Instruction Set Break Down Version 1 (Choice 1)

					Immediate Value				
Instruction	**COND**	**Op Code**	**S bit**	**Dest Register**	**Source-1**	**Source-2**	**Shift/ROR bits**	**Part of Immediate**	**Shift/ROR Cmd**
MOV R0,#7	0000	0110	0	0000	0000000000000111				000
MOV R1,#2	0000	0110	0	0001	0000000000000010				000
SUBS R2,R0,R1 LSL#2	0000	0001	1	0010	0000	0001	00010	000	010
CMP R1,R0	0000	1000	0	0000	0001	0000	00000	000	000
ANDLT R3, R1, R0 LSR#3	0011	0100	0	0011	0001	0000	00011	000	001
EORGT R4,R0,R2	0010	0101	0	0100	0000	0010	00000	000	000
LDR R5,[R3]	0000	1001	0	0101	0011	0000	00000	000	000
STR R0,[R3]	0000	1010	0	0000	0011	0000	00000	000	000

Result after each instruction:

R0=7

R1=2

R2=-1, NZCV=1000

CMP→LT

R3=0

NOP (Condition not satisfied)

R5=32’b00000110000000000000000000111000 = $(100663352)_{10}$

Output data file: First line contains #7

Instruction Set Break Down Version 2 (Choice 1)

					Immediate Value				
Instruction	**COND**	**Op Code**	**S bit**	**Dest Register**	**Source-1**	**Source-2**	**Shift/ROR bits**	**Part of Immediate**	**Shift/ROR Cmd**
MOV R6,#9	0000	0110	0	0110	0000000000001001				000
MOV R7,#9	0000	0110	0	0111	0000000000001001				000
SUBS R8,R7,R6	0000	0001	1	1000	0111	0110	00000	000	000
EOR R9,R7,R6	0000	0101	0	1001	0111	0110	00000	000	000
CMP R9,R8	0000	1000	0	0000	1001	1000	00000	000	000
ADDEQ R10,R9,R8	0001	0000	0	1010	1001	1000	00000	000	000
EOR R11,R7,R6 ROR#2	0000	0101	0	1011	0111	0110	00010	000	011
LDR R12,[R8]	0000	1001	0	1100	1000	0000	00000	000	000
STR R11,[R9]	0000	1010	0	0000	1001	1011	00000	000	000

Result after each instruction:

R6=9

R7=9

R8=0, NZCV=0100

R9=0

CMP→EQ

R10=0

R11=32’b01000000000000000000000000001011 = $(1073741835)_{10}$

R12=00000110001100000000000001001000 = $(103809096)_{10}$

Output data file: First line containing #1073741835

Instruction Set Break Down Version 3 (Choice 1)

					Immediate Value				
Instruction	**COND**	**Op Code**	**S bit**	**Dest Register**	**Source-1**	**Source-2**	**Shift/ROR bits**	**Part of Immediate**	**Shift/ROR Cmd**
MOV R0,#1	0000	0110	0	0000	0000000000000001				000
MOV R1,R0 ROR#1	0000	0111	0	0001	0000	0000	00001	000	011
MOV R2,R1	0000	0111	0	0010	0000	0001	00000	000	000
ADDS R3,R2,R1	0000	0000	1	0011	0010	0001	00000	000	000
CMP R0,R3	0000	1000	0	0000	0000	0011	00000	000	000
MULLE R4,R3,R0	0101	0010	0	0100	0011	0000	00000	000	000
LDR R5,[R3]	0000	1001	0	0101	0011	0000	00000	000	000
STR R0,[R3]	0000	1010	0	0000	0011	0000	00000	000	000

Result after each instruction:

R0=1

R1=32'b10000000000000000000000000000000 = $(2147483648)_{10}$

R2=R1

R3=0, NZCV=0111

CMP→GT

NOP (Condition not satisfied)

R5=32'b00000110000000000000000000001000 = $(100663304)_{10}$

Output data file: First line contains #1