



Porting the Windows* CE Operating System to StrongARM** SA-1100 Based Platforms

Application Note

November 1998



Information in this document is provided in connection with Intel products. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Except as provided in Intel's Terms and Conditions of Sale for such products, Intel assumes no liability whatsoever, and Intel disclaims any express or implied warranty, relating to sale and/or use of Intel products including liability or warranties relating to fitness for a particular purpose, merchantability, or infringement of any patent, copyright or other intellectual property right. Intel products are not intended for use in medical, life saving, or life sustaining applications.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

The SA-1100 may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order.

Copies of documents which have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling 1-800-548-4725 or by visiting Intel's website at <http://www.intel.com>.

Copyright © Intel Corporation, 1998

*Third-party brands and names are the property of their respective owners.

**ARM and StrongARM are trademarks of Advanced RISC Machines, Ltd.



Contents

1.0	Introduction.....	1
1.1	Related Documentation.....	1
1.2	Host System Requirement	1
1.3	Target Requirement	2
2.0	Porting Windows* CE to StrongARM** SA-1100 Based Platform	2
2.1	Porting Example - SA-1100 Development Module.....	4
3.0	Loading an Image of Windows CE to the Hardware Platform	5
3.1	Multiple Boot Selections	6

Tables

1	Porting the Windows CE Operating System.....	4
2	Downloading the Image to the Hardware Platform.....	5

1.0 Introduction

The Intel StrongARM** SA-1100 is supported in the *Windows CE 2.1 Enhancement Pack for the Embedded ToolKit* (Enhancement Pack). The *Microsoft® Windows® CE Embedded ToolKit for Visual C++® 5.0* (ETK) provided by Microsoft comes with sample code for the OEM Abstraction Layer (OAL) and the device drivers required to build the Windows* CE operating system for the ODO reference platform. Sample code on the ETK is provided for every processor supported on the ODO reference platform including the SA-1100. This results in a good starting point for any porting effort for a SA-1100 based platform.

Currently, the only other SA-1100 based reference platform which has the Windows CE operating system ported to it is the SA-1100 development board (order number DE-1S110-OA).

Note: This document will be updated as the Windows CE operating system is ported to other Intel reference platforms.

This document lists the steps to port the Windows CE operating system to a StrongARM based platform and uses the SA-1100 development module as an example. Information available in other documents will not be repeated in detail in this document, but referenced appropriately.

1.1 Related Documentation

The following documentation are referenced in this application note:

- *SA-1100 Microprocessor Evaluation Platform User's Guide*
- *SA-1100 Microprocessor Firmware User's Guide*
- *StrongARM** SA-1100 Portable Communication Microcontroller Data Sheet*

1.2 Host System Requirement

The following list describes the host system requirements for porting the Windows CE operating system:

- Recommended development host system—Pentium® or compatible processor with at least 1 GB of free hard drive space and 128MB of DRAM running the Windows NT 4.0 Workstation with Service Pack 3.

For more information about host system requirements and the installation procedure, see the on-line documentation with the ETK and the Release Notes on the ETK CD.

Note: To save disk space, Intel recommends installing only the SA-1100 related files from the ETK.

- Version 2.1 of the ETK—The SA-1100 is only supported in version 2.1 of the ETK. Installation of version 2.1 of the ETK requires prior installation of version 2.0 of the ETK.

To save disk space, see the installation procedure in the Release Notes on the ETK CD to do a minimal installation of the ETK version 2.0.

- ARM Software Development Toolkit SDT 2.11A or newer—The ARM SDT is not necessary for porting the Windows CE operating system, but will be required to prepare the board for the



Windows CE operating system boot loading in certain cases. See the PCMCIA flash card image loading option described in Section 3.0.

1.3 Target Requirement

The following components are required for a target:

- SA-1100 development module with Angel 1.02 or newer on the board, with 16 MB of RAM.
- Null modem cable—9 pin D-Sub female to 9 pin D-Sub female (provided with the SA-1100 development module).
- Flash Management Utility—this selection is optional but may be required for the PCMCIA flash card image loading option mentioned in Section 3.0.
- 12MB Intel compatible linear flash—this selection is optional but may be required for the PCMCIA flash card image loading option mentioned in Section 3.0.
- PCMCIA slot for programming the linear flash card—this selection is optional but may be required for the PCMCIA flash card image loading option mentioned in Section 3.0.
- Linear flash card programming utility—this utility copies the binary image file directly to the flash card and is available from the Intel web site. This selection is optional but may be required for the PCMCIA flash card image loading option mentioned in Section 3.0.

2.0 Porting Windows* CE to StrongARM** SA-1100 Based Platform

This section describes some of the basic steps required to port the Windows CE operating system to a custom platform based on the SA-1100. The SA-1100 development module is used as an example. Some of the board-specific information in this section may not apply to other custom boards. It is assumed that the *Windows CE 2.1 Enhancement Pack for the Embedded ToolKit* (Enhancement Pack) has been installed for the SA-1100 development board with the build tree for the ODO reference platform.

For more information, see the Adapting Windows CE to Your Target Platform section in the on-line documentation for details.

Note: All commands, directory names, and file names are case insensitive in the Windows and Windows CE operating system.

1. To create a new platform directory, make a copy of the platform directory that is the most similar to your intended platform.

Example: If you have the Brutus directory from Intel, copy that entire directory to your platform directory.

```
cd \Wince210\Platform
xcopy Brutus <YOUR_PLATFORM> /E /V /I
```

2. Set up batch files for the user specific environment:
 - a. Create directory Developr\USERNAME where USERNAME is your NT user name.

- b. Create a batch file called setenv.bat in this directory. The setenv.bat in the Developr\User directory is for setting up user-specific environments that will over-write any environment setup done in the wince.bat batch file and in the platform specific environment settings. Use this process to experiment with different variations without affecting the setup file. These modifications can be added to the setup file and specified for a particular user if more than one user is sharing the build tree. The contents of this file build environment setups used for creating a custom image for each user.
 - c. Create %_TGTPLAT%.bat (Brutus.bat)file in %_TGTPLATROOT% (\Win210\Platform\Brutus) directory and set up platform specific environment variables for devices (such as, PLAT_PCMCIA) to be included or excluded for the build (using the BUILD, BUILDER, or MAKEIMG command).
 3. Clean up the build environment from any previous builds.
 - a. Delete the directory %_FLATRELEASEDIR% (by default this is \Wince210\Release). If this directory is not deleted, the Build Tools application may not allow a different binary image to build in the directory %_FLATRELEASEDIR%.
 - b. Delete the file %_TGTPROJ%.BIF created in the %_TGTPLATROOT% directory during the previous build process.
 4. The Enhancement Pack includes a collection of shortcuts, each of which creates a build window for a specific hardware development platform (containing one of the supported types of processors), for the minimum shell configuration MINSHELL.
 - a. Select the existing project shortcut, SA-1100 MINSHELL, in the program group Windows CE 2.1 Enhancement Pack.
 - b. Make a copy of the shortcut and rename it to identify the platform to be built for. For example, the new name would be:


```
BRUTUS MINSHELL
```
 - c. In the tab Shortcut for the icon's properties, modify the target command line parameters that follow the Wince.bat command. The new command should appear similar to:


```
Wince.bat ARM SA-1100 CE MINSHELL BRUTUS
```
 - d. Double-click the project icon to create a new build window. The build-related environment variables will be set to support the project and target platform. Enter the following command to build the project:


```
Blldemo
```
 - e. Modify the Binary Image Builder (BIB) files, which will be used by Makeimg.exe for your platform, project, and memory configuration. The files that need to be modified are:


```
Common.bib (%_PROJECTROOT%\Cesysgen\Oak\Files)
Project.bib (%_PROJECTROOT%\Oak\Files)
Config.bib (%_TARGETPLATROOT%\Files)
Platform.bib (%_TARGETPLATROOT%\Files)
```
- Note:** For an example for the SA-1100 development module, see the files in the Brutus build tree. For memory mapping information of the SA-1100 development module, see the *StrongARM** SA-1100 Microprocessor Evaluation Platform User's Guide*.
5. If the build completes without errors, start modifying, replacing, or adding files for your specific hardware. The Windows CE operating system architecture simplifies the development

process by isolating most platform-specific code in a layer called the OEM Adaptation Layer (OAL). The OAL represents a layer between the Windows CE operating system kernel and the target platform's firmware. To create a version of the Windows CE operating system for the target platform and its CPU, only the implementations of the functions that make up the OAL must be provided. Refer to the Developing the OEM Adaptation Layer chapter in the on-line books that comes with the ETK.

2.1 Porting Example - SA-1100 Development Module

The ODO build tree is used as the starting point for porting the Windows CE operating system to the SA-1100 development module. This results in the SA-1100 development module build tree having the OAL layer and three basic drivers—the keyboard, the LCD, and the touch screen ported for the SA-1100 development module. When files are deleted, added, or modified in a subdirectory, a file named changes.txt is created in that subdirectory that lists the changes that have been made.

Use the following table for porting the Windows CE operating system to your platform.

Table 1. Porting the Windows CE Operating System

Action	Description	File location
Porting the OAL	The firmware file in the HAL\ARM directory (fwbrut.s) contains the processor initialization code. The initialization code sets the proper processor mode, disables the interrupts, makes sure that the MMU is turned off, sets up a stack pointer for debug messages, points to the memory map table (OEMAddressTable) and jumps to KernelStart. KernelStart is the entry point into the kernel for the Windows CE operating system. Its functions are to initialize the MMU based on the OEMAddressTable, perform other operating system set up functions, and start executing threads.	%_TARGETPLATROOT%\KERNEL\HAL and %_TARGETPLATROOT%\KERNEL\HAL\ARM directory (in the case of the SA-1100 development module, the files are at: \\WINCE210\PLATFORM\BRUTUS\KERNEL\HAL and ...\\HAL\ARM).
Mapping memory	The SA-1100 development module loads the image into the DRAM and executes it. The beginning of DRAM is virtual 0x8000 0000, which is mapped to physical 0xD000 0000. Since the memory for the SA-1100 development module is in 4 banks of 4 MB each, the other 4 MB of the 8 MB image built is loaded at virtual 0x8040 0000 which is mapped to physical 0xD8000000. The RAM at 0xC000 0000 and 0xC800 0000 are mapped at 0x8C00 0000 and 0x8C40 0000 giving 8 MB of RAM space for the Windows CE operating system.	MAP1100.H—Contains the memory map for the hardware platform. Config.Bib— Must reflect the memory set up when the image is built.
Sourcing interrupts	Interrupts ^a on the SA-1100 development module use the IRQ. Since only one physical interrupt line is used there is a single interrupt handler called the OEMInterruptHandler. The source of the interrupt is determined by checking the Interrupt Pending Register and the GPIO registers.	Int1100.C—Contains the OEMInterruptHandler.
Keyboard, LCD, and touch screen drivers	The keyboard, LCD, and touch screen drivers are ported by modifying the files in the %_TARGETPLATROOT%\Drivers directory. ^b	%_TARGETPLATROOT%\Drivers directory—Contains files to be modified for the keyboard, LCD, and touch screen drivers.

a. For more information on interrupts, see the Interrupt Controller section in the System Control Module chapter in the *StrongARM** SA-1100 Portable Communication Microcontroller Data Sheet*.

b. For more information about the keyboard, LCD, and touch screen drivers, see the Device Driver Architecture section in the on-line documentation.

3.0 Loading an Image of Windows CE to the Hardware Platform

Once the files have been ported and an image is built, there are a number of ways to download the image of the Windows CE operating system to the hardware platform. Three methods are listed below in Table 2.

Table 2. Downloading the Image to the Hardware Platform

Action	Description	Drawback
Porting the boot loader to the hardware platform	A sample bootloader is available for the ODO reference platform in the %_WINCEROOT%\Platform\Odo\Bootload directory. The image NK.Bin can be downloaded via a serial, parallel or Ethernet port from the %_FLATRELEASEDIR% directory. See the on-line manual for more information.	The bootloader is not ported for SA-1100 development module and hence this method of downloading the image is not supported for SA-1100 development module.
Load the image for the Windows CE operating system over the serial port ^a	<ol style="list-style-type: none"> 1. For the SA-1100 development module, verify that Angel 1.02 or later is installed. For information on upgrading Angel, refer to the <i>SA-1100 Microprocessor Firmware User's Guide</i>. 2. Build two 4MB images of your Windows CE binary (i.e., NK.NB0 and NK.NB1). 3. Run the ARM debugger on your host machine and establish a connection with the Angel Debug Monitor. 4. Run the get file command (FILE menu) and get nk.nb0 at address 0xD000 0000 and then nk.nb1 at 0xD800 0000. 5. After both the images are loaded, set the PSR (status register from the VIEW-REGISTERS-CURRENT Option) to 0xD3. This sets the SA-1100 in supervisor mode. 6. Set the PC (program counter from the VIEW-REGISTERS-CURRENT option) to 0xD000 1000. 7. Enter GO (or F5) to execute the image. This results in an error message in the ARM debugger on your host machine. The error message can be ignored because the Windows CE operating system overwrites the Angel debug monitor making it useless. 	Takes too long (approximately 15 minutes) to download each image. Also, the serial port is rendered useless and debug messages from the image for the Windows CE operating system are not visible.
Load the image for the Windows CE operating system from the PCMCIA Intel compatible linear flash card ^a	<ol style="list-style-type: none"> 1. For the SA-1100 development module, verify that Angel 1.02 or later is installed. For information on upgrading Angel, refer to the <i>SA-1100 Microprocessor Firmware User's Guide</i>. 2. Transfer the image for the Windows CE operating system to the PCMCIA flash card by using the File to Flash utility. This utility is provided by Intel and is also available on the Intel Flash web site. 3. Run a program to copy the image from the flash to the DRAM (first 4 MB at 0xD000 0000 and second 4 MB at 0xD800 0000 for the SA-1100 development module), set the SA-1100 in the supervisor mode and jump to the location of the image of the Windows CE operating system (0xD000 1000 for SA-1100 development module). The 0x1000 offset is required because the actual executable code starts from that offset. This program is available from Intel for the SA-1100 development module. 	—

a. MAKEIMG will make a number of NK.NB images, depending on the size of the image and size of the ROM (ROSIZE parameter set in the config.bib file. In the case of the SA-1100 development module, there are two 4 MB images called NK.NB0 and NK.NB1 or one 8 MB image called NK.NB0.

3.1 Multiple Boot Selections

The program that loads the image of the Windows CE operating system from the PCMCIA flash card to the on-board RAM can be loaded in the extra flash on the SA-1100 development module using the following instructions. This allows multiple boot options—with one option to run the Angel and the other option to load the image of the Windows CE operating system and execute it.

Note: All commands, directory names, and file names are case insensitive in the Windows and Windows CE operating system.

1. Build a stand-alone image (for example, loadce.axf) of the program to load the image for the Windows CE operating system that will be burnt in the flash using the FMU and then selected for running.
2. Establish a connection with the Angel Debug Monitor using the ARM debugger from your host machine.
3. Download the Flash Management Utility program (FMU) and run it. Refer to the *SA-1100 Microprocessor Firmware User's Guide* for details.
4. Program loadce.axf into flash as image 2 in flash block 4, as shown in the following example:

```
FMU> program 2 Loads <YOUR_PLATFORM>/loadce.axf 4
```

5. Refer to the *SA-1100 Microprocessor Firmware User's Guide* for details.
6. Select image 2 to run at boot time by setting GPIO 1 to 1 (S21 no dot) and GPIO 0 to 0 (S22 dot).
7. Quit FMU and the Angel Debug Monitor.
8. Reset the SA-1100 Microprocessor Evaluation Platform. You may have to reset the platform a couple of times. If this utility is running correctly, LED D17 will light up.
9. The LoadCE utility copies the 8 MB image for the Windows CE operating system from the flash card inserted in the upper PCMCIA slot into RAM and starts executing it. Once the image starts executing, it will dump debug messages until the kernel comes up on the serial port J23. You can see these messages using the WinDbg (ARM Debugger) program included with the Windows CE 2.1 ETK.

If the image is built for the MINSHELL project and loaded correctly, upon power up the touch screen calibration screen comes up on the display. After completing the calibration, the screen goes blank. Using the ALT-TAB keys brings up the task manager to allow applications to be executed that have been included with the image. Examples of some of the applications that can be executed from the task manager are CMD, which opens a command window similar to DOS,* GENERIC and SPINTEST.

Support, Products, and Documentation

If you need technical support, a *Product Catalog*, or help deciding which documentation best meets your needs, visit the Intel World Wide Web Internet site:

<http://www.intel.com>

Copies of documents that have an ordering number and are referenced in this document, or other Intel literature may be obtained by calling **1-800-332-2717** or by visiting Intel's website for developers at:

<http://developer.intel.com>

You can also contact the Intel Massachusetts Information Line or the Intel Massachusetts Customer Technology Center. Please use the following information lines for support:

For documentation and general information:	
Intel Massachusetts Information Line	
United States:	1-800-332-2717
Outside United States:	1-303-675-2148
Electronic mail address:	techdoc@intel.com

For technical support:	
Intel Massachusetts Customer Technology Center	
Phone (U.S. and international):	1-978-568-7474
Fax:	1-978-568-6698
Electronic mail address:	techsup@intel.com

