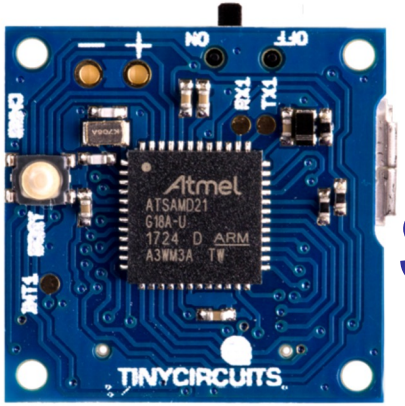
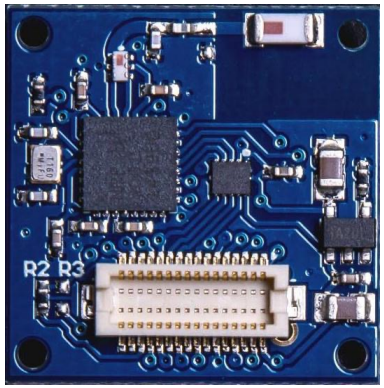
CSE190 Fall 2023



Lecture 11

Serial Busses (cont)



Wireless Embedded Systems

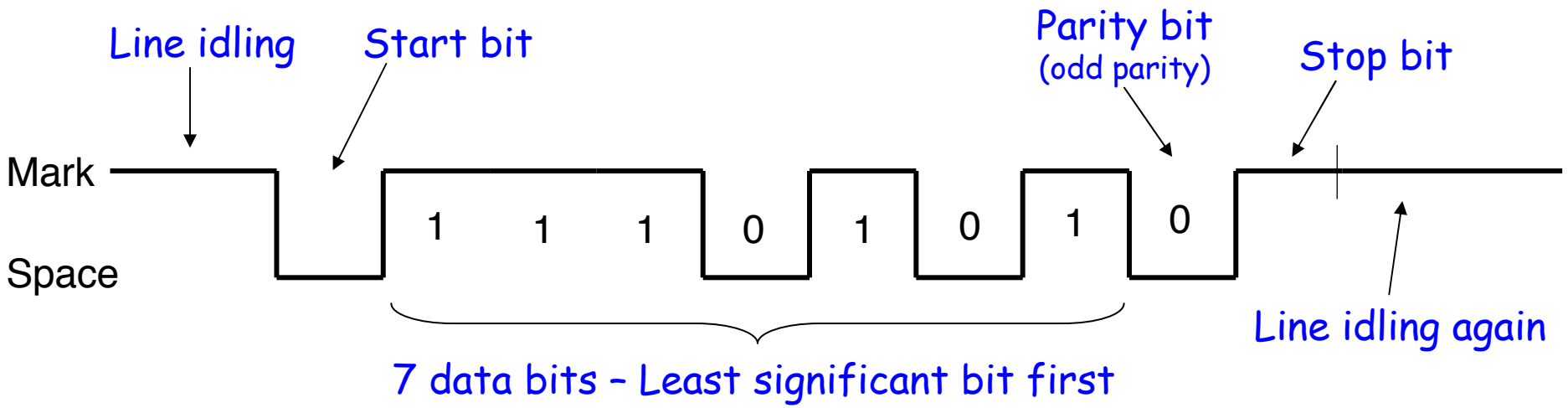
Aaron Schulman

Protocol

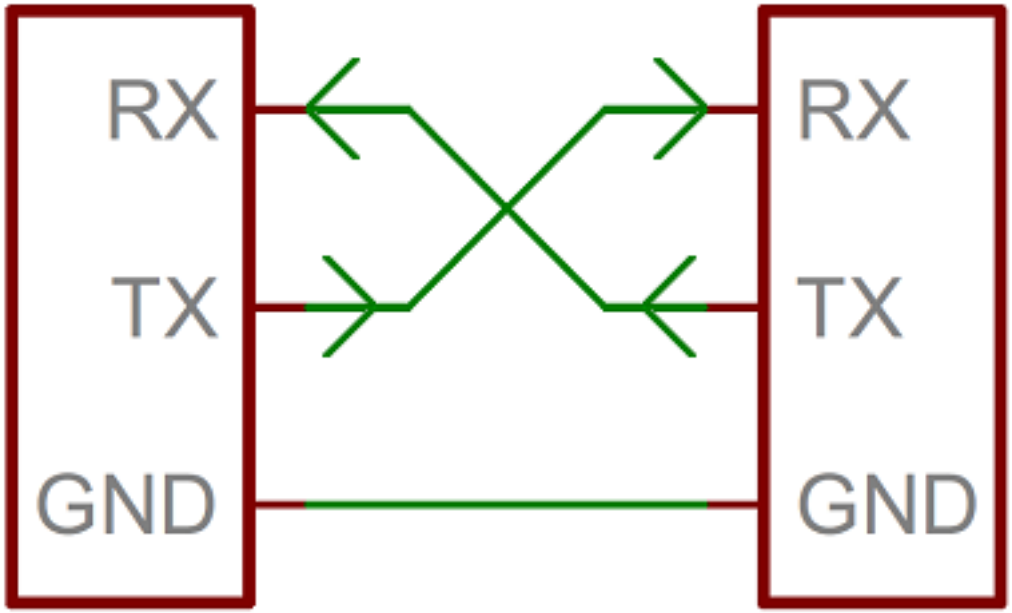
- Each character is sent as
  - a logic *low* **start** bit
  - a configurable number of data bits (usually 7 or 8, sometimes 5)
  - an optional parity bit
  - *one or more logic high* **stop** bits
  - with a particular bit timing (“baud”)

UART Example

- Send the ASCII letter ‘W’ (1010111)

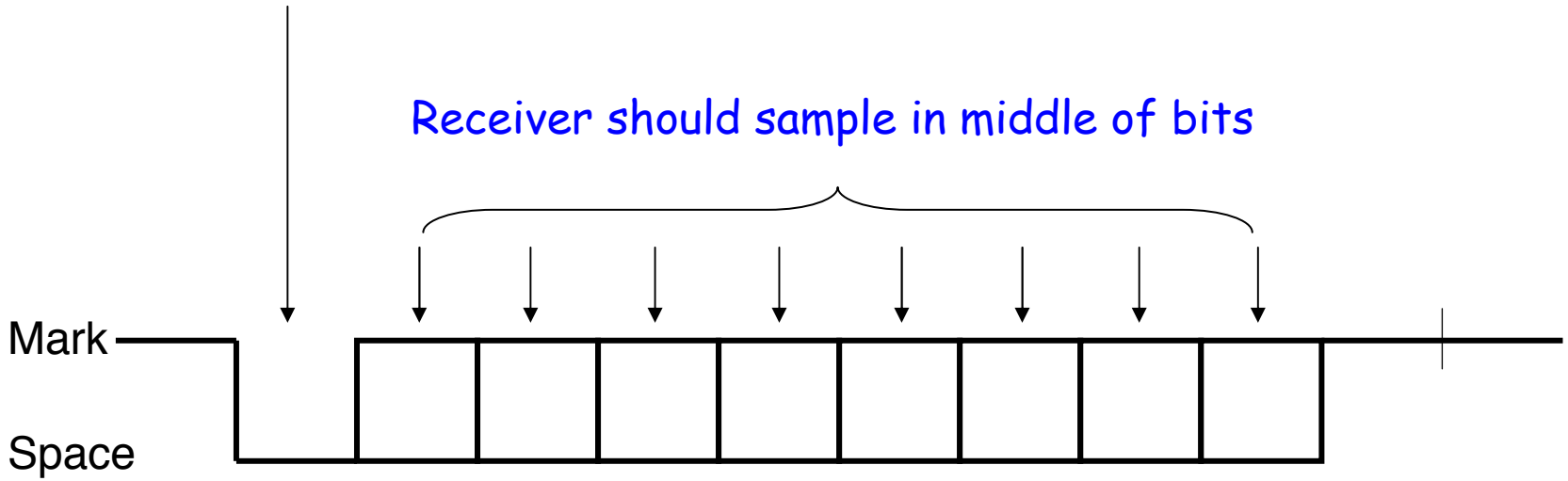


UART Hardware Connection



UART Character Reception

Start bit says a character is coming,
receiver resets its timers

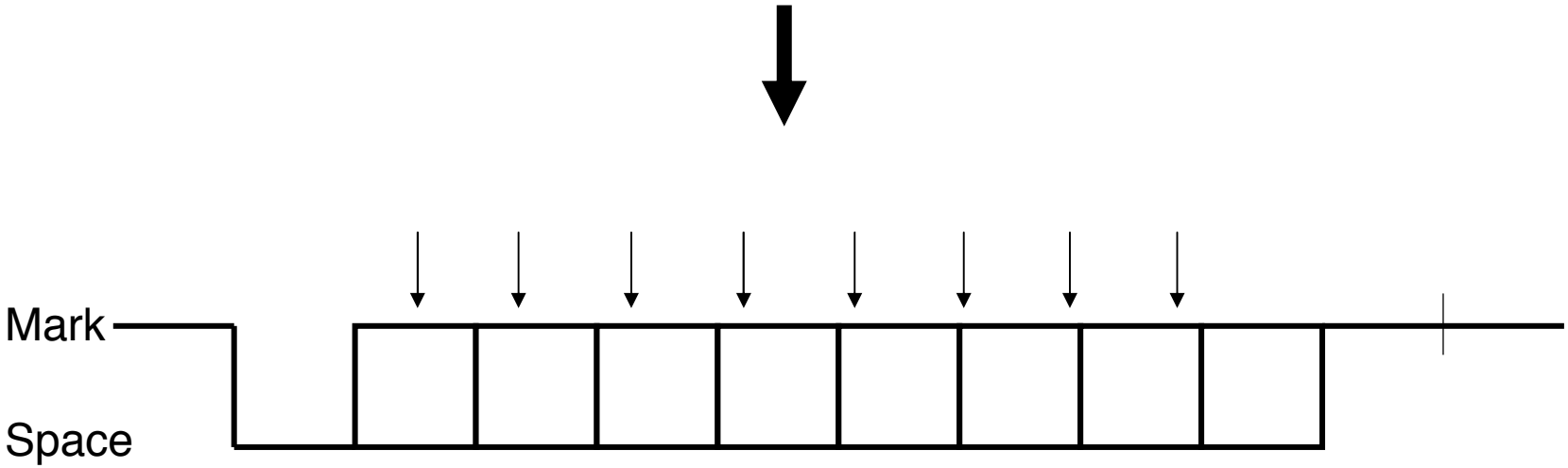


Receiver uses a timer (counter) to time when it samples.
Transmission rate (i.e., bit width) must be known!

Slides from BYU CS 224

UART Character Reception

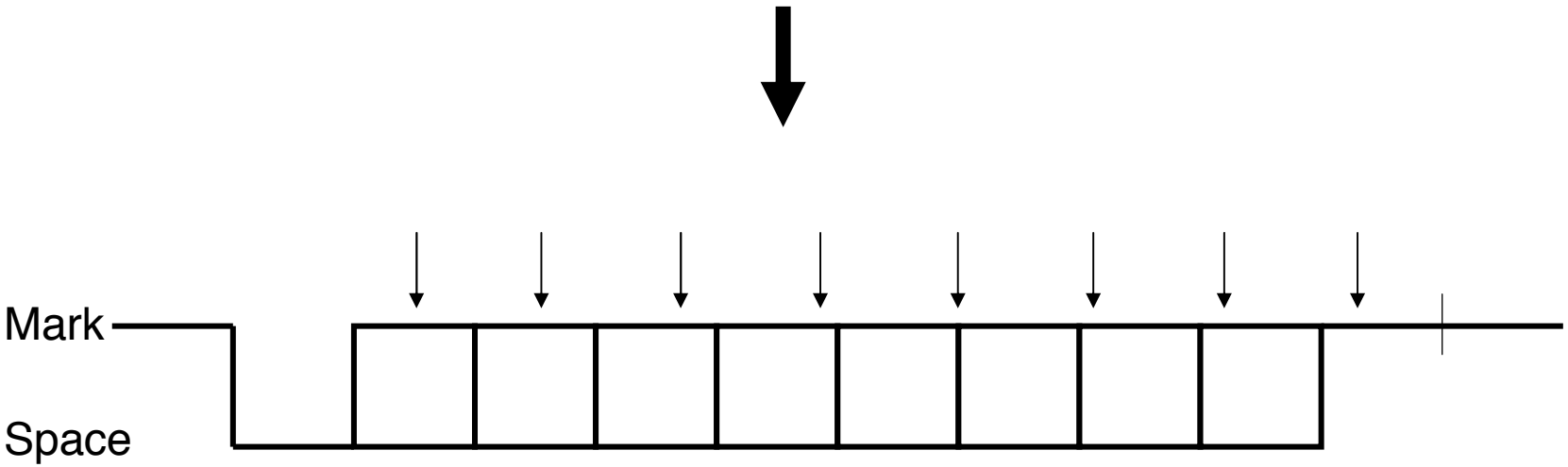
If receiver samples too quickly, see what happens…



Slides from BYU CS 224

UART Character Reception

If receiver samples too slowly, see what happens…



Receiver resynchronizes on every start bit.
Only has to be accurate enough to read 9 bits.

Slides from BYU CS 224

UART Character Reception

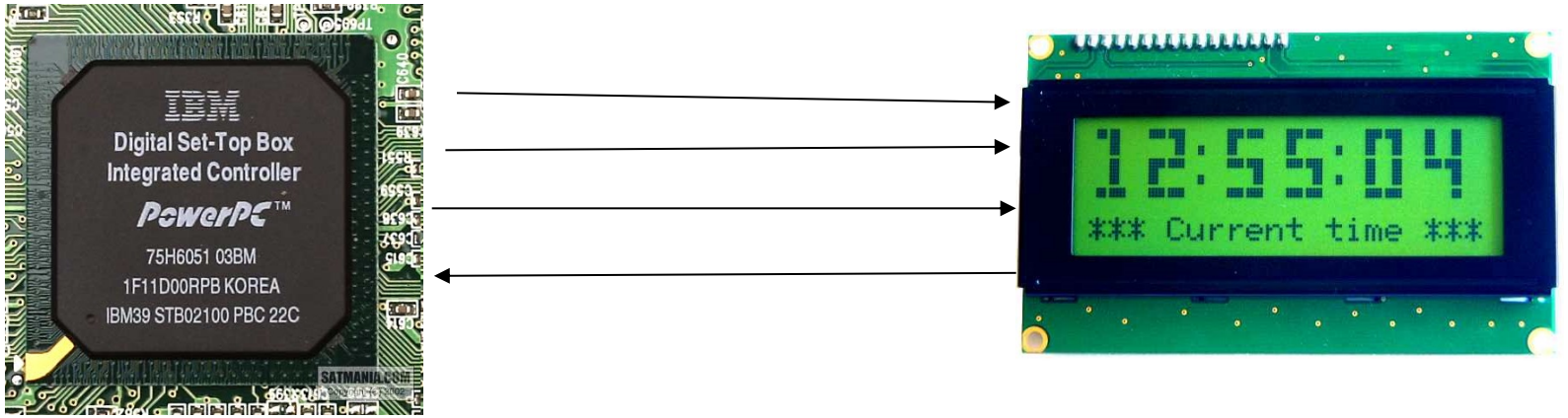
- Receiver also verifies that stop bit is ‘1’
  - If not, reports “framing error” to host system
- New start bit can appear immediately after stop bit
  - Receiver will resynchronize on each start bit

Serial Peripheral Interconnect (SPI)

- Another kind of serial protocol in embedded systems (proposed by Motorola)
- Four-wire protocol
  - SCLK — Serial Clock
  - MOSI/SIMO — Master Output, Slave Input
  - MISO/SOMI — Master Input, Slave Output
  - SS — Slave Select
- Single master device and with one or more slave devices
- Higher throughput than I2C and can do “stream transfers”
- No arbitration required
- But
  - Requires more pins
  - Has no hardware flow control
  - No slave acknowledgment (master could be talking to thin air and not even know it)

What is SPI?

- Serial Peripheral Interface (SPI) protocol [1979]
- Fast (Mbps), easy to use (few wires), simple
- Nearly all microcontrollers support it



SPI Basics

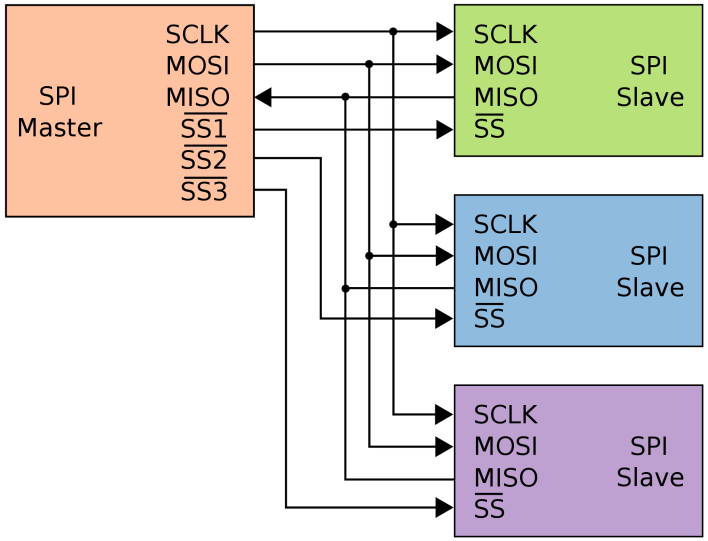
- Uses 4 wires (compared to UART's 2-wires)
  - Also known as a "4 wire" bus
- Used to communicate across short distances
- Multiple Secondaries, Single Primary
- Synchronized

SPI Capabilities

- Always Full Duplex
  - Communicating in two directions at the same time
  - Transmission need not be meaningful
- Multiple Mbps transmission speed
- Transfers data in 4 to 32 bit characters
- Multiple slaves
  - Daisy-chaining possible

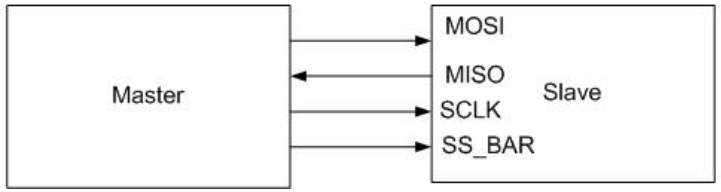
SPI Protocol

- Wires:
  - Master Out Slave In (MOSI)
  - Master In Slave Out (MISO)
  - System Clock (SCLK)
  - Slave Select 1…N
- Master Set Slave Select low
- Master Generates Clock
- Shift registers shift in and out data

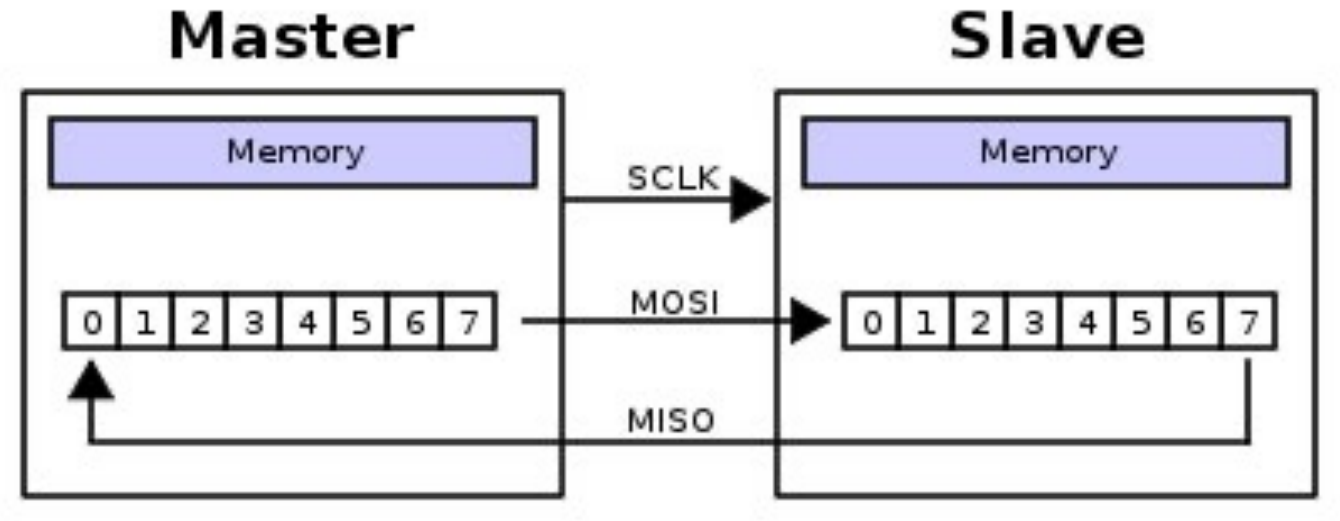


SPI Wires in Detail

- MOSI – Carries data out of Primary to Secondary
- MISO – Carries data from Secondary to Primary
  - Both signals happen for every transmission
- SS_BAR – Unique line to select a secondary
- SCLK – Primary-produced clock to synchronize data transfer



SPI is the quintessential “shift register” communication bus



Primary shifts out data to Secondary, and shifts in data from Secondary

http://upload.wikimedia.org/wikipedia/commons/thumb/b/bb/SPI_8-bit_circular_transfer.svg/400px-SPI_8-bit_circular_transfer.svg.png