

In-System Sources and Probes Editor i Quartus Prime.

Sammanfattning: Tillvägagångssätt för användning av In-System Sources and Probes Editor

Innehållsförteckning

1 Beskrivning av verktyget.....	3
Hårdvaru och mjukvarukrav.....	3
Stödda anordningsfamiljer.....	3
In-System Sources and Probes Editor'n har tre vyer:.....	4
2 Använd följande steg vid användning av verktyget:.....	5
3 Demonstration.....	6

1 Beskrivning av verktyget

In-System Sources and Probes Editor utökar möjligheterna att verifiera en design genom att enkelt kontrollera vilken intern signal som helst och förser Quartus med en totalt dynamisk felsöknings miljö. I samarbete med SignalProbe och SignalTap II Logic Analyzer så får man en kraftfull felsöknings miljö där man kan generera stimuli och få respons från logiken. För att köra virtuella insignaler i designen kan man använda Virtual JTAG IP core eller In-System Memory Content Editor.

In-System Sources and Probes Editor'n består av ALTSOURCE_PROBE IP kärna och ett gränssnitt för att styra instanser av denna under drift. Varje kärna ger in och utgångar där man via in-portarna kan köra valda signaler och via ut-portarna läsa av utvalda signaler. ALTSOURCE_PROBE IP kärnan upprättar en registerkedja till in eller ut portarna i designen. JTAG används för att skifta data mellan ALTSOURCE_PROBE IP kärn-instanser. ALTSOURCE_PROBE IP kärnan hanterar kommunikationen mellan JTAG och registren i designen för att tillhandahålla en enkel byggsten för simulering och sondering av egen design. In-System Sources and Probes Editor förser med en-cykels läs och skriv till logiska noder. Felsökning på låg-nivå genom växling av ingångar, detta tack vare åtkomst till logiska noder. Tillsammans med SignalTap II Logic Analyzer kan man tvinga till utlösningvillkor och på så sätt isolera ett fel.

Man kan lätt med In-System Sources and Probes Editor lägga till kontrollsignaler i sin design som virtuellt stimuli och detta kan förenkla följande:

1. Skapa virtuella tryckknappar
2. Skapa en virtuell frontpanel för att kommunicera med din design
3. Emulera externa data sensor
4. Övervaka och ändra run-time tidskonstanter i farten

Tcl kommandon stöds av In-System Källor och Probes Editor som ett gränssnitt mellan ALTSOURCE_PROBE IP-kärn instanser för att öka graden av automatisering.

Hårdvaru och mjukvarukrav

1. Quartus Prime eller Quartus Prime Lite Edition med TalkBack aktivt.
2. Nedladdningskabel
3. Altera utvecklingskit eller liknande med JTAG anslutning

Stödda anordningsfamiljer

1. Arria
2. Stratix
3. Cyclone
4. MAX

Signaler som man vill titta på är anslutna till en instans av In-System Sources and Probes IP kärnan. Efter kompilering kan man kontrollera varje instans via In-System Sources and Probes Editor vyn

eller Tel gränssnittet. Innan användning av In-System Sources and Probes Editor'n så måste man instansiera In-System Sources and Probes IP kärnan, förslagsvis via IP katalogen.

In-System Sources and Probes Editor stödjer inte simulering så den måste tas bort innan man skapar en netlist.

In-System Sources and Probes Editor ger fullständig kontroll över ALTSOURCE_PROBE IP kärnor i konstruktionen. Editorn ger möjlighet att visa alla styrbara instanser i realtid.

In-System Sources and Probes Editor'n har tre vyer:

1. JTAG Chain Configuration – Programmeringsvy. Här väljer man vilken hårdvara och enhet som ska programmeras.
2. Instance Manager – Information om de genererade instanserna, samt kontrollering av data som krävs för In-System Sources and Probes Editor'n.

Denna vy har några knappar/under vyer:

- Read Probe Data
 - Continuously Read Probe Data
 - Stop Continuously Reading Probe Data
 - Write Source Data
 - Probe Read Interval
 - Event Log
 - Write Source Data
3. In-System Sources and Probes Editor – Loggar all data läst från vald instans. Tillåter modifiering av källdata som skrivs till enheten.

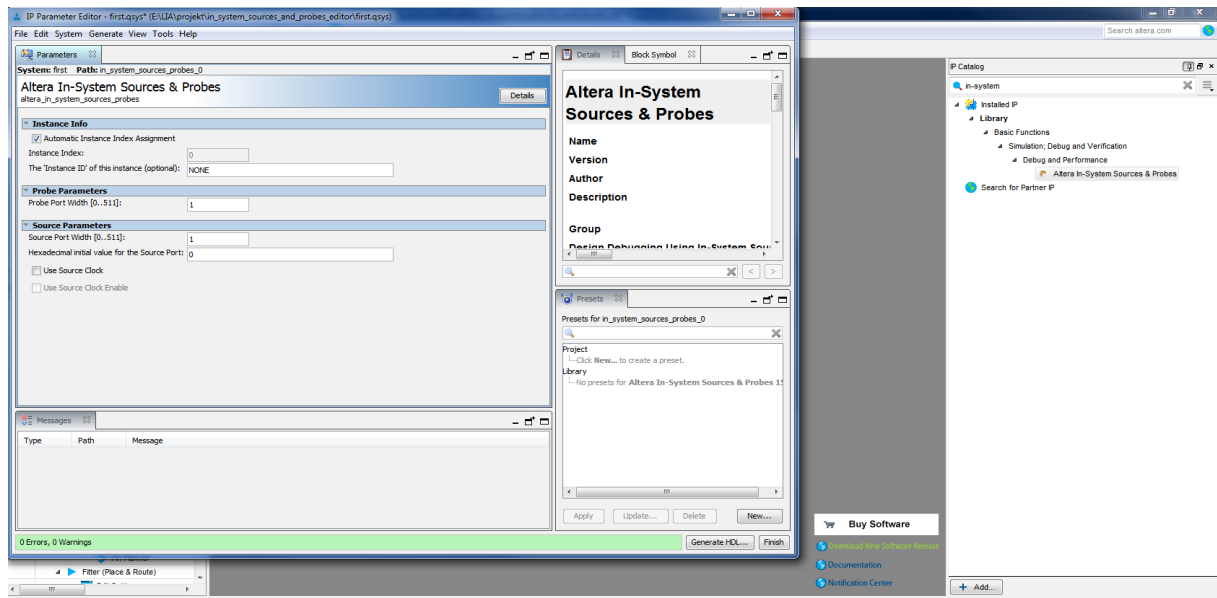
2 Använd följande steg vid användning av verktyget:

Programmering av enheten

1. I Quartus: öppna Tools>IP Catalog.
2. Välj In-System Sources and Probes IP core.
3. Välj ett namn för IP kärnan.
4. Generera IP kärnan. Använd sedan den genererade kärnan, som är baserad på dina specifikationer, till att instansiera In-System Sources and Probes IP kärnan.
5. Kompilera projektet.
6. Öppna In-System Sources and Probes Editor.
7. Välj rätt hårdvara i JTAG Chain Configuration, samt rätt device.
8. Leta efter SRAM objekts filen (.sof), som innehåller instansen eller In-System Sources and Probes instans.
9. Programmera enheten genom att klicka på Program Device.

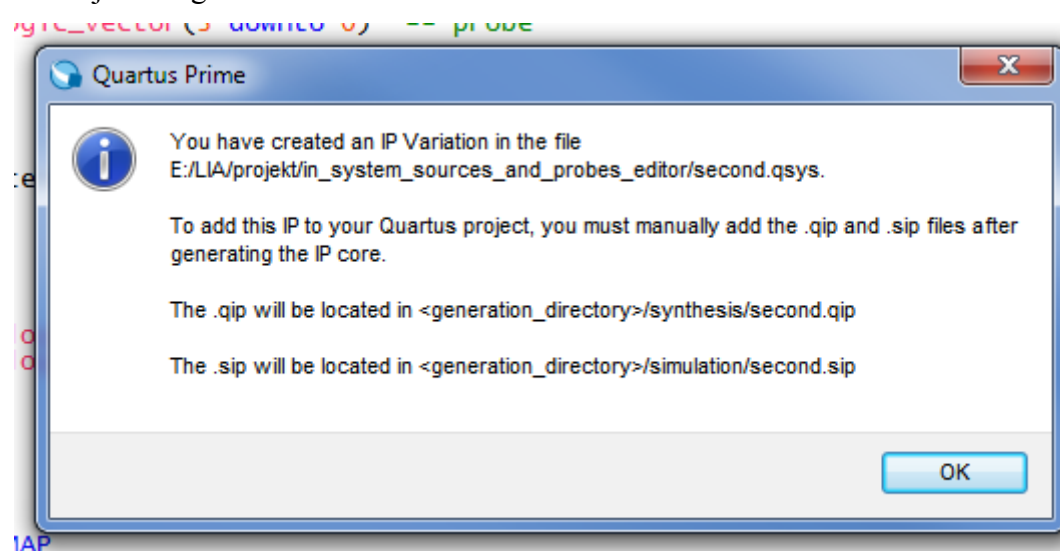
3 Demonstration

Här kommer en kort demonstration av ett mycket enkelt system med bara 4 knappar och 4 lysdioder. Börja med ett nytt projekt. Hämta In-System Sources and Probes IP kärna i IP Katalogen enligt figur 1.



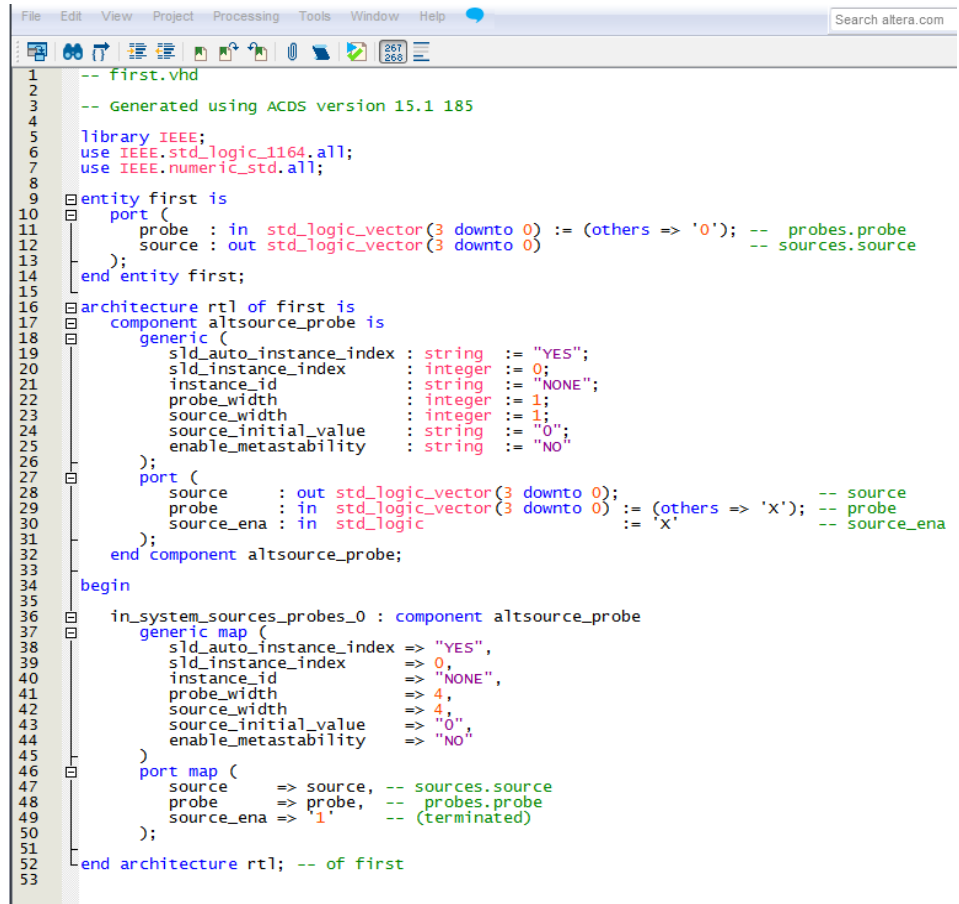
Figur 1: Figuren visar en skärmdump där In-System Sources and Probes IP kärna i IP Katalogen inkluderas till projektet.

Därefter måste filerna inkluderas till projektet och det kommer som regel upp en ruta som påminner om detta. Följande figur visar rutan.



Figur 2: Påminnelse om inkludering.

När kärnan är genererad så måste den genererade filen som syns i figur 3, inkluderas till projektet via Settings menyn.



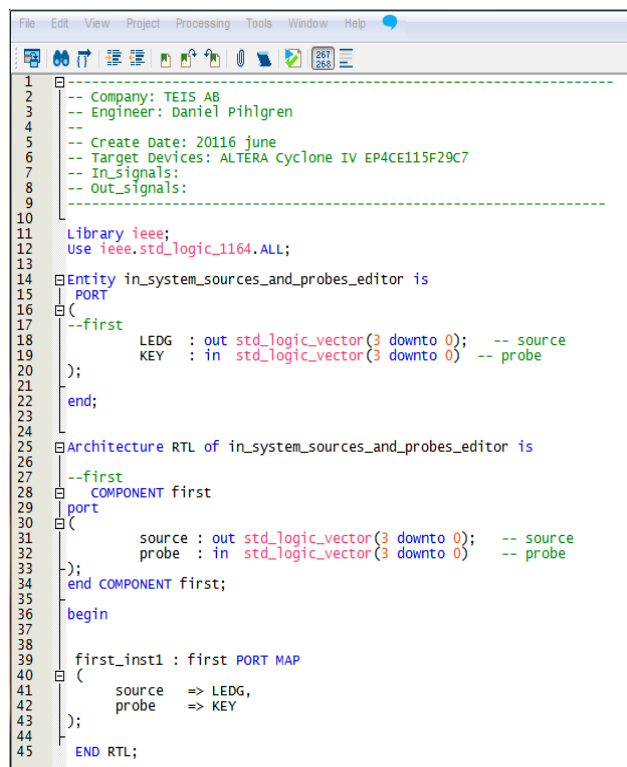
```

1  -- first.vhd
2  -- Generated using ACDS version 15.1 185
3
4  library IEEE;
5  use IEEE.std_logic_1164.all;
6  use IEEE.numeric_std.all;
7
8  entity first is
9  port (
10     probe : in std_logic_vector(3 downto 0) := (others => '0'); -- probes.probe
11     source : out std_logic_vector(3 downto 0) -- sources.source
12 );
13 end entity first;
14
15 architecture rtl of first is
16     component altsource_probe is
17     generic (
18         sld_auto_instance_index : string := "YES";
19         sld_instance_index      : integer := 0;
20         instance_id             : string := "NONE";
21         probe_width             : integer := 1;
22         source_width            : integer := 1;
23         source_initial_value    : string := "0";
24         enable_metastability    : string := "NO";
25     );
26     port (
27         source : out std_logic_vector(3 downto 0);
28         probe  : in std_logic_vector(3 downto 0) := (others => 'x'); -- source
29         source_ena : in std_logic := 'x'; -- probe
30     );
31 end component altsource_probe;
32
33 begin
34
35     in_system_sources_probes_0 : component altsource_probe
36     generic map (
37         sld_auto_instance_index => "YES",
38         sld_instance_index      => 0,
39         instance_id             => "NONE",
40         probe_width             => 4,
41         source_width            => 4,
42         source_initial_value    => "0",
43         enable_metastability    => "NO",
44     )
45     port map (
46         source => source, -- sources.source
47         probe  => probe,  -- probes.probe
48         source_ena => '1', -- (terminated)
49     );
50
51 end architecture rtl; -- of first
52
53

```

Figur 3: Genererad fil.

Därefter instansieras i top_filen och detta syns i figur 4.



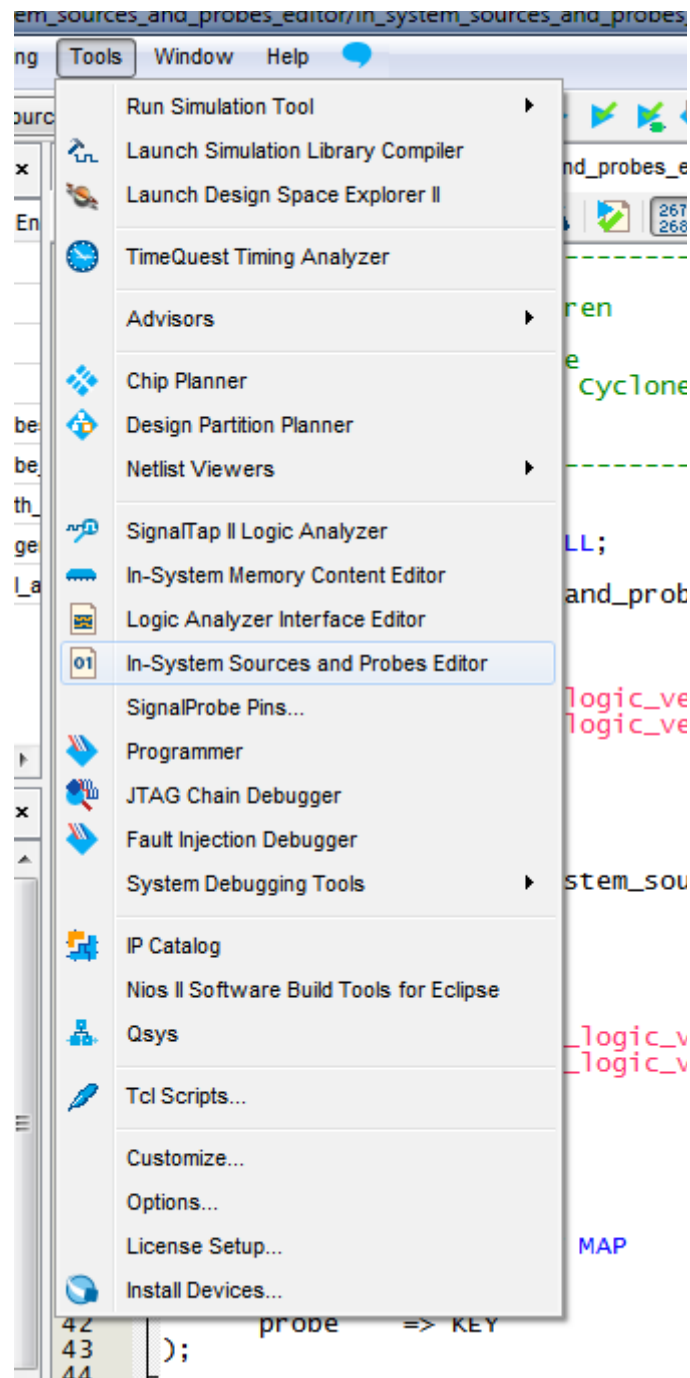
```

1  -- Company: TEIS AB
2  -- Engineer: Daniel Pihlgren
3
4  -- Create Date: 2016 June
5  -- Target Devices: ALTERA Cyclone IV EP4CE115F29C7
6  -- In signals:
7  -- Out signals:
8
9
10
11 library IEEE;
12 use IEEE.std_logic_1164.ALL;
13
14 entity in_system_sources_and_probes_editor is
15 port (
16     --first
17     LEDG : out std_logic_vector(3 downto 0); -- source
18     KEY  : in std_logic_vector(3 downto 0) -- probe
19 );
20 end;
21
22
23
24 architecture RTL of in_system_sources_and_probes_editor is
25     --first
26     component first
27     port
28     (
29         source : out std_logic_vector(3 downto 0); -- source
30         probe  : in std_logic_vector(3 downto 0) -- probe
31     );
32 end component first;
33
34 begin
35
36     first_inst1 : first PORT MAP
37     (
38         source => LEDG,
39         probe  => KEY
40     );
41
42 end RTL;
43
44
45

```

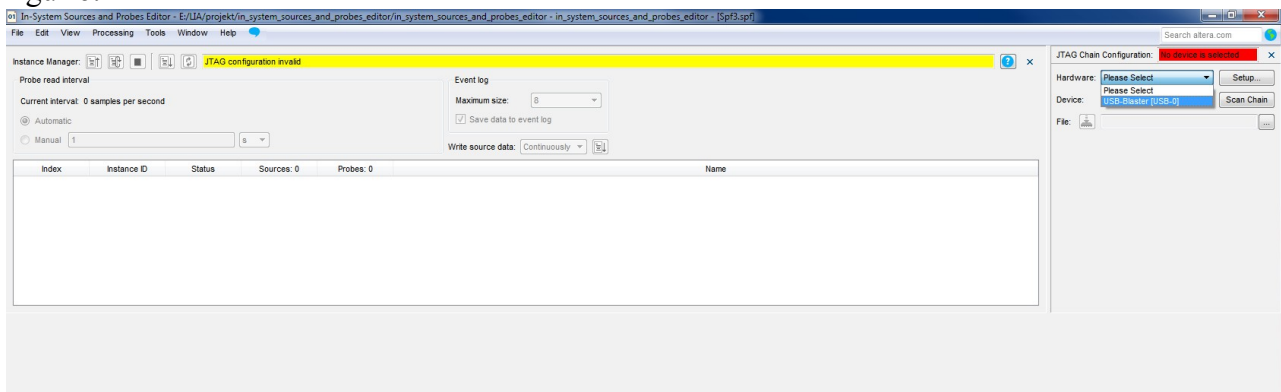
Figur 4: Top_level filen

Gör Pin-Assignment och kompilera. När kompileringen är klar öppnar man In-System Sources and Probes Editor från Tools menyn, enligt figur 5.



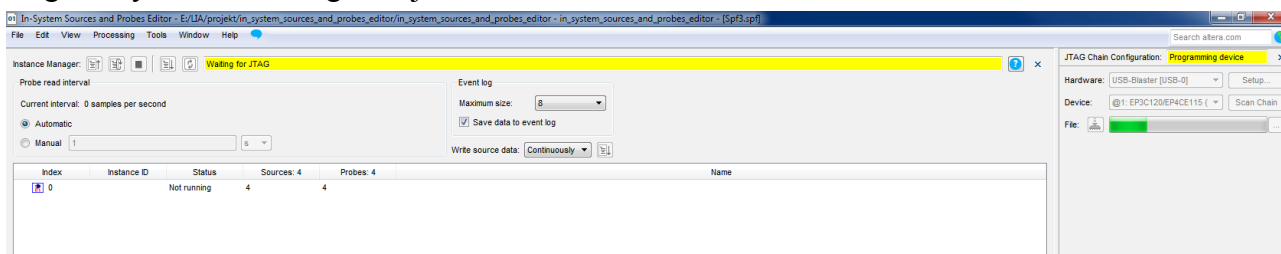
Figur 5: Välj In-System Sources and Probes Editor från Tools menyn.

Här väljs fil att programmera hårdvaran med. Först måste man välja rätt port för JTAG'en att prata med hårdvaran, sen måste .sof filen inkluderas. Se även till att det är rätt device som är vald. Se figur 6.



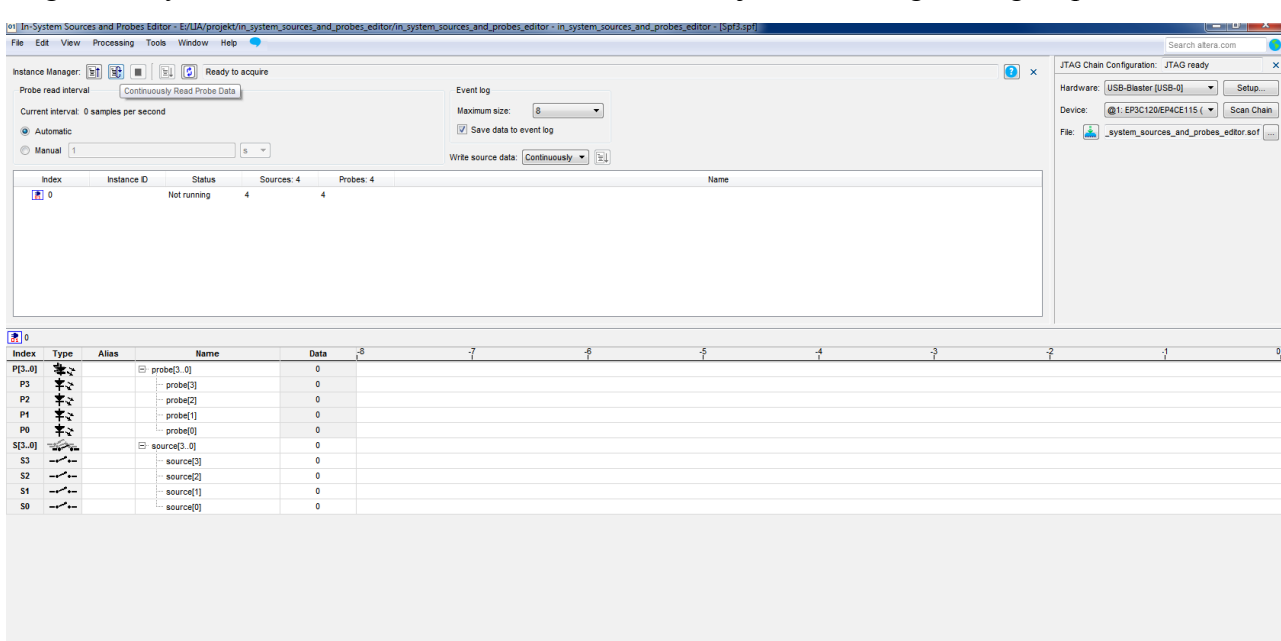
Figur 6: Val av hårdvara att prata med device:n genom.

I figur 7 syns nedladdning av mjukvaran .sof filen till hårdvaran.



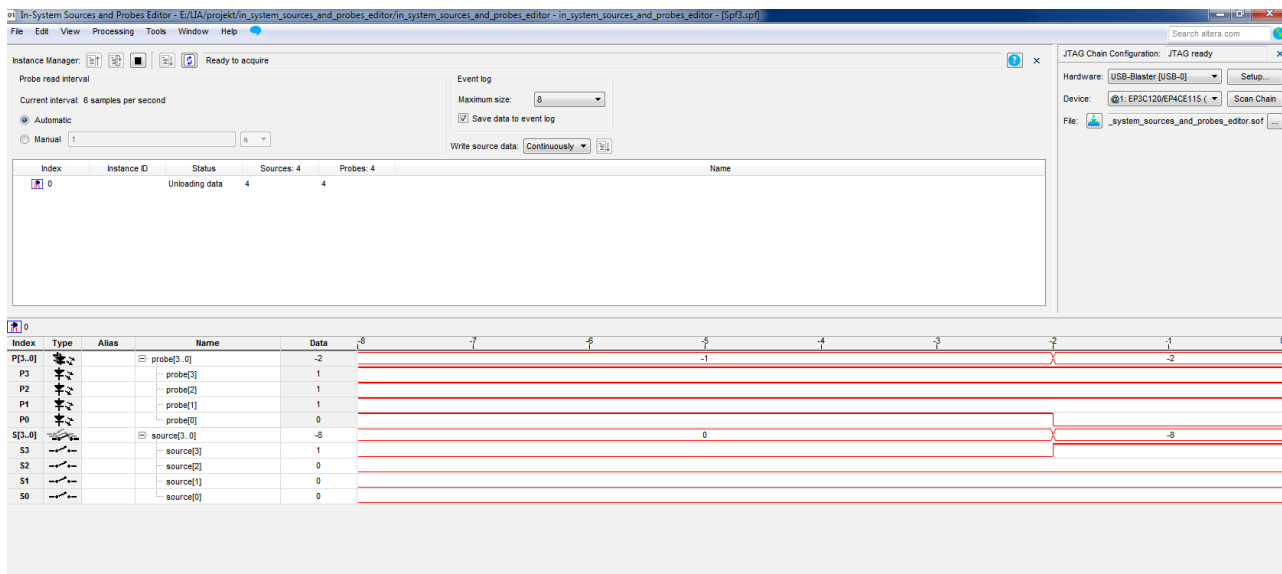
Figur 7: Nedladdning av konfigureringsfil till hårdvaran.

I figur 8 är systemet nedladdat och instans hittad. Här väljs kontinuerlig läsning av probe data.

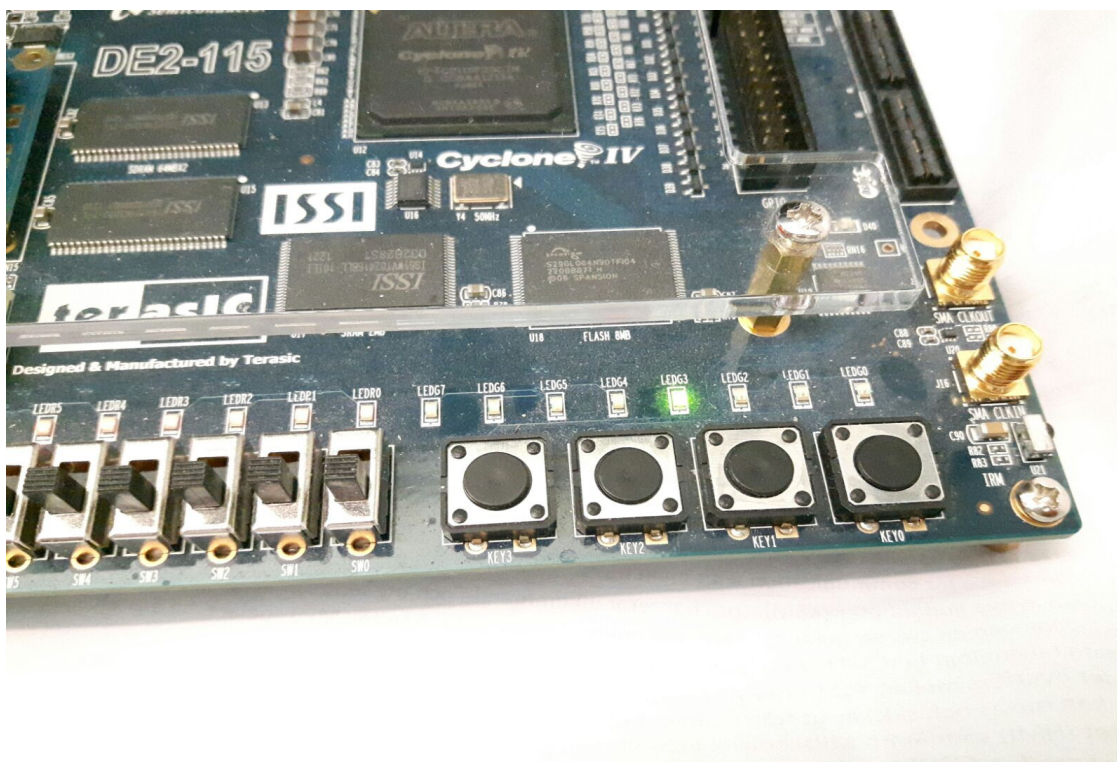


Figur 8: Programmering via JTAG vyn.

Nedan syns den kontinuerliga avläsningen tillslagen samt tryck på knapp 0. Ledg 3 tänds via skrivning direkt till source[3]. Mer av detta syns i nästa bild.



Figur 9: Kontinuerlig läsning av data från probe samt skrivning till source.



Figur 10: Ledg3 tänd via In-System Sources and Probes Editor.