 4-bit dual scan, 4-bit single scan, and 8-bit single scan display type
    - Supports the monochrome, 4 gray levels, and 16 gray levels
    - Supports 256 colors and 4096 colors for color STN LCD panel
    - Supports multiple screen size
  
  Typical actual screen size: 640 x 480, 320 x 240, 160 x 160, and others

- **TFT LCD Displays:**
  - Supports 1, 2, 4 or 8-bpp (bit per pixel) palletized color displays for TFT
  - Supports 16, 24-bpp non-palletized true-color displays for color TFT
  - Supports maximum 16M color TFT at 24bit per pixel mode
  - Supports multiple screen size

  Typical actual screen size: 1024x768 640 x 480, 320 x 240, 160 x 160, and others

- **Interfaces and Resources**
  - One 10MM Ethernet RJ-45 interface (DM9000 ethernet chip adopted)
  - 3 serial ports
  - One USB Host
  - One USB Slave (B-type interface)
  - One SD card interface

http://www.developmentboard.net    tel: 86-898-66761781    E-mail: info@developmentboard.net
* One stereo audio output interface, one MIC interface
* One 10pin (2.0mm space) JTAG interface
* 4 user LEDs
* 6 user buttons
* One PWM control buzzer
* One adjustable resistance, used for AD conversion test
* One I2C bus AT24C08 chip, used for I2C bus test
* One 20pin (2.0mm space) camera interface
* RTC battery on board
* Power supply interface, with switch and indicator

● System clock source
* 12M passive crystal

● RTC
* Internal real time clock, battery backed

● Expansion interfaces
* One 34pin 2.0mm GPIO interface
* One 40pin 2.0mm system bus interface

● Dimension
* 100 x 100 (mm)

● OS supported
* Linux 2.6.13
* WindowsCE.NET 5.0

1.1.3 CD Contents
1. ADS1.2 installer
2. H-JTAG burning & debugging tool
3. SJF2440 (Flash Burning tool for Windows)
4. Jflash-2440 (Nand Flash Burning tool for Linux, source code provided)
5. Serial tools CRT, dnw
6. A software that used to transform picture to C language array
7. USB driver (installed and used under Windows XP/2000)
8. vivi source code, the bootloader used for Linux
9. Simplest LED test program
10. Test program 2440test, containing the project files of ADS1.20, source code provided, used for the tests for: interrupt mode button test, RTC test, AD conversion test, IIS audio playing wav test, IIS audio recording test, touch screen test, I2C bus reading/writing AT24C08 test, Samsung 3.5” LCD, 640x480 TFT test, etc..

11. WindowsCE BSP and example project files

12. Linux development tools and kernel source code package
   - arm-linux-gcc-3.3.2 used for compiling Qtokia
   - arm-linux-gcc-3.4.1 used for compiling kernel
   - arm-linux-gcc-2.95.3 used for compiling vivi
   - mkyaffsimage yaffs filesystem image maker
   - linux-2.6.13 for MINI2440 kernel source code, including DM9000 driver, TFT LCD driver, audio card driver, touch screen driver, YAFFS source code, SD card driver, RTC driver, expansion serial port driver, USB camera driver, USB mouse, keyboard, U-disk driver.

13. Embedded graphic interface Qtokia source code, embedded browser source code

14. Schematics (Protel99SE/PDF format)

15. Users Manual (pdf)

**1.2 Hardware Resources**

**1.2.1 Jumpers**

There is only one jumper (J2) on the MINI2440 development board, it is used to select the input voltage of the LCD driver board.

**1.2.2 Interfaces**

The layout of the interfaces on the MINI2440 are shown in the picture below:
1.2.2.1 Address Space Layout and Chip Selection signal definition

The S3C2440 CPU chip supports two kinds of boot modes: booting at the Nandflash (which is used by the MINI2440) and booting at the Nor flash. The allocation of the storage space of the chip selections is different in the two boot modes. As we see in the picture below:
In the picture above, the left part shows the storage allocation of the nGCS0 chip selection in the Nor Flash boot mode. While the right part of the picture shows the storage allocation in the NandFlash boot mode.

NOTE: the SFR Area refers to SFR (special function registry) address control

The following illustrates the device address space layout and its chip Selection signal definition.

Before going into the device address illustration, we must know that the nGCS0 chip selection maps to different devices in different boot modes. We can see from the picture above that:

1. In the NAND Flash boot mode, the internal 4K Bytes BootSram is mapped to the space of the nGCS0 chip selection.
2. Under the Nor Flash boot mode, the Nor Flash, i.e., the external storage that is linked to the
nGCS0, is mapped to the space of the nGCS0 chip selection.

The address space of the SDRAM is: 0x30000000 ~ 0x34000000

1.2.2.2 SDRAM

Two 32MB SDRAM chip (model: HY57561620FTP) are utilized on the MINI2440. The two chips are joined up in parallel to produce 32bit data bus width so as to increase access speed, both of them use the nGCS6 as the chip selection, and thus, according to the CHAPTER 5-2 in the manual of the S3C2440, the physical offset of these two SDRAM chips are determined to be 0x30000000.

The picture below shows the schematics of the SDRAM chips:

1.2.2.3 Flash Storage

Two Flash storage chips are used on the MINI2440. One is 2MB Nor Flash (Model: SST39VF1601), the other is 64MB Nand Flash (Model: K9F1208). Both of these two storages can be used as boot source, i.e., the system can boot from either Nor Flash or Nand Flash. by switching the switch S2, we can select either Nor Flash or Nand Flash as boot source. Though the Nor Flash is not necessary for the most final products, we keep it on the MINI2440 for users’ development and learning.

There is no address wire in the Nand Flash, it is connected to the CPU via a special controller, with a data bus width of 8-bit. A 8-bit data bus width does not mean a slow access speed, in fact...
most of the u-disks and SD cards are made of Nand Flash chips.

Referring to the schematic below, we can see that the Nor Flash is connected to the CPU via 22 address wires (A1-A22) and 16 data wires. We can see that the address starts from A1, which determines that the minimum write/read unit is 2 byte, therefore, the Nor Flash supported by MINI2440 maximumly 8Mbyte. In fact, on the MINI2440, only A1-A20 are used, therefore the pins on the SST29V1601 which are connected to A21 and A22 are floating.

1.2.2.4 Power System

The power system on the MINI2440 is quit simple, a 5V power supply is used directly, through several voltage regulators the 5V power supply is reduced to 3.3V, 1.8V and 1.25V.

Since the MINI2440 is not specially designed for hand-held mobile devices, it does not embody complete power management circuit. The power supply is controlled by the power switch S1.
For the convenience of connecting other power supply, another power socket (CON8) has been implemented on the MINI2440. The socket can be found beside the TTL serial ports, it is a white 2.0mm 4 pin connector, the first and the last pins are 5V, and the two pins between the 5V are GND.
1.2.2.5 Reset Circuit

A professional reset chip MAX811 is used on the MINI2440 to realise low level reset:

![Reset Circuit Diagram]

1.2.2.6 User LED

LED is a kind of commonly used state indicator on development boards. On the MINI2440 there are totally 4 user programmable LEDs, which are directly connected to the GPIO of the CPU and enabled (lighten) by low power level. The resources occupied by the LEDs are listed in the table below:

<table>
<thead>
<tr>
<th></th>
<th>LED1</th>
<th>LED2</th>
<th>LED3</th>
<th>LED4</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPIO</td>
<td>GPB5</td>
<td>GPB6</td>
<td>GPB7</td>
<td>GPB8</td>
</tr>
<tr>
<td>Multiplexed as</td>
<td>nXBACK</td>
<td>nXREQ</td>
<td>nXDACK1</td>
<td>nDREQ1</td>
</tr>
<tr>
<td>Name in the schematic</td>
<td>nLED_1</td>
<td>nLED_2</td>
<td>nLED_3</td>
<td>nLED_4</td>
</tr>
</tbody>
</table>

1.2.2.7 User Keys

There are totally 6 user keys on the MINI2440, which are directly led out from the CPU interrupt pins and triggered by low power level. These pins can also be multiplexed as GPIO or interfaces with special function. In order to make them usable for other functions, these pins are also led out via the connector CON12. The definition of the 6 user keys and the CON12 are shown in the table below:

![User Keys Diagram]
### 1.2.2.8 A/D Output

On the MINI2440, totally there are 4 A/D conversion channels that can be led out, which are located on the CON4-GPIO interface. For the convenience of testing ADC, the AIN0 has been connected to the adjustable resistance W1. The schematic is as shown below:

![Schematic diagram of A/D output](image)

### 1.2.2.9 PWM

The buzzer on the MINI2440 is used for testing PWM. The schematic is shown as below. The
GPB0 can be set as PWM output by software.

1.2.2.10 Serial Ports

The S3C2440 microcontroller itself has 3 serial ports in total, namely UART0, UART1, UART2. The UART0 and UART1 can be combined as a full-functional serial port. In most practices, only three simply serial port functions are used, i.e., send (TXD) and receive (RXD), they are respectively related to the connectors CON1, CON2, and CON3. CON1, CON2, and CON3 are three TTL serial ports directly led out from the CPU. For the convenience of users, UART0 has been converted to RS232 and led to COM0.
1.2.2.11 USB Interfaces

There are two USB interfaces on the MINI2440. One is USB Host interface and the other is USB Device interface. The USB Host is the same with the USB interfaces on a PC, it can be used to connect USB camera, USB keyboard, USB mouse, U-disk and etc.; The USB Device interface is usually used for downloading programs to the MINI2440 or WinCE synchronization. For the convenience of controlling the communication between the USB Device and PC, a signal USB_EN is set as shown in the picture below, which uses the CPU resource GPC5.

![USB Interfaces Diagram](image_url)

1.2.2.12 LCD Interface

The LCD Interface on the MINI2440 is a 41Pin 0.5mm white socket, which contains most of the commonly used LCD controlling signals (line-field scan, clock, enable, etc..) and complete RGB data signals (the RGB output is 8:8:8, supports maximumly 16MP LCD). The PWM output (GPB1 can be configured by register as PWM) and reset signal (nRESET) are led out for the

http://www.developmentboard.net  tel: 86-898-66761781  E-mail: info@developmentboard.net
convenience of testing. The LCD_PWR is the back light controlling signal.

The pins 37, 38, 39, 40 are the interface for the 4-wire touch screen, through which the touch screen can be connected directly.

The jumper J2 in the picture below is the LCD driver board power voltage selector. Currently all of our driver boards are supplied with 5V power supply.

1.2.2.13 EEPROM

On the MINI2440, a 256 byte EEPROM chip AT24C08 is connected to the I2C signal pins on the CPU, it is mainly used for testing the I2C bus. No specific parameters are stored in the EEPROM.
1.2.2.14 Ethernet Interface

An 10/100M self-adapting ethernet chip DM9000 and a ethernet connector RJ45 are used on the MINI2440. The RJ45 connector contains a coupled inductor, therefore we do not need a network transformer but a simple ethernet cable to connect the MINI2440 development board to a router or switch.
1.2.2.15 Audio Interface

The S3C2440 micro controller embodies an I2S bus interface, to which we can directly connect a 8/16 bit stereo CODEC. On the MINI2440, an I2S bus-based UDA1341 audio chip is used as stereo CODEC to realise audio codec system. The initialization and configuration of the registries in the UDA1341 chip are controlled by the L3-bus. On the MINI2440, the ports GPB2, GPB3, GPB4 on the CPU are taken to simulate the L3MODE, L3DATA, L3CLOCK signals of the L3-bus, they are no longer used after initializing the UDA1341 chip, therefore these three wires can also be simulated by using a simple single chip.

A 3.5mm stereo hole connector is used on the MINI2440 as audio output interface. The audio input system consists of two channels: one MIC on the board and one 2.0mm CON10 connector. The drivers of these two channels are different from each other, at this moment the MIC is not working yet, only the CON10 connector can be used for recording. Anyway, sooner or later we are going to get the MIC driven and usable.

1.2.2.16 JTAG Interface

A JTAG interface is essential in a bare-mental environment, where the serial ports and USB...
interfaces are not driven yet. In bare-mental environment we can take the JTAG interface to
download the first program, i.e., boot loader, to the development board.

Moreover, the most common usage of a JTAG interface is for single step debugging. Most of
the common emulators (debuggers) like JLINK, ULINK and etc take the JTAG interface as the
connector to the board.

A standard JTAG interface consists of six wires, namely TMS, TCK, TDI, TDO (respectively,
mode selection, clock, data in, and data out), power and GND. For the convenience of debugging,
most of the debuggers also provide a reset signal.

Therefore, a standard JTAG interface is an interface that contains the JTAG signals
mentioned above, but not a modal 20-pin or 10-pin connector. No matter how many pins a JTAG
interface embodies, whatever its shape is, so long as it contains all of the JTAG signals, this
interface can be recognized as JTAG interface. On the MINI2440 we provide a 10 pin JTAG
interface which contains complete standard JTAG signals, the definition of the 10 pins are shown in
the picture below.

P.S.: For Linux or WinCE-dedicated developers, the JTAG interface is nonsensel and useless,
as most of the development boards always provide complete BSP, which contains the most
commonly used serial ports, ethernet ports and USB interfaces. When the system is loaded with
working Linux or WinCE operating system, users can debug with the functions provided by these
advanced operating systems without JTAG interface. Due to the complex structure and plentiful
interfaces of the operating systems, single step debugging is nonsensel but just like searching for a
needle in a haystack. Take a PC for instance, you’ve never seen a developer debugging Windows
XP or Linux driver for the PCI interface with a simulator connected to the main board, do you? We
have seen the phrase "Driver PORTING" a lot, as most developers do not develop drivers all by
themselves, but by porting others' drivers.

JTAG interface is only useful for those bare-mental system or simple OS (uCOS2 for instance)
developers. On most development boards, the boot loader or BIOS is already installed as a complete
system, therefore single step debugging is not needed.
1.2.3.17 GPIO

"GPIO" is the abbreviation of General Purpose Input Output interfaces. The MINI2440 embodies a 2.0mm 34-pin GPIO interface, named as CON4 shown in the picture below.

In fact, the CON4 connector contains not only some unused GPIO pins, but also some other CPU pins such as AD0-AIN3, CLKOUT, etc.. The interfaces like SPI, I2C, GPB0 and GPB1, though named as special function interface as you see in the picture, are actually GPIOs too, their functions can be changed by setting the CPU registry.

Detailed interface resources are shown in the table below:

<table>
<thead>
<tr>
<th>Pin</th>
<th>Name</th>
<th>Description</th>
<th>Pin</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VDD5V</td>
<td>5v power (in/out)</td>
<td>2</td>
<td>VDD33V</td>
<td>3.3v power (output)</td>
</tr>
</tbody>
</table>
1.2.3.18 CMOS Camera Interface

The CMOS camera interface on the MINI2440 is a 2.0mm 10pin socket, with which users can directly use the 1.3 Mpixels CMOS camera module on our website. In fact, the CMOS camera module does not contain any circuit but a simple ZT130G2 camera module. The definition of the pins on the CMOS camera interface is shown in the schematic below.

NOTE: The CMOS camera interface is actually a multiplexing port, by setting corresponding registries we can use it as GPIO. The table below shows the multiplexing GPIO function of the pins:

<table>
<thead>
<tr>
<th></th>
<th>GND</th>
<th>Ground</th>
<th>4</th>
<th>NRESET</th>
<th>Reset signal (output)</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>AIN0</td>
<td>AD input channel 0</td>
<td>6</td>
<td>AIN1</td>
<td>AD input channel 1</td>
</tr>
<tr>
<td>7</td>
<td>AIN2</td>
<td>AD input channel 2</td>
<td>8</td>
<td>AIN0</td>
<td>AD input channel 3</td>
</tr>
<tr>
<td>9</td>
<td>EINT0</td>
<td>EINT0/GPF0</td>
<td>10</td>
<td>EINT1</td>
<td>EINT1/GPF1</td>
</tr>
<tr>
<td>11</td>
<td>EINT2</td>
<td>EINT2/GPF2</td>
<td>12</td>
<td>EINT3</td>
<td>EINT3/GPF3</td>
</tr>
<tr>
<td>13</td>
<td>EINT4</td>
<td>EINT4/GPF4</td>
<td>14</td>
<td>EINT5</td>
<td>EINT5/GPF5</td>
</tr>
<tr>
<td>15</td>
<td>EINT6</td>
<td>EINT6/GPF6</td>
<td>16</td>
<td>EINT8</td>
<td>EINT8/GPG0</td>
</tr>
<tr>
<td>17</td>
<td>EINT9</td>
<td>EINT9/GPG1</td>
<td>18</td>
<td>EINT11</td>
<td>EINT11/GPG3/nSS1</td>
</tr>
<tr>
<td>19</td>
<td>EINT13</td>
<td>EINT13/GPG5/SPIMISO1</td>
<td>20</td>
<td>EINT14</td>
<td>EINT14/GPG6/SPIMOSII</td>
</tr>
<tr>
<td>21</td>
<td>EINT15</td>
<td>EINT15/GPG7/SPICLK1</td>
<td>22</td>
<td>EINT17</td>
<td>EINT17/GPG9/nRST1</td>
</tr>
<tr>
<td>23</td>
<td>EINT18</td>
<td>EINT18/GPG10/nCTS1</td>
<td>24</td>
<td>EINT19</td>
<td>EINT19/GPG11</td>
</tr>
<tr>
<td>25</td>
<td>SPIMISO</td>
<td>SPIMISO/GPE11</td>
<td>26</td>
<td>SPIMOSI</td>
<td>SPIMOSI/EINT14/GPG6</td>
</tr>
<tr>
<td>27</td>
<td>SPICLK</td>
<td>SPICLK/GPE13</td>
<td>28</td>
<td>nSS_SPI</td>
<td>nSS_SPI/EINT10/GPG2</td>
</tr>
<tr>
<td>29</td>
<td>I2CSCl</td>
<td>I2CSCl/GPE14</td>
<td>30</td>
<td>I2CSDA</td>
<td>I2CSDA/GPE15</td>
</tr>
<tr>
<td>31</td>
<td>GPB0</td>
<td>TOUT0/GPB0</td>
<td>32</td>
<td>GPB1</td>
<td>TOUT1/GPB1</td>
</tr>
<tr>
<td>33</td>
<td>CLKOUT0</td>
<td>CLKOUT0/GPH9</td>
<td>34</td>
<td>CLKOUT1</td>
<td>CLKOUT1/GPH10</td>
</tr>
</tbody>
</table>

http://www.developmentboard.net    tel: 86-898-66761781    E-mail: info@developmentboard.net
<table>
<thead>
<tr>
<th>Pin</th>
<th>Name</th>
<th>Multiplexing function</th>
<th>Pin</th>
<th>Name</th>
<th>Multiplexing function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>I2CSDA</td>
<td>GPE15</td>
<td>2</td>
<td>I2CSCL</td>
<td>GPE14</td>
</tr>
<tr>
<td>3</td>
<td>EINT20</td>
<td>GPG12</td>
<td>4</td>
<td>CAMRST</td>
<td>GPJ12</td>
</tr>
<tr>
<td>5</td>
<td>CAMCLK</td>
<td>GPJ11</td>
<td>6</td>
<td>CAM_HREF</td>
<td>GPJ10</td>
</tr>
<tr>
<td>7</td>
<td>CAM_VSYNC</td>
<td>GPJ9</td>
<td>8</td>
<td>CAM_PCLK</td>
<td>GPJ8</td>
</tr>
<tr>
<td>9</td>
<td>CAMDATA7</td>
<td>GPJ7</td>
<td>10</td>
<td>CAMDATA6</td>
<td>GPJ6</td>
</tr>
<tr>
<td>11</td>
<td>CAMDATA5</td>
<td>GPJ5</td>
<td>12</td>
<td>CAMDATA4</td>
<td>GPJ4</td>
</tr>
<tr>
<td>13</td>
<td>CAMDATA3</td>
<td>GPJ3</td>
<td>14</td>
<td>CAMDATA2</td>
<td>GPJ2</td>
</tr>
<tr>
<td>15</td>
<td>CAMDATA1</td>
<td>GPJ1</td>
<td>16</td>
<td>CAMDATA0</td>
<td>GPJ0</td>
</tr>
<tr>
<td>17</td>
<td>VDD33V</td>
<td>3.3V power</td>
<td>18</td>
<td>VDD_CAM</td>
<td>VDD_CAM</td>
</tr>
<tr>
<td>19</td>
<td>VDD18V</td>
<td>1.8V power</td>
<td>20</td>
<td>GND</td>
<td>Ground</td>
</tr>
</tbody>
</table>
1.2.3.19 System Bus Interface

The system bus, named as CON5 on the MINI2440, contains 16 data wires (D0-D15), 8 address wires (A0-A6, A24) and some controlling signal wires (chip selection, R/W, reset, etc.). The CON5 connector can also supply 5V power to external devices. The definition of the pins on the system bus is shown in the table below:

<table>
<thead>
<tr>
<th>Pin</th>
<th>Name</th>
<th>Description</th>
<th>Pin</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>VDD5V</td>
<td>5V power</td>
<td>2</td>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>3</td>
<td>EINT17</td>
<td>Interrupt 17 (input)</td>
<td>4</td>
<td>EINT18</td>
<td>Interrupt 18 (input)</td>
</tr>
<tr>
<td>5</td>
<td>EINT3</td>
<td>Interrupt 3 (input)</td>
<td>6</td>
<td>EINT9</td>
<td>Interrupt 9 (input)</td>
</tr>
<tr>
<td>7</td>
<td>nGCS1</td>
<td>Chip selection 1 physical address 0x08000000</td>
<td>8</td>
<td>nGCS2</td>
<td>Chip selection 2 physical address 0x10000000</td>
</tr>
<tr>
<td>9</td>
<td>nGCS3</td>
<td>Chip selection 3 physical address 0x18000000</td>
<td>10</td>
<td>nGCS5</td>
<td>Chip selection 5 physical address 0x28000000</td>
</tr>
</tbody>
</table>

http://www.developmentboard.net   tel: 86-898-66761781   E-mail: info@developmentboard.net
<table>
<thead>
<tr>
<th></th>
<th>LnOE</th>
<th>Read-enable signal</th>
<th></th>
<th>LnWE</th>
<th>Write-enable signal</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>LnOE</td>
<td>Read-enable signal</td>
<td>12</td>
<td>LnWE</td>
<td>Write-enable signal</td>
</tr>
<tr>
<td>13</td>
<td>nWAIT</td>
<td>Wait</td>
<td>14</td>
<td>nRESET</td>
<td>Reset</td>
</tr>
<tr>
<td>15</td>
<td>nXDACK0</td>
<td>nXDACK0</td>
<td>16</td>
<td>nXDREQ0</td>
<td>nXDREQ0</td>
</tr>
<tr>
<td>17</td>
<td>LADDR0</td>
<td>Address 0</td>
<td>18</td>
<td>LADDR1</td>
<td>Address 1</td>
</tr>
<tr>
<td>19</td>
<td>LADDR2</td>
<td>Address 2</td>
<td>20</td>
<td>LADDR3</td>
<td>Address 3</td>
</tr>
<tr>
<td>21</td>
<td>LADDR4</td>
<td>Address 4</td>
<td>22</td>
<td>LADDR5</td>
<td>Address 5</td>
</tr>
<tr>
<td>23</td>
<td>LADDR6</td>
<td>Address 6</td>
<td>24</td>
<td>LADDR7</td>
<td>Address 7</td>
</tr>
<tr>
<td>25</td>
<td>LDATA0</td>
<td>Data 0</td>
<td>26</td>
<td>LDATA1</td>
<td>Data 1</td>
</tr>
<tr>
<td>27</td>
<td>LDATA2</td>
<td>Data 2</td>
<td>28</td>
<td>LDATA3</td>
<td>Data 3</td>
</tr>
<tr>
<td>29</td>
<td>LDATA4</td>
<td>Data 4</td>
<td>30</td>
<td>LDATA5</td>
<td>Data 5</td>
</tr>
<tr>
<td>31</td>
<td>LDATA6</td>
<td>Data 6</td>
<td>32</td>
<td>LDATA7</td>
<td>Data 7</td>
</tr>
<tr>
<td>33</td>
<td>LDATA8</td>
<td>Data 8</td>
<td>34</td>
<td>LDATA9</td>
<td>Data 9</td>
</tr>
<tr>
<td>35</td>
<td>LDATA10</td>
<td>Data 10</td>
<td>36</td>
<td>LDATA11</td>
<td>Data 11</td>
</tr>
<tr>
<td>37</td>
<td>LDATA12</td>
<td>Data 12</td>
<td>38</td>
<td>LDATA13</td>
<td>Data 13</td>
</tr>
<tr>
<td>39</td>
<td>LDATA14</td>
<td>Data 14</td>
<td>40</td>
<td>LDATA15</td>
<td>Data 15</td>
</tr>
</tbody>
</table>
1.3 Linux Features

- **Version**
  - Linux2.6.13

- **Supported Filesystems**
  - yaffs (readable & writable filesystem, recommended)
  - cramfs (compressed read only filesystem, recommended only when online update is not needed.)
    - Ext2
    - Fat32
    - NFS (Network filesystem, recommended when developing drivers and applications)

- **Fundamental Drivers**
  - 3 serial port standard driver
  - DM9000 driver
  - Sound driver
  - RTC driver
  - User LED driver
  - USB Host driver
  - Common LCD driver
  - Touch screen driver
  - USB camera driver
  - USB mouse, keyboard, U-disk, mobile disk driver
  - SD card driver, supporting SD memory up to 2G

- **Linux applications and service programs**
  - busybox 1.2.0 (Linux utilities, contains common Linux instructions)
  - Telnet, Ftp, inetd (telnet tools and services)
  - boa (web server)
  - madplay (console oriented mp3 player)
  - snapshot (console oriented image capture software)
  - ishow (console oriented image browser)
  - ifconfig, ping, route (common network tool commands)

- **Embedded graphic system (Provided as source code)**
  - Qt/Embedded

---

http://www.developmentboard.net  tel: 86-898-66761781  E-mail: info@developmentboard.net
1.4 WindowsCE Features

- **Version**
  - WindowsCE.net 5.0

- **Features**
  - DM9000 Ethernet card driver source code
  - USB mouse, keyboard, U-disk, mobile disk driver
  - Audio driver
  - SD card driver
  - RTC
  - Registry Saving
  - Saving data to spare Flash space in the event of power down
  - Revolving screen

- **Default system features**
  - XP style interface
  - Windows Media Player 9.0 (supporting mp3, mpeg2, mpeg4, wmv, wav, etc)
  - Superplayer (similar to the Storm Codec in Windows)
  - Image browser, wordpad
  - IE6 browser
  - ftp, telnet, httpd server
  - Serial port debugging assistant