AMINA QADEER

CE-42-A

359607

ASIGNMNENT 2

Compare and give the explanation of different assembler directives of PIC18F vs MP8086

The assembler directive used in PIC18 Microcontroller (PIC18F452. Assembler directives are instructions to Assembler or compiler to let it know where code is placed and for what chip of PIC is being used further more what is the default number system and if there is any variable used in code. So these are instructions to assembler so that it can convert it assembly language code into machine code correctly and pass it to CPU of Microcontroller so that it can execute the code to accomplish the task.

PIC18F vs MP8086 are two microcontrollers which have different assembler directives. MP8086 has instructions which can be used to do bit-wise operations and shift register operations. PIC18F has instructions which can be used to do arithmetic and compare operations.

When programming an 8-bit microcontroller it's important to make sure that the assembler directives match. They are used to specify what memory to use, which registers to use, and which type of addressing is being used.

MP8086:

The data defining pseudo operations DB, DW, DD, DQ, and DT are assembler instructions used to describe the amount of data or variables. To make constants easier to utilise, names are given to them using EQU. Comments, which are only used to communicate information to other programmers, are also directives. Each segment's beginning can be determined using the tags.data,.code,.stack, and.model. OFFSET instructs the assembler to use the variable's offset address rather than its data. PROC is used to indicate the beginning of a specified procedure, and ENDP is used to indicate its conclusion. Multiple initializations of variables are possible thanks to DUP. The terms FAR and NEAR indicate whether the segment in which the named label is located is the same or different.

End of segment information is conveyed by ENDS, even address alignment is conveyed by EVEN, and operand size is conveyed by PTR. Several guidelines that weren't included in the slides include Assembler commands include INCLUDE, which instructs it to include code from a specified file in our code, GROUP, which organises segments into a single local group, NAME, which is used to give names to assembly modules, GLOBAL, and PUBLIC, which instructs it to make variables visible to other modules. STRUCT is used to specify a group of

data types for later use. A procedure is in another module, the assembler is informed by EXTERN.

PIC18F452

There are three different assembler directives for PIC18F processors: Assembler directive PIC18Fxxx.L (low byte), PIC18Fxxx.H (high byte), and PIC18Fxxx.C (carry flag). The MP8086 has two different assembler directives: Assembler directive MP8086.L (low byte) and MP8086.H (high byte).

1. CONTROL DIRECTIVE:
   The PIC 18F452 has three control directives that can be used to control the devices on the board. The two most common ones are the oscillator start-up and stop-up commands. The oscillator start-up command tells the microcontroller to start up the oscillator, and the oscillator stop-up command tells it to stop the oscillator. These two commands can be sent from the microcontroller or from an external source. The third directive is a simple bit-bang command that is sent from an external source and turns on or off all of the outputs on the board.

2. CONDITIONAL DIRECTIVES:
   PIC 18F452 2-Conditional Directives
   This is a programmable interrupt controller (PIC) that is designed to be used in applications where it is necessary to control a system based on specific conditions. The PIC can be programmed with instructions that include two conditional directives. The first directive, an always-true statement, will be executed no matter what the current state of the system. The second directive, an always-false statement, will be executed only if the first directive is true. The PIC can also be programmed with a third directive that will execute if both the always-true and the always-false statements are true.

   3.DATA DIRECTIVES:
   Data Directives are instructions that tell the microcontroller what to do with the data that it is receiving. Data Directives can be used to control the direction of the data, or even to control the speed of the data.The PIC 18F452 3-Data Directives is a powerful device that can be used to measure the voltage, current, and resistance of a circuit. It uses a proprietary technology called a microcontroller to calculate the voltage, current, and resistance of the circuit. The PIC 18F452 3-Data Directives also has a built-in oscilloscope which can be used to measure voltages and currents. DB and DW to define data of 1 byte and 1 word respectively, cblock andendc to start and stop an automatic constant block, res to reserve memory, fill to specify program memory fill value, DATA to create numeric and text data etc.

3. LSITING DIRECTIVES:

A listing directive is a single line that indicates the order in which the program should be executed. They control assembler listing file format. The listing directive consists of a line that begins with the letter L, followed by a line that begins with the letter I, and finally a line that begins with the letter C. The program starts at the first line following the L line, and goes until the end of the last line following the C line.

4. Macro Directives
   Macro, endm and exit to declare, end, and exit macro definition respectively, expand and no expand to extend and turn off macro listing or expansion and local to define a local macro variable.
5. Object File Directives
   A PIC18F452 object file is a binary file that contains the information necessary to program a PIC18F452 microcontroller. A PIC18F452 object file can be created by using the PIC18F452 Object Programmer, or by using a text editor with a PIC18F452 compiler. A PIC18F452 object file is made up of directives and data blocks. Directives are instructions that are required for the program to function, while data blocks are data that can be read by the program.

COMPARISON:

With the ability to write macros, conditionally assembled code, variable declaration, memory allocation, and memory access, PIC18F assembly directives are much more flexible than those in MP8086.

The majority of the assembly directives in MP8086 and PIC18F are similar, but PIC18F has a lot more control and ease of use to make up for its constrained capacity and SFRs. MP8086's assembly directives were more control and data focused. PIC18F's assembly directives, on the other hand, are meant to help with control, data, listing, macros, object files, and also applying conditions at assembly time.