Week 2

How UARTs Work

Parallel vs. Serial Communication

- Parallel communication: Fast, uses many I/O pins (e.g., 32 bits = 32 wires).
- Serial communication: Slower, but uses fewer wires (typically 2: signal + ground).
- Serial transfers use shift registers to transmit one bit at a time.

Common Serial Standards

- Examples: I2C, SPI, Ethernet, Bluetooth (wireless), RS-232.
- All rely on shift registers but differ in protocol and wiring.

RS-232 and UART

- UART (Universal Asynchronous Receiver Transmitter) implements RS-232.
- RS-232 dates back to 1960 and requires only 2 wires: signal and ground.
- UART is asynchronous (no clock line); saves wiring at the cost of timing complexity.

Synchronizing Without a Clock

- UART begins transmission with a falling edge (start bit).
- Receiver must know baud rate (transmission speed) to time incoming bits.
- Baud rate mismatches result in garbled data.

UART Data Frame Format

- Start bit: 1 bit (low).
- Data bits: usually 8 bits.
- Optional parity bit: for error checking.
- Stop bits: 1 or 2 bits (high).
- Total frame length: typically 10 or 11 bits per byte.

Parity Bit

- Optional but must be agreed on by both devices.
- Can use even or odd parity.
- Helps detect single-bit errors; cannot detect all multi-bit errors.

Stop Bits

- Required: at least 1, up to 2 allowed.

- Mismatched stop bits between transmitter and receiver can cause data errors.

UART Errors

- Overrun error: Occurs when CPU writes data faster than UART can transmit.
- Always check if UART is ready before writing new data.

Basic Sensor Interface

Sensor Power (Excitation)

- Some sensors (e.g., thermocouples) self-generate voltage.
- Others need external power—must ensure the power supply is clean (filtered).

Voltage Range and Protection

- Microcontrollers (like PSoC) operate from 0 to VDD (typically 5V).
- Use external components to clamp inputs to safe levels.
- Diodes protect pins but have current limits (~100 μA).

Capacitance and Signal Filtering

- GPIOs have input capacitance (~5–20 pF).
- Combined with series resistance, they form an RC low-pass filter.
- At high frequencies (≥100 kHz), signals may be attenuated.

Signal Amplification

- Use a PGA (Programmable Gain Amplifier) to boost small sensor signals.
- Example: 0.2 V signal × gain of 25 = 5 V output.
- Only certain gain values are available.

High-Pass Filtering

- Series capacitor blocks DC offset → forms a high-pass filter with resistance.
- Cutoff frequency depends on R and C values.

Analog Filtering Before ADC

- Prevents aliasing (Nyquist rule: sample ≥ 2× highest signal frequency).
- Reduce wideband noise.
- Rule of thumb: Signal bandwidth ≤ 1/10 of sample rate.

Thermistor Lab, Part 1

Common Temperature Sensors

- Thermistors: Cheap, sensitive, nonlinear, limited to ~150°C.
- RTDs: Platinum-based, linear, expensive, wide range (–50°C to 500°C).
- Thermocouples: Two dissimilar metals, self-powered, wide range.
- Semiconductor sensors (diodes): Easy to integrate, limited temperature range.

Thermistor Characteristics

- Resistance drops with increasing temperature.
- Curve is nonlinear (Negative Temperature Coefficient).

Temperature Calculation

- Use the Steinhart-Hart equation:
  $1/T = a + b \cdot \ln(R) + c \cdot (\ln(R))^3$
- Output in Kelvin. Convert to Celsius by subtracting 273.
- Requires floating-point math support.

Thermistor Lab, Part 2

ADC Considerations

- ADC reads voltage, not resistance.
- Use a voltage divider to convert thermistor resistance to voltage.

Positive-Only Input

- ADC input range is ±1.024 V.
- With GND on negative input, ADC only sees positive voltages → half resolution lost.

Improving Accuracy

- Don’t use power supply as a voltage reference—it may fluctuate.
- Use precision resistors (e.g., 0.1% tolerance) for accurate current measurement.
- Measure voltage across known resistor → calculate current.
- Measure voltage across thermistor → use V = IR to compute resistance.
- Then use resistance to determine temperature (via Steinhart-Hart equation or lookup table).

Which of the following files are stored in the Workplace Explorer Module of the Cypress PSoC Creator 4 workspace? All of the files related to your project

How do you keep track of external components on the schematic? Use the Off Chip tab in the component catalog

What code is stored in the PSoC file main.c when you first start your project? No executable code because you haven't created the project yet

Which of the following is not an off-chip component, available for selection from the Cypress Component Catalog? Motor controller

Which of the following is not an on-board component available for selection from the Cypress Component Catalog? 16 gigabyte memory

Which items are configurable for a digital pin in the PSoC system? Interrupt, Threshold, Sync Mode

How would you compare the resolution of the PSoC Delta Sigma ADC to that of the PSoC SAR ADC? The Delta Sigma ADC resolution can be configured between 8 and 20 bits in intervals of 1, while that of the SAR ADC can be configured to 8, 10, or 12 bits.

How do you connect a pin on the schematic to a physical pin on the PSoC development kit? Pin mapping is done in the pins tab (selected from Workspace Explorer). You click on the 'Port' and 'Pin' drop down, and select the appropriate port and pin number. Then the physical pins are connected to the pins in the schematic.

Which one of these items are commonly used LCD API in the PSoC system (assume the "instance name" has been configured to be just "LCD" on the schematic)? Use Cydelay () or other methods to make sure you don't write to the LCD too fast for it to respond, Write a string of characters (contained in double quotes) to the LCD using LCD_PrintString (), Insert the function LCD_Start () into main.c, Set where the cursor is on the LCD using the function LCD_Position (0,0)

What type of software protocol is implemented in a UART? RS-232, RS-485