

Instruction class	Instruction name(s)	Function
Data transfer	Load/Store Non-temporal pair	Loads/stores a pair of registers with an indication not to cache the data. Base + scaled offset addressing mode only.
ALU	Add Extended word/double word	Add 2 registers after left shifting the second register operand and extending it.
	Add with shift; add immediate with shift	Adds with shift of the second operand.
	Address of page	Computes the address of a page based on PC (similar to ADDUIPC, which is the same as ADR in ARMv8)
	AND, OR, XOR, XOR NOT shifted register	Logical operation on a register and a shifted register.
	Bit field clear shifted	Shift operand, invert and AND with another operand
	Conditional compare, immediate, negative, negative immediate	If condition true, then set condition flags to compare result, otherwise leave condition flags untouched.
	Conditional increment, invert, negate	If condition then set destination to increment/invert/negate of source register
	CRC	Computes a CRC checksum: byte, word, halfword, double
	Multiply add, subtract	Integer multiply-add or multiply-subtract
	Multiply negate	Negate the product of two integers; word & double word
	Move immediate or inverse	Replace 16-bits in a register with immediate, possibly shifted
	Reverse bit order	Reverses the order of bits in a register
	Signed bit field move	Move a signed bit field; sign extend to left; zero extend to right
	Unsigned divide, multiple, multiply negate, multiply-add, multiply-sub	Unsigned versions of the basic instructions
Control transfer	CBNZ, CBZ	Compare branch $\neq 0$ , indicating this is not a call or return.
	TBNZ, TBZ	Tests bit in a register $\neq 0$ , and branch.

**FIGURE E.22 Additional instructions provided in ARMv8, the AArch64 instruction set.** Unless noted the instruction is available in a word and double word format, if there is a difference. Most of the ALU instructions can optionally set the condition codes; these are not included as separate instructions here or in earlier tables.