DoCD™

DCD on Chip Debug System

ver 4.70

OVERVIEW

The most important possibility for SoC designers are pre-silicon debugging. DCD has introduced into the market a new 8051 and 80390 IP Cores family with an embedded Debug IP Core. DCD on Chip Debug System (DoCD™) prominently cuts debugging time. Integrating DCD IP Cores with a Hardware Assisted Debugger and Debug IP Core provides a powerful SoC development tool with advanced features.

The DoCD™ system consists of three major blocks:

- Debug IP Core
- Hardware Assisted Debugger
- Debug Software

The Debug IP Core block is a **real-time hardware debugger** provides access to whole chip registers, memories and peripherals connected to DCD's IP Core (Dx8051/Dx80390/DRPIC/DFPIC), and controls CPU work by **non-intrusive** method. A high-performance Hardware Assisted Debugger (**USB-xTAG**) is connected to the target system containing the DCD's core either in FPGA or ASIC. The Hardware Assisted Debugger manages communication between the Debug IP Core inside silicon using DTAG, TTAG, or JTAG protocol, and Debug Software using USB port.

FLASH Programming

DoCD debugger fully supports programming of all FLASH memory devices. Such support is assured by configurability of FLASH programming algorithm, and supported devices database. New FLASH device can be easily added to existing base using build-in editor. DoCD debugger allows user to simply perform in-system programming of its FLASH memory without using any external equipment. FLASH programming task is performed directly within Debug software, and after uploading of code, it is ready for debugging. Programming time is very short, because of HAD2 support. This feature saves time, and makes usage of DCD's debugger very comfortable and flexible.

http://www.DigitalCoreDesign.com
http://www.dcd.pl

Non-Intrusive System

In typical intrusive systems a debugging tool consumes for its own needs some system resources e.g.: part of program space, several cells of RAM memory, ports' pins sometimes system is loosing interrupts or the program code is manipulated to support software breakpoints, and so on.
Even simple debugging systems consumes the UART and timer resources to support own tasks. These simple 'emulators' cannot provide trace and other advanced debugging functions, while also being very intrusive in the debugging cycle. Imagine trying to debug an interrupt problem while the 'emulator' is manipulating interrupts itself!

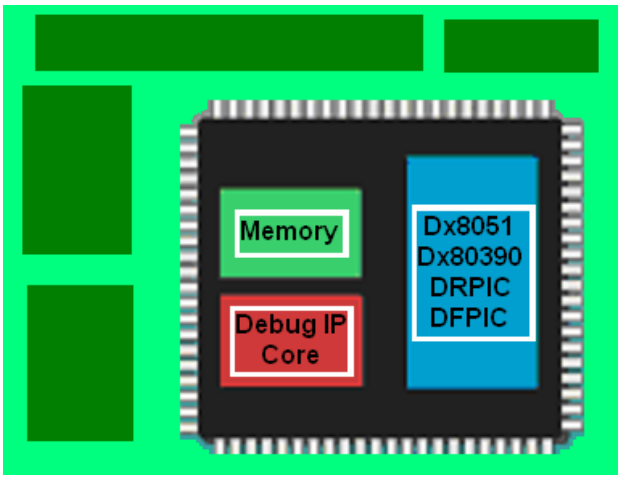
Developing firmware is all about producing code that is 100% reliable in operation and fully understood in how it will perform in adverse conditions. A real non intrusive on-chip debugger that assists user in this task is the most important tool user can have. That is the reason why using of non intrusive systems is so important, that is also the reason, why the DoCD™ debugging tool, has been designed as a non-intrusive system.

Real-time Hardware Debugger

Real-time hardware debugger we call for a tool that is able to detect processor internal properties that are not visible outside the processor without any violation of real-time operations. DoCD™ gives you the chance to track down hidden bugs within the application running with microcontroller. Internal events such as the reading of the SBUF-control register are not mirrored on the external address-data bus. However, by using special logic to detect operations that affect internal resources, DoCD™ gives user ability to track such internal events without any violation of real-time operation. There is no need to use a special external logic for the emulation.

Debug IP Core

The Debug IP Core is provided as VHDL or Verilog source code as well as CPLD/FPGA EDIF netlist depending on the customer requirements. DoCD™ provides a scaled solution, because many SoC designs have both power and area limitations. Debug IP Core can be scaled to control gate count. The benefit is fewer gates for lower power and core size while trading off debug capability. Typically, all of the features are utilized in pre-silicon debug (i.e. hardware emulation or FPGA evaluation) with a lesser feature set shipped in final silicon.



KEY FEATURES OF DEBUG IP CORE

- Processor execution control
  - *Run, Halt*
  - *Reset*
  - *Step into instruction*
  - *Skip Instruction*
- Read-write all processor contents
  - *Program Counter (PC)*
  - *Program Memory*
  - *Internal (direct) Data Memory*
  - *Special Function Registers (SFRs)*
  - *External Data Memory*
- Code execution breakpoints
  - *first real-time PC breakpoint*
  - *second real-time PC breakpoint (v 4.50 and above)*
  - *unlimited number of real-time OPCODE breakpoints (v 4.00 and above)*
- Hardware execution watch-points
  - *two at Internal (direct) Data Memory*
  - *two at Special Function Registers (SFRs)*
  - *two at External Data Memory*
- Hardware watch-points activated at a
  - *certain address by any write into memory*
  - *certain address by any read from memory*
  - *certain address by write into memory a required data*
  - *certain address by read from memory a required data*
- Unlimited number of software watch-points
  - *Internal (direct) Data Memory*

http://www.DigitalCoreDesign.com
http://www.dcd.pl

- *Special Function Registers (SFRs)*
- *External Data Memory*

- Unlimited number of software breakpoints
  - *Program Memory(PC)*
- Automatic adjustment of debug data transfer speed rate between HAD and Silicon
- Communication interface
  - *TTAG interface – v4.70 and above*
  - *JTAG interface – v4.00 and above*
  - *DTAG three wire communication – v3.xx*
- Fully static synchronous design with no internal tri-states

HARDWARE ASSISTED DEBUGGER

Hardware Assisted Debugger (HAD) is a small hardware adapter that manages communication between the Debug IP Core inside silicon and a USB port of the host PC running DoCD™ Debug Software.

KEY FEATURES OF HAD2 PCB BOARD

- USB communication interface to target host at FULL speed
- Synchronous communication interface to Debug IP Core
  - *TTAG interface – Debug IP v4.70 and above*
  - *JTAG interface – Debug IP v4.00 and above*
  - *DTAG interface – Debug IP v3.xx*
- Supports following I/O voltage standards
  - *3.3 Volt systems*
  - *2.5 Volt systems*
  - *1.8 Volt systems*
  - *1.5 Volt systems*
- Single power supply directly from USB host
- Small physical dimensions – pendrive package

DEBUG SOFTWARE

The DoCD™ Software (DS) is a Windows® based application. It is fully compatible with all existing 8051/80390 C compilers and Assemblers. The DS allows user to work in three major modes: software simulator mode, in circuit emulator mode and hardware debugger mode. Those modes assure possibility to pre-silicon software validation in simulation mode and then real-time debugging of developed software inside silicon – using debugger mode. Once loaded, the program may be observed in Source Window, run at full-speed, single stepped by machine or C-level instructions, or stopped at any of the breakpoints and watchpoints.

The DoCD™ Debug Software supports all DCD's DST8051x/DP8051x/DP80390x in the following architectures:

- Super Tiny RISC – (DST -2 cycle)
- Pipelined High Performance RISC – (DP -1 cycle)

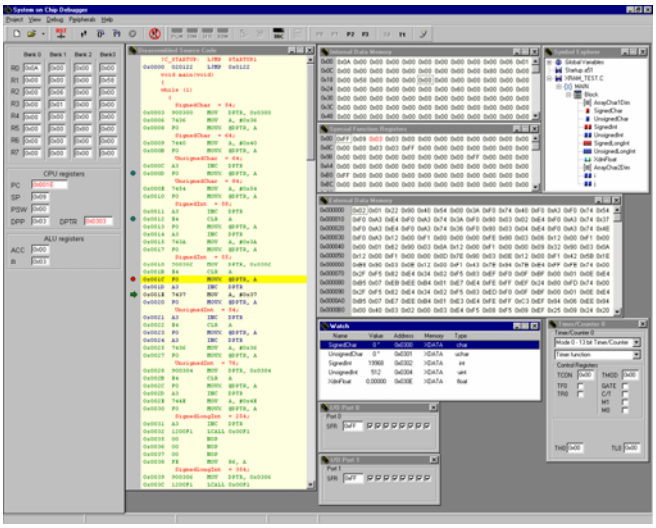
with their particular configurations.

KEY FEATURES OF DEBUG SOFTWARE

- In-system FLASH programming
- Three working modes
  - *hardware debugger*
  - *in circuit emulator*
  - *software simulator*
- Source Level Debugging:
  - *C level hardware/software breakpoints*
  - *C code execution*
    - *line by line*
    - *over line*
    - *out of function*
    - *skip line*
  - *ASM code execution*
    - *Instruction by instruction*
    - *over instruction*
    - *out of function*
    - *skip instruction*
  - *ASM, C source view of code*
- Symbol Explorer provides hierarchical tree view of all symbols:
  - *modules*
  - *functions*
  - *blocks*
  - *variables and more*
- Contents sensitive Watch window:
  - *Local variables view*
  - *Up to 3 independent watches*
- Symbolic debug including:
  - *code*
  - *variables*
  - *variable types*

http://www.DigitalCoreDesign.com
http://www.dcd.pl

- **Unlimited** number of Real-time hardware breakpoints
  - *Program Memory (CODE)*
- Two real-time hardware watch-points for each space:
  - *Internal (direct) Data Memory (IDM)*
  - *Special Function Registers (SFR)*
  - *eXternal Data Memory (XDM)*
- Unlimited number of software breakpoints
  - *Program Memory*
  - *Internal (direct) Data Memory (IDM)*
  - *Special Function Registers (SFR)*
  - *eXternal Data Memory (XDM)*
- Set/clear software or hardware break-points, watch-points in Disassembled and C Source Code windows
- 1024 steps deep Software Trace
- Load Program Memory content from:
  - *OMF-51, extended OMF-51 files*
  - *OMF-251 file*
  - *CDB object format*
  - *Intel HEX-51, HEX-386 files*
  - *BIN file*
- Auto refresh of all windows during execution of program
  - *Registers' panel including ACC, B, PSW, PC, SP, DPTR, DPP and four banks of general purpose registers R0-R7*
  - *Internal (direct) Data Memory (IDM)*
  - *Special Function Registers (SFR)*
  - *eXternal Data Memory (XDM)*
  - *Timers/Counters*
  - *UARTs*
  - *I/O Ports*
- Dedicated windows for peripherals
- Configurable auto refresh time period with 1s step resolution
- Status bar containing number of actually executed instructions, number of clock periods and real processor speed rate
- Hardware Assisted Debugger interface
  - *TTAG interface*
  - *JTAG interface*
  - *DTAG interface*
- The system runs on a Windows® 98/Me/ 2000/XP/Vista PC
- Supports software tools from Keil, Archimedes, IAR, Tasking, Franklin, SDCC and the others



http://www.DigitalCoreDesign.com
http://www.dcd.pl

DEBUG IP CORE PINOUT

The following pins are used by DoCD™ debug IP Core.

A. TTAG interface – v 4.70 and above:

PIN	TYPE	DESCRIPTION	
ttdi	input	DoCD™ data input	Connected together as **ttdio** pin
ttdoen	output	DoCD™ data output enable	
ttdo	output	DoCD™ data output	
ttck	output	DoCD™ clock line	

B. JTAG interface – v 4.00 and above:

PIN	TYPE	DESCRIPTION
tdi	input	DoCD™ TAP data input
tck	input	DoCD™ TAP clock line
tms	input	DoCD™ TAP mode select
tdo	output	DoCD™ TAP data output
rtck	output	DoCD™ return clock

C. DTAG interface – v 3.xx :

PIN	TYPE	DESCRIPTION
docddatai	input	DoCD™ data input
docddatao	output	DoCD™ data output
docdclk	output	DoCD™ clock line

AREA UTILIZATION

The following table gives a survey about the Debug IP Core area inside the FPGA and ASIC devices.

Device vendor		Area		
		DTAG	JTAG	TTAG
ALTERA	[LCs]	720	600	470
XILINX	[Slices]	360	300	240
LATTICE	[LUT4s]	720	610	500
ASIC	[gates]	2500	2100	1650

SYSTEM FEATURES

- FLASH PROGRAMMING:

  All FLASH memory devices are supported by DoCD debug system. Such support is assured by configurability of FLASH programming algorithm, and supported devices database. New FLASH device can be easily added to existing base using build-in editor. DoCD debugger allows user to simply perform in-system programming of its FLASH memory without using any additional equipment. DoCD programming task is performed directly within Debug software, and after uploading of code, it is ready for debugging. Programming time is very short, and writes operations are performed with maximal allowed speed by certain FLASH device.

- HARDWARE BREAKPOINTS:

  The number of hardware breakpoints is **unlimited**. Like software breakpoints, hardware execution breakpoints can be set in Program Memory space. They stop program execution just **prior** an instruction pointed by Program Counter (PC). In the other words instruction located at the PC breakpoint address is not executed. The difference is found in the method of program execution. In this case program is run with full clock speed (in real-time), and processor is halted when hardware signalizes true breakpoint condition.

- HARDWARE WATCH-POINTS:

  The number of hardware watch-points is limited to six in different address spaces. Like software breakpoints, hardware execution watch-points can be set in direct RAM, SFRs and external RAM. They stop program execution **after** an instruction being executed. The difference is found in the method of program execution. In this case program is run with full clock speed (in real-time), and processor is halted when hardware signalizes true watch-point condition.

- SOFTWARE BREAKPOINTS:

  An unlimited number of software breakpoints can be set anywhere in the physical address space of the processor. This means that breakpoints can be set in Program Memory space, direct RAM, SFRs and external RAM. If at least one software breakpoint is set program is executed in

automatic step by step mode, with checking if certain breakpoint condition is met. Program execution is halted when breakpoint condition is already met, and its execution can be resumed at any time in any appropriate mode.

- **MIXED MODE BREAKPOINTS:**

  Mixed breakpoint mode is also allowed and it means that software and hardware breakpoints, and watch-points are mixed in the system. This gives user flexibility in the debugging. For example two different break conditions can be set using watch-points and hardware breakpoints.

  In each breakpoint mode halt means: CPU is halted and instructions are no longer being fetched, and all internal peripherals are also stopped (e.g. timers, watchdog). The UARTs work correctly in any case.

- **SYMBOL EXPLORER:**

  Symbol Explorer provides hierarchical tree view of all C project symbols. It supports all C types, variables, constants, functions, and symbolic names of registers. Along with watch window provides flexible and powerful debugging feature at high C language level.

- **SCALED SOLUTION:**

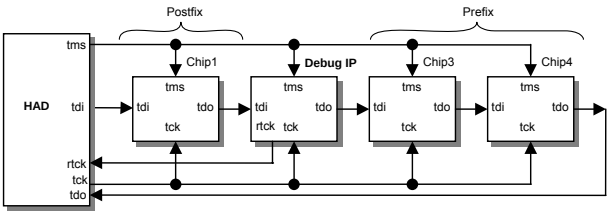
  Because many SoC designs have both power and gate limitations, DCD provides a scaled solution. Debug extensions can be scaled to control gate counts. The benefits are fewer gates, lower power and core size while trading off debug capability.

- **HOST REQUIREMENTS:**

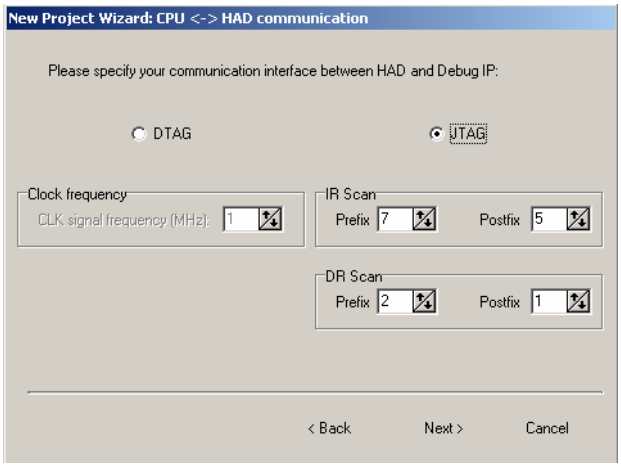
  A Pentium class computer with minimum 32 MB of memory, 10 MB of free space on Hard Disk, CD-ROM drive, USB port, and Windows® 98/Me or Windows® 2000/XP/Vista operating system are required.

DOCD in JTAG chain

The DoCD™ debug IP Core v 4.00 and above can be used as standalone device, as well as plugged into JTAG chain. It means that standard JTAG pins can be used, and other JTAG devices can be controlled along with DoCD Debug IP. Such solution saves off-chip pins of ASIC/FPGA device.



The example target shown in figure above consists of DoCD Debug IP and three devices being fully JTAG compliant. The Chip1 has 5-bit long IR (Instruction Register), Chip3 IR's has 3-bit long, and Chip4 has 4-bit long IR. A DR (Data Register) is always 1-bit long for each JTAG device.
The following values should be written into DoCD Windows Debug Software configuration window:
IR–prefix (3+4), DR-prefix (1+1)
IR–postfix (5), DR-postfix (1)
It is shown in figure below.



The 0 value should be written in an appropriate IR and DR field, in case where there would be no any prefix or postfix devices.

http://www.DigitalCoreDesign.com
http://www.dcd.pl

CONTACTS

For any modification or special request please contact to Digital Core Design or local distributors.

Headquarters:

Wroclawska 94

41-902 Bytom, POLAND

e-mail: info@dcd.pl

tel. : +48 32 282 82 66

fax : +48 32 282 74 37

Distributors:

Please check http://www.dcd.pl/apartn.php