

Designing UART

S.Smirnov

Introduction

The task of a UART is to convert parallel (8 bit bytes) from memory to a coherent serial stream of bits at the transmitter and do the reverse at the receiver. Add to that the ability for each of a communicating pair of UARTs to both transmit and receive simultaneously and you have a modern full-duplex link. "coherent serial stream of bits" means adding some control bits to maintain the ability of the receiver to discern the beginning of a symbol and its end, and to add the optional ability to detect single bit errors in a symbol (character, byte, etc.)

Some early telegraph schemes used variable-length pulses and rotating clockwork mechanisms to transmit alphabetic characters. The first UART-like devices (with fixed-length pulses) were rotating mechanical switches (commutators). These sent 5-bit Baudot codes for mechanical teletypewriters, and replaced Morse code. Later, ASCII required a seven bit code. When IBM built computers in the early 1960s with 8-bit characters, it became customary to store the ASCII code in 8 bits.

Frame structure

Frame consists of a Start bit, 8 data bits, and a Stop bit. Other variants are also possible in RS-232 applications - the data packet could be 5, 6, or 7 bits long, there could be 2 Stop bits, and a Parity bit could be inserted between the data packet and the Stop bit for elementary error detection. Figure 1 shows typical data frame structure.

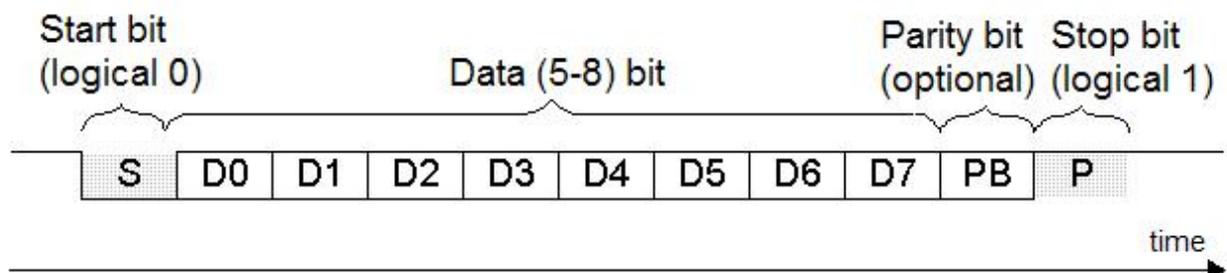


Figure1. UART frame structure

UART structure

A UART usually contains the following components:

- a clock generator, usually a multiple of the bit rate to allow sampling in the middle of a bit period.
- input and output shift registers
- transmit/receive control
- read/write control logic
- First-in, first-out (FIFO) buffer memory (optional)

UART structure is shown at figure 2.

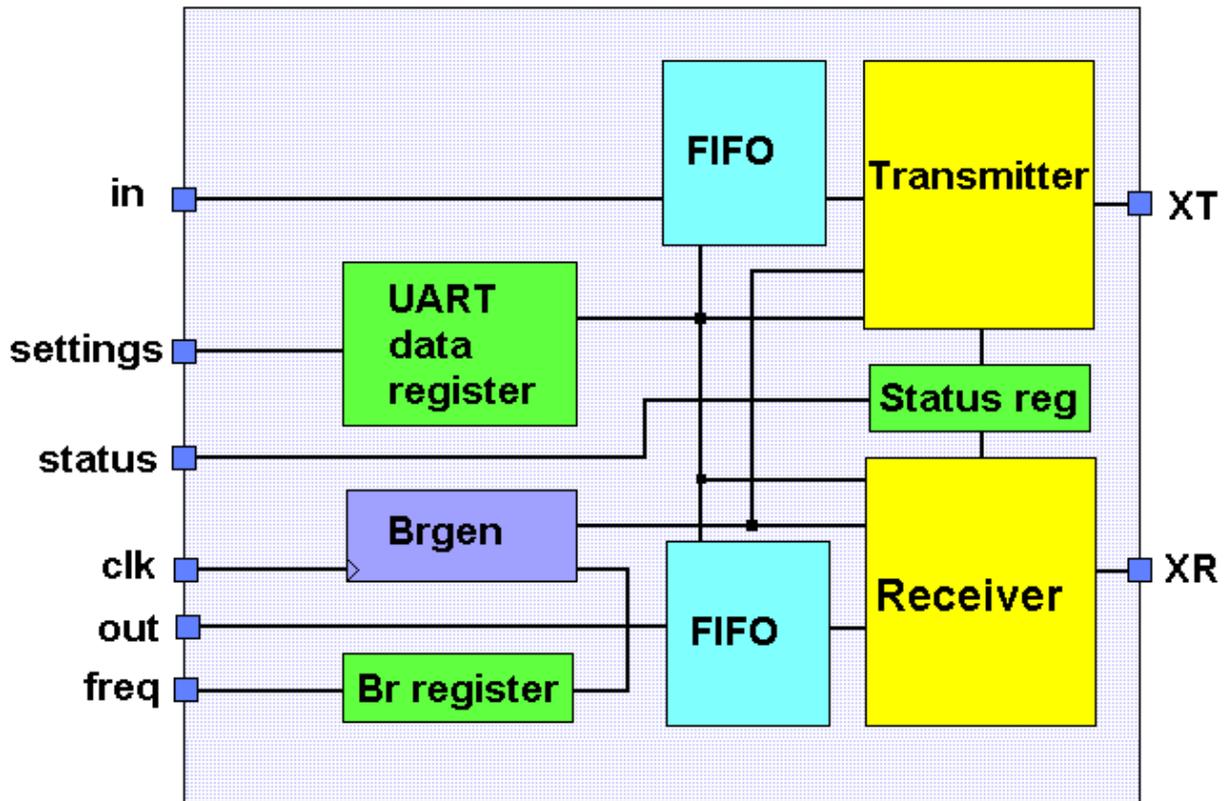


Figure2. UART structure

Baud rate generator

All operations of the UART hardware are controlled by a clock signal which runs at a multiple (say, 16) of the data rate - each data bit is as long as 16 clock pulses.

Typical serial ports used with personal computers connected to modems use eight data bits, no parity, and one stop bit; for this configuration the number of ASCII character per seconds equals the bit rate divided by 10.

Receiver

The receiver tests the state of the incoming signal on each clock pulse, looking for the beginning of the start bit. If the apparent start bit lasts at least one-half of the bit time, it is valid and signals the start of a new character. If not, the spurious pulse is ignored. After waiting a further bit time, the state of the line is again sampled and the resulting level clocked into a shift register. After the required number of bit periods for the character length (5 to 8 bits, typically) have elapsed, the contents of the shift register is made available (in parallel fashion) to the receiving system. The UART will set a flag indicating new data is available, and may also generate a processor interrupt to request that the host processor transfers the received data. In some common types of UART, a small first-in, first-out (FIFO) buffer memory is inserted between the receiver shift register and the host system interface. This allows the host processor more time to handle an interrupt from the UART and prevents loss of received data at high rates.

Transmitter

Transmission operation is simpler since it is under the control of the transmitting system. As soon as data is deposited in the shift register, the UART hardware generates a start bit, shifts the required number of data bits out to the line, generates and appends the parity bit (if used), and appends the stop bits. Since transmission of a single character may take a long time relative to CPU speeds, the UART will maintain a flag showing busy status so that the host system does not deposit a new character for transmission until the previous one has been completed; this may also be done with an interrupt. Since full-duplex operation requires characters to be sent and received at the same time, practical UARTs use two different shift registers for transmitted characters and received characters.

Transmitting and receiving UARTs must be set for the same bit speed, character length, parity, and stop bits for proper operation. The receiving UART may detect some mismatched settings and set a "framing error" flag bit for the host system; in exceptional cases the receiving UART will produce an erratic stream of mutilated characters and transfer them to the host system.