CPE 221 Worked Examples: Machine Code in ARM

Instructor: Rahul Bhadani

1 Decode Data-Processing Instruction

Decode the ARM instruction 0xE0831002 into its assembly language form.

Solution

```
ADD R1, R3, R2
```

Explanation

- op code (27-26): 00 indicates a data-processing instruction
- funct (25-20): 0 0100 0 where bits 24-21 are 0100 which is the ADD operation
- Rn (19-16): 0011 = R3 (first source register)
- Rd (15-12): 0001 = R1 (destination register)
- Src2 (11-0): 000000000010 = R2 (second source register)

2 Identify Conditional Execution

What is the condition code for the instruction 0x1A000003 and what does it mean?

Solution

Condition code is 0001 (NE), meaning “Not Equal”.

Explanation

- Condition code is in bits 31-28: 0001
- According to the ARM condition codes, 0001 corresponds to NE (Not Equal)
- This instruction will execute only if the Z flag is not set (Z=0)
- The full instruction is a branch instruction (BNE) that branches if the result is not equal

3 Decode Memory Instruction

Decode the machine code `0xE5845000` into its assembly form.

Solution

```
STR R5, [R4]
```

Explanation

- op code (27-26): 01 indicates a memory instruction
- L bit (20): 0 indicates store (STR)
- Rn (19-16): 0100 = R4 (base register)
- Rd (15-12): 0101 = R5 (source register to be stored)
- P bit (24): 1 indicates pre-indexing
- U bit (23): 1 indicates addition
- Offset (11-0): 000000000000 = 0 (no offset)

4 Encode ADD Instruction

Encode the assembly instruction `ADD R7, R2, R3, LSL #2` into machine code.

Solution

0xE0827103

Explanation

- Cond (31-28): 1110 (AL - Always execute)
- Op (27-26): 00 (data-processing)
- Funct (25-20): 001000 (ADD operation, I=0, S=0)
- Rn (19-16): 0010 (R2)
- Rd (15-12): 0111 (R7)
- Shift (11-4): 0001 0000 (LSL #2)
- Rm (3-0): 0011 (R3)

5 Identify Immediate Value

What is the immediate value in the instruction 0xE2851014?

Solution

The immediate value is 20 (decimal).

Explanation

- This is a data-processing instruction with I=1 (immediate)
- The immediate field is in bits 7-0: 00010100 = 0x14 = 20 decimal
- The rotation field (bits 11-8) is 0000, meaning no rotation
- The instruction is ADD R1, R5, #20 (adding immediate value 20 to R5)

6 Calculate Branch Target Address

If the instruction at address 0x8400 is 0xEA000005, what is the branch target address?

Solution

The branch target address is 0x841C.

Explanation

- This is an unconditional branch (B) instruction
- The immediate field is 5
- Branch target = PC+8 + (imm24 × 4)
- PC+8 for the instruction at 0x8400 is 0x8408
- 5 × 4 = 20 (0x14)
- 0x8408 + 0x14 = 0x841C

7 Decode STR Instruction with Offset

Decode the machine code 0xE5073018 into its assembly form.

Solution

STR R3, [R7, #-24]

Explanation

- op code (27-26): 01 indicates a memory instruction
- L bit (20): 0 indicates store (STR)
- Rn (19-16): 0111 = R7 (base register)
- Rd (15-12): 0011 = R3 (source register)
- P bit (24): 1 indicates pre-indexing
- U bit (23): 0 indicates subtraction
- Offset (11-0): 000000011000 = 24 decimal
- The negative sign comes from U=0 (subtraction)

8 Identify Function Code

What operation is performed by the instruction `0xE0413002`?

Solution

This is a SUB instruction that performs subtraction.

Explanation

- op code (27-26): `00` indicates a data-processing instruction.
- funct (24-21): `0100` corresponds to SUB operation.
- The instruction subtracts R2 from R1 and stores the result in R3

9 Decode Load Multiple Instruction

Decode the machine code `0xE92D4030` into its assembly form.

Solution

PUSH {R4, R5, LR}

Explanation

- This is a STMFD instruction (Store Multiple Full Descending)
- Register list (bits 15-0): `0100000000110000` sets bits 4, 5, and 14
- So registers R4, R5, and LR (R14) are being stored
- Base register (bits 19-16): `1101` = R13 (SP)
- When SP is the base register with writeback, this is equivalent to the PUSH instruction

10 Encode Conditional Execution

Encode the assembly instruction `ADDEQ R0, R1, R2` into machine code.

Solution

`0x00810002`

Explanation

- Cond (31-28): 0000 (EQ - Equal)
- Op (27-26): 00 (data-processing)
- Funct (25-20): 001000 (ADD operation, I=0, S=0)
- Rn (19-16): 0001 (R1)
- Rd (15-12): 0000 (R0)
- Src2 (11-0): 000000000010 (R2)
- This instruction adds R2 to R1 and stores the result in R0, but only if the Z flag is set (indicating equality)