Eventually, the register window overflows when it rotates to a position in the AR register file that already holds the contents of a previous window. This overflow generates a window-overflow exception, and the exception handler spills the contents of the older window to a preallocated spill area on the stack. When returning from a function call and rotating back to a window that has been spilled, a window-underflow exception occurs. The underflow exception handler copies the register values from the stack back into the register file.

Only the general-purpose AR register file employs register windows. Additional register files added through Xtensa LX configuration options and register files created through Tensilica's TIE extension language do not employ register windowing.

In addition to the general-purpose AR register file, the Xtensa LX processor has a number of other architectural and special-purpose registers, listed in Table 5. Special registers are accessed with dedicated "read special register" (RSR), "write special register" (WSR), and "exchange special register" (XSR) instructions. Coprocessor registers and user registers (created with the TIE processor-extension language) are accessed with dedicated "read user register" (RUR), "write user register" (WUR) instructions. Note that special registers and coprocessor/user registers are numbered and these numbers may overlap in some cases, although this overlap does not cause a problem because these two sets of registers are accessed with different instructions.

Table 5. Alphabetical List of Xtensa LX Registers

Name[1]	Description	Package (Core or Option)	Special Register Number
ACCHI	Accumulator high bits	MAC16 option	17
ACCLO	Accumulator low	MAC16 option	16
AR	General-purpose register file	Core architecture	n/a
BR	Boolean registers	Boolean option	4
CCOMPARE0..3	Cycle number to generate interrupt	Timer Interrupt option	240-243
CCOUNT	Cycle count	Timer Interrupt option	234
CPENABLE	Coprocessor enable bits	Coprocessor option	224
DBREAKA0..15	Data break address	Debug option	144-159
DBREAKC0..15	Data break control	Debug option	160-175
DEBUGCAUSE	Cause of last debug exception	Debug option	233
DDR	Debug data register	Debug option	104
DEPC	Double exception PC	Exception option	192
EPC1	Level-1 exception PC	Exception option	177
EPC2..15	High-level exception PC	High Priority Interrupt option	178-191
EPS2..15	High-level exception PS	High Priority Interrupt option	194-207

Table 5. Alphabetical List of Xtensa LX Registers (continued)

Name[1]	Description	Package (Core or Option)	Special Register Number
EXCCAUSE	Cause of last exception/level-1 interrupt	Exception option	232
EXCSAVE1	Level-1 exception save location	Exception option	209
EXCSAVE2..15	High-level exception save location	High Priority Interrupt option	210-223
EXCVADDR	Exception virtual address	Exception option	238
FCR	Floating-point control register	Floating-Point Coprocessor option	User[2] 232
FR	Floating-point registers	Floating-Point Coprocessor option	n/a
FSR	Floating-point status register	Floating-Point Coprocessor option	User[2] 233
IBREAKA0..15	Instruction break address	Debug option	128-143
IBREAKENABLE	Instruction break enable bits	Debug option	96
ICOUNT	Instruction count	Debug option	236
ICOUNTLEVEL	Instruction count level	Debug option	237
INTENABLE	Interrupt enable bits	Interrupt option	228
INTCLEAR	Clear request bits in INTERRUPT	Interrupt option	227
INTERRUPT	Read interrupt request bits[3]	Interrupt option	226
INTSET	Set request bits in INTERRUPT[3]	Interrupt option	226
LBEG	Loop-begin address	Loop option	0
LCOUNT	Loop count	Loop option	2
LEND	Loop-end address	Loop option	1
LITBASE	Literal base	Extended L32R option	5
MISC0..3	Misc register 0-3	Miscellaneous Special Registers option	244-247
M0..3	MAC16 data registers	MAC16 option	32-35
PC	Program counter	Core Architecture	n/a
PRID	Processor ID	Processor ID option	235
PS	Processor state	Various	230
SAR	Shift-amount register	Core Architecture	3
SAV	Shift-amount valid	Speculation option	11
SCOMPARE1	Expected data value for S32C1I	Multiprocessor Synchronization option	12
WindowBase	Base of current AR window	Windowed Register option	72
WindowStart	Call-window start bits	Windowed Register option	73

1 Used in RSR, WSR, and XSR instructions.

2 FCR and FSR are User Registers where most are System Registers. These names are used in RUR and WUR instructions.

3 The INTERRUPT and INTSET registers share special register number 226. An RSR instruction directed at register 226 reads the INTERRUPT register, and a WSR instruction directed at register 226 writes to the INTSET register. Do not use the XSR instruction with special register 226.