SERIAL COMMUNICATION INTERFACE

P. C. Pandey

EE Dept, IIT Bombay

Feb’04, rev. March’05, rev. Apr’16

<pcpandey@ee.iitb.ac.in> serial_comm_01apr16.doc
1. Interfacing for Data Communication between Processors & Digital Peripherals

- Multiple processors to handle complex tasks
- A central processor & dedicated processors for specialized tasks
- Multiple remote sensors, actuators, embedded systems for distributed data acquisition and control
1.1 Communication Port
Collection of signal wires: data, handshake/control/status, clock

1.2 Data Communication Modes
• Parallel: Several data bits at a time
• Serial: Single data bit at a time

1.3 Serial Communication Modes
• Asynchronous communication (UART, ACIA, SCI, etc.): Clock generated at Tx and Rx with the same nominal value. Clock not transmitted.
• Synchronous serial communication (SRT, SPI, I2C, etc.): Clock generated by the master; used by Tx & Rx; transmitted using a separate line or by combining it with data (Manchester coding).
1.4 Serial Communication Standards

- Interface Logic Levels
- Physical Link (cables & connectors)
- Data Transfer Protocol
- Bandwidth, Noise, Range

1.5 Communication Devices

- Data terminal equipment (DTE): computer, terminal, etc.
- Data communication equipment (DCE): modem, printer, etc.
1.6 Data Frame

Non-divisible packet of bits (start bit, data bits, error checking/correction bits, stop bits)

- Bit Time: basic time interval, Bit Rate: no. of bits / s
- Baud Rate: no. of pulses / s
- Data: information data bits
- Overhead: start / stop / parity, synchronization messages, etc.
- Data Bandwidth / Throughput: no. of information bits (excluding overhead) / s
1.7 Simplex/Duplex Communication

- Simplex: Information transfer in one direction only (excluding status / handshakes, etc.).
- Half-duplex: Information transfer in one direction at a time.
- Full-duplex: Simultaneous bi-directional information transfer.
### 1.8 Communication Logic Levels

- **CMOS (processor port pins):** true/mark: $\approx 5\text{V}$, false/space: $<0.1\text{V}$.  
- **RS 232 (drivers):** Negative logic, Non-return-to-zero (NRZ), true/mark: $-12\text{V}$, false/space: $+12\text{V}$, idle state: true ($-12\text{V}$).  
- **Differential voltage (drivers):** To reduce the effect of electrostatic interference and ground noise. RS 485: true/mark: $-3\text{V}$, false/space: $+3\text{V}$.  
- **Open collector (processor port pins / drivers):** Low & high Z, with passive pull-up.  
- **Tri-stated (processor port pins / drivers):** Low, high, & high Z (idle).  
- **Current loop (drivers, 4/20 mA):** To reduce the effect of inductive interference.  
- **Opto-coupler:** For electrical isolation.
1.9 Serial Bus with Multi-drop Network

- Tri-state logic: Disable the driver after transmission. RTS = 0 for transmission, RTS = 1 after completion. Data: 0-127, Address: 128-255.

- Collision detection & avoidance: Transmit a frame. Receive it & check for integrity. If collision is detected, wait for a random delay & retransmit.

1.10 Data Transfer Protocols

- Fixed length messages
- Message length after the address
- Special character as terminator
1.11 Interconnection Topology

Point-to-point

Multi-drop

Star

Ring
2. Asynchronous Communication

No clock transmission. Only data & handshake lines. Tx & Rx use local clocks with same nominal value, not synchronized.

- Transmission: Idle state, start bit, data bits, (parity, error correction bits), stop bit(s) / idle state. Reception: Detect start (1 → 0) transition, wait 1/2 bit time, sample the input at bit time intervals.
• **Clock tolerance**

  \[
  \text{Tx bit time} = T_b
  \]
  \[
  \text{Rx bit time} = T_b + \Delta
  \]

  Cumulative error = \( N\Delta < 0.5T_b \)
  
  \( (N = \text{No. of bits (including start, excluding stop/idle}) \)
  
  \[ \Rightarrow \text{For } N=10, \Delta/T_b < 5\%. \]

• **Baud rate**: Limited by clock tolerance.

• **Throughput (data bandwidth)** for a given baud rate: Low due to overheads per frame and small frame size.
3. Synchronous Communication

Tx & Rx use clock generated by the master.

- Output at one clock edge (falling) & input at the other edge (rising).
- Signal lines: Data, Clock, [Select] (Clock & data may be combined)
- Clock [& select] generated by the master
- Drivers may be needed
- No basic restriction on frame length
4. Interface Cables & Connectors

4.1 Cables

- Parallel wires
  - Higher possibility of interference between lines carrying signal in opposite directions. Ground between critical lines.
  - Suited for short distance, high throughput.
• **Shielded cable**
  - Shield connected to frame ground at one end
    (signal ground → power supply ground)
  - Reduced RF & electrostatic interference

• **Twisted pair**
  - Reduced inductive pick-up
  - Baud rate limited due to increased capacitive loading

### 4.2 Connectors

 DB25 / RS232: 1-13,14-25; DB9 / E1A-574: 1-5, 6-9;
 RJ45: 1-8 (Telephone type jack)
5. Serial Interface Standards

- **Single ended**
- **Differential**
- **Differential balanced**

Diagram showing the different types of serial interface standards.
Problem to be tackled

- Signal attenuation (caused by loading & distance)
- Pulse transition delay / double pulsing due to reflection (caused by impedance mismatches & sharp transition)
- Interference between signal lines
- External pick up (electrostatic, inductive, electromagnetic)
- Difference in ground potential
- Overvoltage & overcurrent
Some solutions

- Large voltage or current levels
- Trapezoidal pulses
- Matched termination
- Use of current loop
- Differential voltage transmission
- Special cables: Shielded (reduces EM pick up), Grounded shielded (reduces EM & electrostatic pickup), Twisted pair (reduces inductive pickup)
5.1 Simple Digital Logic

Simple & inexpensive, for short distance on the same board or in the same box.
5.2 Full Duplex with Drivers

5.3 Simplex with Driver
5.4 Ring Network

- Communication by address/data format
- Data packet received checked for address. Retransmitted if for another node
- No collision of data packet
5.5 Half Duplex Link with Tri-state Logic

Data link with local echo-back to Rx

- Possibility of collision: Detection & recovery needed
- $RTS = 0$ during transmission.
- Can be used for forming network by putting Rx buffers to avoid loading.
5.6 Half Duplex Link with Open Drain Logic

- **Driver**: open drain gate or processor pin.
- **No enabling/ disabling**.
- **Master/s slave protocol needed for implementing collision detection / avoidance.**
5.7 Serial Bus with Open-Drain.Half-Duplex Link

Master/slave protocol: collision detection & avoidance.

- Destination address followed by data packets (8-bit frame. Data:0-127, address: 128 255)

- Data termination
  - Fixed length data
  - Termination character (eg. FF).
  - Length as 2\textsuperscript{nd} character.
5.8 Isolated Digital Logic Link

Ground isolation → Ground noise isolation
5.9 Current Loop

- True: 20 mA, False: 0-4 mA
- Rejection of inductive pick-ups
- Optical coupling for electrical isolation

Polarity-Insensitive Current Loop
6. RS 232 Serial Link

Negative, Non-return to zero (NRZ) logic

- Tx: ± 5V to ±15 V
- Rx: threshold: ±3V
6.1 Signaling

- Single ended link
- Shielded cable with shield connected to frame ground at DTE.
- Signal ground connected to power supply ground at both ends.

6.2 Connectors

DB25 (RS 232): 25 pins, 21 signals
DB9 (EIA 574): 9 pins, 9 signals
RJ45 (EIA 561): 8 pins, 8 signals
### Serial Port Pins & Signals

<table>
<thead>
<tr>
<th>DB 25 (RS 232)</th>
<th>DB9 (EIA 574)</th>
<th>RJ 45 (EIA 561)</th>
<th>Signal (true)</th>
<th>DTE (in/ out)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>Frame Ground</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>6</td>
<td>TxD</td>
<td>-12V Out</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>5</td>
<td>RxD</td>
<td>-12V In</td>
</tr>
<tr>
<td>4</td>
<td>7</td>
<td>8</td>
<td>RTS</td>
<td>+12V Out</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>7</td>
<td>CTS</td>
<td>+12V In</td>
</tr>
<tr>
<td>6</td>
<td>6</td>
<td></td>
<td>DSR</td>
<td>+12V</td>
</tr>
<tr>
<td>7</td>
<td>5</td>
<td>4</td>
<td>Signal ground</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>1</td>
<td>2</td>
<td>Data carrier detect</td>
<td>+12V</td>
</tr>
<tr>
<td>15</td>
<td></td>
<td></td>
<td>Tx clock</td>
<td>In</td>
</tr>
<tr>
<td>17</td>
<td></td>
<td></td>
<td>Rx clock</td>
<td>In</td>
</tr>
<tr>
<td>18</td>
<td></td>
<td></td>
<td>Local loop back</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>4</td>
<td>3</td>
<td>DTR</td>
<td>Out</td>
</tr>
<tr>
<td>22</td>
<td>9</td>
<td>1</td>
<td>Ring indicator</td>
<td>+12V In</td>
</tr>
</tbody>
</table>
6.3 Drivers

e.g., Max 232
±12 V from 5 V supply using charge pump & four 100nF capacitors
6.4 RS232 i/o specs

Output specs

• Short circuit protection: short to ground or any signal line

• True: $-15 \text{ V} \leq V_{\text{out}} \leq -5 \text{ V}$, False: $+15\text{ V} \geq V_{\text{out}} \geq +5\text{ V}$

• Max: $|V_{\text{out}}| < 25\text{ V}$, $I_{\text{omax}}$ (short ckt current) < 0.5 A

• Transition time ($-3 \text{ V}$ to $+3 \text{ V}$) $\leq 4\%$

Input specs

• $|dV_{\text{in}}/dt| \leq 30 \text{ V}/\mu\text{S}$

• True: $-15 \text{ V} \leq V_{\text{in}} \leq -3 \text{ V}$, False: $3 \text{ V} \leq V_{\text{in}} \leq 15 \text{ V}$

• $R_{\text{in}}$: 3 – 7 kΩ, $C_{\text{in}} \leq 2500 \text{ pF}$
7. RS 485: Multi-Tx Multi-Rx Serial Link with Balanced Differential Transmission

Balanced differential signaling
- Larger signal swing with same supply; Common mode noise rejection
- Ground may not be explicitly connected
- Max. distance decreases with baud rate

![Diagram of RS 485 connection]
7.2 Specifications

- No. of nodes (Tx/Rx) ≤ 32 nodes

- True : \(-5 \text{ V} < V_{\text{out}} < -1.5 \text{ V}; \quad V_{\text{in}} < -0.2 \text{ V}\)
  
  False : \(+1.5 \text{ V} < V_{\text{out}} < +5 \text{ V}; \quad V_{\text{in}} > +0.2 \text{ V}\)
  
  Transn. : \(-1.5 \text{ V} < V_{\text{out}} < +1.5 \text{ V}; \quad |V_{\text{in}}| < +0.2 \text{ V}\)

- Balanced o/p & i/p impedances
  
  \(R_{o+} \approx R_{o-} \approx 54 \Omega, \quad R_{i+n} \approx R_{i-n} > 12 \text{ K}\Omega\)

- Termination for high baud rate or long distance, so that \(R_{\text{term.}} \approx 50 \Omega\)
  
  (combination of all terminations) \(\rightarrow 120 \Omega\) termination at two far ends.
7.2 Multi-drop Half Duplex Link

- Trapezoidal o/p; Output short circuit protection (against data collision, wrong connection etc.); High Z o/p when disabled; Surge protection on i/p.

- To avoid false data reception in idle state (all drivers disabled), i/p must be in a defined state, using internal pull-up & pull down.

- External driver IC’s preferred.
7.3 RS485 Cable Terminations

• Unterminated Configuration
  o Minimal load on driver; Better DC noise margin (noise margin : driver swing – receiver sensitivity)
  o Clock rate and distance restriction due to signal reflections on the cable (time to traverse < bit interval, rise time > 4 propagation time): < 200 kbps, short distances;

• Parallel Termination
  o Termination at two ends : $R_T \sim 1.1 Z_0 \rightarrow$ negligible reflection
  o Increased loading & reduced noise margin
  o Disturbance of Rx internal biasing
• **Power Termination**

![Power Termination Diagram]

• **AC Termination**
  - No effect at lower frequencies;
  - No static loading
  - $R_t \approx 100 - 150 \Omega$
### 7.4 RS 232 & RS 485 Specs

<table>
<thead>
<tr>
<th>Specs</th>
<th>RS 232</th>
<th>RS 485</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Mode</strong></td>
<td>Single ended</td>
<td>Differential</td>
</tr>
<tr>
<td><strong>Maximum Drivers</strong></td>
<td>1</td>
<td>32</td>
</tr>
<tr>
<td><strong>Maximum Receivers</strong></td>
<td>1</td>
<td>32</td>
</tr>
<tr>
<td><strong>Distance</strong></td>
<td>15 m</td>
<td>1.2 km</td>
</tr>
<tr>
<td><strong>Data rate</strong></td>
<td>20 kbps</td>
<td>10 Mbps</td>
</tr>
<tr>
<td><strong>Driver output Max.</strong></td>
<td>±25 V</td>
<td>–7 to 12 V</td>
</tr>
<tr>
<td><strong>Driver output loaded</strong></td>
<td>±5 V</td>
<td>±1.5 V</td>
</tr>
<tr>
<td><strong>Driver output unloaded</strong></td>
<td>±15 V</td>
<td>±5 V</td>
</tr>
<tr>
<td>$R_o$</td>
<td>3 – 7 kΩ</td>
<td>54 Ω</td>
</tr>
<tr>
<td>$R_{in}$</td>
<td>3 – 7 kΩ</td>
<td>&gt;12 kΩ</td>
</tr>
<tr>
<td><strong>Receiver input</strong></td>
<td>±15 V</td>
<td>–7 to +12 V</td>
</tr>
<tr>
<td><strong>Receiver sensitivity</strong></td>
<td>±3 V</td>
<td>±200 mV</td>
</tr>
</tbody>
</table>
8. Inter-Integrated Circuit Bus (I2C or I²C)

Two-wire multi-master multi-slave synchronous half-duplex bus

- Signals: Single-ended voltage, 100 kHz to 5 MHz, short-range.

- 2 open-drain lines with passive pull-up to +5 V or +3.3 V: Serial Data (SDA), Serial Clock.

- Bit rate: 10 kbps (low-speed), 100 kbps (standard), 400 kbps (fast mode, Fm), 1 Mbps (Fm+), 3.4 Mbps (high-speed).

- Devices with unique addresses (7, 10, or 16 bit address space).

- No. of nodes: Limited by the address space & the total bus capacitance of 400 pF.
8.1 Basic Features

• **Lines:** Serial Clock (SCL), Serial Data (SDA); Device address: 7-bit

• **Node types**
  
  Master: Generates clock; Initiates communication with slaves.
  Slave: Receives clock; Responds when addressed by the master.

• **Multi-master bus:** Master and slave roles may be changed between messages, after a STOP bit.

• **Hardware overhead:** Clock stretching by slave.

• **Protocol overheads:** Slave address, [Register address within the slave device], Per-byte ACK/NACK bits.

• **Throughput:** limited by overheads & clock stretching by slave.
8.2 Example

One master (microcontroller) & 3 slaves (ADC, DAC, microcontroller)
8.3 Data Transfer

- Operation sequence
  Master: Sends START bit, slave address (7-bit), read/write bit (write = 0, read = 1).
  Slave: Responds (after receiving the address and read/write bit) with ACK bit (active low).
  Master: Continues in Tx/Rx mode.
  Slave: Continues in complementary (Rx/Tx) mode.

- Bit sequence
  Address & data bits: SDA transitions with SCL low; MSB first.
  Start bit: SDA high-to-low transition with SCL high.
  Stop bit: SDA low-to-high transition with SCL high.
• **Write-to-Slave:** Master repeatedly sends a byte with the slave sending an ACK bit.

• **Read-from-Slave:** Master repeatedly receives a byte from the slave & sends an ACK bit after every byte but the last one.

• **End of transfer:** Master sends STOP to release the bus or another START bit to retain bus control for another transfer.

• **Logic:** Pulled low (any device) = 0, Floating (all devices) = 1.

• **Clock stretching using SCL:** Addressed slave holds SCL low after receiving (or sending) a byte, if not ready for more data. The master waits for SCL to go high. Waits for an additional minimum time (standard: 4 µs) before pulling it low.
8.4 Bidirectional Buffering & Multiplexing

- **Buffering**: Splitting large bus segments into smaller ones to limit the capacitance of a bus segment.
- **Multiplexing**: Separating multiple devices with the same address.
8.5 Timing Diagram

SDA changed after the SCL falling edge & sampled on the SCL rising edge (avoids false marker detection)

- START bit (S): SDA pulled low while SCL high.
- First bit (B1) written on SDA by Tx while SCL low. SDA read by Rx when SCL rises.
- Write & read repeated (B2, ..): SDA transitioning while SCL low; SDA read while SCL rises.
- STOP bit (P): SDA high while SCL is high.
8.6 Applications

Low pin count, Low cost, Low to moderate speed

• EEPROM for configuration data; NVRAM for user settings.
• Real-time clock; Low speed DACs and ADCs; Sensors with digital readout; Power supplies with digital control.

8.7 Limitations

• Conflict of slave addresses. May be solved by having device pins for user settable address.
• Spurious address detection due to speed mismatch.
• Throughput degradation due to clock stretching. Separate segments for low and high latency devices.
• Problems due to shared bus.
9. Serial Peripheral Interface Bus (SPI)

Four-wire single-master multi-slave synchronous full-duplex interface

- Signals: Single-ended voltage, Clock: a few MHz, Range: short.
- Lines: Clock (generated by Master), Data Out, Data In, Slave Select.
- No device addresses.
- Full-duplex synchronous communication between the master and the selected slave.
- Multiple slave configurations (i) Data-Out and Data-In lines of slaves connected in parallel with independent slave-select lines from the master, (ii) Daisy-chaining of Data-Out and Data-In lines and common slave select line.
9.1 Basic Features

- Lines
  - **SCLK**: Serial Clock output from master. (SCK / CLK)
  - **SS**: Slave Select output from master (CS / EN). Active low. One independent line from master for each slave.
• **Device types**
  o **Master (single):** Generates the clock & originates the frame for reading and writing by selecting the slave.
  o **Slave (multiple):** Selected slave reads and writes data.

• **Slave selection:** Through slave select line; No device address. Only one slave may communicate with the master.

• **Low-overhead full-duplex data transfer under complete control by Master.**
  o Clock generated by master. Rate (up to a few MHz) should be supported by slave.
  o Delay between slave select & clock to allow for slave response time.
9.2 Single Slave Configuration

9.3 Configuration with Multiple Independent Slaves

- Independent SS line for each slave. Pull-up on SS near each device to reduce cross-talk.
- Paralleled MISO & MOSI. Tri-state Slave MISO.
9.4 Data Transmission

Inter-chip circular buffer formed by the master & slave data registers. Full-duplex operation: Register contents get exchanged at the end of data transfer.

- Basic Operations
  - Master & slave data registers (of same word size: 8/12/16 bits) act as a virtual ring. Data bit shifted out from the master register (MSB first) while shifting in from the slave register (LSB first).
  - Register values exchanged after the bits have been shifted out and in.
- **Operation sequence**
  
  - **Master generates Slave Select (logic 0) & produces clock cycles,** after a delay if required for response by the Slave.
  
  - **A full-duplex transmission occurs during each clock cycle.** Master sends data bit on MOSI & slave reads it. Simultaneously, slave sends a bit on MISO & master reads it. Sequence maintained even if only one-directional data transfer intended.
  
  - **For more data exchange,** the shift registers are reloaded and process repeated. After data transfer, master stops toggling the clock & deselects the slave.
  
  - **Unless selected,** slave device disregards Clock & MOSI, & does not drive MISO.
9.5 Timing Diagram

Clock polarity & phase
(Master & Slave should use same notation)

- Polarity
  - CPOL = 0: 0 idle & 1 active
  - CPOL = 1: 1 idle & 0 active

- Phase
  - CPHA = 0: Out on active-to-idle & Sample on idle-to-active.
  - CPHA = 1: Out on idle-to-active & Sample on active-to-idle.

Data out & sample on alternate clock edges: Half clock cycle between outputting and sampling for signal stabilization.
9.6 Daisy Chain Configuration with Multiple Cooperative Slaves

- Multiple slaves connected using a single SS from the master.
- Slave daisy-chaining:
  - Master_MOSI to Slave1_MOSI,
  - Slave1_MISO to Slave2_MOSI,
  - ....
  - SlaveN-MISO to Master_MISO.
- Master and Slave data registers form a virtual ring. Data bits received by a slave during a group of clock pulses transferred out during the next group of clock pulses.
9.7 Variations

Clock handling: (i) Ignore additional clocks after the specified number of clocks, (ii) Continue shifting data bits if SS active.

Interrupts: Extra line from slave to send an interrupt to the host Master. Examples: Pen-down from touch-screen sensor, thermal limit alert from temperature sensor, alarm from RTC, headset jack insertion from the sound codec, etc.
9.7 Advantages

- Default full-duplex communication.
- High speed & good signal integrity due to push-pull output (as opposed to open-drain). Low power requirement (no pull-up resistors). No transceivers needed. Simple hardware interfacing.
- Unidirectional signals, permitting easy Galvanic isolation.
- Higher throughput due to low overheads.
- Protocol flexibility (not limited to 8-bit words); Arbitrary choice of message size.
- No arbitration or associated failure modes. Slaves do not need unique addresses, permitting multiple identical slaves.
- Low processing overhead for microcontrollers with on-chip SPI controllers capable of running in either master or slave mode.
9.8 Disadvantages

- Needs more pins (than I2C), particularly in case of multiple independent slaves.

- No hardware flow control by the slave.

- No hardware slave acknowledgment.

- Typically only one master supported.

- Many existing variations, which may not be supported by development tools like host adapters.

- Fixed configuration; No hot swapping (dynamic adding of nodes).

- Interrupt from the slave to the host master to be implemented using extra line, or by periodic polling.
9.9 Applications

Microcontrollers & FPGAs

Peripherals: ADCs, DACs, touch-screens, video game controllers, audio codecs, digital potentiometers, Digital MEMS (temperature, pressure, accelerometer, magnetometer, etc.).

Communication chips: Ethernet, USB, USART, CAN, etc.

Flash memory, EEPROM, RTC.

Display controllers: LCD controllers, LED drivers.