Application Note

Porting RZK/ZTP Applications to the eZ80F91x150MODG Module

AN036201-0413

Abstract

This document describes procedures for porting ZTP 2.4.x applications to the eZ80F91x150MODG Module (Zilog Module Label 99C1380) and the eZ80F91 Development Board (99C0858-001). This new eZ80F91x150MODG Module has the same physical form factor as the eZ80F915050MODG and eZ80F917050SBCG modules (99C0879) it replaces; however, it uses a different Ethernet PHY controller and external Flash module.

This new eZ80F91x150MODG Module (99C1380) is affixed to the Board as indicated in Figure 1.

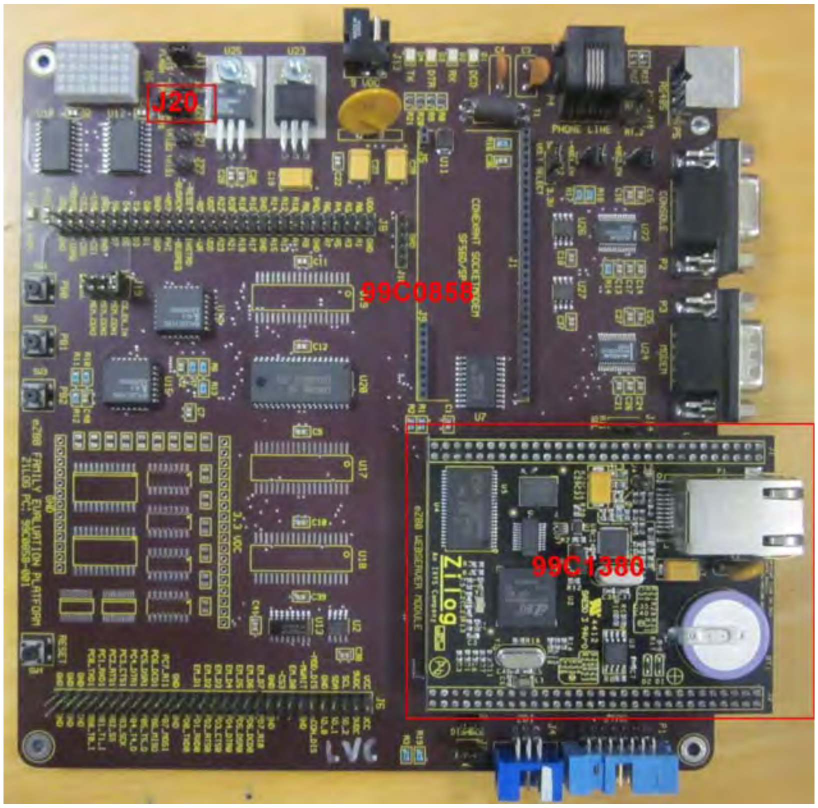


Figure 1. The eZ80F91 Development Board

Note: The application discussed in this document is applicable to ZDS II for eZ80Acclaim! version 5.2.1 (and later) and to ZTP version 2.4.1 (and later).

Prior to a discussion of the differences between the older module and the newer module, it is important to present a matrix for easy identification of the two modules, as indicated in Table 1. The Part Number Mnemonic column in this table represents how each module is referenced throughout the remainder of this document.

Table 1. Module Identification Matrix

Description	Module To Be Replaced	Replacement Module
eZ80Acclaim!		
Zilog Part Number	eZ80F915150MODG	eZ80F915150MODG
Module Label	99C0879-002G	99C1380-001G
eZ80Acclaim*Plus!*		
Zilog Part Number	eZ80F917050SBCG	eZ80F917150MODG
Module Label	99C0879-001G	99C1380-002G

Note: In the event that the module label is either not visible or is lost, the unit can be identified using the topmark on the MCU – in particular, the 10th. The eZ80Acclaim!/99C1380-001G module displays a topmark of `eZ80F91NA050SG`, and the eZ80Acclaim*Plus!*/99C1380-002G module displays a topmark of `eZ80F91NAA50SG`.

Differences Between the Modules

The hardware differences between the eZ80F91x050MODG/SBCG and eZ80F91x150MODG modules are summarized in Table 2. Click the links in the Affected Project Settings and Application Modification columns to learn more about how these changes affect ZTP applications.

Table 2. Hardware Implementation: Differences Between Modules

Component/ Feature	99C0879 Hardware Implementation	99C1380 Hardware Implementation	Affected Project Settings	Application Modification
External Flash on CS0	1MB AMD AM29LV008B	• 8MB Spansion S29GL064N • eZ80F91 Development Board Jumper Changes – see page 4	• CS0 Configuration – see page 3 • New ZTGT Files and Address Spaces – see page 4 • eZ80F910300KITG Preprocessor Definition – see page 11	Optional ZFS Configuration – see page 12

Table 2. Hardware Implementation: Differences Between Modules

Component/ Feature	99C0879 Hardware Implementation	99C1380 Hardware Implementation	Affected Project Settings	Application Modification
Ethernet PHY Controller	AMD NetPHY AM79C874	IDT ICS1894	eZ80F910300KITG Preprocessor Definition – see page 11	N/A
External Serial Flash	N/A	Spansion S25FL032P on SPI	N/A	Optionally include new code to interface to the external SPI Flash.
IrDA	Yes	No	N/A	Remove IRDA code.

These hardware changes require modifications to your ZTP project before your application will function with the eZ80F91x150MODG Module. Although this document describes the changes required specifically when using the eZ80F91 Development Board, similar project file changes may also be required when using custom hardware platforms with the eZ80F91x150MODG Module.

To learn more about the eZ80F91x150MODG Module, refer to the eZ80Acclaim!/ eZ80Acclaim Plus! Ethernet Module Product Specification (PS0306).

For a demonstration program that includes the changes required to support the eZ80F91x150MODG Module, refer to the `ZTPDemo_eZ80F91x150MODG.zdsproj` file in the `AN0362-SC01_<version>.exe` installation package.

The remaining sections of this document describe the project and/or application changes required to support the hardware changes.

CS0 Configuration

1MB of external Flash memory is mapped to CS0 on the eZ80F91x050MODG/SBCG Module. Applications that used this older module typically configured CS0 as follows:

```
CS0 Lower Bound   0x100000
CS0 Upper Bound   0x1FFFFF
CS0 Control       0xA8
CS0 Bus Mode      0x02
```

The eZ80F91x150MODG Module is equipped with 8MB of Flash mapped to CS0. To avoid conflicts with SRAM memory that is mapped to the first 512KB to 1MB of the eZ80F91 address space in RAM build configurations, CS0 should continue to be configured with a lower bound of `0x100000`. This configuration would imply a CS0 upper bound of `0x8FFFFF` on the eZ80F91x150MODG Module.

However, be advised that when using the eZ80F91 Development Board with the eZ80F91x150MODG Module, the GPIO and LED matrix are mapped into CS2 memory at

0x800000 and the gate array logic on the Development Board limits the upper bound of the CS0 map to 0x7FFFFF. Consequently, Zilog recommends the following CS0 settings for use with the eZ80F91x150MODG Module on the eZ80F91 Development Board:

```
CS0 Lower Bound   0x100000
CS0 Upper Bound   0x7FFFFF
CS0 Control       0x88
CS0 Bus Mode      0x01
```

eZ80F91 Development Board Jumper Changes

When using the eZ80F91x150MODG Module with the eZ80F91Development Board, it is typically necessary to install Jumper J20. By default, the external chip select signal on the eZ80F91Development Board, nCS_EX, has a jumper in the 1-2 position, which routes memory requests in the CS0 range of 0x400000 to 0x7FFFFF to the eZ80F91 Development Board. As a result, access is prevented to the upper 4MB of the external 8MB Flash space on the eZ80F91x150MODG Module. To maximize the amount of external Flash memory the eZ80F91 MCU can access on the eZ80F91x150MODG Module, it is necessary to install a jumper on J20.

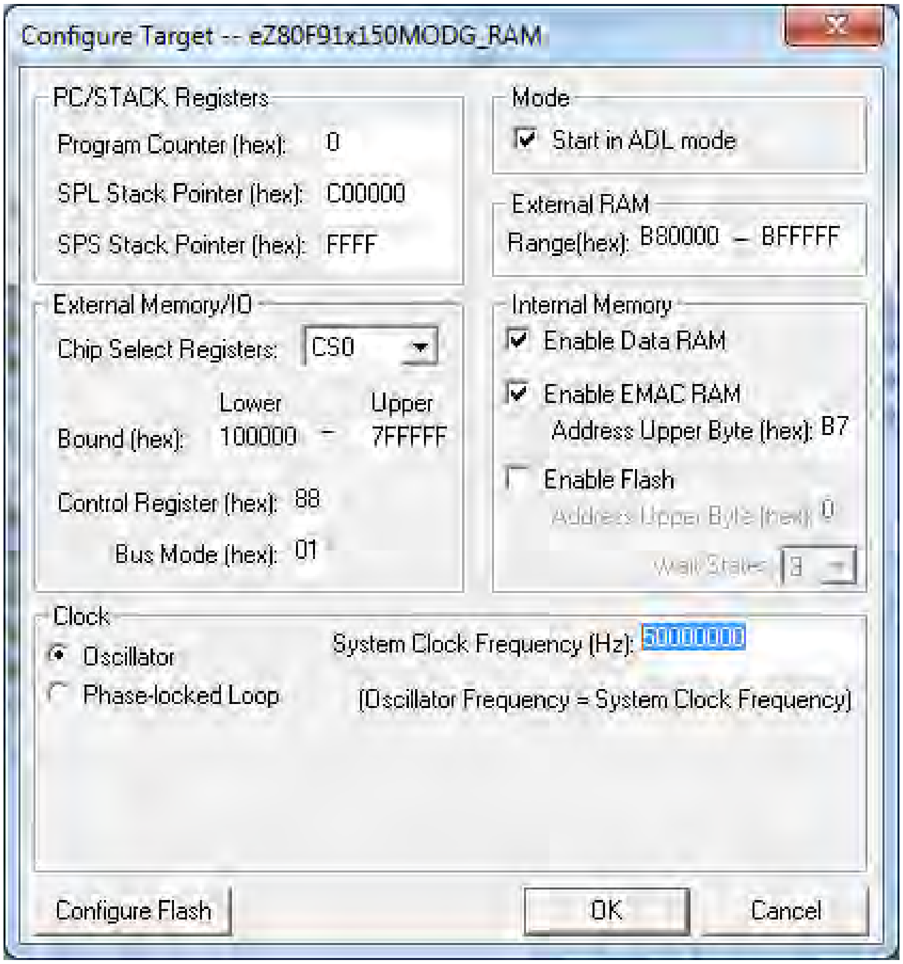
New ZTGT Files and Address Spaces

The AN0362-SC01_<version>.exe installation and later ZDS II – eZ80Acclaim! releases include two new Zilog Target (ZTGT) files for the eZ80F91x150MODG Module. These files are named eZ80F91x150MODG_RAM.ztgt and eZ80F91x150MODG_Flash.ztgt.

RAM Build Configuration

For applications using the RAM build configuration, select the eZ80F91x150MODG_RAM target by observing the following brief procedure.

1. From the **Project** menu in ZDS II, choose **Settings...** to open the Project Settings dialog.
2. In the left pane, click **Debugger**. In the Target pane, select the eZ80F91x150MODG_RAM target.
3. Click the **Setup** button. The Configure Target dialog will appear, as shown in Figure 2, which presents the default settings for the eZ80F91x150MODG_RAM ZTGT file. If you have not modified the contents of the eZ80F91x150MODG_RAM file, then you should see the values in Figure 2 after clicking the Setup button. Verify that your settings match the values shown in this figure.



Configure Target -- eZ80F91x150MODG_RAM

PC/STACK Registers
Program Counter (hex): 0
SPL Stack Pointer (hex): C00000
SPS Stack Pointer (hex): FFFF

Mode
☑ Start in ADL mode

External RAM
Range(hex): B80000 – BFFFFF

External Memory/IO
Chip Select Registers: CS0
Bound (hex): Lower 100000 – Upper 7FFFFF
Control Register (hex): 88
Bus Mode (hex): 01

Internal Memory
☑ Enable Data RAM
☑ Enable EMAC RAM
Address Upper Byte (hex): B7
□ Enable Flash
Address Upper Byte (hex): 0
Wait States: 3

Clock
● Oscillator
○ Phase-locked Loop
System Clock Frequency (Hz): 50000000
(Oscillator Frequency = System Clock Frequency)

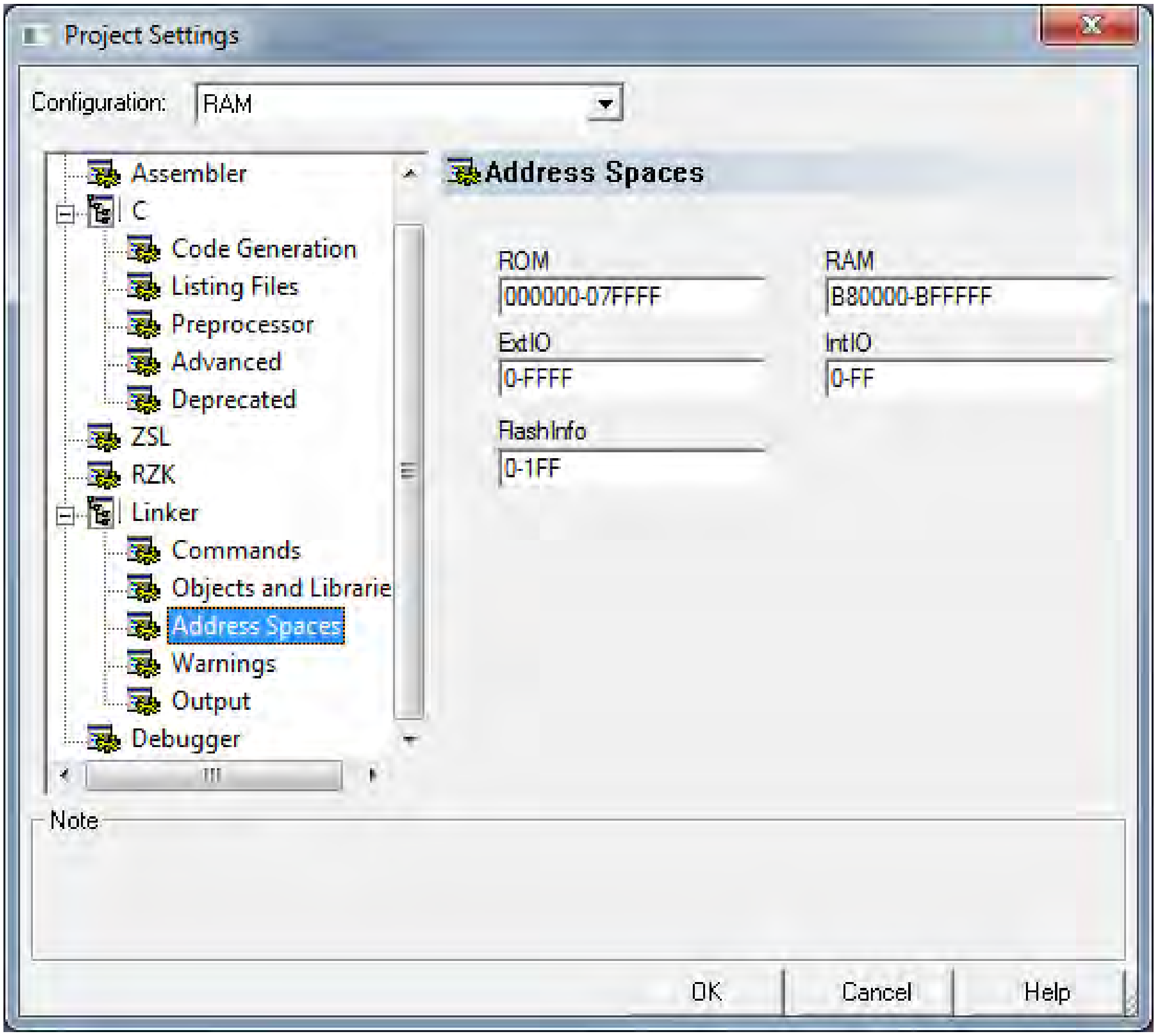
Figure 2. RAM Build Configuration: Default Settings

4. Examine the settings of the remaining chip selects in the **Chip Select Registers:** drop-down menu, and verify that the settings for CS0 to CS3 correspond to the values shown in Table 3.

Table 3. RAM Build Configuration: Chip Select Settings

Chip Select	Memory Range	CS Control	CS Bus Mode
CS0	0x100000–0x7FFFFF	0x88	0x01
CS1	0x000000–0x07FFFF	0x28	0x01
CS2	0x800000–0xBFFFFF	0x28	0x81
CS3	0x000000–0x000000	0x00	0x00

5. When configuring RAM build configurations using the eZ80F91x150MODG Module with the eZ80F91 Development Board, the recommended settings in the **Address Spaces** selection of the Project Settings dialog are shown in Figure 3. Ensure that your project's settings match the values displayed in this figure.



Configuration: RAM

Address Spaces

ROM: 000000-07FFFF

RAM: B80000-BFFFFF

ExtIO: 0-FFFF

IntIO: 0-FF

FlashInfo: 0-1FF

Figure 3. RAM Build Configuration: Recommended Address Space Settings

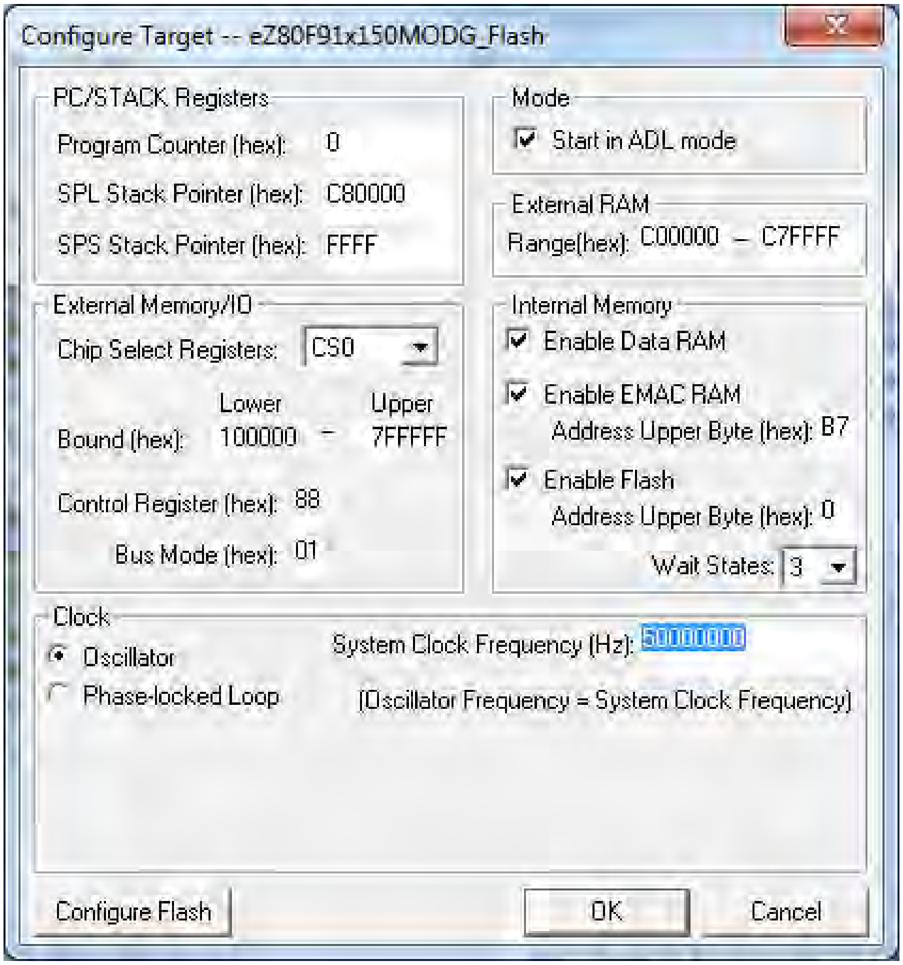
Flash and Copy to RAM Build Configurations

Applications using the FLASH or COPY TO RAM build configuration should select the `eZ80F91x150MODG_Flash` target by observing the following brief procedure:

1. From the **Project** menu in ZDS II, choose **Settings...** to open the Project Settings dialog.

2. In the left pane, click **Debugger**. In the Target pane, select the `eZ80F91x150MODG_Flash` target.

3. Click the **Setup** button. The Configure Target dialog will appear, as shown in Figure 4, which presents the default settings for the eZ80F91x150MODG_Flash ZTGT file. If you have not modified the contents of the eZ80F91x150MODG_Flash file, then you should see the values in Figure 4 after clicking the Setup button. Verify that your settings match the values shown in this figure.



Configure Target -- eZ80F91x150MODG_Flash

PC/STACK Registers
Program Counter (hex): 0
SPL Stack Pointer (hex): C80000
SPS Stack Pointer (hex): FFFF

Mode
☑ Start in ADL mode

External RAM
Range(hex): C00000 – C7FFFF

External Memory/IO
Chip Select Registers: CS0
Bound (hex): Lower 100000 – Upper 7FFFFF
Control Register (hex): 88
Bus Mode (hex): 01

Internal Memory
☑ Enable Data RAM
☑ Enable EMAC RAM
Address Upper Byte (hex): B7
☑ Enable Flash
Address Upper Byte (hex): 0
Wait States: 3

Clock
● Oscillator
○ Phase-locked Loop
System Clock Frequency (Hz): 50000000
(Oscillator Frequency = System Clock Frequency)

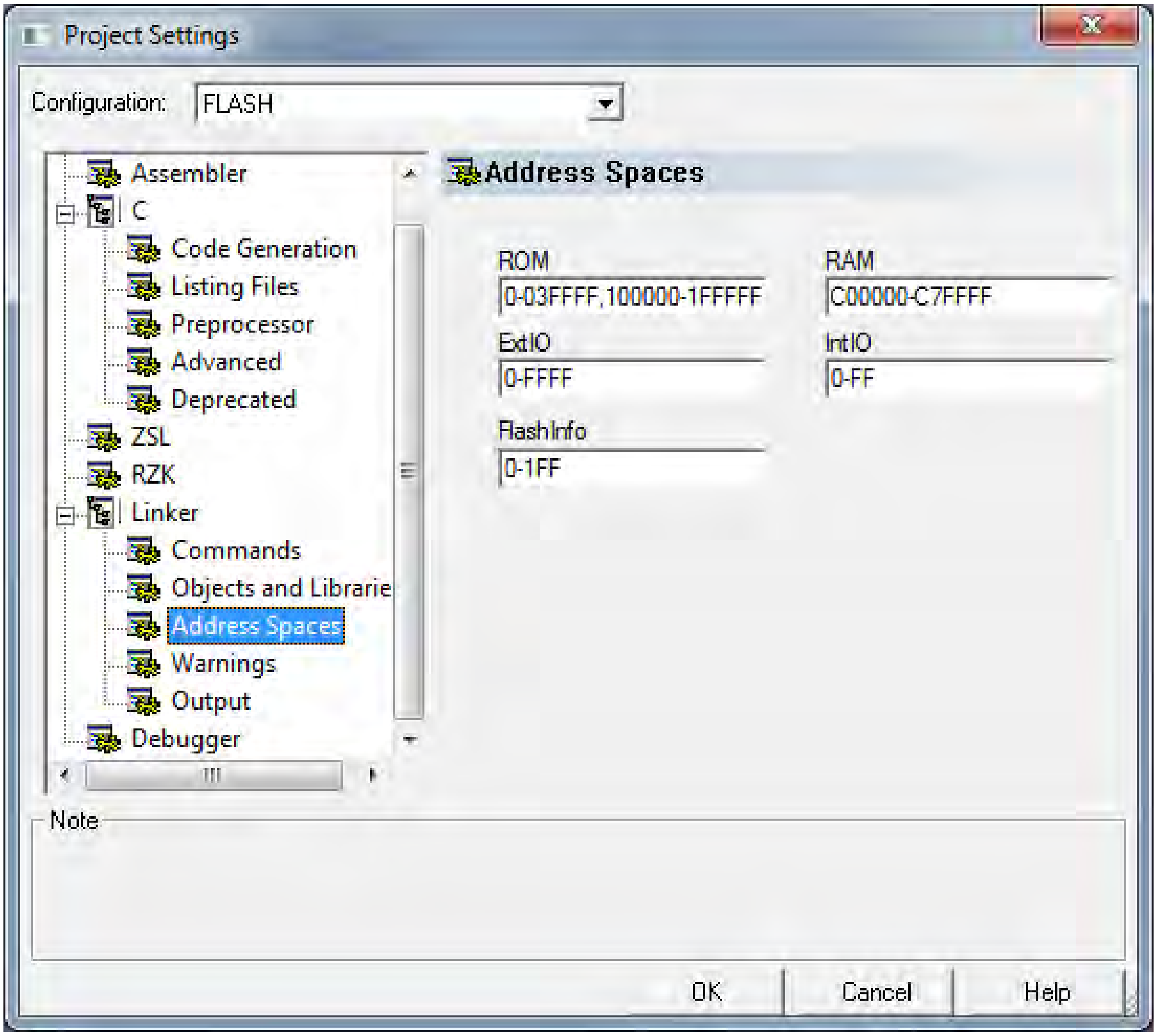
Figure 4. Flash and Copy to RAM Build Configuration: Default Settings

4. Examine the settings of the remaining chip selects in the **Chip Select Registers:** drop-down menu, and verify that the settings for CS0 to CS3 correspond to the values shown in Table 4.

Table 4. Flash and Copy to RAM: Chip Select Settings

Chip Select	Memory Range	CS Control	CS Bus Mode
CS0	0x100000–0x7FFFFF	0x88	0x01
CS1	0xC00000–0xC7FFFF	0x28	0x01
CS2	0x800000–0xBFFFFF	0x28	0x81
CS3	0x000000–0x000000	0x00	0x00

When using the eZ80F91x150MODG Module with the eZ80F91 Development Board, the recommended settings for the Address Spaces tab in the Project Settings menu for Flash build configurations are shown in Figure 5. Verify that your settings match the values in this figure.



Configuration: FLASH

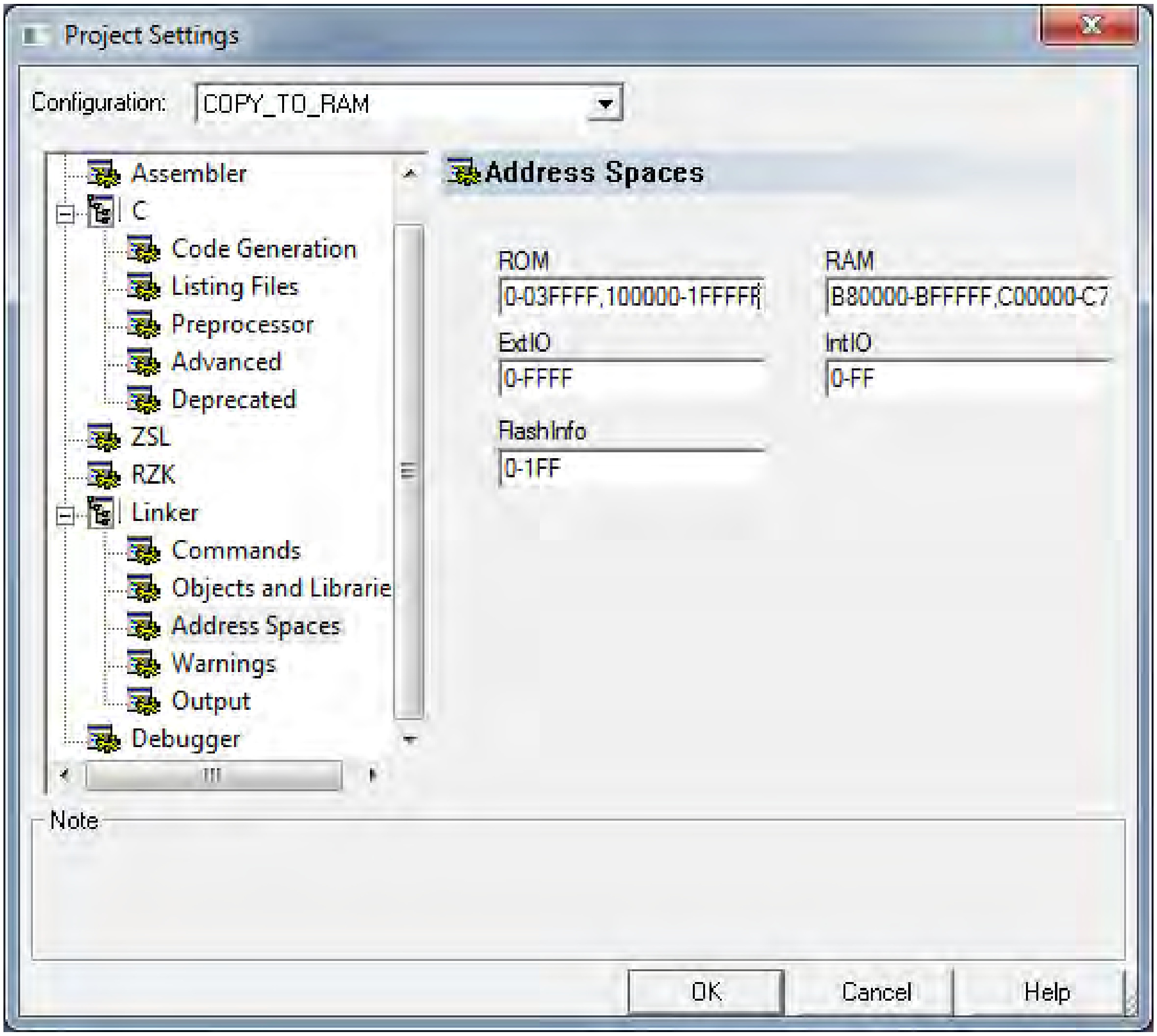
Address Spaces

ROM: 0-03FFFF,100000-1FFFFF
RAM: C00000-C7FFFF
ExtIO: 0-FFFF
IntIO: 0-FF
FlashInfo: 0-1FF

Figure 5. Flash Build Configuration: Recommended Address Space Settings

Note: The ROM address space shown in Figure 5 only includes the first 2 MB of external Flash memory. If your project requires additional ROM memory (i.e., code space), the upper range of the ROM address space can be increased up to `0x7FFFFF`.

5. When configuring Copy to RAM build configurations using the eZ80F91x150MODG Module with the eZ80F91 Development Board, the recommended settings in the Address Spaces selection of the Project Settings dialog are shown in Figure 6. Ensure that your project's settings match the values displayed in this figure.



Configuration: COPY_TO_RAM

ROM: 0-03FFFF,100000-1FFFFF

RAM: B80000-BFFFFF,C00000-C7

ExtIO: 0-FFFF

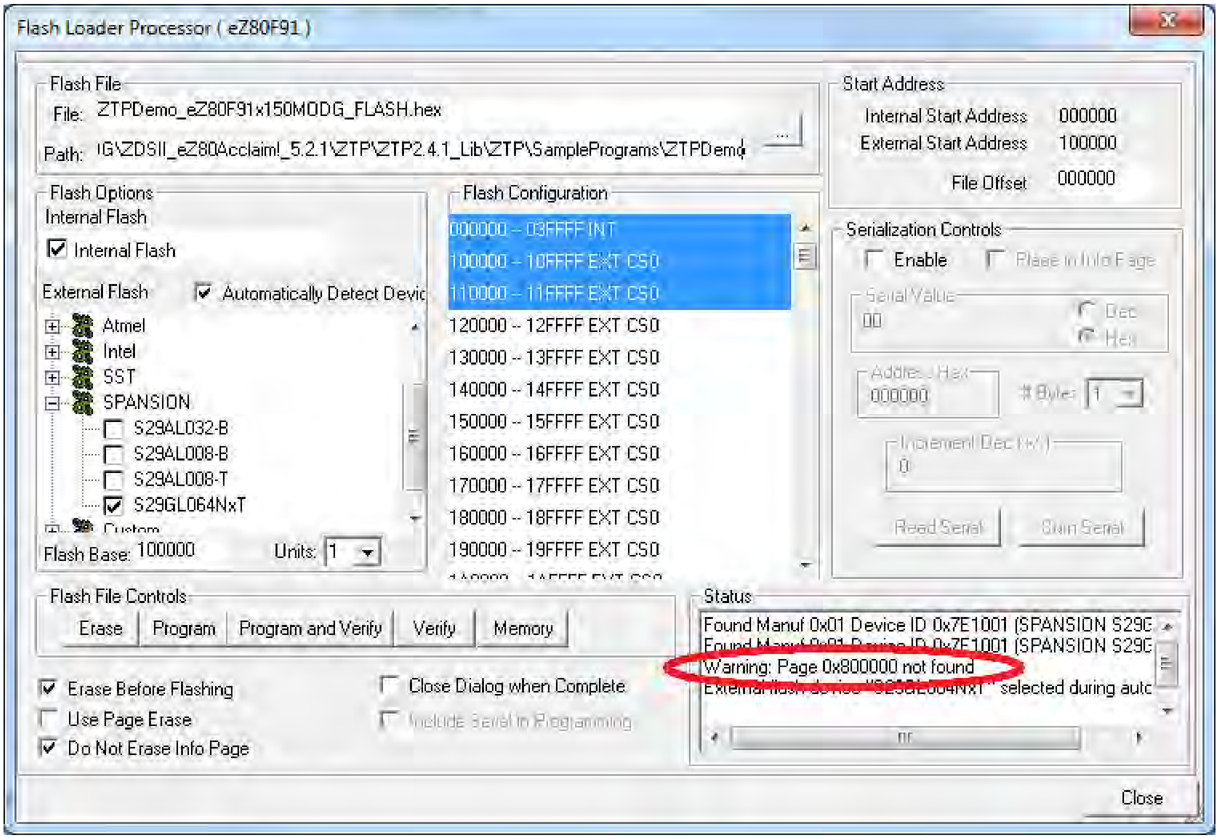
IntIO: 0-FF

FlashInfo: 0-1FF

Figure 6. Copy to RAM Build Configuration: Recommended Address Space Settings

ZDSII Flash Loader Support

When using the ZDSII Flash Loader to to program hex files for either the FLASH or COPY_TO_RAM build configurations of the `ZTPDemo_eZ80F91x150MODG.zdsproj` project, a warning message will be displayed in the Flash Loader Processor dialog's Status panel, as shown in Figure 7.



Flash Loader Processor (eZ80F91)

Flash File
File: ZTPDemo_eZ80F91x150MODG_FLASH.hex
Path: IG\ZDSII_eZ80Acclaim!_5.2.1\ZTP\ZTP2.4.1_Lib\ZTP\SamplePrograms\ZTPDemo

Start Address
Internal Start Address 000000
External Start Address 100000
File Offset 000000

Flash Options
Internal Flash
☑ Internal Flash
External Flash ☑ Automatically Detect Devic
- Atmel
- Intel
- SST
- SPANSION
  - □ S29AL032-B
  - □ S29AL008-B
  - □ S29AL008-T
  - ☑ S29GL064NxT
- Custom

Flash Base: 100000 Units: 1

Flash Configuration
000000 -- 03FFFF INT
100000 -- 10FFFF EXT CS0
110000 -- 11FFFF EXT CS0
120000 -- 12FFFF EXT CS0
130000 -- 13FFFF EXT CS0
140000 -- 14FFFF EXT CS0
150000 -- 15FFFF EXT CS0
160000 -- 16FFFF EXT CS0
170000 -- 17FFFF EXT CS0
180000 -- 18FFFF EXT CS0
190000 -- 19FFFF EXT CS0

Serialization Controls
□ Enable □ Place in Info Page
Serial Value 00 ○ Dec ● Hex
Address Hex 000000 # Bytes 1
Increment Dec (+/-) 0

Flash File Controls
☑ Erase Before Flashing
□ Use Page Erase
☑ Do Not Erase Info Page
□ Close Dialog when Complete
□ Include Serial in Programming

Status
```
Found Manuf 0x01 Device ID 0x7E1001 (SPANSION S29G
Found Manuf 0x01 Device ID 0x7E1001 (SPANSION S29G
Warning: Page 0x800000 not found
External flash device "S29GL064NxT" selected during auto
```

Figure 7. Flash Loader Status: Warning Message

This warning occurs because the 8MB Flash module on CS0 (Spansion S29GL064N) is mapped into the eZ80F91 MCU address space starting at address `0x100000`. The designs of the 99C1380 Module and the 99C0858 Development Board are such that addresses `0x800000` and above on CS0 are mapped to the 99C0858 Development Board. Therefore, when the ZDSII Flash Loader queries CS0 Flash memory, it determines that the 8MB external Flash memory space should start at address `0x100000` and end at address `0x8FFFFF`. However, the last 1MB of Flash memory will not be accessible (i.e., memory access above address `0x800000` on CS0 will be routed to the 99C0858 Development Board, which does not contain Spansion Flash).

Consequently, the ZDSII Flash Loader will not be able to locate any Flash memory at page `0x800000`, and will display the warning message shown in Figure 7. Because the

ZTPDemo_eZ80F91x150MODG project file in ZDS II limits the address space for ROM to 0x7FFFFF (as is shown in Table 4), there will never be any information in the hex file that the ZDS II Flash Loader must program to page 0x800000. Consequently, this warning message can be ignored.

eZ80F910300KITG Preprocessor Definition

The Ethernet PHY controller and the external Flash Module used on the eZ80F1x150MODG Module are functionally the same as the components used on the eZ80Acclaim*Plus!* Development Kit (eZ80F910300KITG/99C1322). RZK/ZTP projects that target the eZ80Acclaim*Plus!* Development Kit (eZ80F910300KITG) hardware include the eZ80F910300KITG preprocessor definition, which can be selected in the "C" Preprocessor tab in the Project Settings menu. This same definition should be used in all build configurations that target the eZ80F91x150MODG Module.

The eZ80F910300KITG preprocessor definition causes the compiler to use the ICS1894_phy.h header file when compiling the F91_PhyInit.c file, as shown in the following code fragment from F91_Phy_Init.c:

```
#ifdef eZ80F910300KITG
  #include "ICS1894_phy.h"
#else
  #include "AMD79C874_phy.h"
#endif
```

If the eZ80F910300KITG preprocessor definition is not included in the project settings, then it is necessary to modify the F91_PhyInit.c file to unconditionally include the ICS1894_phy.h header file.

The eZ80F910300KITG preprocessor definition also causes the compiler to use the ZFS configuration settings in the ..\RZK\Conf\ZFS_Conf_eZ80F910300KITG.c file, as shown in the following code fragment from ZFS_Conf.c.

```
#ifdef eZ80F910300KITG
  #include "ZFS_Conf_eZ80F910300KITG.c"
#else
  #include "ZFS_Conf_old.c"
#endif
```

If the eZ80F910300KITG preprocessor definition is not included in the project settings, then it is necessary to modify the ZFS_Conf.c file to explicitly include the ZFS_Conf_eZ80F910300KITG.c file.

Note: Learn more about the eZ80Acclaim*Plus!* Development Kit (eZ80F910300KITG) on the Zilog website.

Optional ZFS Configuration

RZK/ZTP applications that use the Zilog File System (ZFS) can optionally modify the ZFS configuration file to allocate more or less Flash memory to ZFS. The default settings in the ZFS_Conf_eZ80F910300KITG.c file are shown in the following code.

```
#define ZFS_TOTAL_NUM_BLOCKS        ( 32 )
#define ZFS_TOTAL_NUM_SECTORS       ( 0x200000/ZFS_SEC_SIZE )
#define ZFS_TOTAL_NUM_VOLUMES       ( 1 )

#define ZFS_MAX_FILE_OPEN_COUNT     ( 20 )
#define ZFS_MAX_DIRS_SUPPORTED      ( 50 )
#define ERASE_FLASH                 ( 0 )

ZFS_CONFIG_t g_zfs_cfg[ ZFS_TOTAL_NUM_VOLUMES ] =
{
  "EXTF",                           // vol name
  ZFS_EXT_FLASH_DEV_TYPE,           // vol type
  (UINT8*)0x300000,                 // vol_start_addr
  0x200000,                         // vol_size
  ZFS_TOTAL_NUM_BLOCKS,             // vol_blocks
  ZFS_TOTAL_NUM_SECTORS,            // number of sectors
  FS_AM29LV160_Init,
  FS_AM29LV160_Read,
  FS_AM29LV160_Write,
  FS_AM29LV160_Erase,
  FS_AM29LV160_Close
};
```

These settings will configure ZFS to use 2MB of CS0 Flash from addresses 0x300000 to 0x4FFFFF for ZFS. In total, 32 Flash erase blocks (each 64KB in size) are allocated to ZFS, resulting in a volume size of 0x200000 bytes. Source code to the Flash driver used to read and write data to this volume is contained in the AM29LV160_Driver.c file.

Note: The AM29LV160_Driver.c file (and other ZFS Flash drivers) was modified in the ZTP 2.4.2 release to explicitly disable interrupts while modifying the contents of Flash. This modification was necessary to ensure that an interrupt did not cause a higher-priority thread to start running from external Flash while Flash memory is being programmed or erased.

References

The following document is associated with the eZ80F91x150MODG Module and is available free for download from the Zilog website.

- eZ80Acclaim!™/eZ80Acclaim*Plus!*™ Ethernet Modules Product Specification (PS0306)

Customer Support

To share comments, get your technical questions answered, or report issues you may be experiencing with our products, please visit Zilog's Technical Support page at http://support.zilog.com.

To learn more about this product, find additional documentation, or to discover other facets about Zilog product offerings, please visit the Zilog Knowledge Base at http://zilog.com/kb or consider participating in the Zilog Forum at http://zilog.com/forum.

This publication is subject to replacement by a later edition. To determine whether a later edition exists, please visit the Zilog website at http://www.zilog.com.

Warning: DO NOT USE THIS PRODUCT IN LIFE SUPPORT SYSTEMS.

LIFE SUPPORT POLICY

ZILOG'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS PRIOR WRITTEN APPROVAL OF THE PRESIDENT AND GENERAL COUNSEL OF ZILOG CORPORATION.

As used herein

Life support devices or systems are devices which (a) are intended for surgical implant into the body, or (b) support or sustain life and whose failure to perform when properly used in accordance with instructions for use provided in the labeling can be reasonably expected to result in a significant injury to the user. A critical component is any component in a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system or to affect its safety or effectiveness.

Document Disclaimer

Information in this publication concerning the devices, applications, or technology described is intended to suggest possible uses and may be superseded. ZILOG, INC. DOES NOT ASSUME LIABILITY FOR OR PROVIDE A REPRESENTATION OF ACCURACY OF THE INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED IN THIS DOCUMENT. ZILOG ALSO DOES NOT ASSUME LIABILITY FOR INTELLECTUAL PROPERTY INFRINGEMENT RELATED IN ANY MANNER TO USE OF INFORMATION, DEVICES, OR TECHNOLOGY DESCRIBED HEREIN OR OTHERWISE. The information contained within this document has been verified according to the general principles of electrical and mechanical engineering.

Z8, Z8 Encore! and Z8 Encore! XP are trademarks or registered trademarks of Zilog, Inc. All other product or service names are the property of their respective owners.