# 8087 Coprocessor

# Co-processor – Intel 8087

## Multiprocessor system

- A microprocessor system comprising of two or more processors
- Distributed processing: Entire task is divided in to subtasks

## Advantages

- Better system throughput by having more than one processor
- Each processor have a local bus to access local memory or I/O devices so that a greater degree of parallel processing can be achieved
- System structure is more flexible.
  One can easily add or remove modules to change the system configuration without affecting the other modules in the system

# Co-processor – Intel 8087

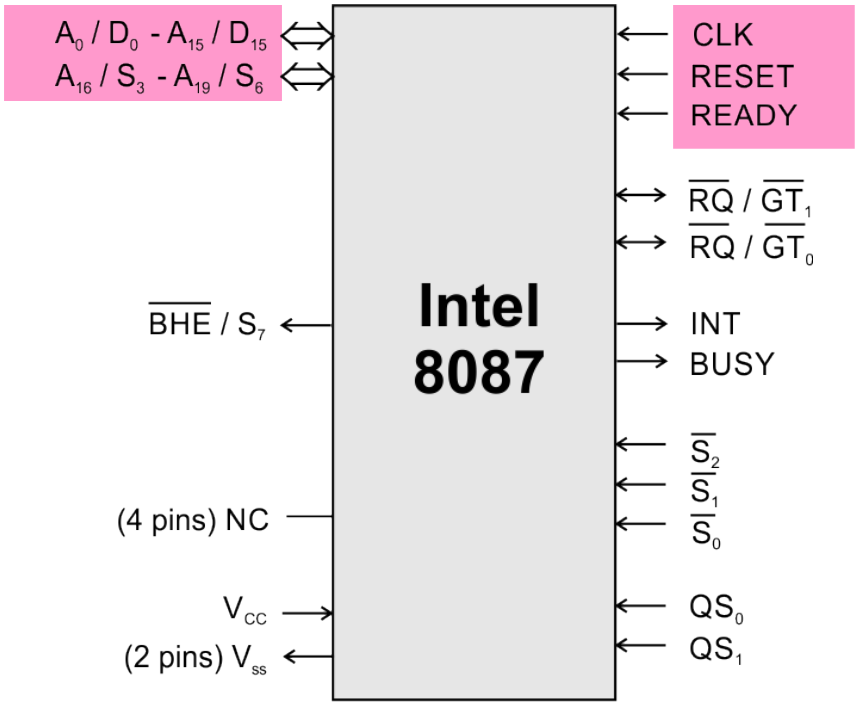
## 8087 coprocessor

- Specially designed to take care of mathematical calculations involving integer and floating point data
- “Math coprocessor” or “Numeric Data Processor (NDP)”
- Works in parallel with a 8086 in the maximum mode

## Features

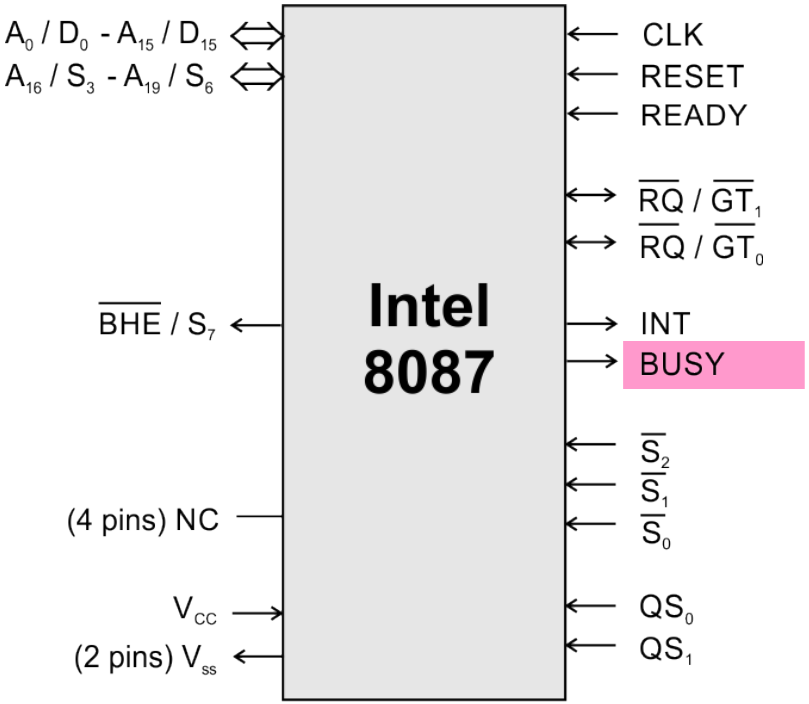
1) Can operate on data of the integer, decimal and real types with lengths ranging from 2 to 10 bytes
2) Instruction set involves square root, exponential, tangent etc. in addition to addition, subtraction, multiplication and division.
3) High performance numeric data processor, it can multiply two 64-bit real numbers and calculate square root
4) Follows IEEE floating point standard
5) It is multi bus compatible

# Co-processor – Intel 8087




- **16 multiplexed address / data pins and 4 multiplexed address / status pins**
- **Hence it can have 16-bit external data bus and 20-bit external address bus like 8086**
- **Processor clock, ready and reset signals are applied as clock, ready and reset signals for coprocessor**

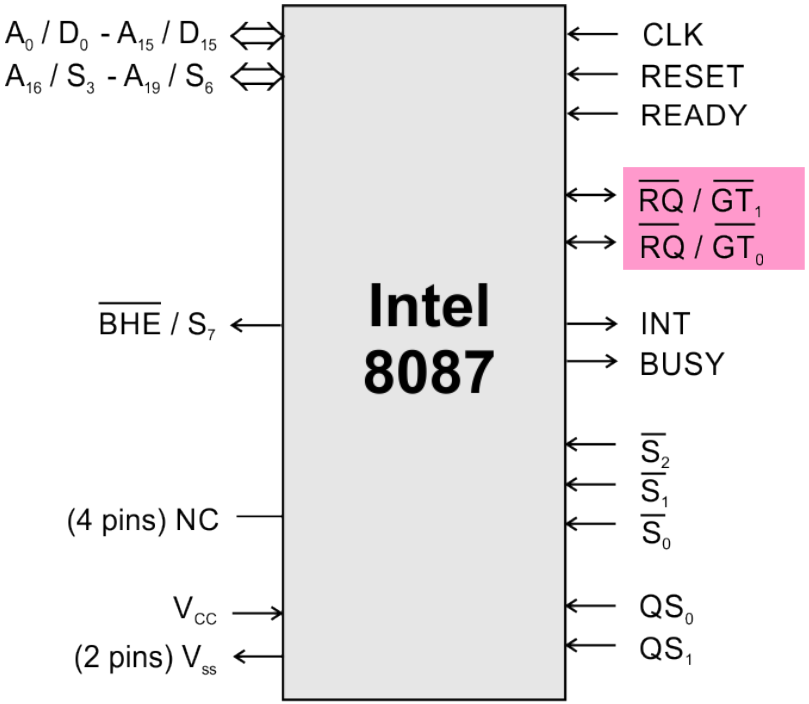
# Co-processor – Intel 8087




## BUSY

- **BUSY signal from 8087 is connected to the $\overline{TEST}$ input of 8086**
- **If the 8086 needs the result of some computation that the 8087 is doing before it can execute the next instruction in the program, a user can tell 8086 with a WAIT instruction to keep looking at its $\overline{TEST}$ pin until it finds the pin low**
- **A low on the BUSY output indicates that the 8087 has completed the computation**

# Co-processor – Intel 8087



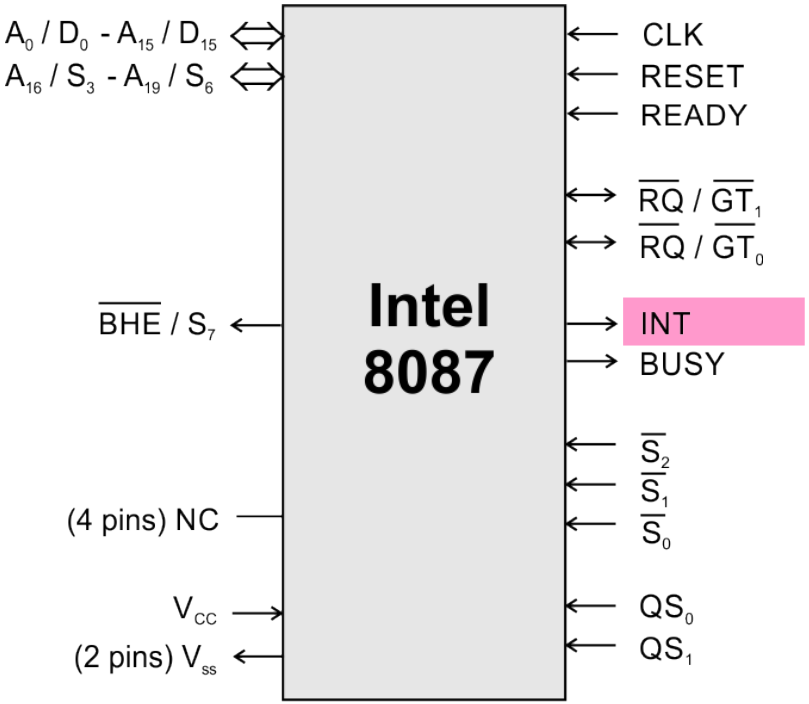

**$\overline{RQ}$ //$\overline{GT_0}$**

- **The request / grant signal from the 8087 is usually connected to the request / grant ($\overline{RQ}$ / $\overline{GT_0}$ or $\overline{RQ}$ / $\overline{GT_1}$) pin of the 8086**

**$\overline{RQ}$ //$\overline{GT_1}$**

- **The request / grant signal from the 8087 is usually connected to the request / grant pin of the independent processor such as 8089**

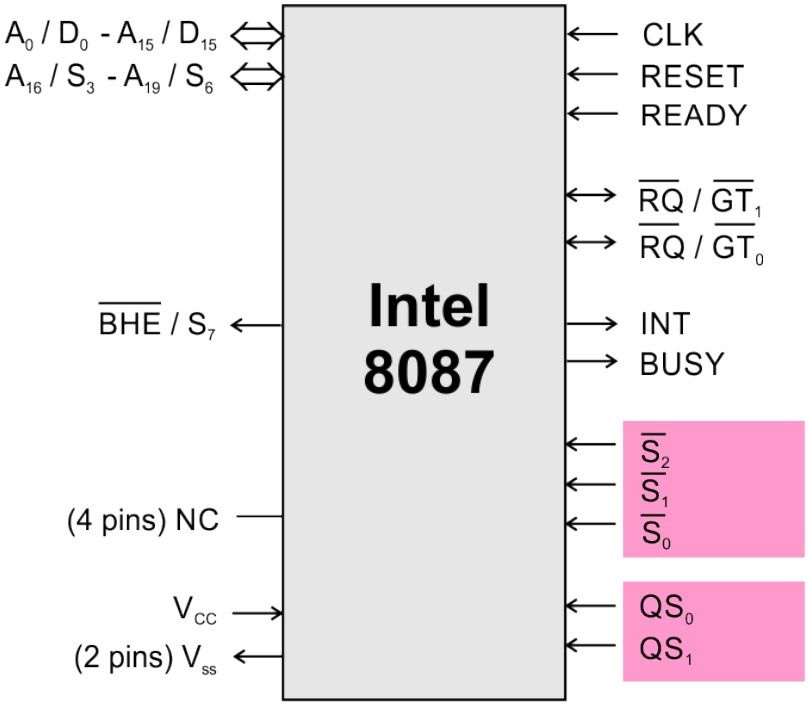
# Co-processor – Intel 8087




## INT

- **The interrupt pin is connected to the interrupt management logic.**
- **The 8087 can interrupt the 8086 through this interrupt management logic at the time error condition exists**

# Co-processor – Intel 8087




$\overline{S_0} = \overline{S_2}$

| $S_2$ | $S_1$ | $S_0$ | Status |
|---|---|---|---|
| 1 | 0 | 0 | Unused |
| 1 | 0 | 1 | Read memory |
| 1 | 1 | 0 | Write memory |
| 1 | 1 | 1 | Passive |

**$QS_0$ – $QS_1$**

| $QS_0$ | $QS_1$ | Status |
|---|---|---|
| 0 | 0 | No operation |
| 0 | 1 | First byte of opcode from queue |
| 1 | 0 | Queue empty |
| 1 | 1 | Subsequent byte of opcode from queue |

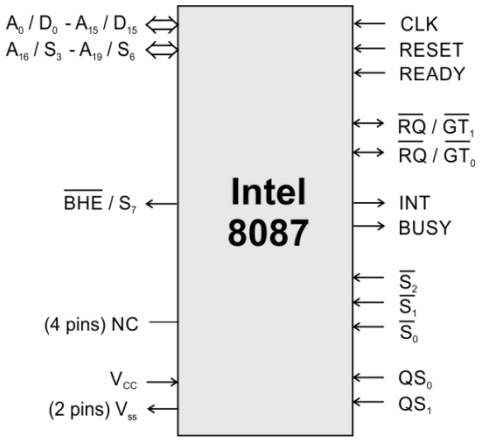
# Co-processor – Intel 8087

- 8087 instructions are inserted in the 8086 program

- 8086 and 8087 reads instruction bytes and puts them in the respective queues
- NOP
- 8087 instructions have 11011 as the MSB of their first code byte

- 8087 keeps track for ESC instruction by monitoring - and $AD_0$ – $AD_{15}$ of 8086.
- Also keeps track of $QS_0$ – $QS_1$.
- Q status 00; does nothing
- Q status 01; 8087 compares the five MSB bits with 11011
- If there is a match, then the ESC instruction is fetched and executed by 8087
- If there is error during decoding an ESC instruction, 8087 sends an interrupt request

- Memory read/ write
- Additional words : $\overline{RQ}$ - $\overline{GT_0}$
- 8087 BUSY pin high
- $\overline{TEST}$
- WAIT



# Co-processor – Intel 8087