Condition codes:

Condition Code	Action	Condition Code	Action
HS	Unsigned higher or same	**GE**	Signed Greater than or equal
LO	Unsigned lower	**LT**	Signed Less than
HI	Unsigned higher	**GT**	Signed Greater than
LS	Unsigned lower or same	**LE**	Signed Less than or equal
EQ	Equal	**NE**	Not equal

Note: Use the above codes with the instructions to do conditional execution of instructions. e.g., 'BEQ' is 'Branch if equal'.

Instructions:

B *label*	Branch to *label*
BL *label*	Branch to *label*, store current PC in LR register
BX Rm	Branch to address in *Rm*

ADD Rd, Rn, op2	Rd = Rn + op2	RSB Rd, Rn, op2	Rd = op2 - Rn
ADC Rd, Rn, op2	Rd = Rn + op2 + carry	RSC Rd, Rn, op2	RD = op2 – Rn – !carry
SUB Rd, Rn, op2	Rd = Rn – op2	MUL Rd, Rn, Rm	Rd = Rn * Rm
SBC Rd, Rn, op2	Rd = Rn – op2 – !carry	UMULL RdLo, RdHi, Rn, Rd	RdHi:RdLo = Rn * Rd

MOV Rd, op2	Rd = op2	MOV Rd, Rn, shift	For example: Mov R0, R1, LSL #2
LDR Rd, [Rn]	Rd = *Rn	STR Rd, [Rx]	*Rx = Rd
LDRH Rd, [Rn]	Load 2 bytes from *Rn	STRH Rd, [Rx]	Store 2 bytes from Rd to &Rx
LDRB Rd, [Rn]	Load byte from *Rn	STRB Rd, [Rx]	Store byte from Rd to &Rx
LDRSH Rd, [Rn]	Load signed 2 bytes from *Rn	LDRSB Rd, [Rx]	Load signed byte from *Rn

AND Rd, Rn, op2	Rd = Rn & op2	EOR Rd, Rn, op2	Rd = Rn (xor) op2
ORR Rd, Rn, op2	Rd = Rn | Op2	MVN Rd, op2	Rd = !op2

ASR Rd, Rn, Rs	Arithmetic shift right		
LSR Rd, Rn, Rs	Logical shirt right	LSL Rd, Rn, Rs	Logical shift left
ROR Rd, Rn, Rs	Rotate right	RRX Rd, Rn	Rotate right through carry

PUSH {reglist}	Push list of registers to stack	POP {reglist}	Pop from the stack to the list of registers