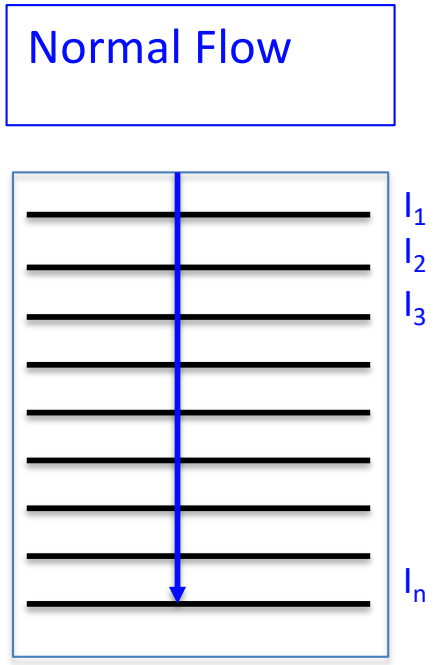
2.3.5 Instruction Execution

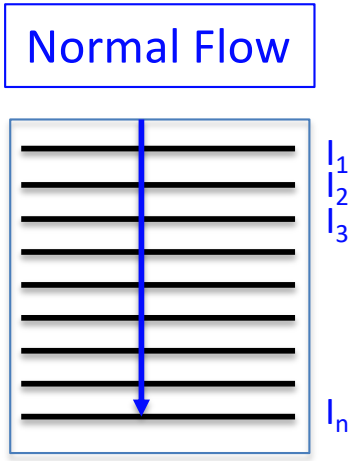
- **Instruction cycle:**
  - Fetch (decode) and Execute
- **Straight-line sequencing, normally:**
  - Instructions executed in **sequential order**
  - PC incremented by 2/4/8 bytes
    - for 16/32/64-bit processor

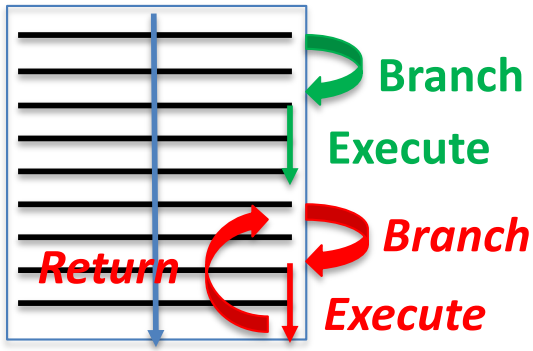


2.3.5 Instruction Sequencing

How to change control flow?

- Change Program Counter
- Two Types:
  - Branch or *Interrupt*





Internal and Expected
Issued by Processor or Program

External and Unpredictable
Issued by Input and Output devices

2.3.6 Branching

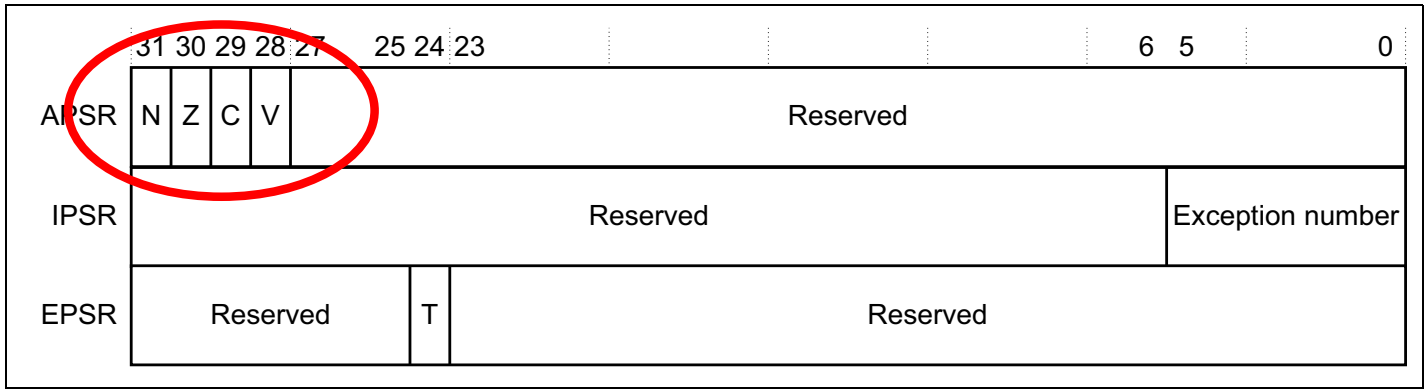
- **Conditional branch:**
- Initialize R2 as a counter and DO ***X*** times
  - LOOP: Instruction 1
    Instruction 2
    …
    Subtract R2, R2, #1 ; may change status
    Branch > 0 LOOP ; check status
- **Branch target is LOOP** if R2 > 0 ; else exit loop
- **Condition code** in a **status register (NZCV in APSR)**
  - previous operation results for subsequent conditional use

ARM Conditional Execution (PM39)

- The processor carries out the branch based on the condition code (CC) set by another instruction:

N=negative
Z=zero
C=carry
V=overflow

Figure 3. APSR, IPSR and EPSR bit assignments



Conditional Branch Placement

- **Branch CC** placement:

1. Immediately after the instruction that set the condition code CC,

OR

2. After any number of in-between (change CC and branch) instructions that must **not** update the **same** condition code

What if the same CC is updated twice ?

The second update overrides the first!

Branch Placement

- Initialize R2 as a counter
  - LOOP1: Instruction 1
    Instruction 2
    …
    Subtract R2, R2, #1 ; may change status
    Branch > 0 LOOP ; check status (Z-bit)
  - LOOP2: Instruction 1
    Subtract R2, R2, #1 ; may change status
    Instructions after CC instruction **; don't touch Z bit**
    Instructions before Branch **; don't touch Z bit**
    Branch > 0 LOOP ; check status (Z-bit)

What if different CCs are updated ?

Thumb Conditional Branch

Cond	THUMB assembler	ARM equivalent	Action
0000	BEQ label	BEQ label	Branch if Z set (equal)
0001	BNE label	BNE label	Branch if Z clear (not equal)
0010	BCS label	BCS label	Branch if C set (unsigned higher or same)
0011	BCC label	BCC label	Branch if C clear (unsigned lower)
0100	BMI label	BMI label	Branch if N set (negative)
0101	BPL label	BPL label	Branch if N clear (positive or zero)
0110	BVS label	BVS label	Branch if V set (overflow)
0111	BVC label	BVC label	Branch if V clear (no overflow)
1000	BHI label	BHI label	Branch if C set and Z clear (unsigned higher)
1001	BLS label	BLS label	Branch if C clear or Z set (unsigned lower or same)
1010	BGE label	BGE label	Branch if N set and V set, or N clear and V clear (greater or equal)
1011	BLT label	BLT label	Branch if N set and V clear, or N clear and V set (less than)
1100	BGT label	BGT label	Branch if Z clear, and either N set and V set or N clear and V clear (greater than)
1101	BLE label	BLE label	Branch if Z set, or N set and V clear, or N clear and V set (less than or equal)

N=negative
Z=zero
C=carry
V=overflow

Code = 1110 => undefined

Code = 1111 => SWI ; software interrupt