

The following code answers the quiz question

```
#include "uart_transmitter.h"

typedef enum{idle, transmit} uart_transmit_states_type;

void uart_transmitter(bool &uart_tx, ap_uint<8> data, bool baud_rate_signal, bool
start) {
#pragma HLS INTERFACE ap_none port=data
#pragma HLS INTERFACE ap_none port=baud_rate_signal
#pragma HLS INTERFACE ap_none port=uart_tx
#pragma HLS INTERFACE ap_none port=start
#pragma HLS INTERFACE ap_ctrl_none port=return

    ap_uint<10> d = ((bool)0b1, (ap_int<8>)data, (bool)0b0);

    static unsigned int          bit_counter = 0;
    static uart_transmit_states_type state = idle;

    uart_transmit_states_type next_state;
    unsigned int              next_bit_counter;
    bool uart_tx_local;

    switch(state) {
    case idle:
        if ( start == 1) {
            next_state = transmit;
            uart_tx_local = 1;
            next_bit_counter = 0;
        } else {
            next_state = idle;
            uart_tx_local = 1;
            next_bit_counter = 0;
        }
        break;
    case transmit:
        if (baud_rate_signal == 1) {
            if (bit_counter == 10) {
                next_state = idle;
                uart_tx_local = 1;
                next_bit_counter = 0;
            } else {
                next_state = transmit;
                uart_tx_local = d[bit_counter];
                next_bit_counter = bit_counter+1;
            }
        }
        else {
            if (bit_counter == 0) {
```

2 Digital System Design with High-Level Synthesis for FPGA

Sequential Circuits

Project: UART HLS: Quiz Solution

www.highlevel-synthesis.com

```
        uart_tx_local    = 1;
    } else {
        uart_tx_local    = d[bit_counter-1];
    }
    next_state          = transmit;
    next_bit_counter    = bit_counter;
}

break;

default:
    break;
}

state                = next_state;
bit_counter          = next_bit_counter;
uart_tx              = uart_tx_local;
}
```