CYPRESS®
EMBEDDED IN TOMORROW™

The following document contains information on Cypress products. The document has the series name, product name, and ordering part numbering with the prefix “MB”. However, Cypress will offer these products to new and existing customers with the series name, product name, and ordering part number with the prefix “CY”.

How to Check the Ordering Part Number

1. Go to www.cypress.com/pcn.
2. Enter the keyword (for example, ordering part number) in the **SEARCH PCNS** field and click **Apply**.
3. Click the corresponding title from the search results.
4. Download the Affected Parts List file, which has details of all changes

For More Information

Please contact your local sales office for additional information about Cypress products and solutions.

About Cypress

Cypress is the leader in advanced embedded system solutions for the world's most innovative automotive, industrial, smart home appliances, consumer electronics and medical products. Cypress' microcontrollers, analog ICs, wireless and USB-based connectivity solutions and reliable, high-performance memories help engineers design differentiated products and get them to market first. Cypress is committed to providing customers with the best support and development resources on the planet enabling them to disrupt markets by creating new product categories in record time. To learn more, go to www.cypress.com.

AN205155

FR Family, MB91F465XA Emulation

This application note describes the emulation system for MB91F465XA series. The current emulation system is based on EVA device MB91V460A, which does not include a FlexRay interface.

Contents

1 Introduction

This application note describes the emulation system for MB91F465XA series.

The current emulation system is based on EVA device MB91V460A, which does not include a FlexRay interface. For that reason the FlexRay interface is emulated by MB88121B series stand alone FlexRay communication controller. The MB88121B is connected to the bus interface of MB91V460A.

2 Hardware Setup

The chapter describes the setup of the Emulation system.

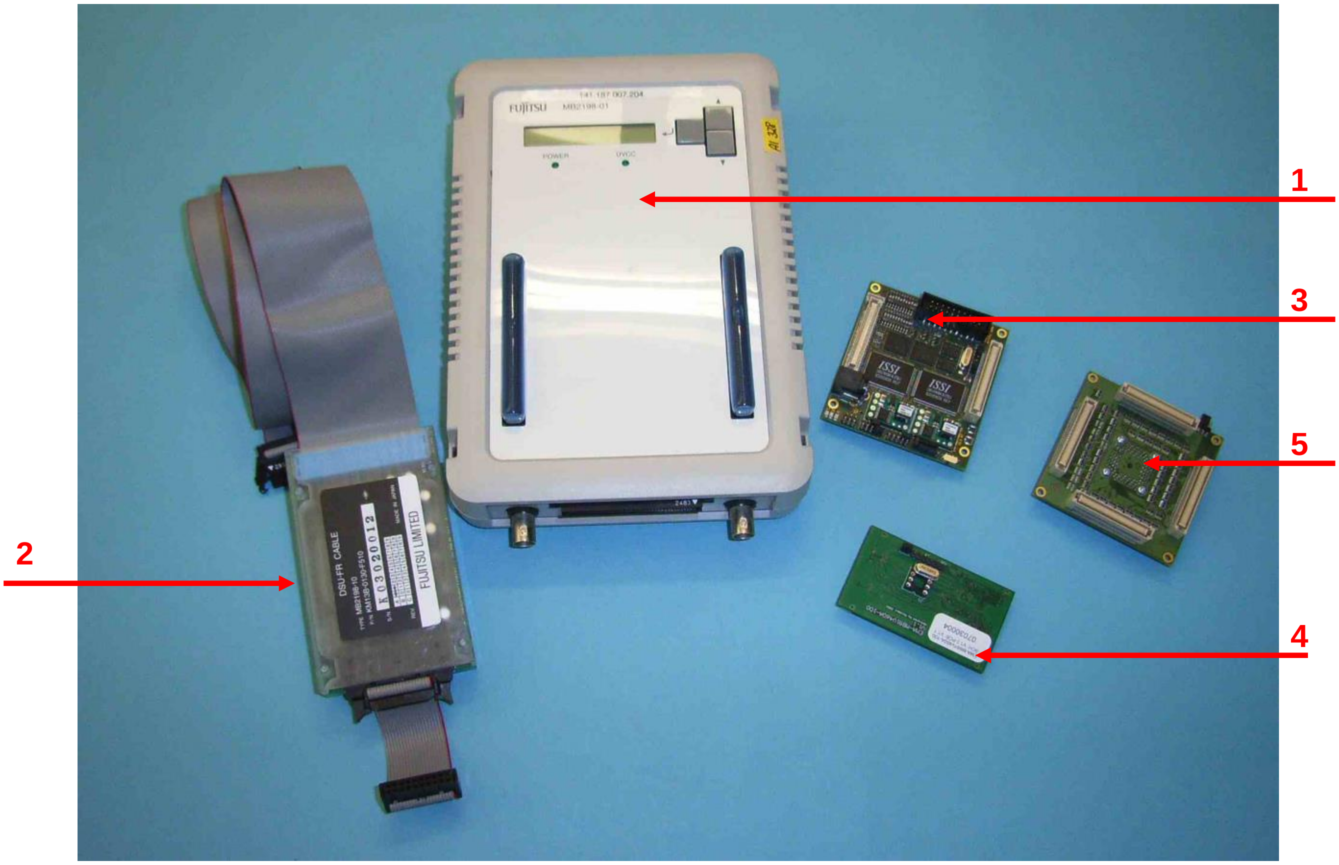
2.1 Required parts

To Emulate the MB91F456XA series following parts are required:

1.	MB2198-01 Emulator Main Unit	
2.	MB2198-10: DSU4 cable	
3.	EMA-MB91V460A-00x:	Adapter board, including MB91V460A
4.	EMA-MB91V460A-100:	FlexRay extension board, including MB88121B
5.	EMA-MB91F465X-NLS-100M20:	Socket adapter board
6.	NQPack100SD-ND:	Socket for package FPT-100P-M20 (Tokyo Eletech Corp http://www.tetc.co.jp/e)

On the target system a NQPack100SD-ND socket is required enabling connection to EMA-MB91F465X-NLS-100M20 board.

Figure 1. Hardware Parts



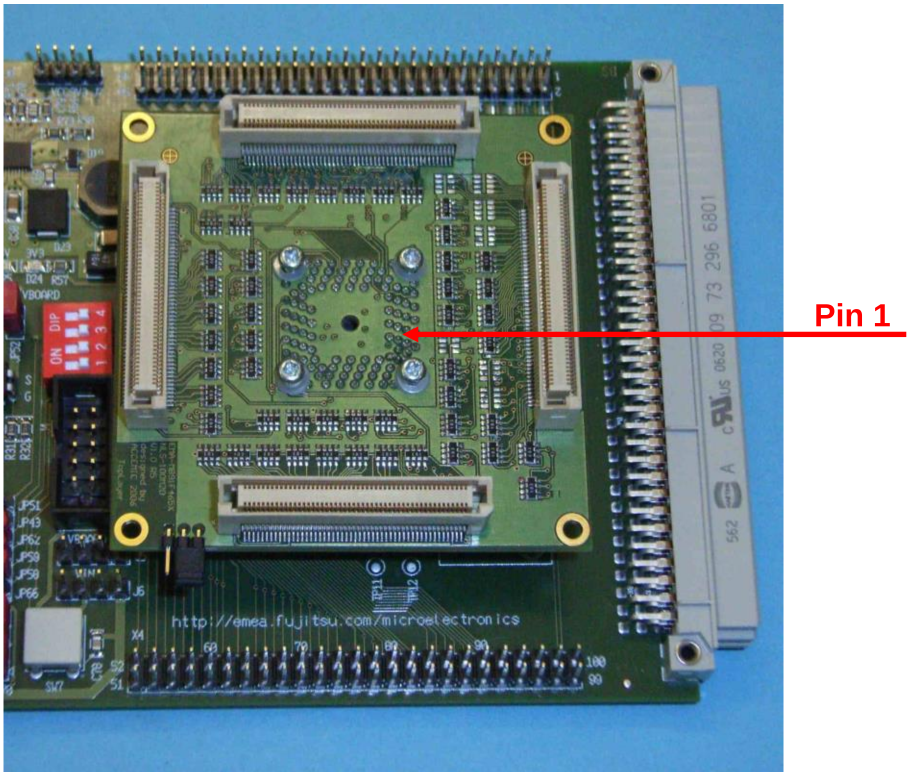
2.2 Installation

For detailed installation instruction see the user guides of the emulation tools.

In addition there are several application notes for MB2198-01 emulator available, describing features and software installation (e.g. USB drivers).

1. Connect the EMA-MB91F465X-NLS-100M20 socket adapter board to the NQPack100SD-ND of the target hardware (PCB). The target hardware must use a socket otherwise it is not possible to connect the socket adapter board to the target board. Fasten the four screws using the screw driver delivered in the package.

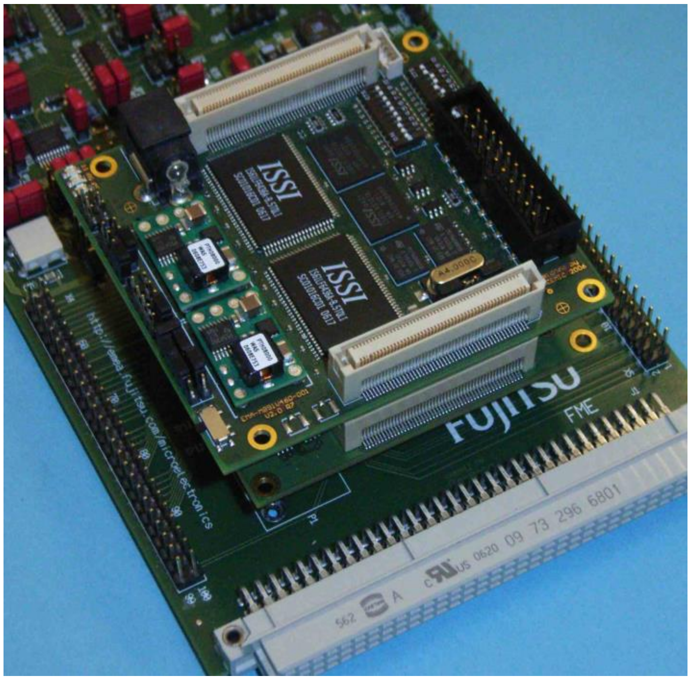
Figure 2. EMA-MB91F465X-NLS-100M20 at target system



Note: Remove the Flash chip from the socket before connecting the socket adapter board to the socket!

2. Plug the EMA-MB91V460A-00x board onto the EMA-MB91F465XA-NLS-100M20 board. The placement of the four connectors allows one position, only.

Figure 3. EMA-MB91V460-00x on top of EMA-MB91F465X-NLS-100M20



3. The EMA-MB91V460A-100 extension board must be plugged into the two connectors at top of the EMA-MB91V460A-00x board. Again, the position of the connectors allows only one possible connection.

Figure 4. EMA-MB91V460A-100 on top of EMA-MB91V460A-00x

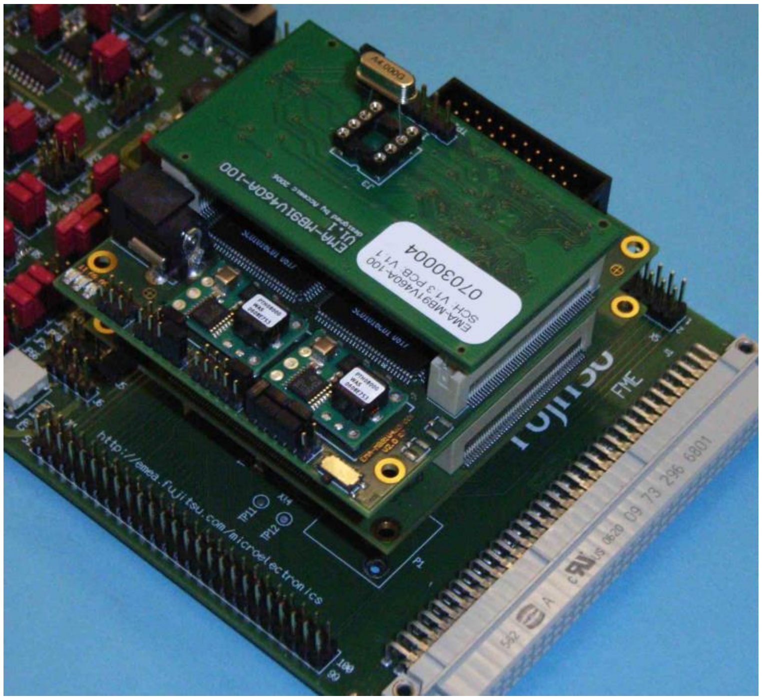
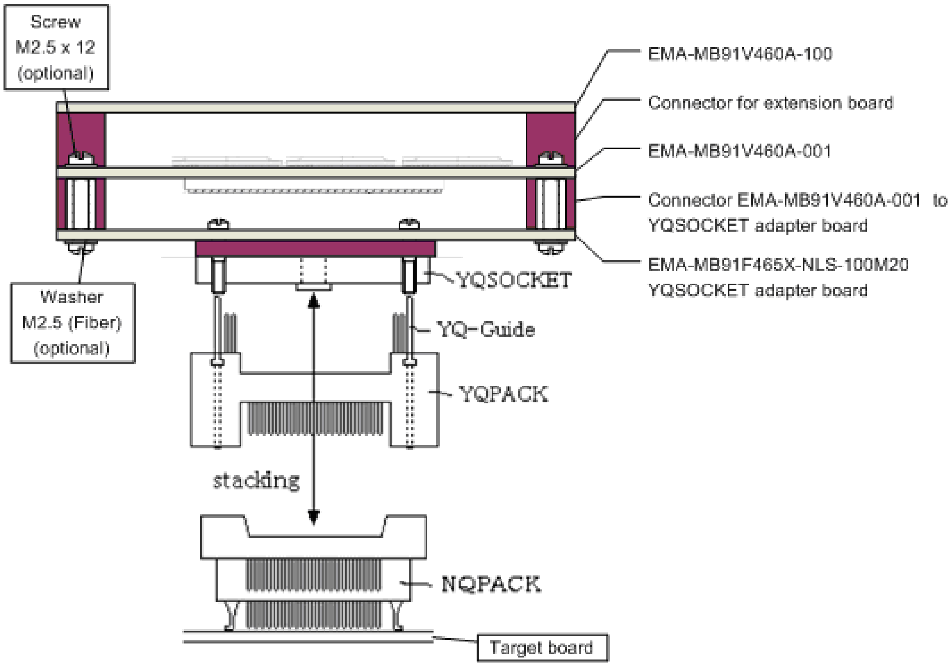
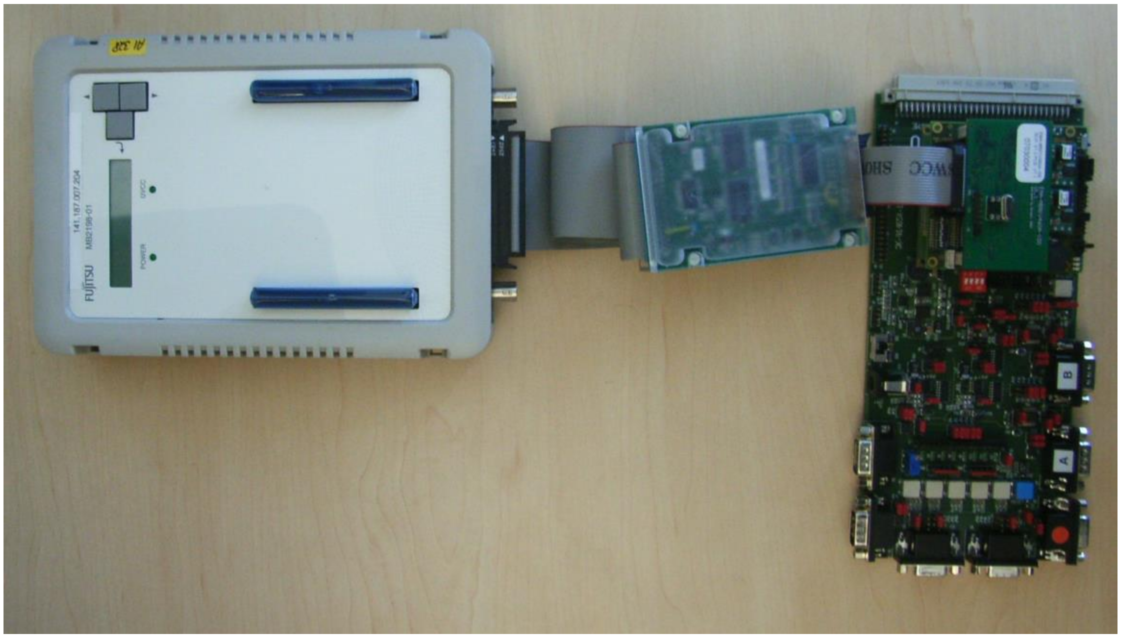


Figure 5. An overview



4. Connect the MB2198-01 DSU cable to the MB2198-01 Emulator and to the EMA-91V460A-00x board.

Figure 6. MB2198-10 Connected to MB2198-01 and EMA-MB91V460A-00x



5. Plug in the power supply of MB2198-01, EMA-MB91V460A-00x and the target system into the according plugs.
6. Power all parts, afterwards the connection to the Softune Workbench Debug Environment is possible.

Following order should be used to power up the emulation system

- Emulator MB2198-01
- Adapter board EMA-MB91V460A-00x
- Target system

To power down the system use the inverse order:

- Target system
- Adapter board EMA-MB91V460A-00x
- Emulator MB2198-01

Note: When using the Cypress SK-91465X-100PMC evaluation board, the polarity of power supply is contrary to that of the EMA-MB91V460A-00x!

3 Differences between MB91F465XA and Emulation System

The chapter describes the differences between the emulation system and MB91F465XA series.

3.1 Overview about differences

Table 1. Overview of the Differences between MB91F465XA and Emulation System

Feature	Emulation system (MB91V460A + MB88121B)	MB91F465XA series
Maximum Operating Frequency	MB91V460A: 80 MHz	100MHz
FlexRay Port 31	Not available	Must be set
E-Ray Register start address	0x50.0000	0x00.D000
E-Ray IP version	1.0 RC1	1.0
CIF0 (CUST0) Register value	0x0430.79FF	0x04FF5BFF
CREL Register value	0x0726.0412	0x1006.0519
E-Ray Clock Register	CIF1 (CUST1) register	PLL2 registers
E-Ray Interrupts	E-Ray interrupts connected to external Interrupts	E-Ray interrupts set in E-Ray interrupt register
Interrupt vector table (Intvect)	External Interrupt	FlexRay Interrupts
DMA	Setup via external bus DMA transfer request	Setup via CIF1 / 2 register

3.2 Details of differences

3.2.1 Maximum operating frequency

The maximum operating frequency of the emulation device MB91V460A is 80MHz. Do not set higher PLL frequencies! The MB91F465XA series has a maximum operating frequency of 100MHz.

By using the Emulator take care that the PLL frequency should not exceed 80MHz. The PLL options can be set in the file start91460.asm (See Chapter 4.7.1 Clock Selection in start91460.asm file).

For further details refer to MB91460A series Hardware Manual and MB91F465XA series Datasheet.

3.2.2 FlexRay port 31

Each pin of the MCU is shared by general I/O port function and peripheral function. Via the Port Function Register (PFR) the pin function is specified.

- **MB91F465XA:**

Port 31 must be set to peripheral function (PFR=1), otherwise the FlexRay functionality is not connected to the pins.

```
Recommended FlexRay port settings:
1. Channel A
   TXDA:   FlexRay Channel A Transmit (P31_0). To enable TXDA set PFR31_0 = 1 and EPFR31_0 = 1
   TXENA: FlexRay Channel A Enable (P31_1). To enable TXENA set PFR31_1 = 1 and EPFR31_1 = 1
   RXDA:   FlexRay Channel A Receive (P31_2). To enable RXDA set PFR31_2 = 1 and EPFR31_2 = 1
2. Channel B
   TXDB:   FlexRay Channel B Transmit (P31_4). To enable TXDB set PFR31_4 = 1 and EPFR31_4 = 1
   TXENB: FlexRay Channel B Enable (P31_5). To enable TXENB set PFR31_5 = 1 and EPFR31_5 = 1
   RXDB:   FlexRay Channel B Receive (P31_6). To enable RXDB set PFR31_6 = 1 and EPFR31_6 = 1

The muster:
  EPFR31 = 0x77;
  PFR31 = 0x77;
  DDR31 = 0x77;     /* output mode */
```

- **Emulation system**:

The FlexRay functionality is always available at port 31. It is not possible to use the general I/O Port function.

For further details refer to MB91460A series Hardware Manual and MB91F465XA series Datasheet.

3.2.3 E-Ray register start address

The E-Ray register start address of the emulation system (MB91V460A + MB88121B) is: **0x50.0000**. By MB91F465XA series the E-Ray register address starts from **0x00.D000**.

Cypress provides two different header files for them.

1. **mb91465x.h**: for MB91F465XA series
2. **mb91465x_emulation.h**: for emulation system (MB91V460A + MB88121B)

For further details refer to MB91460A series Hardware Manual and MB91F465XA series Datasheet.

3.2.4 E-Ray IP version

The MB91F465XA is using E-Ray IP version **1.0**

The MB88121B is using E-Ray IP version **1.0RC1**

For further details refer to MB88121 and MB91F465XA series Datasheet.

3.2.5 CIF0 (CUST0) register value (Customer Register)

The value of register CIF0 (address: 0x00D000) by MB91F465XA series: **0x04FF.5BFF**

The value of register CIF0 (address: 0x500000) by MB88121B: **0x0430.79FF**

For further details refer to MB88121 and MB91F465XA series Datasheet.

3.2.6 CREL register value (Core Release Register)

The value of register CREL (address: 0x00D3F0) by MB91F465XA series: **0x1006.0519**

The value of register CREL (address: 0x5003F0) by MB88121B: **0x0726.0412**

The CREL register might be used by software to identify the used E-Ray IP version. The software should be aware of the emulation system value.

For further details refer to MB88121 and MB91F465XA series Datasheet.

3.2.7 E-Ray clock control register

The FlexRay communication controller works on an internal clock frequency of 80MHz. In order to achieve this, a PLL needs to be set.

- **Emulation System (MB91V460A + MB88121B):**

The CIF1 (address: 0x500004) register specifies the PLL settings for MB88121B.

- **MB91F465XA series:**

The internal FlexRay CC uses a separate PLL (PLL2). The PLL2 registers (pll2divm, pll2divn, pll2divg, pll2mulg, pll2ctrl, pll2clkr2) control the FlexRay PLL.

For further details refer to MB88121 and MB91F465XA series Datasheet.

3.2.8 E-Ray Interrupts

The FlexRay communication controller can be configured to request interrupts to MCU for different occurrence (See E-Ray User's Manual).

- MB91F465XA:

In MB91F465XA series four E-Ray interrupt sources are assigned to **four interrupt vector numbers**:

- E-Ray interrupt line0: number 84
- E-Ray interrupt line1: number 86
- E-Ray timer 0: number 85
- E-Ray timer 1: number 87

The E-Ray interrupt registers must be setup accordingly.

In the interrupt service routine (ISR) the according E-Ray interrupt flag must be cleared.

To define the interrupt level register ICR34 and ICR35 must be set.

For further details refer to file vectors.c and MB91F465XA series Datasheet.

- **Emulation System:**

For the emulation system four E-Ray interrupts are connected to **three MCU external interrupts inputs**:

- E-Ray interrupt line0 is connected to external interrupt 15
- E-Ray interrupt line1 is connected to external interrupt 13
- E-Ray interrupt timer 0 and timer 1 are connected to external interrupt 11

In addition to the E-Ray interrupt registers the according external interrupt registers must also be set. Rising edge detection should be chosen. Port function must be set accordingly for external interrupt input (DDR=0, PFR=1).

In the Interrupt service routine (ISR) both of the E-Ray interrupt flag and the external interrupt flag must be cleared.

To define the interrupt level set the register ICR05, ICR06 and ICR07.

For further details refer to file vectors.c and MB91460A series hardware manual.

3.2.9 Interrupt Vector Table (file vectors.c)

As mentioned above the emulation system uses a different interrupt assignment than the MB91F465XA series. This results in different number usage in the interrupt vector table.

- **MB91F465XA series:**
  - E-Ray interrupt line0 (number 84)
  - E-Ray interrupt line1 (number 86)
  - E-Ray timer 0 (number 85)
  - E-Ray timer 1 (number 87)
- **Emulation System:**
  - E-Ray interrupt line0 is connected to external interrupt 15 (number 31)
  - E-Ray interrupt line1 is connected to external interrupt 13 (number 29)
  - E-Ray interrupt timer 0 and timer 1 are connected to external interrupt 11 (number 27)

The template project of MB91F465XA series contains the file vector.c, where the interrupt vector table can be specified.

3.2.10 DMA

To speed-up the data transfer from output buffer or to input buffer, it is possible to launch a DMA transfer.

- **MB91F465XA:**

In MB91F465XA series register CIF1 (address 0x00D004) and CIF2 (address 0x00D008) are used to setup the DMA transfer.

- **Emulation system:**

In emulation system the DMA transfer is realized via external bus interface. In MB88121B the register CIF2 must be setup accordingly (See the MB88121B CIF2 register description in Appendix).

For further details refer to MB88121 and MB91F465XA series Datasheet.

4 Softune Workbench

The chapter describes the Softune Workbench Development environment for MB91F465XA.

4.1 Update of Softune Workbench

The Softune Workbench V6 for FR series on CD / DVD up to DVD version 5.1 does not fully support MB91F465XA emulation. (Softune Workbench Version <= V06L06)

For this reason the 911.csv file needs to be updated. Extract the ZIP archive (911_csv_01_26_02_465x.zip) into the folder **C:\Softune6\lib\911\** (Softune6 is the default Softune Workbench installation folder).

Of course the Softune Workbench V6 for FR series must be installed before updating the csv file!

4.2 Template Workspace for MB91F465XA series

In order to allow a quick and smooth project start-up Cypress supplies a template project as a reference. The easiest way to start a new project is to make a copy of the template project and use this copy as a start-up. The template includes the latest start91460.asm file, MCU header file (mb91465x.h and mb91465x_emulator.h), IRQ table (vectors.c), basic linker and C-Compiler settings.

Note: In any case the settings done in the template must be checked and have to be adjusted to the specific needs and settings of the final application. The template is providing an example for building up a new project.

THIS SAMPLE CODE IS PROVIDED AS IS AND IS SUBJECT TO ALTERATIONS. CYPRESS ACCEPTS NO RESPONSIBILITY OR LIABILITY FOR ANY ERRORS OR/ELIGIBILITY FOR ANY PURPOSES.

Due to the differences between the FlexRay communication controller in MB91F465XA and emulation system, two configurations are available.

- STANDALONE: for the MB91F465XA series
- EMULATOR: for the emulation system

Depending on the selection of the **macro switch “EMULATOR”** (“menu Project -> Setup Project -> C/C++ Compiler -> Category: Define Macro”)**,** the different interrupt vector table and the different header files are included accordingly.

A muster of the macro switch is:

```
#if (EMULATOR == 0)
#include "mb91465x.h"
#else
#include "mb91465x_emulator.h"
#endif
```

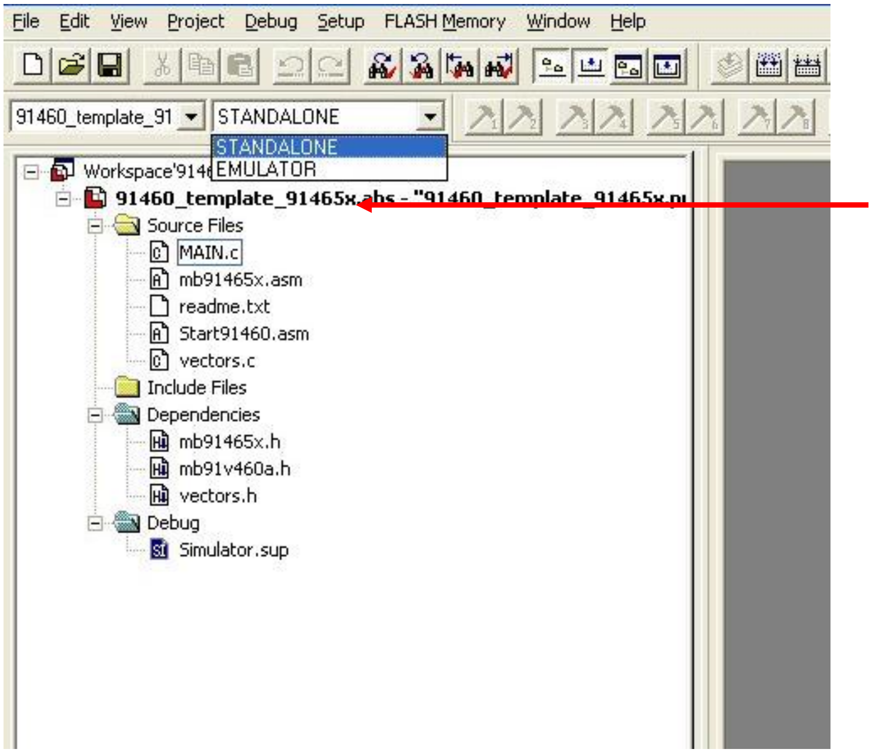
In the template project the files main.c and vector.c have already used this macro switch. Further source files of the project, which require the header file, should use the above mentioned macro switch.

The interrupt vector table and ICR registers in file vector.c are set according to the switch selection. The ICR registers define the interrupt level for the corresponding entry of the interrupt vector table (See file vector.c in Appendix).

4.2.1 Use Configuration EMULATOR

Select the configuration EMULATOR.

Figure 7. Project Configuration Selection



Select the configuration EMULATOR.

This configuration shall be used for the emulation system (MB2198-01 Emulator). The macro switch “EMULATOR” is set to “1”. The header-file and interrupt vector table for the emulation system are included.

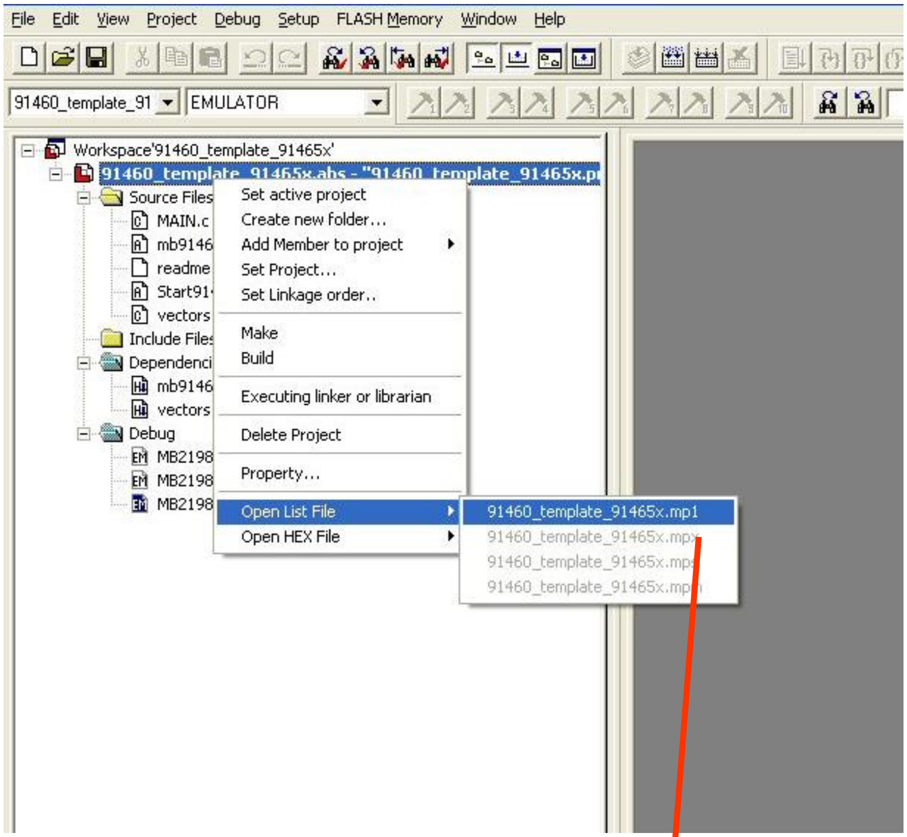
Check MCU settings in the module start91460.asm. Be careful regarding any modifications. In the module start91460.asm some basic MCU settings are already done and some initial data copy transfers are already performed for specific data sections e.g. INIT. Be aware of the maximum frequency of 80 MHz for emulation system. Set the PLL accordingly. Check module vectors.c if any interrupts are used for proper vector table set-up.

Write application code in the module main.c or add any other module to the project.

Write and modify source code and settings corresponding to the needs of the application. Finally use “**Make**”, or “**Build**” to compile and link the project. The generated output files can be found in the sub-folder EMULATOR of the project folder.

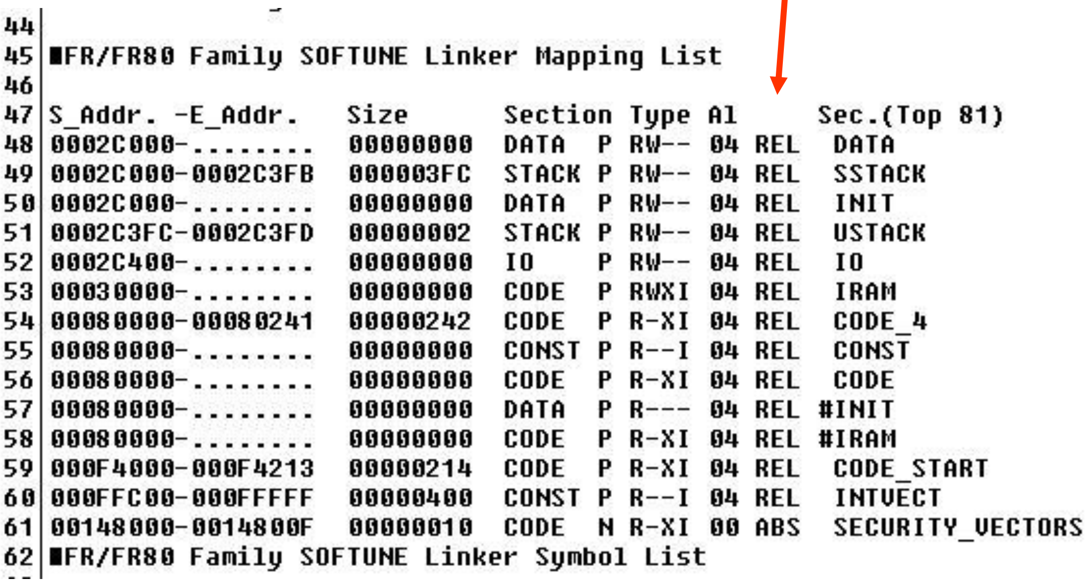
Note: Always check the memory map of the linker settings and ensure that this memory map is suitable for the application and target system in use. Additionally the *.mp1 file should be checked to ensure correct settings.

Figure 8. Open Softune Workbench Linker mapping file



Check linker mapping list in the*.mp1 file by right-click on “91460_template_91f465X.abs”, Open List File, *.mp1.

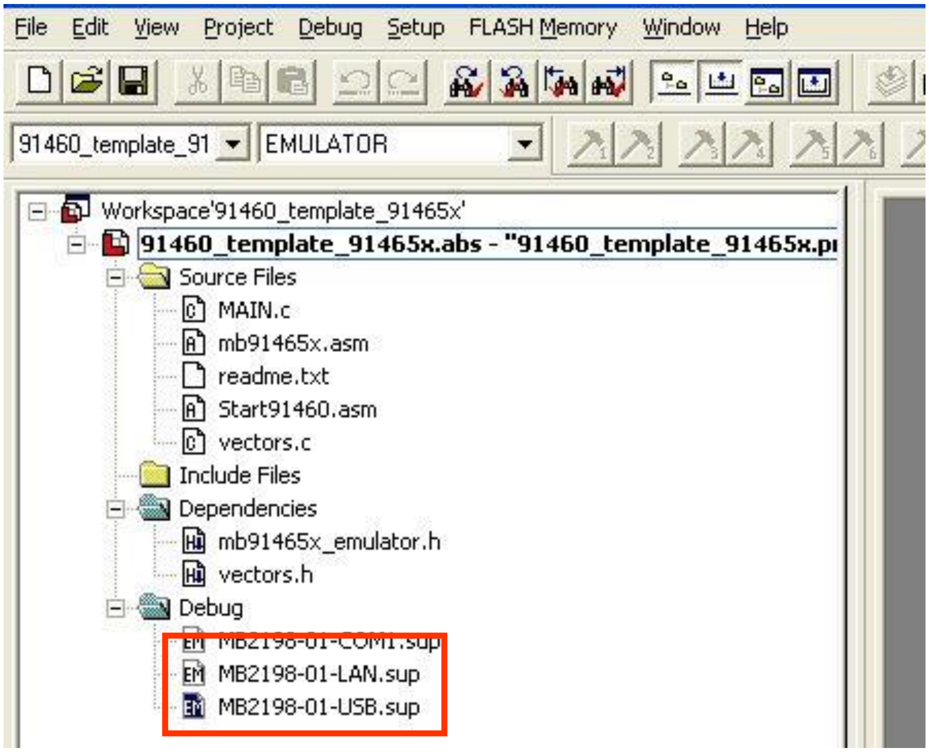
Figure 9. Softune Workbench Linker Mapping



```
■FR/FR80 Family SOFTUNE Linker Mapping List

S_Addr. -E_Addr.   Size      Section Type Al     Sec.(Top 81)
0002C000-........  00000000  DATA   P RW-- 04 REL  DATA
0002C000-0002C3FB  000003FC  STACK  P RW-- 04 REL  SSTACK
0002C000-........  00000000  DATA   P RW-- 04 REL  INIT
0002C3FC-0002C3FD  00000002  STACK  P RW-- 04 REL  USTACK
0002C400-........  00000000  IO     P RW-- 04 REL  IO
00030000-........  00000000  CODE   P RWXI 04 REL  IRAM
00080000-00080241  00000242  CODE   P R-XI 04 REL  CODE_4
00080000-........  00000000  CONST  P R--I 04 REL  CONST
00080000-........  00000000  CODE   P R-XI 04 REL  CODE
00080000-........  00000000  DATA   P R--- 04 REL #INIT
00080000-........  00000000  CODE   P R-XI 04 REL #IRAM
000F4000-000F4213  00000214  CODE   P R-XI 04 REL  CODE_START
000FFC00-000FFFFF  00000400  CONST  P R--I 04 REL  INTVECT
00148000-0014800F  00000010  CODE   N R-XI 00 ABS  SECURITY_VECTORS
■FR/FR80 Family SOFTUNE Linker Symbol List
```

Start Softune Emulator Debugger via the “**Debug**” Command

Figure 10. Start Softune Workbench Emulator Debugger



```
Workspace'91460_template_91465x'
  91460_template_91465x.abs - "91460_template_91465x.pı
    Source Files
      MAIN.c
      mb91465x.asm
      readme.txt
      Start91460.asm
      vectors.c
    Include Files
    Dependencies
      mb91465x_emulator.h
      vectors.h
    Debug
      MB2198-01-COM1.sup
      MB2198-01-LAN.sup
      MB2198-01-USB.sup
```

There are three possible connections between MB2198-01 Emulator and PC:

- COM1: MB2198-01-COM1.sup
- LAN: MB2198-01-LAN.sup
- USB: MB2198-01-USB.sup

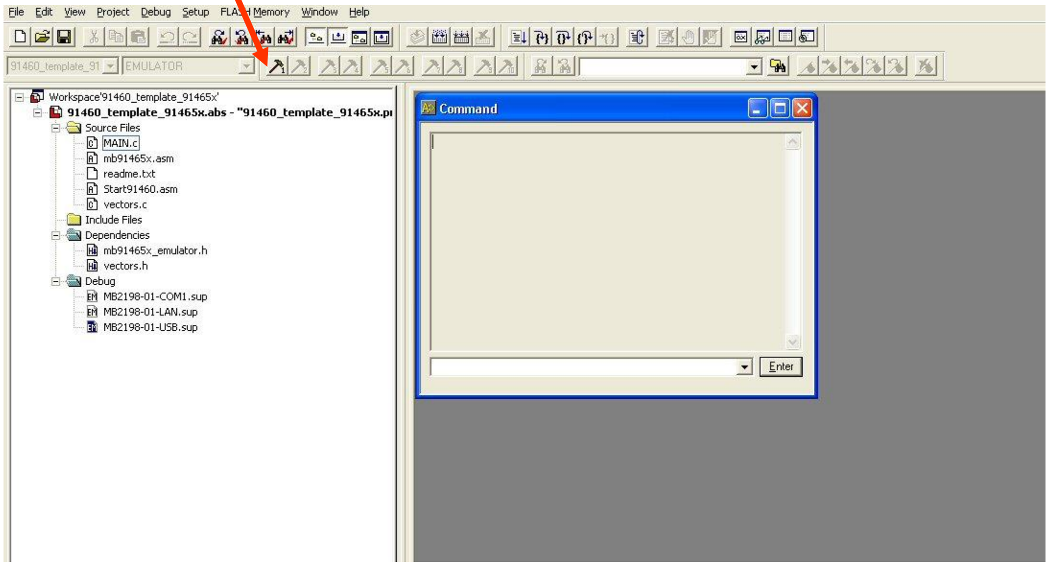
Select the suitable interface.

When starting the debug session the compiled binary file (\EMULATOR\ABS\*.abs) is automatically downloaded to the emulator. Afterwards the procedure file **set_cs.prc** in the PRC folder is executed. This file sets up and activates chip select area **CS3**. The FlexRay communication controller MB88121B is connected at CS3.

By asserting Reset during debugging the MB91V460A is reset. Also CS3 is getting inactive. To activate it again the customize button 1 can be used. By pressing the button the set_CS3.prc will be executed, which initialises CS3.

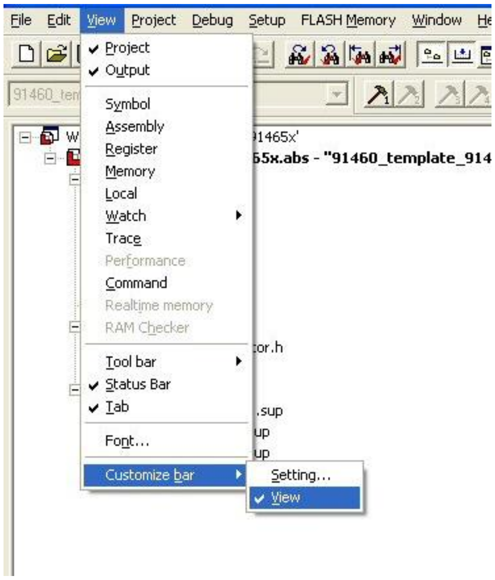
Click on customize button 1 to execute set_CS.prc after asserting Reset.

Figure 11. Emulator Debug Session, Use Customize Button 1



If the customize buttons are not visible, activate them via “View -> Customise bar -> View”

Figure 12. View Customize Bar



After the project is built and debugged successfully switch to the active configuration **STANDALONE**

4.2.2 Use configuration STANDALONE

Remove the emulation system and restore the Flash chip into the socket. The macro switch “Emulator” is set to “0”. The header-file and interrupt vector table for MB91F465XA series are included. Recompile the project.

With the Flash programming utility (e.g. Cypress Flashprogrammer) program the generated mhx file (Motorola S-Record, located in sub-folder \STANDALONE\ABS\*.mhx) into the internal Flash of MB91F465XA series.

A Appendix

A.1 MB88121B Register

A.1.1 MB88121B Register Map

Table 2. MB88121 Register Map

Address	Symbol	Name	Reset	Access
Customer Register				
0x0000	VER (CUST0, CIF0)	Version Information Register	0x043079FF	R
0x0004	CCNT (CUST1, CIF1)	Clock Control Register	0x0000.0000	R/W
0x0008	CUS2 (CIF2)	Customer 2 Register	0x0000.0000	R/W
0x000C	INT	Interrupt Register	0x0000.0000	R/W
Special Registers				
0x0010	-	Reserved (don’t write)	0x0000.0000	R/W
0x0014	-	Reserved (don’t write)	0x0000.0000	R/W
0x0018	-	Reserved	0x0000.0000	R/W
0x001C	LCK	Lock Register	0x0000.0000	R/W
Interrupt Register				
0x0020	EIR	Error Interrupt Register	0x0000.0000	R/W
0x024	SIR	Status Interrupt Register	0x0000.0000	R/W
0x0028	EILS	Error Interrupt Line Select	0x0000.0000	R/W
0x002C	SILS	Status Interrupt Line Select	0x0303.FFFF	R/W
0x0030	EIES	Error Interrupt Enable Set	0x0000.0000	R/W
0x0034	EIER	Error Interrupt Enable Reset	0x0000.0000	R/W
0x0038	SIES	Status Interrupt Enable Set	0x0000.0000	R/W
0x003C	SIER	Status Interrupt Enable Reset	0x0000.0000	R/W
0x0040	ILE	Interrupt Line Enable	0x0000.0000	R/W
0x0044	T0C	Timer 0 Configuration	0x0000.0000	R/W
0x0048	T1C	Timer 1 Configuration	0x0000.0000	R/W
0x004C	STPW1	Stop Watch Register 1	0x0000.0000	R/W
0x0050	STPW2	Stop Watch Register 2	0x0000.0000	R/W
0x0054 – 0x007C	-	Reserved	0x0000.0000	R
CC Control Register				
0x0080	SUCC1	SUC Configuration Register 1	0x0C40.1000	R/W
0x0084	SUCC2	SUC Configuration Register 2	0x0100.0504	R/W
0x0088	SUCC3	SUC Configuration Register 3	0x0000.0011	R/W
0x008C	NEMC	NEM Configuration Register	0x0000.0000	R/W
0x0090	PRTC1	PRT Configuration Register 1	0x084C.0633	R/W

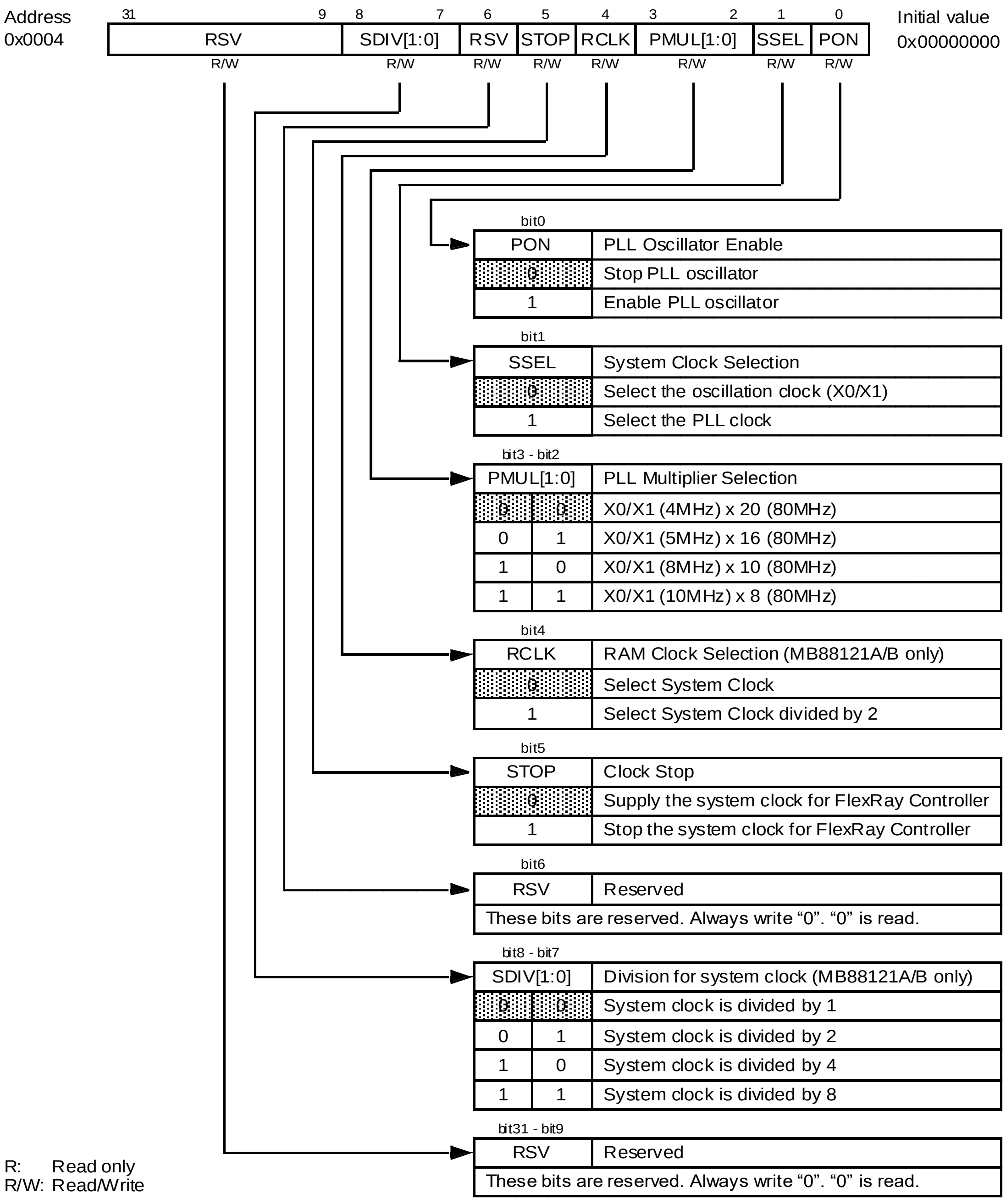
Address	Symbol	Name	Reset	Access
0x0094	PRTC2	PRT Configuration Register 2	0x0F2D.0A0E	R/W
0x0098	MHDC	MHD Configuration Register	0x0000.0000	R/W
0x009C	-	Reserved	0x0000.0000	R
0x00A0	GTUC1	GTU Configuration Register 1	0x0000.0280	R/W
0x00A4	GTUC2	GTU Configuration Register 2	0x0002.000A	R/W
0x00A8	GTUC3	GTU Configuration Register 3	0x0202.0000	R/W
0x00AC	GTUC4	GTU Configuration Register 4	0x0008.0007	R/W
0x00B0	GTUC5	GTU Configuration Register 5	0x0E00.0000	R/W
0x00B4	GTUC6	GTU Configuration Register 6	0x0002.0000	R/W
0x00B8	GTUC7	GTU Configuration Register 7	0x0002.0004	R/W
0x00BC	GTUC8	GTU Configuration Register 8	0x0000.0002	R/W
0x00C0	GTUC9	GTU Configuration Register 9	0x0000.0101	R/W
0x00C4	GTUC10	GTU Configuration Register 10	0x0002.0005	R/W
0x00C8	GTUC11	GTU Configuration Register 11	0x0000.0000	R/W
0x00CF – 0x00FC	-	Reserved	0x0000.0000	R
CC Status Register				
0x0100	CCSV	CC Status Register	0x0010.4000	R
0x0104	CCEV	CC Error Vector	0x0000.0000	R
0x0108 – 0x010C	-	Reserved	0x0000.0000	R
0x0110	SCV	Slot Counter Value	0x0000.0000	R
0x0114	MTCCV	Macrotick and Cycle Counter Value	0x0000.0000	R
0x0118	RCV	Rate Correction Value	0x0000.0000	R
0x011C	OCV	Offset Correction Value	0x0000.0000	R
0x0120	SFS	Sync Frame Status	0x0000.0000	R
0x0124	SWNIT	Symbol Window and NIT Status	0x0000.0000	R
0x0128	ACS	Aggregated Channel Status	0x0000.0000	R
0x012C	-	Reserved	0x0000.0000	R
0x0130 – 0x0168	ESIDn	Even Sync ID [1 … 15]	0x0000.0000	R
0x016C	-	Reserved	0x0000.0000	R
0x0170 – 0x01A8	OSIDn	Odd Sync ID [1 … 15]	0x0000.0000	R
0x01AC	-	Reserved	0x0000.0000	R
0x01B0 – 0x01B8	NMVn	Network Management Vector [1..3]	0x0000.0000	R
0x01BC – 0x02FC	-	Reserved	0x0000.0000	R
Message Buffer Control Register				

Address	Symbol	Name	Reset	Access
0x0300	MRC	Message RAM Configuration	0x0180.0000	R/W
0x0304	FRF	FIFO Rejection Filter	0x0180.0000	R/W
0x0308	FRFM	FIFO Rejection Filter Mask	0x0000.0000	R/W
0x030C	FCL	FIFO critical Level	0x0000.0080	R/W
Message Buffer Status Register				
0x0310	MHDS	Message Handler Status	0x0000.0000	R/W
0x0314	LDTS	Last Dynamic Transmit Slot	0x0000.0000	R/W
0x0318	FSR	FIFO Status Register	0x0000.0000	R/W
0x031C	MHDF	Message Handler Constraints Flags	0x0000.0000	R/W
0x0320	TXRQ1	Transmission Request 1	0x0000.0000	R
0x0324	TXRQ2	Transmission Request 2	0x0000.0000	R
0x0328	TXRQ3	Transmission Request 3	0x0000.0000	R
0x032C	TXRQ4	Transmission Request 4	0x0000.0000	R
0x0330	NDAT1	New Data 1	0x0000.0000	R
0x0334	NDAT2	New Data 2	0x0000.0000	R
0x0338	NDAT3	New Data 3	0x0000.0000	R
0x033C	NDAT4	New Data 4	0x0000.0000	R
0x0340	MBSC1	Message Buffer Status Changed 1	0x0000.0000	R
0x0344	MBSC2	Message Buffer Status Changed 2	0x0000.0000	R
0x0348	MBSC3	Message Buffer Status Changed 3	0x0000.0000	R
0x034C	MBSC4	Message Buffer Status Changed 4	0x0000.0000	R
0x0350 – 0x03EC	-	Reserved	0x0000.0000	R
Identification Registers				
0x03F0	CREL	Core Release Endian Register	0x0726.0412	R
0x03F4	ENDN	Endian Register	0x8765.4321	R
0x03F8 – 0x03FC	-	Reserved	0x0000.0000	R
Input Buffer				
0x0400 – 0x04FC	WRDSn	Write Data Setcion [1..64]	0x0000.0000	R/W
0x0500	WRHS1	Write Header Section 1	0x0000.0000	R/W
0x0504	WRHS2	Write Header Section 2	0x0000.0000	R/W
0x0508	WRHS3	Write Header Section 3	0x0000.0000	R/W
0x050C	-	Reserved	0x0000.0000	R/W
0x0510	IBCM	Input Buffer Command Mask	0x0000.0000	R/W
0x0514	IBCR	Input Buffer Command Request	0x0000.0000	R/W
0x0518 – 0x05FC	-	Reserved	0x0000.0000	R/W

Address	Symbol	Name	Reset	Access
Output Buffer				
0x0600 – 0x06FC	RDDSn	Read Data Section [1..64]	0x0000.0000	R
0x0700	RDHS1	Read Header Section 1	0x0000.0000	R
0x0704	RDHS2	Read Header Section 2	0x0000.0000	R
0x0708	RDHS3	Read Header Section 3	0x0000.0000	R
0x071C	MBS	Message Buffer Status	0x0000.0000	R
0x0710	OBCM	Output Buffer Command Register	0x0000.0000	R/W
0x0714	OBCR	Output Buffer Command Request	0x0000.0000	R/W
0x0718 – 0x07FC	-	Reserved	0x0000.0000	R

4.2.3 CIF1 (CCNT) Register / Clock Control Register

The CLOCK CONTROL Register is writeable in DEFAULT_CONFIG (CCSV[5:0] = 00 0000) or CONFIG state (CCSV[5:0] = 00 1111), only



Address 0x0004

31 … 9	8 … 7	6	5	4	3 … 2	1	0
RSV	SDIV[1:0]	RSV	STOP	RCLK	PMUL[1:0]	SSEL	PON
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Initial value 0x00000000

bit0

PON	PLL Oscillator Enable
0	Stop PLL oscillator
1	Enable PLL oscillator

bit1

SSEL	System Clock Selection
0	Select the oscillation clock (X0/X1)
1	Select the PLL clock

bit3 - bit2

PMUL[1:0]		PLL Multiplier Selection
0	0	X0/X1 (4MHz) x 20 (80MHz)
0	1	X0/X1 (5MHz) x 16 (80MHz)
1	0	X0/X1 (8MHz) x 10 (80MHz)
1	1	X0/X1 (10MHz) x 8 (80MHz)

bit4

RCLK	RAM Clock Selection (MB88121A/B only)
0	Select System Clock
1	Select System Clock divided by 2

bit5

STOP	Clock Stop
0	Supply the system clock for FlexRay Controller
1	Stop the system clock for FlexRay Controller

bit6

RSV	Reserved
These bits are reserved. Always write "0". "0" is read.	

bit8 - bit7

SDIV[1:0]		Division for system clock (MB88121A/B only)
0	0	System clock is divided by 1
0	1	System clock is divided by 2
1	0	System clock is divided by 4
1	1	System clock is divided by 8

bit31 - bit9

RSV	Reserved
These bits are reserved. Always write "0". "0" is read.	

R: Read only
R/W: Read/Write

Table 3. CCNT Register Bit Description

<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td>Bit31- Bit9</td><td>RSV: Reserved</td><td>These bits are reserved. “0” is read. Write “0”.</td></tr>
<tr><td>bit8 - bit7</td><td>SDIV[1:0]:
Division for system clock</td><td>These bits control the division for system clock. This function is supported in MB88121A and MB88121B. These bits are reserved in MB88121. In MB88121, “0” is read and write “0”.
<table>
<tr><td>SDIV[1]</td><td>SDIV[0]</td><td>Function</td></tr>
<tr><td>0</td><td>0</td><td>System clock is divided by 1</td></tr>
<tr><td>0</td><td>1</td><td>System clock is divided by 2</td></tr>
<tr><td>1</td><td>0</td><td>System clock is divided by 4</td></tr>
<tr><td>1</td><td>1</td><td>System clock is divided by 8</td></tr>
</table>
&lt;&lt;Note>>
When FlexRay controller can receive or transmit data, these bits must not be changed.</td></tr>
<tr><td>Bit 6</td><td>RSV: Reserved</td><td>This bit is reserved. Always write “0”.</td></tr>
<tr><td>Bit 5</td><td>STOP:
Clock Stop</td><td>This bit stops the system clock. If this bit set to “1”, the system clock is stopped. But the oscillator is active.
When this bit is set to “1”, please carry out the following procedures.
- PLL On
1) Stop receiving and transmitting for FlexRay controller.
2) Set “0” to SSEL bit.
3) Set “0” to PON bit.
4) Set “1” to STOP bit.
- PLL Off
1) Stop receiving and transmitting for FlexRay controller.
2) Set “1” to STOP bit.

When this bit is changed into “0” from “1”, please carry out in the following procedures.
- PLL On
1) Set “1” to PON bit.
2) Set “0” to STOP bit.
3) Set “1” to SSEL bit after PLL lock up time (600us).
4) Enable to receive and transmit data for FlexRay controller.
- PLL Off
1) Set “0” to STOP bit.

<<Note>>
When FlexRay controller can receive or transmit data, these bits must not be changed.</td></tr>
<tr><td>Bit 4</td><td>RCLK:
RAM Clock Selection</td><td>This bit selects the RAM clock in MB88121A, MB88121B.
If this bit is “0”, the system clock is selected as the RAM clock.</td></tr>
</table>

<table>
<tr><th>Bit</th><th>Name</th><th>Function</th></tr>
<tr><td></td><td></td><td>If this bit is “1”, the system clock divided by 2 is selected as the RAM clock.
<<Note>>
When FlexRay controller can receive or transmit data, these bits must not be changed.</td></tr>
<tr><td>Bit 3 – Bit 2</td><td>PMUL[1:0]:
PLL Multiplier Selection</td><td>These bits control the PLL multiplier. These bits must set up so that the PLL clock is set to 80MHz.
For MB88121B, the evaluation of the PLL performance is pending. For this reason, do not use other settings than PMUL[1:0] = “11”.
<table>
<tr><td>PMUL[1]</td><td>PMUL[0]</td><td>Function</td></tr>
<tr><td>0</td><td>0</td><td>X0/X1 (4MHz) x 20 (80MHz) (tbd)</td></tr>
<tr><td>0</td><td>1</td><td>X0/X1 (5MHz) x 16 (80MHz) (tbd)</td></tr>
<tr><td>1</td><td>0</td><td>X0/X1 (8MHz) x 10 (80MHz) (tbd)</td></tr>
<tr><td>1</td><td>1</td><td>X0/X1 (10MHz) x 8 (80MHz)</td></tr>
</table>
<<Note>>
These bits must be changed before PON bit is set to “1”.</td></tr>
<tr><td>Bit1</td><td>SSEL:
System Clock
Selection</td><td>This bit selects the system clock.
"0": Select the clock of X0/X1
"1": Select the clock of PLL
In MB88121 and MB88121A, the functionality of the PLL is not guaranteed.
<<Note>>
Must be changed into "1" from "0" after "1" is set as a PON bit and PLL lock-up time (600us) passes.
If the oscillator of PLL is stopped, PON bit is set to "0" after this bit is changed to "0".
When FlexRay controller can receive or transmit data, these bits must not be changed.</td></tr>
<tr><td>Bit0</td><td>PON:
PLL Oscillator Enable</td><td>This bit controls PLL oscillator.
"0": Stop PLL oscillator
"1": PLL oscillator enable
<<Note>>
This bit must be changed when SSEL bit is "0".</td></tr>
</table>

4.2.4 CIF2 (CUS2) Register

The Customer2 Register (CUS2) is a 32-bit register, at address 0x0008.

The upper 16 bit (B16..31) are called Debug support Register (DBGS).

The lower 16 bit (Bit 0..15) are called DMA support register (DMAS)

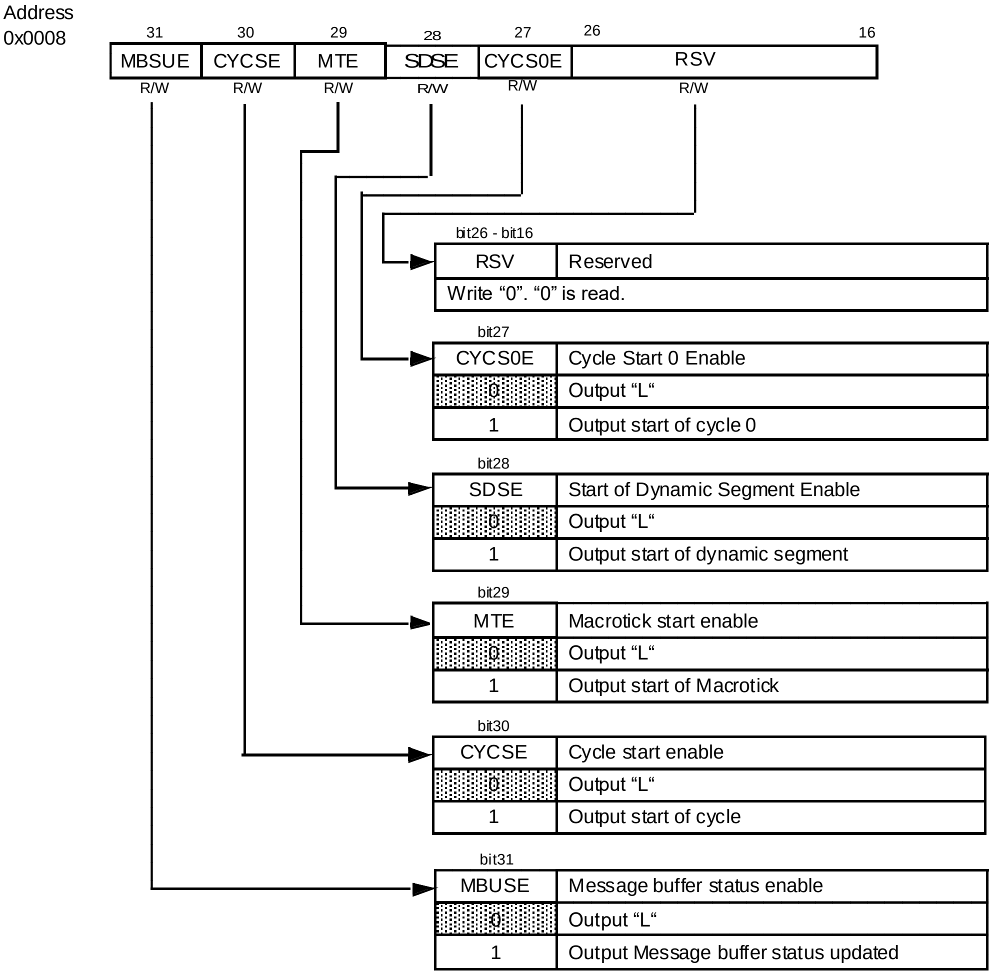
Always access the customer 2 register 32-bit wise.

Address	31 … 16	15 … 0
0x0008	DBGS	DMAS

■ Debug support Register (DBGS)

The Debug support register is available only in MB88121B. It is reserved in MB88121(A).

Bits 31 - 27 are available in MB88121B, only. For MB88121(A) they are reserved



Address
0x0008

31	30	29	28	27	26 - 16
MBSUE	CYCSE	MTE	SDSE	CYCS0E	RSV
R/W	R/W	R/W	R/W	R/W	R/W

bit26 - bit16

RSV	Reserved
Write “0”. “0” is read.	

bit27

CYCS0E	Cycle Start 0 Enable
0	Output “L“
1	Output start of cycle 0

bit28

SDSE	Start of Dynamic Segment Enable
0	Output “L“
1	Output start of dynamic segment

bit29

MTE	Macrotick start enable
0	Output “L“
1	Output start of Macrotick

bit30

CYCSE	Cycle start enable
0	Output “L“
1	Output start of cycle

bit31

MBUSE	Message buffer status enable
0	Output “L“
1	Output Message buffer status updated

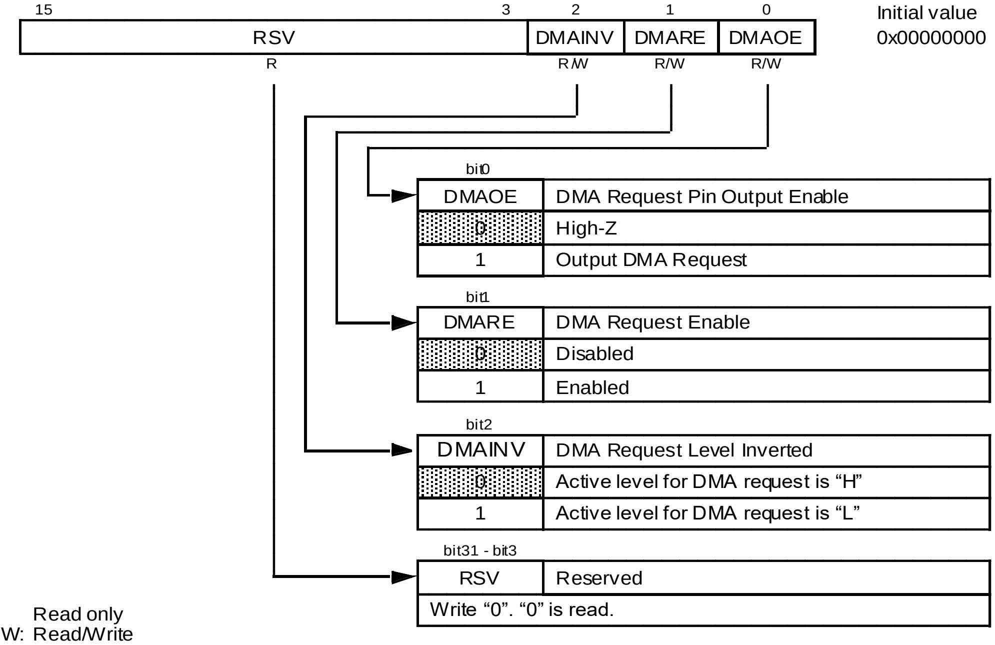
R: Read only
R/W: Read/Write

Table 4. DBGS Bit Description

Bit	Name	Function
Bit 31	MBSUE: Message buffer Status update enable	This bit controls the Message buffer status update ports: MBSU_RX1; MBSU_TX1, MBSU_TX2, MBSU_RX2 "0": Disabled "1": Enabled <<Note>> If enabled output “High” at every Message buffer status update. High duration: One RAM clock cycle. If “0“ is set outputs “L” at pin.
Bit 30	CYCSE: Cycle start output enable	This bit controls the Cycle start output "0": Disabled "1": Enabled <<Note>> If enabled output “High” at every cycle start. High duration: One RAM clock cycle If “0“ is set outputs “L” at pin.
Bit 29	MTE: Start of Macrotick output enable	This bit controls the Macrotick start output "0": Disabled "1": Enabled <<Note>> If enabled output “High” at every Macrotick start. High duration: One RAM clock cycle If “0“ is set outputs “L” at pin.
Bit 28	SDSE: Start of dynamic segment output enable	This bit controls the start of dynamic segment output "0": Disabled "1": Enabled <<Note>> If enabled Output “High” at every start of dynamic segment. High duration: One RAM clock cycle If “0“ is set outputs “L” at pin.
Bit 27	CYCS0E: Start of cycle 0 output enable	This bit controls the Cycle 0 start output "0": Disabled "1": Enabled <<Note>> If enabled output “High” at every cycle 0 start. High duration: One RAM clock cycle If “0“ is set outputs “L” at pin.
Bit 26 – Bit 16	RSV: Reserved	These bits are reserved. "0" is read. Write "0".

■ DMA SUPPORT REGISTER (DMAS)

The DMA support register is available only in MB88121A/B. It is reserved in MB88121.



bit0

DMAOE	DMA Request Pin Output Enable
0	High-Z
1	Output DMA Request

bit1

DMARE	DMA Request Enable
0	Disabled
1	Enabled

bit2

DMAINV	DMA Request Level Inverted
0	Active level for DMA request is “H”
1	Active level for DMA request is “L”

bit31 - bit3

RSV	Reserved
Write “0”. “0” is read.	

R: Read only
R/W: Read/Write

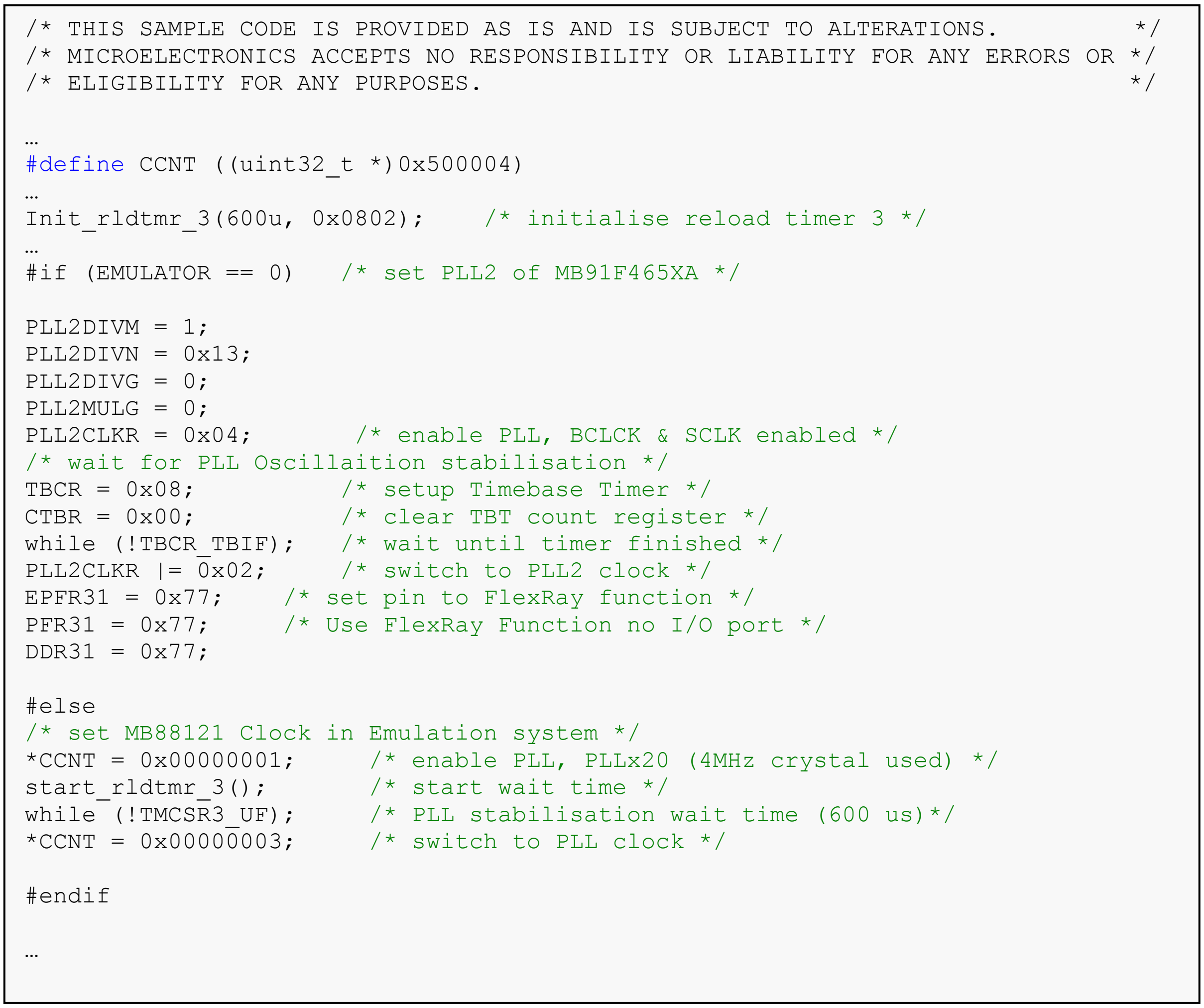
Table 5. DMAS Register Bit Description

Bit	Name	Function
Bit 15 – 3	RSV: Reserved	These bits are reserved. "0" is read. Write "0".
Bit 2	DMAINV: DMA Request Level Inverted	This bit controls the DMA request level. "0": Active level for DMA request is "H" "1": Active level for DMA request is "L" <<Note>> It is valid when DMAOE bit is "1".
Bit 1	DMARE: DMA Request enable	This bit controls the DMA request. "0": Disabled "1": Enabled <<Note>> It is valid when DMAOE bit is "1".
Bit 0	DMAOE: DMA Request Pin Output Enable	This bit controls output enable for DMA request pin. "0": High-Z at DMA_REQ pin "1": Output DMA request at DMA_REQ pin

A.2 Software part

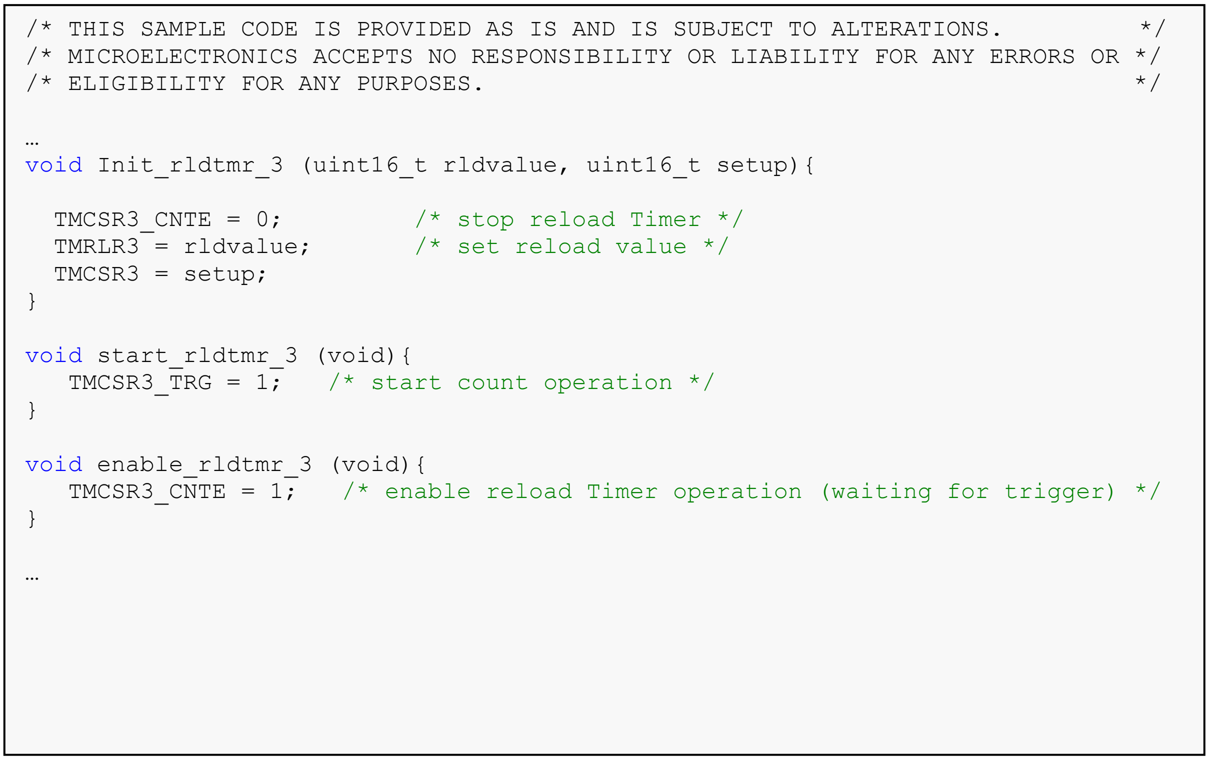
A.2.1 Activating FlexRay PLL

Figure 13. FlexRay Clock Settings



```
/* THIS SAMPLE CODE IS PROVIDED AS IS AND IS SUBJECT TO ALTERATIONS.          */
/* MICROELECTRONICS ACCEPTS NO RESPONSIBILITY OR LIABILITY FOR ANY ERRORS OR */
/* ELIGIBILITY FOR ANY PURPOSES.                                              */

…
#define CCNT ((uint32_t *)0x500004)
…
Init_rldtmr_3(600u, 0x0802);    /* initialise reload timer 3 */
…
#if (EMULATOR == 0)   /* set PLL2 of MB91F465XA */

PLL2DIVM = 1;
PLL2DIVN = 0x13;
PLL2DIVG = 0;
PLL2MULG = 0;
PLL2CLKR = 0x04;        /* enable PLL, BCLCK & SCLK enabled */
/* wait for PLL Oscillaition stabilisation */
TBCR = 0x08;          /* setup Timebase Timer */
CTBR = 0x00;          /* clear TBT count register */
while (!TBCR_TBIF);   /* wait until timer finished */
PLL2CLKR |= 0x02;     /* switch to PLL2 clock */
EPFR31 = 0x77;    /* set pin to FlexRay function */
PFR31 = 0x77;     /* Use FlexRay Function no I/O port */
DDR31 = 0x77;

#else
/* set MB88121 Clock in Emulation system */
*CCNT = 0x00000001;     /* enable PLL, PLLx20 (4MHz crystal used) */
start_rldtmr_3();       /* start wait time */
while (!TMCSR3_UF);     /* PLL stabilisation wait time (600 us)*/
*CCNT = 0x00000003;     /* switch to PLL clock */

#endif

…
```

Note: The Reload timer 3 has a 600us period. For this example the peripheral Clock is set to 16MHz.



```
/* THIS SAMPLE CODE IS PROVIDED AS IS AND IS SUBJECT TO ALTERATIONS.         */
/* MICROELECTRONICS ACCEPTS NO RESPONSIBILITY OR LIABILITY FOR ANY ERRORS OR */
/* ELIGIBILITY FOR ANY PURPOSES.                                             */

…
void Init_rldtmr_3 (uint16_t rldvalue, uint16_t setup){

  TMCSR3_CNTE = 0;          /* stop reload Timer */
  TMRLR3 = rldvalue;        /* set reload value */
  TMCSR3 = setup;
}

void start_rldtmr_3 (void){
   TMCSR3_TRG = 1;    /* start count operation */
}

void enable_rldtmr_3 (void){
   TMCSR3_CNTE = 1;    /* enable reload Timer operation (waiting for trigger) */
}

…
```

Figure 14. Reload Timer Functions

4.2.5 Vectors.c file

```
/* THIS SAMPLE CODE IS PROVIDED AS IS AND IS SUBJECT TO ALTERATIONS.         */
/* MICROELECTRONICS ACCEPTS NO RESPONSIBILITY OR LIABILITY FOR ANY ERRORS OR */
/* ELIGIBILITY FOR ANY PURPOSES.                                             */
/*-------------------------------------------------------------------------
  VECTORS.C
  - Interrupt level (priority) setting
  - Interrupt vector definition

  31.04.05  1.00   UMa    Initial Version
  08.11.05  1.01   MSt    SWB Mondeb switch for ICR00 Register added
  27.02.06  1.02   UMa    added comment in DefaultIRQHandler
  17.03.06  1.03   UMa    comment out ICR01
  28.07.06  1.04   UMa    changed comment
  06.10.06  1.05   UMa    changed DefaultIRQHandler
  19.03.07  1.06   MSt    Define setting for MB91F465XA emulation system added
-------------------------------------------------------------------------*/

#if (EMULATOR == 0)
#include "mb91465x.h"
#else
#include "mb91465x_emulator.h"
#endif

/*-------------------------------------------------------------------------
  InitIrqLevels()

  This function  pre-sets all interrupt control registers. It can be used
  to set all interrupt priorities in static applications. If this file
  contains assignments to dedicated resources, verify  that the
  appropriate controller is used. Not all devices of the MB91460 Series
  offer all recources.

  NOTE: value 31 disables the interrupt and value 16 sets highest priority.
-------------------------------------------------------------------------*/
void InitIrqLevels(void)
{
    /*  ICRxx */
    ICR00 = 31;    /* External Interrupt 0         */
                   /* External Interrupt 1         */
    ICR01 = 31;    /* External Interrupt 2         */
                   /* External Interrupt 3         */
…

    ICR63 = 31;    /* DMA Controller               */
                   /* Main/Sub OSC stability wait  */
#if (EMULATOR == 1)
    ICR05 = 31;    /* External Interrupt 10        */
                   /* External Interrupt 11  -> FlexRay Timer 0 & timer 1 */
    ICR06 = 31;    /* External Interrupt 12        */
                   /* External Interrupt 13  -> FlexRay INT1 */
    ICR07 = 31;    /* External Interrupt 14        */
                   /* External Interrupt 15  -> FlexRay INT0 */
    ICR34 = 31;    /* USART (LIN, FIFO) 12 RX      */
                   /* USART (LIN, FIFO) 12 TX      */
    ICR35 = 31;    /* USART (LIN, FIFO) 13 RX      */
                   /* USART (LIN, FIFO) 13 TX      */
#endif
}
```

Figure 15. Vector.c File Macro Switch Settings

```
…

/*------------------------------------------------------------------------
   Vector definiton

   Use following statements to define vectors. All resource related
   vectors are predefined. Remaining software interrupts can be added here
   as well.
------------------------------------------------------------------------*/
#pragma intvect 0xBFF8            0     /* (fixed) reset vector         */
#pragma intvect 0x06000000        1     /* (fixed) Mode Byte            */

#pragma intvect DefaultIRQHandler 15    /* Non Maskable Interrupt       */
#pragma intvect DefaultIRQHandler 16    /* External Interrupt 0         */
…
#if (EMULATOR == 0)
#pragma intvect DefaultIRQHandler 27    /* External Interrupt 11        */
#pragma intvect DefaultIRQHandler 29    /* External Interrupt 13        */
#pragma intvect DefaultIRQHandler 31    /* External Interrupt 15        */
#else
#pragma intvect DefaultIRQHandler 27    /* External Interrupt 11
       -> FlexRay Timer 0 & timer 1 */
#pragma intvect DefaultIRQHandler 29    /* External Interrupt 13
 -> FlexRay INT1 */
#pragma intvect DefaultIRQHandler 31    /* External Interrupt 15
       -> FlexRay INT0 */
#endif
…
#if (EMULATOR == 0)
#pragma intvect DefaultIRQHandler 84    /* FlexRay INT0 line            */
#pragma intvect DefaultIRQHandler 85    /* FlexRay Timer 0              */
#pragma intvect DefaultIRQHandler 86    /* FlexRay INT1 line            */
#pragma intvect DefaultIRQHandler 87    /* FlexRay Timer 1              */
#else
#pragma intvect DefaultIRQHandler 84    /* USART (LIN, FIFO) 12 RX      */
#pragma intvect DefaultIRQHandler 85    /* USART (LIN, FIFO) 12 TX      */
#pragma intvect DefaultIRQHandler 86    /* USART (LIN, FIFO) 13 RX      */
#pragma intvect DefaultIRQHandler 87    /* USART (LIN, FIFO) 13 TX      */
#endif
…
#pragma intvect DefaultIRQHandler 142   /* DMA Controller               */
#pragma intvect DefaultIRQHandler 143   /* Main/Sub OSC stability wait  */
#pragma intvect 0xFFFFFFFF        144   /* Boot Sec. Vector (MB91V460A) */
```

The above code part of Vectors.c file shows the settings using the macro switch EMULATOR.

Note: Always check the Internet / CD for latest versions.

A.3 Related Documentation

Appnotes:

- AN205222 - FR Family MB2198-01 Emulator System Getting Started Guide
- AN204828 - F2MC-16FX Family, Emulating and Debugging with Softune and MB2198-01
- AN205146 - FR Family, MB91460 Emulation System

Documentation:

- MB91460 Super Series Hardware Manual
- MB91F465X Series Datasheet
- MB88121 Series Datasheet
- EMA-MB91V460A-00x User Guide
- EMA-MB91V460A-100 (FlexRay extension board) User Guide
- EMA-MB91F465X-NLS-100M20 (socket adapter board) User Guide

5 Document History

Document Title: AN205155 - FR Family, MB91F465XA Emulation

Document Number: 002-05155

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	-	NOFL	04/24/2007	Initial release
			06/26/2007	Further description added, typos corrected and Figure 2-5 added.
*A	5083734	NOFL	01/14/2016	Migrated Spansion Application Note from MCU-AN-300015-E-V11 to Cypress format.
*B	5868093	AESATMP9	08/30/2017	Updated logo and copyright.
*C	6052824	NOFL	02/02/2018	Updated links. Updated Sales page.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at Cypress Locations.

Products

Arm® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

PSoC® Solutions

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP | PSoC 6 MCU

Cypress Developer Community

Community | Projects | Videos | Blogs | Training | Components

Technical Support

cypress.com/support

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. No computing device can be absolutely secure. Therefore, despite security measures implemented in Cypress hardware or software products, Cypress does not assume any liability arising out of any security breach, such as unauthorized access to or use of a Cypress product. In addition, the products described in these materials may contain design defects or errors known as errata which may cause the product to deviate from published specifications. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.