

Dual Core(D1088) AICE Debug with Demo Project

Driving Innovations™

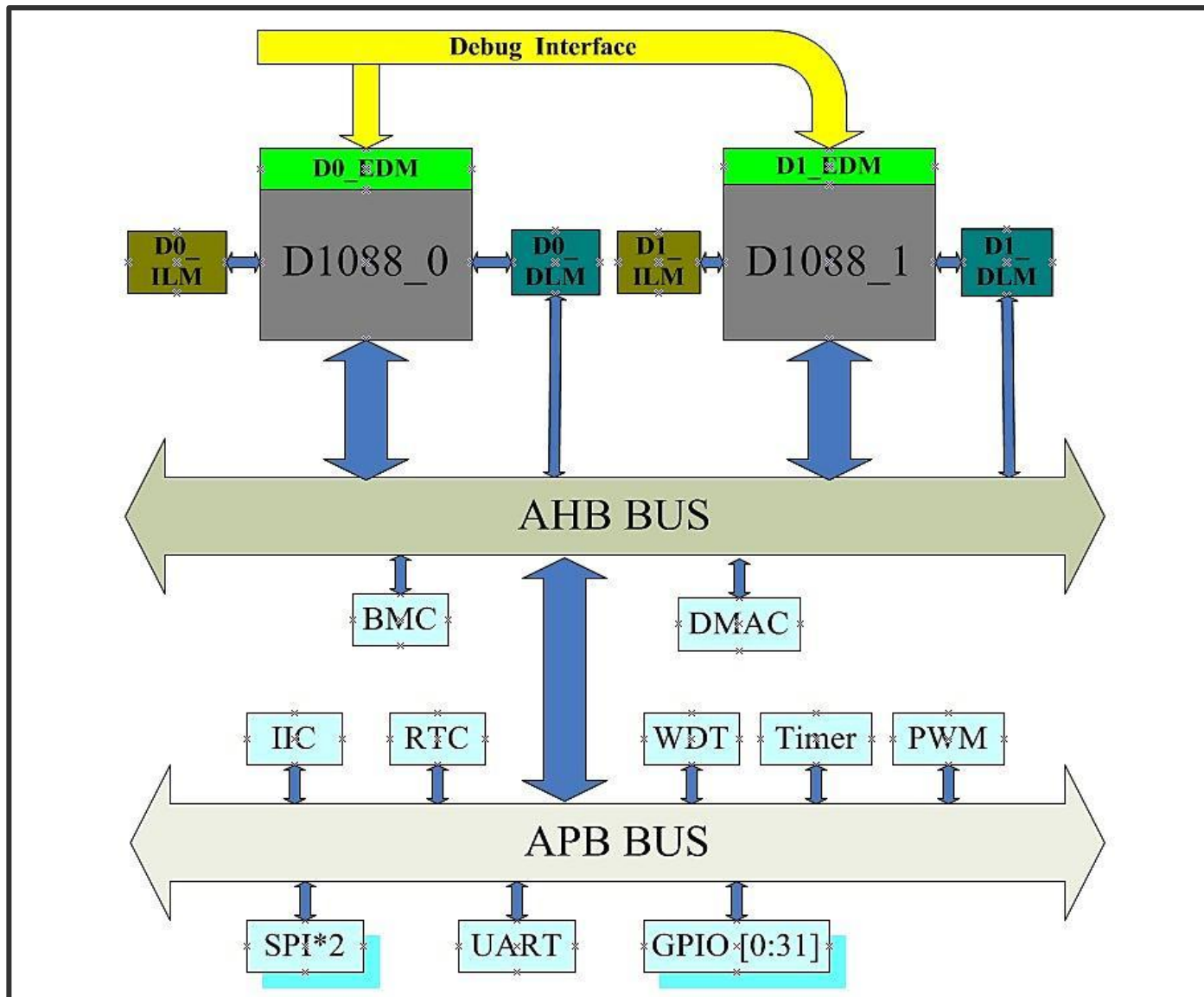


Agenda

- ❖ Block Diagram of Dual core(D1088)
- ❖ Need for Multi Core Debugging
- ❖ Andes Multi Core Debug Connection
- ❖ Demo Environment Setting
 - Demo Environment
 - Demo Project Introduction
 - Flow chart
- ❖ Demonstration (For Execution in AndeSight™ v3.0.0)
 - Check Diagnostic Report (For Confirmation of cores)
 - Procedure to follow
- ❖ Demonstration (For Execution in Cygwin Window)
 - Issue ICEman.exe
 - Open Cygwin
 - Execute GDB
 - Load and Run Project

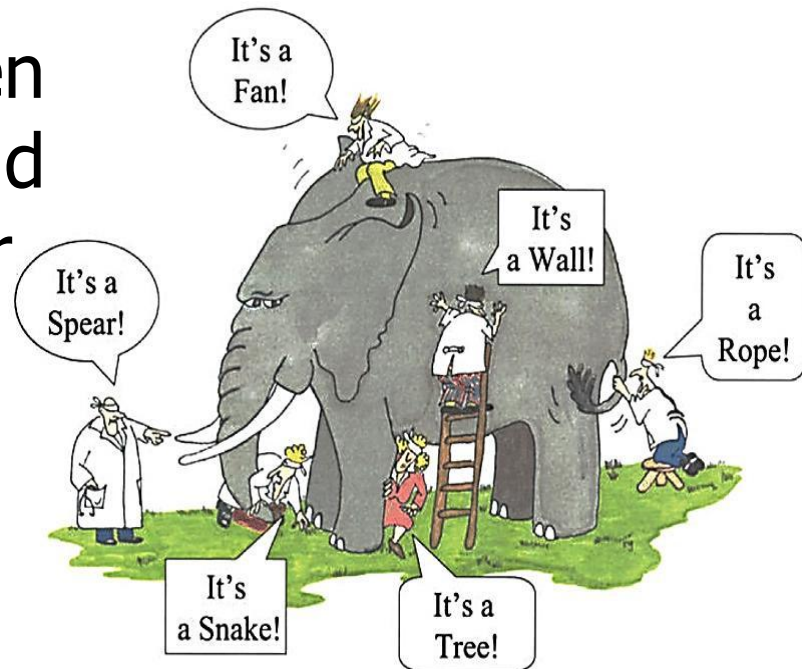
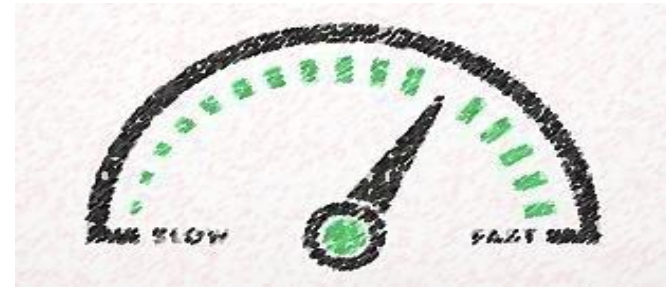


Block Diagram of Dual Core(D1088)



Need for Multi Core Debugging

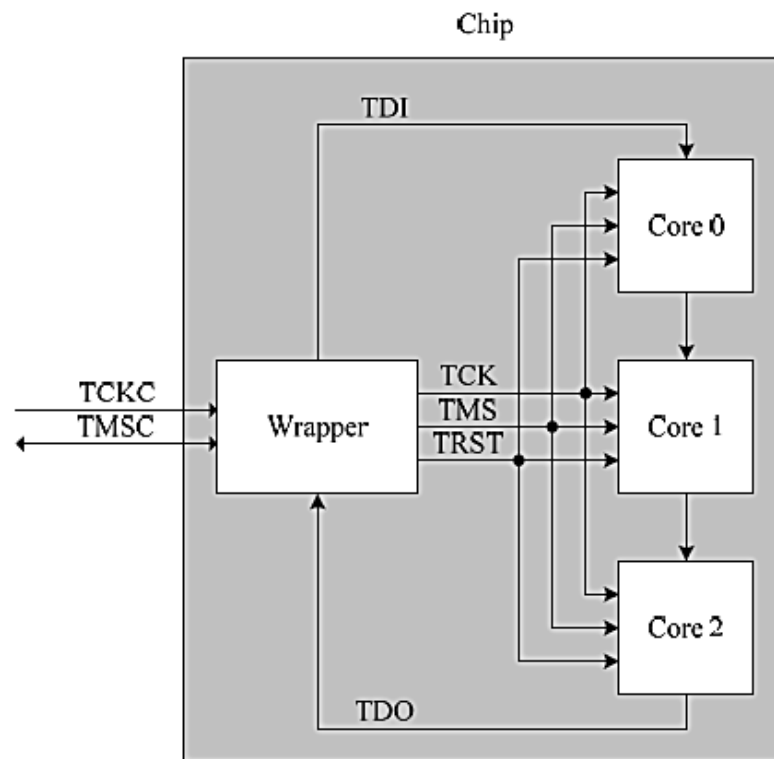
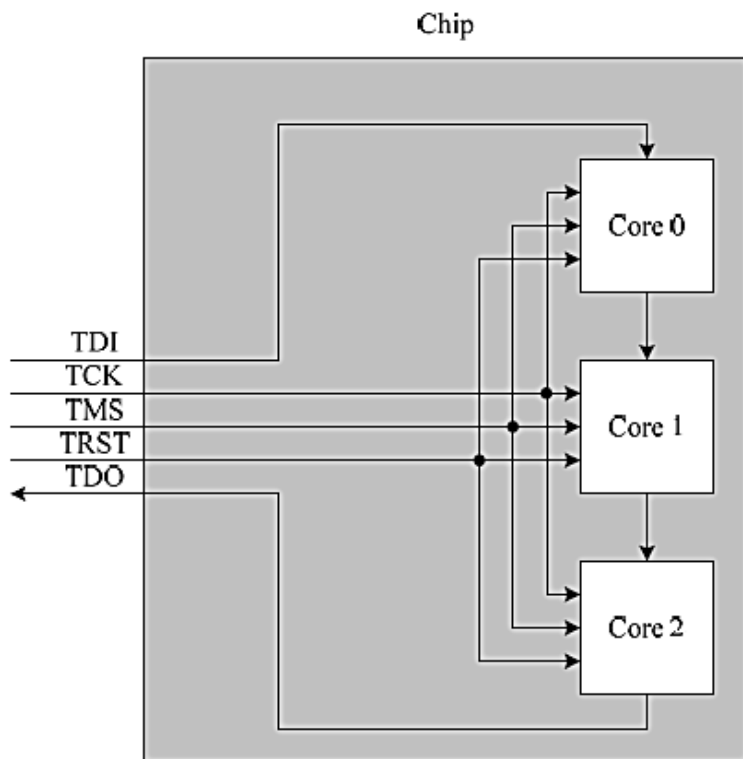
- ❖ In complex SoCs, just to observe and control a single core is insufficient.
- ❖ Rather the interactions of multiple cores are needed when user wants to detect, trace, and eliminate software problems or to profile system behavior for performance optimization.



Andes Multi Core Debug Connection

❖ Single-chip Multi-Core System

- 5-wire interface (AICE-MCU)
- 2-wire interface (AICE-MINI)



Demo Environment Setting

Driving Innovations™



Demo Environment

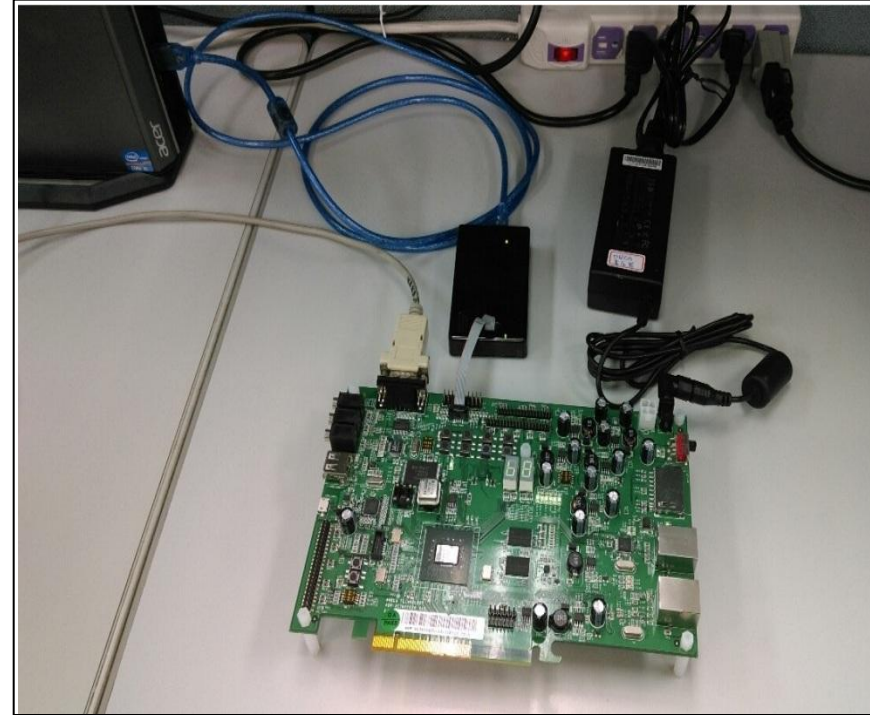
❖ Netlist: Dual D1088+AE210P

❖ Demo project:

- Core 0: C0_7SEG
- Core 1: C1_int

❖ Components:

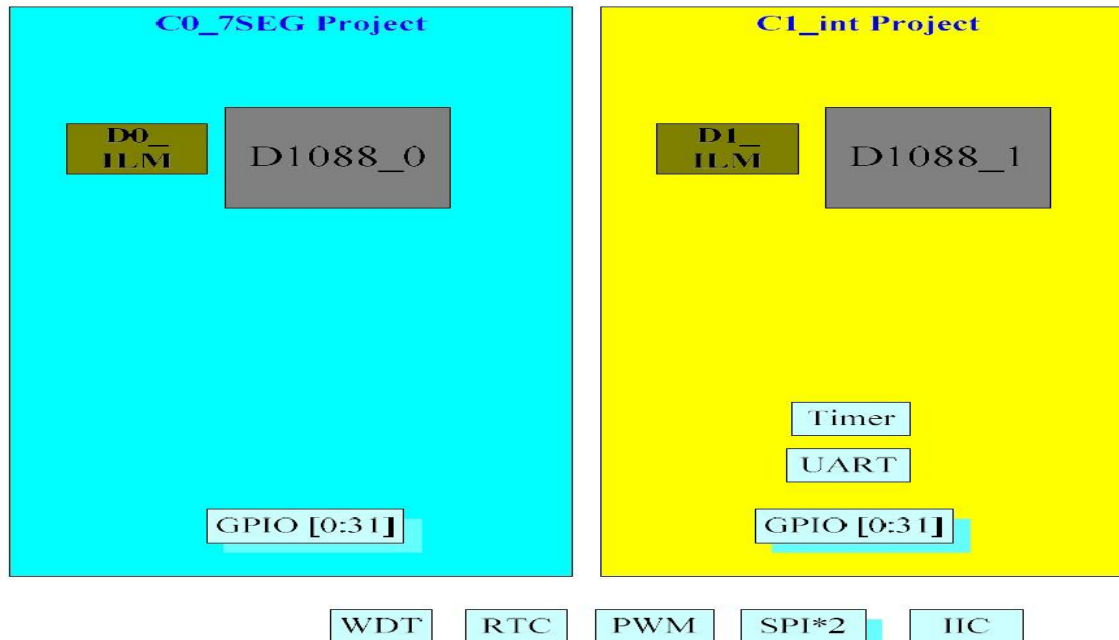
XC7_FPGA Board , AICE- MCU & UART cable.



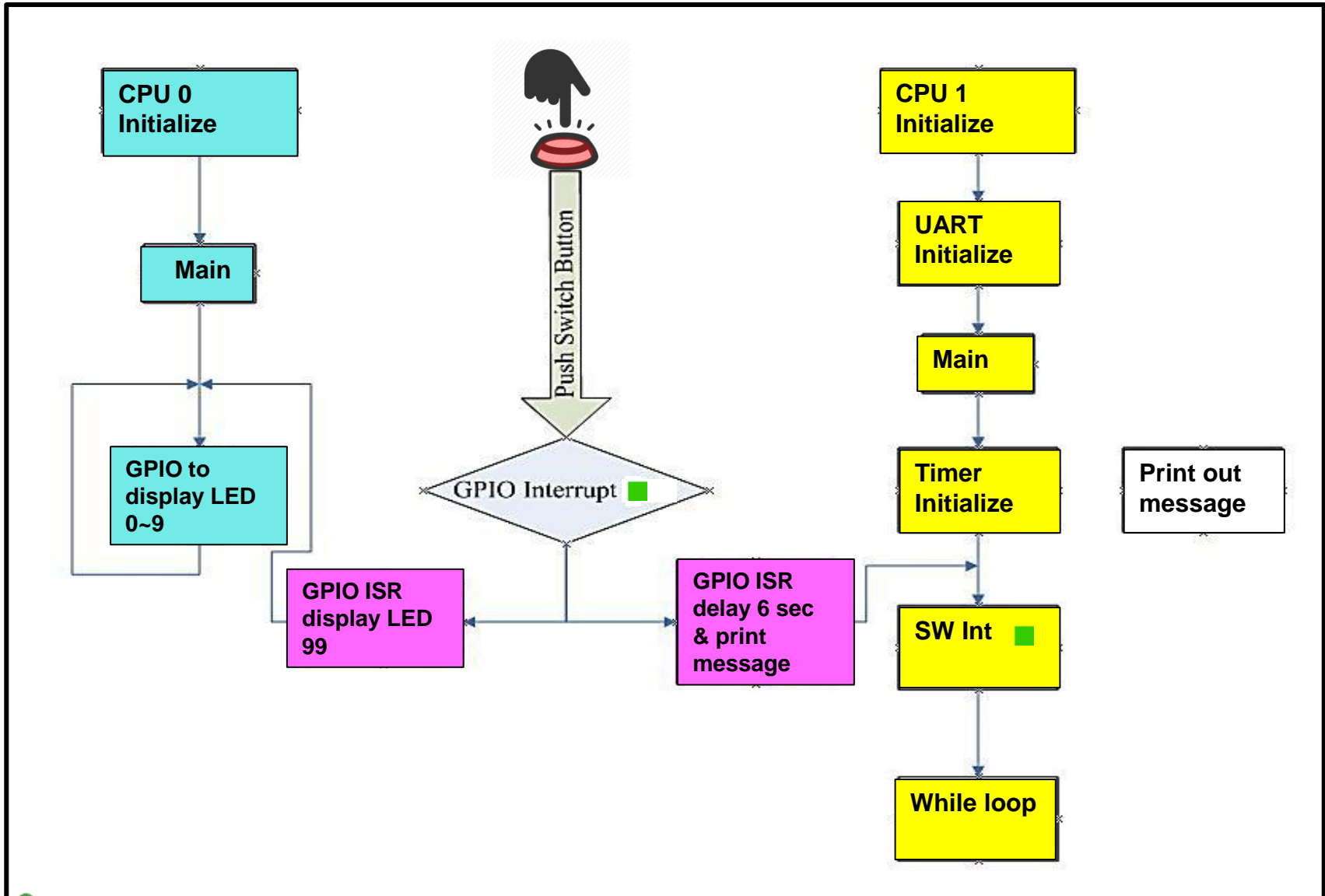
- NOTE: Though ICEman supports the assignment of AICE configuration file to multi-cores at a time, BSP v4.0 accepts only one coprocessor for a CPU core.

Demo Project Introduction

- ❖ The demo scenario is:
 - Let both CPU(D1088) do "Reset and Hold"
 - Load individual project(C0_7SEG & C1_int) to ILM in each CPU Core.
- ❖ Both C0_7SEG & C1_int program execute at separate core(D1088)
- ❖ A custom script can be used to perform AICE RESET-HOLD in a multi-core system if the ICEman default reset timing is not suitable.

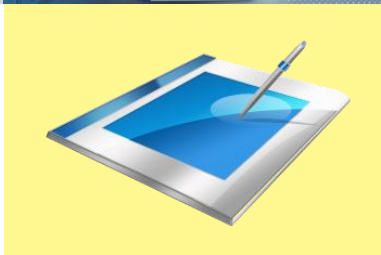
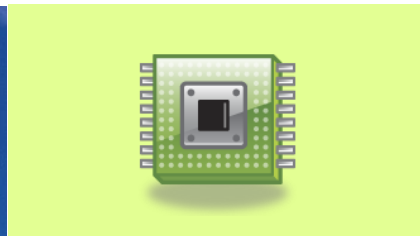
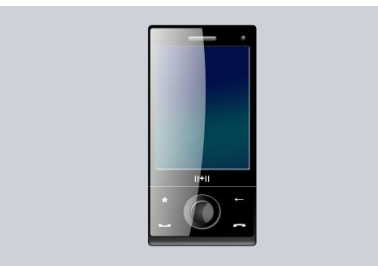


Flow Chart



Demonstration (For Execution in AndeSight™ v3.0.0)

Driving Innovations™



Check Diagnostic Report (For Confirmation of cores)

Andes ICEman v3.4.0 (OpenOCD) BUILD_ID:
2017022117

Burner listens on 2354

Telnet port: 4444

TCL port: 6666

Andes AICE-MCU v1.8.8

There are 2 cores in target

JTAG frequency 24 MHz

Diagnostic Report

JTAG frequency 24 MHz, 9

set JTAG frequency 12 MHz

set JTAG frequency 24 MHz

ice_state = 00000009

There are 2 cores in target

Core #0: EDM version 0x1010

hardware reset-and-hold success

[PASS] check USB connectivity

[PASS] check AICE versions

[PASS] check the detected JTAG frequency

[PASS] check changing the JTAG frequency

[PASS] check JTAG connectivity

[PASS] check that JTAG domain is operational

[PASS] check that TRST resets the JTAG domain

[PASS] check that SRST does not resets JTAG domain

[PASS] check selecting core

[PASS] check reset-and-debug

[PASS] check that DIM and CPU domain are
operational

Core #1: EDM version 0x1010

hardware reset-and-hold success

[PASS] check USB connectivity

[PASS] check AICE versions

[PASS] check the detected JTAG frequency

[PASS] check changing the JTAG frequency

[PASS] check JTAG connectivity

[PASS] check that JTAG domain is operational

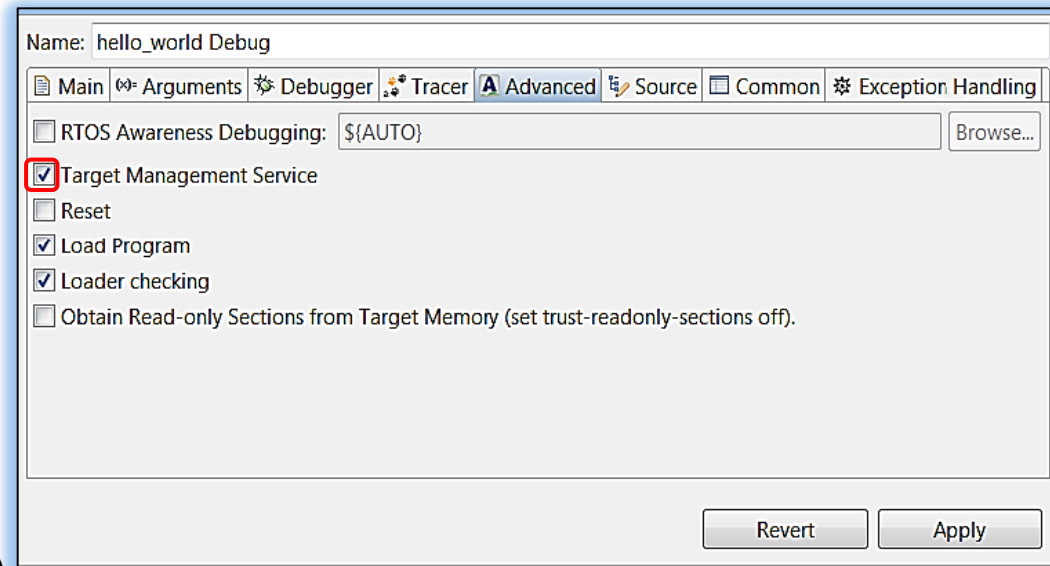
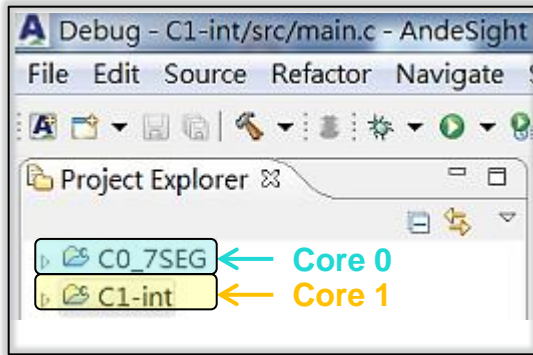
[PASS] check that TRST resets the JTAG domain

[PASS] check that SRST does not resets JTAG domain

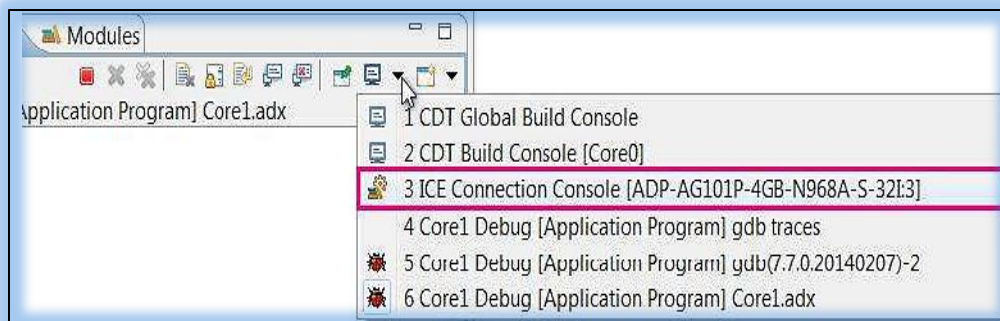
[PASS] check selecting core

Procedure to follow

- 1) Create and build project for each core
- 2) Launch a debug session for the **Core 0** project.



- 3) Check the Console view and switch to “ICE Connection Console” (For Core 0)



- Note: Target Management Service” in the Debug Configurations dialog must be selected.

Procedure to follow (1)



✓ Take note of the port number that each core listens to in the ICE Connection Console. For example:

- The **Core #0** listens on **9902**.
- The **Core #1** listens on **9903**.

3) Run the debug session and note the CPU core and Port number

The screenshot displays the ANDES IDE interface with the following components:

- Source Editor:** Shows a C program snippet:

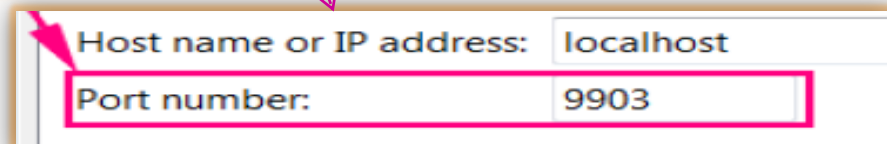
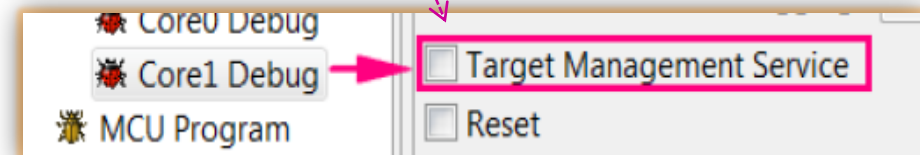
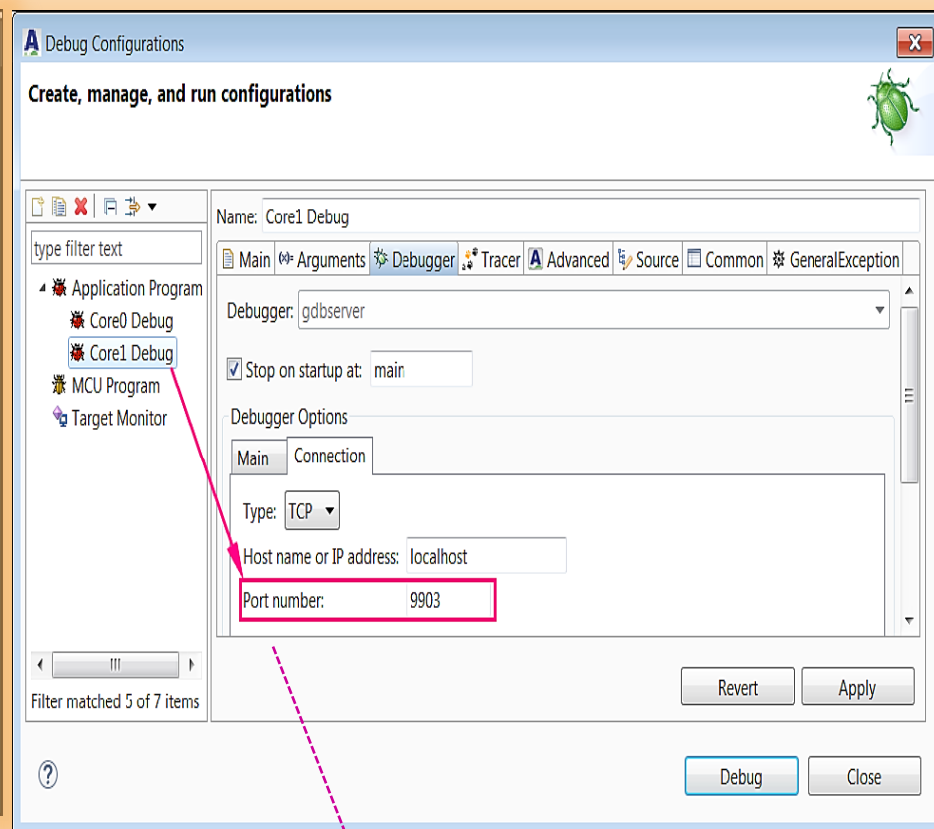
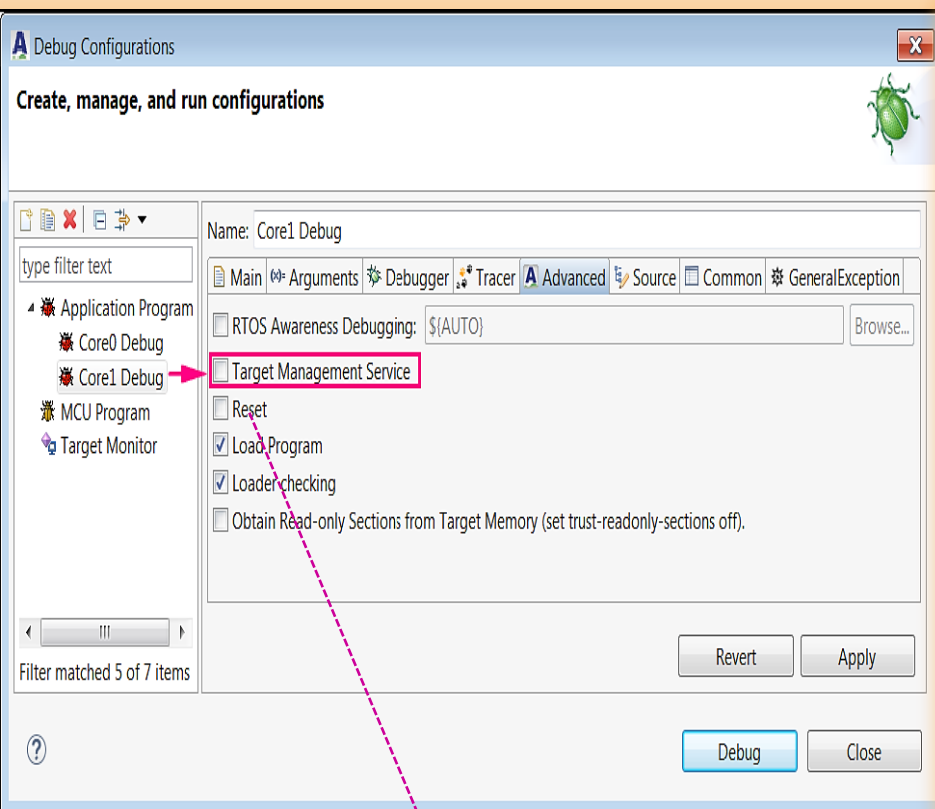
```
11 }  
12 int main()  
13 {  
14 {  
15 while (1){  
16     outw(GPIO_BASE + GPIO_DIR_OFFSET, 0x00BF0000); /*SEG'0'*/  
17     delay(1000000);  
18 }
```
- Console:** Displays GDB startup messages for CO_7SEG.adx, including warranty information and configuration details.

```
CO_7SEG.adx [Application Program] gdb(7.7.0.20140207)-4  
and "show warranty" for details.  
This GDB was configured as "--host=i686-pc-cygwin --target=nds32le-elf".  
Type "show configuration" for configuration details.  
For bug reporting instructions, please see:  
<http://www.gnu.org/software/gdb/bugs/>.  
Find the GDB manual and other documentation resources online at:  
<http://www.gnu.org/software/gdb/documentation/>.  
For help, type "help".  
Type "apropos word" to search for commands related to "word".
```
- Breakpoints:** A single breakpoint is set for `djpeg.c` at line 554.
- Target Information:** A tooltip (circled in red) provides the following details:
 - Target: ADP-AE210P-D1088
 - CPU0 GDBServer: 9902
 - CPU1 GDBServer: 9903
 - Burner Port: 9900
- Status Bar:** Shows "Select process" and a system clock indicating 11:49 AM on 2017/5/10.

Procedure to follow (2)



4) Uncheck “Target Management Service” and go for “Debugger > Connection” tab to specify the port no. (For Core 1)



NOTE:



Multiple projects may be used simultaneously for multi-core debugging. As breakpoints in AndeSight are global across projects in the workspace, a breakpoint that user sets for a project at a source line or on an address may be hit in other projects having the same filename or address.

Please close unused debug sessions to avoid such a case – or just resume the debug session if an unexpected breakpoint is hit.

Procedure to follow (3)

The screenshot displays the Andes Studio IDE interface during a debugging session. The **Project Explorer** on the left shows a project with files like `C0_7SEG`, `C1-int`, and `demo-cache`. The **Debug** window in the center shows two application programs: `C0_7SEG.adx` and `C1-int Debug [Application Program]`, each with a running thread. A blue bracket groups these two threads, and a text overlay reads: **C0 and C1 simultaneous debugging**. The **Source** window shows the `main.c` file with the following code:

```
74  uart_puts("data section copy failed.\n");
75  while (1) ;
76  }
77  uart_puts("data section copy successfully.\n");
78  if (global_bss != 0) {
79      uart_puts("bss section clear failed.\n");
80      while (1) ;
81  }
82  uart_puts("bss section clean successfully.\n");
83
84  /* This is syscall test.
85   * You can comment it if it is not necessary. */
86  /* Generate system call */
87  asm("syscall 0x5000:::$r0");
88
89  /* Initialize interrupt */
```

The **Variables** window on the right is empty. The **SoC Registers** window shows a list of registers with their values and descriptions. The **Console** window at the bottom shows the output of the program, including messages about timer ISR and GIE enablement.

Serial COM3 (5/10/17 11:32 AM)

```
* Bottom-Half of Timer ISR is done and it takes 2 secs. Enable it self.*
* Enter Timer ISR, It comes in every 4 secs. *
* Top-Half of Timer ISR is done. Enable GIE *
* Bottom-Half of Timer ISR is done and it takes 2 secs. Enable it self.*
* Enter Timer ISR, It comes in every 4 secs. *
* Top-Half of Timer ISR is done. Enable GIE *
* Bottom-Half of Timer ISR is done and it takes 2 secs. Enable it self.*
* Enter Timer ISR, It comes in every 4 secs. *
* Top-Half of Timer ISR is done. Enable GIE *
* Bottom-Half of Timer ISR is done and it takes 2 secs. Enable it self.*
```

Demonstration (For Execution in Cygwin Window)

Driving Innovations™



Issue ICEman.exe

- ❖ Open **Cygwin** window
- ❖ Issue ICEman.exe -p 1234 -H to get the GDB Port ID.

```
C:\ /cygdrive/c/Andestech/AndeSight211MCU/ice
The core #0 listens on 1234.
The core #1 listens on 1235.
ICEman is ready to use.

jchen@APC171 /cygdrive/c/Andestech/AndeSight211MCU/ice
$ ICEman.exe -p 1234 -H
Andes ICEman v3.2.2 (OpenOCD) BUILD_ID: 2015082815
Burner listens on 2354
Telnet port: 4444
TCL port: 6666
Open On-Chip Debugger 0.8.0-dev-g1a84463 (2015-08-28-15:34)
Licensed under GNU GPL v2
For bug reports, read
    http://openocd.sourceforge.net/doc/doxygen/bugs.html
Andes AICE-MCU v1.8.8
There are 2 cores in target
Core #0: EDM version 0x1010
Core #1: EDM version 0x1010
JTAG frequency 24 MHz
The core #0 listens on 1234.
The core #1 listens on 1235.
ICEman is ready to use.

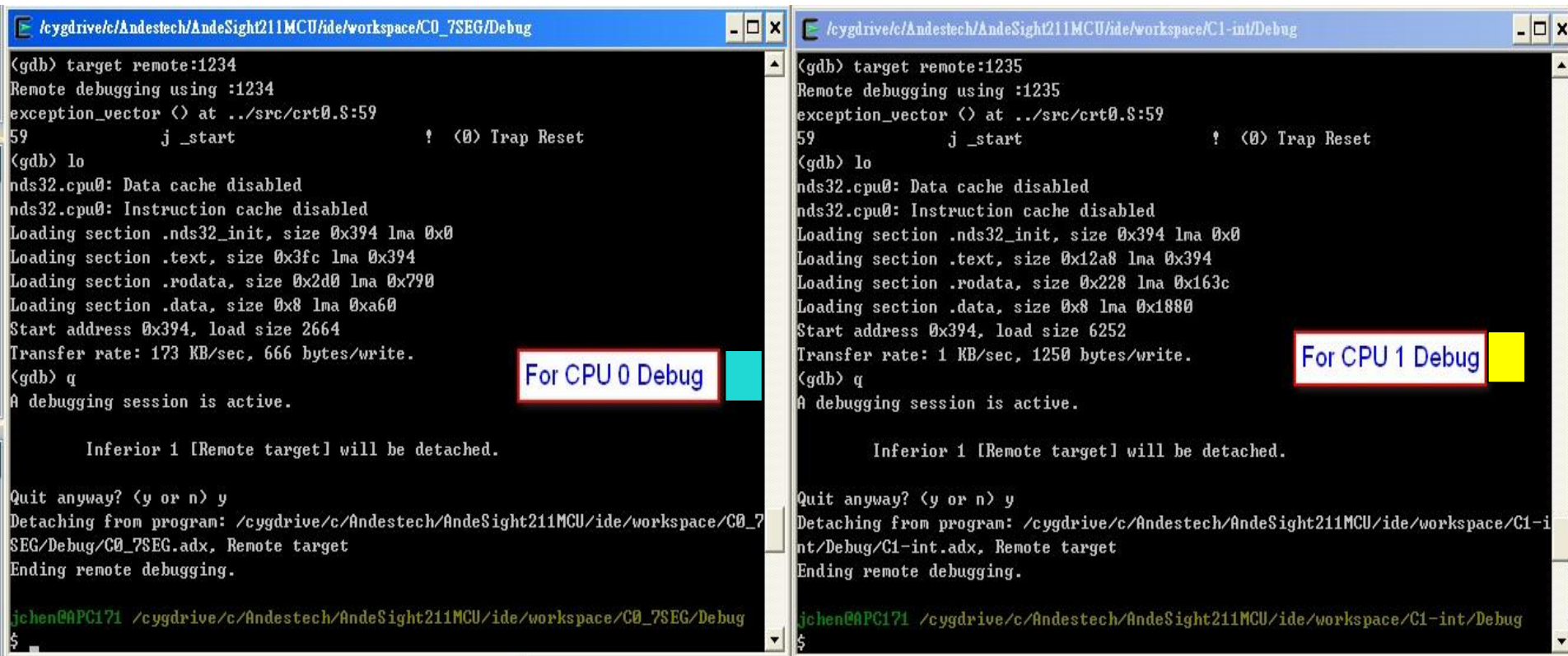
jchen@APC171 /cygdrive/c/Andestech/AndeSight211MCU/ice
$
```

Diagram illustrating the mapping of core listening ports to CPU identifiers:

- For CPU 0 (represented by a cyan square)
- For CPU 1 (represented by a yellow square)

Open Cygwin

- ❖ Open two Cygwin windows for separate CPU .



The image shows two side-by-side Cygwin terminal windows. The left window is titled '/cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C0_7SEG/Debug' and the right window is titled '/cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C1-int/Debug'. Both windows show a GDB session for a remote target. The left window is for CPU 0 (target 1234) and the right window is for CPU 1 (target 1235). Both sessions show the same sequence of commands and output: setting the target, loading sections, and starting the debugging session. The left window has a red box labeled 'For CPU 0 Debug' and the right window has a red box labeled 'For CPU 1 Debug'.

```
<gdb> target remote:1234
Remote debugging using :1234
exception_vector (<) at ../src/crt0.S:59
59          j _start          ! (<0) Trap Reset
<gdb> lo
nds32.cpu0: Data cache disabled
nds32.cpu0: Instruction cache disabled
Loading section .nds32_init, size 0x394 lma 0x0
Loading section .text, size 0x3fc lma 0x394
Loading section .rodata, size 0x2d0 lma 0x790
Loading section .data, size 0x8 lma 0xa60
Start address 0x394, load size 2664
Transfer rate: 173 KB/sec, 666 bytes/write.
<gdb> q
A debugging session is active.

Inferior 1 [Remote target] will be detached.

Quit anyway? (y or n) y
Detaching from program: /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C0_7SEG/Debug/C0_7SEG.adx, Remote target
Ending remote debugging.

jchen@APC171 /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C0_7SEG/Debug
$
```

```
<gdb> target remote:1235
Remote debugging using :1235
exception_vector (<) at ../src/crt0.S:59
59          j _start          ! (<0) Trap Reset
<gdb> lo
nds32.cpu0: Data cache disabled
nds32.cpu0: Instruction cache disabled
Loading section .nds32_init, size 0x394 lma 0x0
Loading section .text, size 0x12a8 lma 0x394
Loading section .rodata, size 0x228 lma 0x163c
Loading section .data, size 0x8 lma 0x1880
Start address 0x394, load size 6252
Transfer rate: 1 KB/sec, 1250 bytes/write.
<gdb> q
A debugging session is active.

Inferior 1 [Remote target] will be detached.

Quit anyway? (y or n) y
Detaching from program: /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C1-int/Debug/C1-int.adx, Remote target
Ending remote debugging.

jchen@APC171 /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C1-int/Debug
$
```


Execute GDB

- ❖ Go through individual GDB port to connect separate CPU

```
Quit anyway? (y or n) y
Detaching from program: /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C0_7SEG/Debug/C0_7SEG.adx, Remote target
Ending remote debugging.

jichen@APC171 /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C0_7SEG/Debug
$ nds32le-elf-gdb.exe C0_7SEG.adx
GNU gdb (2015-08-22_nds32le-elf) 7.7.0.20140207-cvs
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "--host=i686-pc-cygwin --target=nds32le-elf".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
[info] Loading .Andesgdbinit.
[info] .Andesgdbinit loaded.
Reading symbols from C0_7SEG.adx...done.
(gdb) target remote:1234
```

```
Quit anyway? (y or n) y
Detaching from program: /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C1-int/Debug/C1-int.adx, Remote target
Ending remote debugging.

jichen@APC171 /cygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C1-int/Debug
$ nds32le-elf-gdb.exe C1-int.adx
GNU gdb (2015-08-22_nds32le-elf) 7.7.0.20140207-cvs
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "--host=i686-pc-cygwin --target=nds32le-elf".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
[info] Loading .Andesgdbinit.
[info] .Andesgdbinit loaded.
Reading symbols from C1-int.adx...done.
(gdb) target remote:1235
```


Load and Run Project

- ❖ Load into different projects to the implementation of the ILM on each CPU, and start Debug.

```
lcygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C0_7SEG/Debug
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
[info] Loading .Andesgdbinit.
[info] .Andesgdbinit loaded.
Reading symbols from C0_7SEG.adx...done.
(gdb) target remote:1234
Remote debugging using :1234
0x00000558 in delay (tt=...) at ../src/main-printf.c:11
11          for (ii=0; ii < tt; ii++)
(gdb) lo
nds32.cpu0: Data cache disabled
nds32.cpu0: Instruction cache disabled
Loading section .nds32_init, size 0x394 lma 0x0
Loading section .text, size 0x3fc lma 0x394
Loading section .rodata, size 0x2d0 lma 0x790
Loading section .data, size 0x8 lma 0xa60
Start address 0x394, load size 2664
Transfer rate: 41 KB/sec, 666 bytes/write.
(gdb) c
Continuing.
```

Load project into CPU 0

Starting Free run or debug project

```
lcygdrive/c/Andestech/AndeSight211MCU/ide/workspace/C1-int/Debug
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
[info] Loading .Andesgdbinit.
[info] .Andesgdbinit loaded.
Reading symbols from C1-int.adx...done.
(gdb) target remote:1235
Remote debugging using :1235
exception_vector (<) at ../src/crt0.S:59
59          j _start
(gdb) lo
nds32.cpu0: Data cache disabled
nds32.cpu0: Instruction cache disabled
Loading section .nds32_init, size 0x394 lma 0x0
Loading section .text, size 0x12a8 lma 0x394
Loading section .rodata, size 0x228 lma 0x163c
Loading section .data, size 0x8 lma 0x1880
Start address 0x394, load size 6252
Transfer rate: 1 KB/sec, 1250 bytes/write.
(gdb) c
Continuing.
```

Load Project into CPU 1

Starting Free run or debug project