

vaja 04

7-segmentni prikazovalnik

Digitalno načrtovanje – laboratorijske vaje
asistent: Nejc Ilc

Osnove (1)

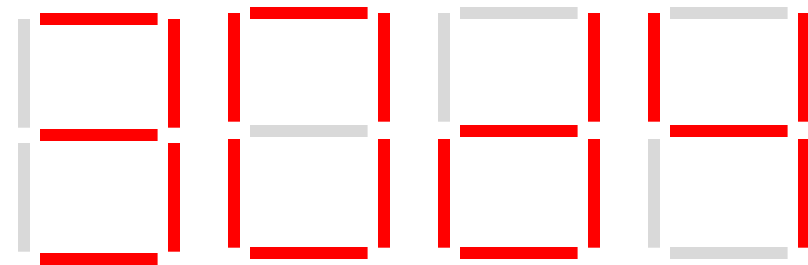
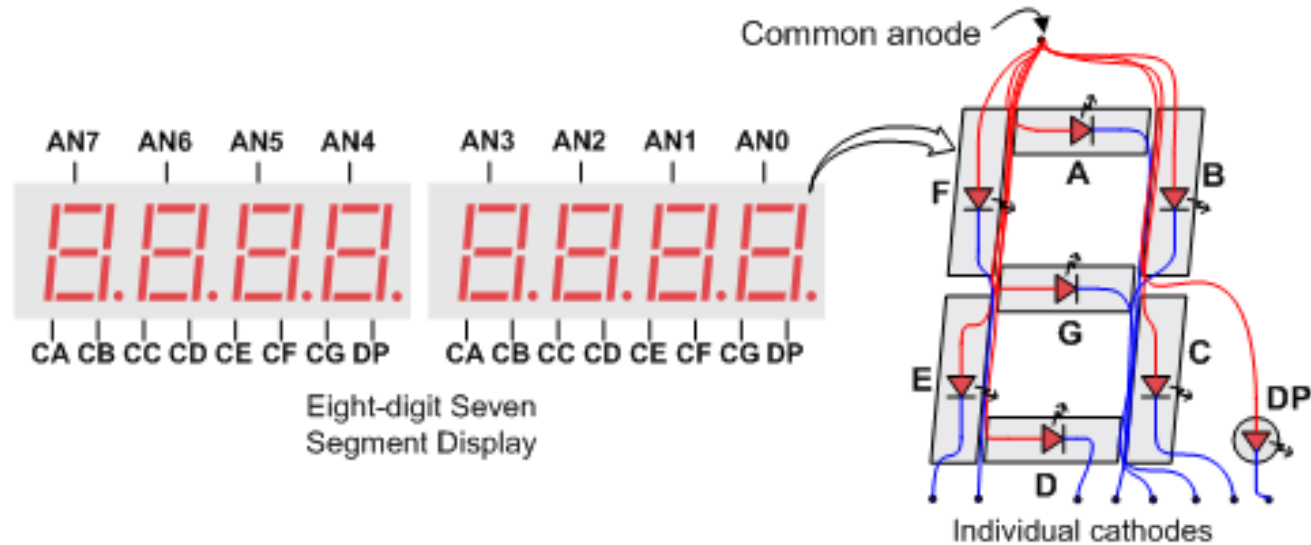
- Razvojna plošča vsebuje 7-segmentni prikazovalnik z osmimi mesti oziroma števki, če prikazujemo števila.



0 1 2 ... A B C d e F

Osnove (2)

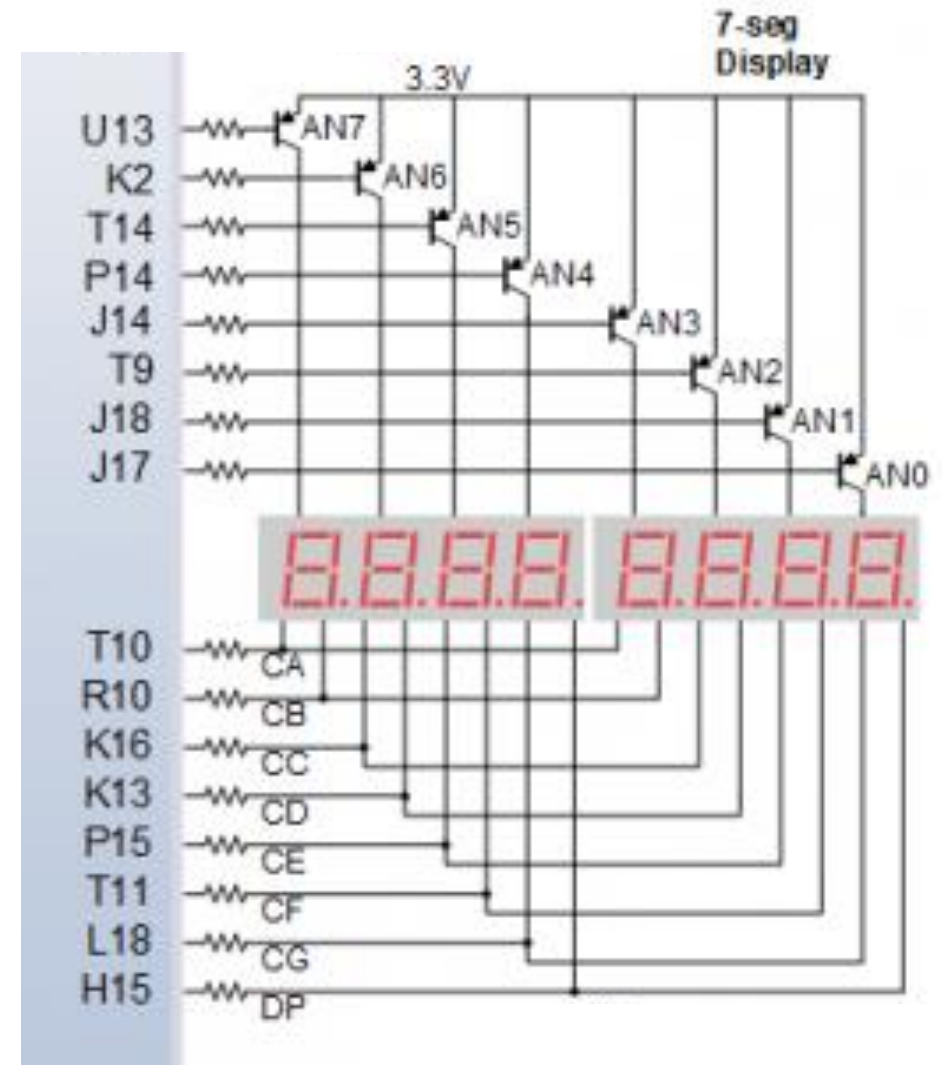
- Vsako števk krmilimo z 8 signali, ki predstavljajo katode na diodi LED. Anoda je skupna za vse segmente ene števk.
- Vsaka od osmih števk ima svojo anodo (AN0-AN7).
- 8 katod je skupnih za vse števk (CA ... CG + DP).
- Obratna logika:
 - 0: prižgan segment
 - 1: ugasnjen segment



prikaz števila 0x30d4 = 12500

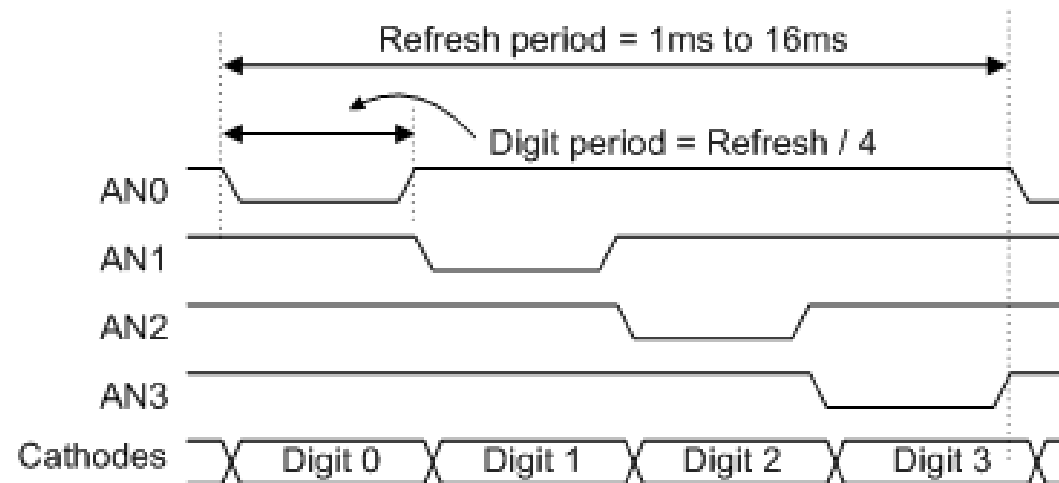
Osnove (3)

- Uporabniku prijazno bi bilo, da bi za vsako od osmih števk imeli svojo množico osmih signalov za prižiganje segmentov. To je žal preveč potratno.
- Rešitev: časovno multipleksiranje
 - 8 signalov na števkko - katode
 - 8 signalov za izbiro števkke - anode
 - Skupaj 16 signalov (namesto 64)



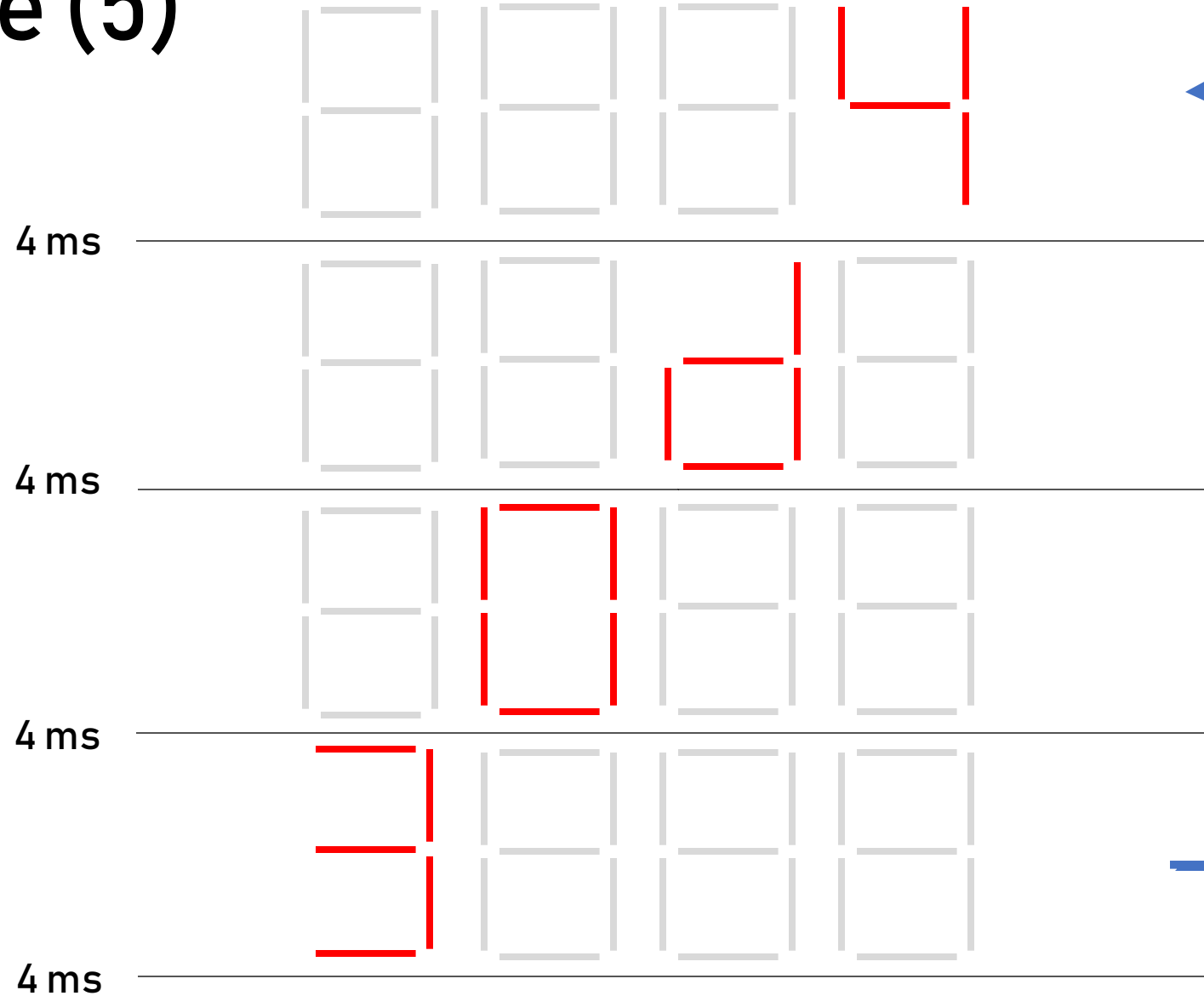
Osnove (4)

- Časovno multipleksiranje: periodično osvežujemo vsako od števk. Če to delamo dovolj hitro, človeško oko ne opazi utripanja.
- Pri frekvenci osveževanja ~ 60 Hz (dolžina periode ~ 16 ms):
 - 8 števk: 2 ms/števko
 - 4 števke: 4 ms/števko
- Števko izbiramo z aktivacijo ustrezne anode (AN0-AN7).



Primer za 4 števke

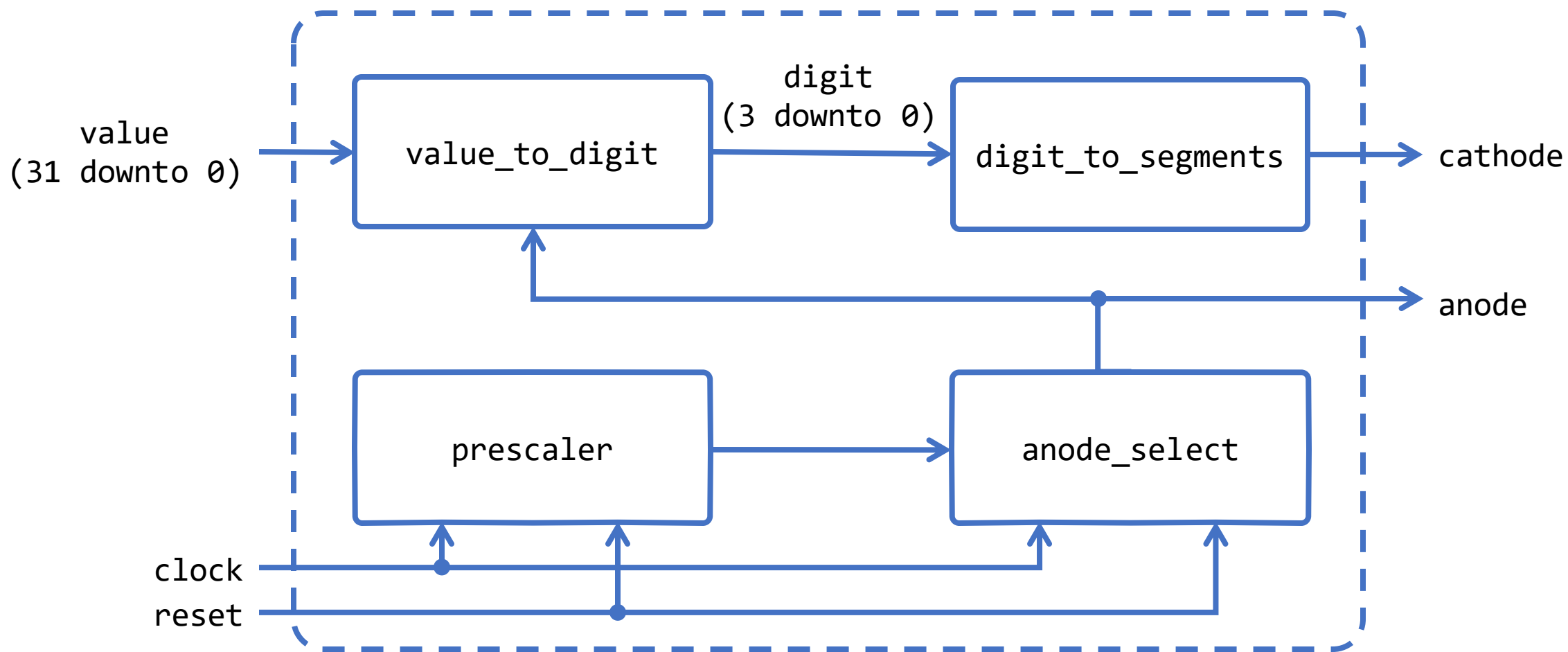
Osnove (5)



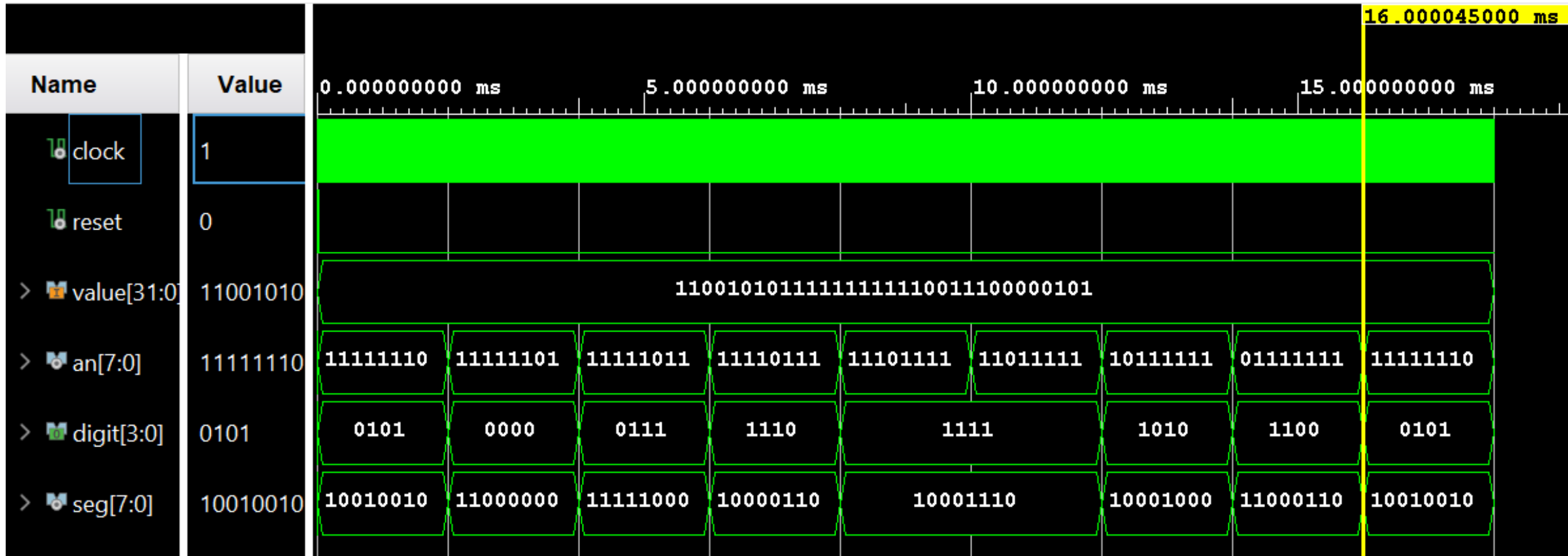
Modul `seven_seg_display`: komponente

- `prescaler`
 - delilnik ure, ciljna frekvenca 500 Hz (zakaj?)
- `anode_select`
 - periodična aktivacija anode posamezne števk
 - `1111110` → `11111101` → `11111011` → ... → `01111111`
- `value_to_digit`
 - glede na aktivirano anodo izmed 32 bitov izbere ustrezne 4 bite za prikaz ene števk
- `digit_to_segments`
 - pretvori 4-bitno število v segmente (aktivacija katod)

Modul seven_seg_display: diagram



Simulacija



Izziv

Na prejšnjih vajah smo izdelali števec, ki povečuje oziroma zmanjšuje svojo vrednost vsako sekundo. Sedaj prikažite vrednost števca na sedem-segmentnem prikazovalniku.

- števec naj ima 32 bitov
- glede na stikala naj števec šteje navzgor, navzdol ali pa stoji
- uporabite vseh 8 števk prikazovalnika