DEVICES AND COMMUNICATION BUSES FOR DEVICES NETWORK

Lesson-1: IO port types- Serial and parallel IO ports

1. IO Port

Port

A port is a device

- to receive the bytes from external peripheral(s) [or device(s) or processor(s) or controllers] for reading them later using instructions executed on the processor or
- to send the bytes to external peripheral or device or processor using instructions executed on processor

Port

- A Port connects to the processor using address decoder and system buses
- The processor uses the addresses of the port-registers for programming the port functions or modes, reading port status and for writing or reading bytes.

Example

- SI serial interface in 8051
- SPI serial peripheral interface in 68HC11
- PPI parallel peripheral interface 8255
- Ports P0, P1, P2 and P3 in 8051 or PA, PB, PC and PD in 68HC11
- COM1 and COM2 ports in an IBM PC

2. IO Port Types

Types of Serial ports

- Synchronous Serial Input
- Synchronous Serial Output
- Asynchronous Serial UART input
- Asynchronous Serial UART output
- Both as input and as output, for example, modem.

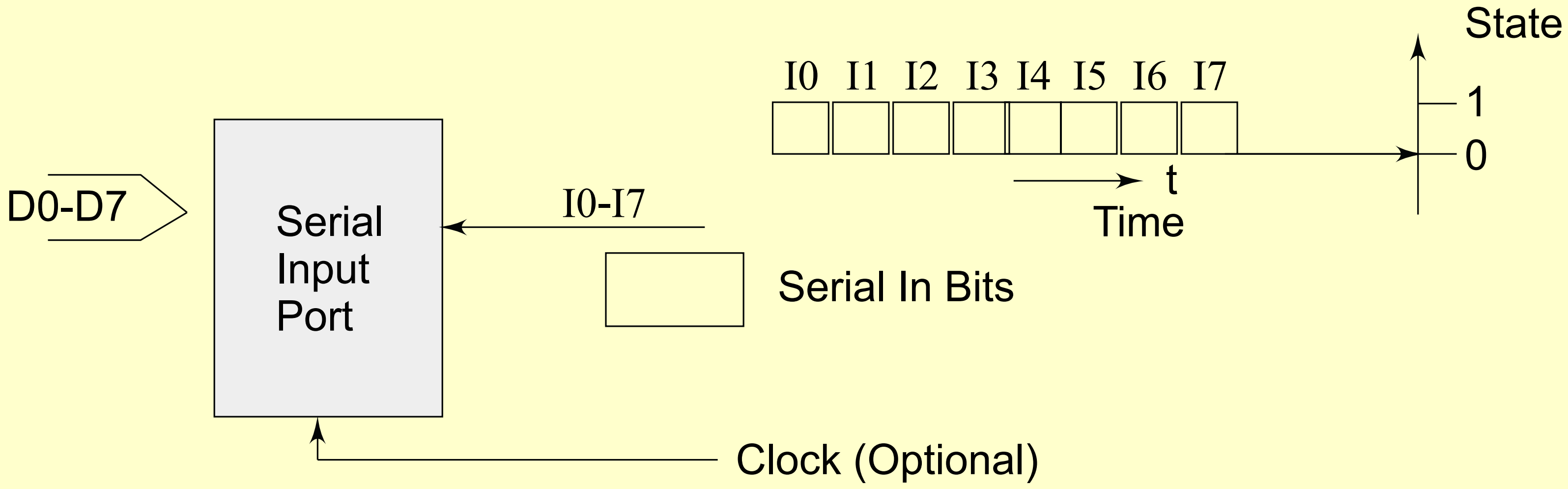
Types of parallel ports

- Parallel port one bit Input
- Parallel one bit output
- Parallel Port multi-bit Input
- Parallel Port multi-bit Output

Synchronous Serial Input Example

- Inter-processor data transfer, reading from CD or hard disk, audio input, video input, dial tone, network input, transceiver input, scanner input, remote controller input, serial I/O bus input, writing to flash memory using SDIO (Secure Data Association IO based card)

Synchronous Serial Input Device (Serial Bits and a clock signal used for synchronisation of a port input)



Synchronous Serial Input

- The sender along with the serial bits also sends the clock pulses SCLK (serial clock) to the receiver port pin. The port synchronizes the serial data-input bits with clock bits. Each bit in each byte as well as each byte in synchronization
- Synchronization means separation by a constant interval or phase difference. If clock period = T, then each byte at the port is received at input in period = 8T.
- The bytes are received at constant rates. Each byte at input port separates by 8T and data transfer rate for the serial line bits is (1/T) bps. [1bps = 1 bit per s]

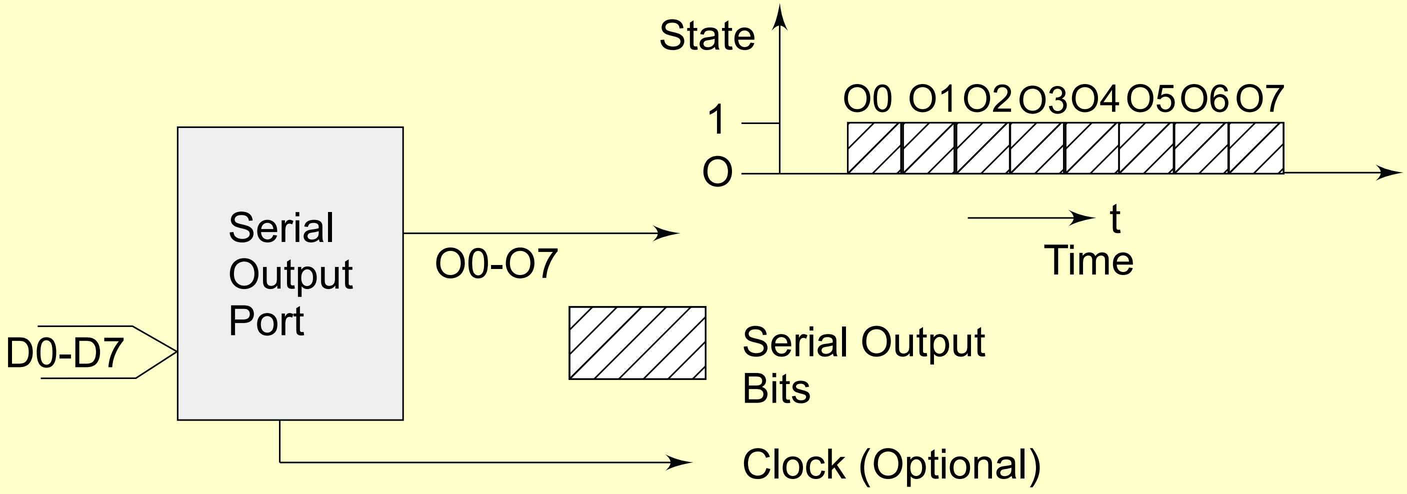
Serial data and clock pulse-inputs

- On same input line – when clock pulses either encode or modulate serial data input bits suitably. Receiver detects the clock pulses and receives data bits after decoding or demodulating.
- On separate input line – When a separate SCLK input is sent, the receiver detects at the middle or + ve edge or –ve edge of the clock pulses that whether the data-input is 1 or 0 and saves the bits in an 8-bit shift register. The processing element at the port (peripheral) saves the byte at a port register from where the microprocessor reads the byte.

Master output slave input (MOSI) and Master input slave output (MISO)

- MOSI when the SCLK is sent from the sender to the receiver and slave is forced to synchronize sent inputs from the master as per the inputs from master clock.
- MISO when the SCLK is sent to the sender (slave) from the receiver (master) and slave is forced to synchronize for sending the inputs to master as per the master clock outputs.
- Synchronous serial input is used for inter-processor transfers, audio inputs and streaming data inputs.

Synchronous Serial Output Device (Device Serial Bits and synchronisation clock signal at a port output)



Example Synchronous Serial Output

- Inter-processor data transfer, multiprocessor communication, writing to CD or hard disk, audio Input/output, video Input/output, dialer output, network device output, remote TV Control, transceiver output, and serial I/O bus output or writing to flash memory using SDIO

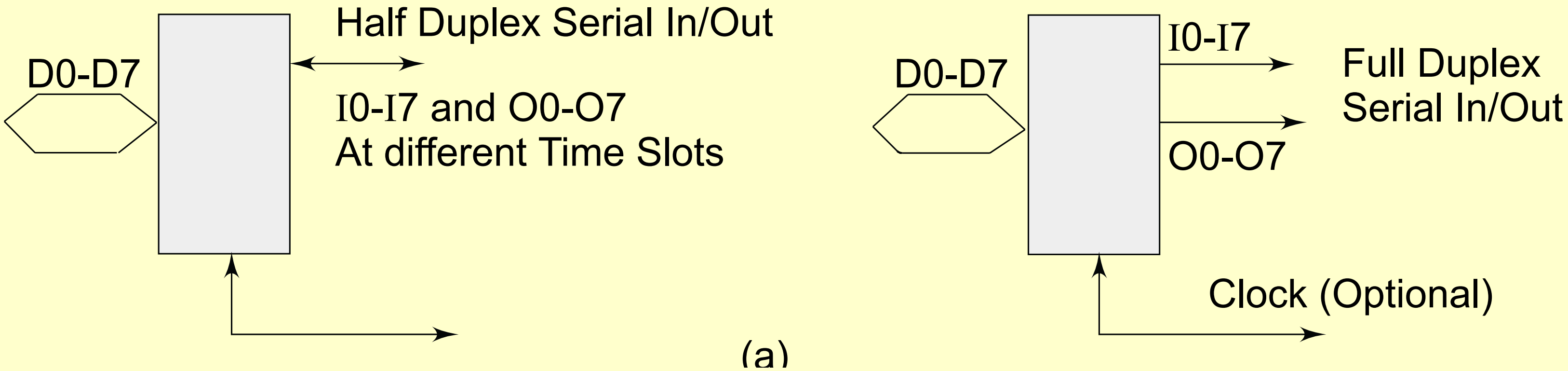
Synchronous Serial Output

- Each bit in each byte sent in synchronization with a clock.
- Bytes sent at constant rates. If clock period = T, then data transfer rate is (1/T) bps.
- Sender either sends the clock pulses at SCLK pin or sends the serial data output and clock pulse-input through same output line with clock pulses either suitably modulate or encode the serial output bits.

Synchronous serial output using shift register

- The processing element at the port (peripheral) sends the byte through a shift register at the port to where the microprocessor writes the byte.
- Synchronous serial output is used for inter-processor transfers, audio outputs and streaming data outputs.

Synchronous Serial Input/Output

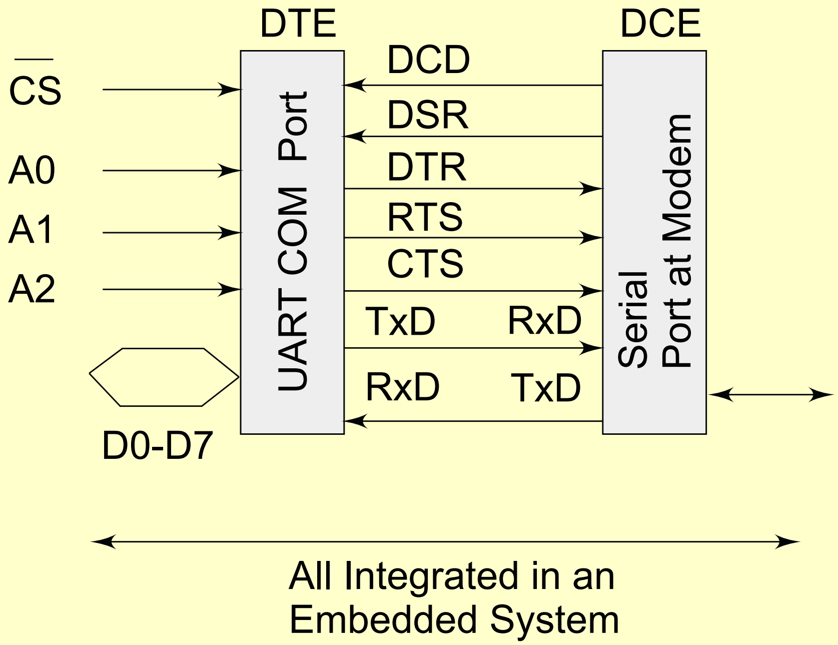


(a)

Synchronous Serial Input/Output

- Each bit in each byte is in synchronization at input and each bit in each byte is in synchronization at output with the master clock output .
- The bytes are sent or received at constant rates. The I/Os can also be on same I/O line when input/output clock pulses either suitably modulate or encode the serial input/output, respectively. If clock period = T, then data transfer rate is (1/T) bps.
- The processing element at the port (peripheral) sends and receives the byte at a port register to or from where the microprocessor writes or reads the byte

Asynchronous Serial input RxD at UART COM Port



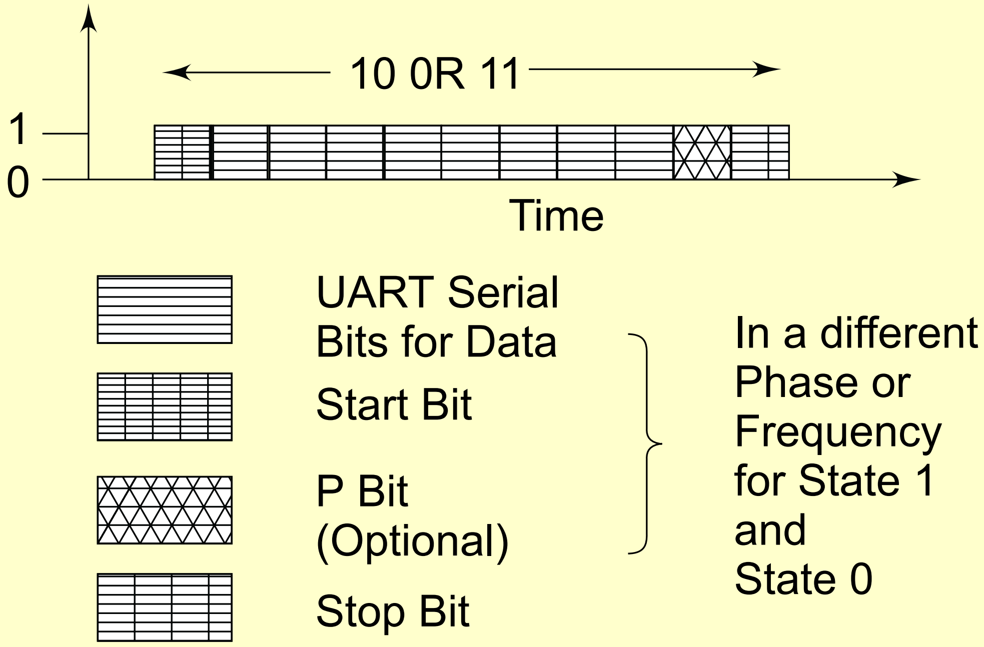
Asynchronous Serial port line RxD (receive data).

- Does not receive the clock pulses or clock information along with the bits.
- Each bit is received in each byte at fixed intervals but each received byte is not in synchronization.
- Bytes separate by the variable intervals or phase differences
- Asynchronous serial input also called UART input if serial input is according to UART protocol

Example Serial Asynchronous Input

- Asynchronous serial input is used for keypad inputs and modem inputs in computers
- Keypad controller serial data-in, mice, keyboard controller, modem input, character send inputs on serial line [also called UART (universal receiver and transmitter) input when according to UART mode]

Format of bits at UART protocol



UART Serial Bits for Data

Start Bit

P Bit (Optional)

Stop Bit

In a different Phase or Frequency for State 1 and State 0

UART protocol serial line format

- Starting point of receiving the bits for each byte is indicated by a line transition from 1 to 0 for a period = T. [T^{-1} called baud rate.]
- If sender's shift-clock period = T, then a byte at the port is received at input in period = 10.T or 11.T due to use of additional bits at start and end of each byte.

UART protocol serial line format

- Receiver detects *n* bits at the intervals of T from the middle of the start indicating bit. The *n* = 0, 1, …, 10 or 11 and finds whether the data-input is 1 or 0 and saves the bits in an 8-bit shift register.
- Processing element at the port (peripheral) saves the byte at a port register from where the microprocessor reads the byte.

Asynchronous Serial Output

- Asynchronous_output serial port line TxD (transmit data).
- Each bit in each byte transmit at fixed intervals but each output byte is not in synchronization (separates by a variable interval or phase difference). Minimum separation is 1 stop bit interval

TxD

- Does not send the clock pulses along with the bits.
- Sender transmits the bytes at the minimum intervals of *n*.T. Bits receiving starts from the middle of the start indicating bit,
- n = 0, 1, …, 10 or 11 and sender sends the bits through a 10 or 11 -bit shift register.

TxD

- The processing element at the port (peripheral) sends the byte at a port register to where the microprocessor is to write the byte.
- Synchronous serial output is also called UART output if serial output is according to UART protocol

Example Serial Asynchronous Output

- Output from modem, output for printer, the output on a serial line [also called UART output when according to UART]

Half Duplex

- Half duplex means as follows: at an instant communication can only be one way (input or output) on a bi-directional line.
- An example of half-duplex mode— telephone communication. On one telephone line, the talk can only in the half-duplex way mode.

Full Duplex

- Full duplex means that at an instant, the communication can be both ways. An example of the full duplex asynchronous mode of communication is the communication between the modem and the computer though TxD and RxD lines or communication using SI in modes 1, 2 and 3 in 8051

Parallel Port single bit input

- Completion of a revolution of a wheel,
- Achieving preset pressure in a boiler,
- Exceeding the upper limit of permitted weight over the pan of an electronic balance,
- Presence of a magnetic piece in the vicinity of or within reach of a robot arm to its end point and
- Filling of a liquid up to a fixed level.

Parallel Port Output- single bit

- PWM output for a DAC, which controls liquid level, or temperature, or pressure, or speed or angular position of a rotating shaft or a linear displacement of an object or a d.c. motor control
- Pulses to an external circuit
- Control signal to an external circuit

Parallel Port Input- multi-bit

- ADC input from liquid level measuring sensor or temperature sensor or pressure sensor or speed sensor or d.c. motor rpm sensor
- Encoder inputs for bits for angular position of a rotating shaft or a linear displacement of an object

Parallel Port Output- multi-bit

- LCD controller for Multilane LCD display matrix unit in a cellular phone to display on the screen the phone number, time, messages, character outputs or pictogram bit-images for display screen or e-mail or web page
- Print controller output
- Stepper-motor coil driving bits

Parallel Port Input-Output

- PPI 8255
- Touch screen in mobile phone

Summary

We learnt

- Port definition
- Port examples
- Ports in embedded system connects to external processors, controllers or devices like keypad, multi-line display unit, printer or modem
- A serial or parallel I/O port connects and accessed from and to the system-processor through the system buses and address decoder.

We learnt

- Synchronous serial inputs and outputs
- Synchronization Clock, MOSI, MISO
- Clock on separate line or encoded or clock signal modulated or encoded with I/O bits.

We learnt

- Asynchronous serial inputs and outputs
- RxD and TxD and UART protocol format asynchronous IOs
- full duplex or half-duplex
- Examples of serial and parallel IOs
- Examples of single bit and multi-bit IOs at parallel ports

End of Lesson 1 of Chapter 3