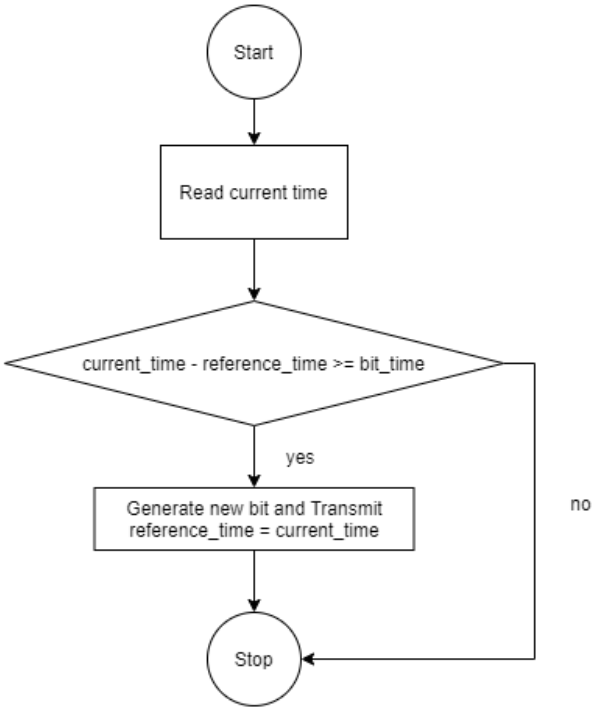
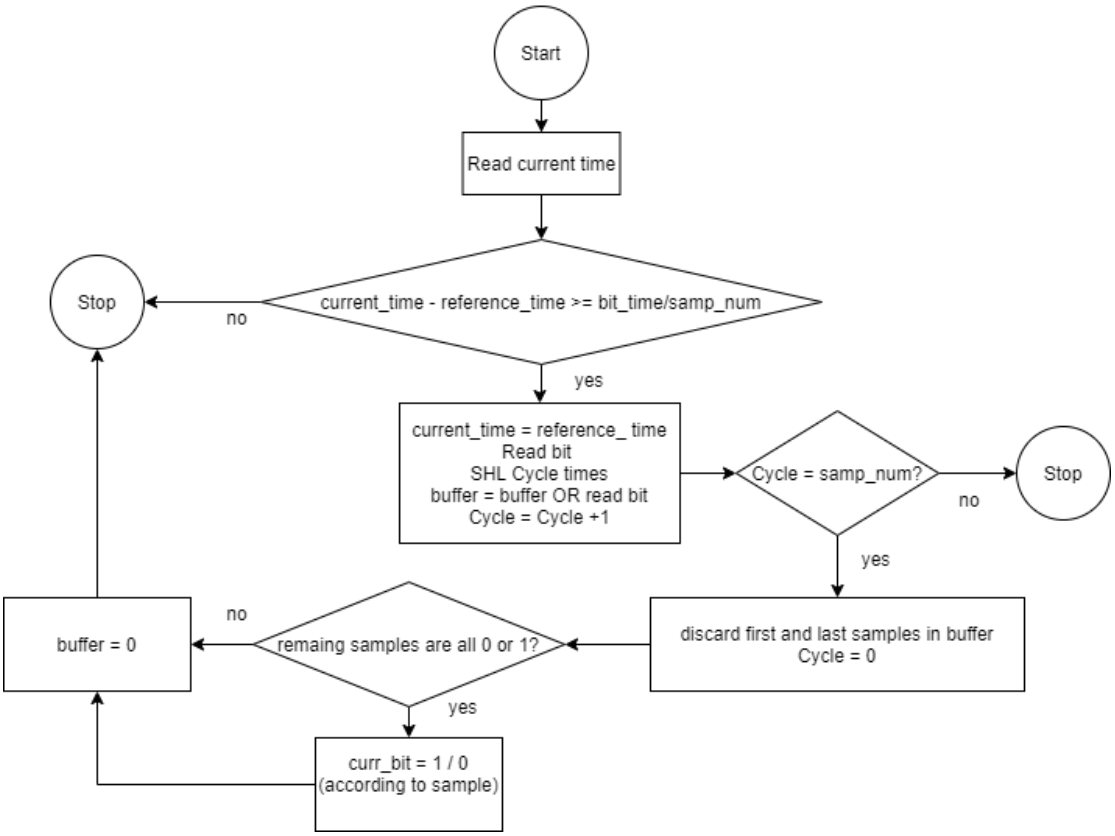
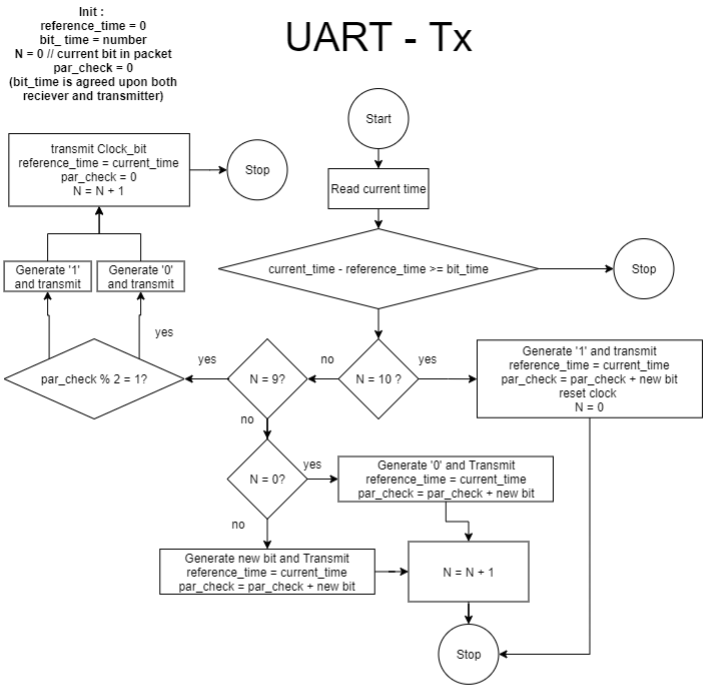
NRZ - Tx


Start
Read current time
current_time - reference_time >= bit_time
yes
no
Generate new bit and Transmit
reference_time = current_time
Stop

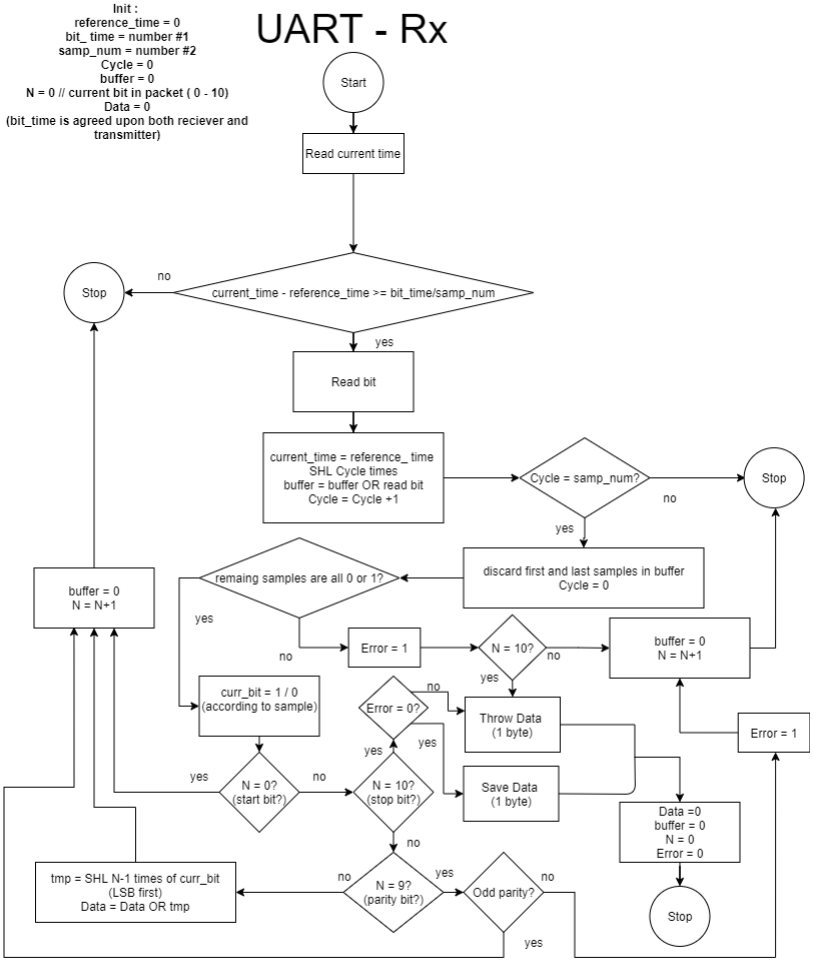
NRZ - Rx


Start
Read current time
current_time - reference_time >= bit_time/samp_num
Stop
no
yes
current_time = reference_ time
Read bit
SHL Cycle times
buffer = buffer OR read bit
Cycle = Cycle +1
Cycle = samp_num?
no
Stop
yes
discard first and last samples in buffer
Cycle = 0
remaing samples are all 0 or 1?
no
buffer = 0
yes
curr_bit = 1 / 0
(according to sample)

UART - Tx



UART - Rx

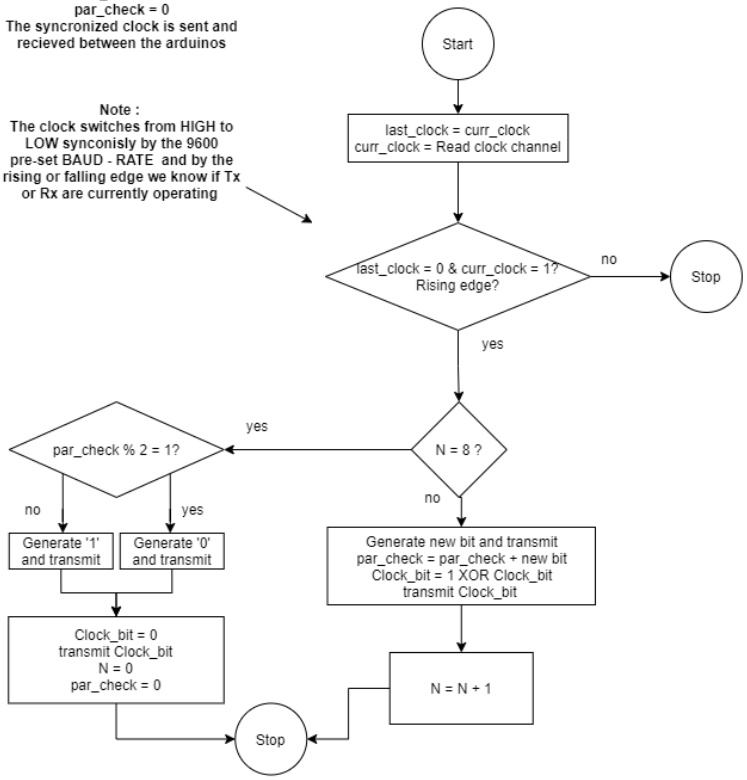


*Note : we do not check if the stop bit is '1', it does not affect the 8 data bits and we assume that after every packet the transmitter sends a '1' signal by deafult (for the next falling edge - start)

USART - Tx

Init :
bit_ time = number
N = 0 // current bit in packet
curr_clock = 1
last_clock = 1
par_check = 0
The syncronized clock is sent and recieved between the arduinos

Note :
The clock switches from HIGH to LOW synconisly by the 9600 pre-set BAUD - RATE and by the rising or falling edge we know if Tx or Rx are currently operating



USART - Rx

Init :
bit_ time = number #1
samp_num = number #2
curr_clock = 1
last_clock = 1
Error = 0
N = 0 // current bit in packet (0 - 8)
Data = 0

Note :
The clock switches from HIGH to LOW synconisly by the 9600 pre-set BAUD - RATE and by the rising or falling edge we know if Tx or Rx are currently operating