Opcode	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
AND<cond><S> Rd, Rn, Rm OP #					0	0	0	0	0	0	0	S	R				R				#						0	R			
AND<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	0	0	0	S	R				R				R			0			1	R			
MUL<cond><S> Rd, Rm, Rs					0	0	0	0	0	0	0	S	R				SB				R			1	0	0	1	R			
STR<cond>H Rd, <address>					0	0	0	P	U	I	W	0	R				R							1	0	1	1				
LDR<cond>H Rd, <address>					0	0	0	P	U	I	W	1	R				R							1	0	1	1				
Undefined Instruction					0	0	0					0												1	1	0	1				
LDR<cond>SB Rd, <address>					0	0	0	P	U	I	W	1	R				R							1	1	0	1				
Undefined Instruction					0	0	0					0										[9]		1	[9]	1	1				
LDR<cond>SH Rd, <address>					0	0	0	P	U	I	W	1	R				R							1	[9]	1	1				
EOR<cond><S> Rd, Rn, Rm OP #					0	0	0	0	0	0	1	S	R				R				#						0	R			
EOR<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	0	0	1	S	R				R				R			0			1	R			
MLA<cond><S> Rd, Rm, Rs, Rn					0	0	0	0	0	0	1	S	R				R				R			1	0	0	1	R			
SUB<cond><S> Rd, Rn, Rm OP #					0	0	0	0	0	1	0	S	R				R				#						0	R			
SUB<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	0	1	0	S	R				R				R			0			1	R			
RSB<cond><S> Rd, Rn, Rm OP #					0	0	0	0	0	1	1	S	R				R				#						0	R			
RSB<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	0	1	1	S	R				R				R			0			1	R			
ADD<cond><S> Rd, Rn, Rm OP #					0	0	0	0	1	0	0	S	R				R				[9]	#					[9]	0	R			
ADD<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	1	0	0	S	R				R				R			0			[9]	1	R			
UMULL<cond><S> RdLo, RdHi, Rm, Rs					0	0	0	0	1	0	0	S	R	H			R	L			R			1	0	0	1	R			
ADC<cond><S> Rd, Rn, Rm OP #					0	0	0	0	1	0	1	S	R				R				[9]	#					[9]	0	R			
ADC<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	1	0	1	S	R				R				R			0			[9]	1	R			
UMLAL<cond><S> RdLo, RdHi, Rm, Rs					0	0	0	0	1	0	1	S	R	H			R	L			R			1	[9]	0	1	R			
SBC<cond><S> Rd, Rn, Rm OP #					0	0	0	0	1	1	0	S	R				R				[9]	#					[9]	0	R			
SBC<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	1	1	0	S	R				R				R			0			[9]	1	R			
SMULL<cond><S> RdLo, RdHi, Rm, Rs					0	0	0	0	1	1	0	S	R	H			R	L			R			1	[9]	0	1	R			
RSC<cond><S> Rd, Rn, Rm OP #					0	0	0	0	1	1	1	S	R				R				#						0	R			
RSC<cond><S> Rd, Rn, Rm OP Rs					0	0	0	0	1	1	1	S	R				R				R			0			[9]	1	R			
SMLAL<cond><S> RdLo, RdHi, Rm, Rs					0	0	0	0	1	1	1	S	R	H			R	L			R			1	[9]	0	1	R			
TST<cond> Rn, Rm OP #					0	0	0	1	0	0	0	1	R				SB				#						0	R			
TST<cond> Rn, Rm OP Rs					0	0	0	1	0	0	0	1	R				SB				R			0	[9]		1	R			
MRS<cond> Rd, CPSR					0	0	0	1	0	0	0	0	SBO				R				SB											
SWP<cond> Rd, Rm, [Rn]					0	0	0	1	0	0	SB		R				R				SB			1	0	0	1	R			
TEQ<cond> Rn, Rm OP #					0	0	0	1	0	0	1	1	R				SB				[9]	#					[9]	0	R			
TEQ<cond> Rn, Rm OP Rs					0	0	0	1	0	0	1	1	R				SB				R			0	[9]		1	R			
MSR<cond> CPSR_<fields>, Rm					0	0	0	1	0	0	1	0					SBO				SB						0	R			
BX<cond> Rm					0	0	0	1	0	0	1	0	SBO				SBO				SBO			0	0	[9]	0	1	R			
CMP<cond> Rn, Rm OP #					0	0	0	1	0	1	0	1	R				SB				#					[9]	0	R			
CMP<cond> Rn, Rm OP Rs					0	0	0	1	0	1	0	1	R				SB				R			0			1	R			
MRS<cond> Rd, SPSR					0	0	0	1	0	1	0	0	SBO				R				SB											
SWP<cond>B Rd, Rm, [Rn]					0	0	0	1	0	1	SB		R				R				SB			1	0	0	1	R			
CMN<cond> Rn, Rm OP #					0	0	0	1	0	1	1	1	R				SB				#					[9]	0	R			
CMN<cond> Rn, Rm OP Rs					0	0	0	1	0	1	1	1	R				SB				R			0			1	R			
MSR<cond> SPSR_<fields>, Rm					0	0	0	1	0	1	1	0					SBO				SB						0	R			
ORR<cond><S> Rd, Rn, Rm OP #					0	0	0	1	1	0	0	S	R				R				[9]	#					[9]	0	R			
ORR<cond><S> Rd, Rn, Rm OP Rs					0	0	0	1	1	0	0	S	R				R				R			0	[9]		1	R			
MOV<cond><S> Rd, Rm OP #					0	0	0	1	1	0	1	S	SB				R				#						0	R			
MOV<cond><S> Rd, Rm OP Rs					0	0	0	1	1	0	1	S	SB				R				R			0			1	R			
BIC<cond><S> Rd, Rn, Rm OP #					0	0	0	1	1	1	0	S	R				R				[9]	#					[9]	0	R			
BIC<cond><S> Rd, Rn, Rm OP Rs					0	0	0	1	1	1	0	S	R				R				R			0	[9]		1	R			
MVN<cond><S> Rd, Rm OP #					0	0	0	1	1	1	1	S	SB				R				#						0	R			
MVN<cond><S> Rd, Rm OP Rs					0	0	0	1	1	1	1	S	SB				R				R			0	[9]		1	R			
AND<cond><S> Rd, Rn, #					0	0	1	0	0	0	0	S	R				R				[9]	#										
EOR<cond><S> Rd, Rn, #					0	0	1	0	0	0	1	S	R				R				[9]	#										
SUB<cond><S> Rd, Rn, #					0	0	1	0	0	1	0	S	R				R				[9]	#										
RSB<cond><S> Rd, Rn, #					0	0	1	0	0	1	1	S	R				R				[9]	#										
ADD<cond><S> Rd, Rn, #					0	0	1	0	1	0	0	S	R				R				[9]	#										
ADC<cond><S> Rd, Rn, #					0	0	1	0	1	0	1	S	R				R				[9]	#										
SBC<cond><S> Rd, Rn, #					0	0	1	0	1	1	0	S	R				R				[9]	#										
RSC<cond><S> Rd, Rn, #					0	0	1	0	1	1	1	S	R				R				[9]	#										
TST<cond> Rn, #					0	0	1	1	0	0	0	1	R				SB					#										
TEQ<cond> Rn, #					0	0	1	1	0	0	1	1	R				SB					#										
MSR<cond> CPSR_f, #					0	0	1	1	0	0	1	0					SBO					#										
CMP<cond> Rn, #					0	0	1	1	0	1	0	1	R				SB					#										
CMN<cond> Rn, #					0	0	1	1	0	1	1	1	R				SB					#										
MSR<cond> SPSR_f, #					0	0	1	1	0	1	1	0					SBO					#										
ORR<cond><S> Rd, Rn, #					0	0	1	1	1	0	0	S	R				R					#										
MOV<cond><S> Rd, #					0	0	1	1	1	0	1	S	SB				R					#										
BIC<cond><S> Rd, Rn, #					0	0	1	1	1	1	0	S	R				R					#										
MVN<cond><S> Rd, #					0	0	1	1	1	1	1	S	SB				R					#										
STR<cond> Rd, Rn, #					0	1	0	P	U	0	W	0	R				R					#										
LDR<cond> Rd, Rn, #					0	1	0	P	U	0	W	1	R				R					#										
STR<cond>B Rd, Rn, #					0	1	0	P	U	1	W	0	R				R					#										
LDR<cond>B Rd, Rn, #					0	1	0	P	U	1	W	1	R				R					#										
STR<cond>T Rd, Rn, #					0	1	0	0	U	0	1	0	R				R					#										
LDR<cond>T Rd, Rn, #					0	1	0	0	U	0	1	1	R				R					#										
STR<cond>BT Rd, Rn, #					0	1	0	0	U	1	1	0	R				R					#										
LDR<cond>BT Rd, Rn, #					0	1	0	0	U	1	1	1	R				R					#										
STR<cond> Rd, Rn, #					0	1	1	P	U	0	W	0	R				R				#						0	R			
LDR<cond> Rd, Rn, #					0	1	1	P	U	0	W	1	R				R				#						0	R			
STR<cond>B Rd, Rn, #					0	1	1	P	U	1	W	0	R				R				#						0	R			
LDR<cond>B Rd, Rn, #					0	1	1	P	U	1	W	1	R				R				#						0	R			
STR<cond>T Rd, Rn, #					0	1	1	0	U	0	1	0	R				R				#						0	R			
LDR<cond>T Rd, Rn, #					0	1	1	0	U	0	1	1	R				R				#						0	R			
STR<cond>BT Rd, Rn, #					0	1	1	0	U	1	1	0	R				R				#						0	R			
LDR<cond>BT Rd, Rn, #					0	1	1	0	U	1	1	1	R				R				[9]	#			[9]		0	R			
Undefined Instruction					0	1	1					1										[9]			[9]		1				
STM<cond><addrmode> Rm<!>, reg list					1	0	0	P	U	0	W	0	R				[9]					[9]										
LDM<cond><addrmode> Rm<!>, reg list					1	0	0	P	U	0	W	1	R				[9]					[9]										
STM<cond><addrmode> Rm, reg list^					1	0	0	P	U	1	0	0	R				0				[9]											
UNPREDICTABLE					1	0	0			1	1	0									[9]											
LDM<cond><addrmode> Rm, reg list^					1	0	0	P	U	1	0	1	R				0				[9]											
UNPREDICTABLE					1	0	0			1	1	1									[9]											
STM<cond><addrmode> Rm<!>, reg list^					1	0	0	P	U	1	W	0	R				1															
LDM<cond><addrmode> Rm<!>, reg list^					1	0	0	P	U	1	W	1	R				1															
B<cond> <target addr>					1	0	1	0	24																							
BL<cond> <target addr>					1	0	1	1	24																							
STC<cond> p<cp_num>, CRd, #					1	1	0	P	U	N	W	0	R				CR					#										
LDC<cond> p<cp_num>, CRd, #					1	1	0	P	U	N	W	1	R				CR					#										
CDP<cond> p<cp#>,<o1>,CRd,CRn,CRm,<o2>					1	1	1	0	1				CR				CR								2		0	CR			
MCR<cond> p<cp#>,<o1>,Rd,CRn,CRm,<o2>					1	1	1	0	1		0		CR				R								2		1	CR			
MRC<cond> p<cp#>,<o1>,Rd,CRn,CRm,<o2>					1	1	1	0	1		1		CR				R								2		1	CR			
SWI <swi_number>					1	1	1	1																							

M	M	O	S F
E	EQ	0 0 0 0	= 1
N E	NE	0 0 0 1	= 0
C S	CS	0 0 1 0	C = 1
C C	CC	0 0 1 1	C = 0
U H S	HS	0 0 1 0	C = 1
U L	LO	0 0 1 1	C = 0
M /N	MI	0 1 0 0	N = 1
P /P	PL	0 1 0 1	N = 0
O	VS	0 1 1 0	V = 1
N O	VC	0 1 1 1	V = 0
U H	HI	1 0 0 0	C = 1, = 0
U L [9] S	LS	1 0 0 1	C = 0, = 1
S G E	GE	1 0 1 0	N = V
S L	LT	1 0 1 1	N != V
S G	GT	1 1 0 0	= 0, N = V
S L E	LE	1 1 0 1	= 1, N != V
A	AL	1 1 1 0	-
N [9]	NE	1 1 1 1	-

Opcode	Notes	Z	C	N	V
ADC<cond><S> Rd, Rn, <sh_op>	-	s	s	s	s
ADD<cond><S> Rd, Rn, <sh_op>	-	s	s	s	s
AND<cond><S> Rd, Rn, <sh_op>	-	s	s	s	
B<cond> <target_addr>	-				
BIC<cond><S> Rd, Rn, <sh_op>	-	s	s	s	
BL<cond> <target_addr>	-				
BX<cond> Rm	-				
CDP<cond> p<cp#>,o1,CRd,CRn,CRm,o2	-				
CMN<cond> Rn, <sh_op>	-	x	x	x	x
CMP<cond> Rn, <sh_op>	-	x	x	x	x
EOR<cond><S> Rd, Rn, <sh_op>	-	s	s	s	
LDC<cond> p<cp_num>, CRd, #	-				
LDM<cond><adm> Rm, {reg list}^	-				
LDM<cond><adm> Rm<!>, {reg list}	"!" K R V				
LDM<cond><adm> Rm<!>, {reg list}^	-				
LDR<cond> Rd, Rn, #					
LDR<cond>B Rd, Rn, #					
LDR<cond>BT Rd, Rn, #					
LDR<cond>H Rd, <address>					
LDR<cond>SB Rd, <address>					
LDR<cond>SH Rd, <address>					
LDR<cond>T Rd, Rn, #					
MCR<cond> p<cp#>,o1,Rd,CRn,CRm,o2	-				
MLA<cond><S> Rd, Rm, Rs, Rn	-	s	s	s	
MOV<cond><S> Rd, <sh_op>	-	s	s	s	
MRC<cond> p<cp#>,o1,Rd,CRn,CRm,o2	R ≙ 15	*	*	*	*
MRS<cond> Rd, CPSR	[9]-				
MRS<cond> Rd, SPSR	-				
MSR<cond> CPSR_<fields>, Rm	-				
MSR<cond> CPSR_f, #	-				
MSR<cond> SPSR_<fields>, Rm	-				
MSR<cond> SPSR_f, #	-[9]				
MUL<cond><S> Rd, Rm, Rs	-	s	s	s	
MVN<cond><S> Rd, <sh_op>	-	s	s	s	
ORR<cond><S> Rd, Rn, <sh_op>	-	s	s	s	
RSB<cond><S> Rd, Rn, <sh_op>	-	s	s	s	s
RSC<cond><S> Rd, Rn, <sh_op>	-	s	s	s	s
SBC<cond><S> Rd, Rn, <sh_op>	-	s	s	s	s
SMLAL<cond><S> RdLo, RdHi, Rm, Rs	64	s	s	s	s
SMULL<cond><S> RdLo, RdHi, Rm, Rs	64	s	s	s	s
STC<cond> p<cp_num>, CRd, #	-				
STM<cond><adm> Rm, {reg list}^	-				
STM<cond><adm> Rm<!>, {reg list}	"!" K R V				
STM<cond><adm> Rm<!>, {reg list}^	-				
STR<cond> Rd, Rn, #					
STR<cond>B Rd, Rn, #					
STR<cond>BT Rd, Rn, #					
STR<cond>H Rd, <address>					
STR<cond>T Rd, Rn, #					
SUB<cond><S> Rd, Rn, <sh_op>	-	s	s	s	s
SWI <swi_number>	-				
SWP<cond> Rd, Rm, [Rn]	-				
SWP<cond>B Rd, Rm, [Rn]	-				
TEQ<cond> Rn, <sh_op>	-	x	x	x	
TST<cond> Rn, <sh_op>	[9] -	x	x	x	
UMLAL<cond><S> RdLo, RdHi, Rm, Rs	64	s	s	s	s
UMULL<cond><S> RdLo, RdHi, Rm, Rs	64	s	s	s	s

Data Processing Opcode
Load/Store Opcode
Branching Opcode
Multiplication Opcode
Other Opcodes
CoProcessor Opcodes

Flag Settings
s - if flag set
x - always
* - special

Opcode	Operation
C<cond><S> Rd, Rn, <sh_op>	R = R + < > + C
D<cond><S> Rd, Rn, <sh_op>	R = R + < >
D<cond><S> Rd, Rn, <sh_op>	R = R & < >
cond> <target_addr>	PC = PC + < >
C<cond><S> Rd, Rn, <sh_op>	R = R & !< >
<cond> <target_addr>	LR = PC+4; PC = PC + < >
<cond> Rm	PC = R ; M =THUMB
P<cond> p<cp#>,o1,CRd,CRn,CRm,o2	
N<cond> Rn, <sh_op>	< > = R + < >
P<cond> Rn, <sh_op>	< > = R - < >
R<cond><S> Rd, Rn, <sh_op>	R = R < >
C<cond> p<cp_num>, CRd, #	#
M<cond><adm> Rm, {reg list}^	*special, see doc*
M<cond><adm> Rm<!>, {reg list}	< > = R +=4
M<cond><adm> Rm<!>, {reg list}^	*special, see doc*
R<cond>B Rd, Rn, #	R = R + #
R<cond>BT Rd, Rn, #	R = R + #
R<cond>H Rd, <address>	R = ⁹
R<cond>SB Rd, <address>	R =
R<cond>SH Rd, <address>	R = ⁹
R<cond>T Rd, Rn, #	R = R + #
R<cond> p<cp#>,o1,Rd,CRn,CRm,o2	- ARM
A<cond><S> Rd, Rm, Rs, Rn	R = R * R + R
V<cond><S> Rd, <sh_op>	R = < >
C<cond> p<cp#>,o1,Rd,CRn,CRm,o2	ARM -
S<cond> Rd, SPSR	R = SPSR
R<cond> CPSR_<fields>, Rm	CPSR = R ( )
R<cond> CPSR_f, #	CPSR = # ( )
R<cond> SPSR_<fields>, Rm	SPSR = R ( )
R<cond> SPSR_f, #	SPSR = # ( )
L<cond><S> Rd, Rm, Rs	R = R * R
N<cond><S> Rd, <sh_op>	R = -< >
R<cond><S> Rd, Rn, <sh_op>	R = R < >
B<cond><S> Rd, Rn, <sh_op>	R = < > - R
C<cond><S> Rd, Rn, <sh_op>	R = < > - R + C
C<cond><S> Rd, Rn, <sh_op>	R = R - < > + C
LAL<cond><S> RdLo, RdHi, Rm, Rs	R H..R L = R *R +(R H..R L)
ULL<cond><S> RdLo, RdHi, Rm, Rs	R H..R L = R *R
C<cond> p<cp_num>, CRd, #	S R #
M<cond><adm> Rm, {reg list}^	*special, see doc*
M<cond><adm> Rm<!>, {reg list}	R +=4 = < >
M<cond><adm> Rm<!>, {reg list}^	*special, see doc*
R<cond> Rd, Rn, #	R + # = R
R<cond>B Rd, Rn, #	R + # = R
R<cond>BT Rd, Rn, #	R + # = R
R<cond>H Rd, <address>	= R ⁹
R<cond>T Rd, Rn, #	R + # = R