BSP Porting Guide for SMDK6410
(Windows Embedded CE 6.0)

S3C6410

March 31, 2009

(Preliminary) REV 0.3

Important Notice

The information in this publication has been carefully checked and is believed to be entirely accurate at the time of publication. Samsung assumes no responsibility, however, for possible errors or omissions, or for any consequences resulting from the use of the information contained herein.

Samsung reserves the right to make changes in its products or product specifications with the intent to improve function or design at any time and without notice and is not required to update this documentation to reflect such changes.

This publication does not convey to a purchaser of semiconductor devices described herein any license under the patent rights of Samsung or others.

Samsung makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does Samsung assume any liability arising out of the application or use of any product or circuit and specifically disclaims any and all liability, including without limitation any consequential or incidental damages.

"Typical" parameters can and do vary in different applications. All operating parameters, including "Typicals" must be validated for each customer application by the customer's technical experts.

Samsung products are not designed, intended, or authorized for use as components in systems intended for surgical implant into the body, for other applications intended to support or sustain life, or for any other application in which the failure of the Samsung product could create a situation where personal injury or death may occur.

Should the Buyer purchase or use a Samsung product for any such unintended or unauthorized application, the Buyer shall indemnify and hold Samsung and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, expenses, and reasonable attorney fees arising out of, either directly or indirectly, any claim of personal injury or death that may be associated with such unintended or unauthorized use, even if such claim alleges that Samsung was negligent regarding the design or manufacture of said product

S3C6410 RISC Microprocessor
BSP Porting Guide

Samsung Electronics Co., Ltd.
San #24 Nongseo-Dong, Giheung-Gu
Yongin-City Gyeonggi-Do, Korea
446-711

Home Page: http://www.samsungsemi.com/

E-Mail: mobilesol.cs@samsung.com

Printed in the Republic of Korea

Preliminary product information describe products that are in development, for which full characterization data and associated errata are not yet available. Specifications and information herein are subject to change without notice.

Revision History

Revision No	Description of Change	Refer to	Author(s)	Date
0.1	Preliminary draft	-	Hyuk Lee	2008-09-16
0.2	Revised for RC2 Release		WINCE Team	2009-01-19
0.3	Hive registry and Multiple XIP were added.		Jungchul Park	2009-03-31

NOTE: REVISED PARTS ARE WRITTEN IN BLUE.

Contents

Figures

1 Boot Media Configuration

It is available to boot through both NAND Flash and AMD Flash (NOR Flash) in the SMDK6410 Board.

This information is applied to SMDK6410 CPU Board revision(0.0, 0.1) and Base Board revision(0.0, 0.1, 0.2)

1.1 NOR Flash Boot

In CPU Board

Description	**CFG3[6:1]**				
	[6]	[5]	[4]	[3]	[2]
NOR Boot (8bit Data Width)	X	OFF	**ON**	OFF	OFF
NOR Boot (16bit Data Width)	X	OFF	**ON**	OFF	**ON**

In Base Board

Description	**CFGB1[4:1]**			
	[4]	[3]	[2]	[1]
Connected NorFlash to Xm0CSn0	X	OFF	OFF	**ON**
Connected SRAM to Xm0CSn0	X	OFF	**ON**	OFF

1.2 NAND Flash Boot

In CPU Board

Description	**CFG3[6:1]**				
	[6]	[5]	[4]	[3]	[2]
Normal NAND, 512-byte, 3 addr. Cycle	**ON**	OFF	OFF	OFF	OFF
Normal NAND, 512-byte, 4 addr. Cycle	**ON**	OFF	OFF	OFF	**ON**
Advanced NAND, 2K-byte, 4 addr. Cycle	**ON**	OFF	OFF	**ON**	OFF
Advanced NAND, 2K-byte, 5 addr. Cycle	**ON**	OFF	OFF	**ON**	**ON**

In Base Board

Description	**CFGB3[4:1]**			
	[4]	[3]	[2]	[1]
Connected NandFlash to Xm0CSn2	OFF	OFF	OFF	**ON**
Connected XD Picture Card to Xm0CSn2	OFF	OFF	**ON**	OFF

1.3 Internal ROM - NAND Flash Boot

In CPU Board

Description	**CFG3[6:1]**				
	[6]	[5]	[4]	[3]	[2]
Internal ROM	X	**ON**	**ON**	**ON**	**ON**

In Base Board

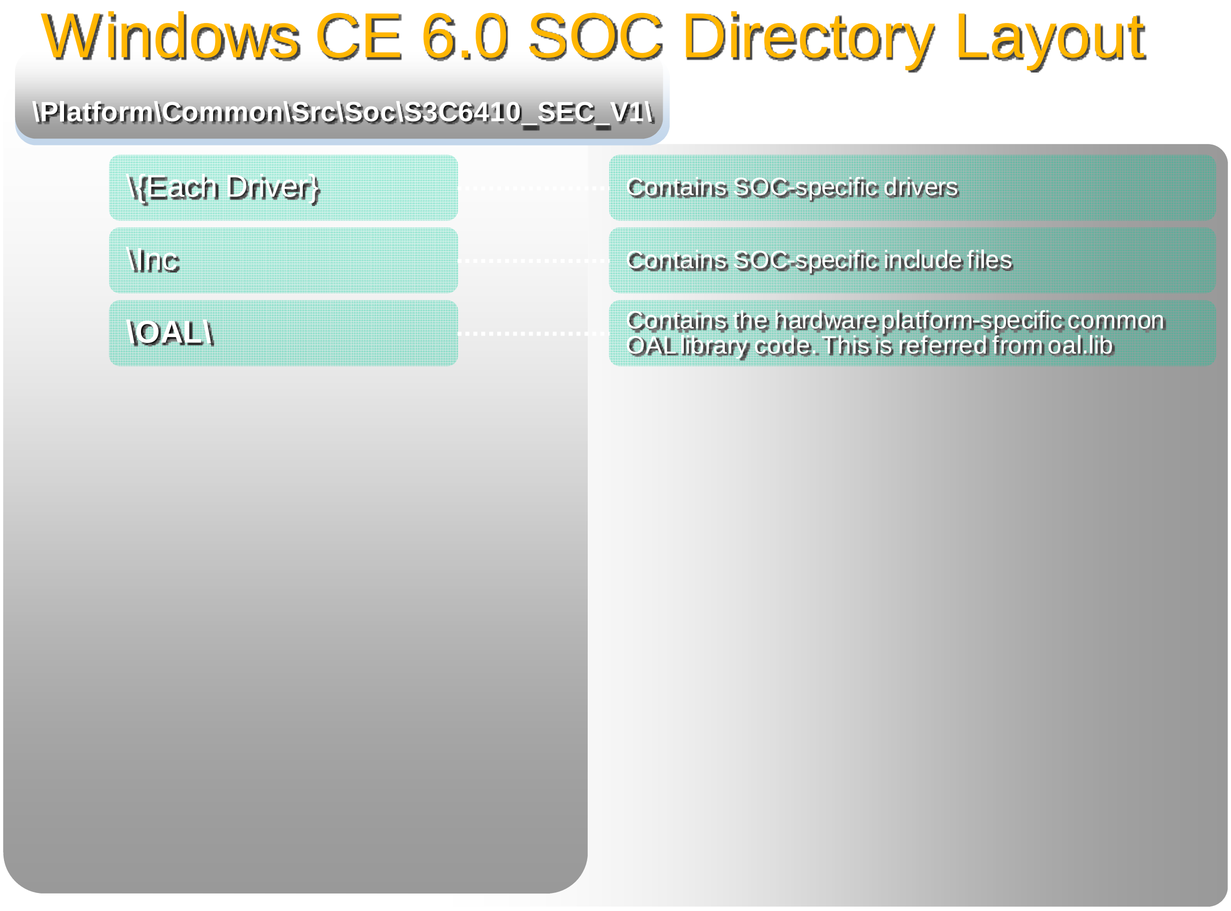
Description	**CFGB3[4:1]**			
	[4]	[3]	[2]	[1]
Connected NandFlash to Xm0CSn2	OFF	OFF	OFF	**ON**
Connected XD Picture Card to Xm0CSn2	OFF	OFF	**ON**	OFF

To make an image that can boot up using Internal ROM, you should do the followings:

```
In smdk6410\smdk6410.bat file
    set BSP_IROMBOOT=1
```

2 BSP Directory Layout

Windows CE 6.0 BSP Directory Layout is as follows. This is changed from Beta Release BSP. The main change is OAL common parts. Please Refer to “Appendix I. Directory Layout Change list”. This Change list will be updated up to Final Release. If you have used old BSP, Appendix will help you to understand what is changed. and Release Notes also help you.



Windows CE 6.0 SOC Directory Layout

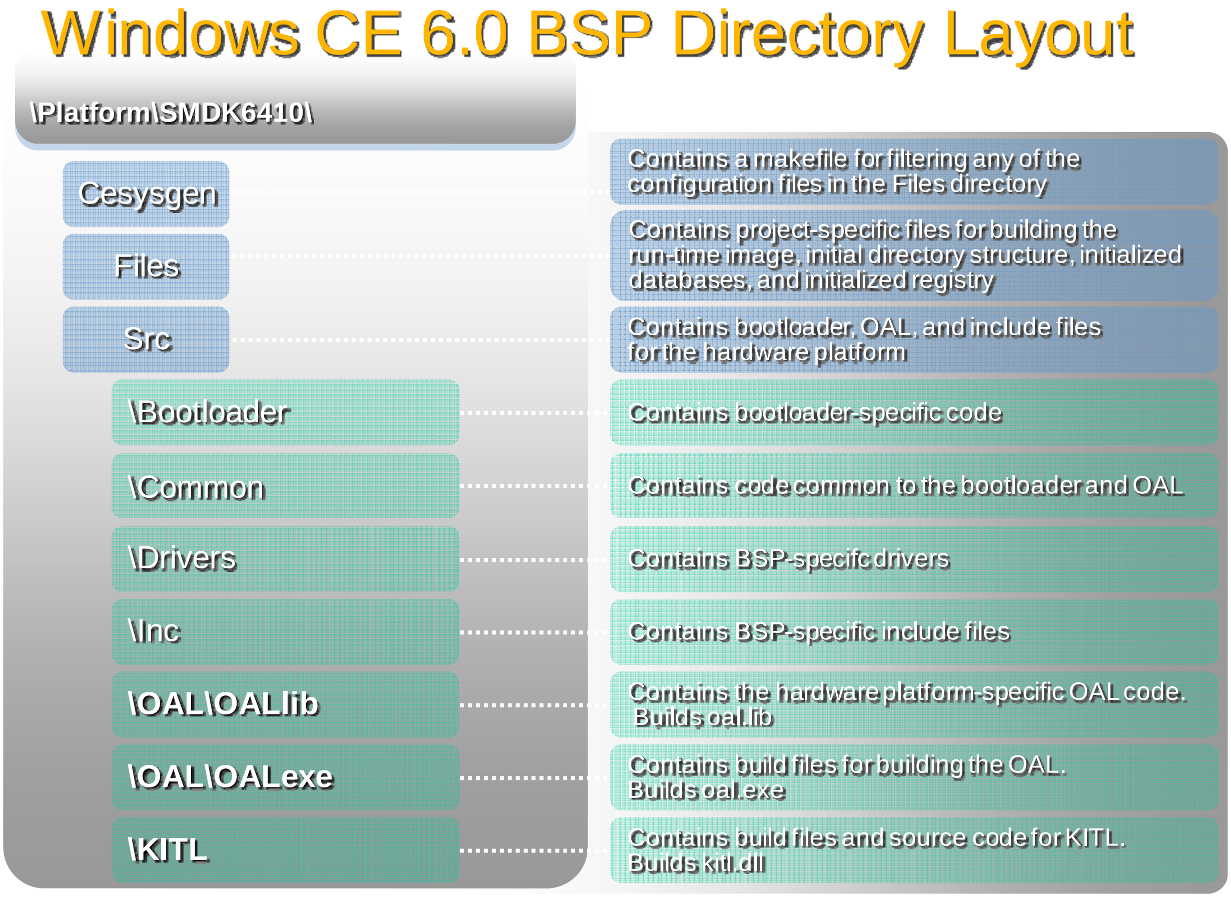
\Platform\Common\Src\Soc\S3C6410_SEC_V1\

\{Each Driver} — Contains SOC-specific drivers

\Inc — Contains SOC-specific include files

\OAL\ — Contains the hardware platform-specific common OAL library code. This is referred from oal.lib

Figure 2-1 WinCE 6.0 SOC Directory Layout



Windows CE 6.0 BSP Directory Layout

\Platform\SMDK6410\

Cesysgen	Contains a makefile for filtering any of the configuration files in the Files directory
Files	Contains project-specific files for building the run-time image, initial directory structure, initialized databases, and initialized registry
Src	Contains bootloader, OAL, and include files for the hardware platform
\Bootloader	Contains bootloader-specific code
\Common	Contains code common to the bootloader and OAL
\Drivers	Contains BSP-specifc drivers
\Inc	Contains BSP-specific include files
\OAL\OALlib	Contains the hardware platform-specific OAL code. Builds oal.lib
\OAL\OALexe	Contains build files for building the OAL. Builds oal.exe
\KITL	Contains build files and source code for KITL. Builds kitl.dll

Figure 2-2 WinCE 6.0 BSP Directory Layout

3 OAL option configuration

3.1 CPU Selection

This is obsolete. This BSP only support SMDK6410.

3.2 System Clock Configuration

- You can change System Clock Speed (including CPU Clock)
- There are three pre-defined values for control system clock speed in two files (PLATFORM\COMMON\SRC\SOC\S3C6410_SEC_V1\OAL\INC\soc_**cfg.h** and PLATFORM\COMMON\SRC\SOC\S3C6410_SEC_V1\OAL\INC\s3c6410.inc)
- Default setting is 532Mhz/133Mhz/33.25Mhz. (ARM/HCLK/PCLK)

In \PLATFORM\COMMON\SRC\SOC\S3C6410_SEC_V1\OAL\INC\soc_cfg.h file

```
// Change This Definition to change SOC Clock !!! (and "s3c6410.inc")
//#define S3C6410_FCLK        FCLK_400MHz
#define S3C6410_FCLK          FCLK_532MHz
//#define S3C6410_FCLK        FCLK_634MHz
```

In PLATFORM\COMMON\SRC\SOC\S3C6410_SEC_V1\OAL\INC\s3c6410.inc file

```
;-------------------------------------------------
; Change S3C6410_FCLK definition for StartUp code
;-------------------------------------------------
;S3C6410_FCLK  SETA    FCLK_400MHZ
S3C6410_FCLK   SETA    FCLK_532MHZ
;S3C6410_FCLK  SETA    FCLK_634MHZ
FIN            EQU     12000000
;-------------------------------------------------;
```

- SDRAM parameter and System Timer parameter will be recalculated properly as clock speed
- If you want use any other value than pre-defined clock speed, you should define several definitions related to clock speed configuration as sample code in **soc_cfg.h** and **s3c6410.inc** file
  - Operation with 666MHz ARM CLK needs 1.3V as VDD_INT and 800MHz ARM CLK needs 1.3V as VDD_ARM.
  - **Clock setting process on booting is as follows:**
    - 532MHz/133MHz/66MHz (ARM/HCLK/PCLK) in the **NBL1** and this is fixed.

- Setting the Proper voltage to fit target ARM clock in the **NBL2**.
- Setting the **real target clock** In the **OAL**.

-

3.3 LCD Display Module Configuration

- You can change LCD module type for SMDK6410 board

- There are four pre-defined LCD module configuration in BSP

  LTS222 : Portrait 2.2” QVGA

  LTV350 : Landscape 3.5” QVGA (SMDK6410 Base Board Rev0.0)

  LTE480 : Landscape 4.8” WVGA (SMDK6410 Base Board Rev0.1)

  EMUL48_D1 : Landscape 4.8” WVGA works as D1 (720x480)

  EMUL48_QV : Landscape 4.8” WVGA works as QVGA (320x240)

  EMUL48_PQV : Landscape 4.8” WVGA works as PQVGA (240x320) Rev0.1)

  LTP700 : Landscape 7” WVGA

In smdk6410\SRC\Inc\bsp_cfg.h

```
//---------------------------------------------------------------------------
// SMDK6410 Display Dimension
//---------------------------------------------------------------------------
#define LCD_MODULE_LTS222          (0)     // Portrait 2.2" QVGA RGB16
#define LCD_MODULE_LTV350          (1)     // Landscape 3.5" QVGA RGB16
#define LCD_MODULE_LTE480          (2)     // Landscape 4.8" WVGA RGB16
#define LCD_MODULE_EMUL48_D1       (3)     // Landscape 4.8" WVGA as D1 (720x480)
#define LCD_MODULE_EMUL48_QV       (4)     // Landscape 4.8" WVGA as QVGA (320x240)
#define LCD_MODULE_EMUL48_PQV      (5)     // Landscape 4.8" WVGA as PQVGA (240x320)
#define LCD_MODULE_LTP700          (6)     // Landscape 7" WVGA RGB24
#define SMDK6410_LCD_MODULE        (LCD_MODULE_LTE480)
```

- If you want use any other module than pre-defined in BSP, you should define several definitions related to LCD module dimensions as sample code in **bsp_cfg.h,** and implement **LDI_fill_output_device_information()** function and module control functions (LDI_XXX()) in **\smdk6410\SRC\drivers\Display\s3c6410_disp_ldi.c** file

3.4 UART Debug Port Configuration

- There are two UART port available for debug. You can use one port at a time.
- If you set BSP_DEBUGPORT as following code, Debug port will use UART0
- Default setting is UART0

```
In smdk6410\smdk6410.bat file
    set BSP_DEBUGPORT=SERIAL_UART0
    @REM set BSP_DEBUGPORT=SERIAL_UART1
    @REM set BSP_DEBUGPORT=SERIAL_UART2
    @REM set BSP_DEBUGPORT=SERIAL_UART3
```

- You can change Baudrate for debug port also
- Default setting is 115200 bps

```
In smdk6410\SRC\Inc\bsp_cfg.h
    //------------------------------------------------------------------------------
    // SMDK6410 UART Debug Port Baudrate
    //------------------------------------------------------------------------------
    #define DEBUG_UART0          (0)
    #define DEBUG_UART1          (1)
    #define DEBUG_UART2          (2)
    #define DEBUG_UART3          (3)
    #define DEBUG_BAUDRATE       (115200)
```

- **You should configure the baudrate of terminal program in your host PC same as SMDK6410 board**

3.5 NAND Flash

- For using NAND Flash, timing parameter should be set appropriately
- The default setting leaves a margin. (TACLS: 7, TWRPH0: 7, TWRPH1: 7)
- **Please set optimal timing parameter for NAND Flash your platform uses**

In smdk6410\SRC\Inc\bsp_cfg.h file

```
//------------------------------------------------------------------------------
// SMDK6410 NAND Flash Timing Parameter
//------------------------------------------------------------------------------
#if      (S3C6410_HCLK == FCLK_100MHz)
#define NAND_TACLS          (7)
#define NAND_TWRPH0         (7)
#define NAND_TWRPH1         (7)
#elif    (S3C6410_HCLK == FCLK_133MHz)
#define NAND_TACLS          (7)
#define NAND_TWRPH0         (7)
#define NAND_TWRPH1         (7)
#endif
```

3.6 Dynamic Voltage and Frequency Scaling (DVS & DFS)

- The SMDK6410 BSP supports DFS and DVS feature for improving power consumption.
- Kernel changes system frequency (ARMCLK and HCLK) and voltage (VDD_ARM and VDD_INT) dynamically as idle rate changes.

- **The default setting is disabled.**
- If you want to use DFS and DVS, Change as follow.

```
In smdk6410\smdk6410.bat file
    set BSP_USEDVS=1
```

- Setting **BSP_USEDVS=1** means that DFS and DVS is enabled, but DVS is basically disabled.
- If you want to **use DFS with DVS**, uncomment definition for voltage change as follow.

```
In smdk6410\Src\Common\PM\DVS.h file
    #define ENABLE_VOLTAGE_CONTROL
```

- **Note that DFS and DVS are disabled temporarily while USB cable attached. After detaching cable, DFS and DVS are enabled automatically.**
- **Note that DFS and DVS have been tested 800, 666, 532 MHz on Target ARM CLK.**
- **For other clocks, you need to modify the DVFS transition table on smdk6410\Src\Common\PM\DVS.h for each clock.**
- L4 and L5 are additional level which is aimed for decreasing power. **The parameter regarding DRAM access on DMC(Memory Controller) should be changed at L4 and L5 mode because HCLK is changed**. All of DRAM accesses should be stopped and DMC is in sleep state when that parameter is changed. Keep consider the side-effect in aspect of system configuration when L4 and L5 are used. **Transition from L3 upto L0 is recommended for this reason.**

4 Display Driver Configuration

- Display device can be disabled. **(set BSP_NODISPLAY=1)**
- If you clear BSP_NODISPLAY as following code, Display driver will be included in OS image
- Default setting is enabled.

```
In smdk6410\smdk6410.bat file
    set BSP_NODISPLAY=
```

- Setting **BSP_NODISPLAY=1** means that display driver is removed from OS image.

- If you want to enable or disable 2D Hardware accelerator, you may modify this.

```
In smdk6410\src\drivers\display\s3c6410_disp_drv\precomp.h file
#define G2D_ACCELERATE(TRUE)                    //< If you want to use 2D HW for GDI, set
this to "TRUE", if not, set to "FALSE"
#define G2D_TRY_CBLT    (TRUE)          //< Try to bitblt from cached source surface to non
cached destinatino surface, this do cache flush
#define G2D_MSG  (FALSE)


/// For using Physically Linear Surface on System Memory to wide 2D HW usage.
/// 2D HW need physically contiguous memory, and its address.
/// This will consume System Memory and allocate Physically and Virtually contiguous
memory.
/// So if system has small memory, allocation may fail.
/// Then 2D HW will not work for that memory.
/// BUGBUG: in Media Player, Occasionly PACSurf object cannot be bitblted correctly.
#define USE_PACSURF                 (TRUE)
```

- There are some optimization options provided for 2D Hardware

```
In smdk6410\src\drivers\display\s3c6410_disp_drv\precomp.h file
#define PAC_ALLOCATION_BOUNDARY    (160*120*2)    //(320*240*2)          //<  PACSurf
creation request is processed only for the surface has over QVGA 16bpp size


#define G2D_BLT_OPTIMIZE        (FALSE)                              //< This option will enable
above two optimization method. This can increase 2D processing overhead.
#define G2D_COMPROMISE_LIMIT  (28800)                     //< Transferring below this
size(byte) using HW will be poor than using SW. so we will use software 2D flow under this
size transfer request.
#define    G2D_BYPASS_HW_STRETCHBLT        (FALSE)          //< HW Stretchblt algorithm
differs from MS'SW Stretching BLT algorithms,
                                             //< So, CETK 218, 219 can fails.
```

5 Touch Screen Driver Configuration

- Touch screen device can be disabled. **(set BSP_NOTOUCH=1)**
- If you clear BSP_NOTOUCH as following code, Touch screen driver will be included in OS image
- Default setting is enabled.

```
In smdk6410\smdk6410.bat file
    set BSP_NOTOUCH=
```

- Setting **BSP_NOTOUCH=1** means that touch screen driver is removed from OS image.
- After Changing, Build display driver and make image.

- Default calibration data is defined in registry. Set proper value for the touch panel you have.

```
In smdk6410\files\platform.reg file
;------------ Touch Driver ------------------------------------------------
; @CESYSGEN IF CE_MODULES_POINTER
IF BSP_NOTOUCH !
[HKEY_LOCAL_MACHINE\HARDWARE\DEVICEMAP\TOUCH]
    "DriverName"="s3c6410_touch.dll"
    "MaxCalError"=dword:7
[HKEY_LOCAL_MACHINE\HARDWARE\DEVICEMAP\TOUCH]
    "DriverName"="s3c6410_touch.dll"
    "MaxCalError"=dword:7
; 3.5" QVGA Landscape
;   "CalibrationData"="504,508 233,236 231,788 785,792 790,227"
; 4.8" WVGA Landscape
    ; for S3C6410 SMRP 12bit
    "CalibrationData"="2108,1981 1131,2643 1133,1343 3079,1339 3078,2616"
    ; for S3C6410 12bit
;   "CalibrationData"="2098,1998 1013,2663 997,1339 3187,1312 3195,2665"
    ; for S3C6410 10bit
;   "CalibrationData"="519,501 253,671 246,335 794,331 794,663"
    ; for S3C6400
;   "CalibrationData"="508,490 200,735 204,246 820,240 823,731"
; 4.8" WVGA Landscape as D1 (720x480)
;   "CalibrationData"="473,492 189,747 188,244 751,238 748,748"
; 4.8" WVGA Landscape as QVGA Landscape
;   "CalibrationData"="278,642 158,766 159,517 404,520 405,757"
; 4.8" WVGA Landscape as QVGA Portrait
;   "CalibrationData"="244,586 152,754 152,426 332,425 334,747"
; 4.8" WVGA Landscape as 480x320
;   "CalibrationData"="362,592 171,761 173,431 547,427 543,755"
; 4.8" WVGA Landscape as 320x480
;   "CalibrationData"="276,489 156,740 156,240 402,238 396,732"ENDIF BSP_NOTOUCH !
; @CESYSGEN ENDIF CE_MODULES_POINTER
;-------------------------------------------------------------------------
```

6 Keypad Driver Configuration

- Keypad/Keyboard device can be disabled. **(set BSP_NOKEYBD=1)**
- If you clear BSP_NOKEYBD as following code, Keypad/Keyboard driver will be included in OS image
- Default setting is enabled.

```
In smdk6410\smdk6410.bat file
      set BSP_ NOKEYBD=
```

- Setting **BSP_NOKEYBD=1** means that keypad/keyboard driver is removed from OS image.
- After Changing, Build keypad/keyboard driver and make image.
- If you want to use Keypad on SMDK6410 Board, set CFG to the following **CFG4 (all OFF)**
- To use the Keypad, Set the CFG4 in the base board as below.

	1	**2**	**3**	**4**
CFG4 :	**OFF**	**OFF**	**OFF**	**OFF**

7 Power Button Driver Configuration

- You can enter sleep mode and trigger S/W reset with power button driver
- Power Button is switch 68 (SW68 in bottom of base board)
- Reset Button is switch 66 (SW66 in bottom of base board)
- Power Button driver can be disabled. **(set BSP_NOPWRBTN=1)**
- If you clear BSP_NOPWRBTN as following code, Power Button driver will be included in OS image
- Default setting is enabled.

```
In smdk6410\smdk6410.bat file
    set BSP_NOPWRBTN=
```

- Setting **BSP_NOPWRBTN=1** means that Power Button driver is removed from OS image.
- After Changing, Build Power driver and make image

8 Audio Driver Configuration

SMDK6410 supports two audio interfaces AC97 and IIS.

- Audio driver can be disabled **(set BSP_NOAUDIO=1)**
- Default setting is enabled and AC97 interface

8.1 AC97 interface

- For including AC97 driver in OS image , change as follow

```
In smdk6410\smdk6410.bat file
    set BSP_NOAUDIO=
    set BSP_AUDIO_AC97=1
```

- Rebuild audio driver and make image
- To use the AC97 interface as audio device Set the CFG1 and CFG2 in the base board as below.

	1	2	3	4
CFG1 :	**ON**	**ON**	**ON**	**ON**
CFG2 :	**OFF**	**OFF**	**OFF**	**OFF**

- If you want to change the Key layout, set like the following.

```
In smdk6410\SRC\Inc\bsp_cfg.h file
    //---------------------------------------------------------------------------
    // SMDK6410 Keypad Layout
    //---------------------------------------------------------------------------
    #define LAYOUT0                              (0)            // 8*8 Keypad board
    #define LAYOUT1                              (1)            // On-Board Key
    #define LAYOUT2                              (2)            // Qwerty Key board
    #define MATRIX_LAYOUT                (LAYOUT1)
```

8.2 IIS interface

- For including IIS driver in OS image , change as follow

```
In smdk6410\smdk6410.bat file
    set BSP_NOAUDIO=
    set BSP_AUDIO_AC97=
    set BSP_NOI2C=
```

- Rebuild audio driver and make image
- **Because IIS driver use I2C interface to control external audio codec chip. You have to include I2C driver in the OS image**

- To use the IIS interface as audio device Set the CFG1 and CFG2 in the base board as below.

	1	2	3	4
CFG1 :	**OFF**	**OFF**	**OFF**	**OFF**

8.3 Board Revision

- In SMDK6410 Base Board Rev 0.0, AC97 does not work properly, so, you should remove the register, R48. IIS 5.1 channel path is able to use. But, IIS 2 channel does not work properly.
- In the SMDK6410 Base Board Rev 0.1, AC97, IIS 5.1 channel path and IIS 2 channel path work properly. If the resister R48 is connected, you should remove the R48.
- In AC97, delay value is configurable. Sometimes you need to adjust the delay value.

9 Camera Driver Configuration

- Camera driver can be disabled. **(set BSP_NOCAMERA=1)**
- Default setting is disabled.
- If you clear BSP_NOCAMERA as following code, Camera driver will be included in OS image

```
In smdk6410\smdk6410.bat file
    set BSP_NOCAMERA=
    set BSP_NOI2C=
```

- Setting **BSP_NOCAMERAI=1** means that Camera driver is removed from OS image.
- After Changing, Build Camera driver and make image
- **Because Camera driver use I2C interface to control external camera module. You have to include I2C driver in OS image**

10 I2C Driver Configuration

- I2C driver can be disabled. **(set BSP_NOI2C=1)**
- Default setting is disabled.
- If you clear BSP_NOI2C as following code, I2C driver will be included in OS image

```
In smdk6410\smdk6410.bat file
    set BSP_NOI2C=
```

- Setting **BSP_NOI2C=1** means that I2C driver is removed from OS image.
- After Changing, Build I2C driver and make image
- **IIS Audio Driver and Camera Driver use I2C driver to control external device. You should include I2C driver for that kind of drivers**

11 SPI Driver Configuration

- SPI driver can be disabled. **(set BSP_NOSPI=1)**
- Default setting is disabled.
- If you clear BSP_NOSPI as following code, SPI driver will be included in OS image

```
In smdk6410\smdk6410.bat file
      set BSP_NOSPI=
```

- Setting **BSP_NOSPI=1** means that SPI driver is removed from OS image.
- After Changing, Build SPI driver and make image

12 USB Driver Configuration

12.1 USB Device

- USB device can be disabled. **(set BSP_NOUSBFN=1)**
- Default setting is enabled and Serial function.
- If you clear BSP_NOUSBFN as following code, USB device driver will be included in OS image

```
In smdk6410\smdk6410.bat file
    set BSP_NOUSBFN=
```

- Setting **BSP_NOUSBFN=1** means that USB device driver is removed from OS image.
- After Changing, Build display driver and make image.
- **You can not use USB function device and USB KITL at a time.**

12.1.1 Serial Function Driver

- You can use USB Serial Function driver by setting as follows.
- Default setting is serial function driver

```
In smdk6410\smdk6410.bat file
    set BSP_NOUSBFN=
    set BSP_USBFNCLASS=SERIAL
    @REM set BSP_USBFNCLASS=MASS_STORAGE
```

- After changing, Build USB function driver and make image.

12.1.2 Mass Storage Function Driver

- You can use USB Mass Storage Function driver by setting as follows.

```
In smdk6410\smdk6410.bat file
    set BSP_NOUSBFN=
   @REM set BSP_USBFNCLASS=SERIAL
   set BSP_USBFNCLASS=MASS_STORAGE
```

- After changing, Build USB function driver and make image.

12.2 Host

- USB device can be disabled. **(set BSP_NOUSBHCD=1)**
- Default setting is enabled.
- If you clear BSP_NOUSBHCD as following code, USB Host driver will be included in OS image

```
In smdk6410\smdk6410.bat file
    set BSP_NOUSBHCD=
```

- Setting **BSP_NOUSBHCD=1** means that USB Host driver is removed from OS image.
- After Changing, Build USB Host driver and make image.

13 Serial Driver Configuration

- All serial drivers can be disabled. **(set BSP_NOSERIAL=1)**
- Default setting is disabled.
- So if you want to enable serial driver, clear BSP_NOSERIAL

In smdk6410\smdk6410.bat file

```
    set BSP_NOSERIAL=
```

- Setting **BSP_ NOSERIAL =1** means that Serial driver is removed from OS image.

13.1 UART

- UART0, UART1, UART2 and UART3 can be disabled by each setting in BSP.
- Default BSP setting is disabled.
- The following codes means enable UART1

In smdk6410\smdk6410.bat file

```
    set BSP_NOUART0=1
    set BSP_NOUART1=
    set BSP_NOUART2=1
    set BSP_NOUART3=1
```

- After Changing, Build driver and make image.

- The following codes means enable UART0

In smdk6410\smdk6410.bat file

```
    set BSP_NOUART0=
    set BSP_NOUART1=1
    set BSP_NOUART2=1
    set BSP_NOUART3=1
```

- **Be careful when using UART0 as general purpose COM port. UART0 is default debug port. You must change debug port to UART1.**
- You can change debug port to UART1 like the following. Then you can use UART0 as COM port without conflict

In smdk6410\smdk6410.bat file

```
    set BSP_NOUART0=
        set BSP_DEBUGPORT=SERIAL_UART1
```

- After Changing, Build driver and make image.

- In SMDK6410 board, UART1, UART2 and UART3 share same COM2 port.
- If you want to use UART1 on COM2 port, set CFG3 in the base board to the following.

	1	2	3	4
CFG3 :	**OFF**	**DC**	**DC**	**DC**

***DC means don't care**

- If you want to use UART2 on COM2 port, set CFG3 in the base board to the following.

	1	2	3	4
CFG3 :	**ON**	**OFF**	**OFF**	**OFF**

- If you want to use UART3 on COM2 port, set CFG3 in the base board to the following.

	1	2	3	4
CFG3 :	**ON**	**ON**	**ON**	**OFF**

13.2 IrDA

- IrDA uses UART2, UART3
- Default BSP setting is disabled.

- If you want to use IrDA on UART channel2, set like the following.
- **Be careful When using UART Channel 2 as IrDA, you cannot use UART channel 2 as UART. So, Do not Enable IrDA2 and UART2 at the same time.**
- If you clear BSP_NOIRDA2 like following code, you can use IrDA device.

```
    In smdk6410\smdk6410.bat file
        set BSP_NOIRDA2
```

- After Changing, Build driver and make image.
- And you need SMDK base board setting.
- Set CFG3 to the following for IrDA2 test.

	1	2	3	4
CFG3 :	DC	DC	ON	OFF

***DC means don't care**

- If you want to use IrDA on UART channel3, set like the following.
- **Be careful When using UART Channel 3 as IrDA, you cannot use UART channel 3 as UART. So, Do not Enable IrDA3 and UART3 at the same time.**
- If you clear BSP_NOIRDA3 like following code, you can use IrDA device.

```
In smdk6410\smdk6410.bat file
    set BSP_NOIRDA3
```

- After Changing, Build driver and make image.
- And you need SMDK base board setting.
- Set CFG3 to the following for IrDA3 test.

	1	2	3	4
CFG3 :	DC	DC	OFF	OFF

***DC means don't care**

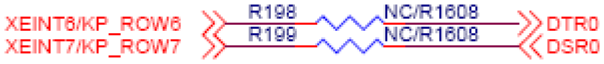
13.3 Active Sync

- If you want to use Active Sync with UART Serial, You should use UART0 in SMDK6410
- The following codes means enable UART0
- You cannot use same UART and serial KITL at once.

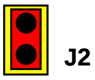
```
In smdk6410\smdk6410.bat file
    set BSP_NOUART0=
    set BSP_NOUART1=1
    set BSP_NOUART2=1
    set BSP_NOUART3=1
```

- **Be careful when using UART0 as general purpose COM port. UART0 is default debug port. You must change debug port to UART1.**

- In SMDK6410 Base Board, DTR and DSR is not connected as follow Figure. So, you should connect R198 and R199.



- Short jumper J2 on SMDK base board.



- And configure WinCE and ActiveSync to use COM port.

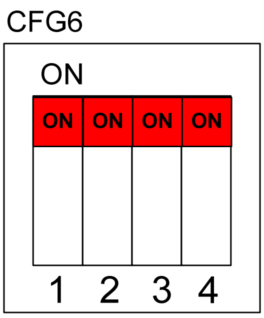
14 SD / HSMMC Driver Configuration

14.1 Channel Configuration

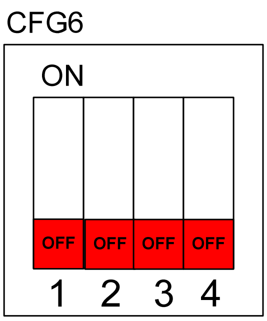
- The driver supporting 4 bit DAT bus width on SD/HSMMC Channel 0 can be disable, Default setting is disabled.
  - ■ **set BSP_NOHSMMC_CH0=1**
- The drivers supporting 4 bit DAT bus width on SD/HSMMC Channel 1 can be disabled, Default setting is enabled.
  - ■ **set BSP_NOHSMMC_CH1=1**
- Driver Supporting 8 bit DAT bus width on Channel 1 can be enabled, Default setting is disabled.
  - ■ **set BSP_HSMMC_CH1_8BIT=**
- If you clear BSP_NOHSMMC_CHx as following code, SD/HSMMC drivers will be included in OS image. (Channel 0 is 4 bit, Channel 1 is 8 bit.)

```
In smdk6410\smdk6410.bat file
      set BSP_NOHSMMC_CH0=
      set BSP_NOHSMMC_CH1=
      set BSP_HSMMC_CH1_8BIT=1
```

- According to setting the DAT bus width of Channel 1, CFG6 on CPU board should be set as following, except SMDK REV 0.0:

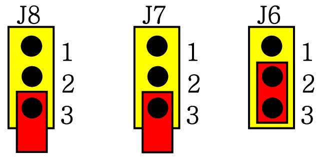


< 8 bit DAT bus >

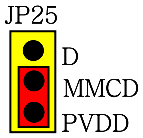


< 4 bit DAT bus >

- **In this time, SD/HSMMC Drivers are under developing. Some kind of HS-MMC does not work with Driver.**
- Open jumper J7 and J8, and Short pin 2 and 3 of jumper J6 on SMDK base board.



- Short pin MMCD and PVDD of jumper JP25 on SMDK CPU board.



14.2 Fast-Path

“*Fast-Path operations improve transfer performance on high speed SDIO and SD cards by utilizing polling in the host controller driver.*” In MSDN. And Fast-Path is enabled basically. But CPU utilization can be increased as result of Fast-Path. So, you can disable Fast-Path with comment out _FASTPATH_ENABLE_ on CDEFINES in sources file. **But, you are responsible for the result on Fast-Path disability**.

```
In SMDK6410\Src\Drivers\HSMMC\SDBus\sources file
    CDEFINES=$(CDEFINES) … …  # -D_FASTPATH_ENABLE_
```

14.3 Using the Channel 0

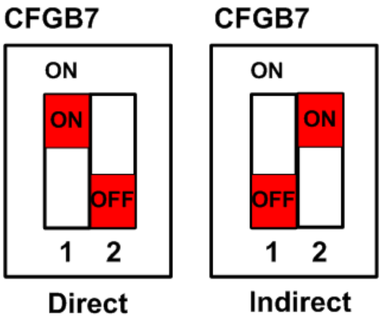
- Board modification is needed for using HSMMC channel 0 to work correctly according to the SMDK board revision number Except for REV 0.0. If you have the later board than REV 0.0, it is needed to modify the CPU Board in the board level.
- Remove R156 and connect R158 with a resistor less than 100K Ohm for using external interrupt as card detection signal.

15 CF Driver Configuration

- **TRUE-IDE mode on S3C6410 CF Controller does not support HOT-PLUG. CF device must be in the slot before booting OS up.**
- CF ATAPI driver can be disabled. **(set BSP_NOCFATAPI=1)**
- Default setting is disabled.
- **Default value in registry is Direct Mode.**
- If you clear BSP_NOCFATAPI as following code, CF ATAPI driver will be included in OS image

```
In smdk6410\smdk6410.bat file
    set BSP_NOCFATAPI=
```

- Setting **BSP_NOCFATAPI=1** means that CF ATAPI driver is removed from OS image.
- After Changing, Build CF ATAPI driver and make image
- To set the operating mode, direct mode or indirect mode, set the CFGB7 as below.

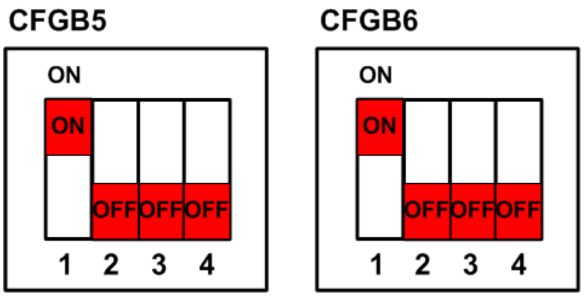


- To use Ultra-DMA mode of ATA-Device, You should set it as Direct mode.

Ultra-DMA mode is only supported on Direct mode

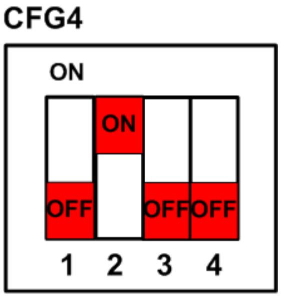
- To use the CF ATAPI, Set the CFGB5 and CFGB6 as below.

-



- **CF ATAPI Interface is conflict with Keypad H/W in SMDK6410 board. Do not Enable CF ATAPI Driver and Keypad Driver at the same time**

- **In some version of SMDK board, you will be able to be faced with some trouble. If encounters a trouble in the direct mode, it is recommended to set the CFG4 in the base board as below.**



-

- If you want to change PIO ,PDMA ,UDMA and In/Direct Mode, change registry as follows,

```
[HKEY_LOCAL_MACHINE\Drivers\BuiltIn\6410_CF\Device0]
"Prefix"="DSK"
"Dll"="S3C6410_CF_atapi.dll"
"Order"=dword:31
"IClass"=multi_sz:"{A4E7EDDA-E575-4252-9D6B-4195D48BB865}",
             "{8DD679CE-8AB4-43c8-A14A-EA4963FAA715}"
"InterruptDriven"=dword:00           ; en(1) dis(0)able interrupt driven I/O
"DMA"=dword:00                       ; disable DMA. DMA is not supported
"DoubleBufferSize"=dword:10000       ; 128 sector (65536 byte) double buffer
"DrqDataBlockSize"=dword:200         ; 1 sector (512 byte) DRQ data block
"WriteCache"=dword:01                ; enable on-disk write cache
"LookAhead"=dword:01                 ; enable on-disk look-ahead
"DeviceId"=dword:00                  ; device 0, i.e., primary master
"TransferMode"=dword:ff              ; use mode 0;
"EnablePDMA"=dword:1                 ; 0 = PIO, 1 = PDMA. We recommend PDMA mode.
"EnableUDMA"=dword:1                 ; 0 = PIO, 1 = UDMA. We recommend UDMA mode.
"IndirectMode"=dword:0               ; 0 = Direct, 1 = Indirect.
                                     ; To use UDMA, It should be '0'
                                     ; UDMA is only working on DIRECT MODE.
```

16 Hive-Based Registry

You can implement that hive files are stored to NandFlash. By default, RAM-based registry is used in the BSP. For implementing hive-based registry, set it as follows.

- Add the Hive-based Registry Catalog item to your OS design.
- And then set configuration like following code

```
In smdk6410\smdk6410.bat file
set IMGHIVEREG=1
```

- After changing, rebuild solution for clean build. Without clean build, Hive-registry properties may be not included in OS image.

17 Multiple XIP

Our BSP support Multiple XIP on NAND flash media by BinFS. The Hive-Based registry is required when using BINFS(demand-paged, pseudo-multi-XIP) because all of the file system waiting/blocking is implemented in the hive-based registry init code, and not in the RAM-Based registry init code.

- Single .bin: Nk.bin file

  The run time image is single image and is loaded to RAM in booting time.
- Multiple .bin(for demand paging): xipkernel.bin, nk.bin and chain.bin

  The XIPKERNEL region is where files that must be loaded prior to BinFs are stored. The NK region is the location for files not stored in the XIPKERNEL region. The CHAIN region is used by the bootloader to access information on where each region is located.
- **By default, Multiple-XIP images are used.**
- Hive-registry must be used for multiple-XIP images.

In smdk6410\smdk6410.bat file

```
set IMGHIVEREG=1
set IMGMULTIXIP=1
```

- The XIPKERNEL region files must include nk.exe, nandflash.dll, kitl.dll, kernel.dll, kd.dll, hd.dll, osaxst0.dll, osaxst1.dll, coredll.dll, oalioctl.dll, k.coredll.dll, fpcrt.dll, k.fpcrt.dll, filesys.dll, romfsd.dll, device.dll, udevice.dll, devmgr.dll, regenum.dll, busenum.dll, pm.dll, servicesEnum.dll, servicesd.exe, services.exe, serviceStar.exe, zlib.dll, softkb.dll, binfs.dll, fsdmgr.dll, mspart.dll, ceddk.dll, wince.nls and boot.hv.
- To make the XIPKERNEL as small as possible, XIPKERNEL should have above files only – as a minimum set of files which are required to boot and initialize the BINFS.
- The MultipleXIP.bib file is included in the config.bib to apply the above list into ce.bib on "makeimg" process. The including code is like as follows:

In SMDK6410\FILES\config.bib file

```
IF IMGMULTIXIP
… …
#include "$(_TARGETPLATROOT)\FILES\MultipleXIP.bib"
ENDIF IMGMULTIXIP
```

- All driver which has Powerdown/PowerUp function must be made in non-pageable for power handler scheme in WinCE6.0. You can find all drivers that need to be non-pageable with below command.

```
flatreleasedir> findstr /m "PowerDown" *.map
```

After searching such as those drivers with above command, you must make them be non-pageable with adding 'M'flag in the .bib file.

```
s3c6410_iic.dll       $(_FLATRELEASEDIR)\s3c6410_iic.dll           NK   SHMK
smdk6410_serial.dll   $(_FLATRELEASEDIR)\smdk410_serial.dll        NK   SHMK
irsir.dll             $(_FLATRELEASEDIR)\irsir.dll                 NK   SHMK
s3c6410_mfc.dll       $(_FLATRELEASEDIR)\s3c6410_mfc.dll           NK   SHMK
s3c6410_jpeg.dll      $(_FLATRELEASEDIR)\s3c6410_jpeg.dll          NK   SHMK
s3c6410_cmm.dll       $(_FLATRELEASEDIR)\s3c6410_cmm.dll           NK   SHMK
s3c6410_uao.dll       $(_FLATRELEASEDIR)\s3c6410_uao.dll           NK   SHMK
s3c6410_PwrBtn.dll    $(_FLATRELEASEDIR)\s3c6410_PwrBtn.dll        NK   SHMK
s3c6410_fimg.dll      $(_FLATRELEASEDIR)\s3c6410_fimg.dll          NK   SHMK
s3c6410_wavedev.dll   $(_FLATRELEASEDIR)\s3c6410_wavedev.dll       NK   SHMK
s3c6410_ohci2.dll     $(_FLATRELEASEDIR)\s3c6410_ohci2.dll         NK   SHMK
s3c6410_usbfn.dll     $(_FLATRELEASEDIR)\s3c6410_usbfn             NK   SHMK
s3c6410_hsmmc0.dll    $(_FLATRELEASEDIR)\s3c6410_hsmmc0.dll        NK   SHMK
s3c6410_hsmmc1.dll    $(_FLATRELEASEDIR)\s3c6410_hsmmc1.dll        NK   SHMK
s3c6410_camera.dll    $(_FLATRELEASEDIR)\s3c6410_camera.dll        NK   SHMK
s3c6410_touch.dll     $(_FLATRELEASEDIR)\s3c6410_touch.dll         NK   SHQM
s3c6410_keypad.dll    $(_FLATRELEASEDIR)\s3c6410_keypad.dll        NK   SHMK
backlight.dll         $(_FLATRELEASEDIR)\backlight.dll             NK   SHMK

btsvc.dll             $(_FLATRELEASEDIR)\btsvc.dll                 NK   SHM
btagsvc.dll           $(_FLATRELEASEDIR)\btagsvc.dll               NK   SHM
credsvc.dll           $(_FLATRELEASEDIR)\credsvc.dll               NK   SHM
serial.dll            $(_FLATRELEASEDIR)\com_card.dll              NK   SHMK
netbios.dll           $(_FLATRELEASEDIR)\netbios.dll               NK   SHMK
gwes.dll              $(_FLATRELEASEDIR)\gwes.dll                  NK   SHMK
ndis.dll              $(_FLATRELEASEDIR)\ndis.dll                  NK   SHMK
nleddrvr.dll          $(_FLATRELEASEDIR)\nleddrvr.dll              NK   SHMK
sdbus.dll             $(_FLATRELEASEDIR)\sdbus.dll                 NK   SHMK
sdmemory.dll          $(_FLATRELEASEDIR)\sdmemory.dll              NK   SHMK
serialusbfn.dll       $(_FLATRELEASEDIR)\serialusbfn.dll           NK   SHMK
sio950.dll            $(_FLATRELEASEDIR)\sio950.dll                NK   SHMK
softkb.dll            $(_FLATRELEASEDIR)\softkb.dll                NK   SHM
timesvc.dll           $(_FLATRELEASEDIR)\timesvc.dll               NK   SHM
uiproxy.dll           $(_FLATRELEASEDIR)\uiproxy.dll               NK   SHM
usbdisk6.dll          $(_FLATRELEASEDIR)\usbdisk6.dll              NK   SHMK
wcestreambt.dll       $(_FLATRELEASEDIR)\wcestreambt.dll           NK   SHMK
wendyser.dll          $(_FLATRELEASEDIR)\wendyser.dll              NK   SHMK
```