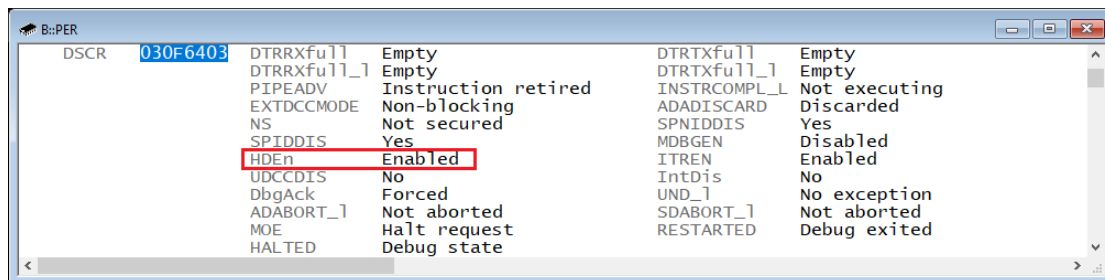


[Knowledgebase](#) > [FAQs by core architecture](#) > [Arm](#) > [\[Arm\] Is there a register on Cortex cores that can be used by the target program to detect if a debugger is connected?](#)

## [Arm] Is there a register on Cortex cores that can be used by the target program to detect if a debugger is connected?

2023-05-22 - [Comments \(0\)](#) - [Arm](#)

There is a register called DBGDSR (Debug Status and Control Register) which can be used for this purpose. In fact, if the debugger connects to the core, it will set the bit 14 "DBGDSR.HDen" ("Halting debug-mode enable" bit). This bit is cleared on reset.



Register	Value	Field	Status
DSR	030F6403		
		DTRRXfull	Empty
		DTRRXfull_1	Empty
		PIPEADV	Instruction retired
		EXTDCCMODE	Non-blocking
		NS	Not secured
		SPIDDIS	Yes
		HDen	Enabled
		UDCCDIS	No
		DbgAck	Forced
		ADABORT_1	Not aborted
		MOE	Halt request
		HALTED	Debug state
		DTRTXfull	Empty
		DTRTXfull_1	Empty
		INSTRCOMPL_1	Not executing
		ADADISCARD	Discarded
		SPNIDDIS	Yes
		MDGEN	Disabled
		ITREN	Enabled
		IntDis	No
		UND_1	No exception
		SDABORT_1	Not aborted
		RESTARTED	Debug exited

### Cortex-M:

For stop mode debugging the debugger usually sets inside the DHCSR the C\_DEBUGEN flag (bit 0). Stop mode debugging is usually the standard way to debug Arm cores using our tools. DHCSR is a debug register inside the SCS block of each Cortex-M core at address 0xE000EDF0. It should be readable by the target application, too.

But please be aware: the flag will be set immediately, when connecting using SYStem.Mode Up/Go/Attach and will be kept set even on SYStem.Down.

- [Tags](#)
- [Arm Cortex](#)