INTRODUCTION

- 8087 was the first math coprocessor for 16-bit processors designed by Intel.
- It was built to pair with 8086 and 8088.
- The purpose of 8087 was to speed up the computations involving floating point calculations.
- Addition, subtraction, multiplication and division of simple numbers is not the coprocessor’s job.
- It does all the calculations involving floating point numbers like scientific calculations and algebraic functions.

INTRODUCTION

- By having a coprocessor, which performs all the calculations, it can free up a lot of CPU’s time.
- This would allow the CPU to focus all of its resources on the other functions it has to perform.
- This increases the overall speed and performance of the entire system.
- This coprocessor introduced about 60 new instructions available to the programmer.
- All the mnemonics begin with “F” to differentiate them from the standard 8086 instructions.
- For e.g.: in contrast to ADD/MUL, 8087 provide FADD/FMUL.

ARCHITECTURE OF 8087

- 8087 coprocessor is designed to operate with 8086 microprocessor.
- The microprocessor and coprocessor can execute their respective instructions simultaneously.
- Microprocessor interprets and executes the normal instruction set and the coprocessor interprets and executes only the coprocessor instructions.
- All the coprocessor instructions are ESC instructions, i.e. they start with “F”.

Math coprocessor is also called as:

- Numeric Processor Extension (NPX)
- Numeric Data Processor (NDP)
- Floating Point Unit (FPU)
ARCHITECTURE OF 8087
- The internal structure of 8087 coprocessor is divided into two major sections:
  - Control Unit (CU)
  - Numerical Execution Unit (NEU)

CONTROL UNIT (CU)
- It interfaces coprocessor to the microprocessor system bus.
- It also synchronize the operation of the coprocessor and the microprocessor.
- This unit has a Control Word, Status Word and Data Buffer.
- If an instruction is ESC instruction, then coprocessor executes it.
- If not, then microprocessor executes.

NUMERIC EXECUTION UNIT (NEU)
- This unit is responsible for executing all coprocessor instructions.
- It has an 8 register stack that holds the operands for instructions and result of instructions.
- The stack contains 8 registers that are 80-bits wide.
- Numeric data is transferred inside the coprocessor in two parts:
  - 64-bit mantissa bus
  - 16-bit exponent bus

STATUS REGISTER
- Status Register tells the overall status of 8087 coprocessor.
- It is a 16-bit register.
- It is accessed by executing the FSTSW instruction.
- This instruction stores the contents of status register into memory.
- Once the status is stored in memory, the bit positions of the status register can be examined.

Busy: It indicates that the coprocessor is busy executing the task.
Condition Codes (C₀-C₅): They indicate various conditions about the coprocessor.
Top of Stack: It indicates a register as top of stack register, out of the eight stack registers.
Exception Flag: It is set if any of the exception flag bits (SF, PR, UF, OF, ZD, DN, IO) are set.
**Stack Register**

- **Stack Fault**: It is not available in 8087. It is active only in 80387 and above.
- **Precision**: It indicates that the result has exceeded the selected precision.
- **Underflow**: It tells if the result is too small to fit in a register.
- **Overflow**: It tells if the result is too large to fit in a register.

**Control Register**

- **Rounding Control**: It determines the type of rounding or truncating to be done.
- **Precision Control**: It sets the precision of the result.
- **Exception Masks**: It determines whether an error effects the exception bits in the status register.
  - If it is one, then the corresponding error is ignored.
  - If it is zero and the corresponding error occurs, then it generates an interrupt, and the corresponding bit in status register is set.

**Tag Register**

- **Zero Divide**: It indicates that you try to divide a non-zero value by zero.
- **Denormalized**: It indicates that at least one of the operands is de-normalized.
- **Invalid Operation**: It indicates an invalid operation. For e.g.: pushing more than eight items onto the stack, attempting to pop an item off an empty stack or taking the square root of a negative number.
**TAG REGISTER**

- Tag Register is used to indicate the contents of each register in the stack.
- There are total 8 tags (Tag 0 to Tag 7) in this register and each tag uses 2 bits to represent a value.
- Therefore, it is a 16-bit register.

**Tag Values:**
- 00 = Valid
- 01 = Zero
- 10 = Invalid
- 11 = Empty

**INTERFACING OF 8086 AND 8087**

- Multiplexed address-data bus lines are connected directly from 8086 to 8087.
- The status lines and the queue status lines are connected directly from 8086 to 8087.
- The Request/Grant (RQ/GT₀ and RQ/GT₁) signals of 8087 are connected to RQ/GT₀ and RQ/GT₁ of 8086.
- BUSY signal of 8087 is connected to TEST pin of 8086.

**Thank You 😊
Have a Nice Day**