

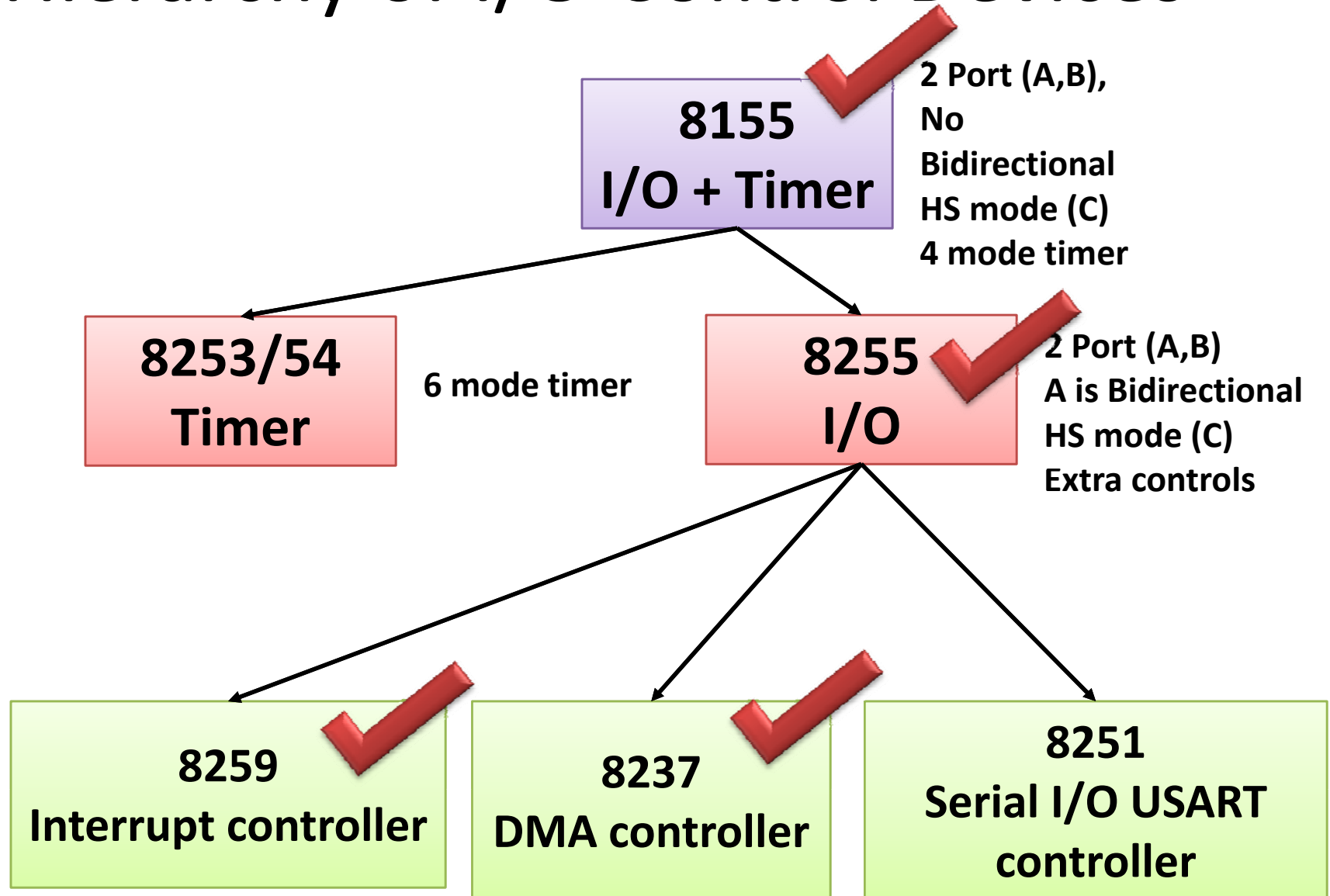
# Programming 8251 USART Controller

Dr A Sahu

Dept of Comp Sc & Engg.

IIT Guwahati

# Hierarchy of I/O Control Devices

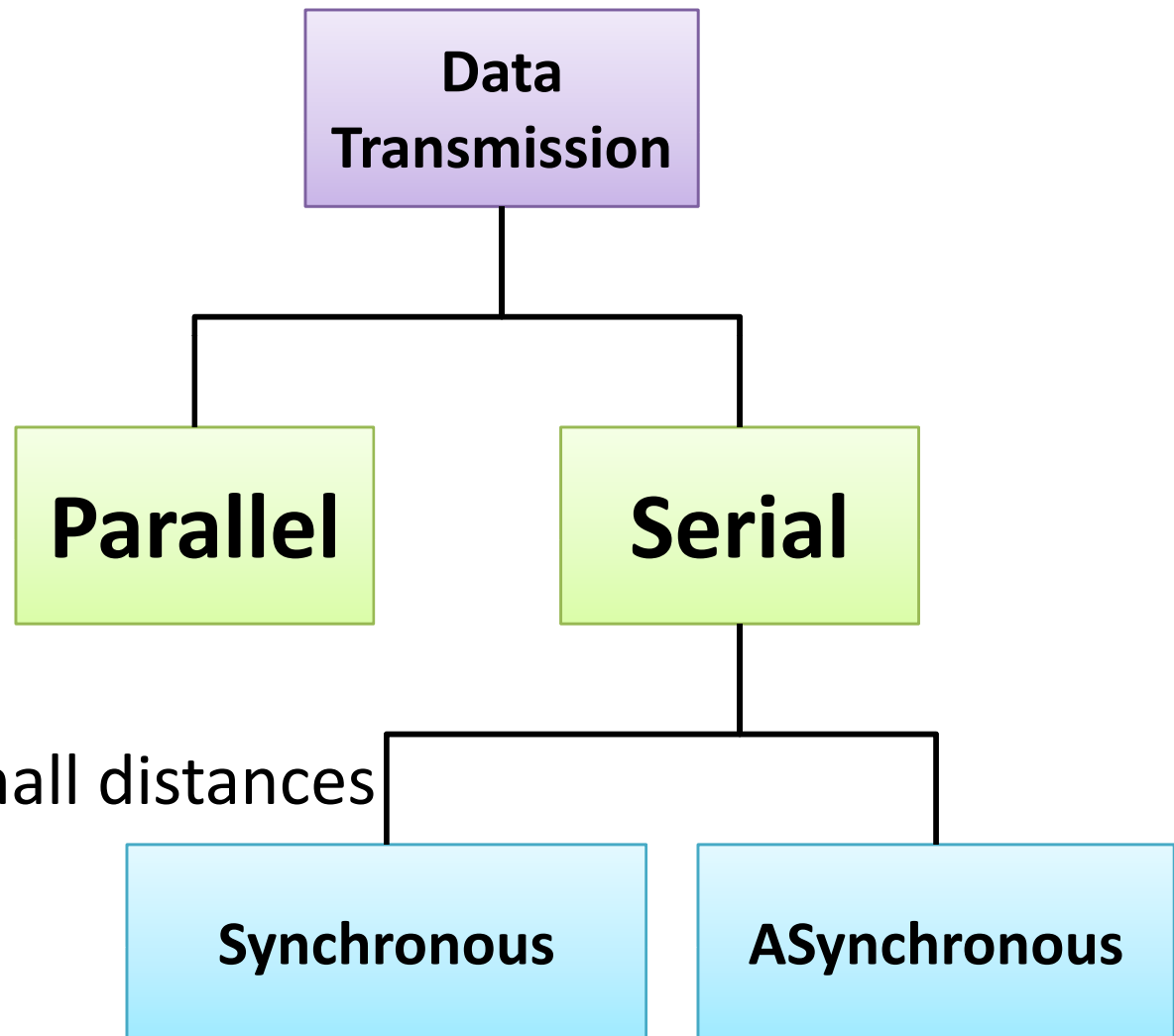


# Outline

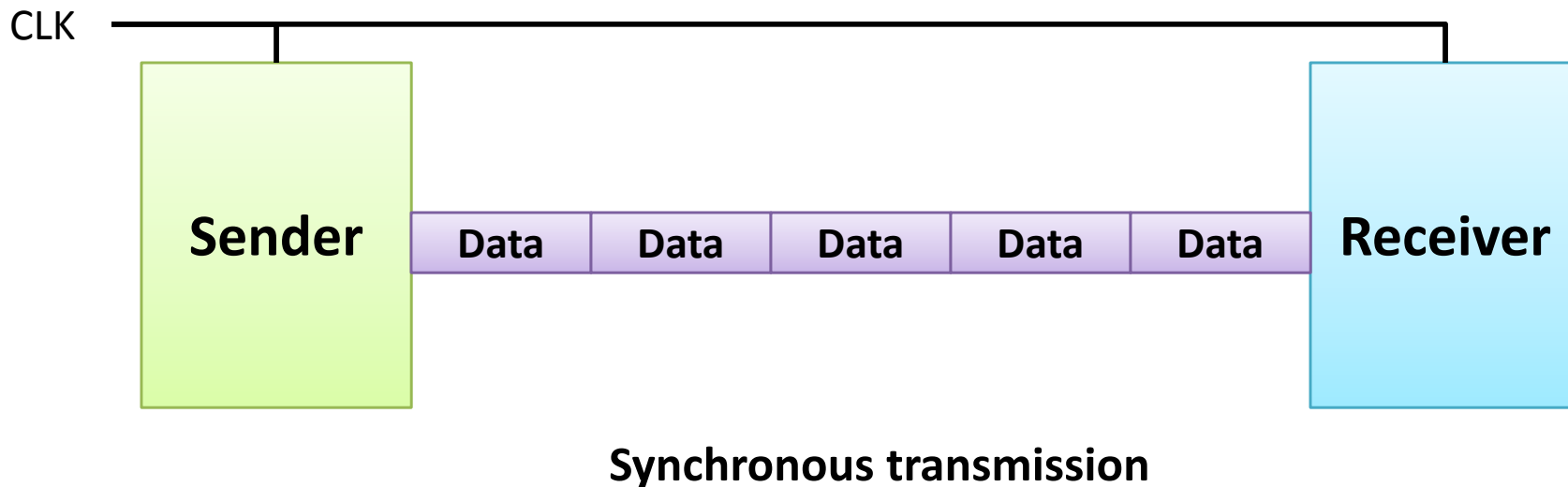
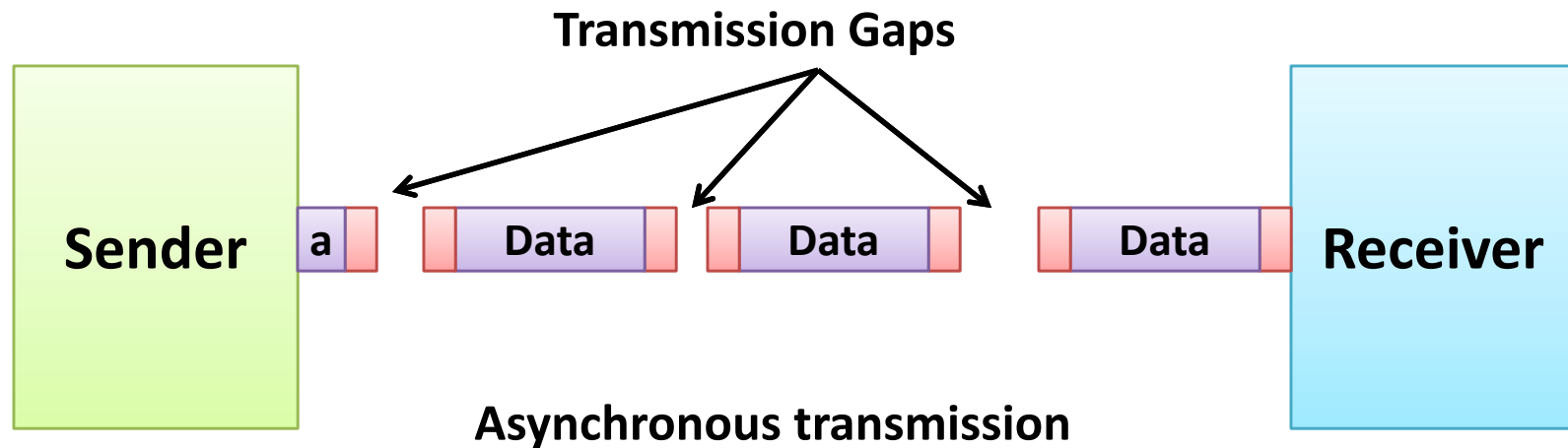
- Asynchronous Communication
- 8251 USART Architecture
- USART Registers
- Programming UART
- RS 232 Port
- Interfacing CRT Monitor using a UART and RS-232 port

# Data Comm: Serial Vs Parallel

- Serial
  - Cheaper
  - Slower
- Parallel
  - Faster
  - Data skew
  - Limited to small distances



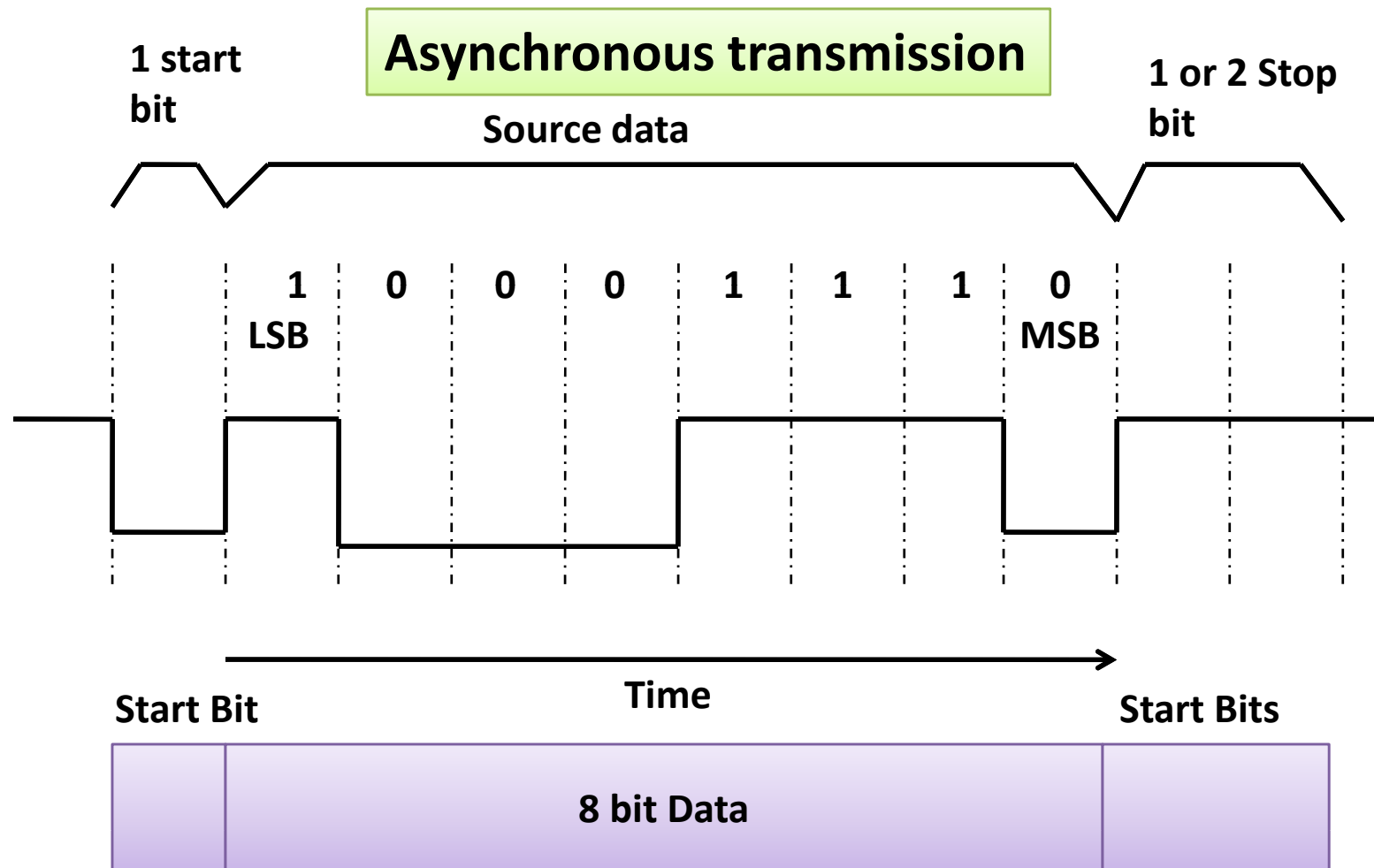
# Type of Serial Communication



# Framing in Asynchronous

- Character oriented
- Each character carried start bit and stop bits
- When No data are being transmitted
  - Receiver stay at logic 1 called mark, logic 0 is Space
- Framing:
  - Transmission begins with one start bit (low/0)
  - Followed by DATA (8bit) and
  - Stop bits (1 or 2 bits of logic high)

## Type of Serial Communication



# 8085 Serial I/O lines

- Serial Input Data (SID)
- Serial Output Data (SOD)
  - Instruction SIM is necessary to output data
  - Interpretations (ACC contents)

D7	D6	D5	D4	D3	D2	D1	D0
SOD	SDE (0/1 Dis/Ena SOD)	X	For interrupts				

MVI	A, 80	; Set D <sub>7</sub> in the ACC=1
RAR		;Set D <sub>6</sub> =1 and bring carry into D <sub>7</sub>
SIM		; output D <sub>7</sub>



# Data transmission Program on SOD

- Transmit an ASCII Char stored in Register B

```

MVI    B ASCIIDataByte    ; get data byte in B
MVI    C,0BH              ; set up counter for 11 bits
XRA    A                  ; reset carry to 0
NXTbit: MVI    A,80H        ;set D7=1 in ACC
RAR                      ;bring Carry in D7 and set D6=1
SIM                      ;output D7
CALL    DELAYBittime       ;wait for fixed time (BWT)
STC                      ;set Carry 1
MOV     A,B                ;Place ASCII car in acc
RAR                      ; place ASCII D0 in Carry
                        ;and shift 1 in D7

MOV     B,A                ;Save B
DCR     C
JNZ     NXTbit
RET
```

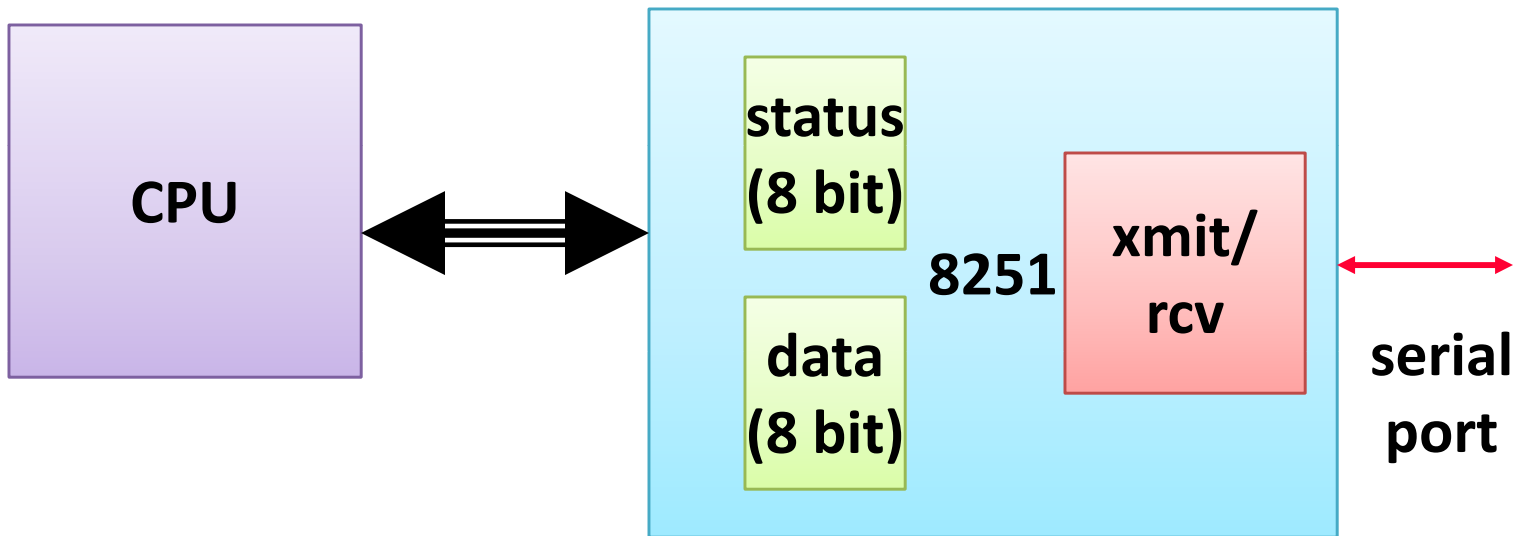
# Hardware control Serial I/O

- Programmable chip 8251
- Requirement of HW control serial I/O
  - An input/output port are required for interfacing
  - Converts data bits in to Parallel to serial & vice versa
  - Data transfer to be synchronized between I/O
  - USART (Universal Synchronous Asynchronous Receiver and Transmitter )

# UART/USART

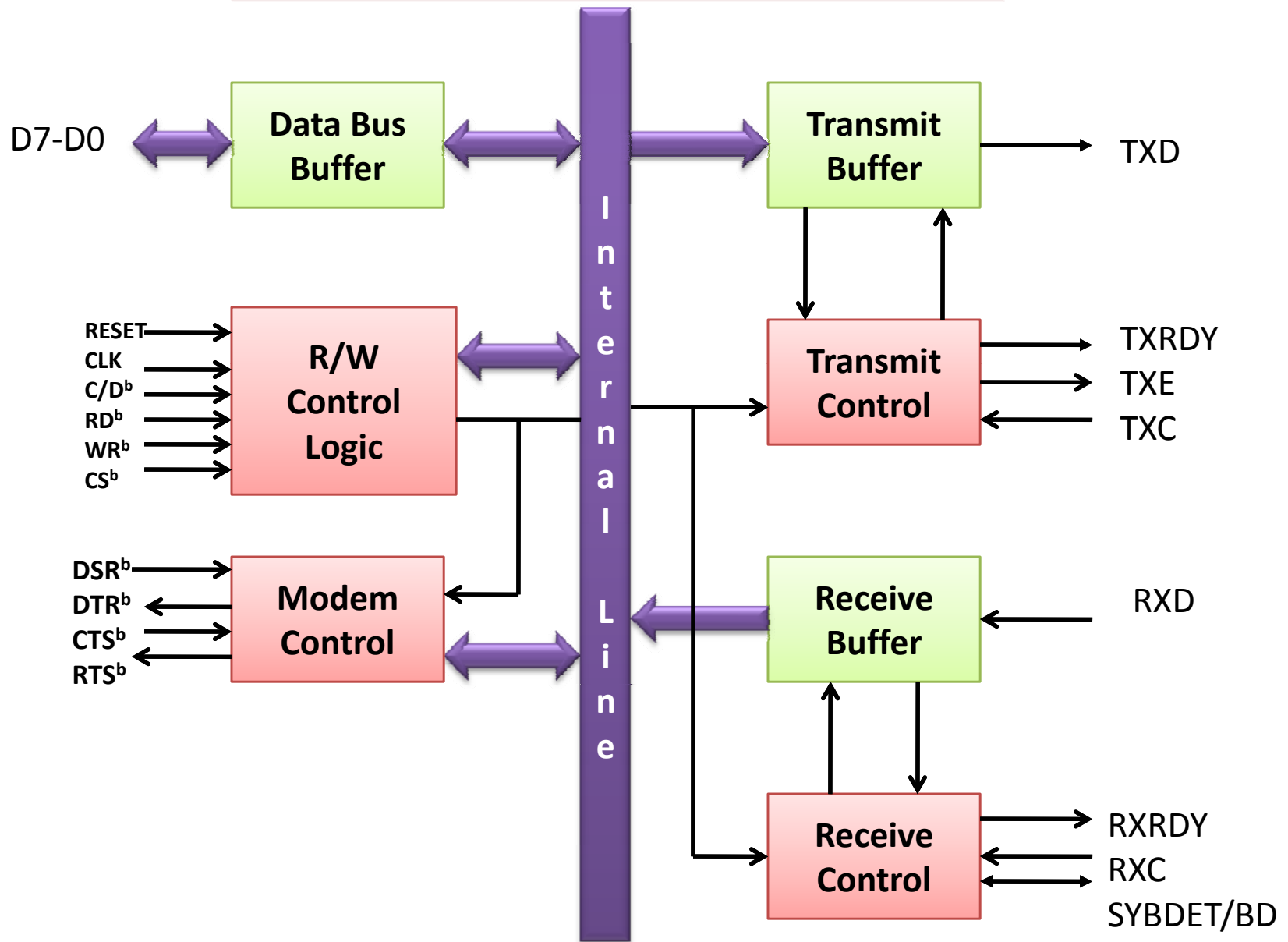
- Writing a program compatible with all different serial communication protocols is difficult and it is an inefficient use of microprocessor.
- UART: Universal Asynchronous Receiver/Transmitter chip.
- USART: Universal Synchronous/Asynchronous Receiver/Transmitter chip.
- The microprocessor sends/receives the data to the UART in parallel, while with I/O, the UART transmits/receive data serially.
- 8251 functions are integrated into standard PC interface chip.

# UART / CPU interface



- UART/USART
  - 8251 USART
  - 8250/16450 UART is a newer version of 8251.
  - 16550 is the latest version UART.

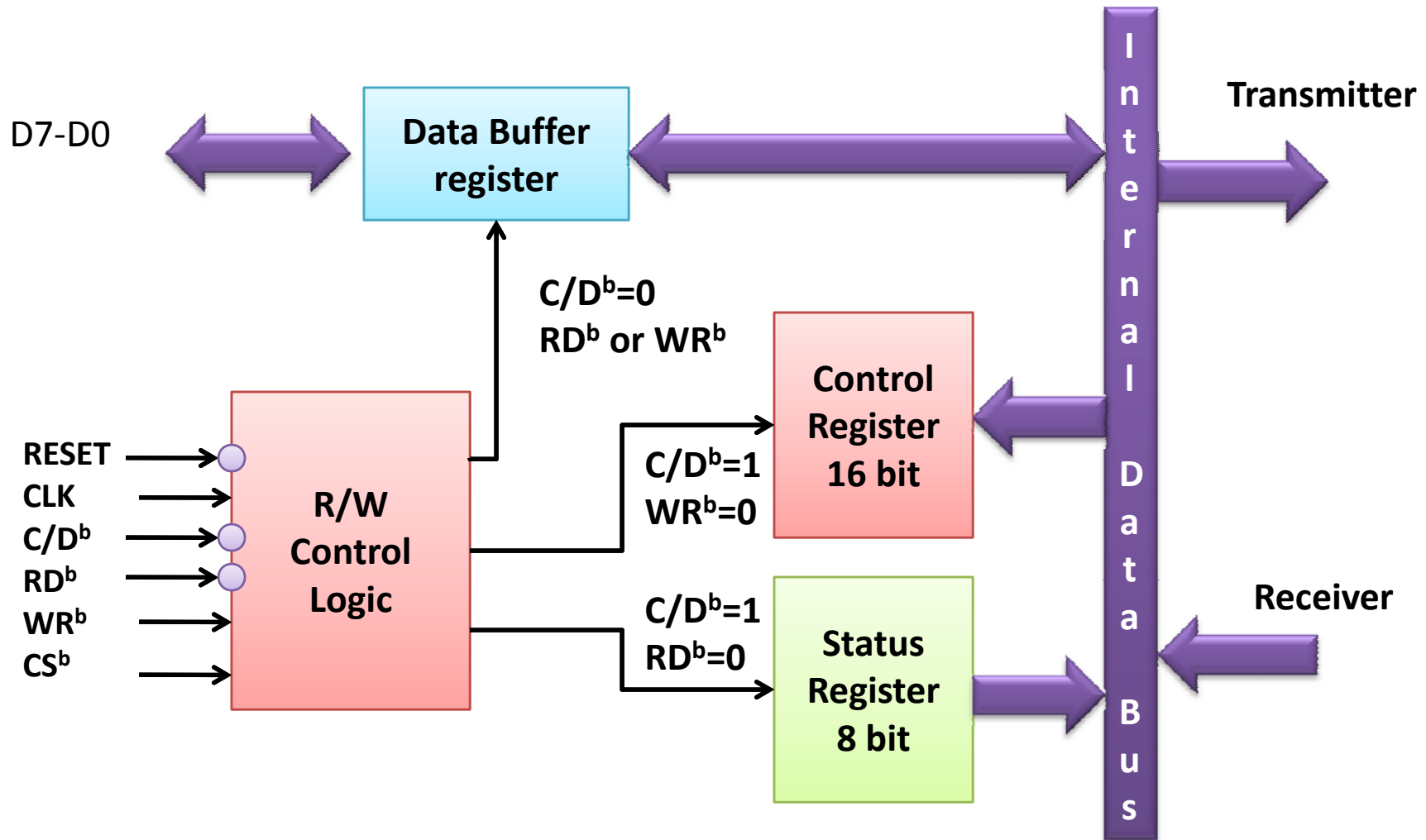
# 8251 Block Diagram



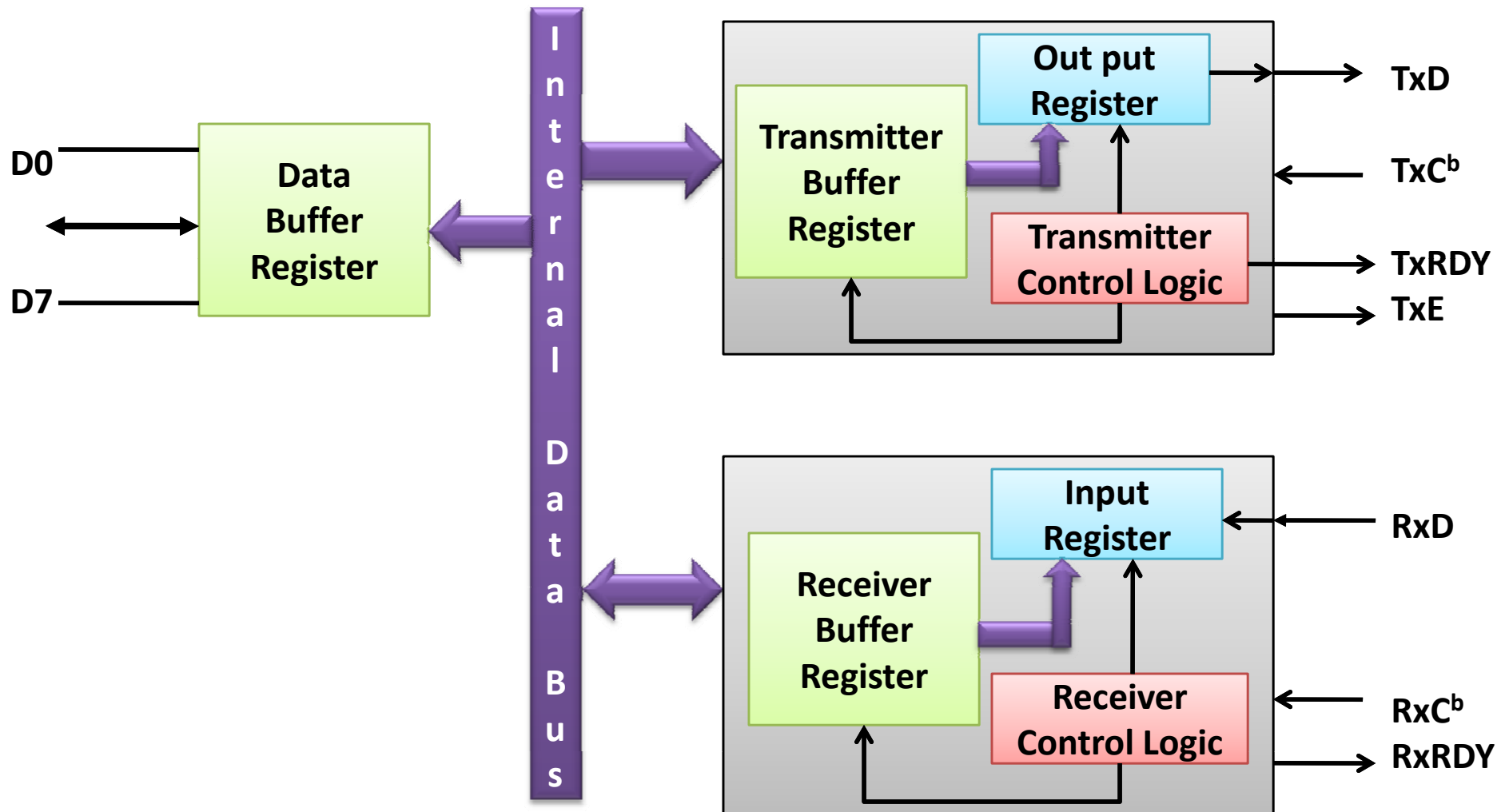
# 8251 Registers

CS <sup>b</sup>	C/D <sup>b</sup>	RD <sup>b</sup>	WR <sup>b</sup>	Meaning
1	X	X	X	<i>Data Bus Tri-state</i>
0	X	1	1	Data Bus Tri-state
0	1	0	1	<b>Status → CPU</b>
0	1	1	0	<b>Control Word ← CPU</b>
0	0	0	1	Data → CPU (accept data from Data Buffer)
0	0	1	0	Data ← CPU (Out put data to Data buffer)

# Control Logic & Registers

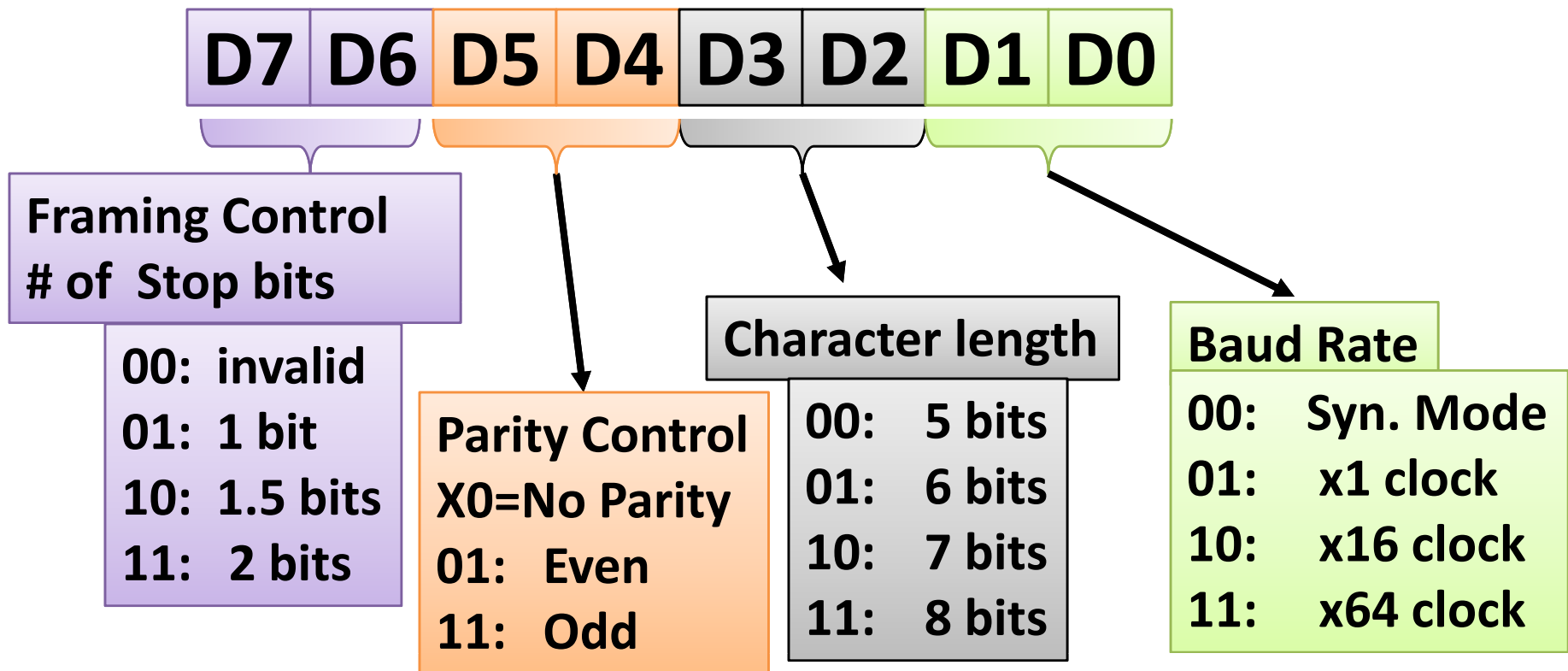


# Transmitter and Receiver





# 8251: Command Register (Mode word format)



# Command Register (Command Word Format)

<b>EH</b>	<b>IR</b>	<b>RTS</b>	<b>ER</b>	<b>SBRK</b>	<b>RxE</b>	<b>DTR</b>	<b>TxE</b>
-----------	-----------	------------	-----------	-------------	------------	------------	------------

TxE: transmit enable (0/1 Enable Disable)

DTR: data terminal ready (1=ENABLE DTR)

RxE: receiver enable (1/0=EN/DISABLE)

SBPRK: send break character 1= force TxD low

ER: error reset (Reset Flags: Parity ,Over run,  
Framing Error of Status Word)

RTS: request to send (1= Enable Request to send)

IR: internal reset (Reset 8251 to mode)

EH: enter hunt mode (1=search for Sync Character)

## 8251: Status Register

<b>DSR</b>	<b>SYN DET</b>	<b>FE</b>	<b>OE</b>	<b>PE</b>	<b>Tx EMPTY</b>	<b>RxRDY</b>	<b>TxRDY</b>
------------	--------------------	-----------	-----------	-----------	---------------------	--------------	--------------

TxRDY      transmit ready (DB Buffer is empty)

RxRDY      receiver ready

TxEMPTY    transmitter empty

PE          parity error (1=when PE detected)

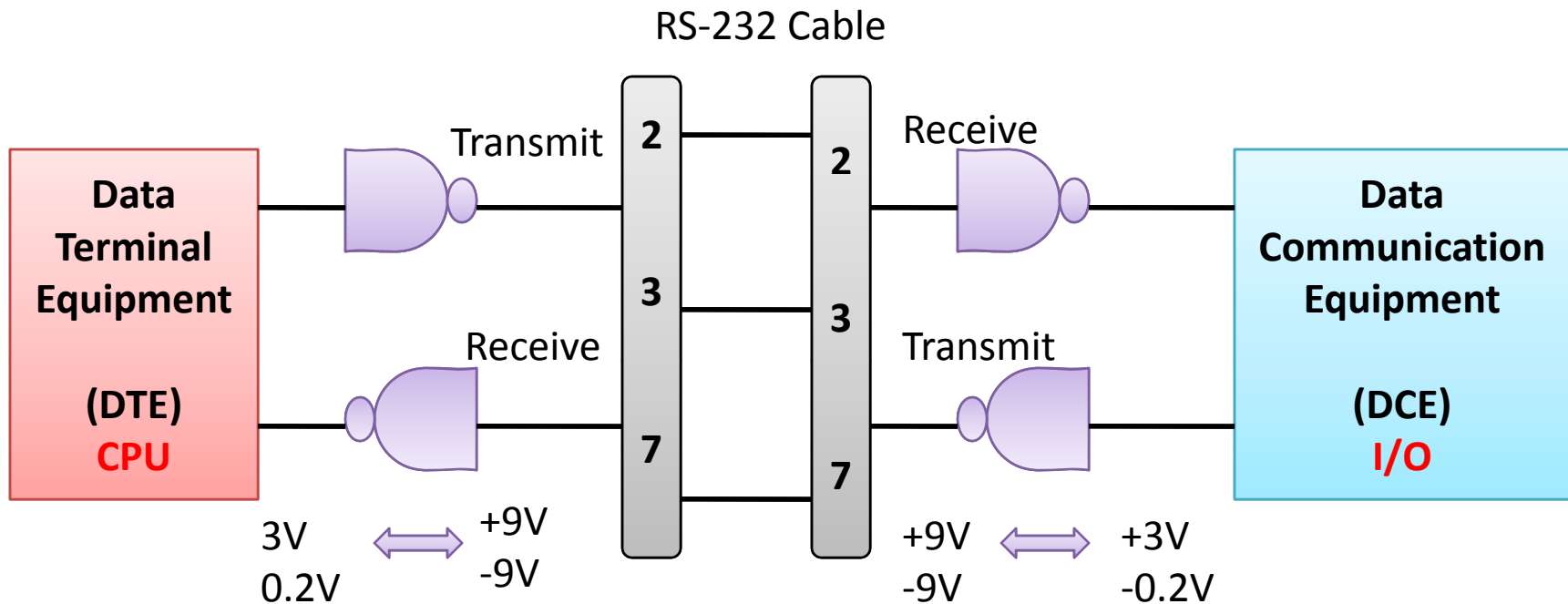
OE          overrun error

FE          framing error (Aynsc only, Valid stop bit  
not detected)

SYNDET    sync. character detected

DSR        data set ready (DSR set at 0 level)

# RS 232 (Recommended Standard)



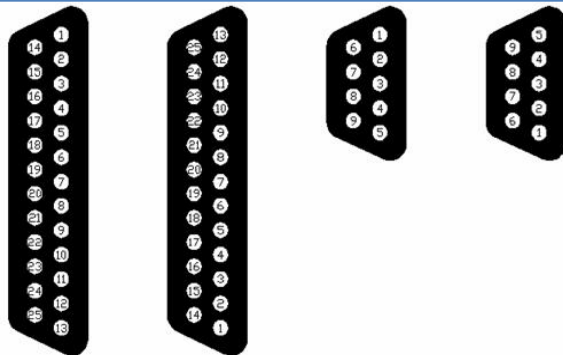
- RS232: Data transmitted as Voltage to terminal
  - 20KBps, 50Mters only
  - Improved to RS 422A (9 pine), RS 423A (15 pin-VGA)
- Modem (Data transmitted by Frequency)

# RS-232: Mostly used for Monitor

Pin	Signal	Function
2	TxD: transmitted Data	Output CPU to I/O
3	RxD :Received Data	Input I/O receive from CPU
4	RTS :Request to Send	Output from I/O
5	CTS :Clear to send	Input to I/O, HS signal
6	DSR: Data set ready	CPU send to I/O is ready
7	GND	Comm. Ref GND
8	DCD: Data Carrier Detect	I/O to disable reception
20	DTR: Data terminal ready	Output to indicate I/O is ready

DB-25

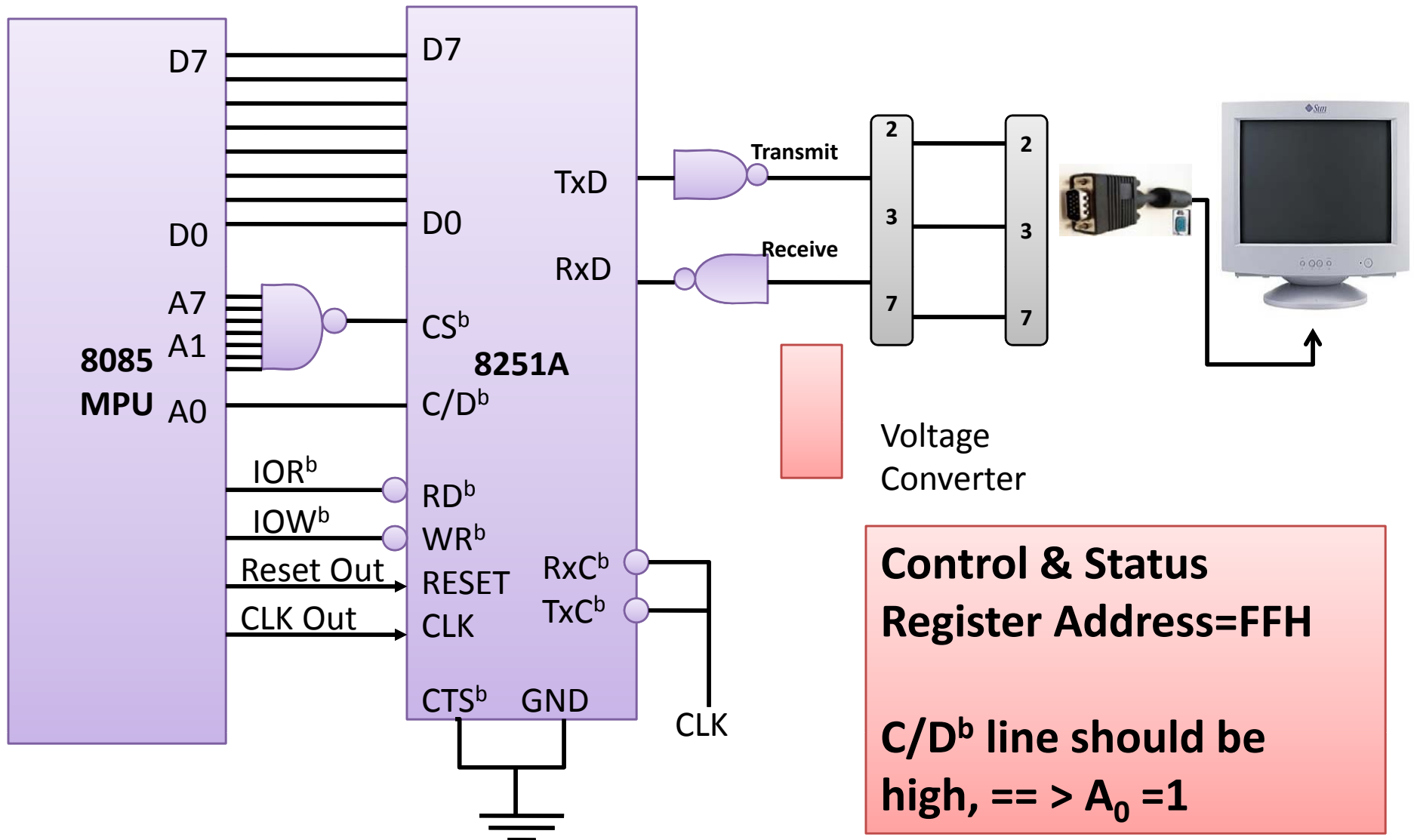
DB9



# Interfacing a CRT monitor using RS-232 terminal using 8251

- Connect a RS 232 port onto a CRT terminal
- Address the 8251A USART at FF to control transmission
- Specify initialization instructions and status word to transmit characters
  - Async mode with 9600 buad
  - Character length= 7 bit + 2 stop bit
  - No parity check
- Write instruction to initialize USART and read status word and Setup a loop until the transmitter is ready

# Interface Diagram



# Initialization of UART

Mode Word	D7	D6	D5	D4	D3	D2	D1	D0	CAH
	1	1	0	0	1	0	1	0	
	Two Stop bits		No parity		7 bit characters		Baud=Tx C/16 =153.6k/16 =9600		

COMMAND WORD	D7	D6	D5	D4	D3	D2	D1	D0	11H
	X	0	X	1	X	0	X	1	
				ERR Reset		Receive Disable		Transmit Enable	

STATUS	D7	D6	D5	D4	D3	D2	D1	D0	01H
	X	X	X	X	X	X	X	1	
								Transmit Ready	



# Initialization instructions

```
SETUP:      MVI    A,CAH ; load mode word
            OUT    FFH   ;Write mode word in control register
            MVI    A,11H ; load command word to enable TX
            OUT    FFH   ;Enable the transmitter
STATUS:     IN     FFH   ; Read the status register
            ANI    01H   ; Mask all bit except D0
            JZ     STATUS ; if D0=0 the TX buffer is full
```

# Display message stored at memory location 2070 to CRT

Message is "HELLO CS421"

2070 0E ; 13 characters to follow

2071 48; Letter H

2072 45; Letter E

2073 4C; Letter L

2074 4C ; Letter L

2075 4F ; Letter H

2076 20; space

2077 43; Letter C

2078 53; Letter S

2079 34; Digit 4

2080 32; Digit 2

2081 31; Digit 1

2082 0D; Carriage return

2083 0A; Linefeed

# Write a program to display message at CRT terminal

```
LXI      H 2070H ; Meory ptr for Message
MOV      C, M ; Set up Ctr register
MVI      A,40; Reset 8251
OUT      FFH
MVI      A,CA; Initialize 82512
OUT      FFH
MVI      A,11 ; initialize for transmit
OUT      FFH
```

```
STATUS:  IN      FFH
          ANI      01H      ;Ckeck TxRDY
          JZ       STATUS   ; is txRDY 1 ? If not wait
          INX      H        ; Pont to Next Char
          MOV      A,M      ; place the Char in ACC
          OUT      FEH      ; Send the Char to Transmitter
          DCR      C        ; DCr cnt
          JNZ      STATUS   ;Again Send the rest of Char
          HLT
```

**Thanks**