This register selects which of the 32-bit portion of the hart array mask register is accessible in hawindow

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID																					A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
ID	R/W	Field	Value ID	Value																															
A	R	HAWINDOWSEL																																	
				The high bits of this field may be tied to 0, depending on how large the array mask register is. E.g. on a system with 48 harts only bit 0 of this field may actually be writable.																																

8.26.1.16 DEBUGIF.HAWINDOW

Address offset: 0x454

Hart Array Window

This register provides R/W access to a 32-bit portion of the hart array mask register. The position of the window is determined by hawindowsel. I.e. bit 0 refers to hart hawindowsel x 32, while bit 31 refers to hart hawindowsel x 32 + 31. Since some bits in the hart array mask register may be constant 0, some bits in this register may be constant 0, depending on the current value of hawindowsel.

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID				A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
Reset 0x00000000				**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
ID	R/W	Field	Value ID	Value																															
A	RW	MASKDATA																																	
				Mask data.																															

8.26.1.17 DEBUGIF.ABSTRACTCS

Address offset: 0x458

Abstract Control and Status

Bit number				31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID							D	D	D	D	D												C		B	B	B				A	A	A	A
Reset 0x01000002				**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**
ID	R/W	Field	Value ID	Value																															
A	R	DATACOUNT																																	
				Number of data registers that are implemented as part of the abstract command interface. Valid sizes are 1..12.																																	
B	RW	CMDERR																																	
				Command error when the abstract command fails.																																	
			NoError	0																																	
				No error.																																	
			Busy	1																																	
				An abstract command was executing while command, abstractcs, or abstractauto was written, or when one of the data or progbuf registers was read or written. This status is only written if cmderr contains 0																																	
			NotSupported	2																																	
				The requested command is notsupported, regardless of whether the hart is running or not.																																	
			Exception	3																																	
				An exception occurred while executing the command (e.g. while executing theProgram Buffer).																																	
			HaltResume	4																																	
				The abstract command couldn't execute because the hart wasn't in the required state (running/halted). or unavailable.																																	
			Bus	5																																	
				The abstract command failed due to abus error (e.g. alignment, access size, or timeout).																																	
			Other	7																																	
				The command failed for another reason.																																	
C	R	BUSY																																	
				Abstract command execution status.																																	