XMC4700 CPU Bus-Fault

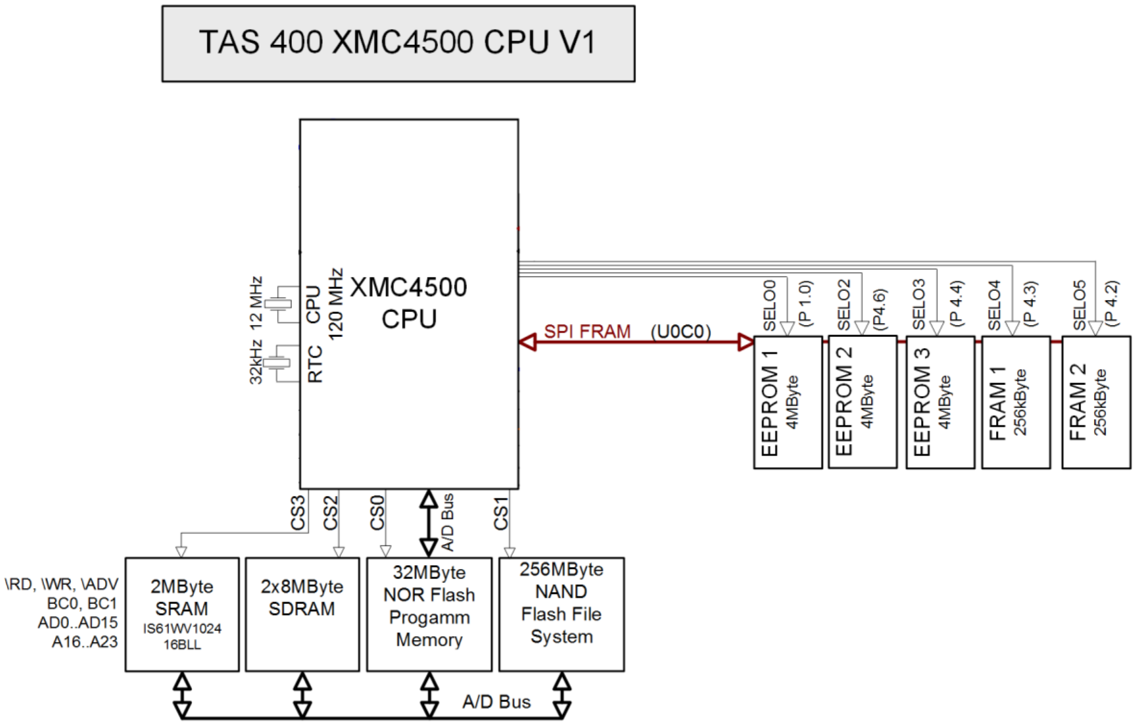
during DMA-Transfer

Mit Sicherheit verbunden.

Inhaltsverzeichnis

1. PROBLEM DESCRIPTION

Our hardware consists of the XMC4x00 CPU and several external components including non-volatile flash memory connected over SPI-Bus, NOR-flash, NAND-flash and external RAM connected over EBU:



RSE-Hardware with XMC4500 / XMC4700 (Simplified Block diagram)

During the migration from the XMC4500 (Step AC) to XMC4700 CPU (Step AA) we have observed rare CPU Bus Faults. After checking CPU-Clocks, EBU-Timings and possible hardware differences, it became clear that the only difference was the CPU.

We managed to narrow the faults down to a sample project that can be compiled for both CPU types. On XMC4500 the code runs more than 12h without problems, on XMC4700 it produces a Bus Fault after about 1 to 60 minutes.

2. SAMPLE PROJECT

The project was created using Keil µVision 5, armcc v5.06u1 and adapted XMC4x00 startup files.
The project contains 2 targets 'XMC4500' and 'XMC4700'.
When switching from one target to the other, the memory map changes ('SDRAM1_RSE.sct' / 'SDRAM1_RSE_4700.sct') and the startup files (startup_XMC4x00.s, system_XMC4x00.c) change.

The CPU runs at 120 MHz.

The project uses the following Modules:

- SPI over USIC0
- DMA
- CCU4

The main() function initializes the hardware and starts 2 CCU-based timers that will produce interrupts in 1 ms and 0.5 ms intervals. Each interrupt just increases a global counter variable.

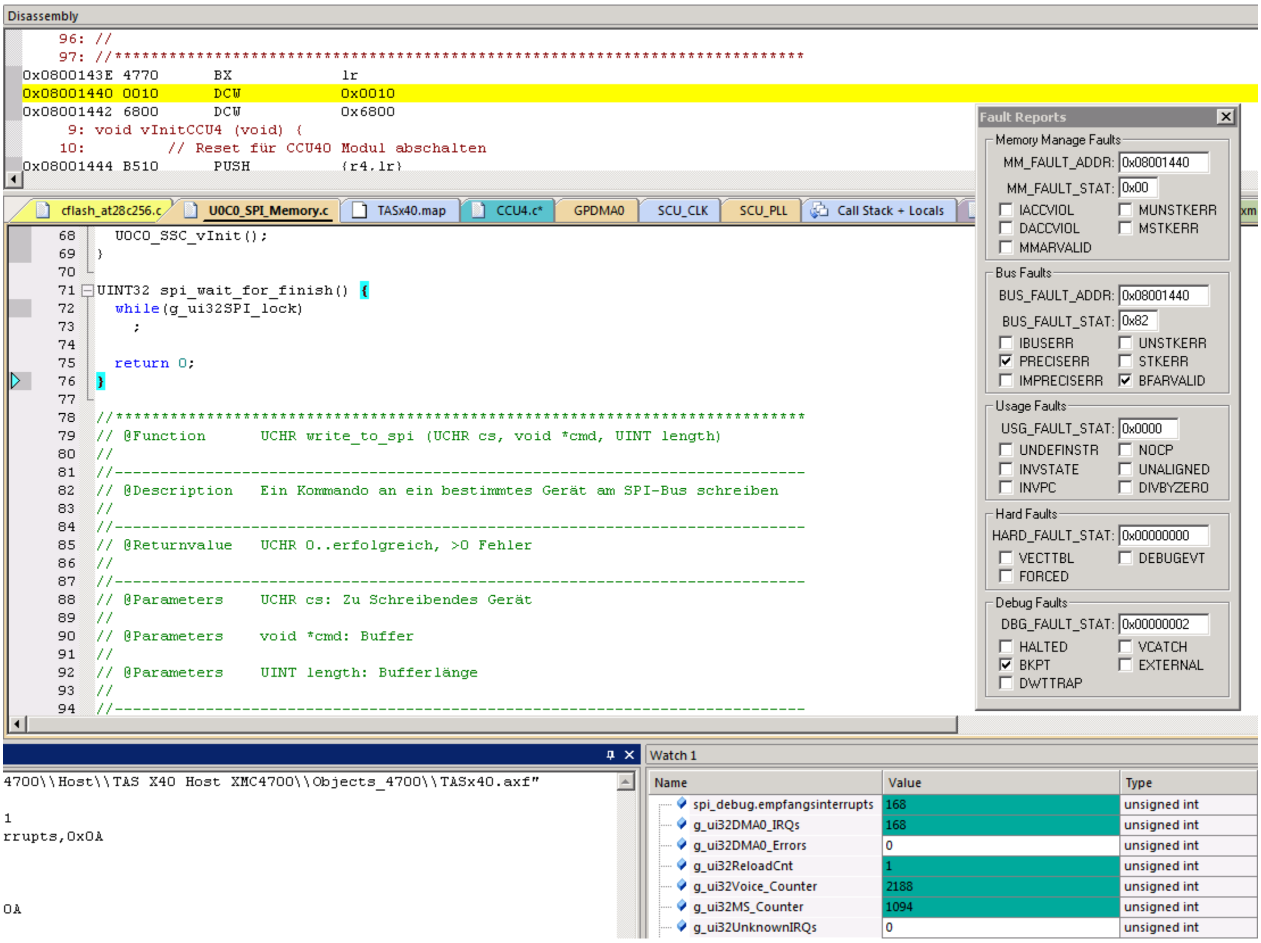
The main while-loop repeatedly loads data from a flash memory connected via SPI-Bus to the same memory range in external SRAM (0x681E0000 ... 0x681E0000 + 0x20000 Byte which are read in 32 blocks á 0x1000 Byte) using DMA-transfers.

The CPU acts as master on the SPI-Bus. 2 DMA-Channels are needed for each read/write combination to the flash memory on the SPI-Bus:
GPDMA0_CH4 .. SPI SIMO,
GPDMA0_CH2 .. SPI SOMI

The load process consists of the following steps:

- for-loop with 32 iterations:
  - setup 2 DMA-Channels
  - 1-byte dummy transfer, if USIC-Module Transfer shift indication flag is set
  - send 4-byte command sequence to flash memory,
  - receive requested 0x1000 Byte from flash memory in 2 DMA-Transfers:
    - First with 4000 Bytes and
    - Second with 96 Bytes

After several successful DMA transfers, the XMC4700 CPU throws a Bus Fault:



Disassembly

```
    96: //
    97: //***************************************************************************
0x0800143E 4770      BX          lr
0x08001440 0010      DCW         0x0010
0x08001442 6800      DCW         0x6800
     9: void vInitCCU4 (void) {
    10:         // Reset für CCU40 Modul abschalten
0x08001444 B510      PUSH        {r4,lr}
```

```
  U0C0_SSC_vInit();
}

UINT32 spi_wait_for_finish() {
  while(g_ui32SPI_lock)
    ;

  return 0;
}

//***************************************************************************
// @Function      UCHR write_to_spi (UCHR cs, void *cmd, UINT length)
//
//---------------------------------------------------------------------------
// @Description   Ein Kommando an ein bestimmtes Gerät am SPI-Bus schreiben
//
//---------------------------------------------------------------------------
// @Returnvalue   UCHR 0..erfolgreich, >0 Fehler
//
//---------------------------------------------------------------------------
// @Parameters    UCHR cs: Zu Schreibendes Gerät
//
// @Parameters    void *cmd: Buffer
//
// @Parameters    UINT length: Bufferlänge
//
//---------------------------------------------------------------------------
```

Fault Reports

Memory Manage Faults

MM_FAULT_ADDR: 0x08001440

MM_FAULT_STAT: 0x00

□ IACCVIOL	□ MUNSTKERR
□ DACCVIOL	□ MSTKERR
□ MMARVALID	

Bus Faults

BUS_FAULT_ADDR: 0x08001440

BUS_FAULT_STAT: 0x82

□ IBUSERR	□ UNSTKERR
☑ PRECISERR	□ STKERR
□ IMPRECISERR	☑ BFARVALID

Usage Faults

USG_FAULT_STAT: 0x0000

□ UNDEFINSTR	□ NOCP
□ INVSTATE	□ UNALIGNED
□ INVPC	□ DIVBYZERO

Hard Faults

HARD_FAULT_STAT: 0x00000000

□ VECTTBL	□ DEBUGEVT
□ FORCED	

Debug Faults

DBG_FAULT_STAT: 0x00000002

□ HALTED	□ VCATCH
☑ BKPT	□ EXTERNAL
□ DWTTRAP	

```
4700\\Host\\TAS X40 Host XMC4700\\Objects_4700\\TASx40.axf"

1
rrupts,0x0A



0A
```

Watch 1

Name	Value	Type
spi_debug.empfangsinterrupts	168	unsigned int
g_ui32DMA0_IRQs	168	unsigned int
g_ui32DMA0_Errors	0	unsigned int
g_ui32ReloadCnt	1	unsigned int
g_ui32Voice_Counter	2188	unsigned int
g_ui32MS_Counter	1094	unsigned int
g_ui32UnknownIRQs	0	unsigned int

Sample Bus Fault

The same project runs without problems on the XMC4500 hardware .