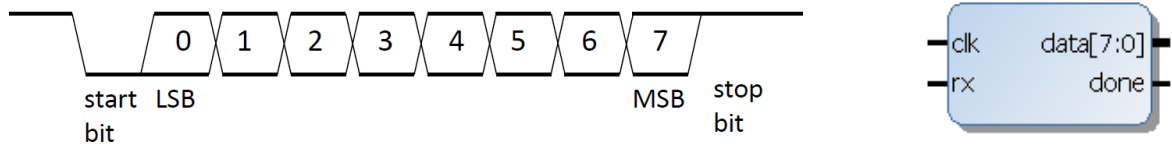


5. vaja: Asinhroni serijski vmesnik

Naredili bomo sprejemni del asinhronega serijskega vmesnika. Serijski vhodni signal ima mirovno stanje logično 1, prenos se začne s startnim bitom (logična 0), ki mu sledi osem podatkovnih bitov, od najnižjega (LSB) proti najvišjemu (MSB).



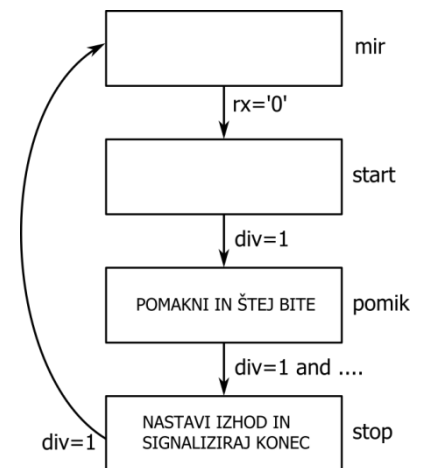
5.1 Opis vezja v jeziku VHDL

Asinhroni sprejemnik bomo naredili v obliki sekvenčnega stroja s štirimi stanji:

- **mir**, mirovno stanje
- **start**, sprejem startnega bita (logična 0)
- **pomik**, pomikanje podatkovnih bitov
- **stop**, zaključek paketa (logična 1)

V opisu deklariramo nov podatkovni tip in notranji signal za stanja:

```
architecture opis of sprejem is
  type stanja is (mir, start, pomik, stop);
  signal st: stanja;
  signal div: unsigned(1 downto 0):="00";
```



Asinhroni sprejemnik temelji na prevzorčenju – uporabili bomo uro s štirikrat višjo frekvenco, kot je frekvenca prenašanja podatkov. V vezju bomo imeli 2-bitni števec, ki bo skrbel za deljenje ure. Sinhronizacijo sprejemnika z vhodnimi podatki naredimo tako, da se števec postavi na 0, ko smo v mirovnem stanju in je vhodni signal rx enak '0'.

```
p1: process(clk)
begin
  if rising_edge(clk) then
    if st=mir and rx='0' then div <= "00";
    else div <= div + 1;
```

Prehajanje stanj opišemo v sinhronem procesu s stavkom **case**, kjer za vsako stanje določimo naslednje stanje. Npr.

```
case st is
  when mir =>
    if rx='0' then st <= start;
```