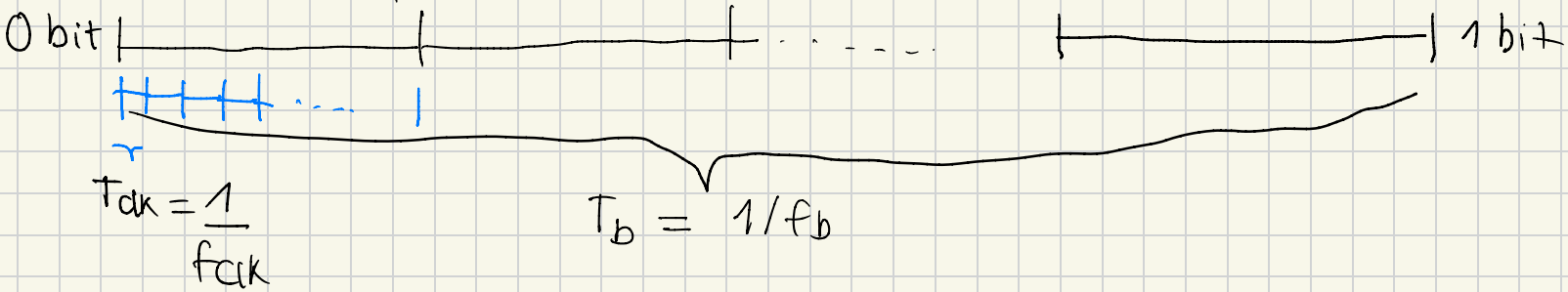
UART

Baud rate generator

Timing diagram

- $f_{clk} \gg f_{baud}$
- The sender, receiver should agree on the same f_{baud}

$T_{tick} = 1/16 f_b$

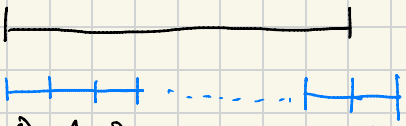


* Usually, we want 16 ticks in a bit (baud)

⇒ Count for $\frac{T_{tick}}{T_{clk}}$ clock cycle ⇒ 1 Tick

Design

. Baud rate generator :



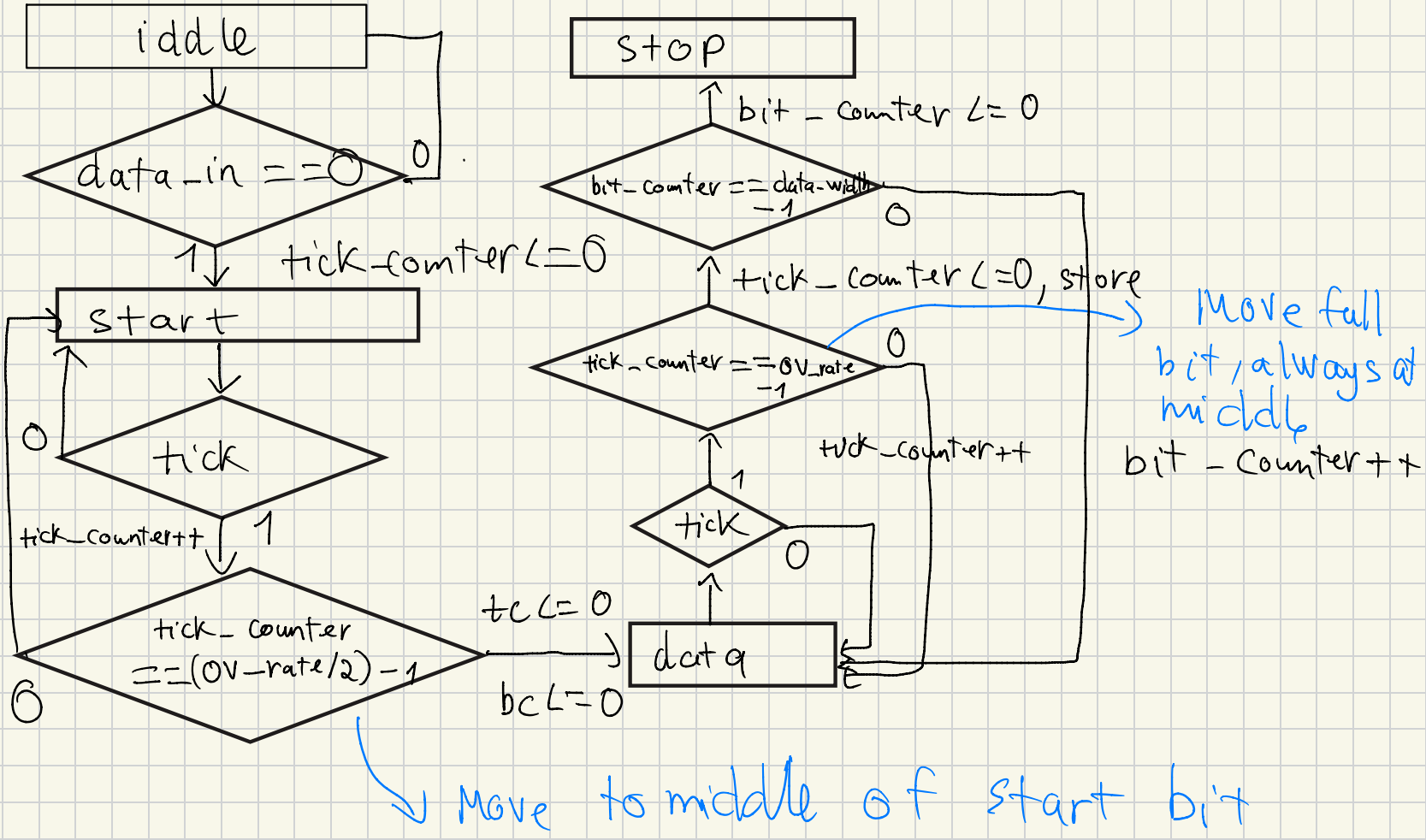
assert tick here

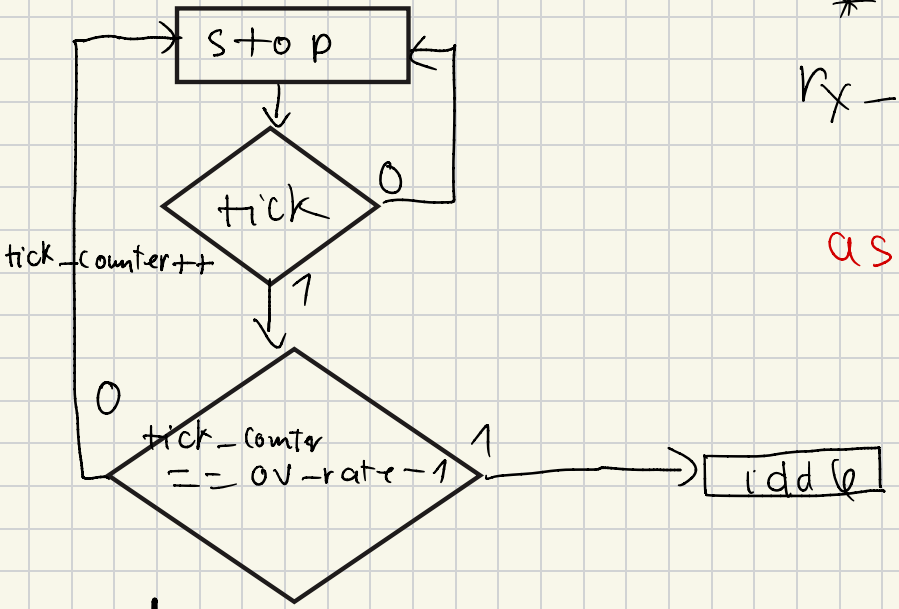
let $N = \frac{T_{tick}}{T_{clk}}$

⇒ A module that count up to N-1 then assert tick at next clk

Receiver

ASMD chart


iddle
data_in == 0
0
1
tick_counter <= 0
start
0
tick
tick_counter++
1
tick_counter == (OV_rate/2) - 1
0
tc <= 0
bc <= 0
data
stop
bit_counter <= 0
bit_counter == data_width - 1
0
tick_counter <= 0, store
tick_counter == OV_rate - 1
0
tick_counter++
1
tick
0
Move full bit, always @ middle
bit_counter++
Move to middle of start bit



* store:

rx_out[bit_counter] <= rx_in

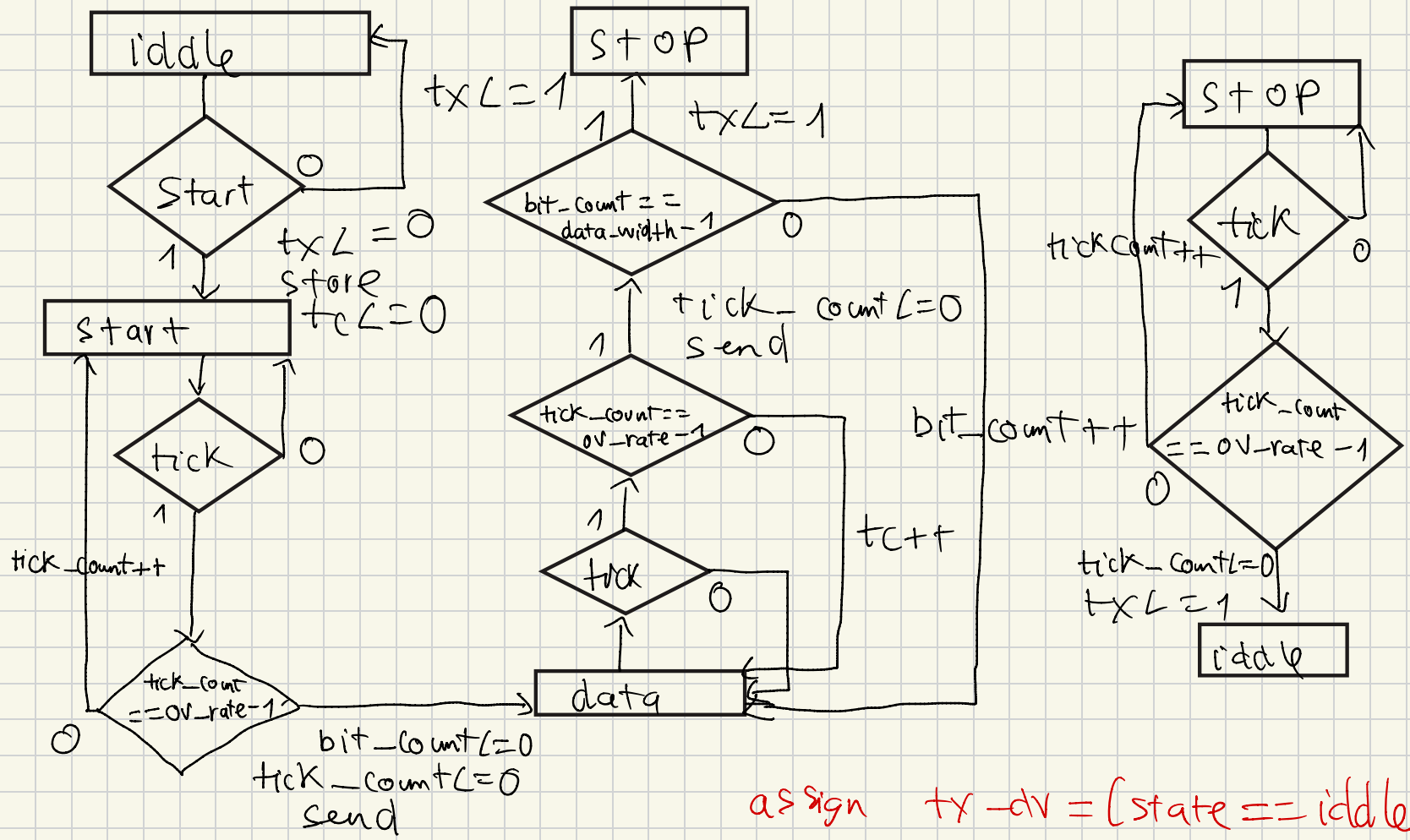
assign rx_dv = (state == idle)

Code

. Github

Transmitter

ASMD


iddle
Start
0
1
txL=0
store
tcL=0
start
tick
0
1
tick_count++
tick_count
==ov_rate-1
0
bit_countL=0
tick_countL=0
send
data
tick
0
1
tick_count==
ov_rate-1
0
1
tick_
countL=0
send
bit_count==
data_width-1
0
1
txL=1
stop
txL=1
bit_count++
tc++
stop
tick
0
1
tickcount++
tick_count
==ov_rate-1
0
tick_countL=0
txL=1
iddle
assign tx_dv = (state == iddle)