

Uvod v Vitis

projekt, prevajanje, razhroščevanje
aplikacija

Andrej Trost

Laboratorij za načrtovanje integriranih vezij



UNIVERZA
V LJUBLJANI

FE

Fakulteta
za elektrotehniko

Vitis: razvoj SW za SoC

perspektiva

platforma
(BSP knjiž.)

aplikacija

lab1.sw - test/src/helloworld.c - Vitis IDE

File Edit Search Vitis Project Window Help

Design Debug

design_1_wrapper

- export
- hw
- logs
- ps7_cortexa9_0
 - standalone_ps7_cortexa9_0
 - bsp
- resources
- platform.spr
- platform.tcl
- test_system [design_1_wrapper]
 - test [standalone_ps7_cortexa9_0]
 - Binaries
 - Includes

```
36 * This application configures UART 16550 to baud rate 9600
37 * PS7 UART (Zyng) is not initialized by this application,
38 * bootrom/bsp configures it to baud rate 115200
39 *
40 * -----
41 * | UART TYPE   BAUD RATE
42 * -----
43 *   uartns550   9600
44 *   uartlite    Configurable only in HW design
45 *   ps7_uart    115200 (configured by bootrom/bsp)
46 */
47
48 #include <stdio.h>
49 #include "platform.h"
50 #include "xil_printf.h"
51
52
53 int main()
54 {
55     init_platform();
```

Assistant

- design_1_wrapper [Platform]
- test_system [System]
 - test [Application]
 - Debug
 - Release
 - Debug
 - Release

Console

uart (CONNECTED)

Hello World

Successfully ran Hello World application

Successfully ran Hello World application

design_1_wrapper

stanje opravila

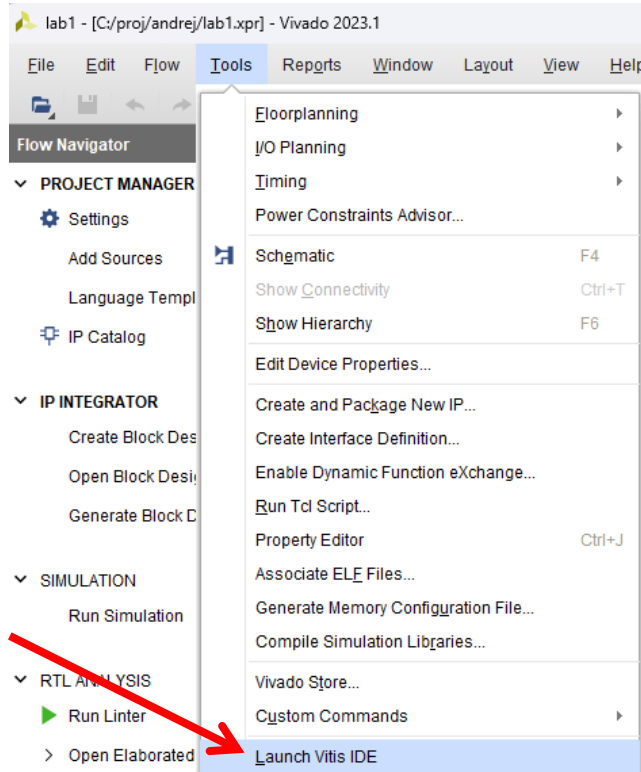
Izvozi opis HW iz orodja Vivado

The screenshot shows the Vivado IDE interface. The 'File' menu is open, and the 'Export' option is highlighted with a red arrow. The 'Export Hardware Platform' dialog box is open, and the 'Include bitstream' option is selected and highlighted by a red arrow. The background shows a project window with a netlist and a device configuration window.

Shrani: design_1_wrapper.xsa

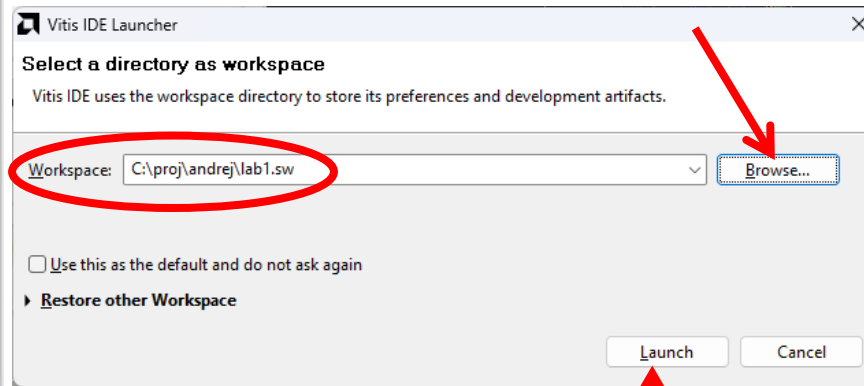
Vitis IDE

1.

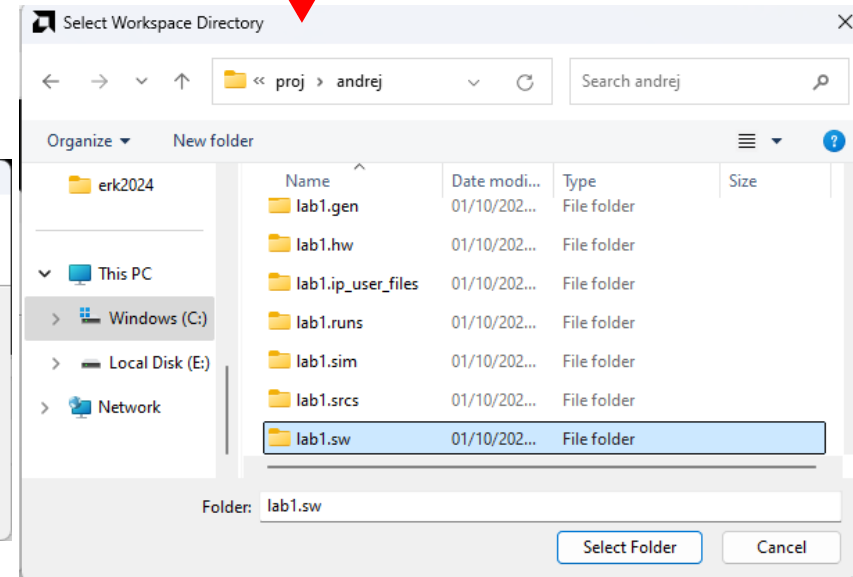


Naredi novo mapo znotraj projekta, npr. lab1.sw

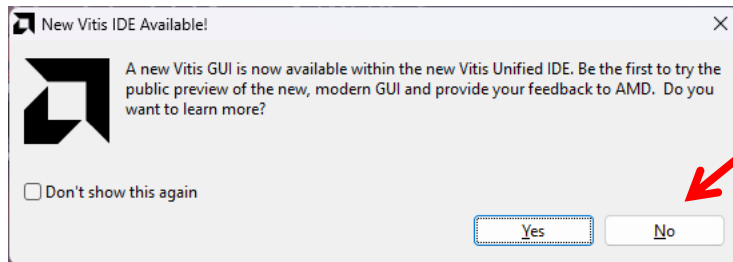
2.



3.

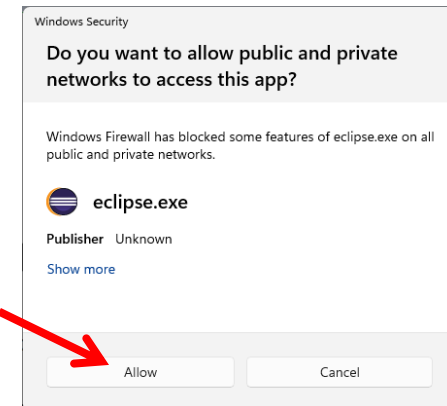


4.

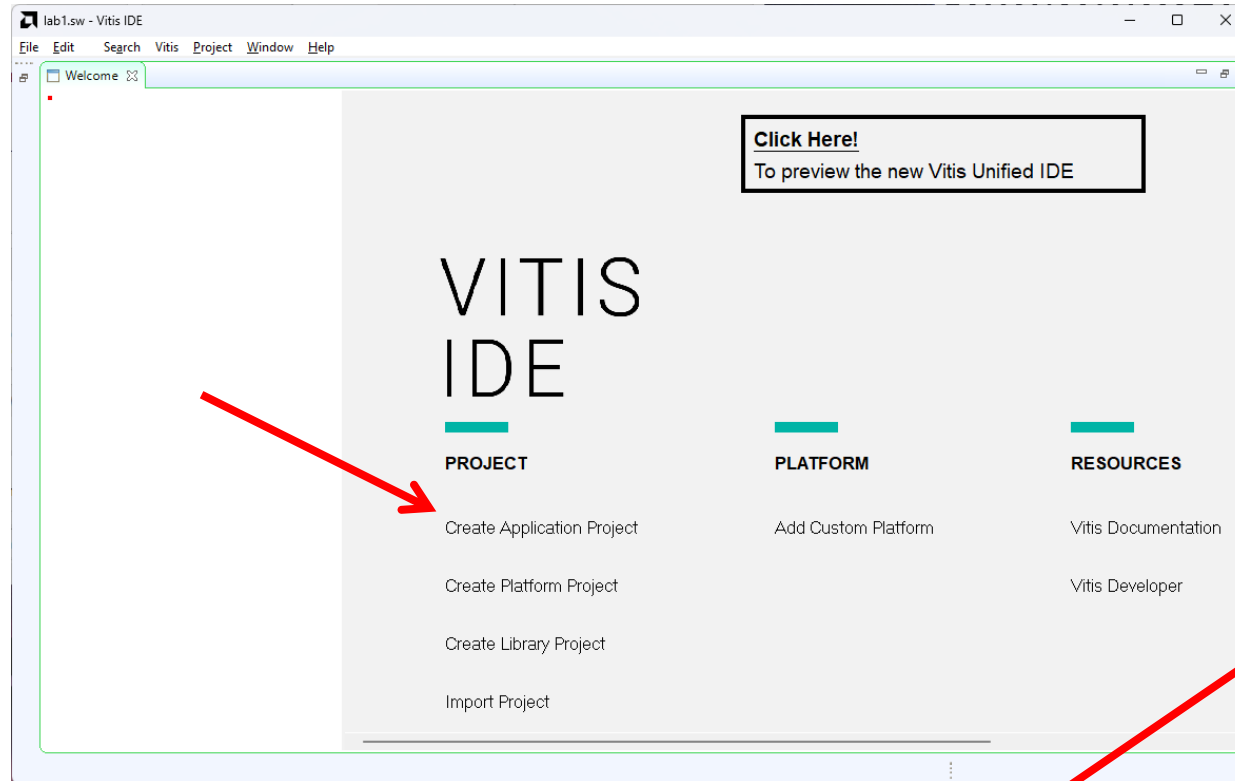


5.

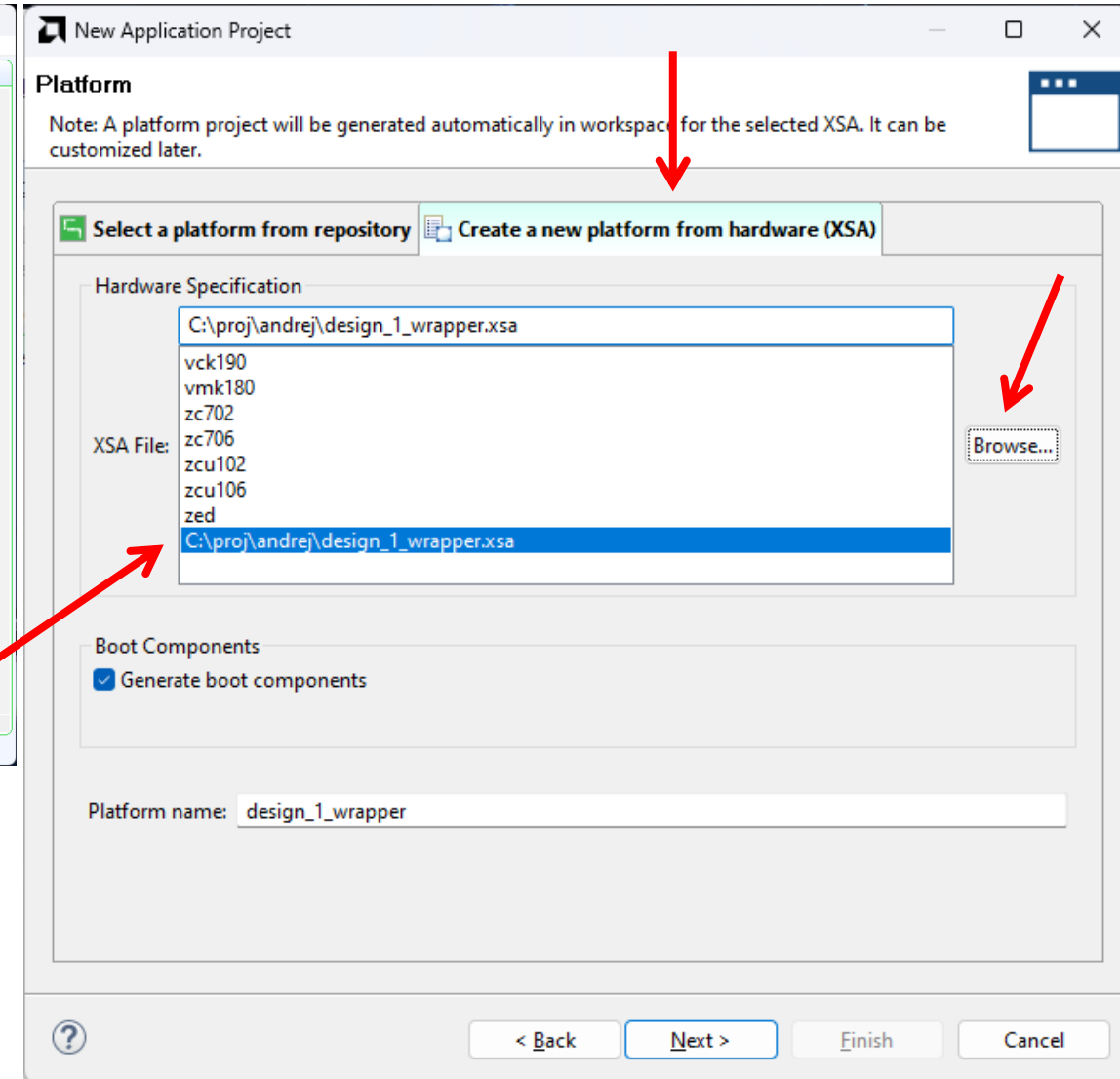
6.



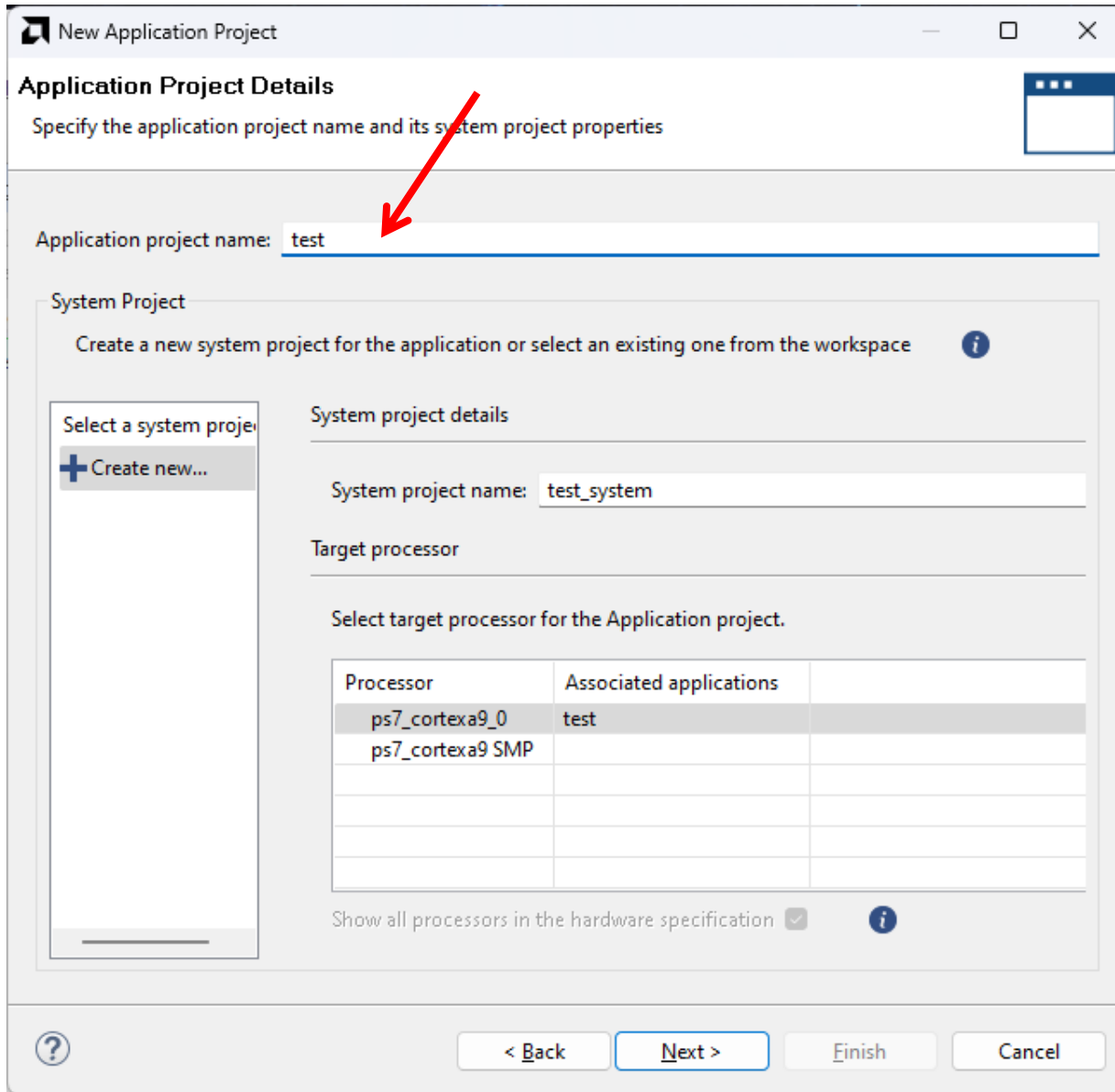
Nov Vitis projekt: 1. korak - izbira platforme



Poiščemo na disku: design_1_wrapper.xsa

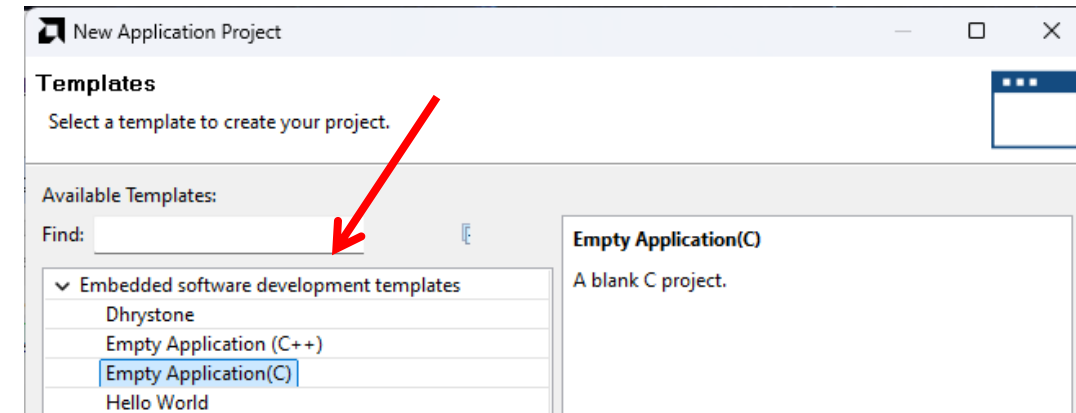


2., 3. in 4. korak izdelave projekta



3. korak, Domain le potrdimo privzeto

4. korak, Templates izberemo Empty Application ali pa Hello World



Pregled nastavitev platforme

The image shows the Vitis IDE interface for a project named 'test'. The 'Application Project Settings' window is open, displaying the following configuration:

- Project name: test
- Platform: design_1_wrapper
- Runtime: cpp
- Domain: standalone_ps7_cortexa
- CPU: ps7_cortexa9_0
- OS: standalone

A red arrow points to the 'Platform' dropdown menu. Another red arrow points to the 'Hardware Specification' link below the settings. The 'Assistant' window shows the project hierarchy, including 'test' and its sub-components like 'Debug' and 'Release'.

The 'Hardware Platform Specification' window is also visible, showing design information and an address map for processor 'ps7_cortexa9[0-1]'. The address map table is as follows:

Cell	Base Address	High Address	Slave Interface	Addr Range	Type
ps7_intc_dist_0	0xf8f01000	0xf8f01fff	-		register
ps7_scutimer_0	0xf8f00600	0xf8f0061f	-		register
ps7_scler_0	0xf8000000	0xf8000fff	-		register
axi_gpio_0	0x41200000	0x4120ffff	S_AXI		register
ps7_scuwdt_0	0xf8f00620	0xf8f006ff	-		register
ps7_l2cachec_0	0xf8f02000	0xf8f02fff	-		register
ps7_scuc_0	0xf8f00000	0xf8f000fc	-		register

A red arrow points to the 'axi_gpio_0' row in the table, which is circled in red. The text 'axi_gpio med perifernimi enotami' is written next to the arrow.

Opcija: nastavitvev BSP (UART1 na MiniZed)

test_system test

Application Project Settings

General

Project name: `test`

Platform: `design_1_wrapper` ...

Runtime: `cpp`

Domain: `standalone_ps7_cortexa9_0`

CPU: `ps7_cortexa9_0`

OS: `standalone`

Options

View current BSP settings, or compiler flags, SW intrusive peripherals, change versions

[Navigate to BSP Settings](#)

[Hardware Specification](#): View processors, memory ranges and peripherals.

test_system test design_1_wrapper

type filter text

- design_1_wrapper
 - ps7_cortexa9_0
 - standalone_ps7_cortexa9_0
 - Board Support Package**

Board Support Package

View current BSP settings, or configure setting assign drivers to peripherals, change versions

[Modify BSP Settings...](#) [Reset BSP Sources](#)

A BSP settings file is generated with the user operation clears any existing modifications do

[Load BSP settings from file](#)

Board Support Package Settings

Board Support Package Settings

Control various settings of your Board Support Package.

Overview

- standalone
- drivers
 - ps7_cortexa9_0

Configuration for OS: `standalone`

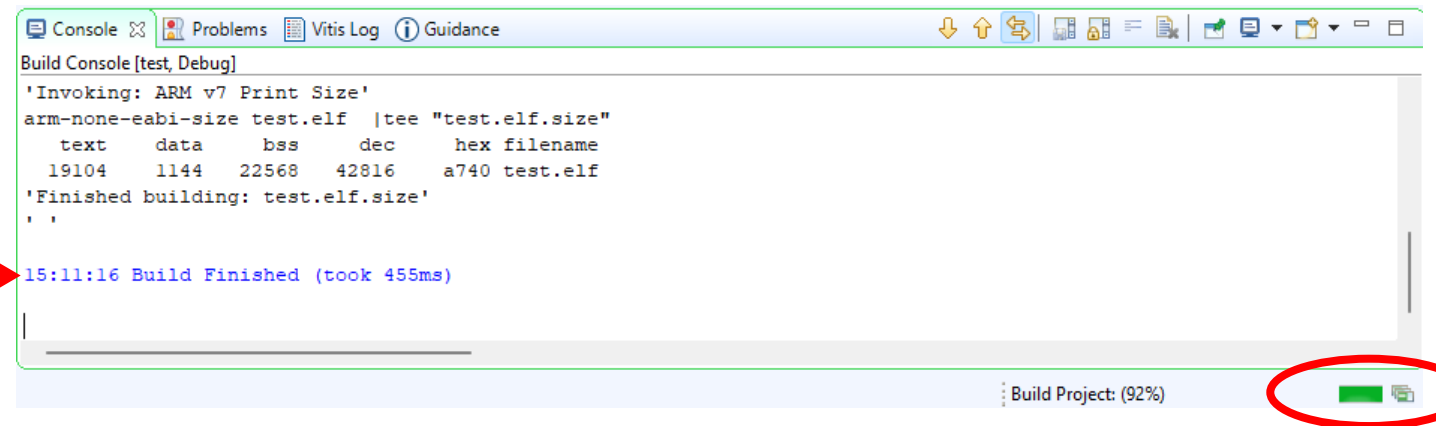
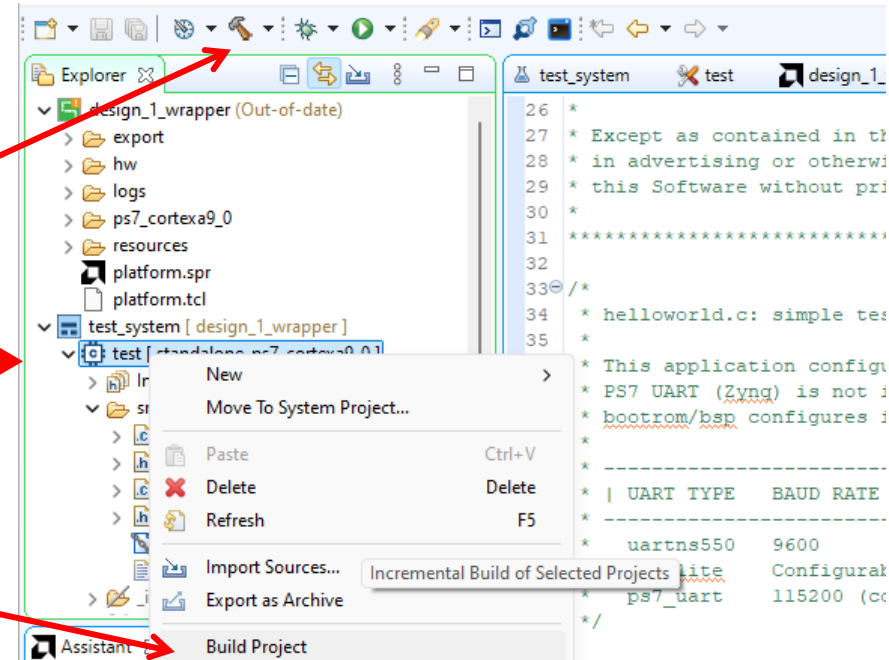
Name	Value	Default
clocking	false	false
enable_minimal_xlat_tbl	true	true
hypervisor_guest	false	false
lockstep_mode_debug	false	false
pmu_sleep_timer	false	false
sleep_timer	none	none
stdin	ps7_uart_1	none
stdout	ps7_uart_1	none
ttc_select_cntr	2	2

Določi uart_1 za stdin/stdout

Prevajanje aplikacije

Okno Explorer

- izberi aplikacijo
- Build Project
- preveri rezultat v konzoli



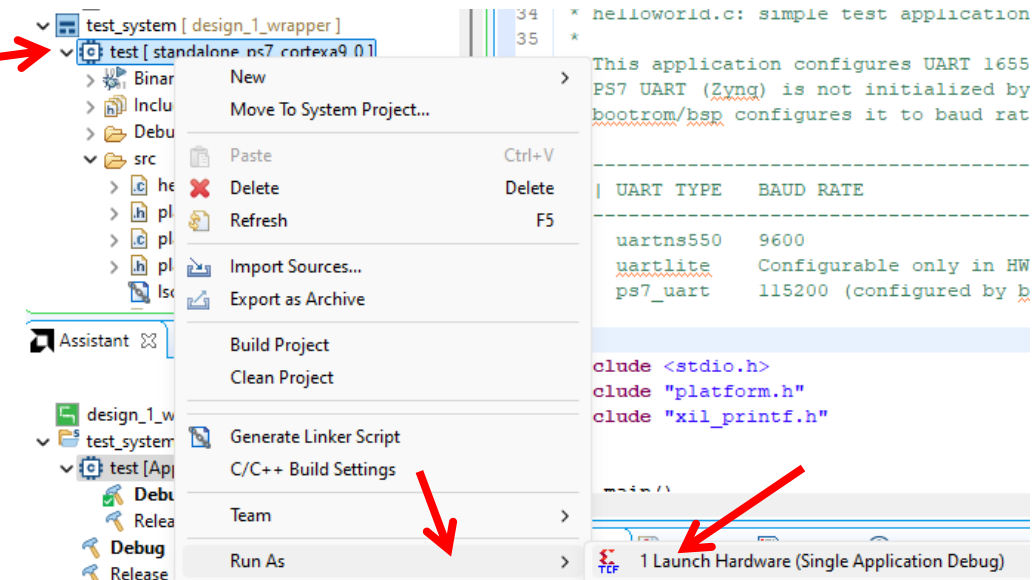
Prenos in zagon na razvojni plošči

MiniZed povezan na USB

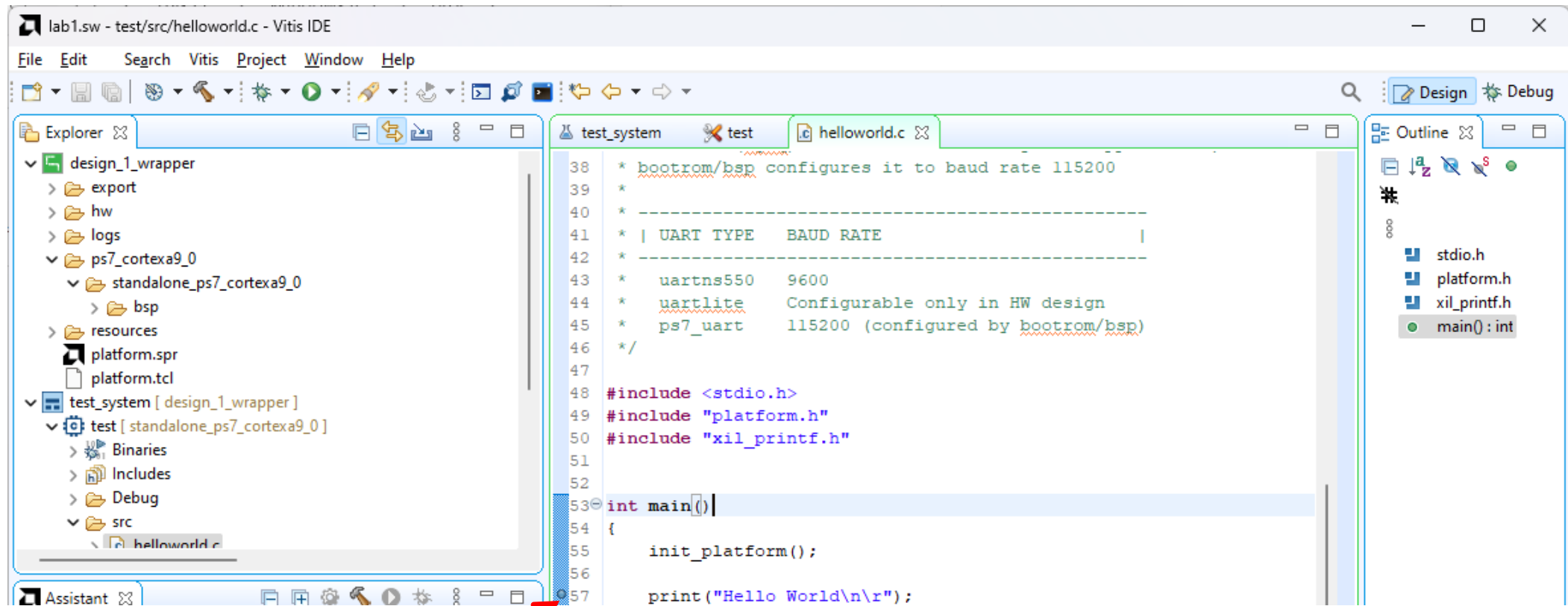
- izberi aplikacijo
- Run As
 - 1. Launch Hardware

Razhroščevanje

- Debug As
 - 1. Launch Hardware
- Izvajanje po korakih
 - npr. Step Over (F6)



Razhroščevanje



klikni za breakpoint

Razhroščevanje

The image shows a screenshot of an IDE interface with three main components:

- Explorer:** Displays a project tree. A red arrow points to the `test` folder under `standalone_ps7_cortexa9_0`. A context menu is open over this folder, with a red arrow pointing to the `Debug As` option.
- Code Editor:** Shows the source code for `helloworld.c`. A red arrow points to line 57, where a breakpoint (a small circle) has been set on the `print("Hello World\n\r");` statement. The text "klikni za breakpoint" (click for breakpoint) is written next to the arrow.
- Debug Menu:** A dropdown menu is visible at the bottom, listing three options: "1 Launch Hardware (Single Application Debug)", "2 Launch SW Emulator (Single Application Debug)", and "3 Launch Hardware (Single Application Debug) (GDB)". A red arrow points to the first option.

```
38 * bootrom/bsp configures it to baud rate
39 *
40 * -----
41 * | UART TYPE   BAUD RATE
42 * -----
43 *   uartns550   9600
44 *   uartlite    Configurable only in HW
45 *   ps7_uart    115200 (configured by bo
46 */
47
48 #include <stdio.h>
49 #include "platform.h"
50 #include "xil_printf.h"
51
52
53 int main()
54 {
55     init_platform();
56
57     print("Hello World\n\r");
58     print("Successfully ran Hello World a
59     cleanup_platform();
60     return 0;
61 }
62
```

TCF Debug Virtual Terminal - ARM Cortex-A9 MPCore #0

- 1 Launch Hardware (Single Application Debug)
- 2 Launch SW Emulator (Single Application Debug)
- 3 Launch Hardware (Single Application Debug) (GDB)

Debug Configurations...

Za UART nastavi terminal

The screenshot displays an IDE interface with several components:

- Debugger:** Shows the ARM Cortex-A9 MPCore #0 with breakpoints set at 0x0010056c and 0x00100718.
- Code Editor:** Displays the source code for `helloworld.c`. The code includes headers and defines UART configurations for `uartns550` (9600 baud) and `ps7_uart` (115200 baud). The `main` function calls `init_platform()` and prints "Hello World" and "Successful".
- Dialog Box:** A "Connect to serial port" dialog is open, showing "COM11" selected in the Port dropdown and "115200" in the Baud Rate dropdown. Other settings include Data Bits: 8, Stop Bits: 1, Parity: None, and Flow Control: None.
- Terminal:** The bottom panel shows the "Console" tab with a message: "Click on + button to add a port to the terminal." A red arrow points to the "+" button in the terminal toolbar.

Red arrows highlight the "COM11" selection in the dialog, the "+" button in the terminal toolbar, and the "Debug" tab in the IDE's bottom panel.

Izvajanje po korakih

The screenshot displays the Vitis IDE interface during a step-through debug session. The main editor shows the source code for `helloworld.c`, with the `main` function highlighted. The code includes UART configuration and prints "Hello World" and "Successfully ran Hello World application".

The left sidebar shows the project structure, including the `test_system` and `test` components. The top toolbar features a red circle around the 'Step Over' button (a play button with a vertical line through it). A red arrow points to the 'Debug' button in the top right corner.

The bottom console window shows the output of the program, with the text "Connected to COM11 at 115200" and "Hello World" circled in red. The XSC1 Process window shows the execution flow, including the `init_platform` function and the print statements.

```
35 *
36 * This application configures UART 16550 to baud rate 9600.
37 * PS7 UART (Zynq) is not initialized by this application, si
38 * bootrom/bsp configures it to baud rate 115200
39 *
40 * -----
41 * | UART TYPE   BAUD RATE
42 * |-----|
43 * | uartns550   9600
44 * | uartlite    Configurable only in HW design
45 * | ps7_uart    115200 (configured by bootrom/bsp)
46 * /
47
48 #include <stdio.h>
49 #include "platform.h"
50 #include "xil_printf.h"
51
52
53 int main()
54 {
55     init_platform();
56
57     print("Hello World\n\r");
58     print("Successfully ran Hello World application\n\r");
59     cleanup_platform();
60     return 0;

```

Console Output:

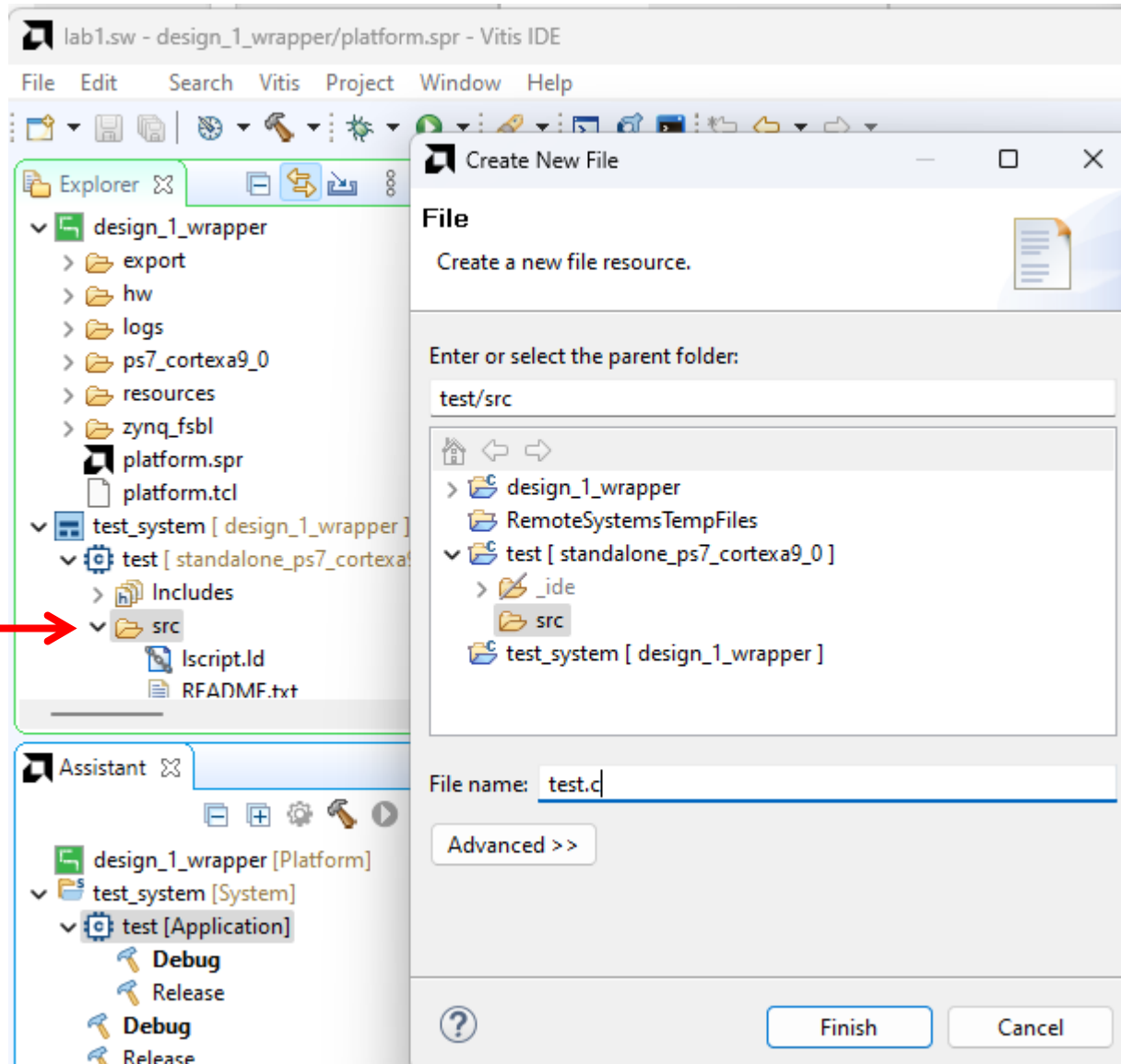
```
Connected to: Serial ( COM11, 115200, 0, 8 )
Connected to COM11 at 115200
Hello World
```

XSC1 Process Output:

```
55:   init_platform();
xsct$ Info: ARM Cortex-A9 MPCore
57:   print("Hello World\n\r");
xsct$ Info: ARM Cortex-A9 MPCore
58:   print("Successfully ran H
xsct$
```

Nova izvorna datoteka

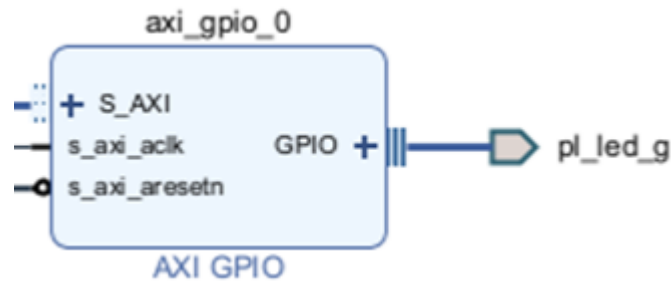
New, File



Aplikacija – dostop do registrov PL

Dostop s kazalci:

```
int *xp=(int *)0x41200000;  
*xp=255;
```



Hardware Platform Specification

Design Information

Target FPGA Device: 7z007s
Part: xc7z007sclg225-1
Created With: Vivado 2023.1
Created On: Tue Oct 1 12:24:23 2024

Note: To view ip parameters, double-click on the cell containing ip name in any of the below tables.

Address Map for processor ps7_cortexa9[0-1]

Filter: Search: 27 Loaded - 27 Shown - 0 Selected - [Custom:

Cell	Base Address	High Address	Slave Interface	Addr Range Type
ps7_intc_dist_0	0xf8f01000	0xf8f01fff	-	register
ps7_scutimer_0	0xf8f00600	0xf8f0061f	-	register
ps7_clk_0	0xf8000000	0xf8000fff	-	register
axi_gpio_0	0x41200000	0x4120ffff	S_AXI	register
ps7_scuwdt_0	0xf8f00620	0xf8f006ff	-	register
ps7_l2cachec_0	0xf8f02000	0xf8f02fff	-	register
ps7_scuc_0	0xf8f00000	0xf8f000fc	-	register

Funkcije za AXI_GPIO (komponenta v FPGA)

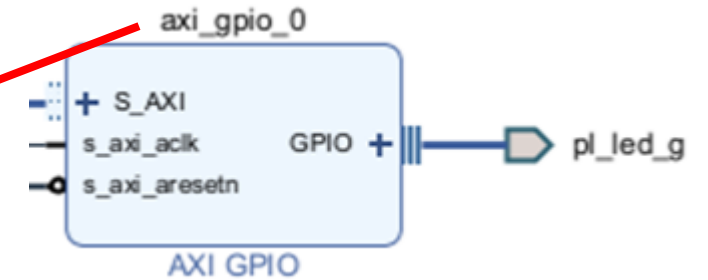
```
#include "xgpio.h"
```

```
XGpio gpio;
```

```
XGpio_Initialize(&gpio, XPAR_AXI_GPIO_0_DEVICE_ID);
```

```
XGpio_DiscreteWrite(&gpio, 1, 0);
```

kanal vrednost



Funkcija za PS GPIO (enota ARM procesorja)

```
#include "xgpiops.h"
```

```
XGpioPs gpio_ps;
```

```
XGpioPs_Config *XGPIO_Config;
```

```
int btn;
```

```
XGPIO_Config = XGpioPs_LookupConfig(XPAR_PS7_GPIO_0_DEVICE_ID);
```

```
XGpioPs_CfgInitialize(&gpio_ps, XGPIO_Config, XGPIO_Config->BaseAddr);
```

```
XGpioPs_SetDirectionPin(&gpio_ps, 0, 0); // button
```

```
XGpioPs_SetDirectionPin(&gpio_ps, 52, 1); // red
```

```
XGpioPs_SetDirectionPin(&gpio_ps, 53, 1); // green
```

```
XGpioPs_SetOutputEnablePin(&gpio_ps, 52, 1);
```

```
XGpioPs_SetOutputEnablePin(&gpio_ps, 53, 1);
```

```
XGpioPs_WritePin(&gpio_ps, 52, 1);
```

```
XGpioPs_WritePin(&gpio_ps, 53, 0);
```

```
btn = XGpioPs_ReadPin(&gpio_ps, 0);
```

inicializacija



nastavi smer



omogoči izhode



nastavi LED



beri tipko