MECE336– Microprocessors I

Lecture 3 – Simple Programs

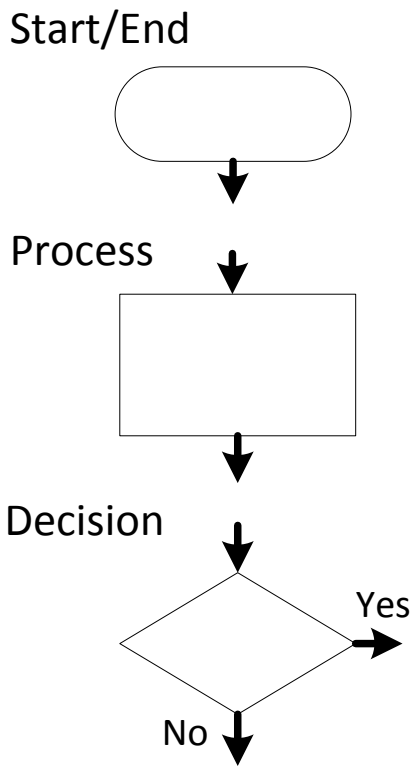
Associate Prof. Dr. Klaus Werner Schmidt

Department of Mechatronics Engineering – Çankaya University

Compulsory Course in Mechatronics Engineering
Credits (3/2/4)

Course Webpage: http://MECE336.cankaya.edu.tr

Flow Charts: Components



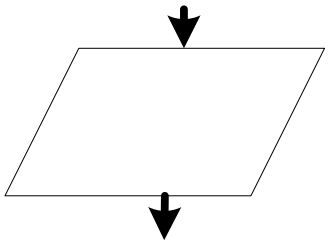
- Shown as oval or rounded rectangle
- Represents the start or end of a process
- Example content: Start, End

- Shown as rectangle
- Used to show that some operation is performed
- Example: "Add 1 to X", "Save X"

- Shown as diamond
- Represents a true/false (Yes/No) decision
- Example: "Is $X \geq 0$?"

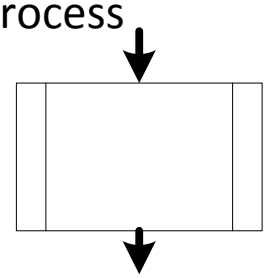
Flow Charts: Components

Data, Input/Output



- Shown as a parallelogram
- Represents receiving data, displaying data
- Examples: Get X from the user, display X

Subprocess



- Shown as rectangle with double lines
- Represents a complex processing step with a separate flowchart
- Example: Subroutine

Flow Line

- Arrow from one symbol to another symbol
- Represents that control passes to the symbol the arrow points to

Flow Charts: Example Programs as Flowcharts

Value Assignment **Flowchart**

Flow Charts: Example Programs

Conditional Assignment **Flowchart**

Basic Instructions: Move

Different Versions

- `movwf f`: Move the contents of the `W` register to memory location `f`
  Example:

- `movf f,d`: Move the contents of the memory location `f`
  - to the `W` register if the `d` bit is set to 0
  - to the memory location `f` if the `d` bit is set to 1

  Example:

- `movlw k`: Move the literal value `k` (8-bit number) into the `W` register
  Example:

Basic Instructions: Clear

Different Versions

- `clrw`: Clears the value in the `W` register to zero. There are no operands to specify.
  Example:

- `clrf f`: Clears the value of the memory location `f`. The programmer has to specify a value for `f`, which must be a valid memory address.
  Example:

Basic Instructions: Bit-wise Operations

Clear and Set

- `bcf f,b`: Set bit `b` (between 0 and 7) in memory location `f` to logic 0. This is called clear.
  Example:

- `bsf f,b`: Set bit `b` in memory location `f` to logic 1.
  Example:

Basic Instructions: Bit Conditions

Skip if Clear

- `btfsc f,b`: Skips the next instruction if bit `b` in memory location `f` is 0. Otherwise performs the next instruction.
  Example:

Basic Instructions: Bit Conditions

Skip if Set

- `btfss f,b`: Skips the next instruction if bit `b` in memory location `f` is 1. Otherwise performs the next instruction.
  Example:

Basic Instructions: Examples

Example 1

- Define `val1` as value 23 and `val2` as 45
- Clear memory locations 0x0C (bank 0) and 0x9C (bank1)
- Move `val1` to 0x0C and `val2` to 0x9C
- Clear bit 5 of 0X0C and bit 0 of 0x9C
- Move 0x9C to the working register `W`

Example 2

- Define `val1` as value 255
- Move `val1` to the memory location 0x0D
- Skip the next instruction if bit 6 of 0x0D is zero
- Clear bit 5,6,7 of 0x0D and jump back to the previous instruction
- Move 0x0D to the working register `W`

Basic Instructions: Example1

Basic Instructions: Example2

Basic Instructions: Increment/Decrement

Increment

- `incf f,d`: Increment the content of memory location `f` and write the result to
  - `W` register if `d` is 0
  - memory location `f` if `d` is 1

  Example:

Basic Instructions: Increment/Decrement

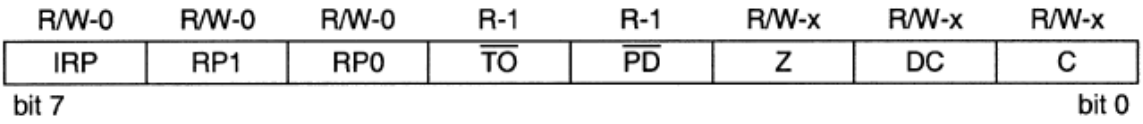
Decrement

- `decf f,d`: Decrement the content of memory location `f` and write the result to
  - `W` register if `d` is 0
  - memory location `f` if `d` is 1

  Example:

Basic Instructions: Status Register

Status Register



Description

- bit 7-6: unimplemented
- bit 5: **RP0** – register bank select bit
- bit 4: $\overline{\textbf{TO}}$ Time-out bit
- bit 3: $\overline{\textbf{PD}}$ Power-down bit
- bit 2: **Z**: Zero bit (1 if result of an operation is 0, otherwise 0)
- bit 1: **DC**: Digit carry/borrow bit (used in arithmetic operations)
- bit 0: **C**: Carry/borrow bit (used in arithmetic operations, rotation)

$\rightarrow$ Check instruction set to see which instruction affects the status register

Basic Instructions: Status Register Examples

`movf`

`infc` **and** `decf`

Basic Instructions: Increment/Decrement

Increment and Skip

- `incfsz f,d`: Increment like `incf`. Skip the next instruction if the result of `incf` is zero.
  Example:

Basic Instructions: Increment/Decrement

Decrement and Skip

- `decfsz f,d`: Decrement like `decf`. Skip the next instruction if the result of `decf` is zero.
  Example:

More Instructions: Rotate

Rotate Left

- `rlf f d`: Rotate left the content of memory location `f` using carry. Write result to `W` register if `d` is 0 and memory location `f` if `d` is 1.
  Explanation:

More Instructions: Rotate

Rotate Right

- `rrf f d`: Rotate right the content of memory location `f` using carry. Write result to `W` register if `d` is 0 and memory location `f` if `d` is 1. Explanation:

Examples:

Description

- Define `val1` with value 73 and write to memory location 0x0C
- Decrement 0x0C until it becomes zero
- Write `val1` to memory location 0x0D
- Rotate 0x0D left 3 times
- Increment 0x0D until it becomes zero

Examples:

Flow Charts Basic Instructions More Instructions Example

Examples:

Klaus Schmidt Department
MECE336 – Microprocessors I