

UCD3xxx 数字电源控制器调试工具:内存调试器(Memory Debugger)的详细介绍

Sundy Xu, Neil Li

China Telecom Application Team

摘要

边界扫描(JTAG)是一种常用的调试方法,大多数 MCU 都支持 JTAG 调试,比如 TI 数字信号处理器(DSP)、数字信号控制器(DSC)、以及基于 ARM 架构的处理器等。

JTAG 在电源调试应用中会有很大风险,因为 JTAG 常常需要中止程序的正常运行去查看寄存器和 变量。这里推荐的 UCD3xxx 主要调试方法是基于通用异步收发器(UART)和电源管理总线 (PMBUS)。它们都支持在线调试功能。

UART 调试方法比较简单,本文不做论述,可以参考 UCD3xxx 示例代码。对于 PMBUS 调试方法,使用的主要工具就是内存调试器(Memory Debugger)。所以,本文主要详细介绍 Memory Debugger 的使用方法。

目录

| 2 2 |
|--------|
| 2 |
| |
| 4 |
| 4 |
| 6 |
| 7 |
| 7 |
| 7 |
| 8 |
| 9 |
| • • • |

图

| 图 1. Memory Debugger 界面 | 2 |
|---|---|
| 图 2. Memory Debugger 在 Fusion Design Online 的位置 | 3 |
| 图 3. Memory Debugger 在 UCD3xxx & UCD9xxx Device GUI 的位置 | 3 |
| 图 4. 预编译(.PP)文件和地址映射(.MAP)文件的产生 | 4 |
| 图 5. 地址映像(.MAP)文件 | 5 |
| 图 6. Memory Debugger 变量显示 | 5 |
| 图 7. 选择.PP 和.MAP 文件 | 6 |
| 图 8. 监控列表(Watch List)视图 | 7 |
| 图 9. DPWM 输出最大占空比修改前 | 8 |
| 图 10. DPWM 输出最大占空比修改后 | 8 |
| 图 11. DPWM 输出最大占空比修改前后的波形 | 9 |
| | |



1 引言

1.1 UCD3xxx 简介

UCD3xxx 系列芯片是 TI 公司开发的专门面向数字电源应用的控制器,主要应用在通讯电源、服务器电源、无线功放电源以及标准砖模块电源等领域。UCD30xx 系列产品主要是 UCD3028、UCD3020 以及 UCD3040。最新一代产品是 UCD31xx 系列,主要产品是 UCD3138、UCD3138064 以及 UCD3138128 等。

| 🜵 Fusion Memory Debugger - DC-DC | HSFB Firmware v0.0.2.1 @ | Address 88d | | | | | | | | _ | |
|---|--------------------------|----------------------------------|---------------|-----------------|------------|--------------|----------|---------------|-----------|------------------|--------|
| File | | | | | | | | | | | - |
| Change Map | | | | | | | | | | | |
| Close | | | | | - | - | | - | - | | |
| Close | S | ubstring Match 🖂 | Filter | Clear Filter S | how: 💿 All | O Registers | () RAM | () DFlash | () PFlas | h 🎠 To | ggle |
| Start Polling Polling Interval: | 500 💮 ms Global Poll Cl | nange: <u>Check All</u> <u>U</u> | ncheck All | | | | | | | | |
| Selection: All () > Dpwm0Regs () | | | | | | | | | | | |
| Name Description | Type | Value | Hex | Address | Size | Category | + - | Refresh | Write | AutoWr | |
| + AdcReas IRO Index Off | set V Struct ADC REGS | | | 0x00040000 | 152 bytes | Register | | | | | T |
| Eine B | ase Struct CIM_REGS | | | 0xFFFFFF20 | 24 bytes | Register | | | | | |
| DecRegs DPWM Individu | al R Struct DEC_REGS | | | 0xFFFFFE00 | 156 bytes | Register | | | | | - |
| 🕀 📌 Dpwm0Regs | Struct DPWM_REGS | | | 0x000D0000 | 140 bytes | Register | | REFRESH | WRITE | | 1 |
| 🕀 🔆 Dpwm 1Regs | Struct DPWM_REGS | | | 0x000A0000 | 140 bytes | Register | | | | | |
| 🗄 🕁 Dpwm2Regs | Struct DPWM_REGS | | | 0x00070000 | 140 bytes | Register | | | | | |
| 🖶 🔆 Dpwm3Regs 🛛 Analog Compar | rator Struct DPWM_REGS | | | 0x00050000 | 140 bytes | Register | | | | | |
| FaultMuxRegs Ramp Control F | Regi Struct FAULT_MUX | | | 0x00030000 | 128 bytes | Register | | | | | |
| FeCtrl0Regs | Struct FE_CTRL_R | | | 0x000E0000 | 68 bytes | Register | | | | | |
| 🕂 🛨 FeCtrl 1Regs | Struct FE_CTRL_R | | | 0x000B0000 | 68 bytes | Register | | | | | |
| 🗄 👷 FeCtrl2Regs 🛛 Filter Status Re | egister Struct FE_CTRL_R | | | 0x00080000 | 68 bytes | Register | | | | | |
| 🗄 🛧 Filter0Regs | Struct FILTER_REGS | | | 0x000C0000 | 100 bytes | Register | | | | | |
| 🕀 🍿 Filter 1 Regs | Struct FILTER_REGS | | | 0x00090000 | 100 bytes | Register | | | | | - |
| + Filter2Regs Fault Port I/O | Direc Struct FILTER_REGS | | | 0x00060000 | 100 bytes | Register | | | | | - |
| GioRegs Front End Con | trol Struct GIO_REGS | | | 0xFFF7FA00 | 64 bytes | Register | | | | | - |
| LoopMuxRegs Clock Trim Reg | ister Struct LOOP_MUX | | | 0x00020000 | 120 bytes | Register | | | | | \sim |
| Expand All Collapse All | Refresh All | Write Changes | Import | Export Se | lected | Export All | C | lear Watch Li | st | | |
| | | | | | | | | | | | |
| Timestamp Message | | | | | | | | 9 | Copy Log | <u>Clear Log</u> | 1 |
| 09:27:02.313 0x000D0084: read 4 byte | (s) 0x0000000 | | | | | | | | | | Ð |
| 09:27:02.313 0x000D0088: read 4 byte | (s) 0x0000000 | | | | | | | | | | ~ |
| Fusion Digital Power Designer v1.8.329 [/ | 2013-05-29] DC-DC HSFB | Firmware v0.0.2.1 | @ Address 880 | d Map Folder: U | CD3138HSF | BEVM_029 USE | B Adapte | r v1.0.10 [PE | C; 400 kH | z] | |

图1. Memory Debugger 界面

1.2 PMBUS 简介

PMBUS(电源管理总线)是一种开放标准的数字电源管理协议。可通过定义传输和物理接口以及命令语言来促进与电源转换器或其他设备的通信。该协议是由一群认为由于没有合适的标准而抑制了全数字电源管理解决方案发展的电源和半导体生产商共同建立的。目前使用标准是 PMBUS 1.2, PMBUS 1.3 正在制定中。

PMBUS 总共可以有 256 条命令,主要有几种:标准命令,是固定的不可改动的,如读取输出电压 命令是 0x8B,那么 0x8B 就不能用做其它命令;保留命令,留作以后扩展使用;用户自定义命令,用户 可以根据自己的应用来定义这些命令。具体命列表可以参考文献 1。



| 🚸 Fusion Digital | Power Designer - DC-DC HSFB @ Address 88d - | Page 0x0 - Texas Instruments | |
|--|---|---|--|
| File Device | Fools Debug Help | | DC-DC HSFB @ 88d - Page 0x0 |
| Monitor | Device/Project Configuration Compare | Vin - Input Voltage | Vout - Output Voltage |
| Show/Hide Plots: | Voltage Switching Tool | 75.000 ÷ V 33.000 ÷ V | 14.000 (m) V 12.000 (m) V |
| Vin V | Debug Console | | 3 000 A V Write |
| | Data Logging | | 16 00 |
| All Temp | PMBus Logging | | 14.00 |
| | Memory Debugger | 80.00 | 12.00 |
| Fit All Plots on | Memory Peek/Poke/Dump | 60.00 | 10.00 |
| Scale Plots to Width | SMBus & SAA Tool | 40.00 | 6.00 8.004 V |
| Height: 2 | Numeric Encode/Decode Tester | 20.00 | 4.00 |
| | Device Read/Write Stress Tester | 0.000 V | 0.00 |
| Limit Editors | Group Command Protocol Tester | 13:00 13:20 13:40 14:00 | 13:00 13:20 13:40 14:00 |
| Show Value La | Configuration Import Tester | Tout - Output Current | Temperature Internal |
| | ASCII Tool | 38.00 (A) Write | 50 A °C Write |
| Polling Rate: 5 (msec) | EEPROM File Tool | 40.00 | 60 |
| | EEPROM File Compare Tool | | |
| Stop Pollin | Isolated GUI Bit Mask Generator | 30.00 | 40 |
| Launch Dashb | Decimal & Mantissa Exponent Tool | | 20 |
| | PEC & SMBus -> 12C Translation Tool | 20.00 | |
| 🧄 Configure | Clear Configuration | 10.00 | 0 |
| JA Design | Download Firmware | 10.00 | -20 |
| w besign | Multi Image | 0.00 A | -40 |
| 🚸 Monitor | Download USB Adapter Firmware | 13:00 13:20 13:40 14:00 | 13:00 13:20 13:40 14:00 |
| 🚸 Status | 1 New PMBus Log Messages Show PMB | us Log 🗸 🗸 Unique open/a | dose settings for Configure, Design, Monitor, and Status 📳 |
| Fusion Digital Pow | ver Designer v1.8.329 [2013-05-29] DC-DC HSFB F | irmware v0.0.2.1 @ Address 88d USB Adapter v1.0.10 [P | EC; 400 kHz] VTEXAS INSTRUMENTS fusion digital power |

图2. Memory Debugger 在 Fusion Design Online 的位置

| atus | Tools |
|---|--|
| ttached: UCD3100ISO1 @ Address 88d ast ROM Found: IC Info: UCD31xx 1p0 ROM Info: ROM v3 IC v2 Package ID: 40-pin ast Program Found: Address: 88d 0x58 DEVICE_ID: UCD3100ISO1[0.0.02.00001 130215 MFR_MODEL: UCD3138HSFBEVM-029 MFR_REVISION: A | Scan Device in ROM Mode Custom: None Scan for Device in Program Mode: DEVICE ID DEVICE CODE PMBUS REVISION ✓ When a device is found, dump additional PMBus commands ✓ ✓ Command ROM to execute its program (SendByte 0xF0 to Address 11) Command Program to jump to ROM (SendByte 0xD9 to Address 88) Multi-image Firmware Download SMBus Debuq USB Adapter (SAA) Settings Memory Peek/Poke ROM API Erase/Set DFlash: 0xFF 0xAA Pflash checksum (includes boot) Dump Calculate Recreate Validate Clear Boot flash 2kB checksum Boot size: 2 ⊕ k8 Dump Calculate Recreate Validate Clear Export Flash Compare Flash Files Dump Flash File Elash Test Tool Full Export Tool X0 to Hex Tool |
| g Imestamp Message Copy Log Clear Log | Device Debug Tool Mantissa/Exponent Tool Iso Bitmask Tool PEC & SMBus -> 12C Translation Tool |

图3. Memory Debugger 在 UCD3xxx & UCD9xxx Device GUI 的位置



2 怎么找到 Memory Debugger

Memory Debugger 是德州仪器数字电源设计调试工具 Fusion Digital Power Designer(下载地址: http://www.ti.com/fusiongui,请下载最新版本,早期版本不支持 Memory Debugger 功能)的一部分,主 要用来读写 UCD3xxx 内部全局变量和寄存器的值。Memory Debugger 界面如图 1 所示。如果是第一次 调用 Memory Debugger,需要输入密码 "forestln"。

那么如何调出 Memory Debugger 工具,有两条途径:一是执行 Fusion Design Online 可执行文件 (桌面直接双击 Fusion Design Online 快捷方式,如果没有创建桌面快捷方式,在开始菜单中找到 Texas Instruments Fusion Digital Power Designer 目录下的 Fusion Digital Power Designer,点击即 可),然后就可以在 Fusion Design Online 上的 Tools 菜单下面找到 Memory Debugger 项,如图 2 所 示;二是执行 UCD3xxx & UCD9xxx Device GUI 可执行文件(桌面直接双击 UCD3xxx & UCD9xxx Device GUI 快捷方式,如果没有创建桌面快捷方式,在开始菜单中找到 Texas Instruments Fusion Digital Power Designer\Device GUIs 目录下的 UCD3xxx & UCD9xxx Device GUI,点击即可),Memory Debugger 在图 3 红圈处。

3 如何产生 Memory Debugger 需要的.PP 和.MAP 文件

在详细介绍 Memory Debugger 之前,需要介绍 Memory Debugger 所需要的两种文件,预编译 (.PP)文件和地址映像(.MAP)文件。在 CCS3.x 中,进入 Project\Build Options 选项,按图 4 红框 中所示配置就可以生成所需要的文件。PP 文件主要保存预编译生成的文件,关于预编译的详细内容,可 以参考文献 2。MAP 文件主要是把变量和它的地址都对应起来,如图 5 所示。MAP 文件的其他功能也可 以参考文献 2。

| Build Options for HSFB.pjt (Debug) | Build Options for HSFB.pjt (Debug) |
|---|--|
| General Compiler Link Order -g k. ppa -ppc -al -as -o2 -ea .asm -fr"\$(Proj_dir)\Debug" - fs"\$(Proj_dir)\Debug" -i"/header files" -d"_DEBUG" -mt -mv4abi=tiabi endian=big | General Compiler Linker Link Order |
| Category: Preprocessor Basic Advanced Advanced Opt. Pre-Define Symbol (-d): Feles Undefine Symbol (-u): Assembly Preprocessor Preprocessor Ivith Comments (-ppc) Diagnostics Ivith Compilation (-ppa) | Category: Basic ABI (-abi=): None, default to ARM9 ▼ Libraries Suppress Banner (-q) Output Module: ▼ Output Filename (-o): .\Debug\cyclone.out Map Filename (-m): .\cyclone.map Autoinit Model: Run-Time Autoinitialization (-c) ▼ Heap Size (+heap): 10 Stack Size (-stack): 200 Fill Value (f): Code Entry Point (-e): |
| OK Cancel Help | OK Cancel Help |

图4. 预编译(.PP)文件和地址映射(.MAP)文件的产生



| MEMORY CONFIGURATION | | | | | |
|---|-----------------|--------------------------|----------|------|------|
| name | origin | length | used | attr | fill |
| VECS | 00000000 | 00000020 | 00000020 | RWIX | |
| SECTION ALLOCATION MAP output section page origin | at length | ttributes/ input sect | ions | | |
| .text 0 0000020 0 | 00057b4 | | | | |
| GLOBAL SYMBOLS: SORTED ALPHAE address name | BETICALLY BY Na | ame | | | |
| 00019054 _debug_buffer | | | | | |
| 00019260 _supply_state | | | | | |
| GLOBAL SYMBOLS: SORTED BY Sym address name | nbol Address | | | | |
| 00019054 _debug_buffer | | | | | |
| 00019260 _supply_state | | | | | |

图5. 地址映像(.MAP)文件

| 😽 Fusion Memory Debugger - DC-DC H | ISFB Firmware v0.0.2 | .1 @ Address 8 | 18d | | | | | | | | | | |
|--|----------------------|----------------|---------------------------|------------|---------------|--------------|-----------------|-------|-------|--------------|----------|----------|--------|
| File | | | | | | | | | | | | | - |
| All 👷 Watch List | All 😾 Watch List | | | | | | | | | | | | |
| Register Variable Name: | | Substring Mat | h 🛛 🤇 | Filter | Clear Filter | Show: () All | | 0 | RAM | | sh () PE | lash 🔭 1 | Foggle |
| | 500 🚔 ma 🛛 Clobal Pr | Changes Cher | | ck All | | 0.0 | | 0 | | 0.00 | 0 | 101 | |
| | | | | | | | | | | | | | |
| Selection: <u>Watch List</u> () > supply_state (| 2 | | | | | | | | | | | | |
| Name | Description | Туре | Value | Hex | Address | Size | Category | + | - | Refresh | Write | AutoWr | Poll |
| 🗐 📌 debug_buffer | | Array unsig | | | 0x00019054 | 8 bytes | RAM | | | | | | |
| | | UInt8 | 1 | 0x01 | 0x00019054 | 1 byte | RAM | | | | | | |
| | | UInt8 | 2 | 0x02 | 0x00019055 | 1 byte | RAM | | | | | | |
| | | UInt8 | 3 | 0x03 | 0x00019056 | 1 byte | RAM | | | | | | |
| | | UInt8 | 4 | 0x04 | 0x00019057 | 1 byte | RAM | | | | | | |
| | | UInt8 | 5 | 0x05 | 0x00019058 | 1 byte | RAM | | | | | | |
| | | UInt8 | 6 | 0x06 | 0x00019059 | 1 byte | RAM | | | | | | |
| | | UInt8 | 7 | 0x07 | 0x0001905A | 1 byte | RAM | | | | | | |
| | | UInt8 | 8 | 0x08 | 0x0001905B | 1 byte | RAM | | | | | | |
| | | Enum SUPP | 5 - S 🖂 | 0x00000 | 0x00019260 | 4 bytes | RAM | | | REFRESH | WRITE | | |
| | | | U-SIAIE | IDLE | | | | | | | | | |
| | | | 1 - STATE | RAMP_UP_ | REBIAS | | | | | | | | |
| | | | 2 - STATE | _RAMP_UP | | | | | | | | | |
| | | | 3 - STATE | SYNFET_RA | MP_UP | | | | | | | | |
| | | | 4-STATE | FF_ENABLE | | | | | | | | | |
| | | | 5 - STATE | | | | | | _ | | | | |
| Expand All Collapse All | Refresh Watch List | Write Chang | 7 - STATE | | port | Selected | Export Watchlis | st) | (c | lear Watch | List | | |
| | | | 8 - STATE | CPCC | | | | _ | _ | | | | |
| | | | 9 - STATE VOUT TRANSITION | | | | | | | | | | |
| T | | | 10 - STATE | E_HICCUP | | | | | | | Copy Loc | Clear L | pq |
| Fusion Digital Power Designer v1.8.329 [2 | 013-05-29] DC-DC H | SFB Firmware v | 0.0.2.1 @ A | aaress ööa | IVIap Folder: | UCD3138HSF | BEVM_029 US | SB Ad | dapte | er v1.0.10 [| PEC; 400 | kHz] | |

图6. Memory Debugger 变量显示



为什么 Memory Debugger 需要.PP 和.MAP 文件? Memory Debugger 通过读取.PP 文件可以知道变 量是什么类型:结构体、数组、枚举等;通过.MAP 文件知道这些变量的地址。在图 5 中,可以看到变 量 debug_buffer 和 supply_state 的首地址分别是 0x00019054 和 00019260。图 5 中红色虚线是指省略 部分。在图 6 中,红框中是变量 debug_buffer 和 supply_state 的地址。debug_buffer 变量是 8 个字节的 数组,如图 6 中红框所示,可以和图 5 地址对照一下,是一样的; supply_state 是枚举变量,如图 6 绿 框所示,此时 supply_state 变量值是 5,处在 STATE_REGULATED 状态。

| 49 | _ 🗆 🔀 | | | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|--|--|--|
| – About Map/PP Files | | | | | | | | | | | | |
| While only IC registers can be debugged by default, i Composer output files are located, non-local global (e Code Composer files that are used are: | While only IC registers can be debugged by default, if you tell the debugger where certain Code Composer output files are located, non-local global (extern) variables can be debugged. The Code Composer files that are used are: | | | | | | | | | | | |
| *.map - A map file defines the top-level variables in your "C" source code and at what address in memory they have been located. There is only one map file. *.pp - Contains declarations from your source code, after they have been run through the C pre-processor. There will be one .pp file for each source code file. | | | | | | | | | | | | |
| Add the -m xxxxx.map linker option to create a map file during compilation. Add the -ppa -ppo pre-processor options to create the .pp files, one per source file. | | | | | | | | | | | | |
| Note that IC register definitions are taken from your .map and .pp files, overriding the default definitions bundled with the GUI. | | | | | | | | | | | | |
| and UCD9222, the debugger will automatically use th Current Setting: Folder: C:\CCStudio_v3.3\MyProject bdr\UCD3138HSFBEVM_029 | ts\PWR029_Test_Play_single_frame_ | | | | | | | | | | | |
| Select Folder Containing Map/PP Files | Select ZIP File Containing Map/PP Files | | | | | | | | | | | |
| Show Registers Only - Auto Select Map | Show Registers Only - Manually Select Map | | | | | | | | | | | |
| Show Registers Only - Auto Select Map Show Registers Only - Manually Select Map Collapse unions in favor of bit fields Because the debugger allows you to edit bit field structures full "hex" values in addition to the lower level bit fields, you may find that alternative byte and word representations of a struct bitfield complicate debugging. Checking this box simplifies the variable node tree by only showing bit field structs when other members of a union are only simple types. Code Composer small enums mode was used Check this if thesmall-enums mode was used in Code Composer when the target firmware was compiled. Small enums mode reduces the size that enums take up in memory (varable length instead of fixed 4 byte llength). | | | | | | | | | | | | |
| ОК | Cancel | | | | | | | | | | | |

图7. 选择.PP 和.MAP 文件

4 Memory Debugger 的具体操作

Memory Debugger 主要使用命令 0xE2 和 0xE3 和 UCD3xxx 通讯,所以 UCD3xxx 代码需要支持这两个命令。这两个命令可以直接读取 UCD3xxx 内部程序 FLASH 和数据 FLASH 的值,可以读取或修改内部 RAM 的值。关于这两个命令的详细代码,请参考文献 3。

图 1 中绿框所示是切换 Memory Debugger 显示内容,包括显示所有(ALL)、寄存器(Register)、 RAM、程序闪存(PFLASH)以及数据闪存(DFLASH)。如果想刷新某个变量,可以点击每个变量右 边的刷新(REFRESH)按钮,如图 1 中粉红框所示。



4.1 选择.PP 和.MAP 文件

在 Memory Debugger 文件菜单下,选择 Change Map...(图 1 红框所示),会弹出图 7 所示界面。 点击图 7 中红框所示,选择.PP 和.MAP 所在的目录(.PP 和.MAP 需要在同一个文件夹里面)。如果不 需要看全局变量,只看寄存器的话,可以不需要执行这一步。因为对于寄存器变量,地址都固定的, Memory Debugger 可以识别这些寄存器。对于变量,每个项目都不一样,即使同一个变量,地址也可能 不同,所以如果看全局变量,需要执行这一步。如果程序没有改动,执行一次就可以一直调试下去, Memory Debugger 会记住上一次的选择的.PP 和.MAP 文件。

| 🜵 Fusion Memory Debugger - DC-DC HS | FB Firmware v0.0.2.1 | l @ Address 8 | 8d | | | | | | | | | _ | |
|--|----------------------|---------------------|---------------------------|------------|--------------|----------------|-----------------|------|-------|--------------|------------|----------|-------|
| File | | | | | | | | | | | | | - |
| All 😪 Watch List | | | | | | | | | | | | | |
| Register/Variable Name: | | Substring Mat | ch 🗸 🤇 | Filter | Clear Filter | Show: All | O Registers | 0 | RAM | O DFlag | sh () PFli | ash 🎠 T | oggle |
| Start Polling Polling Interval: 50 | 00 🕀 ms 🛛 Global Pol | Change: <u>Chec</u> | <u>k All</u> <u>Unche</u> | eck All | | | | | | | | | |
| Selection: <u>Watch List</u> () > Dpwm0Regs () | | | | | | | | | | | | | |
| Name | Description | Туре | Value | Hex | Address | Size | Category | + | - | Refresh | Write | AutoWr | Poll |
| 🕀 📌 Dpwm0Regs | | Struct DPW | |] | 0x000D0000 | 140 bytes | Register | | | REFRESH | WRITE | | |
| 🕀 📌 Dpwm 1Regs | | Struct DPW | | | 0x000A0000 | 140 bytes | Register | | | | | | |
| 🕀 🕁 Dpwm2Regs | | Struct DPW | | | 0x00070000 | 140 bytes | Register | | | | | | |
| 🗄 🐈 Dpwm3Regs | Analog Comparator | Struct DPW | | | 0x00050000 | 140 bytes | Register | | | | | | |
| Expand All Collapse All | Refresh Watch List | Write Change | es 🗌 🗌 | Import | Export | Selected (| Export Watchlis | st) | C | lear Watch | List | | |
| Timestamp Message | | | | | | | | | | | Copy Log | Clear Lo | 20 |
| 09:27:02.313 0x000D0088: read 4 byte(s) | 0x0000000 | | | | | | | | | | | | |
| Fusion Digital Power Designer v1.8.329 [20] | 13-05-29] DC-DC HS | FB Firmware 🗸 |).0.2.1 @ A | ddress 88d | Map Folder: | UCD3138HSF | BEVM_029 US | B Ad | dapte | er v1.0.10 [| PEC; 400 I | (Hz] | |

图8. 监控列表(Watch List)视图

4.2 放置变量到监控列表(Watch List)

进入 Memory Debugger 界面时,是显示所有的变量和寄存器,如图 1 所示。需要注意的是,UCD3xxx 有些寄存器位是读清除的,所以不能随便读取某个寄存器,需要结合编程手册来看这个寄存器 是否可以读取。为了方便使用,可以把经常需要读取的寄存器和变量加入到监控列表(Watch List),如 图 8 红框所示。在图 1 中,如果我们点击每个变量左边灰色的星号(蓝色框所示),那么就可以把想要 监控的寄存器或变量加入到监控列表中,如图 8 所示。在监控列表中,可以设置自动刷新所监控的寄存 器或变量,自动刷新频率也可以设置,如图 8 绿色方框所示。

4.3 实验演示

把寄存器 Fiter0Regs 放到监控列表,展开后找到 OUTPUT_CLAMP_HIGH,如图 9 所示。初始值 是 0x2500。点击 0x2500,输入 0x1250,然后点击右边的 WRITE,写入 UCD3xxx,如图 10 所示。

图 11 是 DPWM0A/B (Ch1-DPWM0A, Ch2-DPWM0B)的波形,可以看出修改前后 DPWM 的最大 占空比的变化。图 11 左边是最大占空比修改前,右边是最大占空比修改后。实验是在 TI 实验板 (UCD3138OL40EVM-032,见文献 4)上验证的,把反馈短接到地,然后把示例代码各种保护屏蔽掉, 那么程序就可以运转起来,环路会一直输出最大占空比。



| 🚸 Fusion Memory Debugger - DC-DC HSFB Firm | ware v0.0.2.1 | @ Address 88 | d | | | | | | | | | _ | |
|---|---------------|-----------------|-------|----------|------------|-----------|----------|---|---|---------|-------|--------|------|
| File | | | | | | | | | | | | | - |
| All 👷 Watch List | | | | | | | | | | | | | |
| Register/Variable Name: Substring Match Sitter Clear Filter Show: All Registers RAM ODFlash PFlash | | | | | | | | | | | | oggle | |
| Start Polling Polling Interval: 500 🕀 ms Global Poll Change: Check All Uncheck All | | | | | | | | | | | | | |
| Selection: Watch List () > Filter0Regs () > FILTEROCLPHI () > bit () > bit Fields () > OUTPUT CLAMP HIGH [17:0] () (Output Clamp High) | | | | | | | | | | | | | |
| Name | Description | Туре | Value | Hex | Address | Size | Category | + | - | Refresh | Write | AutoWr | Poll |
| Filter0Regs | | Struct FILT | | | 0x000C0000 | 100 bytes | Register | | | | | | |
| E 📌 FILTEROCLPHI | Filter Outp | Union FILT | | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| 🖨 🤆 bit | | Struct FILT | | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| 😑 👷 Bit Fields | | Bit Fields | | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| | Reserved | U Bit Field: 14 | 0 | 0x0000 | 0x000C0058 | 14 bits | Register | | | | | | |
| | Output Cla | U Bit Field: 18 | 9,472 | 0x002500 | 0x000C0059 | 18 bits | Register | • | | REFRESH | WRITE | | |
| | | UInt32 | 9,472 | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| Expand All Collapse All Refresh WatchList Write Changes Import Export Selected Export Watchlist Clear Watch List Timestamn Message Copy Log Clear Log All | | | | | | | | | | | | | |
| 16:00:10.294 0x000C0014: read 4 byte(s) 0x00000 | 000 | | | | | | | | | | | | |

图9. DPWM 输出最大占空比修改前

| 😽 Fusion Memory Debugger - DC-DC HSFB Firmy | vare v0.0.2.1 | @ Address 88 | d | | | | | | | | | - | |
|--|---------------|-----------------|------------|------------|-------------|---------------------|----------------------|------|-------|------------|----------|-----------------|------|
| File | | | | | | | | | | | | | - |
| All 👷 Watch List | | | | | | | | | | | | | |
| Register/Variable Name: Substring Match V Filter Clear Filter Show: All Registers RAM DEFlash Figure Toggle | | | | | | | | | | | | | |
| Start Polling Polling Interval: 500 ⊕ ms Global Poll Change: Check All Uncheck All | | | | | | | | | | | | | |
| Selection: Watch List () > FilterORegs () > FILTER | ROCLPHI C) > | bit 🞝 > Bit | Fields 🕻 | > OUTPUT | CLAMP HIGH | [17:0] Č) ((| Output Clamp Hig | h) | | | | | |
| | Description | Type | Value | Hev | Address | Size | Category | | | Defrech | Write | AutoWr | Poll |
| | Description | Struct FILT | Value | TIEX | 0x000C0000 | 100 bytes | Register | 1 | - | Keirean | WITCE | | |
| | Filter Outp | Union FILT | | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| 🗐 🥋 bit | | Struct FILT | | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| 😑 😾 Bit Fields | | Bit Fields | | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| | Reserved | U Bit Field: 14 | 0 | 0x0000 | 0x000C0058 | 14 bits | Register | | | | | | |
| UTPUT_CLAMP_HIGH [17:0] | Output Cla | U Bit Field: 18 | 4,688 | 0x001250 | 0x000C0059 | 18 bits | Register | • | | REFRESH | WRITE | | |
| i | | UInt32 | 4,688 | 0x0000 | 0x000C0058 | 4 bytes | Register | | | | | | |
| | | Write Channes | | T | Gunt | Colored) | [Even a diversaliate | - | | 9 | 1.1.1 | | |
| Expand All Collapse All Refresh | WatchList | write Changes | | Import | Export | Selected | Export Watchil | st | | lear water | LIST | | |
| | | | | | | | | | | | | | |
| Timestamp Message | | | | | | | | | | | Copy Loc | <u>Clear Lo</u> | 20 |
| 16:02:04.500 0x000C0058: read 4 byte(s) 0x000012 | 250 | | | | | | | | | | | | |
| Fusion Digital Power Designer v1.8.329 [2013-05-29] | DC-DC HSF | B Firmware v0. | 0.2.1 @ Ad | ldress 88d | Map Folder: | UCD3138HS | FBEVM_029 | SB A | dapte | er v1.0.10 | PEC; 400 | kHz] | |

图10. DPWM 输出最大占空比修改后

5 总结

本文主要介绍使用 Memory Debugger 来调试 UCD3xxx。通过本文可以看出,利用 Memory Debugger,可以灵活地监控及修改变量,达到调试的目的。



| Te <u>k Stop</u> | | | Ŭ U | | | | lek Stop | | | T | | | |
|------------------|---|------------------------------|--------------------|-----------|----------------|------------|----------|---|----------|--------------------|----------------------|----------------|-------------|
| | | | | | | | | | | | | | |
| | promotion and the | | | | | | | yennen gebennen | | | | | |
| 2 | | | | | | | 2 | | | | | | |
| | | | 2 0005 | 500MS/8 | | 6 Sap 2012 | | | | 2.000 | 500MS/c | | 6 Sap 2012) |
| Coupling | V N 2 Termination (Ω) 1M 75 50 | 2.00 V N Invert On Off | Bandwidth 20MHz | 10k point | 2.52 V More | 17:58:00 | Coupling | / % 2 Termination (Ω) 1M 75 50 | 2.00 V N | Bandwidth 20MHz | 10k point 2 Label | 2.52 V More | 17:59:01 |

图11. DPWM 输出最大占空比修改前后的波形

6 参考文献

1. PMBUS Power System Management Protocol Specification Part II – Command Language, System Management Interface Forum (SMIF), Inc., 2007;

2. SPNU151H - ARM Optimizing C/C++ Compiler v5.0 User's Guide, Texas Instruments, 2012

3. UCD3138HSFBEVM-029 Firmware (PWR029_Test_Play_single_frame_bdr), Texas Instruments, 2013

4. SLUUA80 - Using the UCD3138OL40EVM-032 User's Guide, Texas Instruments, 2013

5. SLUSAP2C - UCD3138 Highly Integrated Digital Controller for Isolated Power, Texas Instruments, 2013

重要声明

德州仪器(TI)及其下属子公司有权根据 JESD46 最新标准,对所提供的产品和服务进行更正、修改、增强、改进或其它更改,并有权根据 JESD48 最新标准中止提供任何产品和服务。客户在下订单前应获取最新的相关信息,并验证这些信息是否完整且是最新的。所有产品的销售 都遵循在订单确认时所提供的TI 销售条款与条件。

TI 保证其所销售的组件的性能符合产品销售时 TI 半导体产品销售条件与条款的适用规范。仅在 TI 保证的范围内,且 TI 认为 有必要时才会使用测试或其它质量控制技术。除非适用法律做出了硬性规定,否则没有必要对每种组件的所有参数进行测试。

TI 对应用帮助或客户产品设计不承担任何义务。客户应对其使用 TI 组件的产品和应用自行负责。为尽量减小与客户产品和应 用相关的风险,客户应提供充分的设计与操作安全措施。

TI不对任何 TI 专利权、版权、屏蔽作品权或其它与使用了 TI 组件或服务的组合设备、机器或流程相关的 TI 知识产权中授予 的直接或隐含权 限作出任何保证或解释。TI 所发布的与第三方产品或服务有关的信息,不能构成从 TI 获得使用这些产品或服 务的许可、授权、或认可。使用 此类信息可能需要获得第三方的专利权或其它知识产权方面的许可,或是 TI 的专利权或其它 知识产权方面的许可。

对于 TI 的产品手册或数据表中 TI 信息的重要部分,仅在没有对内容进行任何篡改且带有相关授权、条件、限制和声明的情况 下才允许进行 复制。TI 对此类篡改过的文件不承担任何责任或义务。复制第三方的信息可能需要服从额外的限制条件。

在转售 TI 组件或服务时,如果对该组件或服务参数的陈述与 TI 标明的参数相比存在差异或虚假成分,则会失去相关 TI 组件 或服务的所有明示或暗示授权,且这是不正当的、欺诈性商业行为。TI 对任何此类虚假陈述均不承担任何责任或义务。

客户认可并同意,尽管任何应用相关信息或支持仍可能由 TI 提供,但他们将独力负责满足与其产品及在其应用中使用 TI 产品 相关的所有法律、法规和安全相关要求。客户声明并同意,他们具备制定与实施安全措施所需的全部专业技术和知识,可预见 故障的危险后果、监测故障及其后果、降低有可能造成人身伤害的故障的发生机率并采取适当的补救措施。客户将全额赔偿因 在此类安全关键应用中使用任何 TI 组件而对 TI 及其代理造成的任何损失。

在某些场合中,为了推进安全相关应用有可能对 TI 组件进行特别的促销。TI 的目标是利用此类组件帮助客户设计和创立其特 有的可满足适用的功能安全性标准和要求的终端产品解决方案。尽管如此,此类组件仍然服从这些条款。

TI 组件未获得用于 FDA Class III(或类似的生命攸关医疗设备)的授权许可,除非各方授权官员已经达成了专门管控此类使 用的特别协议。

只有那些 TI 特别注明属于军用等级或"增强型塑料"的 TI 组件才是设计或专门用于军事/航空应用或环境的。购买者认可并同 意,对并非指定面向军事或航空航天用途的 TI 组件进行军事或航空航天方面的应用,其风险由客户单独承担,并且由客户独 力负责满足与此类使用相关的所有法律和法规要求。

TI 己明确指定符合 ISO/TS16949 要求的产品,这些产品主要用于汽车。在任何情况下,因使用非指定产品而无法达到 ISO/TS16949 要求,TI不承担任何责任。

| | 产品 | | 应用 |
|---------------|------------------------------------|--------------|--------------------------|
| 数字音频 | www.ti.com.cn/audio | 通信与电信 | www.ti.com.cn/telecom |
| 放大器和线性器件 | www.ti.com.cn/amplifiers | 计算机及周边 | www.ti.com.cn/computer |
| 数据转换器 | www.ti.com.cn/dataconverters | 消费电子 | www.ti.com/consumer-apps |
| DLP® 产品 | www.dlp.com | 能源 | www.ti.com/energy |
| DSP - 数字信号处理器 | www.ti.com.cn/dsp | 工业应用 | www.ti.com.cn/industrial |
| 时钟和计时器 | www.ti.com.cn/clockandtimers | 医疗电子 | www.ti.com.cn/medical |
| 接口 | www.ti.com.cn/interface | 安防应用 | www.ti.com.cn/security |
| 逻辑 | www.ti.com.cn/logic | 汽车电子 | www.ti.com.cn/automotive |
| 电源管理 | www.ti.com.cn/power | 视频和影像 | www.ti.com.cn/video |
| 微控制器 (MCU) | www.ti.com.cn/microcontrollers | | |
| RFID 系统 | www.ti.com.cn/rfidsys | | |
| OMAP应用处理器 | www.ti.com/omap | | |
| 无线连通性 | www.ti.com.cn/wirelessconnectivity | 德州仪器在线技术支持社区 | www.deyisupport.com |

邮寄地址: 上海市浦东新区世纪大道1568 号,中建大厦32 楼邮政编码: 200122 Copyright © 2014, 德州仪器半导体技术(上海)有限公司