ELPL Embedded Low-Power Laboratory

Embedded System Application

4190.303C

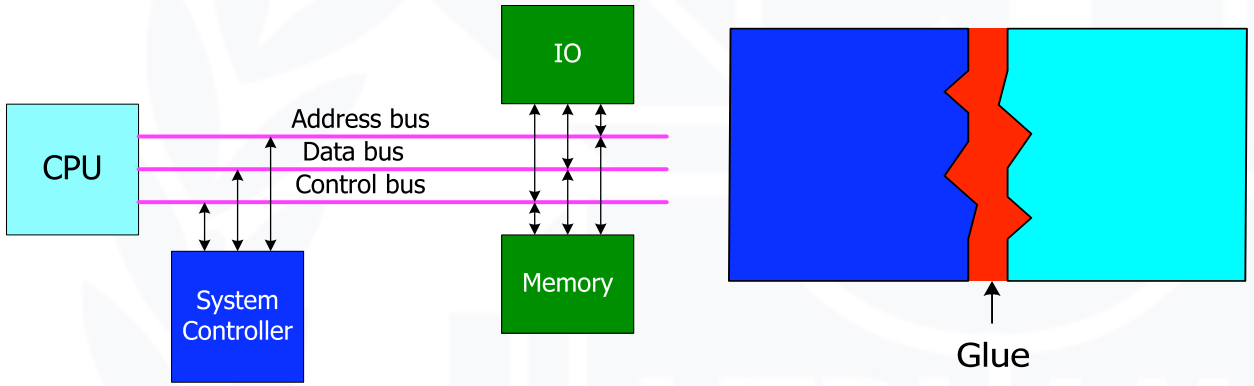
2010 Spring Semester

Serial Communication

Naehyuck Chang
Dept. of EECS/CSE
Seoul National University
naehyuck@snu.ac.kr

Introduction to System-on-Chip

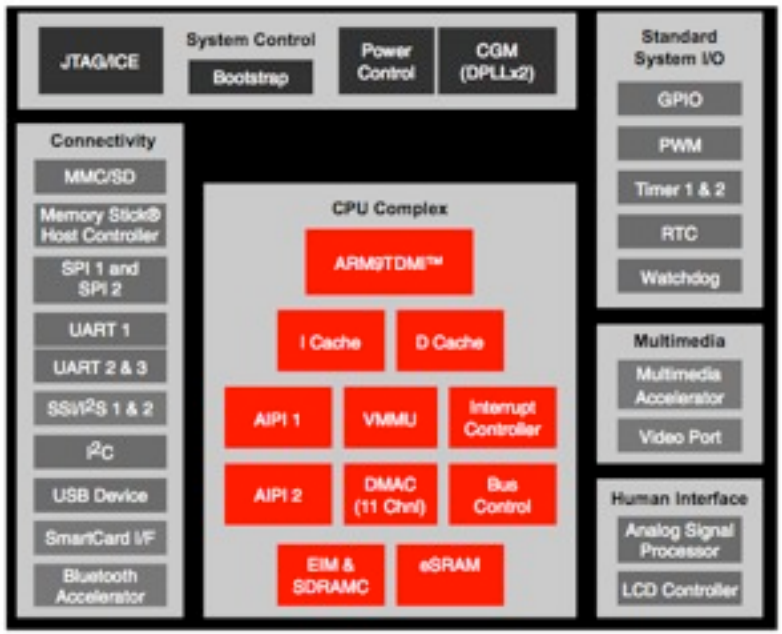
- i4040 - the first-generation microprocessor
  - Very low density even for a complete microprocessor core
  - Too expensive glue logic implementation when TTL is used
  - Companion chips



4008/4009	standard memory and I/O interface set
4040	4-bit Central Processor Unit with 60 instructions
4101	256 x 4 RAM
4201	clock generator
4265	programmable general purpose I/O device
4269	programmable keyboard display device
4289	standard memory interface
4308	1024-bit mask programmable ROM and four 4-bit I/O Ports
4316/2316	2048-bit ROM
4702/1702	2048-bit Erasable and Electrically Reprogrammable MOS ROM (Static)

Introduction to System-on-Chip

- Companion chips continued until 16-bit microprocessors became popular
  - Z80 CPU, SIO, PIO, etc.
- Microcontrollers
  - 16-bit or 32-bit microprocessor cores
  - Enough transistor density to accommodate other necessary peripherals



ARM920T™ Microprocessor Core

- AHB to IP Bus Interfaces (AIPIs)
- External Interface Module (EIM)
- SDRAM Controller (SDRAMC)
- DPLL Clock and Power Control Module
- Three Universal Asynchronous Receiver/Transmitters (UART 1, 2, and 3)
- Two Serial Peripheral Interfaces (SPI1 and SPI2)
- Two General-Purpose 32-bit Counters/Timers
- Watchdog Timer
- Real-Time Clock/Sampling Timer (RTC)
- LCD Controller (LCDC)
- Pulse-Width Modulation (PWM) Module
- Universal Serial Bus (USB) Device
- Multimedia Card and Secure Digital (MMC/SD) Host Controller Module
- Memory Stick® Host Controller (MSHC)
- Direct Memory Access Controller (DMAC)
- Two Synchronous Serial Interfaces and an Inter-IC Sound (SSI1 and SSI2/I2S) Module
- Inter-IC (I2C) Bus Module
- Video Port

Introduction to System-on-Chip

- System-on-chip (SoC) or system-on-a-chip
  - Integrating all components of a computer or other electronic system into a single chip
  - Microcontrollers usually have limited on-chip memory
  - Typical specifications
    - One microcontroller, microprocessor or DSP core(s)
    - Multiple-core systems are called MPSoC
    - Memory blocks including a selection of ROM, RAM, EEPROM and Flash
    - Timing sources including oscillators and phase-locked loops
    - Peripherals including counter-timers, real-time timers and power-on reset generators
    - External interfaces including industry standards such as USB, FireWire, Ethernet, USART, SPI, etc.
    - Analog interfaces including ADCs and DACs
    - Voltage regulators and power management circuits
  - Peripherals actually differentiate the value of an SoC

Introduction to System-on-Chip

- How to promote a microprocessor?
  - ARM-based microcontroller products
    - What makes the designer select a particular ARM-based microcontroller?

- **Analog Devices** (ARM7/ARM9/Cortex Family)
  ADuC7019, ADuC7020, ADuC7021, ADuC7022, ADuC7024, ADuC7025, ADuC7026, ADuC7027, ADuC7032, ADuC7033, ADuC7128, ADuC7129
- **ARM** (ARM7/ARM9/Cortex Family)
  Cortex-M1 (Altera)
- **Atmel** (ARM7/ARM9/Cortex Family)
  AT91C140, AT91F40416, AT91F40816, AT91FR40162, AT91FR4042, AT91FR4081, AT91M40400, AT91M40800, AT91M40807, AT91M42800A, AT91M43300, AT91M55800A, AT91M63200, AT91R40008, AT91R40807, AT91RM3400, AT91RM9200, AT91SAM7A1, AT91SAM7A2, AT91SAM7A3, AT91SAM7S128, AT91SAM7S256, AT91SAM7S32, **New!** AT91SAM7S321, AT91SAM7S512, AT91SAM7S64, AT91SAM7SE256, AT91SAM7SE32, AT91SAM7SE512, AT91SAM7X128, AT91SAM7X256, AT91SAM7XC128, AT91SAM7XC256, AT91SAM9260, AT91SAM9261, AT91SAM9263, AT91SAM9R64, AT91SAM9RL64
- **AustriaMicroSystems** (ARM7/ARM9/Cortex Family)
  AS3525
- **Cirrus Logic** (ARM7/ARM9/Cortex Family)
  CS740110-IQZ, CS740120-IQZ, CS740130-IQZ
- **Freescale Semiconductor** (ARM7/ARM9/Cortex Family)
  MAC7101, MAC7104, MAC7105, MAC7106, MAC7111, MAC7112, MAC7114, MAC7115, MAC7116, MAC7121, MAC7122, MAC7124, MAC7125, MAC7126, MAC7131, MAC7134, MAC7135, MAC7136, MAC7141, MAC7142, MAC7144, **New!** MC9328MX21, **New!** MCIMX27, MCIMX27L
- **Luminary Micro** (ARM7/ARM9/Cortex Family)
  LM3S101, LM3S102, LM3S1110, LM3S1133, LM3S1138, LM3S1150, LM3S1162, LM3S1165, LM3S1332, LM3S1435, LM3S1439, LM3S1512, LM3S1538, LM3S1620, LM3S1635, LM3S1637, LM3S1751, LM3S1850, LM3S1937, LM3S1958, LM3S1960, LM3S1968, LM3S2110, LM3S2139, LM3S2410, LM3S2412, LM3S2432, LM3S2533, LM3S2620, LM3S2637, LM3S2651, LM3S2730, LM3S2739, LM3S2939, LM3S2948, LM3S2950, LM3S2965, LM3S301, LM3S310, LM3S315, LM3S316, LM3S317, LM3S328, LM3S601, LM3S610, LM3S6100, LM3S611, LM3S6110, LM3S612, LM3S613, LM3S615, LM3S617, LM3S618, LM3S628, LM3S6420, LM3S6422, LM3S6432, LM3S6537, LM3S6610, LM3S6633, LM3S6637, LM3S6730, LM3S6753, LM3S6938, LM3S6950, LM3S6952, LM3S6965, LM3S801, LM3S811, LM3S812, LM3S815, LM3S817, LM3S818, LM3S828, LM3S8530, LM3S8538, LM3S8630, LM3S8730, LM3S8733, LM3S8738, LM3S8930, LM3S8933, LM3S8938, LM3S8962, LM3S8970, LM3S8971
- **NXP (founded by Philips)** (ARM7/ARM9/Cortex Family)
  LH75400, LH75401, LH75410, LH75411, LH79520, LH79524, LH79525, LH7A400, LH7A404, LPC2101, LPC2102, LPC2103, LPC2104, LPC2105, LPC2106, LPC2109, LPC2109/01, LPC2114, LPC2119, LPC2124, LPC2129, LPC2131, LPC2131/01, LPC2132, LPC2132/01, LPC2134, LPC2134/01, LPC2136, LPC2136/01, LPC2138, LPC2138/01, LPC2141, LPC2142, LPC2144, LPC2146, LPC2148, LPC2194, LPC2210, LPC2212, LPC2214, LPC2220, LPC2290, LPC2292, LPC2294, LPC2364, LPC2366, LPC2368, LPC2378, LPC2387, LPC2388, LPC2460, LPC2468, LPC2470, LPC2478, LPC2880, LPC2888, LPC2917, LPC2919, LPC3180, **New!** LPC3220, **New!** LPC3230, **New!** LPC3240, **New!** LPC3250
- **OKI** (ARM7/ARM9/Cortex Family)
  ML674000, ML674001, ML674002, ML674003, ML675001, ML675002, ML675003, ML67Q4050, ML67Q4051, ML67Q4060, ML67Q4061, ML696201, ML69Q6203
- **Samsung** (ARM7/ARM9/Cortex Family)
  S3C2410A, S3C2440A, S3C44B0X, S3C4510B, S3F4A0KR, S3F4A1HR, S3F4A2FR
- **Sharp** (ARM7/ARM9/Cortex Family)
  see NXP
- **STMicroelectronics** (ARM7/ARM9/Cortex Family)
  STA2051, **New!** STM32F101C6, **New!** STM32F101C8, STM32F101CB, STM32F101R6, STM32F101R8, STM32F101RB, **New!** STM32F101RC, **New!** STM32F101RD, **New!** STM32F101RE, STM32F101T6, STM32F101T8, STM32F101V8, STM32F101VB, **New!** STM32F101VC, **New!** STM32F101VD, **New!** STM32F101VE, **New!** STM32F101ZC, **New!** STM32F101ZD, **New!** STM32F101ZE, STM32F103C6, STM32F103C8, STM32F103CB, STM32F103R6, STM32F103R8, STM32F103RB, **New!** STM32F103RC, **New!** STM32F103RD, **New!** STM32F103RE, STM32F103T6, STM32F103T8, **New!** STM32F103V8, STM32F103VB, **New!** STM32F103VC, **New!** STM32F103VD, **New!** STM32F103VE, **New!** STM32F103ZC, **New!** STM32F103ZD, **New!** STM32F103ZE, STR710FZ1, STR710FZ2, STR711FR0, STR711FR1, STR711FR2, STR712FR0, STR712FR1, STR712FR2, STR715FR0, STR730FZ1, STR730FZ2, STR731FV0, STR731FV1, STR731FV2, STR735FZ1, STR735FZ2, STR736FV0, STR736FV1, STR750FL2, STR750FV0, STR750FV1, STR750FV2, STR751FR0, STR751FR1, STR751FR2, STR752FR0, STR752FR1, STR752FR2, STR755FR0, STR755FR1, STR755FR2, STR755FV0, STR755FV1, STR755FV2, STR910FAM32, STR910FAW32, STR910FAZ32, STR910FM32, STR910FW32, STR911FAM42, STR911FAM44, STR911FAM46, STR911FAM47, STR911FAW42, STR911FAW44, STR911FAW46, STR911FAW47, STR911FM42, STR911FM44, STR912FAW42, STR912FAW44, STR912FAW46, STR912FAW47, STR912FAZ42, STR912FAZ44, STR912FAZ46, STR912FAZ47, STR912FW42, STR912FW44
- **Texas Instruments** (ARM7/ARM9/Cortex Family)
  TMS470R1A128, TMS470R1A256, TMS470R1A288, TMS470R1A384, TMS470R1A64, TMS470R1B1M, TMS470R1B512, TMS470R1B768
- **Winbond** (ARM7/ARM9/Cortex Family)
  W90N740, W90N745, W90P710
- **Zilog** (ARM7/ARM9/Cortex Family)
  ZA9L

Introduction to System-on-Chip

- Device selection points
  - Not only the CPU core is important!
  - Key to success for Core-A

9 Reasons to Love STR910FA !

1. Ethernet!
2. Very Large SRAM and Dual-Bank Flash Memories
3. 96 MIPS peak performance from ARM9E core
4. Efficient DMA and Rapid Data Flow
5. 1.2uA Max Real-Time Clock with Tamper Detection
6. Connectivity, Excellent Selection
7. Minimal External Components Required
8. Power Management
9. Extensive Tools and Firmware Library Support

http://www.st.com/mcu/files/mcu/1208848572.pdf

- Flash (Bytes)
- SRAM (Bytes)
- Cache Memory (Bytes)
- LCD Controller
- Image Sensor Interface
- External Bus Interface
- SDRAM Interface
- MMU/MPU
- AES Engine (Bits)
- Triple DES Engine (Bits)
- USB Host Controller
- USB Device Controller
- Enhanced USART
- USART/DBGU
- SPI
- TWI
- SSC
- MCI

- CAN
- Ethernet MAC
- RTC/RTT
- ADC Channels
- DAC Channels
- Peripheral DMA Channels
- Max. Clock Speed (MHz)
- PWM Controller
- I/O Pins number
- I/O Voltage Domain (V)
- High Current Pads
- 16-bit Timers
- Power-On-Reset
- Brown Out Detection
- On-chip RC Oscillator
- Crystal Oscillator/PLL
- Period Interval Timer
- Watchdog Timer

Introduction to System-on-Chip

- There is a big gap between the component development and system development
  - Developers often consider solution support more importantly than the performance of the microprocessor core
  - System design solution is a bridge across a death valley

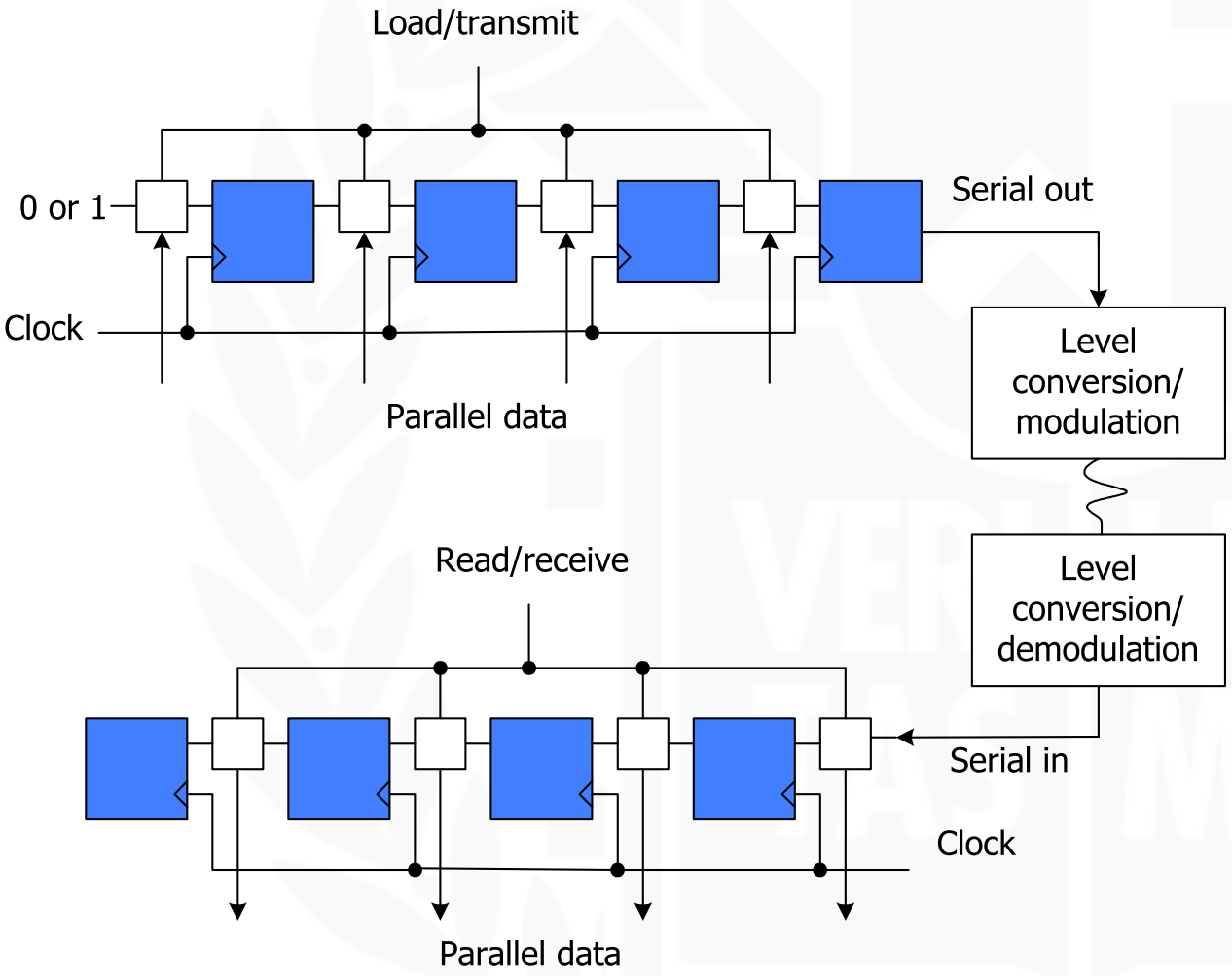
Serial Communication

- Communication
  - Data transmission between two different domains
  - On-chip-, on-board, in system, or inter system communications
- Parallel communication
  - Transmit and receive a byte or word data simultaneously with parallel links such as a printer port
- Serial communication
  - Parallel data → serialize → transmit → parallelize → parallel data
  - Better for long distance communication
    - Low wire cost
    - Low signal conditioning cost
  - Speed penalty for serial transmission → overcome by high bit rate

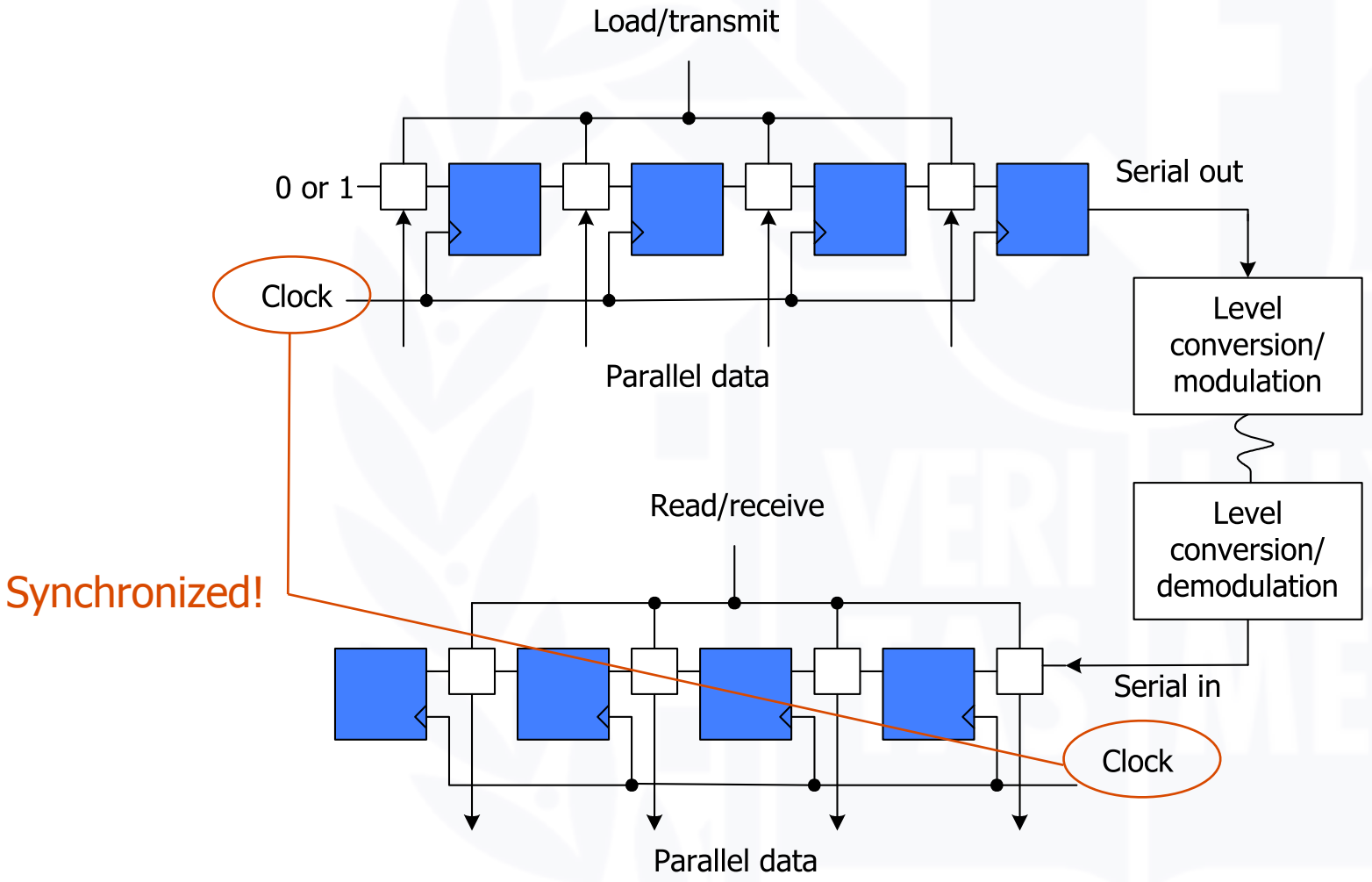
Serial Communication

- Transmit bits in a single channel
  - Simplex (one way)
  - Half-duplex (one direction at a time)
  - Full-duplex (two way)
- A sequence of bits – packet or character
  - ASCII code – 7 bits for 128 characters (alphabet, numerical, and control)
    - Start, stop, and parity bits
  - Binary code - not necessarily be 8 bit multiple
    - Fixed length or variable length

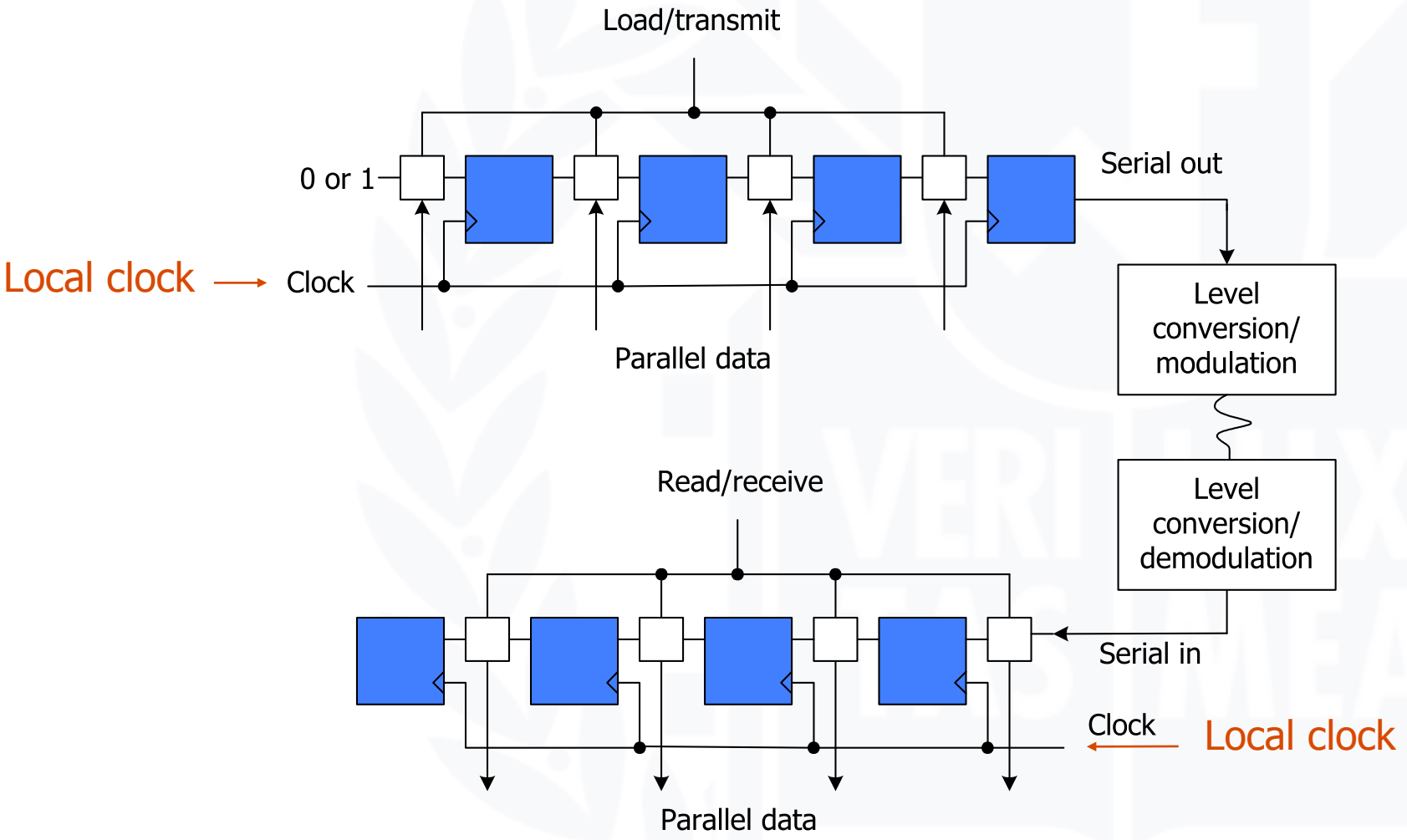
Principle of Operation



Synchronous Communication

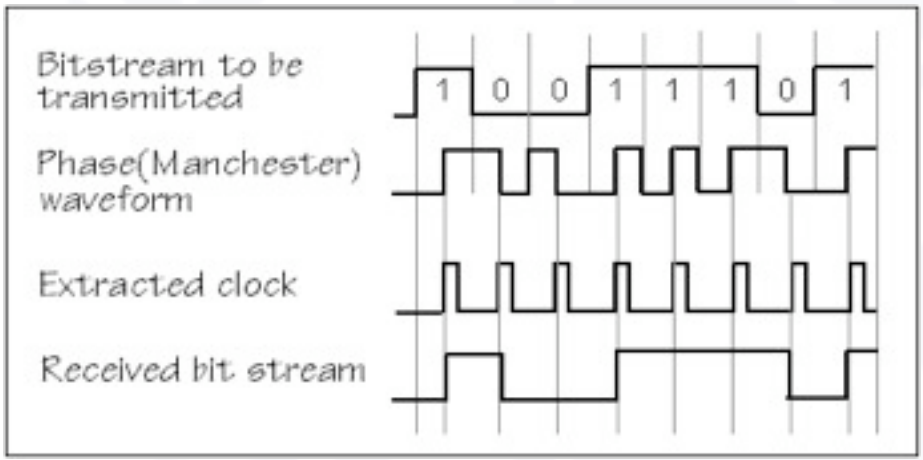


Synchronous Communication



Synchronous Communication

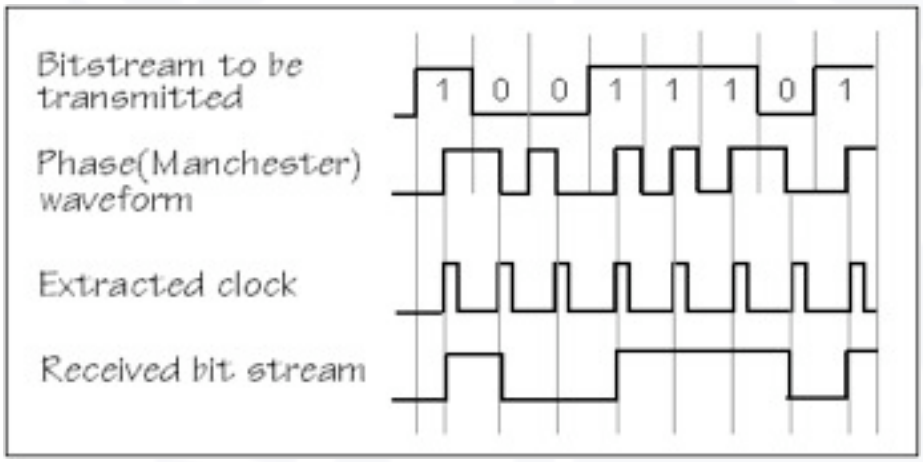
- Synchronous communication
  - Use of the same clock is not feasible
  - Direct transmission of the clock signal is not practical
  - Send clock information with data
  - Complex encoding but high bit rate
  - DC balance



Synchronous Communication

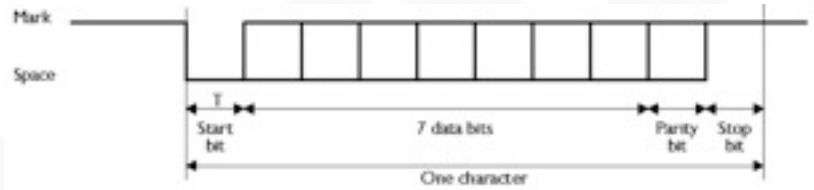
- Synchronous communication
  - Use of the same clock is not feasible
  - Direct transmission of the clock signal is not practical
  - Send clock information with data
  - Complex encoding but high bit rate
  - DC balance

Long packet is feasible



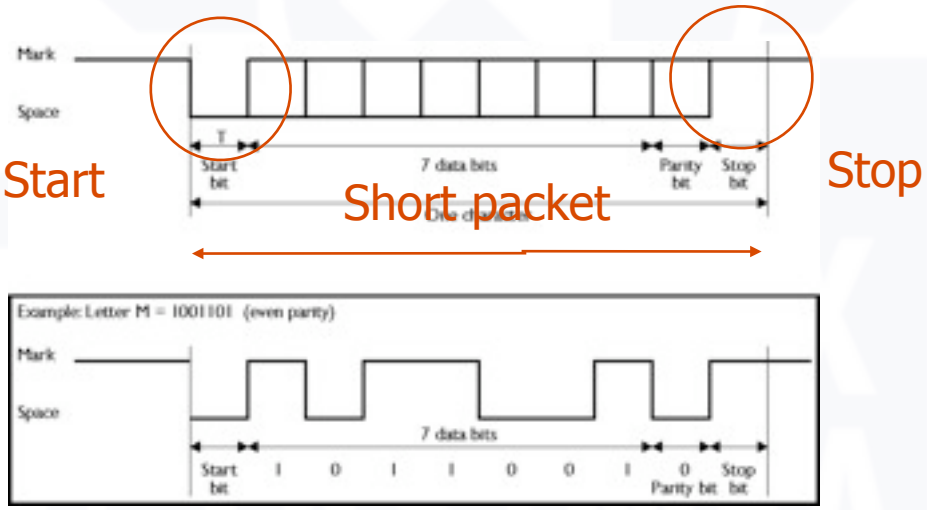
Asynchronous Serial Communication - UART

- Asynchronous communication
  - Use of lock clocks for both ends
  - Local clocks are not synchronized → eventually skewed
    - Oversampling and re-synchronization with start and stop bit information
  - Simple encoding but slow bit rate → start and stop for every 8 bits
- Universal asynchronous receiver/transmitter
  - General purpose
- Baud rate
  - Generally 57600 bps, 115200 bps
  - Up to ~10 Mbps



Asynchronous Serial Communication - UART

- Asynchronous communication
  - Use of lock clocks for both ends
  - Local clocks are not synchronized → eventually skewed
    - Oversampling and re-synchronization with start and stop bit information
  - Simple encoding but slow bit rate → start and stop for every 8 bits
- Universal asynchronous receiver/transmitter
  - General purpose
- Baud rate
  - Generally 57600 bps, 115200 bps
  - Up to ~10 Mbps



General UART Definitions

- Bit time
  - The period of time required to serially transmit or receive 1 bit of data (1 cycle of the baud rate frequency)
- Start bit
  - The bit time of a logic 0 that indicates the beginning of a data frame
  - A start bit begins with a 1-to-0 transition, and is preceded by at least 1 bit time of logic 1
- Stop bit
  - 1 bit time of logic 1 that indicates the end of a data frame
- BREAK
  - A frame in which all of the data bits, including the stop bit, is logic 0
  - This type of frame is usually sent to signal the end of a message or the beginning of a new message

General UART Definitions

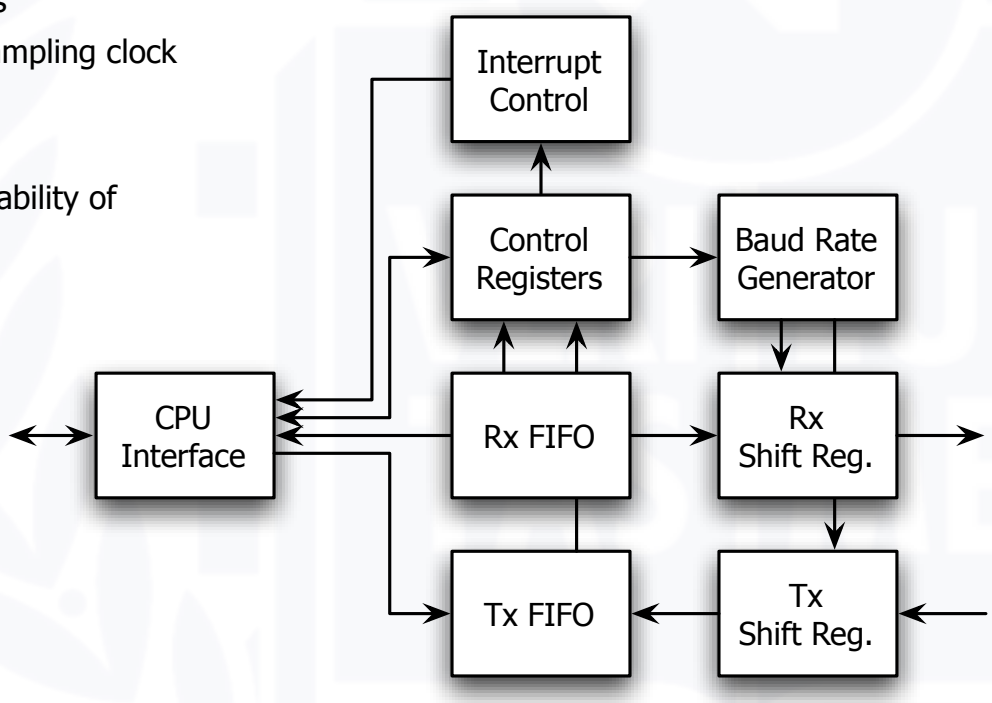
- Frame
  - A start bit followed by a specified number of data or information bits and terminated by a stop bit
  - The most common frame format is 1 start bit followed by 8 data bits (least significant bit first) and terminated by 1 stop bit
  - An additional stop bit and a parity bit also can be included
- Framing error
  - An error condition that occurs when the stop bit of a received frame is missing
  - Usually when the frame boundaries in the received bit stream are not synchronized with the receiver bit counter
  - Framing errors can go undetected if a data bit in the expected stop bit time happens to be a logic 1
  - A framing error is always present on the receiver side when the transmitter is sending BREAKs
  - However, when the UART is programmed to expect 2 stop bits and only the first stop bit is received, this is not a framing error by definition

General UART Definitions

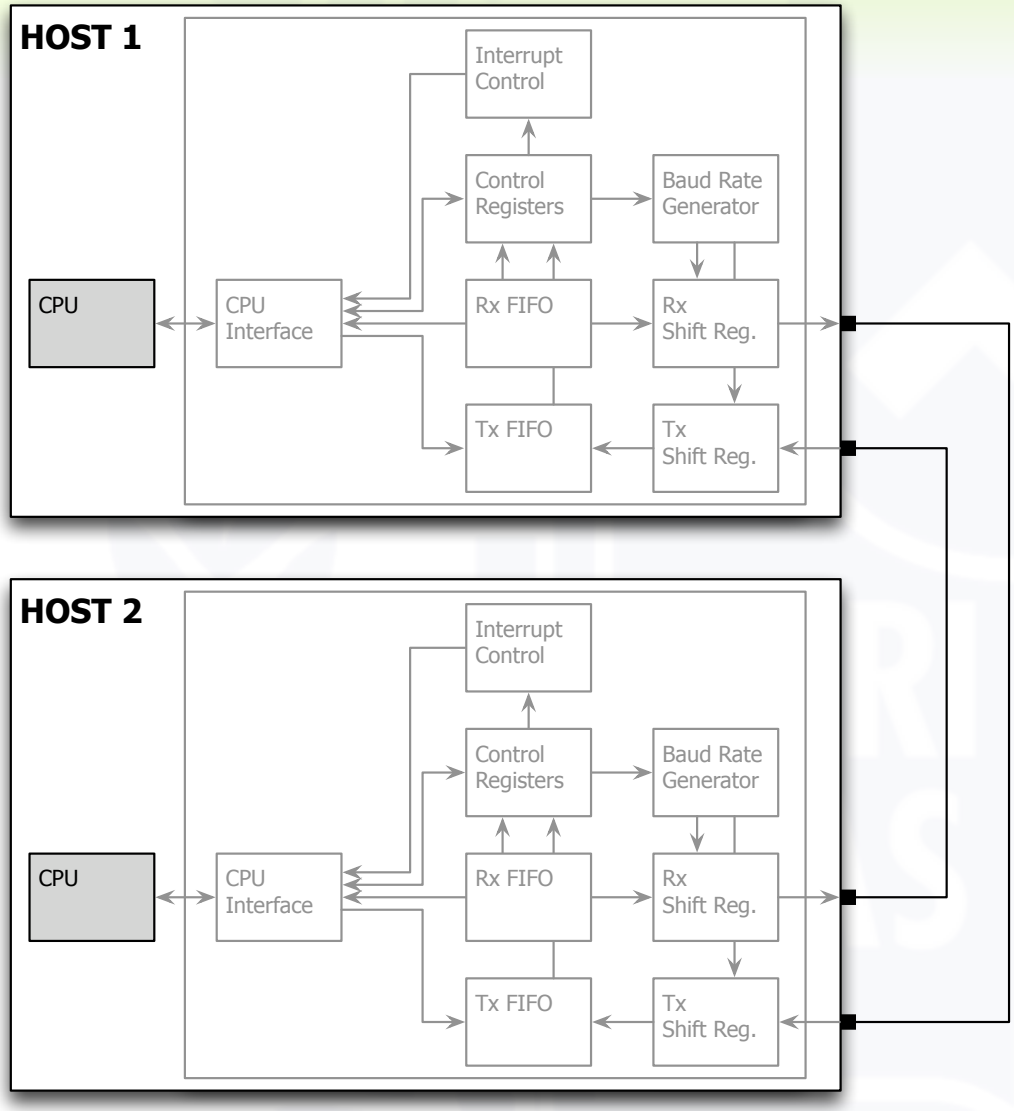
- Parity error
  - An error condition that occurs when the calculated parity of the received data bits in a frame does not match the parity bit received on the RXD input
  - Parity error is calculated only after an entire frame is received
- Overrun error
  - An error condition that occurs when the latest character received is ignored to prevent overwriting a character already present in the UART receive buffer (RxFIFO)
  - An overrun error indicates that the software reading the buffer (RxFIFO) is not keeping up with the actual reception of characters on the RXD input

UART Architecture

- Functional block diagram
  - Serializer
  - CPU interface
    - From the perspective of CPU, the whole module is a memory which can be read (Rx) and written (Tx) with control registers
    - Each registers are assigned with a fixed address in a on-chip bus address space
  - Baud rate generator
    - Provides clocks for Rx/Tx modules
    - Generates a high frequency up-sampling clock for the Rx module
  - FIFOs and interrupt control
    - CPU cannot keep polling the availability of shift registers

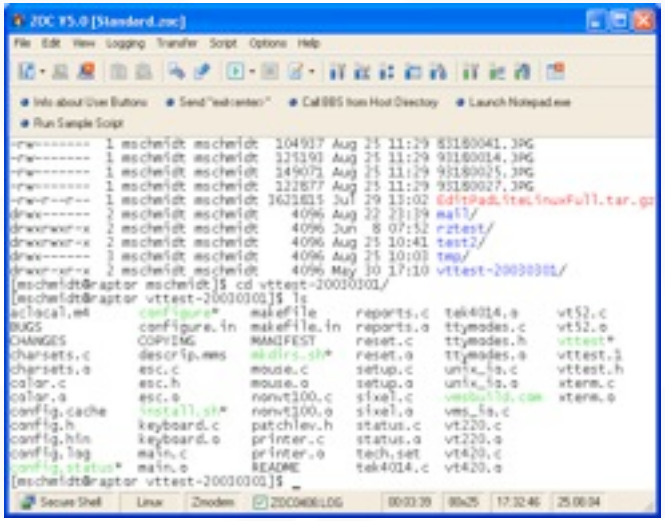


UART Architecture



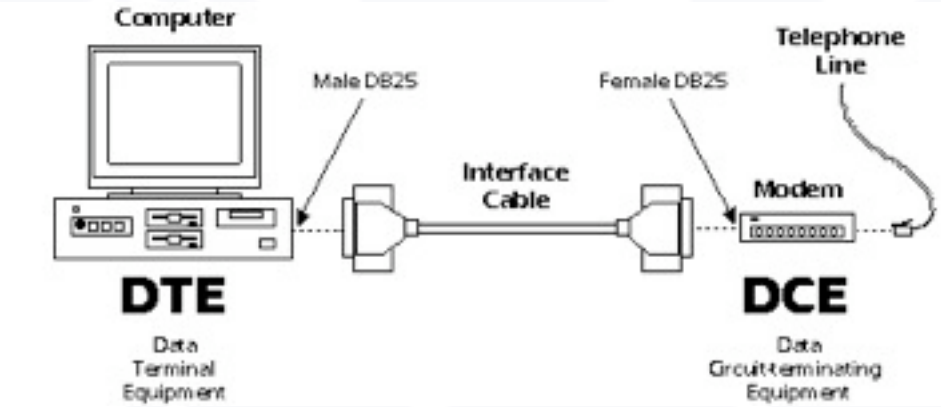
Terminal and Terminal Emulator

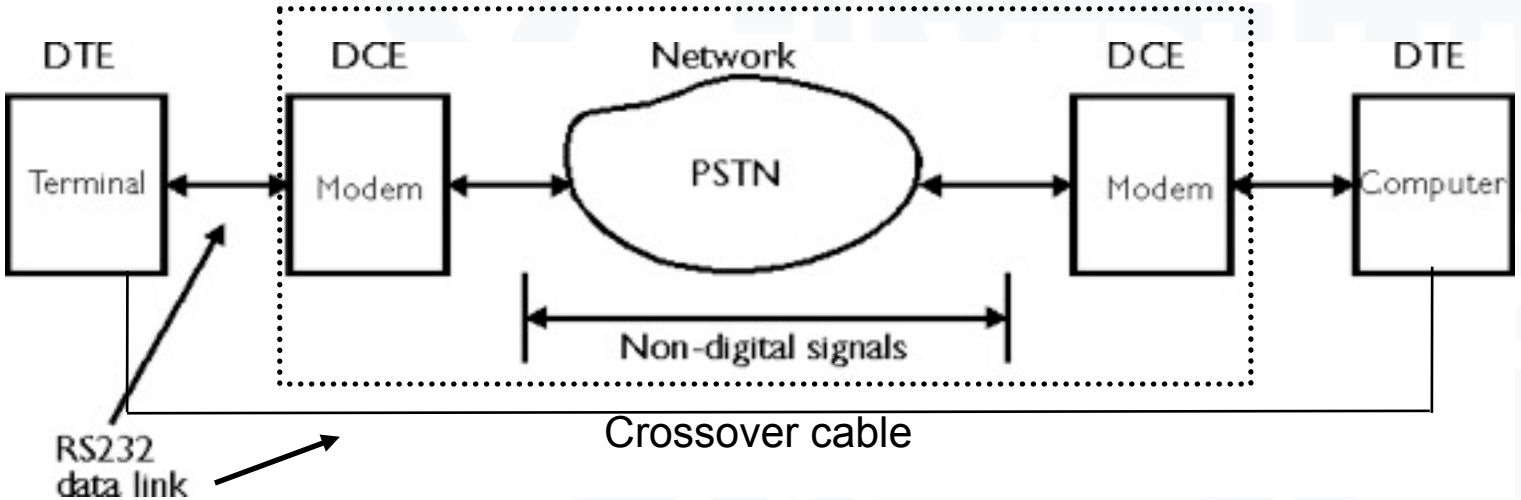
- Dummy terminal
- Terminal emulator



DTE and DCE

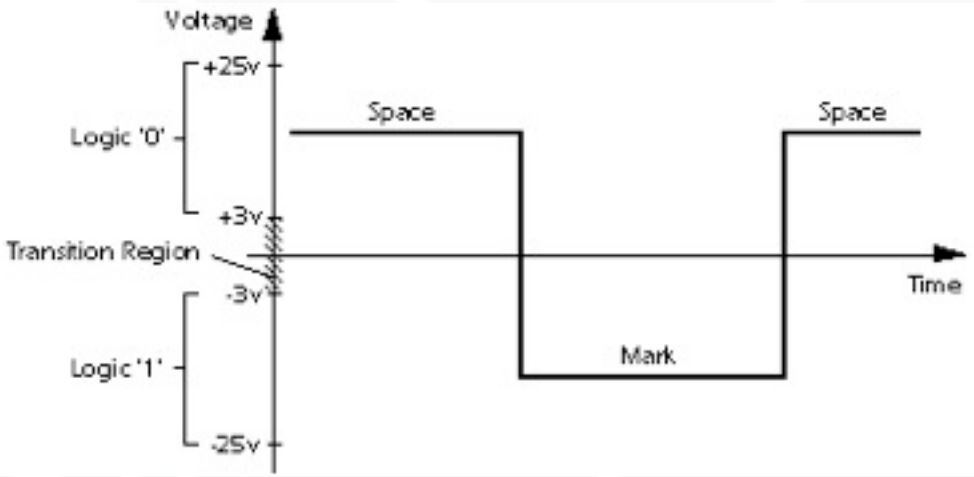
- Data terminal equipment
- Data circuit-terminating equipment





EIA RS232

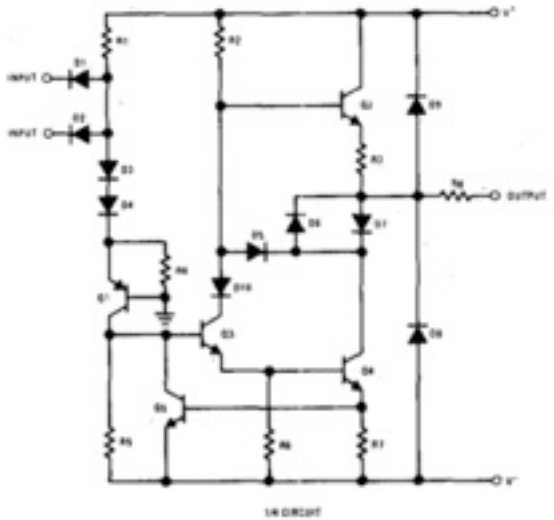
- International standard for serial communication
- Connection and signal characteristics
- Data terminal equipment and data communication equipment
- Logic 1 (marking)
  - -3 V to -25 V with respect to signal ground
- Logic 0 (spacing)
  - +3 V to +25 V
- Not assigned
  - between -3 V and +3 V
    (a transition region)



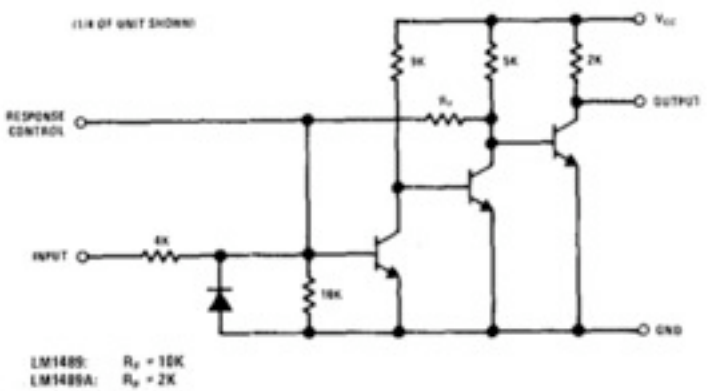
TTL – RS232C Level Conversion

- Most UARTs' signals are TTL, LVTTL, CMOS, LVCMOS, etc.
- Level converters are mandatory
- Old-fashioned TTL to RS232 level converter
  - LM1488 and LM1489

schematic and connection diagrams

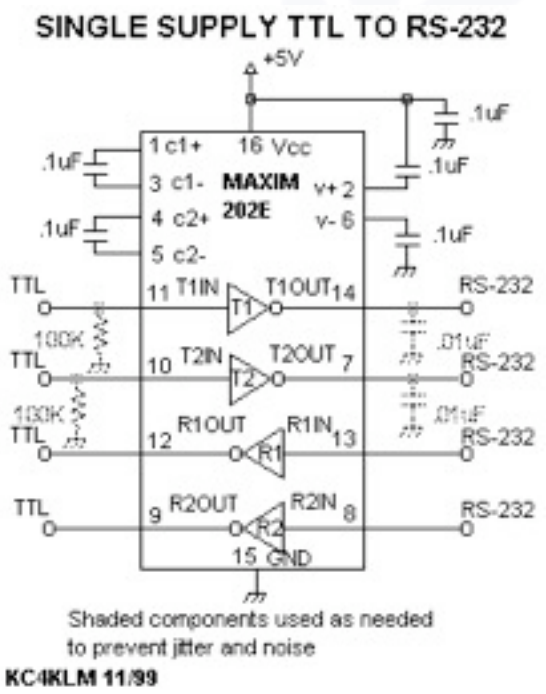


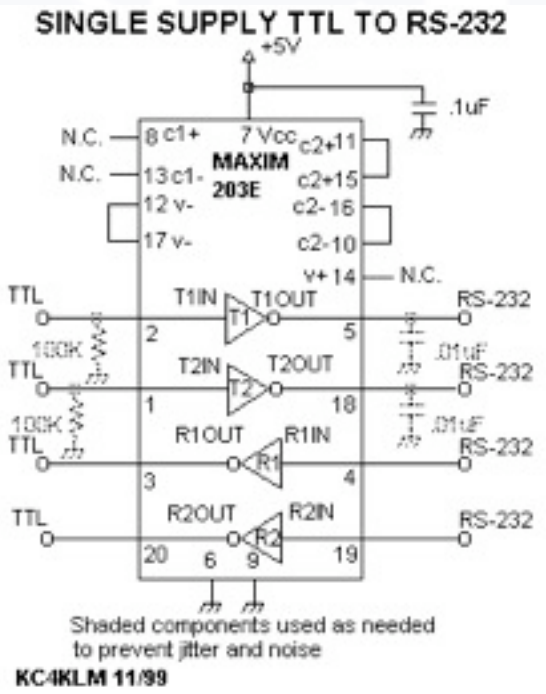
Supply Voltage	
V^+	+15V
V^-	−15V
Input Voltage (V_{IN})	$-15V \leq V_{IN} \leq 7.0V$
Output Voltage	±15V
Power Derating (Note 2)	
(Package Limitation, J Package)	1000 mW
Derating above T_A = +25°C ($1/\theta_{JA}$)	6.7 mW/°C
Operating Temperature Range	0°C to +75°C
Storage Temperature Range	−65°C to +175°C
Lead Temperature (Soldering, 10 sec)	300°C



TTL – RS232C Level Conversion

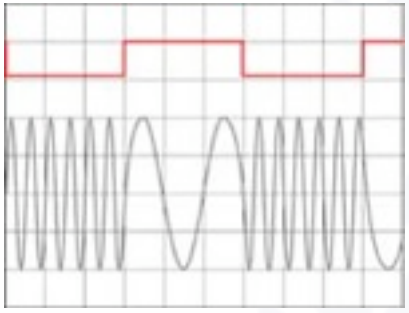
- Modern TTL to RS232 level converters
  - Built-in DC-DC converters

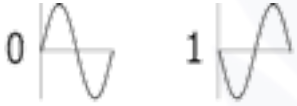


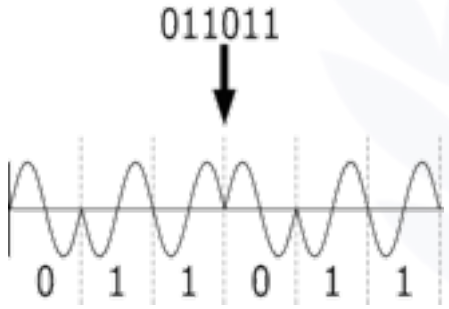


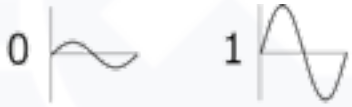
Modem (Modulation and Demodulation)

- Square wave unnecessarily has high-frequency components for communication

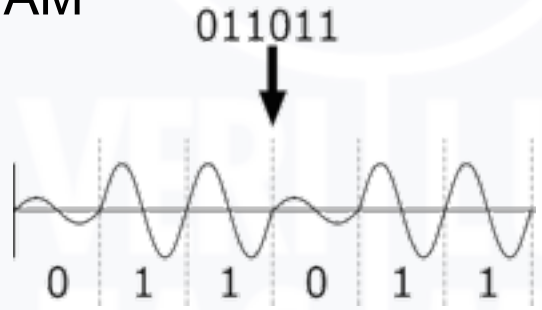








PAM



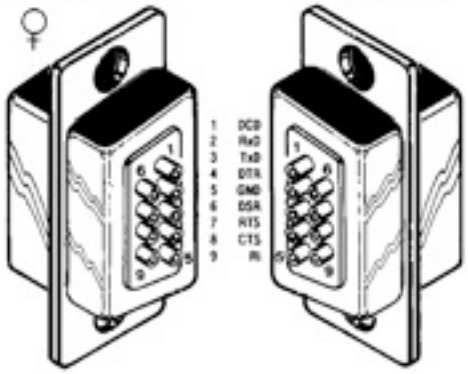
Modem Control Signals

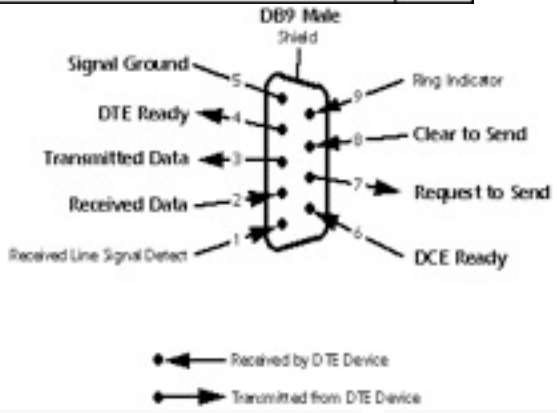
- Flow control (handshaking) signals to avoid buffer overflow or lock-up

Modem Cable - Straight Cable DB9 to DB9

DTE Device (Computer) **DB9** — **DTE to DCE** Connections — DCE Device (Modem) **DB9**

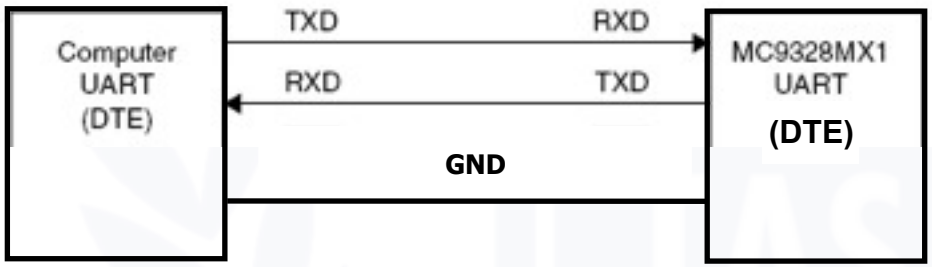
Pin# DB9 RS-232 Signal Names		Signal Direction	Pin# DB9 RS-232 Signal Names	
#1 Carrier Detector (DCD)	CD	←	#1 Carrier Detector (DCD)	CD
#2 Receive Data (Rx)	RD	←	#2 Receive Data (Rx)	RD
#3 Transmit Data (Tx)	TD	→	#3 Transmit Data (Tx)	TD
#4 Data Terminal Ready	DTR	→	#4 Data Terminal Ready	DTR
#5 Signal Ground/Common (SG)	GND	—	#5 Signal Ground/Common (SG)	GND
#6 Data Set Ready	DSR	←	#6 Data Set Ready	DSR
#7 Request to Send	RTS	→	#7 Request to Send	RTS
#8 Clear to Send	CTS	←	#8 Clear to Send	CTS
#9 Ring Indicator	RI	←	#9 Ring Indicator	RI
Soldered to DB9 Metal - Shield	FGND	—	Soldered to DB9 Metal - Shield	FGND





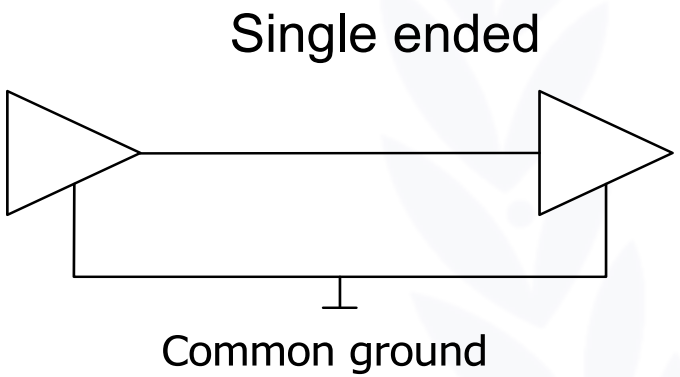
Simplified Asynchronous Serial Communication

- Three-wire connection
  - Null-modem or crossover cable
    - Connect two DTEs whose connectors are the same
    - Connects the TxD (transmit data) on one end with the RxD (receive data) on the other end



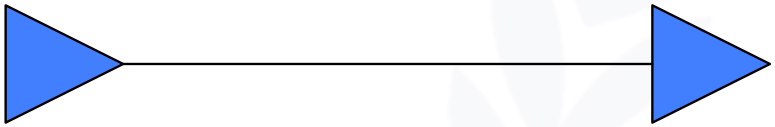
Single-ended and Differential Data Transmission

- Single-ended methods are often inadequate
  - High data rates
  - Long distances in real world environments
  - Different ground level
- Differential data transmission (balanced differential signal) offers superior performance in most applications

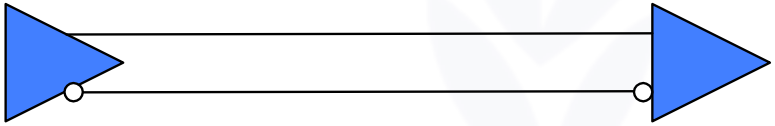


Single-ended and Differential Data Transmission

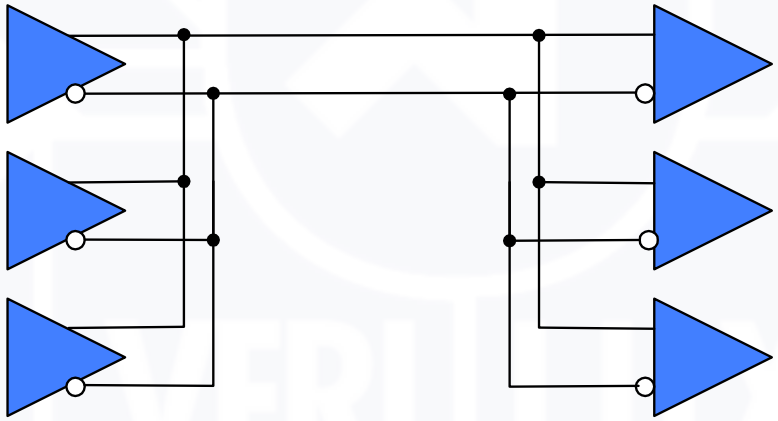
- RS232, RS422 and RS485



Point-to-point single ended: RS232



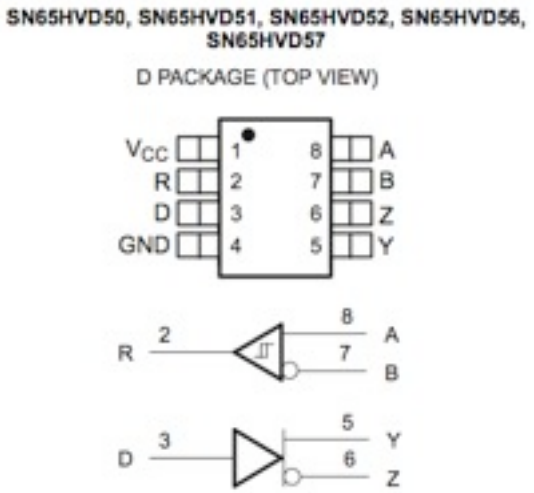
Point-to-point differential: RS422

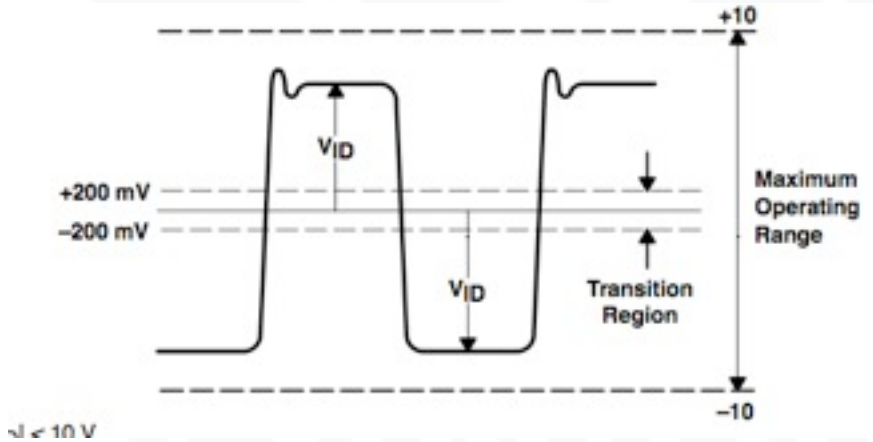


Multi-drop differential: RS485

Single-ended and Differential Data Transmission

- Modern high-speed duplex TTL to RS485 driver
  - Common-mode rejection ratio (CMRR)





$V_{(A)}$, $V_{(B)}$, $V_{(Y)}$, $V_{(Z)}$	Voltage range at any bus terminal (A, B, Y, Z)	–9 V to 14 V
$V_{(TRANS)}$	Voltage input, transient pulse through 100 Ω. See Figure 12 (A, B, Y, Z)(3)	–50 to 50 V

Single-ended and Differential Data Transmission

- Serial communication physical layer specifications

SPECIFICATIONS		RS232	RS423	RS422	RS485
Mode of Operation		SINGLE -ENDED	SINGLE -ENDED	DIFFERENTIAL	DIFFERENTIAL
Total Number of Drivers and Receivers on One Line (One driver active at a time for RS485 networks)		1 DRIVER 1 RECVR	1 DRIVER 10 RECVR	1 DRIVER 10 RECVR	32 DRIVER 32 RECVR
Maximum Cable Length		50 FT.	4000 FT.	4000 FT.	4000 FT.
Maximum Data Rate (40ft. - 4000ft. for RS422/RS485)		20kb/s	100kb/s	10Mb/s-100Kb/s	10Mb/s-100Kb/s
Maximum Driver Output Voltage		+/-25V	+/-6V	-0.25V to +6V	-7V to +12V
Driver Output Signal Level (Loaded Min.)	Loaded	+/-5V to +/-15V	+/-3.6V	+/-2.0V	+/-1.5V
Driver Output Signal Level (Unloaded Max)	Unloaded	+/-25V	+/-6V	+/-6V	+/-6V
Driver Load Impedance (Ohms)		3k to 7k	>=450	100	54
Max. Driver Current in High Z State	Power On	N/A	N/A	N/A	+/-100uA
Max. Driver Current in High Z State	Power Off	+/-6mA @ +/-2v	+/-100uA	+/-100uA	+/-100uA
Slew Rate (Max.)		30V/uS	Adjustable	N/A	N/A
Receiver Input Voltage Range		+/-15V	+/-12V	-10V to +10V	-7V to +12V
Receiver Input Sensitivity		+/-3V	+/-200mV	+/-200mV	+/-200mV
Receiver Input Resistance (Ohms), (1 Standard Load for RS485)		3k to 7k	4k min.	4k min.	>=12k

Application

- Modem
- Dummy terminal for development
- Converter to other communication protocols (LAN, Bluetooth, etc.)
- Simple sensors
  - ex) Issys methanol sensor

  - ex) Low resolution camera

larger image

Uart Camera Module with Jpeg compression - C328

$49.90

General Description

The C328 module is a highly integrated serial camera board that can be attached to a wireless or PDA host performing as a video camera or a JPEG

ELPL Embedded Low-Power Laboratory

Embedded System Application

4190.303C

2010 Spring Semester

I^2C Bus

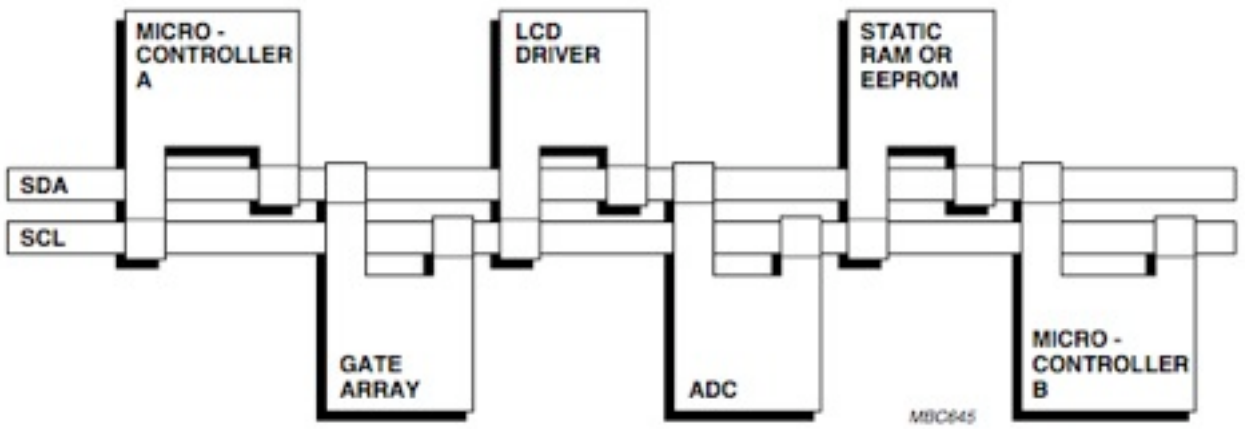
Naehyuck Chang
Dept. of EECS/CSE
Seoul National University
naehyuck@snu.ac.kr

Basic Characteristics

- Two-wired communication
  - SDA and SCL
  - I2C is for **on-board communication** so the ground is shared
- Multi-master
  - Multiple masters allowed with collision detection and arbitration
  - Operate as either a transmitter or receiver, depending on the function of the device
  - Devices can also be considered as masters or slaves when performing data transfers
- Speeds
  - 100 kbps (standard mode)
  - 400 kbps (fast mode)
  - 3.4 Mbps (high-speed mode)
- 8-bit oriented data transfer
- Addressing
  - 7-bit or 10-bit unique addresses

Basic Characteristics

- Device count limit
  - No logical limit
  - Maximum capacitance of 400 pF



Example of an I2C-bus configuration using two microcontrollers

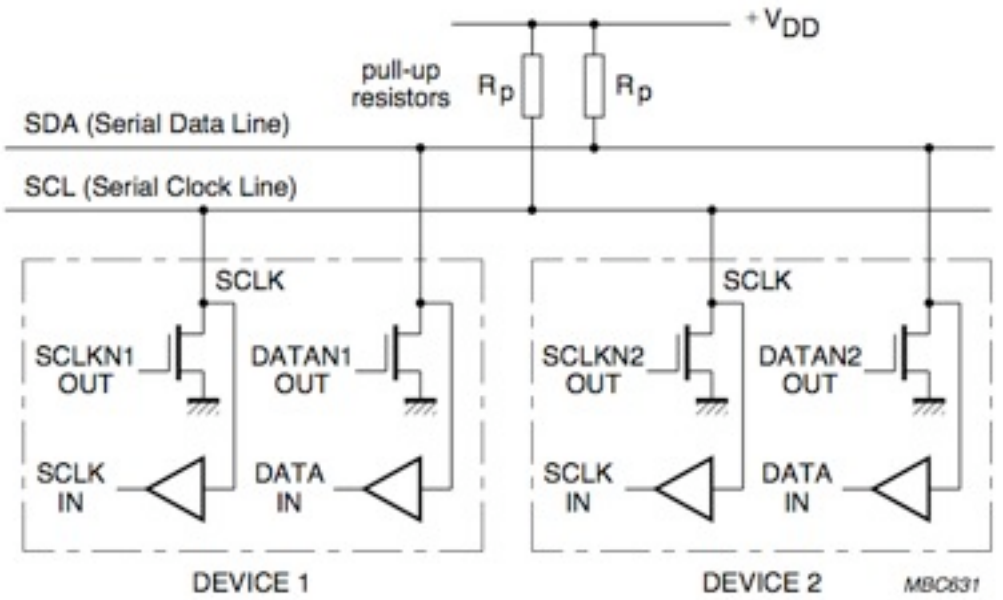
Basic Terminology

- Summary of basic terminology

TERM	DESCRIPTION
Transmitter	The device which sends data to the bus
Receiver	The device which receives data from the bus
Master	The device which initiates a transfer, generates clock signals and terminates a transfer
Slave	The device addressed by a master
Multi-master	More than one master can attempt to control the bus at the same time without corrupting the message
Arbitration	Procedure to ensure that, if more than one master simultaneously tries to control the bus, only one is allowed to do so and the winning message is not corrupted
Synchronization	Procedure to synchronize the clock signals of two or more devices

I2C Electrical Specification

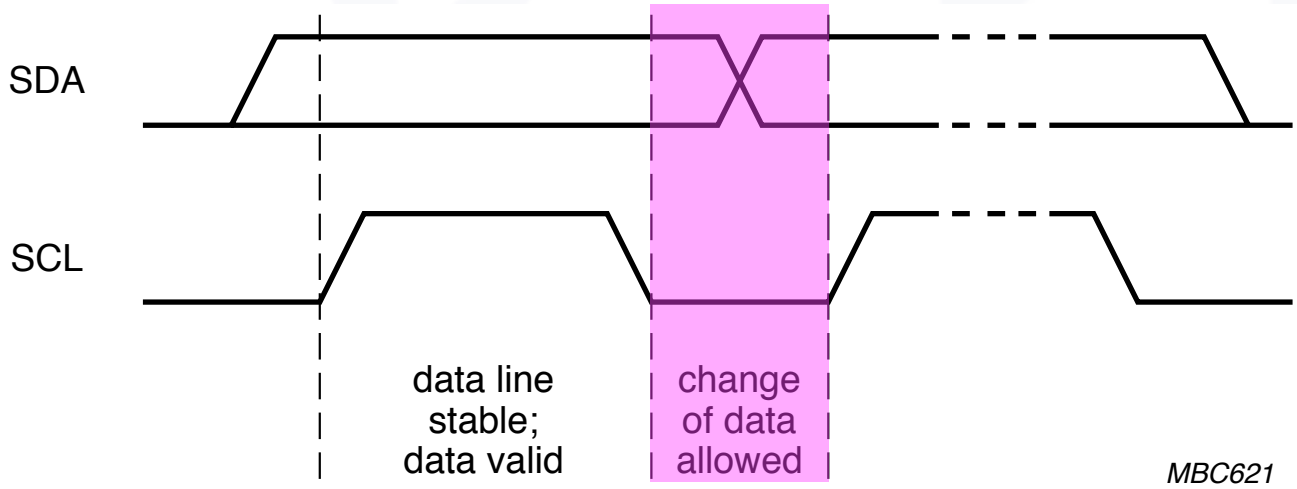
- Open-drain buses



PARAMETER	SYMBOL	STANDARD-MODE		FAST-MODE		UNIT
		MIN.	MAX.	MIN.	MAX.	
LOW level input voltage: fixed input levels V_{DD}-related input levels	V_{IL}	 −0.5 −0.5	 1.5 $0.3V_{DD}$	 n/a −0.5	 n/a $0.3V_{DD}$ (1)	 V V
HIGH level input voltage: fixed input levels V_{DD}-related input levels	V_{IH}	 3.0 $0.7V_{DD}$	 (2) (2)	 n/a $0.7V_{DD}$(1)	 n/a (2)	 V V
Hysteresis of Schmitt trigger inputs: $V_{DD} > 2$ V $V_{DD} < 2$ V	V_{hys}	 n/a n/a	 n/a n/a	 $0.05V_{DD}$ $0.1V_{DD}$	 – –	 V V

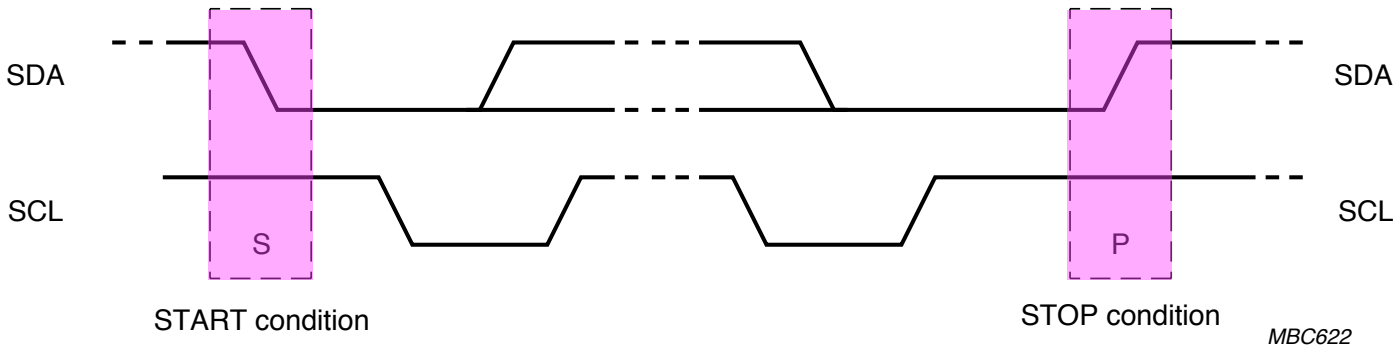
Bit Transfer

- Data validity
  - The data on the SDA line must be stable during the HIGH period of the clock
  - The HIGH or LOW state of the data line can only change when the clock signal on the SCL line is LOW



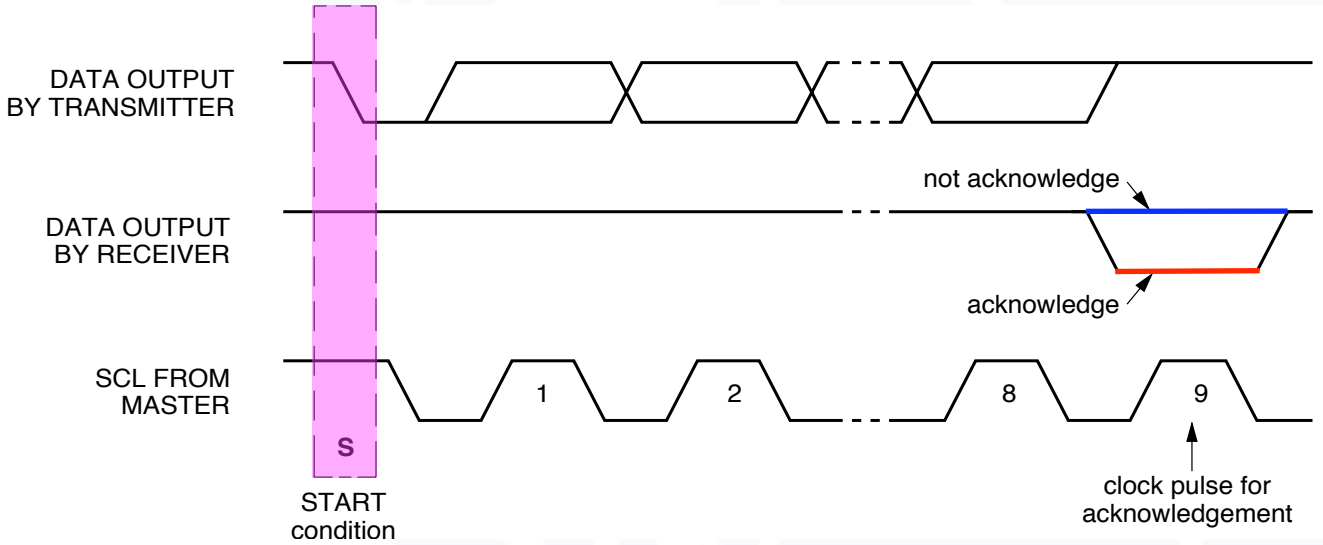
Bit Transfer

- START and STOP conditions
  - Start condition
    - A HIGH to LOW transition on the SDA line while SCL is HIGH
  - Stop condition
    - A LOW to HIGH transition on the SDA line while SCL is HIGH
  - START and STOP conditions are always generated by the master.
  - The bus is considered to be busy after the START condition. The bus is considered to be free again a certain time after the STOP condition.



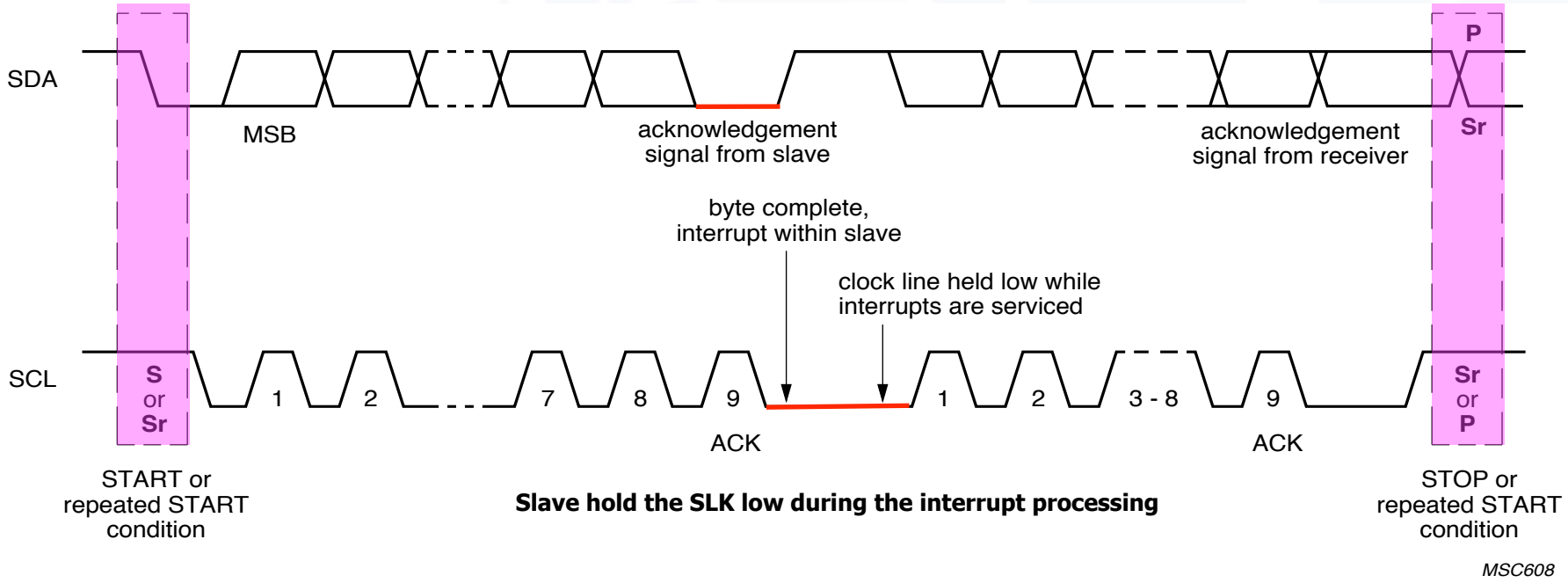
Transferring Data

- Acknowledge
  - Data transfer with acknowledge is obligatory
  - The acknowledge-related clock pulse is generated by the master
  - The transmitter **releases the SDA line (HIGH)** during the acknowledge clock pulse
  - The receiver must **pull down** the SDA line during the acknowledge clock pulse so that it remains stable LOW during the HIGH period of this clock pulse



Transferring Data

- Data transfer diagram



Transferring Data

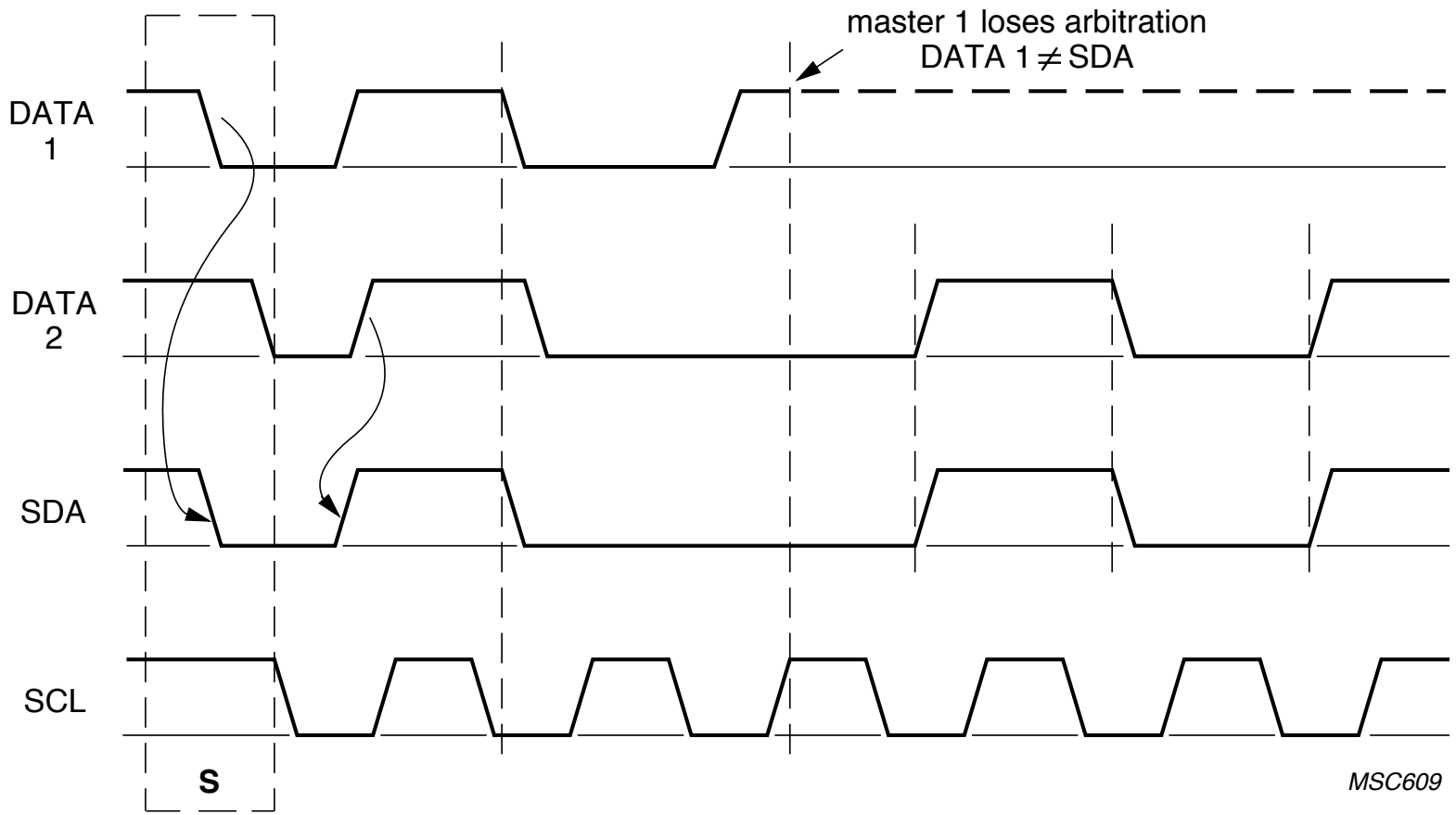
- Byte format
  - Every byte put on the SDA line must be 8-bits long
  - The number of bytes that can be transmitted per transfer is unrestricted
  - Each byte has to be followed by an acknowledge bit
  - Data is transferred with the most significant bit (MSB) first

Arbitration

- Arbitration
  - If the bus is not busy (after STOP condition), any master can initiate a transfer
  - Arbitration process required when more than one master try to initiate a transfer at the same time
  - Logic '0' has higher priority over logic '1' in open-drain bus (wired AND)
    - If any device pulls down the bus, it is pulled down
  - When a master pulls up the bus, it checks if the bus is really pulled up
    - If not, it is occupied by some other devices, so the master releases the bus
  - A transfer to a lower address device has a higher priority
  - No information is lost during arbitration process

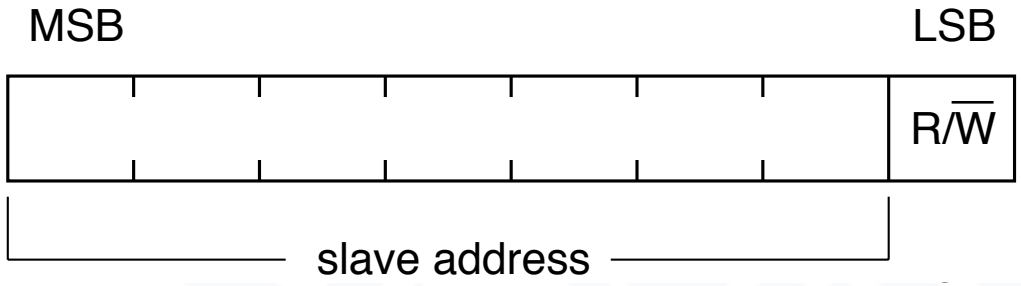
Arbitration

- Arbitration procedure of two masters



7-Bit Addressing

- The addressing procedure for the I2C-bus determines which slave will be selected by the master
  - The first byte after the START condition
- Definition of bits in the first byte
  - The first seven bits of the first byte make up the slave address
  - When an address is sent, each device in a system compares the first seven bits after the START condition with its address
  - If they match, the device considers itself addressed by the master as a slave-receiver or slave-transmitter, depending on the R/Wbit



7-Bit Addressing

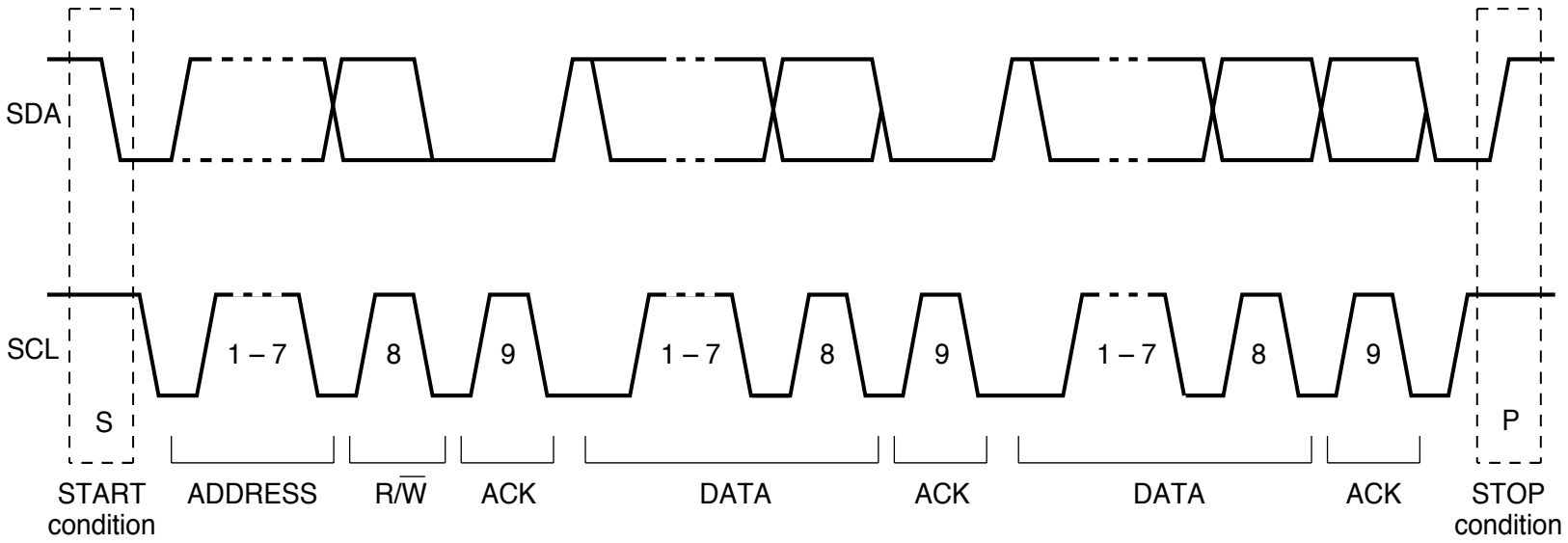
- Definition of bits in the first byte

Table 2 Definition of bits in the first byte

SLAVE ADDRESS	$\overline{\text{R/W}}$ BIT	DESCRIPTION
0000 000	0	General call address
0000 000	1	START byte(1)
0000 001	X	CBUS address(2)
0000 010	X	Reserved for different bus format(3)
0000 011	X	Reserved for future purposes
0000 1XX	X	Hs-mode master code
1111 1XX	X	Reserved for future purposes
1111 0XX	X	10-bit slave addressing

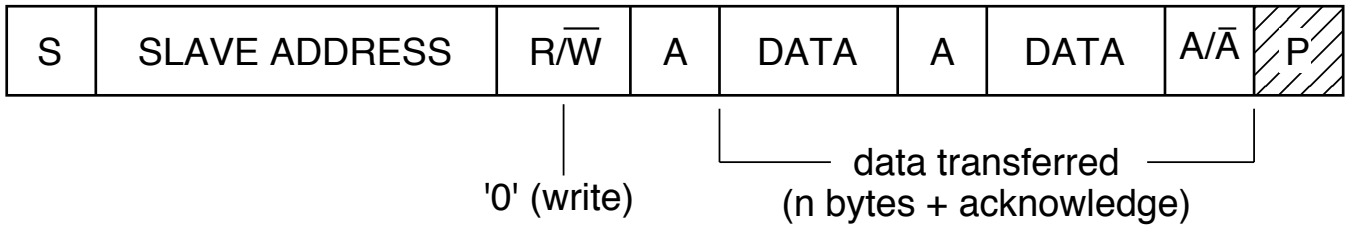
Formats with 7-Bit Addresses

- A complete data transfer



Formats with 7-Bit Addresses

- A master-transmitter addressing a slave receiver with a 7-bit address
  - Master is a transmitter
  - Slaver is a receiver
  - Transfer direction is not changed



from master to slave

from slave to master

MBC605

A = acknowledge (SDA LOW)

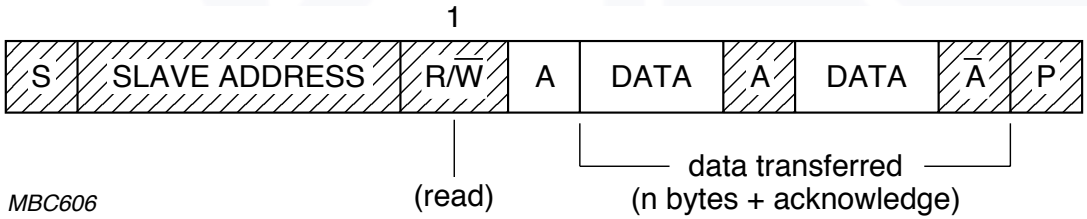
$\overline{A}$ = not acknowledge (SDA HIGH)

S = START condition

P = STOP condition

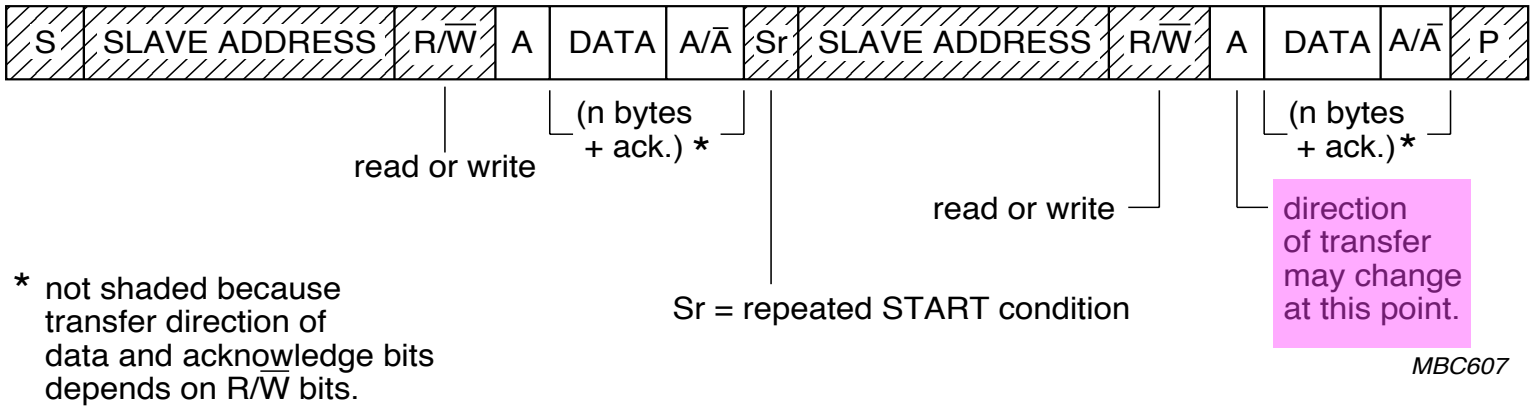
Formats with 7-Bit Addresses

- A master reads a slave immediately after the first byte
  - At the moment of the first acknowledge
    - The master-transmitter becomes a master-receiver
    - The slave-receiver becomes a slave-transmitter
  - This first acknowledge is still generated by the slave
  - The STOP condition is generated by the master



Formats with 7-Bit Addresses

- Combined format



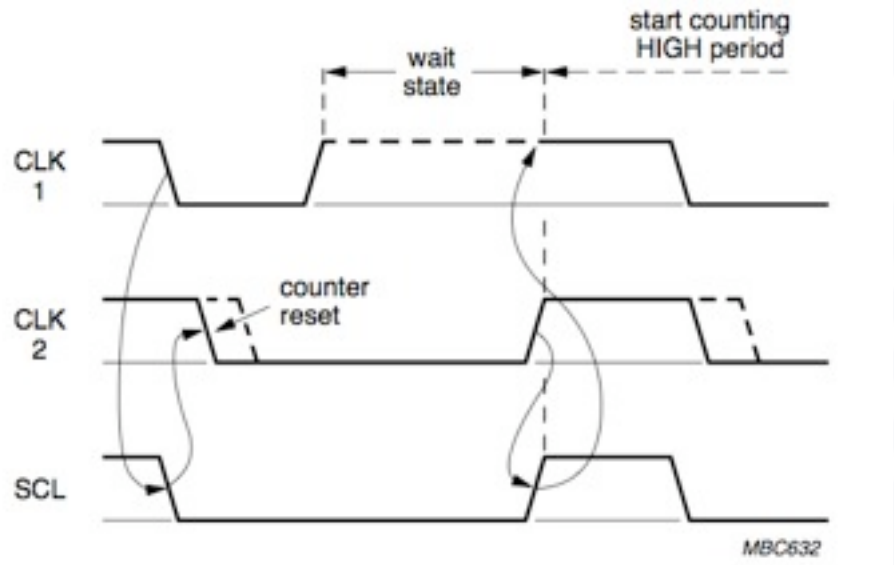
10-Bit Addressing

- If 7 bits are not enough to identify devices, 10-bit addressing is used
- Since I2C is 8-bit oriented, 10-bit address is transfered using 2 frames
- Last 3 bits of the first frame and 7 bits of the second frame is combined into 10-bit address
- First frame start with 11110

First frame	1	1	1	1	0	A0	A1	A2
Second frame	A3	A4	A5	A6	A7	A8	A9	R/W

Clock Synchronization

- All masters generate their own clock on the SCL line to transfer messages on the I2C-bus
- Clock synchronization is performed using the wired-AND connection of I2C interfaces to the SCL line
- A synchronized SCL clock is generated with its LOW period determined by the device with the longest clock LOW period
- A HIGH period determined by the one with the shortest clock HIGH period



ELPL Embedded Low-Power Laboratory

Embedded System Application

4190.303C

2010 Spring Semester

SPI Bus

Naehyuck Chang
Dept. of EECS/CSE
Seoul National University
naehyuck@snu.ac.kr

Basic Characteristics

- SPI - Serial Peripheral Interface Bus
  - De facto standard
    - No unique specification or protocol defined
    - Many variants exist
  - Full duplex communication
  - Single master, single slave without chip-enable signals
    - Multi slave available with multiple chip-enable signals
    - Daisy-chain available with single chip-enable signal

Basic Characteristics

- SPI - Serial Peripheral Interface Bus
  - Complete protocol flexibility for the bits transferred
    - Not limited to 8-bit words
    - Arbitrary choice of message size, content, and purpose
  - Extremely simple hardware interface
    - Typically lower power requirements than I^2C due to less circuitry (including pull-ups)
    - No arbitration or associated failure modes
    - Slaves use the master's clock, and do not need precision oscillators
  - Transceivers are not needed
    - Logic-level on-board communication
  - No electrical specification
    - No formal specification on logic level or input capacitance
    - General digital signals can be used, but all participating devices should match or have tolerance

Signals

- The SPI bus specifies four logic signals (4-wire)
  - SCLK (SCK)
    - Serial Clock (output from master)
  - MOSI/SIMO (SDI, DI, SI)
    - Master Output, Slave Input (output from master)
  - MISO/SOMI (SDO, DO, SO)
    - Master Input, Slave Output (output from slave)
  - SS (nCS, CS, nSS, STE)
    - Slave Select (active low; output from master)
- SPI is an on-chip bus and shares GND
- 3-wire variant
  - Replace MOSI and MISO with a single data line SISO (Slave In/Slave out)

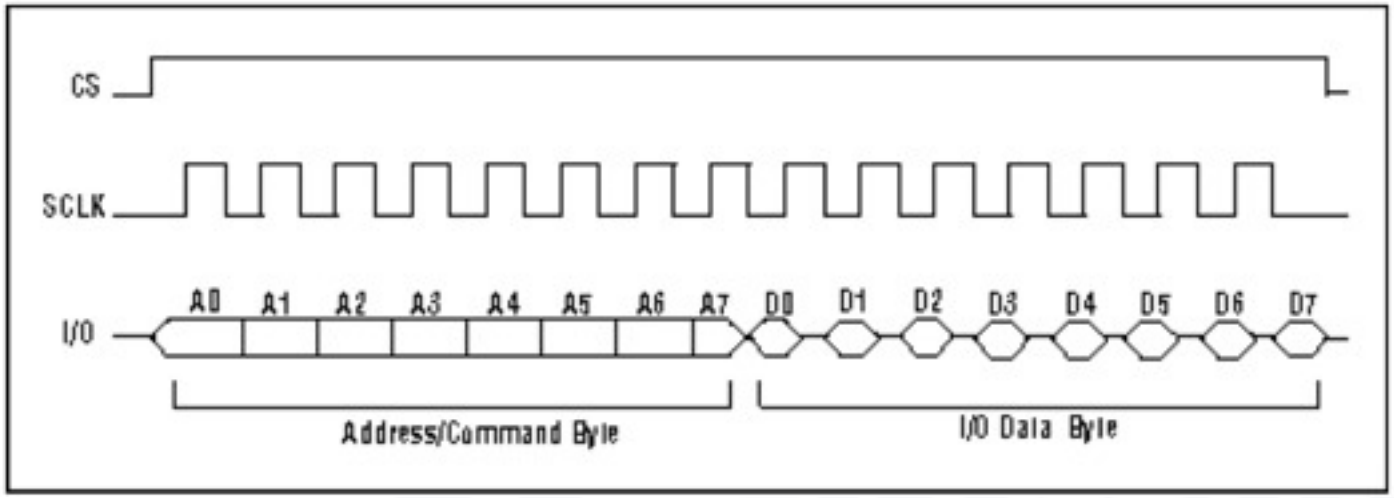
Note: I²C, SPI and 3-Wire Naming Convention

- Some venders call SPI as 3-wire as well

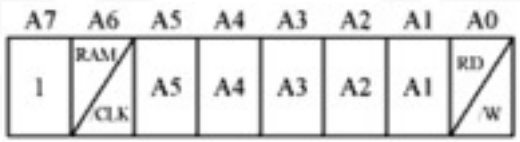
Bus	# Bus Pins	Signals	Signal Description	Bus Speed (Typical)	Data Format
I²C	2	SDA	Data in/out	100kHz to 400kHz	MSB first, LSB last
		SCL	Clock		
SPI	4	DIN	Data in	1MHz to 10MHz	MSB first, LSB last
		DOUT	Data out		
		active-low CS	/Chip select		
		SCLK	Clock		
3-Wire	3	I/O	Data in/out	500kHz to 5MHz	MSB first, LSB last
		CS	Chip select		
		SCLK	Clock		

3-Wire Single Byte Protocol

- 3-wire protocol



- 3-wire command byte

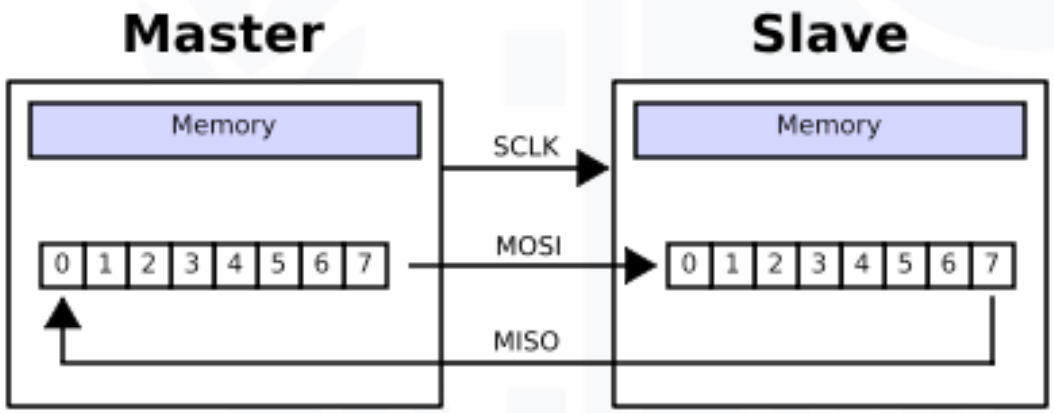


Basic Characteristics

- Disadvantages of SPI
  - Requires more pins on IC packages than I²C, even in the "3-Wire" variant
  - No in-band addressing; out-of-band chip select signals are required on shared busses
    - In-band addressing: addressing (selecting device) signals share data bus
    - Out-of-band addressing: addressing (selecting device) signals are separated from data bus
  - No hardware flow control
    - Sender cannot check READY status of the receiver
  - No slave acknowledgment
    - Sender cannot confirm a successful data transmission
  - Multi-master busses are rare and awkward, and are usually limited to a single slave
    - Without a formal standard, validating conformance is not possible (de facto standard)
  - Only handles short distances compared with RS232, RS485, or CAN

Operation

- Data Transmission
  - The master enables the chip select pin for the desired device
  - During each SPI clock cycle, a full duplex data transmission occurs
    - The master sends a bit on the MOSI line; the slave reads it from that same line
    - The slave sends a bit on the MISO line; the master reads it from that same line
  - Word size (number of bits) should be pre-defined identically between master and slave before transmission



Operation

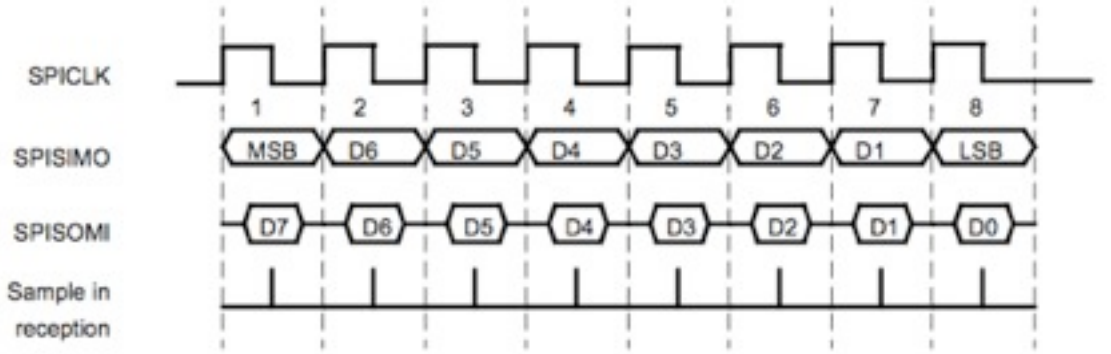
- No standard protocol defined
  - Master and slave should agree in advance on
    - word size (8, 16, 32, ...)
    - bit ordering (MSB or LSB first)
    - clock polarity
    - clock phase
  - Usually, microprocessor can be configured for various slave devices by configuring SPI control registers
    - For example, in TMS470 SPI controller has 3 control registers
      - SPICTRL1: clock speed, word size
      - SPICTRL2: clock polarity, phase
      - SPICTRL3: DMA, interrupt

Operation

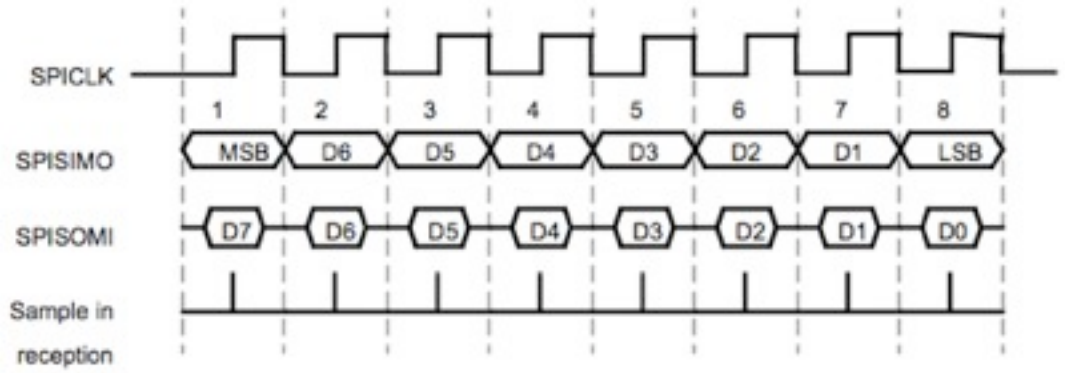
- Clock polarity and phase
  - Clock polarity
    - Clock low when idle
    - Clock high when idle
  - Clock phase
    - Read data on rising edge and change on falling edge
    - Read data on falling edge and change on rising edge
  - Four combinations are available, and should be pre-defined between master and salve

Operation

- Clock polarity and phase
  - Polarity low; Phase rising edge

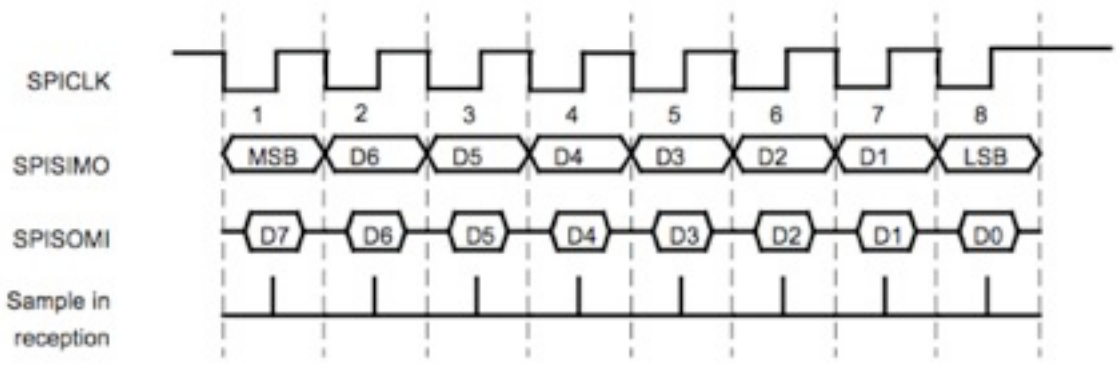


  - Polarity low; Phase falling edge

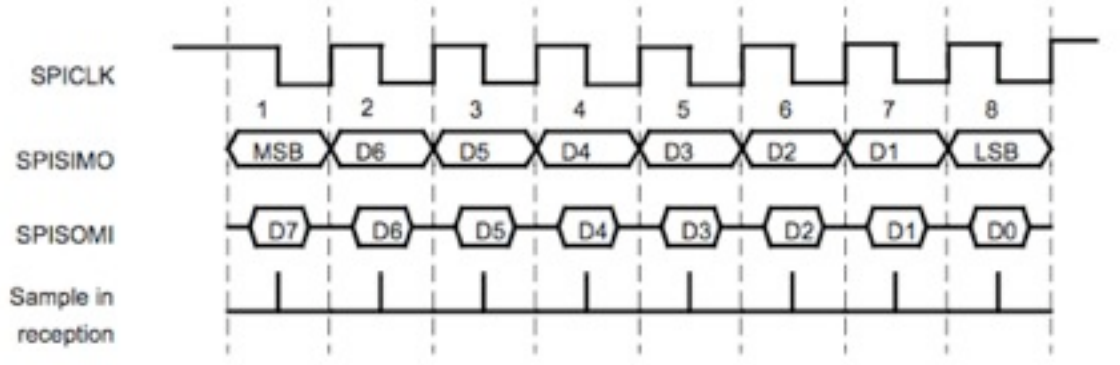


Operation

- Clock polarity and phase
  - Polarity high; Phase rising edge

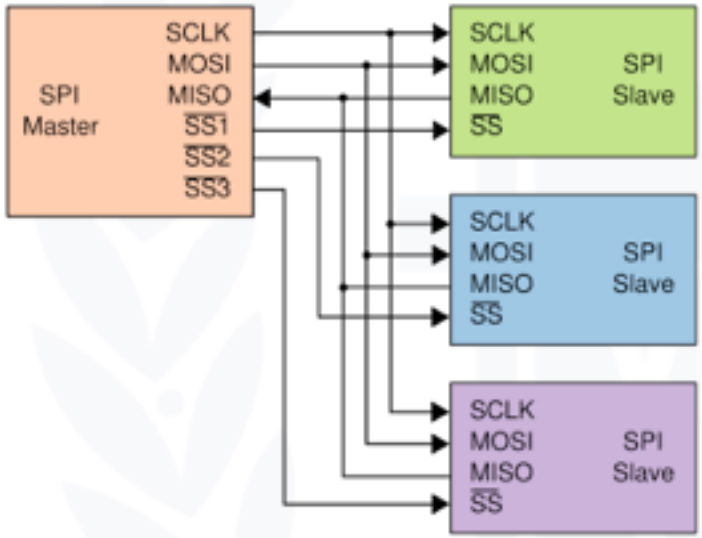


  - Polarity high; Phase falling edge



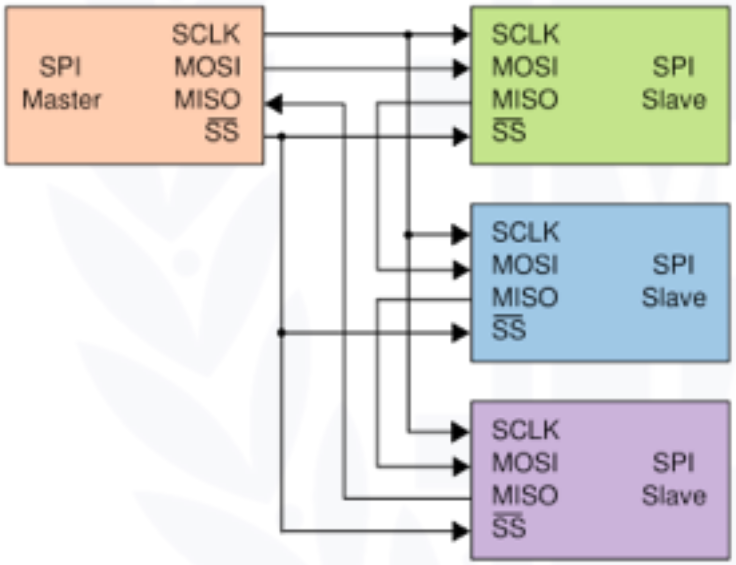
Operation

- Independent slave SPI configuration
  - Each slave communicates with the master in a 1-to-1 manner
  - Each slave is enabled one by one
  - Master needs as many enable pins as the number of slaves



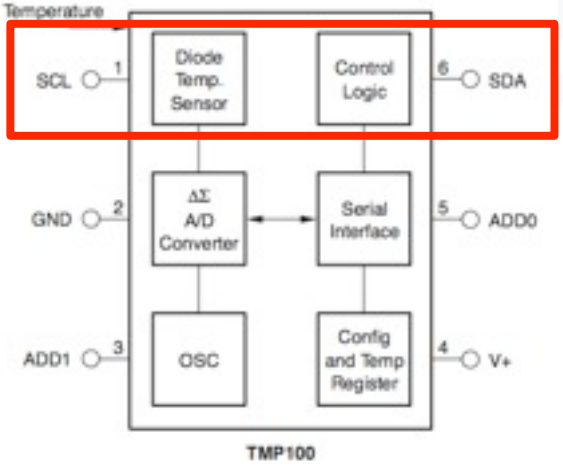
Operation

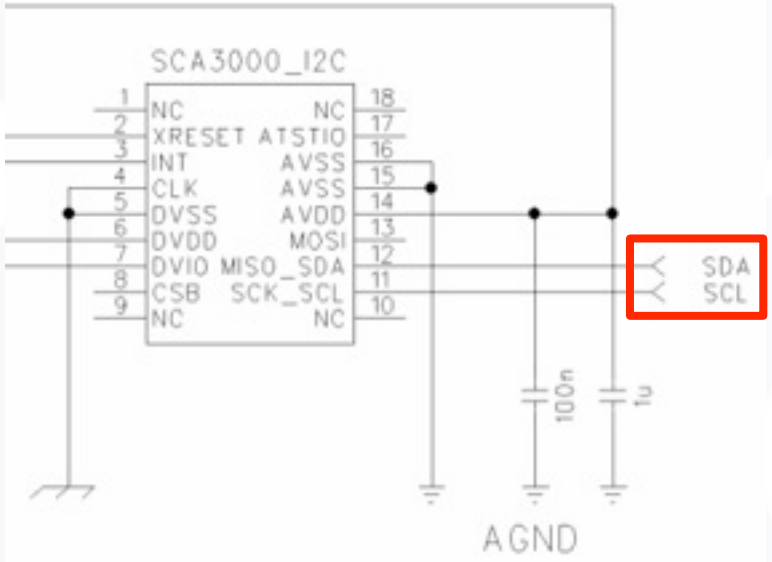
- Daisy chain SPI configuration
  - Some products with SPI bus are designed to be capable of being connected in a daisy chain configuration
  - The SPI port of each slave is designed to send out during the second group of clock pulses an exact copy of what it received during the first group of clock pulses
  - All slaves are enabled all the time
  - Master needs only one enable pin



Application

- Applications of SPI and I2C are overlapped widely
- Sensors
  - ex) Temperature sensor and accelerometer





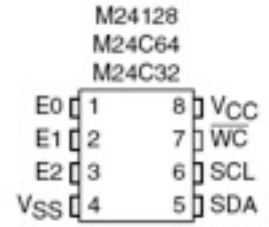
- Register read/write
  - ex) CMOS camera module

Application

- Serial memory devices
  - Small non-volatile memory devices use a serial interface
  - ex) Serial EEPROM, flash memory

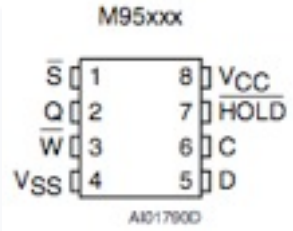
M24128
M24C64 M24C32

128 Kbit, 64 Kbit and 32 Kbit serial I²C bus EEPROM



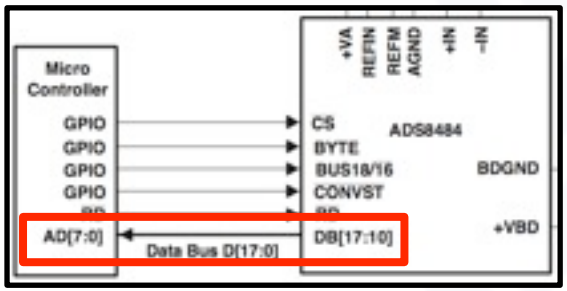
M95320 M95320-W M95320-R
M95640 M95640-W M95640-R

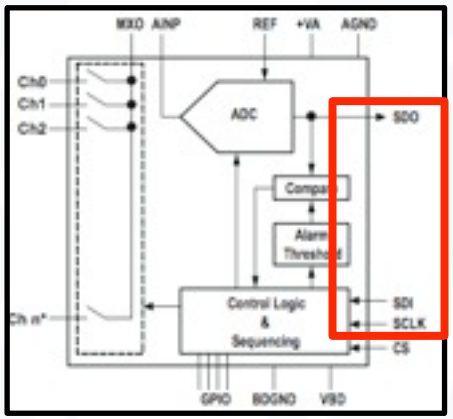
32 Kbit and 64 Kbit serial SPI bus EEPROMs with high-speed clock

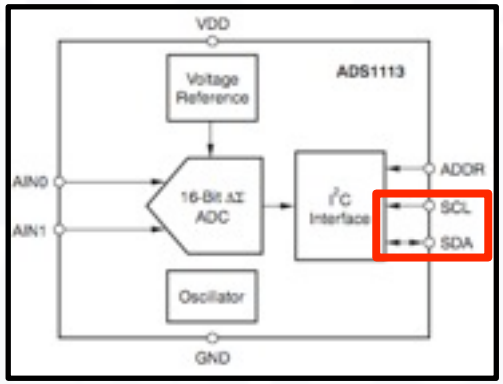


Application

- Simple ICs
  - ADCs, ADCs, and potentiometers
  - ex) Texas Instruments ADC interface
    - Parallel interface: 197 devices
    - Serial SPI interface: 248 devices
    - Serial I2C interface: 14 devices







You've Selected:
Analog to Digital Converters

Select Device Parameters

442 Results

D-Type Latch (3)	Serial I2C Interface (14)	Digital Input (1)	Internal Reference Buffer (64)	Parallel Interface (197)
Serial Interface (13)	On-Chip Driver (3)	Differential Inputs (316)	LVDS (89)	Serial SPI Interface (248)