COURSE: UCS1502 - MICROPROCESSORS AND INTERFACING

I/O Processor - 8089

Dr. K. R. Sarath Chandran
Assistant Professor, Dept. of CSE

This presentation covers

• Details of I/O processor - 8089

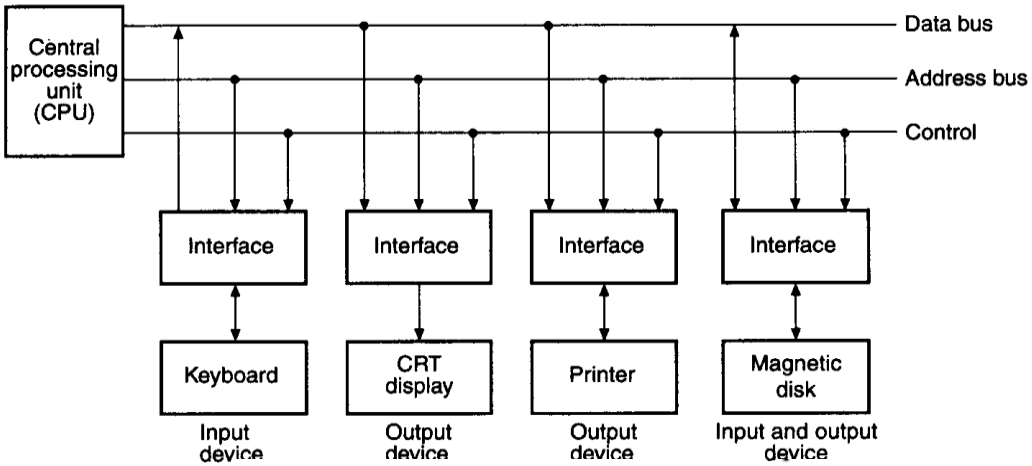
Learning Outcome of this module

• To understand the details of 8089 I/O processor

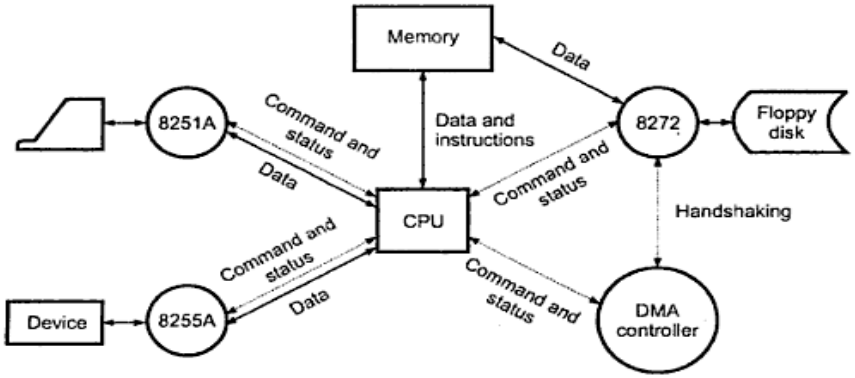
I/O Processors

- I/O Processors handles all of the interactions between the I/O devices and the CPU.
- I/O Processors communicates with input and output devices through separate address, data, and control lines.
  - This provides an independent pathway for the transfer of information between external devices and internal memory.

CPU Connection to I/O Devices



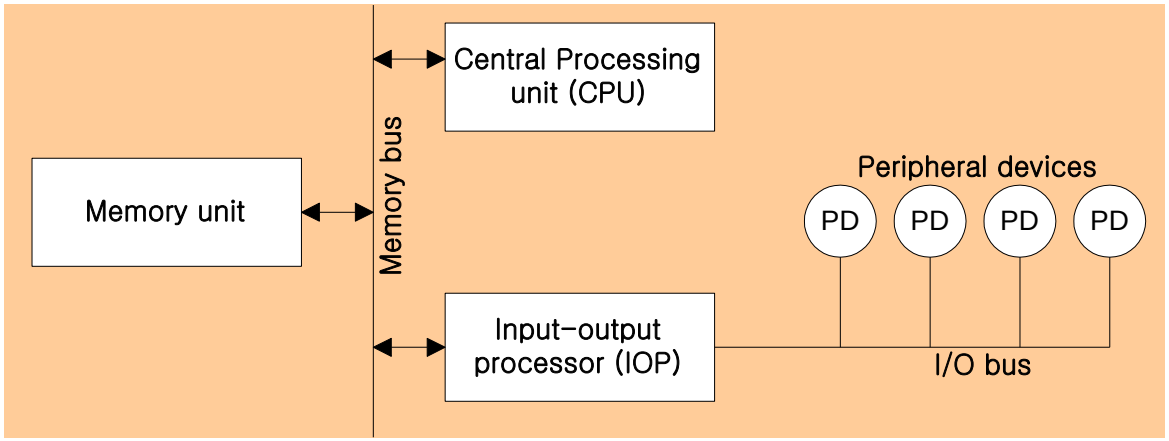
Connection of I/O Devices to CPU



I/O Processors

- Communicate directly with all I/O devices
- Fetch and execute its own instruction
- IOP instructions are specifically designed to facilitate I/O transfer

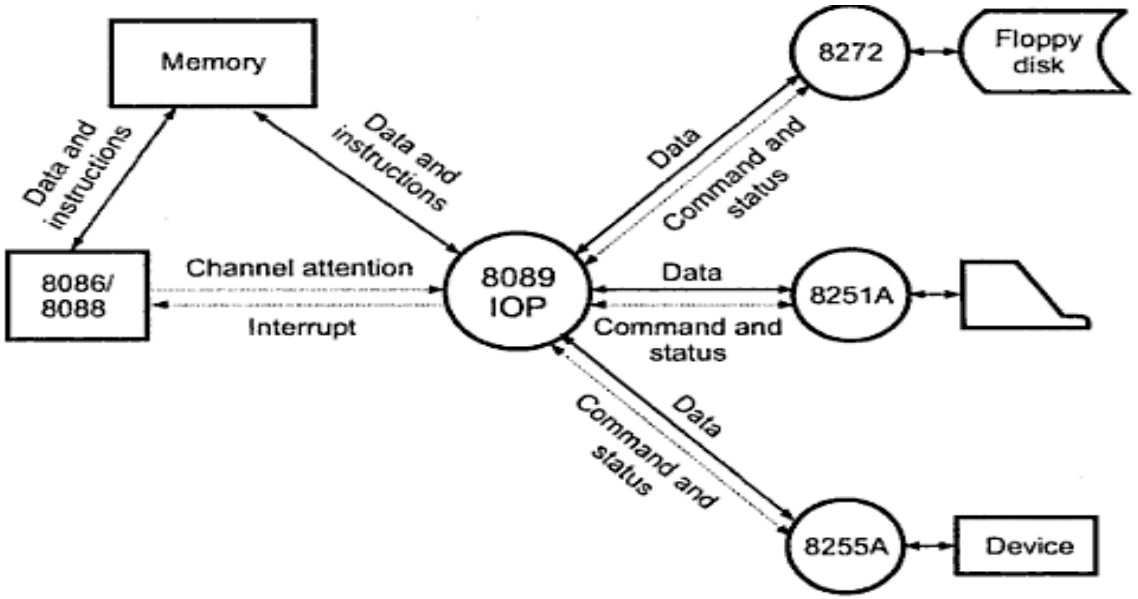
- Used to address the problem of direct transfer after executing the necessary format conversion or other instructions
- In an IOP-based system, I/O devices can directly access the memory without intervention by the processor



I/O Processors

- Three Forms of Commands
- Block transfer commands
  - Moves blocks data to IOP. Usually these instructions swap pages in and out of physical memory, and to load programs from disk memory.
- Arithmetic, logic, and Branch operations
  - IOP uses ALU instructions to manipulate the data so the process time for CPU is shorten.
- Control Command
  - Controls hardware.
    - Eg: Ejecting a CD from a drive.

I/O Processors



I/O handled by IOP

Features

1. An IOP can fetch and execute its own instructions.
2. Instructions are specially designed for I/O processing.
3. In addition to data transfer, 8089 can perform arithmetic and logic operations, branches, searching and translation.
4. IOP does all work involved in I/O transfer including device setup, programmed I/O, and DMA operation.
5. IOP can transfer data from an 8-bit source to 16-bit destination and vice versa.
6. Communication between IOP and CPU is through memory based control blocks. CPU defines tasks in the control blocks to locate a program sequence, called a channel program.

I/O Processors

- CPU communicates with 8089 through memory based **control blocks**
- CPU prepares control blocks that describes the task to be performed and then dispatches the task to IOP through interrupt like signal
- 8089 reads the control block to locate the program sequence called **channel program**
- Channel program is written in 8089's instruction set.
- 8089 performs the assigned task by fetching and executing the instructions from channel program.
- After finishing the execution it interrupt the CPU or update the status location in memory
- 8089 can be operated in loosely coupled or tightly coupled configuration
- In tightly coupled configuration it request the bus through $\overline{RQ}$ / $\overline{GT}$ pins.
- In loosely-coupled, it has its own IO bus and requires bus arbiter and controller.
- 8089 can access I/O address (16 bit) space and system memory address space (20 bit).
- Whether an address is in I/O address space or in system address space is determined by the tag bit of pointer register.

IOP Architecture

- 2 channels
- Each can be programmed and operated independently sharing common control logic and ALU.

CCP: Channel control pointer

- It stores the address of the control block for channel 1 during the initialization.
- Channel 2 CB starts at an address pointed by CCP + 8.

To dispatch a task to the channel

- 8086 sends channel attention signal to the pin CA of 8089 along with SEL signal (channel select).
- if SEL= low, channel 1 is selected
- If a SEL= high, channel 2 is selected.
- Since the channel occupy two consecutive I/O port address, A0 address line is connected to SEL.

IOP Architecture

- Each channel has identical set of registers.
- Two groups based on size : Pointer group, register group

Pointer group

- **20 bit**
- **GA, GB, GC, TP, PP**
- Tag bit indicates whether it has 20 bit or 16 bit address
- If tag bit=0, 20 bit address else 16 bit address
- Except PP, all registers have tag bit.
- PP always points with 20 bit address
- **TP- task pointer** - it stores the address of next instruction to be executed.
- **PP - parameter pointer**
  - It is filled by 8089 at the time of initialization.
  - It points to the address of parameter block.

IOP Architecture

- Each channel has 8 bit Program Status Word (PSW).
- This contains channel status
- Status includes
  1. source and destination address width
  2. channel activity
  3. interrupt control and servicing
  4. bus load limit
  5. Priority
- GA, GB, GC, BC (byte count), IX (index register), MC (mask compare) - General purpose registers in a channel program for ALU operation
- IOP can perform DMA operations.
- It can be from memory to memory, memory to I/O, I/O to I/O.
- For DMA - GA register stores source address,
- GB register stores destination address, BC register stores byte count
- After each by transfer, these registers are updated.
- CC - Channel Control register - it holds the control information about DMA transfer. It contains type of transfer, mode of transfer, when to stop transfer etc.

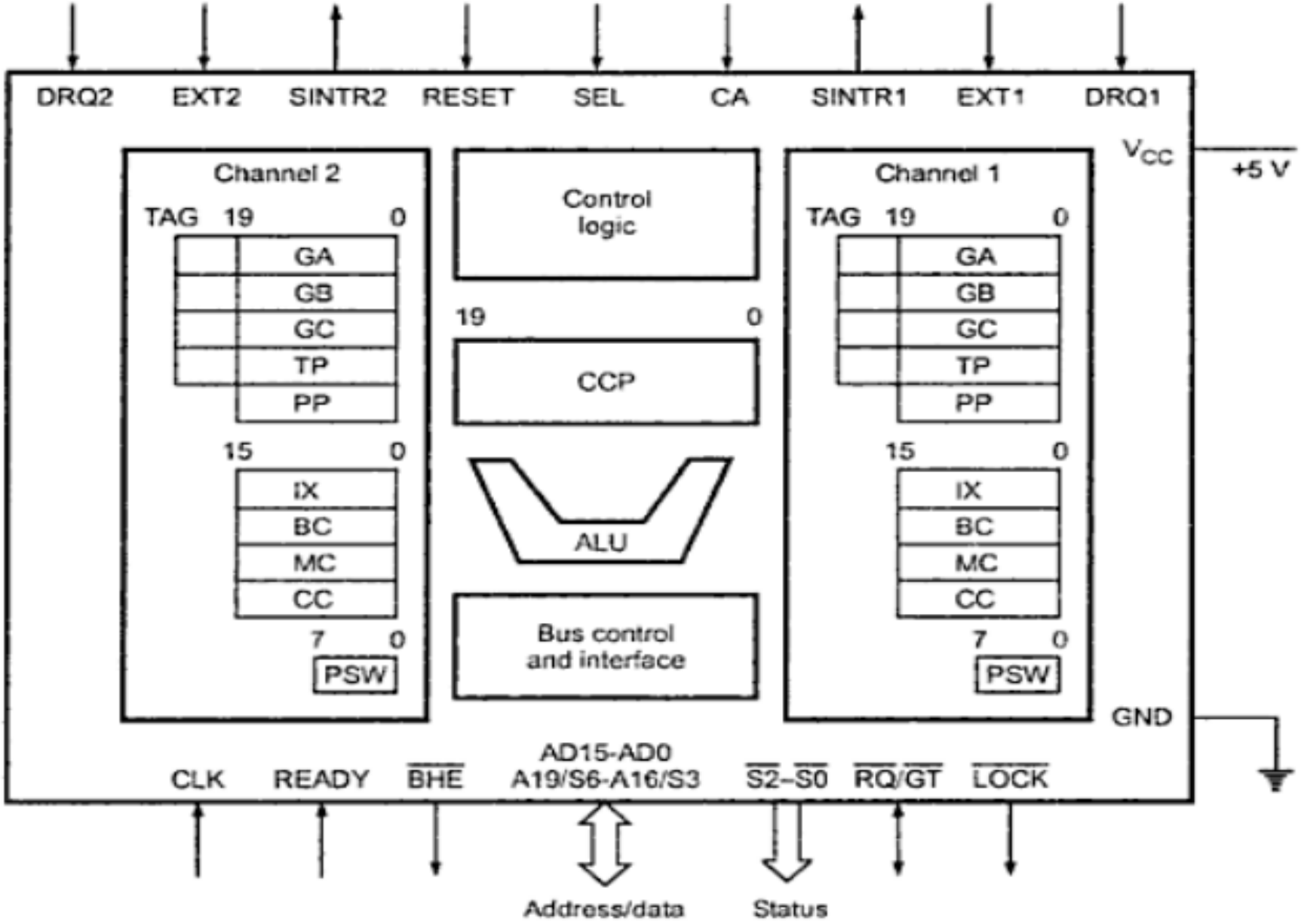
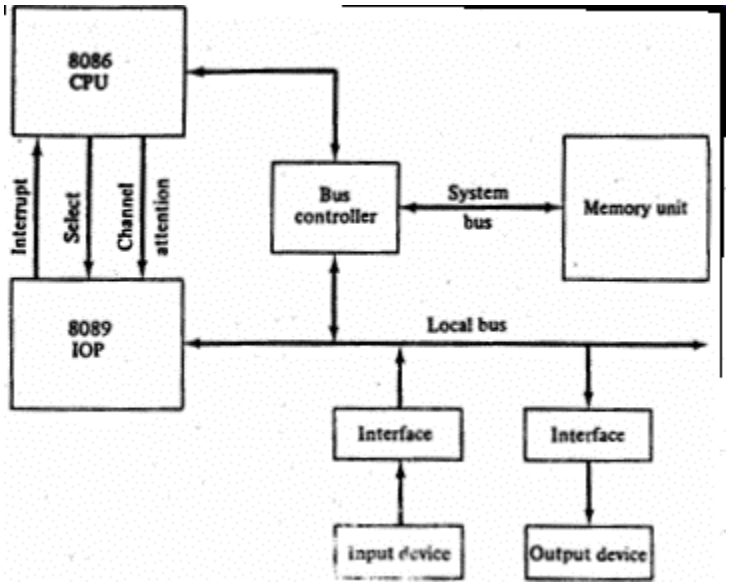
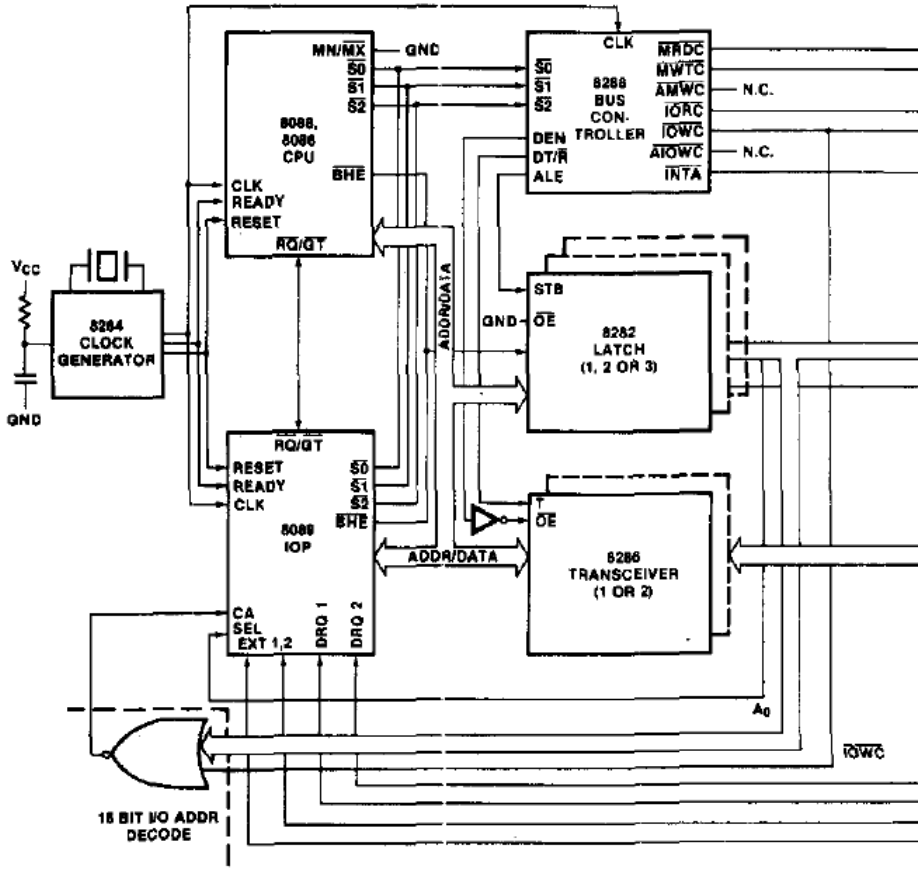
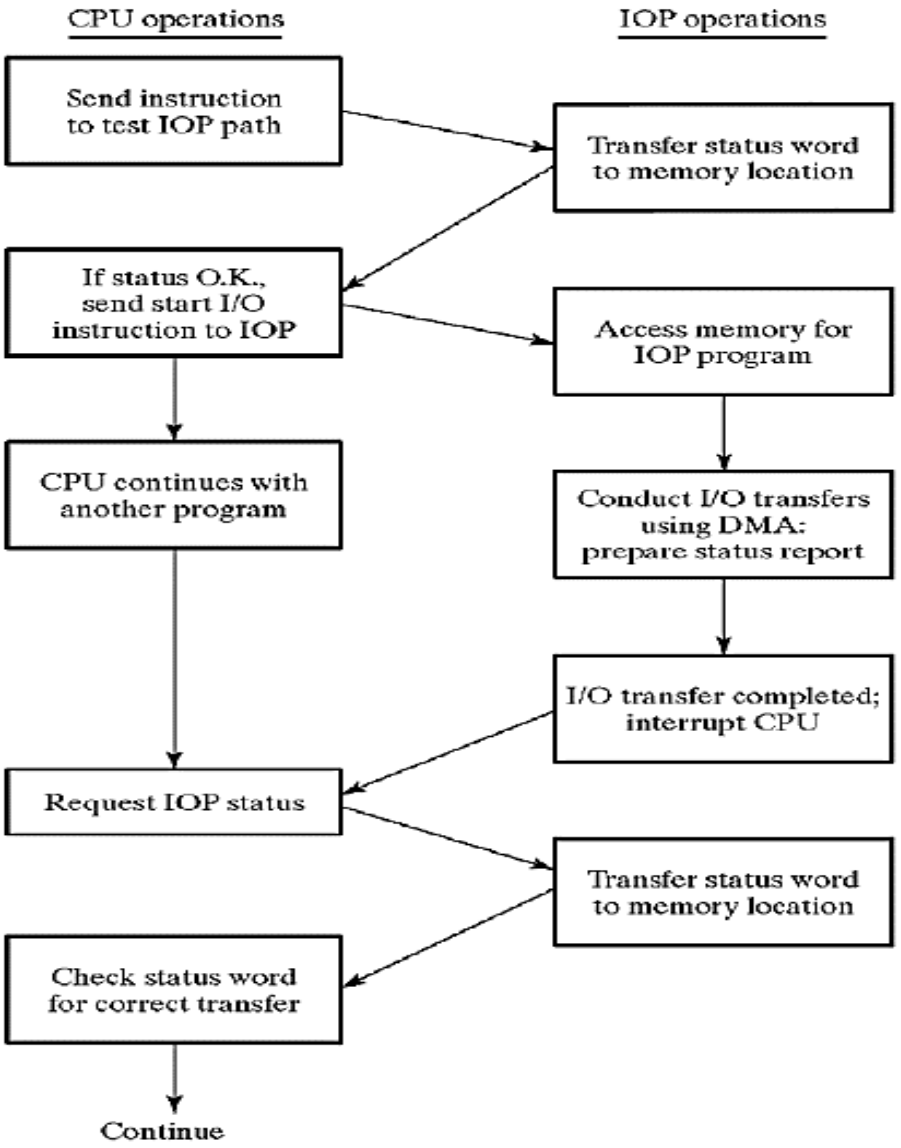


Fig. 8.37 Internal block diagram of 8089

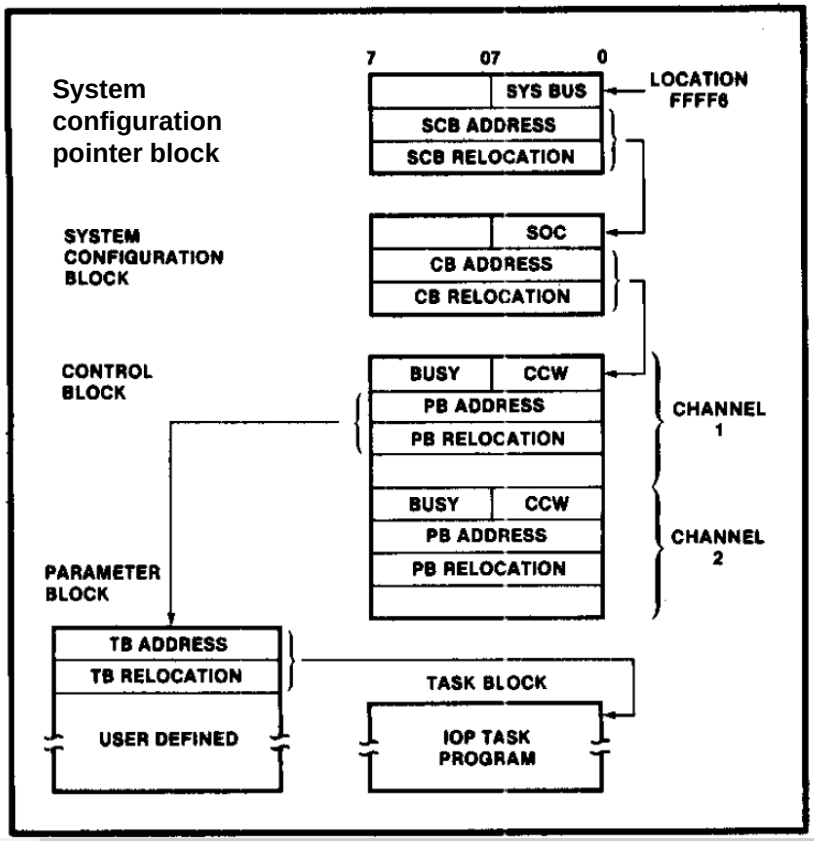
Intel 8086-8089 system design

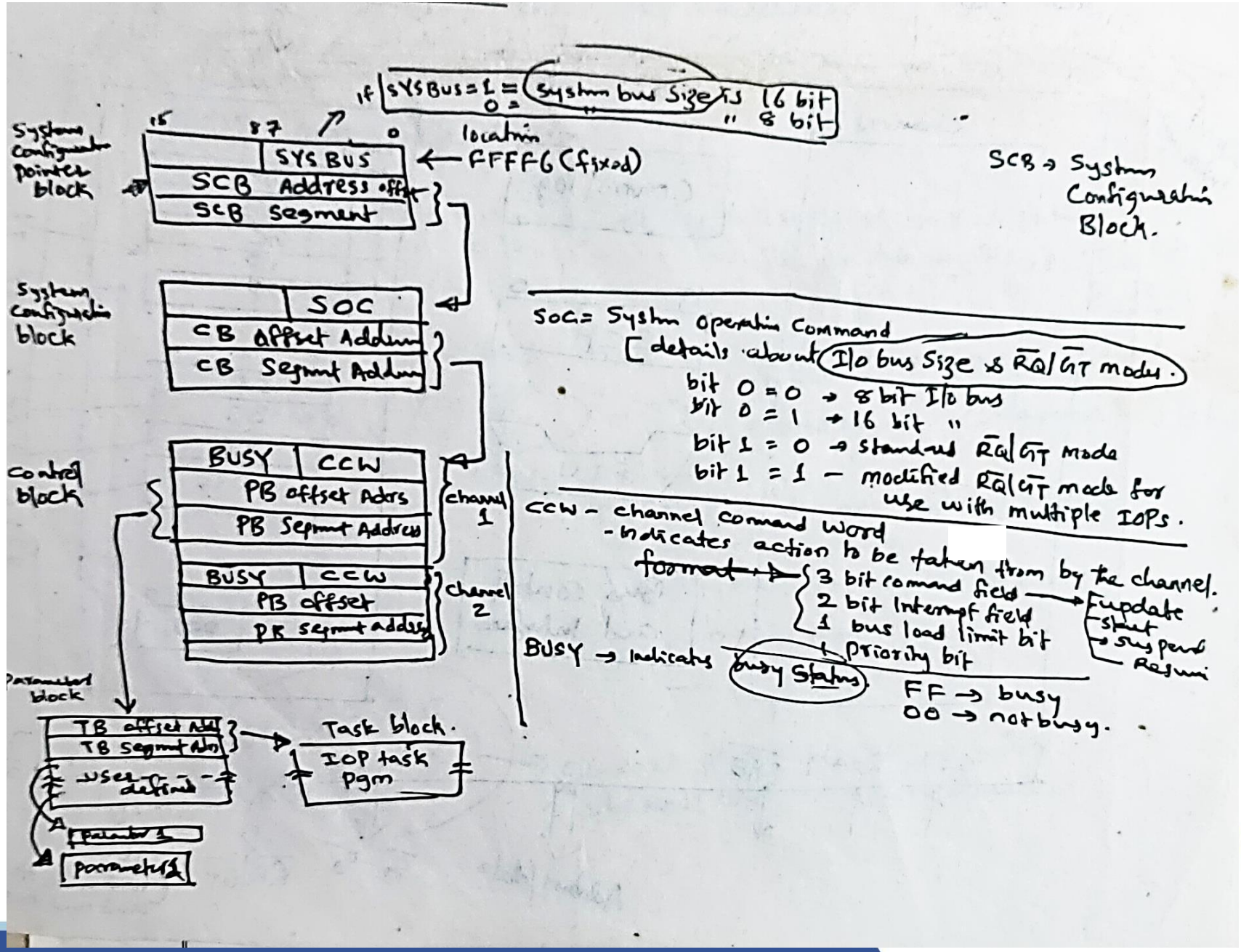


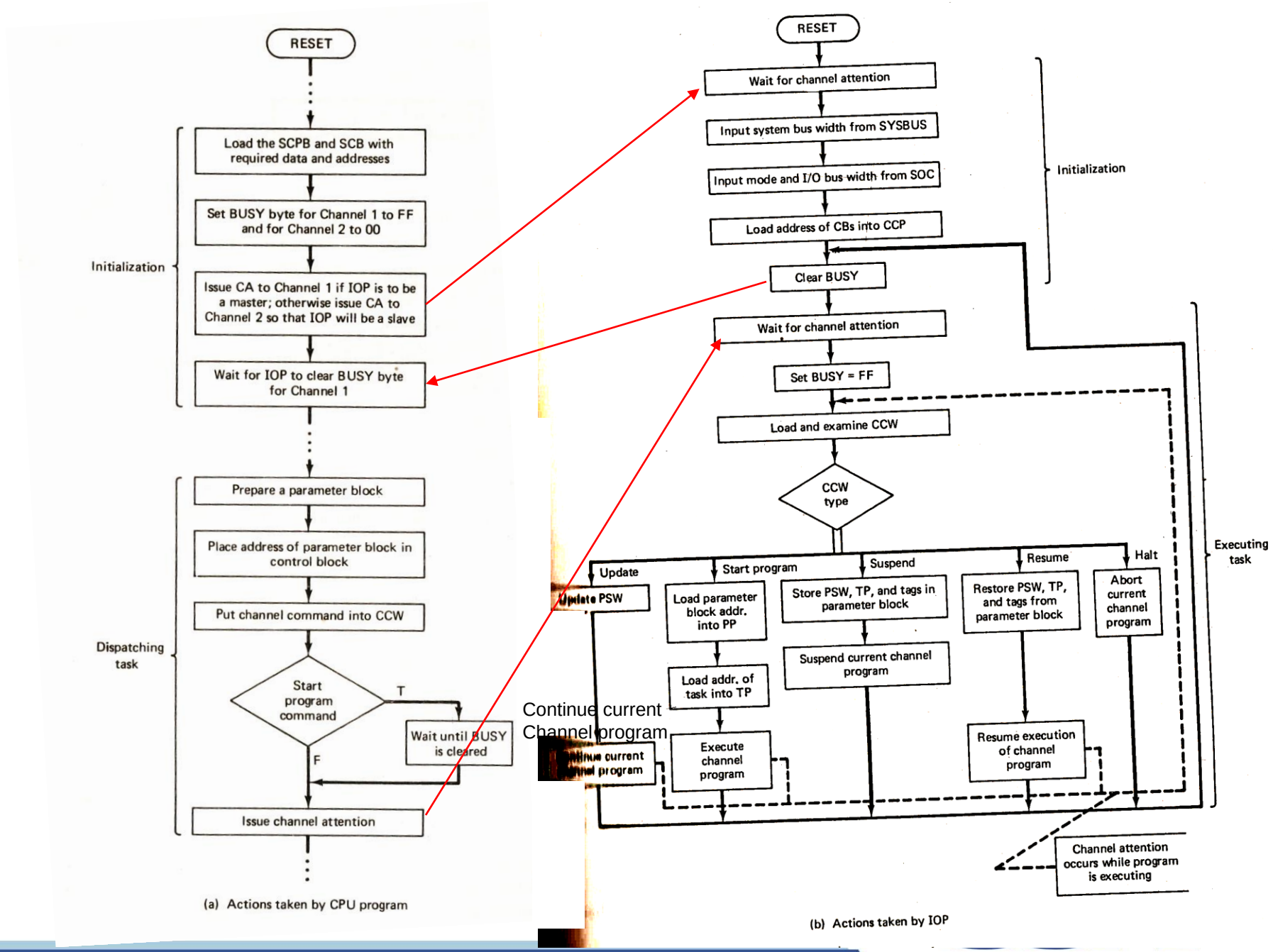




CPU-IOP Communication




if SYSBUS = 1 = System bus size is 16 bit
0 = " 8 bit
location
FFFF6 (fixed)
15 8 7 0
SYS BUS
SCB Address offset
SCB segment
System Configuration pointer block
SCB → System Configuration Block.
SOC
CB offset Addrss
CB Segment Addrss
System configuration block
SOC = System operation Command
[details about I/O bus size & RQ/GT modes.
bit 0 = 0 → 8 bit I/O bus
bit 0 = 1 → 16 bit "
bit 1 = 0 → standard RQ/GT mode
bit 1 = 1 – modified RQ/GT mode for use with multiple IOPs.
BUSY
CCW
PB offset Adrs
PB Segment Address
channel 1
BUSY
CCW
PB offset
PB segment address
channel 2
control block
CCW - channel command word
- indicates action to be taken from by the channel.
format →
3 bit command field → update / start / suspend / resume
2 bit Interrupt field
1 bus load limit bit
1 priority bit
BUSY → indicates busy Status
FF → busy
00 → not busy.
Parameter block
TB offset Addr
TB segment Addr
user defined
Parameter 1
Parameter 2
Task block.
IOP task pgm



Continue current Channel program

Thank you