Microprocessor & Computer Architecture (µpCA)

UE19CS252

Dr. D. C. Kiran

Department of
Computer Science and Engineering

Microprocessor & Computer Architecture (µpCA)

Instruction Encoding

Dr. D. C. Kiran
Department of Computer Science and Engineering

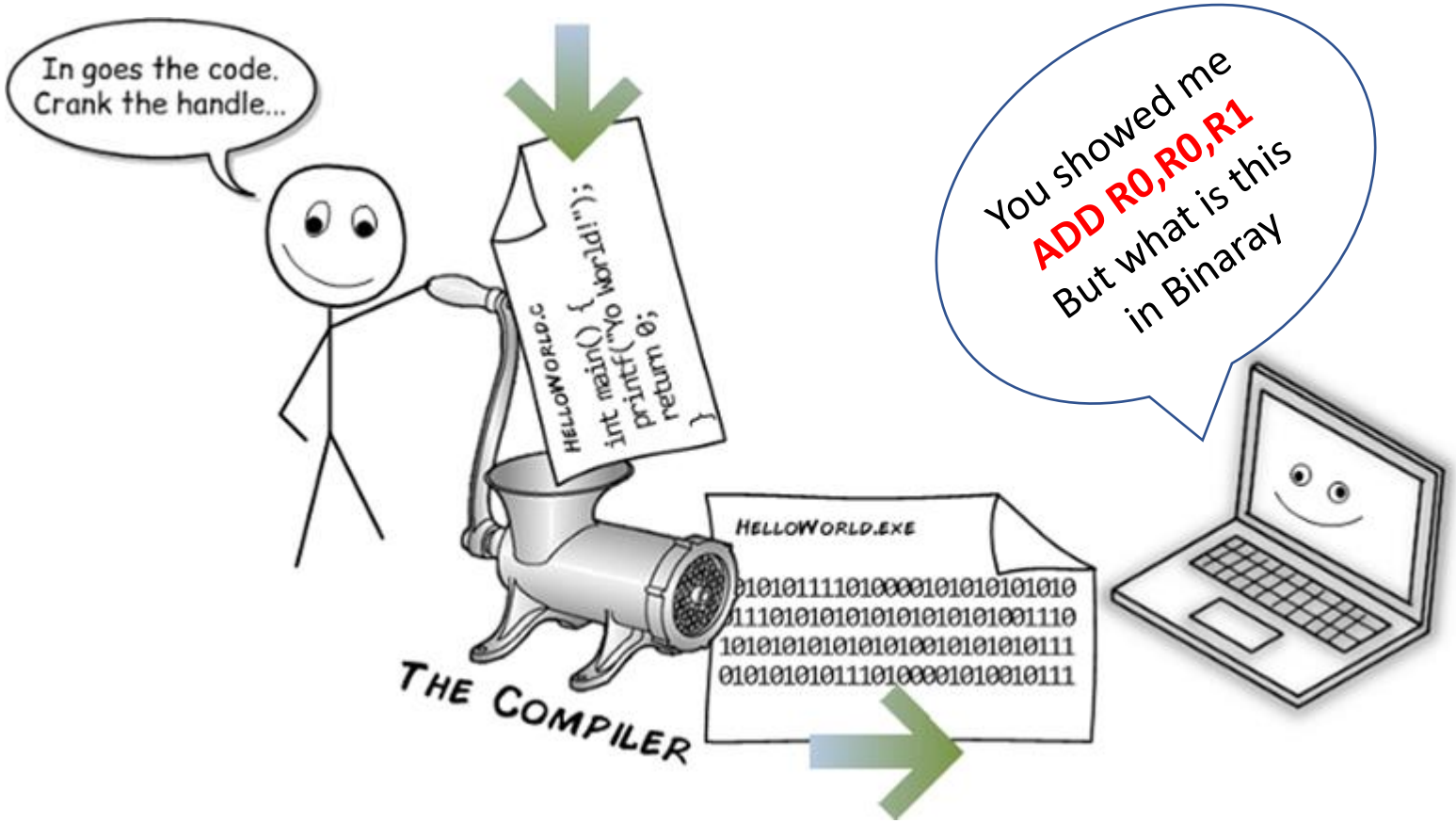
Microprocessor & Computer Architecture (µpCA)

Syllabus

Unit 1: Basic Processor Architecture and Design

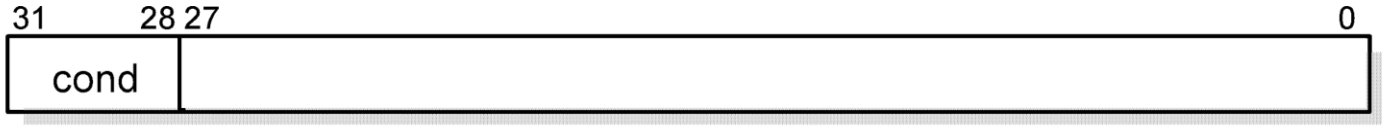
- ~~Microprocessor Overview~~
- ~~CISC VS RISC~~
- ~~Introduction to ARM Processor & Applications~~
- ~~ARM Architecture Overview~~
- ~~Different ARM processor Modes~~
- ~~Register Bank~~
- ~~ARM Program structure~~
- ~~ARM Instruction Format~~
- ~~ARM INSTRUCTION SET~~
- **Instruction Encoding**
  - ✓ **Instruction Layout**
  - ✓ **Data Processing Instruction**

Microprocessor & Computer Architecture (µpCA)

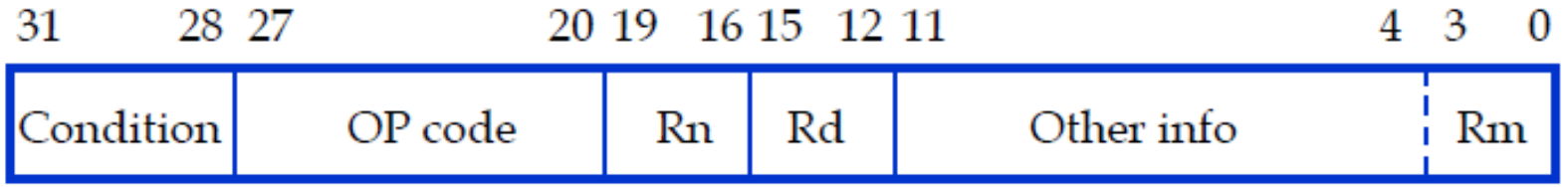


What is an ARM Instruction?

```
OPcode{condition}{S} Rd,Operand1,Operand2
```



Example: Data Processing Instruction

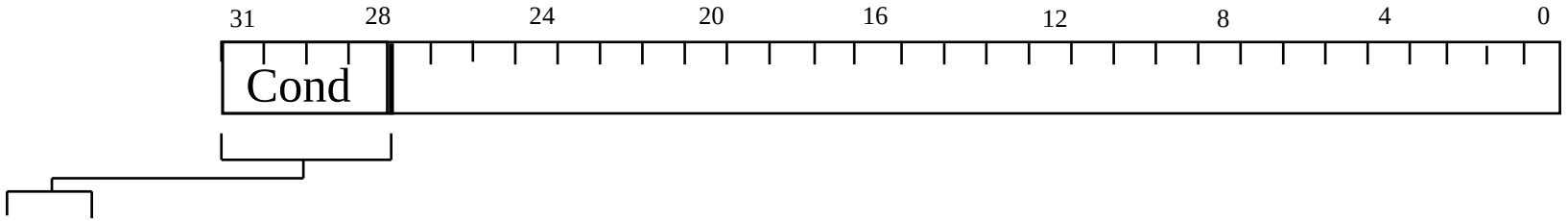


An instruction specifies a **Conditional Execution Code** (Condition), the **OP Code**, **Two or Three Registers** (Rn, Rd, and Rm), and some other information

Reference

Microprocessor & Computer Architecture (µpCA)

The Condition Field



0000 = EQ - Z set (equal)

0001 = NE - Z clear (not equal)

0010 = HS / CS - C set (unsigned higher or same)

0011 = LO / CC - C clear (unsigned lower)

0100 = MI -N set (negative)

0101 = PL - N clear (positive or zero)

0110 = VS - V set (overflow)

0111 = VC - V clear (no overflow)

1000 = HI - C set and Z clear (unsigned higher)

1001 = LS - C clear or Z (set unsigned lower or same)

1010 = GE - N set and V set, or N clear and V clear (>or =)

1011 = LT - N set and V clear, or N clear and V set (>)

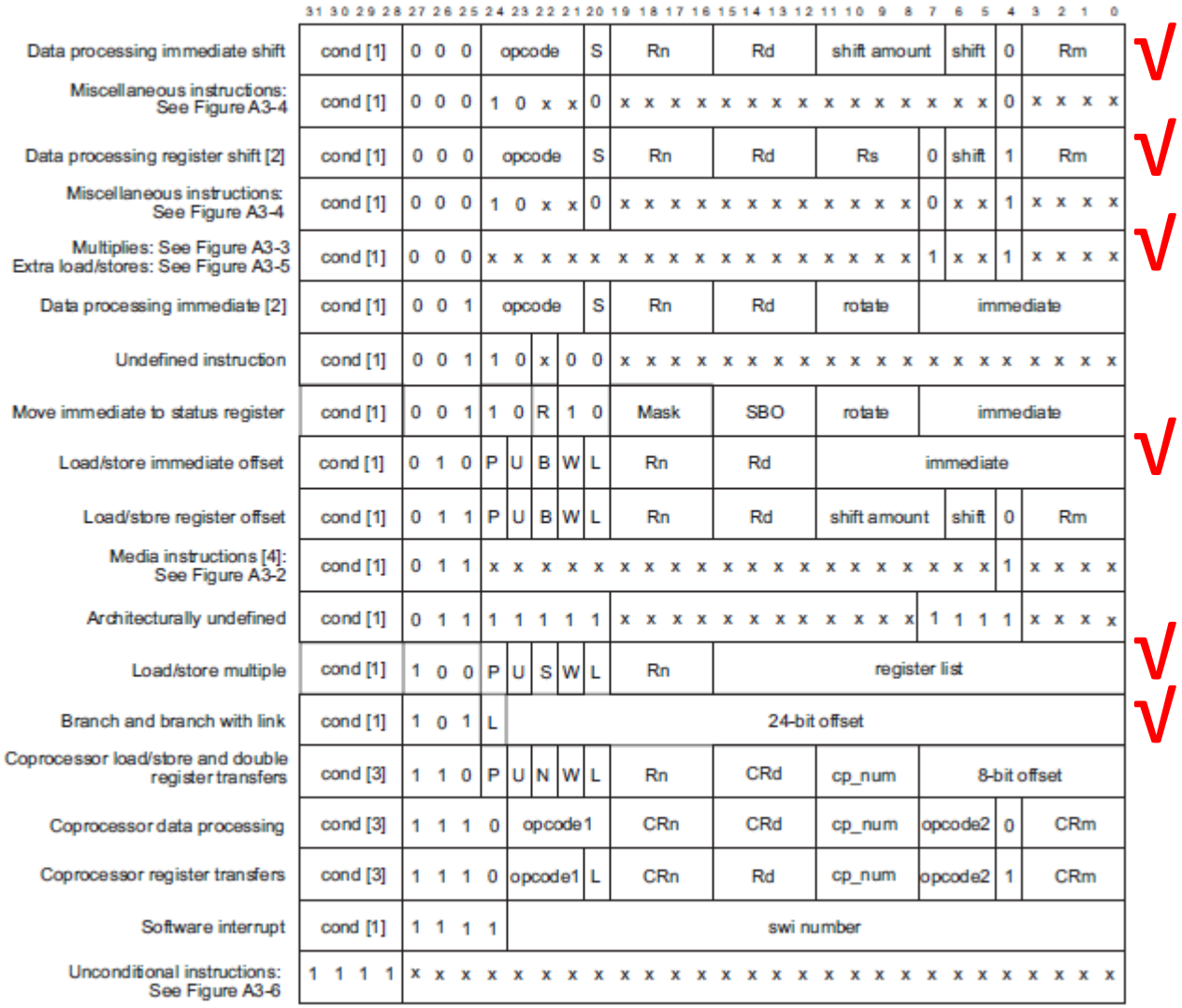
1100 = GT - Z clear, and either N set and V set, or N clear and V set (>)

1101 = LE - Z set, or N set and V clear,or N clear and V set (<, or =)

1110 = AL - always

1111 = NV - reserved.

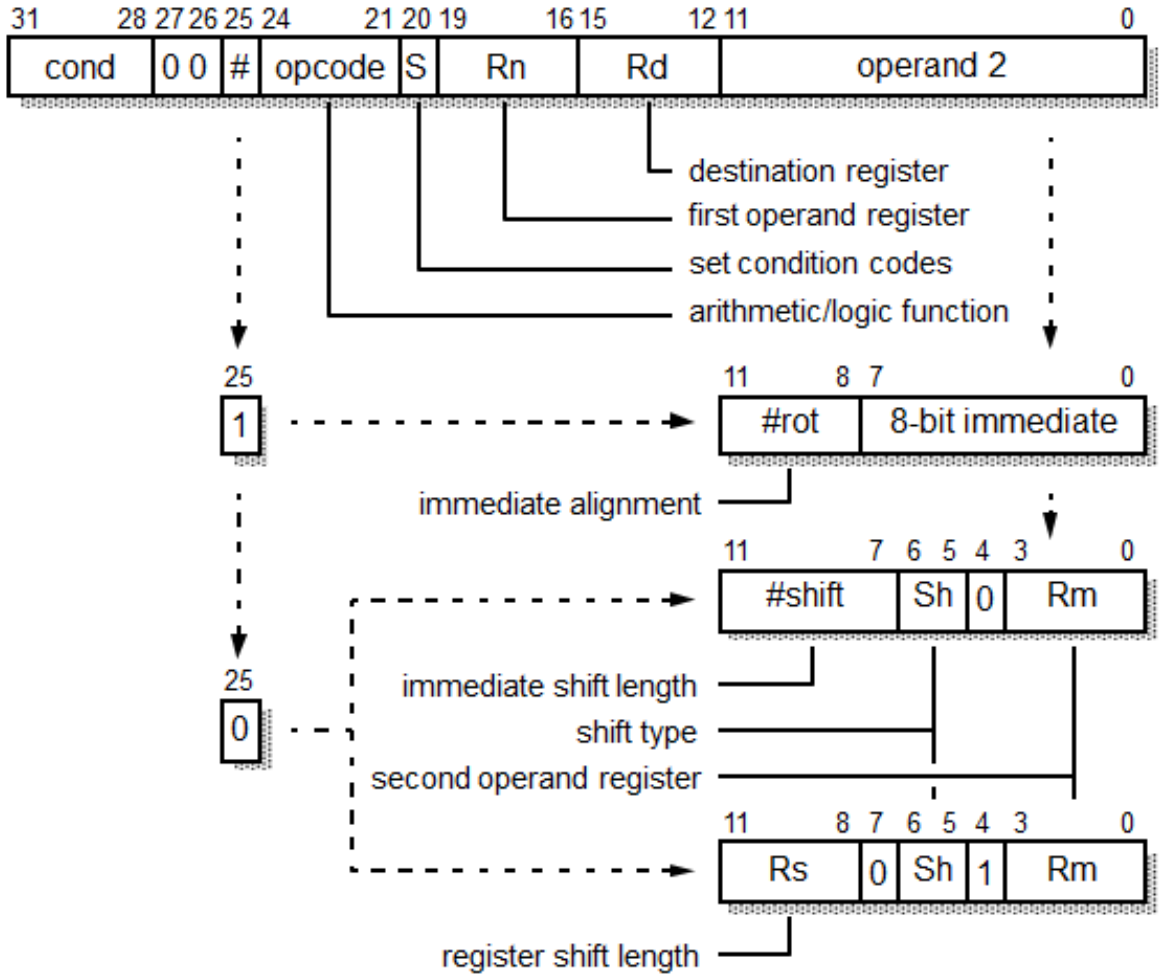
Microprocessor & Computer Architecture (µpCA)



Cond	0	0	I	Opcode				S	Rn				Rd				Operand 2										*Data Processing / PSR Transfer*
Cond	0	0	0	0	0	0	A	S	Rd				Rn				Rs				1	0	0	1	Rm		*Multiply*
Cond	0	0	0	0	1	U	A	S	RdHi				RdLo				Rn				1	0	0	1	Rm		*Multiply Long*
Cond	0	0	0	1	0	B	0	0	Rn				Rd				0	0	0	0	1	0	0	1	Rm		*Single Data Swap*
Cond	0	0	0	1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	1	Rn		*Branch and Exchange*
Cond	0	0	0	P	U	0	W	L	Rn				Rd				0	0	0	0	1	S	H	1	Rm		*Halfword Data Transfer: register offset*
Cond	0	0	0	P	U	1	W	L	Rn				Rd				Offset				1	S	H	1	Offset		*Halfword Data Transfer: immediate offset*
Cond	0	1	I	P	U	B	W	L	Rn				Rd				Offset										*Single Data Transfer*
Cond	0	1	1																		1						*Undefined*
Cond	1	0	0	P	U	S	W	L	Rn				Register List														*Block Data Transfer*
Cond	1	0	1	L	Offset																						*Branch*

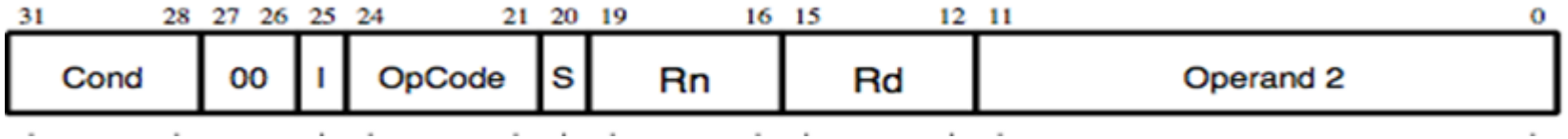
Microprocessor & Computer Architecture (µpCA)

Data Processing Instruction



Opcode [24:21]	Mnemonic
0000	AND
0001	EOR
0010	SUB
0011	RSB
0100	ADD
0101	ADC
0110	SBC
0111	RSC
1000	TST
1001	TEQ
1010	CMP
1011	CMN
1100	ORR
1101	MOV
1110	BIC
1111	MVN

Data Processing Instruction: Example-1



■Ex:

ADD R1, R0, R2

1110 000 0100 S Rn Rd 00000000 Rm

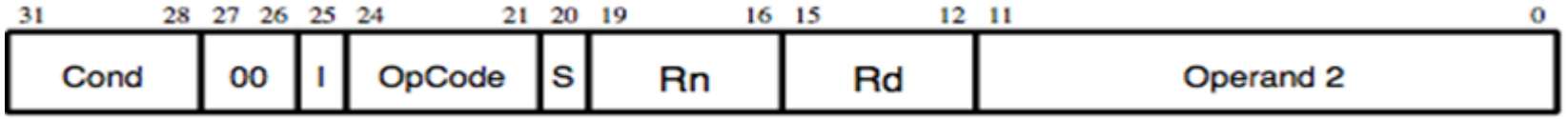
1110 0000 1000 0000 0001 0000 0000 0010

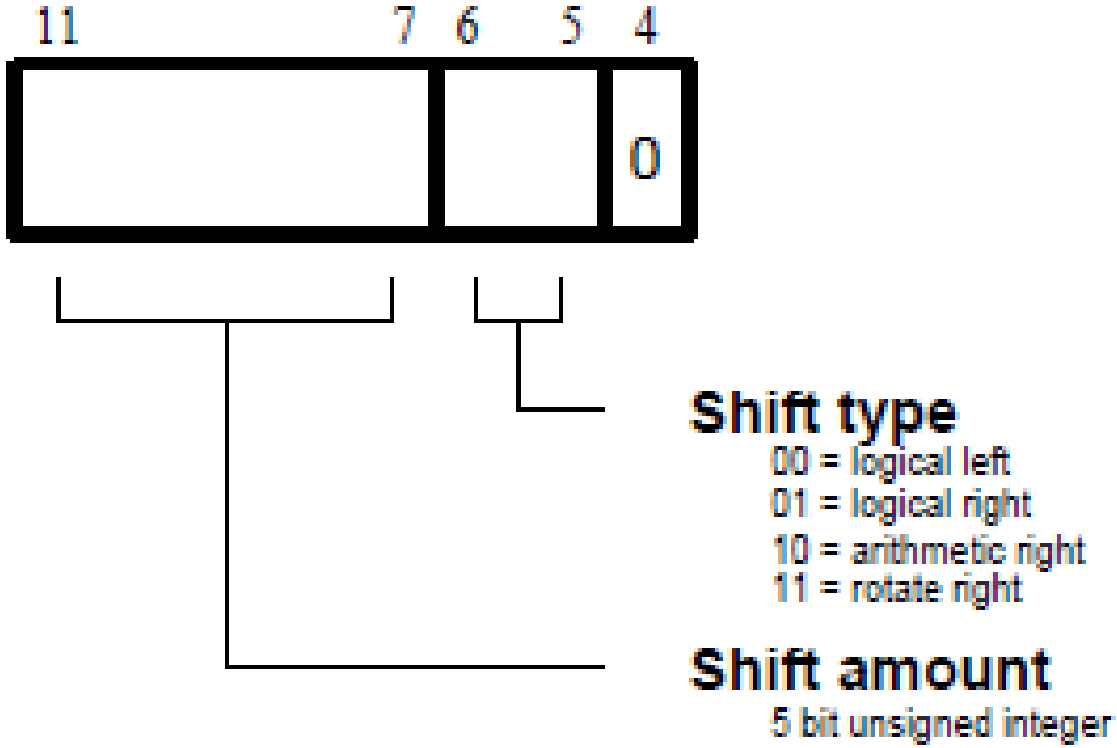
Always ADD R0 R1 R2

E0801002

Microprocessor & Computer Architecture (µpCA)

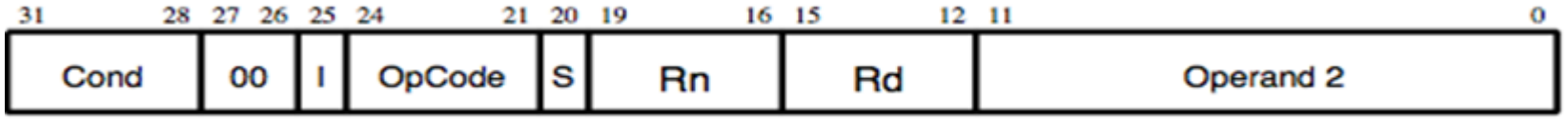
Data Processing Instruction

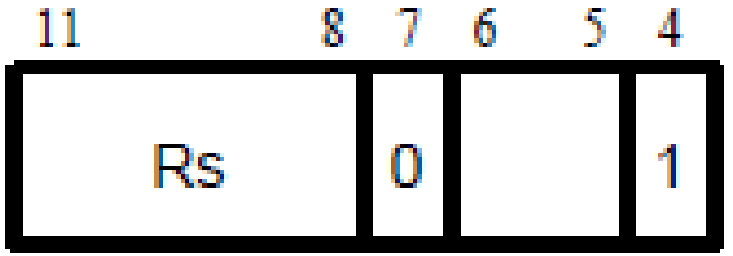




Microprocessor & Computer Architecture (µpCA)

Data Processing Instruction





Shift type

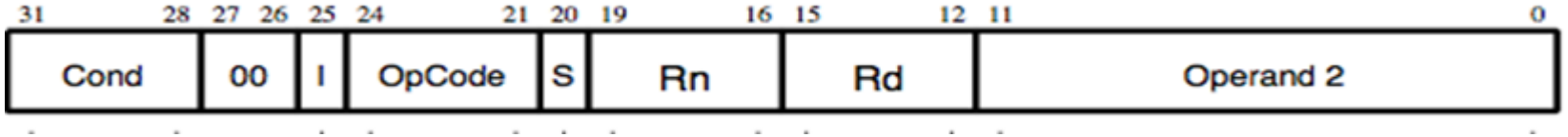
00 = logical left
01 = logical right
10 = arithmetic right
11 = rotate right

Shift register

Shift amount specified in bottom byte of Rs

Microprocessor & Computer Architecture (µpCA)

Data Processing Instrucion: Example-2



■Ex:

ADDS R1, R0, R2 LSR #2

1110 000 0100 S Rn Rd Shift Rm

1110 000 0100 S Rn Rd 00010 01 0 Rm

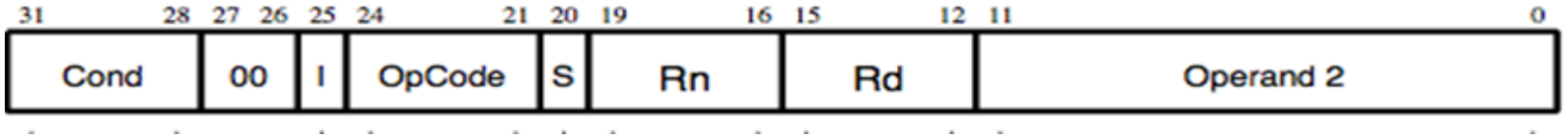
1110 0000 1001 0000 0001 0001 0010 0010

ADDS R0 R1 #2 LSR R2

E0901122

Shift is specified as immediate

Data Processing Instruction: Example-3



■Ex:

ADDS R1, R0, R2 LSR R4

1110 000 0100 S Rn Rd Shift Rm

1110 000 0100 S Rn Rd 0100 0 01 1 Rm

1110 0000 1001 0000 0001 0100 0011 0010

ADDS R0 R1 R4 LSR R2

Shift is specified in Register

E0901432

Next Session

Instruction Encoding

- Branch Instruction
- Multiplication
- Data Transfer Instruction

THANK YOU

Dr. D. C. Kiran
Department of Computer Science and Engineering

dckiran@pes.edu

9829935135