

## 7.-8. vaja: Generator slike

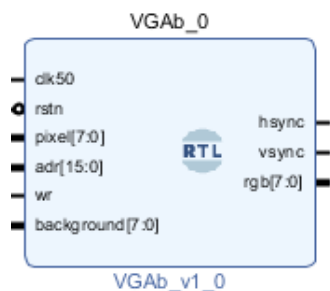
Izdelali bomo grafični krmilnik, ki generira signale za prikaz slike na monitorju s priključkom VGA.

### 7.1 VGA krmilnik

Standard VGA je nastal še v času katodnih monitorjev. Priključek vsebuje tri analogne barvne signale in dva sinhronizacijska signala: **hsync** za prehod v novo vrstico in **vsync** za novo sliko. Krmilnik vsebuje dva števec in primerjalnike za določanje sinhronizacijskih impulzov. Modul števecv bomo nastavili za ločljivost [SVGA 800 x 600](#) točk in osveževanje 72 Hz pri vhodni uri frekvence 50 MHz.

1. Naredi nov Vivado projekt za izbrano ploščo Zedboard [VGAzed.xdc](#) ali Mini Zed [VGAmini.xdc](#). Vključi v projekt vzorčni krmilnik [vga.vhd](#) in datoteko XDC z nastavitvami priključkov (Add Sources, Constraints).
2. Preglej vzorčni krmilnik in preizkusi delovanje sinhronizacijskih signalov na simulaciji, kjer nastavi uro (8 ns) in dovolj dolg čas simulacije (700.000 ciklov ure).

### 7.2 Krmilnik s pomnilnikom



Vezja FPGA vsebuje bloke statičnega pomnilnika (BRAM), ki omogočajo sočasno pisanje in branje podatkov na različnih naslovih (angl. Dual Port RAM). V krmilnik VGA bomo dodali pomnilnik velikosti 64 kB in logiko za branje oz. pisanje v pomnilnik.

3. V jeziku VHDL deklariraj nov podatkovni tip in signal:

```
type ram_type is array (0 to 65535) of std_logic_vector(7 downto 0);
signal ram : ram_type := (others => "00000000");
```

4. Deklariraj še signal **data** kot 8-bitni standardni vektor, **xy** pa kot 16-bitni nepredznačen vektor za naslov prebrane slikovne točke. Če postavimo okno velikosti 256 x 256 v levi zgornji kot slike, dobimo naslov kar iz obeh števcov:  $xy \leftarrow vst(7 \text{ downto } 0) \ \& \ hst(7 \text{ downto } 0)$ ;

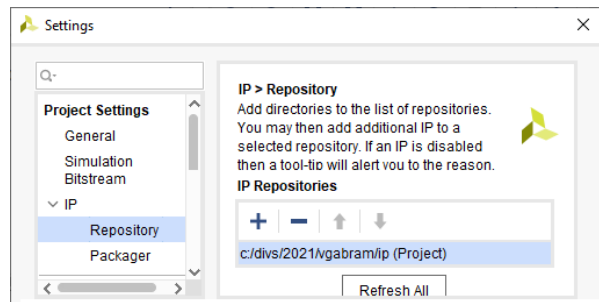
Sintetizator vezja bo uporabil BRAM, če bo model pomnilnika izvajal pisanje in branje sinhrono z uro.

Opiši naprej branje:  $data \leftarrow ram(to\_integer(xy))$ ;

5. Določi 8-bitni barvni izhod **rgb**, ki naj bo enak:
  - vrednosti **data**, ko so števeci v kvadratnem območju  $hst < 256$  in  $vst < 256$ ,
  - vhodu **background** v ostalem delu vidne slike (800 x 600) oziroma
  - x"00" izven vidnega dela slike.
6. Dodaj še priključke in logiko za vpisovanje posameznih točk. Med priključki vezja je že vhod za barvo (**pixel**), dodaj pa še 16-bitni naslov pomnilnika (**adr**) in enobitni krmilni signal za pisanje (**wr**). Ob uri in aktivnem krmilnem signalu (**wr=1**) naj se prenese vrednost vhoda **pixel** v pomnilnik na naslovu **adr**.

## 8.1 Blokovni diagram z vmesnikom axi\_set4

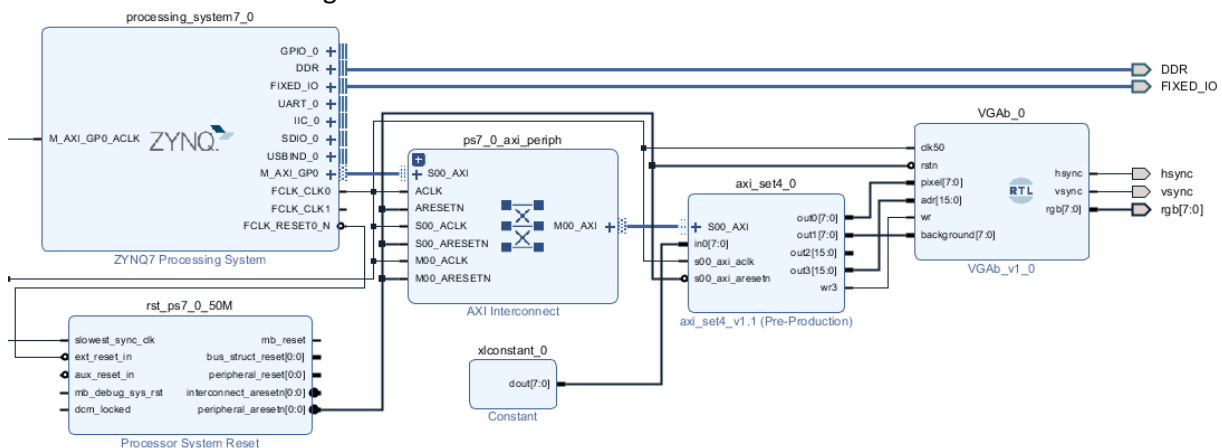
Blokovni diagram naj vsebuje VGA krmilnik in vmesnik AXI Lite s štirimi registri. Datoteke vmesnika [axi\\_set4.zip](#) odpakiraj v lokalno mapo znotraj projektne mape, nato pa nastavi repozitorij za iskanje komponent IP: *Project Settings*, odpri *IP*, *Repository* in s klikom na + dodaj mapo z opisom komponente.



Pomen registrov komponente axi\_set4:

ime signala	vrsta	pomen	naslov
in0	vhod, 8-bit	branje statusa, v našem primeru vezano kar na konstanto 1	0
out0	izhod, 8-bit	barva točke	0
out1	izhod, 8-bit	barva ozadja	+ 4
out2	izhod, 16-bit	trenutno neuporabljen	+ 8
out3	izhod, 16-bit	koordinata za pisanje, ob zapisu v out3 dobimo impulz wr3	+ 12

1. Nariši blokovni diagram:



2. Komponenti Zynq nastavi frekvenco ure FCLK\_CLK0 na 50 MHz (Clock Configuration, PL Fabric Clocks) in dodaj v opis vezja nastavitve priključkov za ustrezno razvojno ploščo:
3. Blokovnemu diagramu dodaj wrapper ter izvedi sintezo in izdelavo izhodne datoteke (Bitstream).

## 8.2 Preizkusni program

V programski opremi nastavljamo barvo ozadja: `Xil_Out8(XPAR_AXI_SET4_0_S00_AXI_BASEADDR+4, 15);` in rišemo posamezne točke, tako da nastavimo barvo točke in njeno koordinato:

```
Xil_Out8(XPAR_AXI_SET4_0_S00_AXI_BASEADDR, 0xFF); // barva točke (bela = 0xff)
Xil_Out16(XPAR_AXI_SET4_0_S00_AXI_BASEADDR+12, 0x0102); // prikaz točke na koordinati (2,1)
```

Preizkusi delovanje s testnim programom, nato pa še s programom [VGAcrte.c](#), ki vsebuje funkcijo za risanje ravnih črt in nariše v okno 256 naključno izbranih barvnih črt.