SPRING 2016
CMPE 364

Microprocessor Based Design

Dr. Ryan Riley
(Slides adapted from Dr. Mohamed Al-Meer)

Logical Instructions

- Logical instructions perform bitwise logical operations on the two source registers.
- Allow individual bit manipulation: at bit level

```
Syntax: <instruction>{<cond>}{S} Rd, Rn, N
```

AND	logical bitwise AND of two 32-bit values	$Rd = Rn \,\&\, N$
ORR	logical bitwise OR of two 32-bit values	$Rd = Rn \mid N$
EOR	logical exclusive OR of two 32-bit values	$Rd = Rn \,\hat{}\, N$
BIC	logical bit clear (AND NOT)	$Rd = Rn \,\&\, \sim N$

AND Instruction

- Performs bitwise AND of two source operands, SRC1 and SRC2, and stores the result in DST.

  AND DST, SRC1, SRC2

- SRC1 is a GP register, while SRC2 could be GP register or an immediate operand. DS must be a GP register.
- EXAMPLE: R1=0x1A89F045, R12=0x96231E8D

  AND R4, R1, R12

Get R4 = ?, show bit by bit solution

AND Instruction

- EXAMPLE: write an AND instruction to clear every bit except 16, 17, and 19 of register R4?
- SOLUTION:

  mask = **0x000B0000**

  = 0000 0000 0000 **1011** 0000 0000 0000 0000

  AND R4, R4, 0x000B0000

Show in bit by bit.

ORR Instruction

- Performs bitwise Inclusive OR of SRC1 and SRC2 and places the result in DST.
- SRC1 is a register while SRC2 could be a register or an immediate operand.

  ORR DST, SRC1, SRC2
- EXAMPLE: R6 = 0xEF486EE9, R1=0x80182ACF
- SOLUTION: ??? (**0xEF586EEF**)

ORR Instruction

- EXAMPLE

Set bit positions 7 and 9 in register R0 to 1 without touching other 30 bits in the same register?

- SOLUTION:

Will use an ORR instruction, Oring R1 register with a 32-bit Mask.

Mask will be = 0x --------?

ORR R0, R0, #0x00000280

EOR (Exclusive OR) Instruction

- Exclusive OR which performs Exclusive OR of 2 sources, SRC1 And SRC2, and places result in DST.

  EOR DST, SRC1, SRC2

- SRC1: GP register, SRC2: GP register or immediate operand, DST must be GP register.
- Very useful instruction for **comparison** to see which bit position they differ. Also used in **Encryption and Decryption with Keys** (**our coming Project ??**)

EOR Instruction

- EXAMPLE: Get R9? if R5 = 0x7C5D6B21, R8 = 0x9624EC30?

  EOR R9, R5, R8
- ANSWER:

  R9 = 0xEA798711.
- EXAMPLE: **Invert bit 30** of R0 using a single instruction?
- SOLUTION:

  MASK = **0100** 0000 0000 0000 0000 0000 0000 0000 =
  = 0x40000000

  EOR R0, R0, 0x40000000

BIC (Bit Clear) Instruction

- Bit Clear Instruction clears 1 bit or more in the first source register SRC1, and stores the result in DST register
- The **bit positions** needed for c**le**ar should be placed in **SRC2** register by using bit-1 and bit-0 means do not invert

BIC DST, SRC1, SRC2

SRC1 AND !SRC2

BIC (Bit Clear) Instruction

- Example: using a single instruction clear bit-5 I register R0?
- Solution:

Mask = **0000 0000 0000 0000 0000 0000 0010 0000 =**

= 0x20

BIC R0, R0, #0x20