Lab-7 Group -4 report

(210020050,210020007)

Objective

1. Program your micro-controller to transmit the 8-bit value "0xF0" if SW1 is pressed and "0xAA" if SW2 is pressed over UART with baud rate 9600 and odd parity. Read the relevant sections of the datasheet and board manual.
2. Sketch the expected waveforms for both cases with indicative timings.

3. Connect the Scope to the TX pin and verify that the captured signals match what you have drawn in part2.

4. Add code to your program to also listen for incoming data on the UART with the same baud and parity config. If "0xAA" is received, turn LED should light up GREEN. If "0xF0" is received, the LED should be BLUE and if any error is detected LED should be RED. Test this by communicating with your neighboring group. Remember to connect RX of one board to TX of the other, and make sure to connect the board grounds together.

Part 1: Sending Data on UART

- The goal : to transmit specific 8-bit data when user input is received via push buttons. The system transmits:
  - The 8-bit value 0xF0 when Switch 1 (SW1) is pressed.
  - The 8-bit value 0xAA when Switch 2 (SW2) is pressed.
- The UART communication is set up with the following specifications:
  - Baud Rate: 9600
  - Data Format: 8 bits
  - Parity: Odd
  - Stop Bits: 1
- This project also involved configuring the microcontroller’s UART peripheral, handling input from the switches, and transmitting the appropriate data over the UART interface.
- The Universal Asynchronous Receiver-Transmitter (UART) on the Tiva C microcontroller was configured to send and receive serial data. UART1 was used for transmission, and its corresponding GPIO pins, PB0 (TX) and PB1 (RX), were configured for alternate functions.
- The switches SW1 and SW2 were connected to GPIO pins PF4 and PF0, respectively. Both switches were configured as inputs with pull-up resistors to ensure a stable high state when not pressed. A simple polling mechanism was used to check the state of the switches.
  - **SW1** (PF4) transmits the value 0xF0.
  - **SW2** (PF0) transmits the value 0xAA.

Part 2: Sketch waveforms

The data transmission was initiated whenever either SW1 or SW2 was pressed. The flow of the process is as follows:

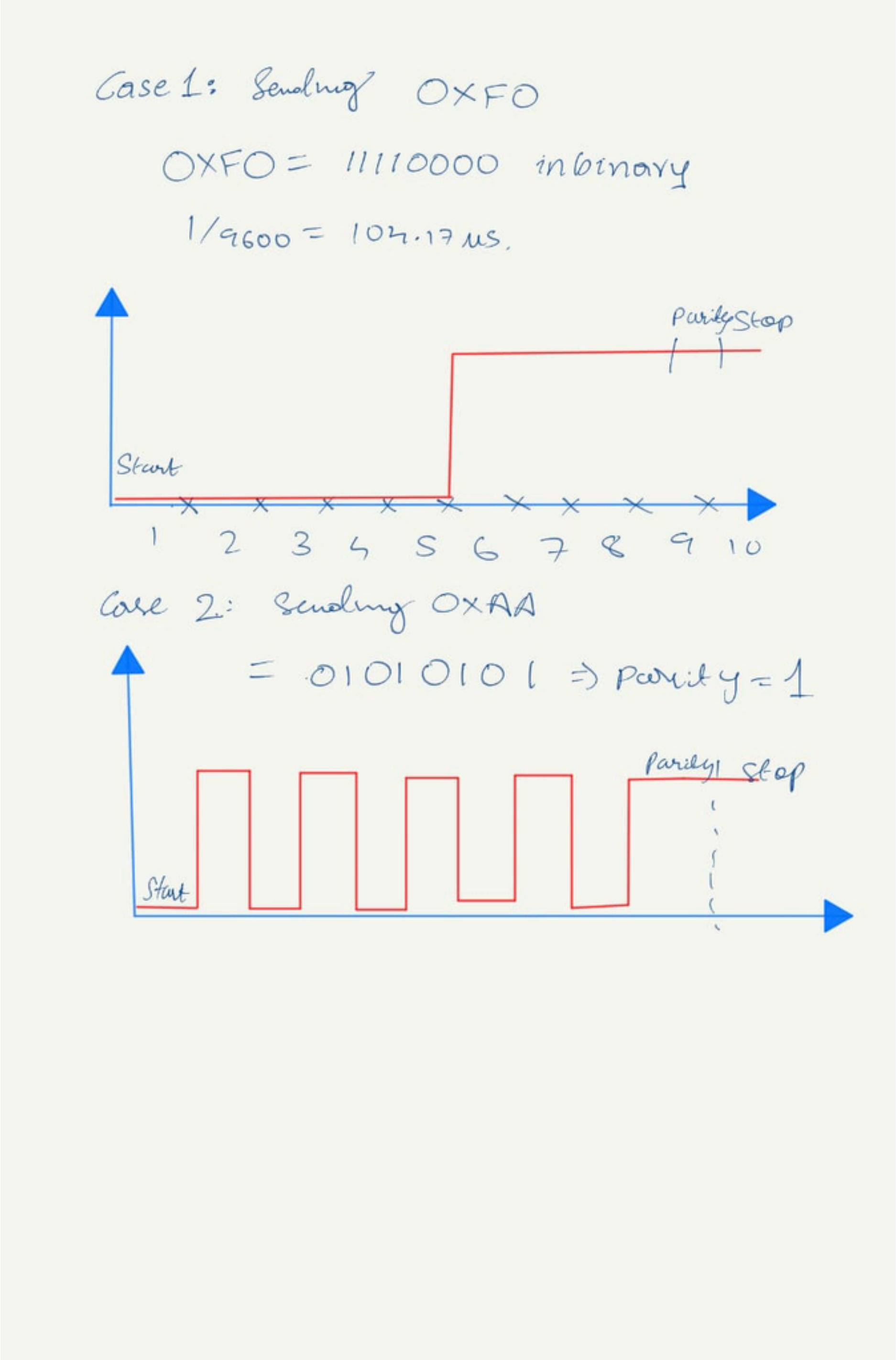
1. **Start Bit**: UART communication begins with a start bit (0).
2. **Data Bits**: The microcontroller sends 8 data bits, least significant bit (LSB) first.
3. **Parity Bit**: Odd parity was used to ensure that the total number of 1s (including the parity bit) is odd.
4. **Stop Bit**: The transmission ends with a stop bit (1), signaling the end of the frame.

=> Case 1: Transmission of 0xF0 (Binary: 11110000)

- **Data Frame**: Start bit (0), data bits 00001111, odd parity (1), and stop bit (1).
- **Bit Stream**: 0 00001111 1 1

=> Case 2: Transmission of 0xAA (Binary: 10101010)

- **Data Frame**: Start bit (0), data bits 01010101, odd parity (1), and stop bit (1).
- **Bit Stream**: 0 01010101 1 1



Case 1: Sending OXFO

OXFO = 11110000 in binary

1/9600 = 104.17 µs.

Case 2: Sending OXAA

= 01010101 ⇒ parity = 1

Part 4: receiving Data

The UART1 peripheral was initialized with the following settings:

- Baud Rate: 9600 (derived using integer and fractional baud rate divisors).
- Frame Format: 8-bit data, odd parity, and 1 stop bit.
- Transmission/Reception: The UART1_Send() and UART1_Receive() functions were implemented to handle UART data exchange.

Error Handling

UART errors such as parity errors or framing errors are detected using the RSR (Receive Status Register). If any error is detected during reception, the Red LED is lit to signal an issue in communication.

LED Control Logic

The control of LEDs is based on the received data or error detection:

- BLUE LED: Turned on when 0xF0 is received.
- GREEN LED: Turned on when 0xAA is received.
- RED LED: Turned on when any transmission error is detected.

Testing Procedure

1. Wiring Setup: The TX pin of one microcontroller (PB1) was connected to the RX pin of another (PB0), and their grounds were connected together.
2. Data Transmission: Button presses on one microcontroller (SW1 for 0xF0 and SW2 for 0xAA) were used to send data to the other microcontroller.
3. Data Reception and LED Indication: The receiving microcontroller processed the incoming data and turned on the corresponding LED (Blue for 0xF0, Green for 0xAA, and Red for any errors).