CHAPTER 1

PROGRAM EXECUTION TIME

$$\text{Execution Time} = \text{ISA} \times \text{CPI} \times \text{Clock Period}$$

$$\text{Clock Period} = \frac{1}{\text{Frequency}}$$

- THROUGHPUT: WORK DONE PER UNIT TIME

SPEEDUP

$$A \text{ on } B: \frac{\text{Time } B}{\text{Time } A} = \frac{\text{Performance } A}{\text{Performance } B}$$

$$\text{Performance} = \frac{1}{\text{CPU Time}}$$

CONVERSION

$$1\,GHz = \frac{1}{1 \times 10^9}\ \text{sec}$$

AMDAHL'S LAW

$$\text{Speedup} = \frac{1}{(1-f) + \frac{f}{p}}$$

$\frac{f}{p}$ = CAN IMPROVE $(1-f)$ = CAN NOT IMPROVE

CHAPTER 2

R-FORMAT

OPCODE 11-bits	Rm 5	SHAMT 6	Rn 5	Rd 5

- OPCODE - OPERATION CODE
- Rm - SECOND REGISTER SOURCE
- Shamt - SHIFT
- Rn - FIRST REGISTER SOURCE
- Rd - REGISTER DESIGNATION

I-FORMAT

OPCODE 10	IMMEDIATE 12	Rn 5	Rd 5

- OPCODE - OPERATION CODE
- Rn - SOURCE REGISTER
- Rd - REGISTER DESIGNATION
- Immediate - VALUE

D-FORMAT

OPCODE 11-bits	ADDRESS 9	Op2 2	Rn 5	Rd 5

LOAD/STORE INSTRUCTIONS

- Rn: BASE REGISTER
- ADDRESS: CONSTANT OFFSET FROM CONTENTS OF BASE REGISTER
- Rt: DESIGNATION (LOAD) OR SOURCE (STORE) REGISTER #

B-TYPE

- B 1000 // GO TO LOCATION 10000_{TEN}

5	10000_{TEN}
6	26

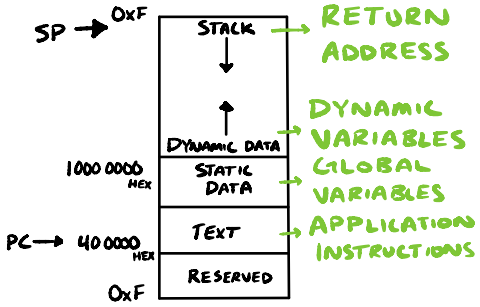
CB-TYPE

- CBNZ X19, EXIT // EXIT IF X19 != 0

181	EXIT	19
8	19	5

- Address = PC + OFFSET (2 Bits)

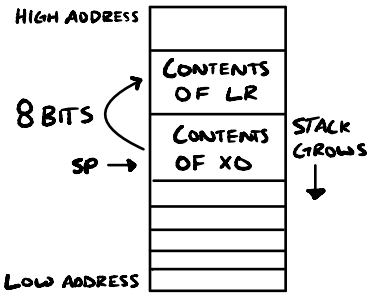
MEMORY ALLOCATION



LEGV8 REGISTERS

- X0-X7: PROCEDURE ARGUMENTS/RESULTS
- X8: INDIRECT RESULT LOCATION REGISTER
- X9-X15: TEMPORARIES
- X16-X17: MAY BE USED BY LINKER AS A SCRATCH REGISTER, OTHER TIMES AS TEMPORARY REGISTER.
- X18: PLATFORM REG FOR PLATFORM IND. CODE. OTHERWISE A TEMP REG.
- X19-X27: SAVED
- X28 (SP): STACK POINTER
- X29 (FP): FRAME POINTER
- X30 (LR): LINK REG (RETURN ADDRESS)
- X2R (REG 31): CONSTANT VALUE 0

DATA ON STACK



CONDITIONAL OPERATIONS

- B.EQ
- B.NE
- B.LT (LESS THAN)
- B.LE (LESS THAN OR EQUAL)
- B.GT (GREATER THAN)
- B.GE (GREATER THAN, EQUAL TO)
- B.HS (UNSIGNED)

OTHER OPERATIONS

- LDUR X1, [X2, 40] • STUR X1, [X2, 40]
  X1 = MEMORY[X2, 40] MEMORY[X2+40] = X1
- LSL BY i BITS MULTIPLIES BY 2^i
- LSR BY i BITS DIVIDES BY 2^i

NON LEAF PROCEDURE

- C CODE:

```
int fact (int n){
  if (n < 1) return 1;
  else return n * fact(n - 1);
}
```

- ARGUMENT IN X0
- RESULT IN X1
- LEGV8 CODE:

```
fact:
SUBI SP,SP,#16      Save return address and n on stack
STUR LR,[SP,#8]
STUR X0,[SP,#0]
SUBIS XZR,X0,#1     compare n and 1
B.GE L1             if n >= 1, go to L1
ADDI X1,XZR,#1      Else, set return value to 1
ADDI SP,SP,#16      Pop stack, don't bother restoring values
BR LR               Return
L1: SUBI X0,X0,#1   n = n - 1
BL fact             call fact(n-1)
LDUR X0,[SP,#0]     Restore caller's n
LDUR LR,[SP,#8]     Restore caller's return address
ADDI SP,SP,#16      Pop stack
MUL X1,X0,X1        return n * fact(n-1)
BR LR               return
```

CHAPTER 3

IEEE 754 FLOATING-POINT STANDARD

SINGLE PERCISION = 127

DOUBLE = 1023

$$(-1)^S \times (1 + \text{FRACTION}) \times 2^{(\text{EXPONENT} - \text{BIAS})}$$

IEEE SINGLE PRECISION AND DOUBLE PRECISION FORMATS

S	EXPONENT	FRACTION
1	8	23

S	EXPONENT	FRACTION
1	11	52

Floating-Point Addition

Now consider a 4-digit binary example
- $1.000_2 \times 2^{-1} + -1.110_2 \times 2^{-2}$ (0.5 + −0.4375)

1. Align binary points
   - Shift number with smaller exponent
   - $1.000_2 \times 2^{-1} + -0.111_2 \times 2^{-1}$
2. Add significands
   - $1.000_2 \times 2^{-1} + -0.111_2 \times 2^{-1} = 0.001_2 \times 2^{-1}$
3. Normalize result & check for over/underflow
   - $1.000_2 \times 2^{-4}$, with no over/underflow
4. Round and renormalize if necessary
   - $1.000_2 \times 2^{-4}$ (no change) = 0.0625

Floating-Point Multiplication

Consider a 4-digit decimal example
- $1.110 \times 10^{10} \times 9.200 \times 10^{-5}$

1. Add exponents
   - For biased exponents, subtract bias from sum
   - New exponent = 10 + −5 = 5
2. Multiply significands
   - $1.110 \times 9.200 = 10.212 \Rightarrow 10.212 \times 10^5$
3. Normalize result & check for over/underflow
   - 1.0212×10^6
4. Round and renormalize if necessary
   - 1.021×10^6
5. Determine sign of result from signs of operands
   - $+1.021 \times 10^6$

DENORMAL NUMBERS

- EXPONENT = 000...0
  $X = (-1)^S \times (0 + \text{FRACTION}) \times 2^{-\text{BIAS}}$
- DENORMAL FRACTION
  $X = -1^S \times (0+0) \times 2^{-\text{BIAS}}$

INFINITY AND NANS

- EXP = 111...1, FRACTION = 000...0
  = ∞
- EXP = 111...1, FRACTION ≠ 000...0
  = NOT A NUMBER

MULTIPLICATION HARDWARE