Access CPU with AndeSight

- Create a project
- Access CPU with “Reset and Hold”
- Access CPU without “Reset and Hold”
- Access CPU with “Target Monitor”

Create Empty Project

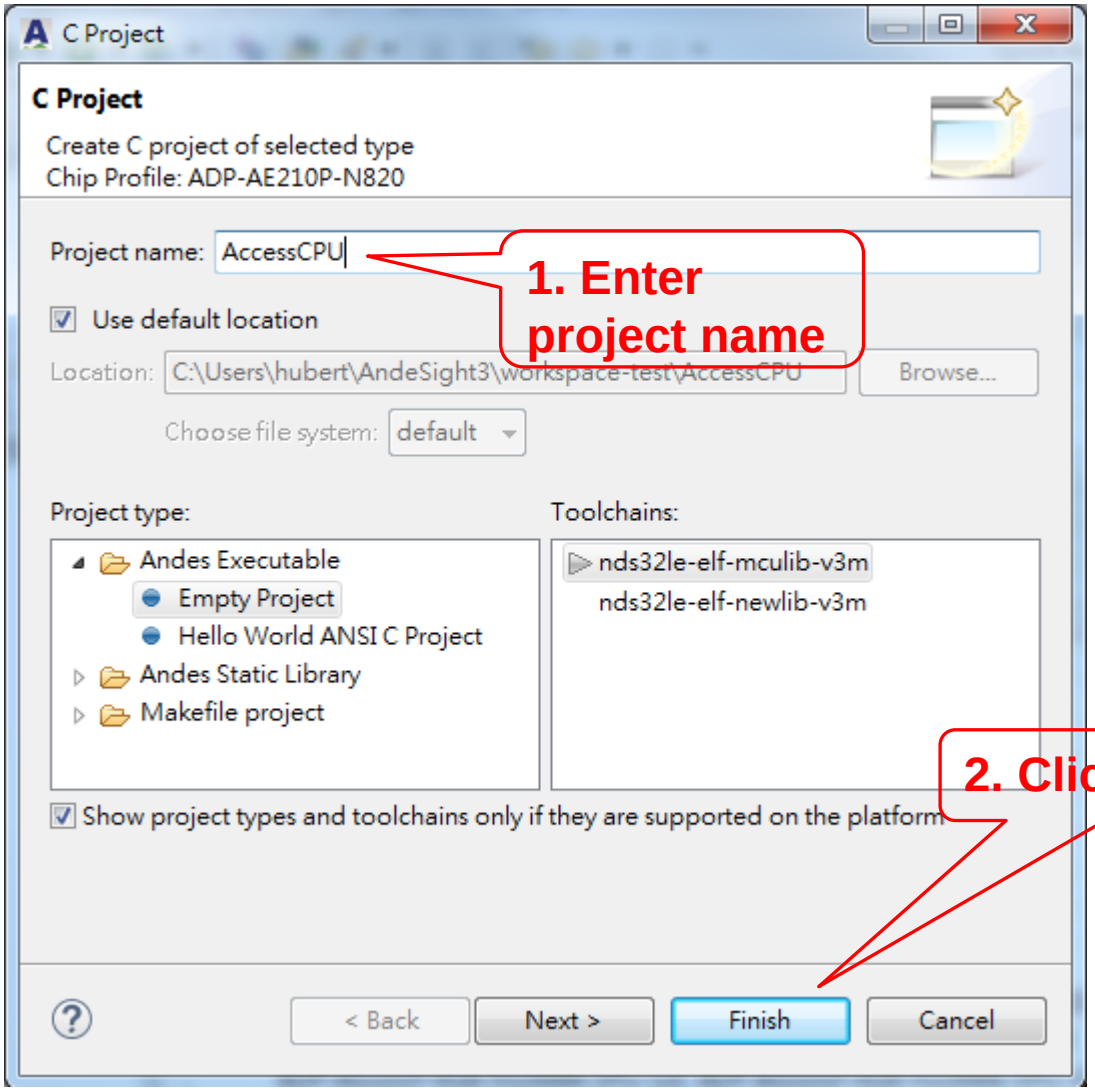
1. Click to create project

2. Select ICE

3. Select your chip profile

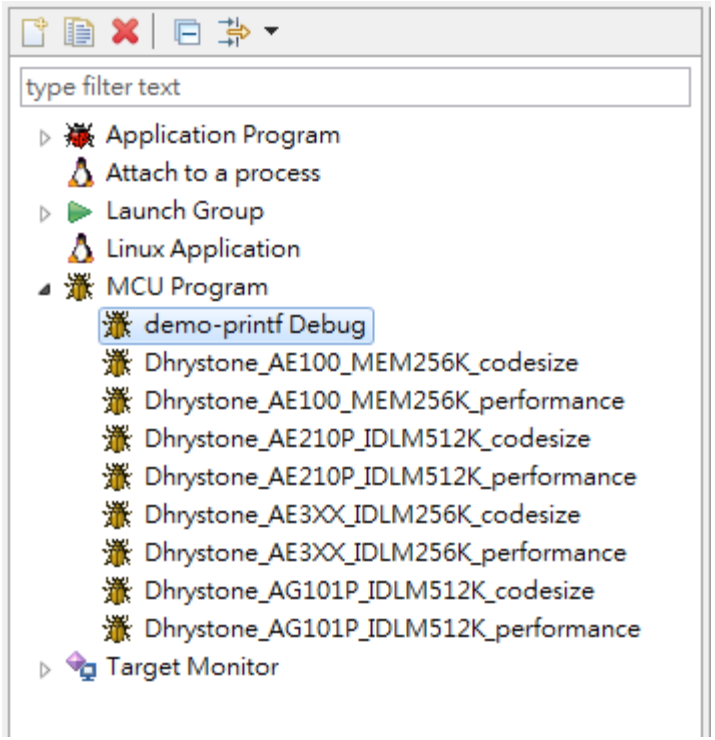
4. Create project

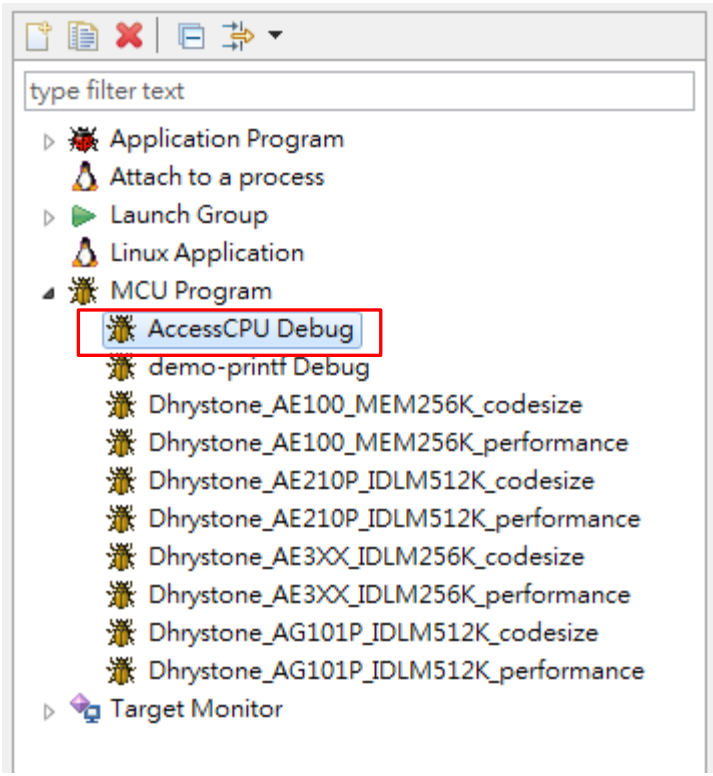
Enter Project Name



MCU Program Settings

- Right click project > Debug As > Debug Configurations > Double click "MCU Program" if AccessCPU project is not in the list





MCU Program Settings

type filter text

- Application Program
- Attach to a process
- Launch Group
- Linux Application
- MCU Program
  - 1 AccessCPU Debug
  - demo-printf Debug
  - Dhrystone_AE100_MEM256K_codesize
  - Dhrystone_AE100_MEM256K_performanc
  - Dhrystone_AE210P_IDLM512K_codesize
  - Dhrystone_AE210P_IDLM512K_performan
  - Dhrystone_AE3XX_IDLM256K_codesize
  - Dhrystone_AE3XX_IDLM256K_performanc
  - Dhrystone_AG101P_IDLM512K_codesize
  - Dhrystone_AG101P_IDLM512K_performan
- Target Monitor

Filter matched 21 of 27 items

Name: AccessCPU Debug

Main | Debugger | 2 Startup | Tracer | Source | Common

1. GDB Initialization Commands

3 Reset and Hold

2. Binary File Options

Load binary file

Use file: Workspace... File System...

Offset (hex):

3. Runtime Options

Set program counter at (hex):

Set breakpoint at:

Resume

4. GDB Run Commands

Revert Apply

4 Debug Close

CPU Registers

Project Expl
- AccessCPU
- D1088

Debug
- AccessCPU Debug [MCU Program]
  - Process
    - Thread #1 (Suspended : User Reque
      - 0x0
    - gdb(7.7.0.20140207)-78

(x)= Variables | Expressions | Memory

Expression	Type	Value
av_reverse		
Add new express		

Tar | Out
- Running Target
  - ADP-AE210P-N820 ICE:
- Targets
- Generic Targets

Andes Project C | 0x0

No source available for "0x0"

View Disassembly...

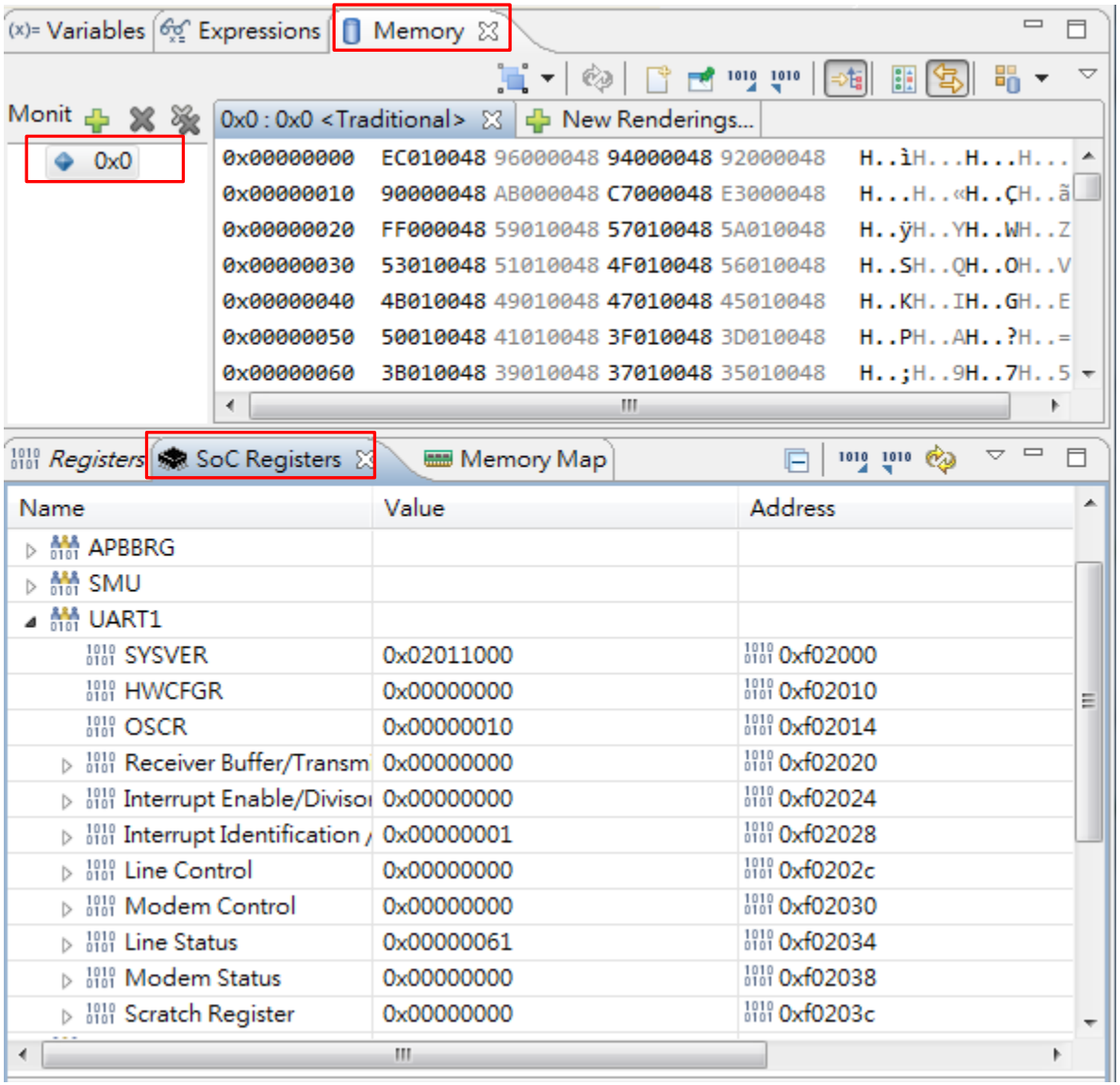
Registers | SoC Registers | Memory Map

Name	Value
All Registers	
General Purpose Registers	
Configuration System Registers	
Interruption System Registers	
MMU System Registers	
mr0 (MMU_CTL)	{0x0, NTC0 = 0x0, NTC1 = 0x
mr6 (ILMB)	{IEN = 0x1, ILMSZ = 0x8, (raw
IBPA	0x0
ILMSZ	0x8 - 1024 kB
IEN	0x1 - Enabled
mr7 (DLMB)	{DEN = 0x0, DLMSZ = 0x7, DE
EDM System Registers	
Performance Monitoring	
Implementation-Dependent Regis	

Target Running

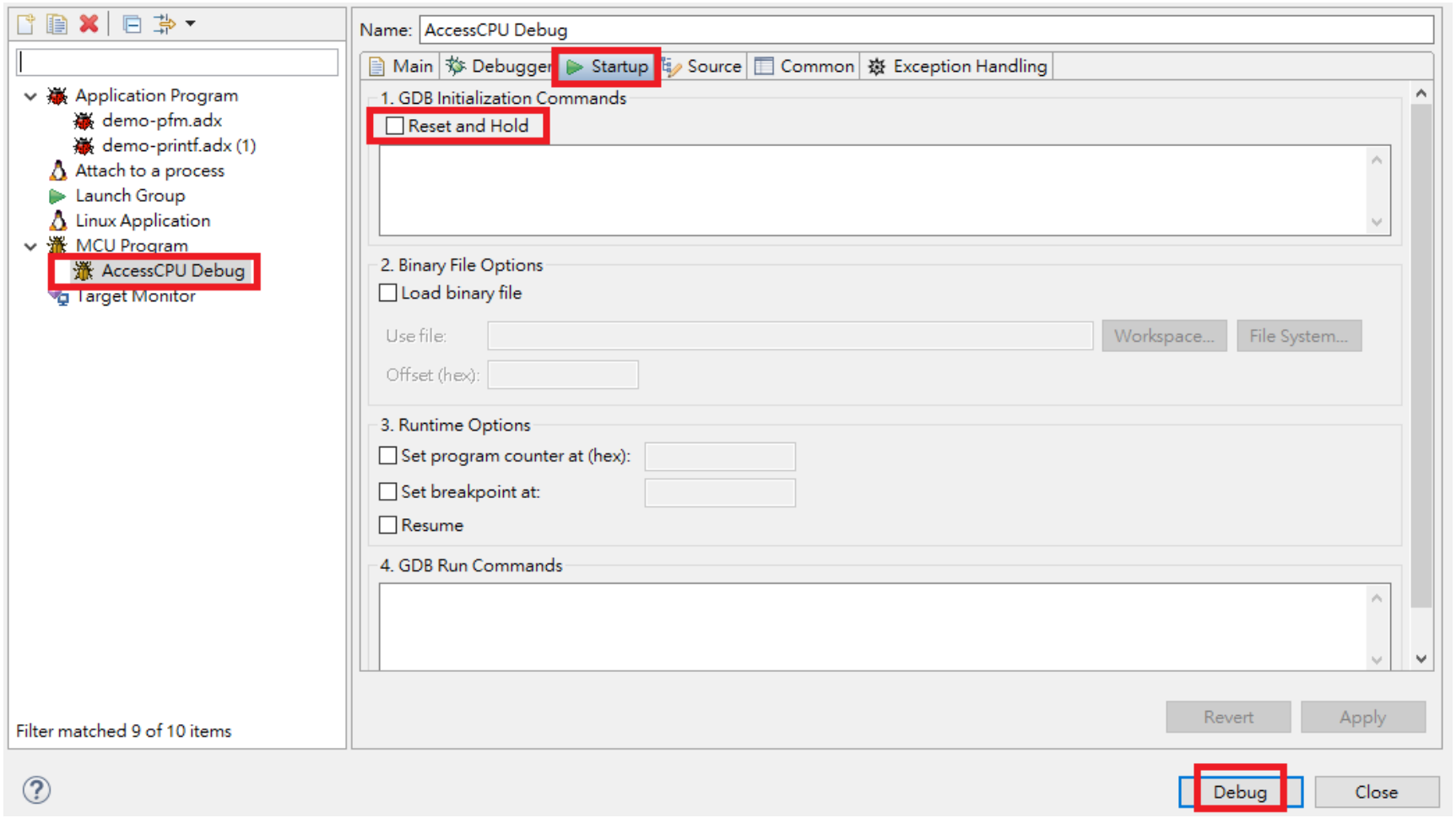
CPU registers

SoC Registers and Memory



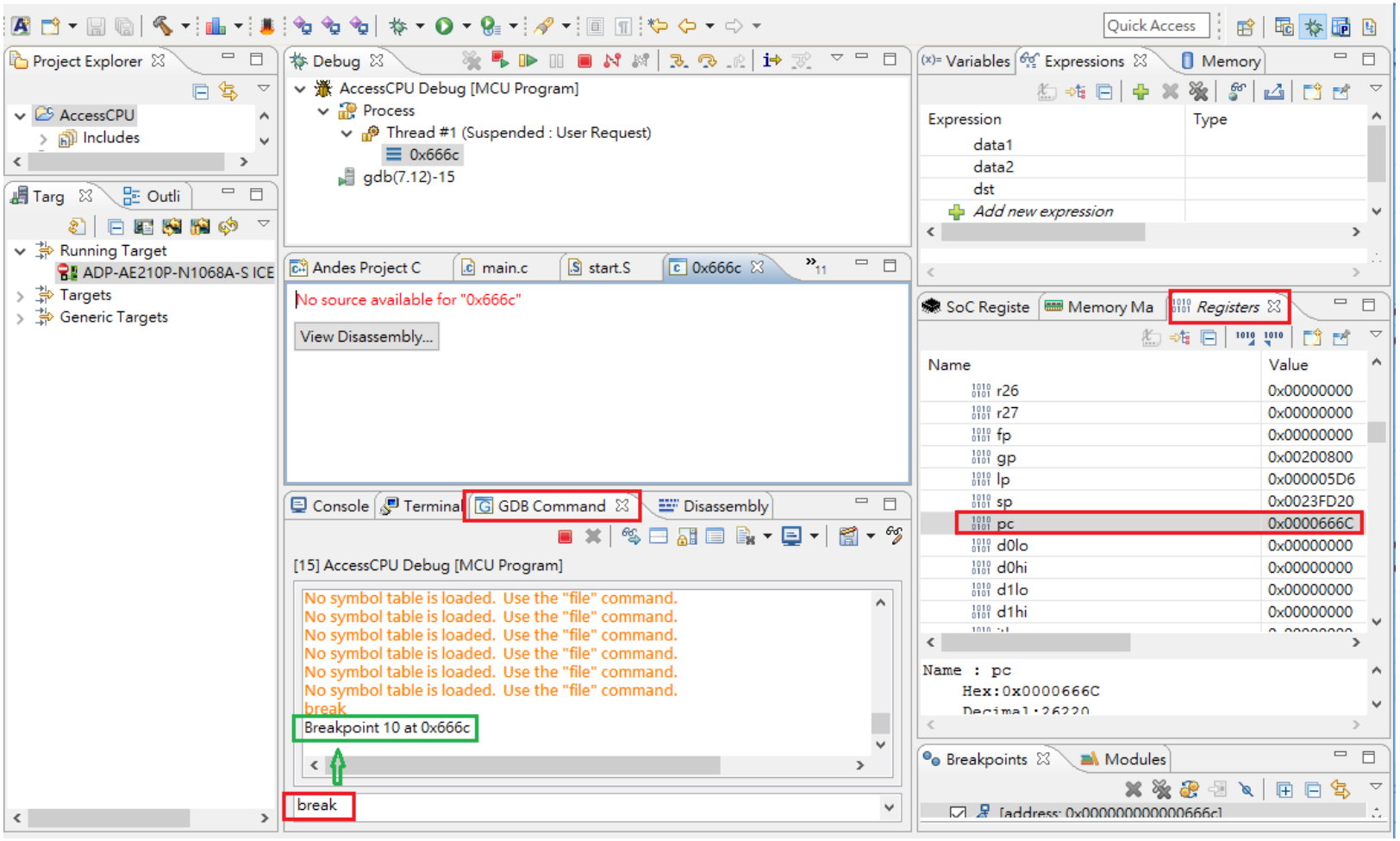
MCU Program without Reset and Hold

- Without "Reset and Hold" to reset the PC to 0



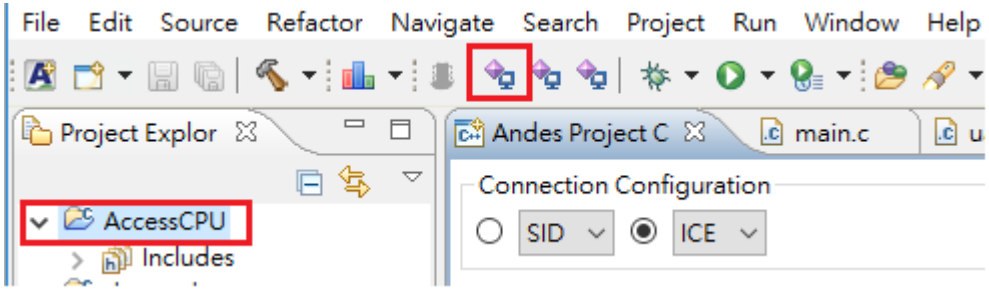
MCU Program without Reset and Hold

- At "GDB command" view, type break to stop CPU
- The PC register at Registers view shows where the CPU stops

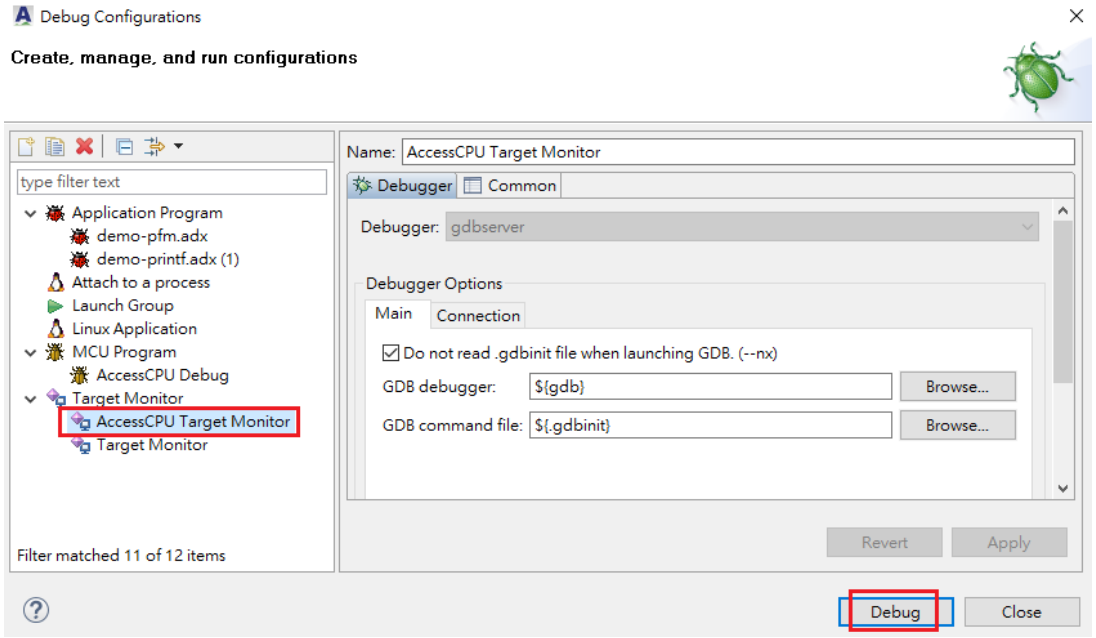


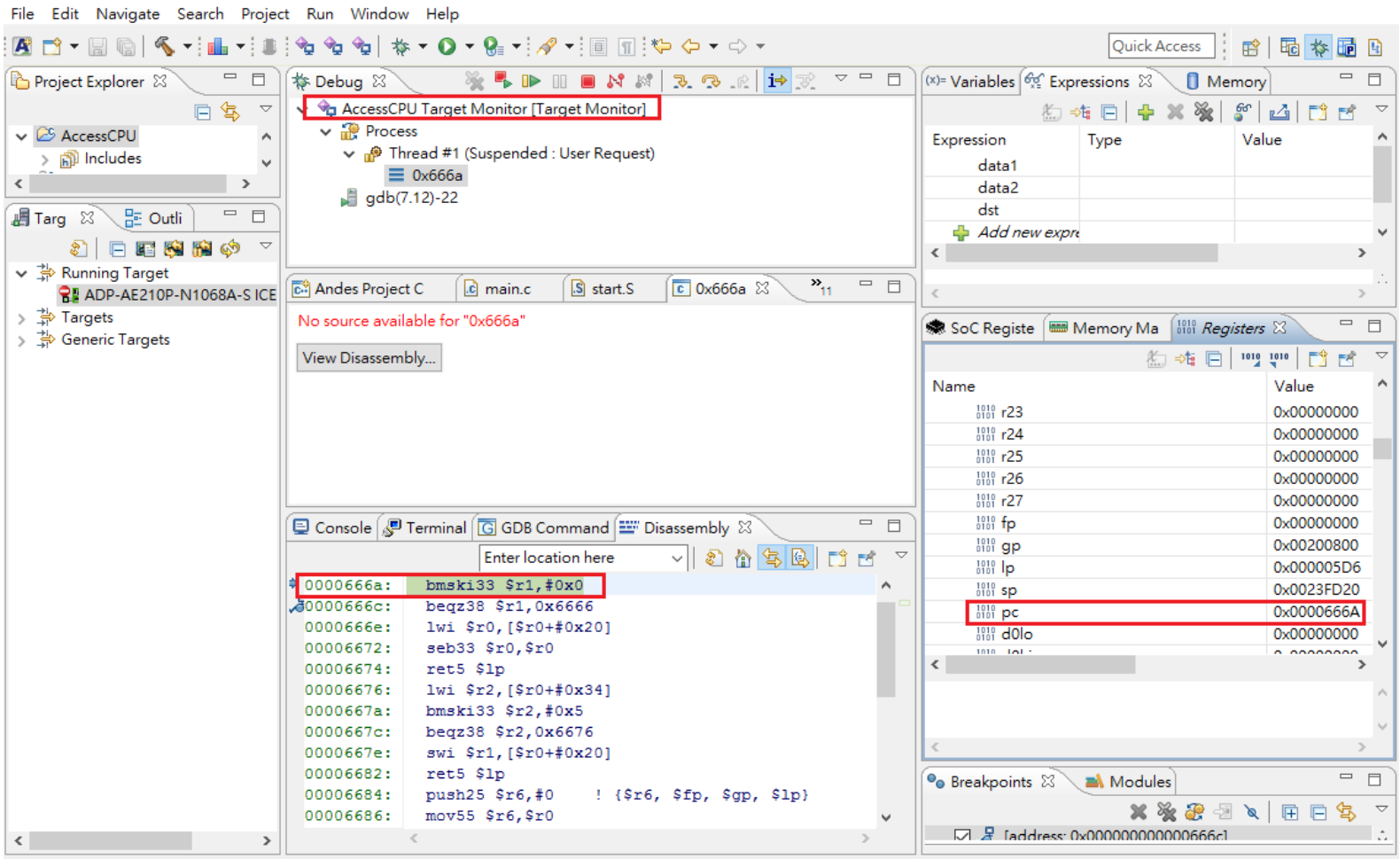
Target Monitor

- Use "Target Monitor" icon



- Use "Debug as" > Debug Configurations > Target Monitor




File Edit Navigate Search Project Run Window Help
Quick Access
Project Explorer
AccessCPU
Includes
Debug
AccessCPU Target Monitor [Target Monitor]
Process
Thread #1 (Suspended : User Request)
0x666a
gdb(7.12)-22
Variables
Expressions
Memory
Expression
Type
Value
data1
data2
dst
Add new expr
Targ
Outli
Running Target
ADP-AE210P-N1068A-S ICE
Targets
Generic Targets
Andes Project C
main.c
start.S
0x666a
No source available for "0x666a"
View Disassembly...
SoC Registe
Memory Ma
Registers
Name
Value
r23 0x00000000
r24 0x00000000
r25 0x00000000
r26 0x00000000
r27 0x00000000
fp 0x00000000
gp 0x00200800
lp 0x000005D6
sp 0x0023FD20
pc 0x0000666A
d0lo 0x00000000
Console
Terminal
GDB Command
Disassembly
Enter location here
0000666a: bmski33 $r1,#0x0
0000666c: beqz38 $r1,0x6666
0000666e: lwi $r0,[$r0+#0x20]
00006672: seb33 $r0,$r0
00006674: ret5 $lp
00006676: lwi $r2,[$r0+#0x34]
0000667a: bmski33 $r2,#0x5
0000667c: beqz38 $r2,0x6676
0000667e: swi $r1,[$r0+#0x20]
00006682: ret5 $lp
00006684: push25 $r6,#0 ! {$r6, $fp, $gp, $lp}
00006686: mov55 $r6,$r0
Breakpoints
Modules