

TU0780
Tutorial
Using Identify ME with Libero SoC



Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo,
CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Fax: +1 (949) 215-4996
Email: sales.support@microsemi.com
www.microsemi.com

© 2017 Microsemi Corporation. All rights reserved. Microsemi and the Microsemi logo are trademarks of Microsemi Corporation. All other trademarks and service marks are the property of their respective owners.

Microsemi makes no warranty, representation, or guarantee regarding the information contained herein or the suitability of its products and services for any particular purpose, nor does Microsemi assume any liability whatsoever arising out of the application or use of any product or circuit. The products sold hereunder and any other products sold by Microsemi have been subject to limited testing and should not be used in conjunction with mission-critical equipment or applications. Any performance specifications are believed to be reliable but are not verified, and Buyer must conduct and complete all performance and other testing of the products, alone and together with, or installed in, any end-products. Buyer shall not rely on any data and performance specifications or parameters provided by Microsemi. It is the Buyer's responsibility to independently determine suitability of any products and to test and verify the same. The information provided by Microsemi hereunder is provided "as is, where is" and with all faults, and the entire risk associated with such information is entirely with the Buyer. Microsemi does not grant, explicitly or implicitly, to any party any patent rights, licenses, or any other IP rights, whether with regard to such information itself or anything described by such information. Information provided in this document is proprietary to Microsemi, and Microsemi reserves the right to make any changes to the information in this document or to any products and services at any time without notice.

About Microsemi

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for aerospace & defense, communications, data center and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; enterprise storage and communication solutions, security technologies and scalable anti-tamper products; Ethernet solutions; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, California, and has approximately 4,800 employees globally. Learn more at www.microsemi.com.

Contents

1	Revision History	1
1.1	Revision 1.0	1
2	Introduction	2
2.1	Identify Flow	2
3	Using Identify ME with Libero SoC	4
3.1	Step 1: Create a Libero SoC Project	4
3.2	Step 2: Create an Identify Implementation	6
3.3	Step 3: Launch Identify Instrumentor	8
3.4	Step 4: Instrument the Design	9
3.5	Step 5: Change the Implementation View (Optional)	11
3.6	Step 6: Launch and View the New Implementation (Optional)	11
3.7	Step 7: Organize Instrumented EDIF in Libero SoC	11
3.8	Step 8: Launch Identify Debugger	14

Figures

Figure 1	Identify Flow Diagram	3
Figure 2	Libero Start Page	4
Figure 3	Libero New Project	5
Figure 4	Tool Profiles	5
Figure 5	Configure Identify Launch in Synplify Pro	6
Figure 6	Identify Instrumentor Set to Integrated Mode	6
Figure 7	Select New Identify Instrumentor	7
Figure 8	Identify Implementation under Synthesis Folder.	7
Figure 9	Launching Identify Instrumentor	8
Figure 10	SRS Instrumentation	8
Figure 11	Launching Identify Instrumentor	9
Figure 12	Instrumentation with Identify Instrumentor	10
Figure 13	Running Final Synthesis on Instrumented Design	10
Figure 14	Re-Launching Identify in Synplify Pro	11
Figure 15	Importing .EDN File in Libero SoC	12
Figure 16	Libero Compile, Place and Route, Generate Programming, and Run Program Action	13
Figure 17	Set Identify Debugger Profile within Libero	14
Figure 18	Set Identify Debugger Profile within Libero	15
Figure 19	Identify Debugger	16

1 Revision History

The revision history describes the changes that were implemented in the document. The changes are listed by revision, starting with the most current publication.

1.1 Revision 1.0

Revision 1.0 is the first publication of this document.

2 Introduction

Synopsys Identify ME is an RTL debugger that allows you to probe internal signals of the design directly from the programmed FPGA in the original RTL code or in a waveform viewer.

Identify consists of two separate tools: instrumentor and debugger. The instrumentor compiles and inserts the required logic into the original RTL to create an instrumented design that allows probing of internal signals. The debugger communicates to the FlashPro, FlashPro Lite, and FlashPro3/4/5 programmer through JTAG to get the probed signal values.

This tutorial describes how to use Identify ME instrumentor and debugger within Libero® System-on-Chip (SoC) flow. It also describes how Identify instrumentor works with Synplify Pro.

2.1 Identify Flow

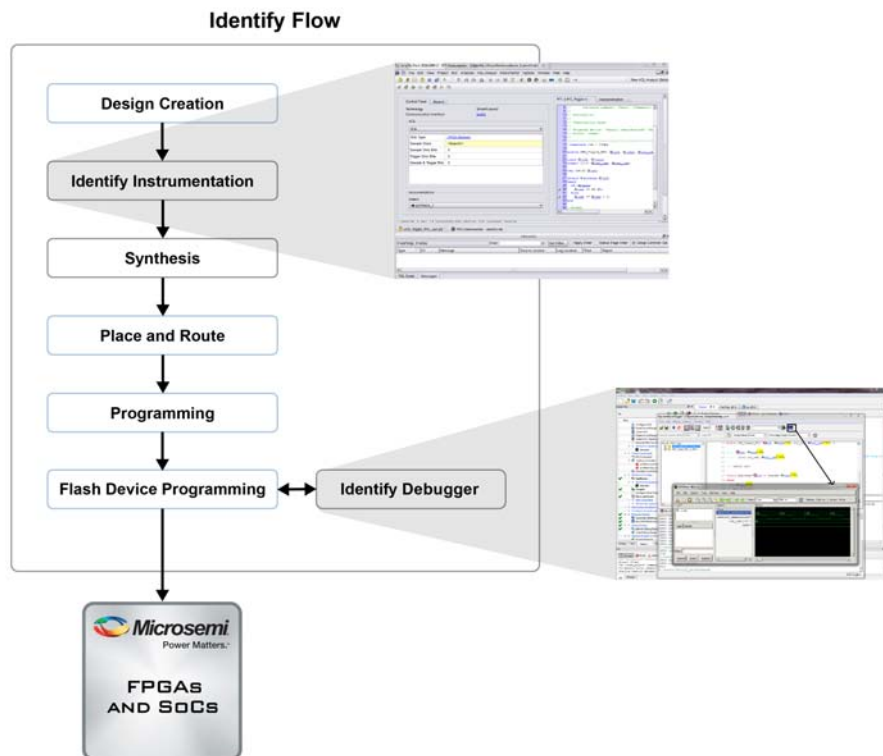
The following diagram shows the debug flow with Identify. Before synthesis, the RTL design is instrumented using the Identify ME instrumentor. Instrumentation is a process of creating instrumented HDL sources that can be used to debug the design in the Identify ME debugger.

The instrumentor allows you to:

1. Set the sample signals
2. Trigger points for conditional sampling
3. Watchpoints for allowing specific event-driven sampling of the signals within the design

Once this is done, the Identify ME debugger communicates with the instrumented source and captures the operation of the device through the flash device programmer. The sampled signal values can be viewed directly in the RTL or on a waveform viewer.

Figure 1 • Identify Flow Diagram

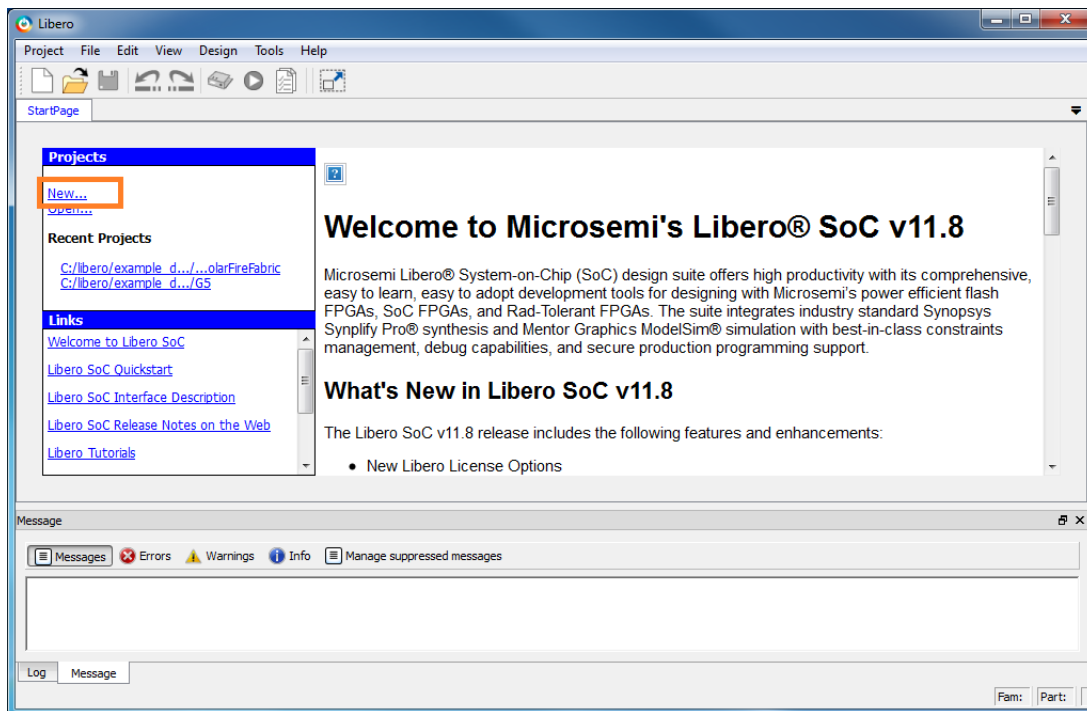


3 Using Identify ME with Libero SoC

3.1 Step 1: Create a Libero SoC Project

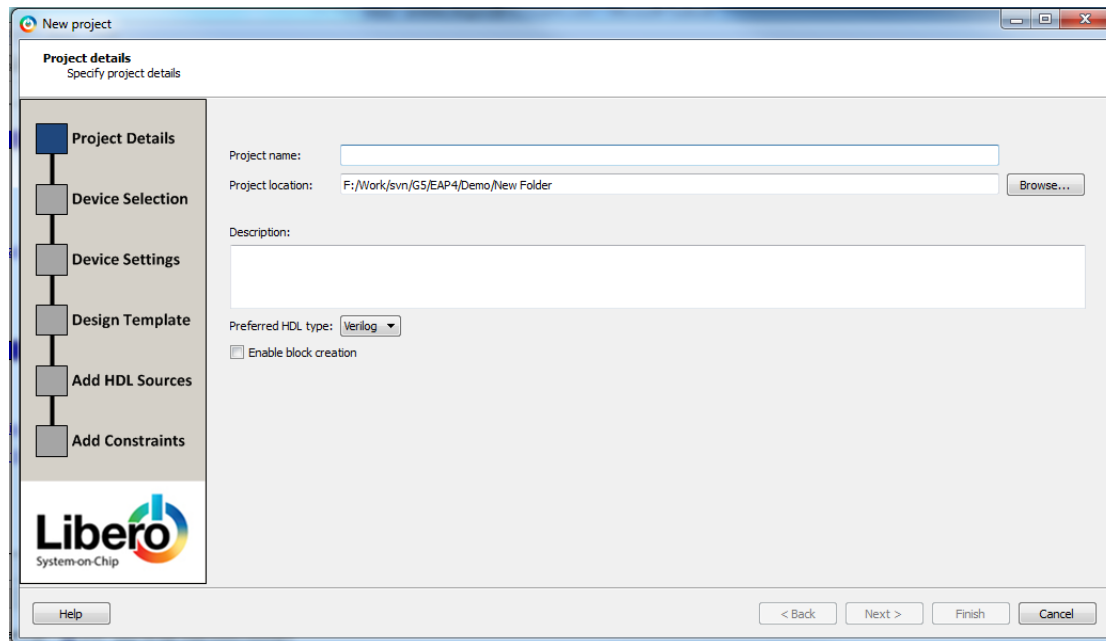
1. Download and install Libero SoC v11.8 on the host PC.
2. Start Libero SoC v11.8, and in the **Project** menu, click **New Project** or click **New** from the start page as shown in the following figure.

Figure 2 • Libero Start Page



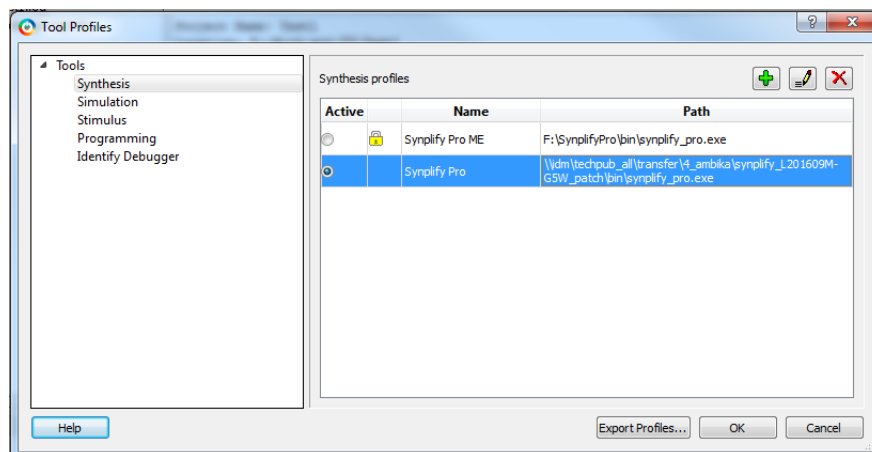
The New project window is displayed as shown in the following figure.

Figure 3 • Libero New Project



3. Select the project settings and click **Finish** to create a new project.
4. From the **Projects** menu, click **Tool Profiles** and select **Synplify Pro** (for Libero SoC Platinum customers) as the synthesis tool.

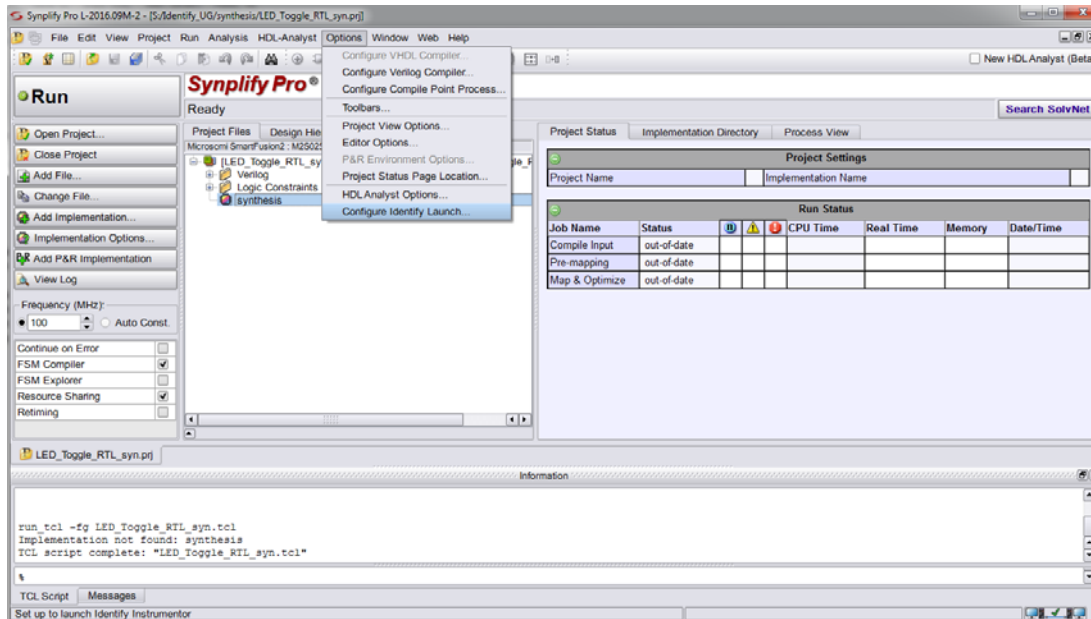
Figure 4 • Tool Profiles



3.2 Step 2: Create an Identify Implementation

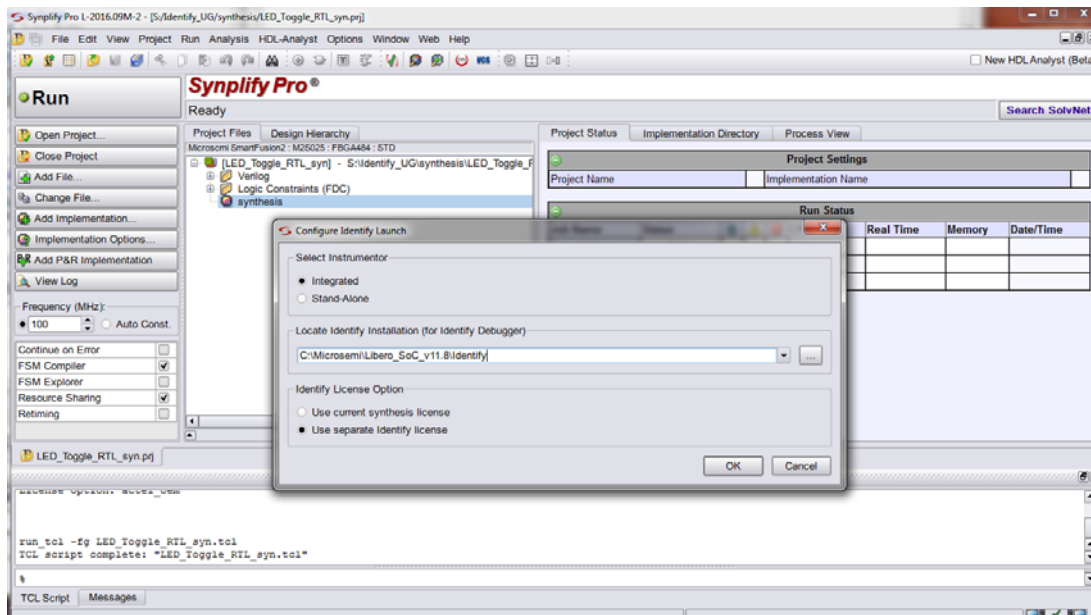
1. After running synthesis with the non-instrumented design in Synplify Pro, create an Identify implementation.
2. In the SynplifyPro window, from the **Options** menu click **Configure Identify Launch** as shown in the following figure.

Figure 5 • Configure Identify Launch in Synplify Pro



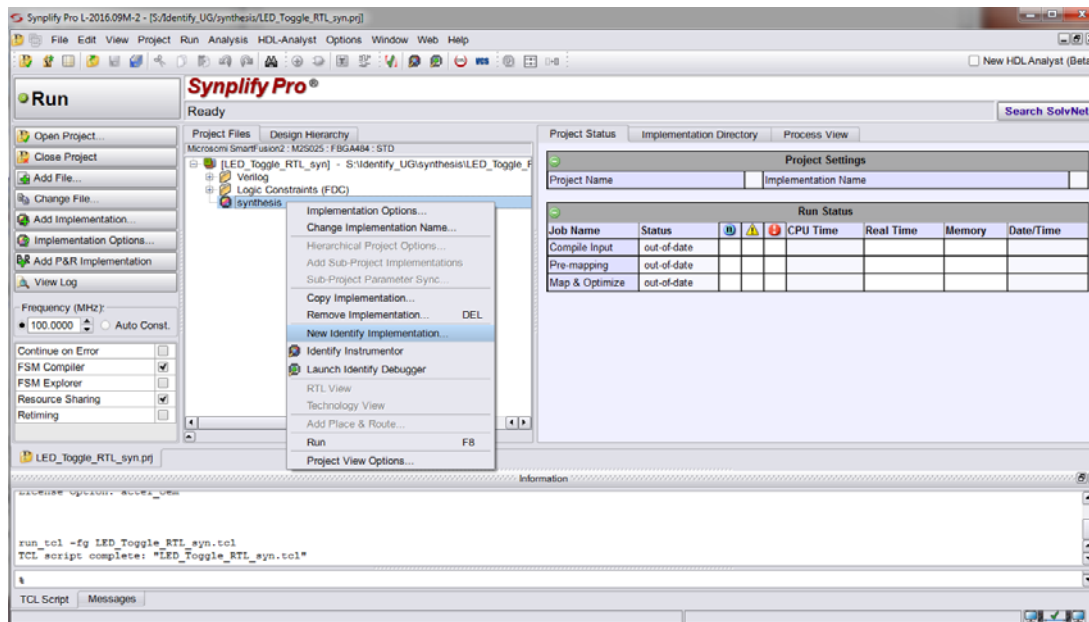
3. Select the location to install the Identify debugger as shown in the following figure.

Figure 6 • Identify Instrumentor Set to Integrated Mode



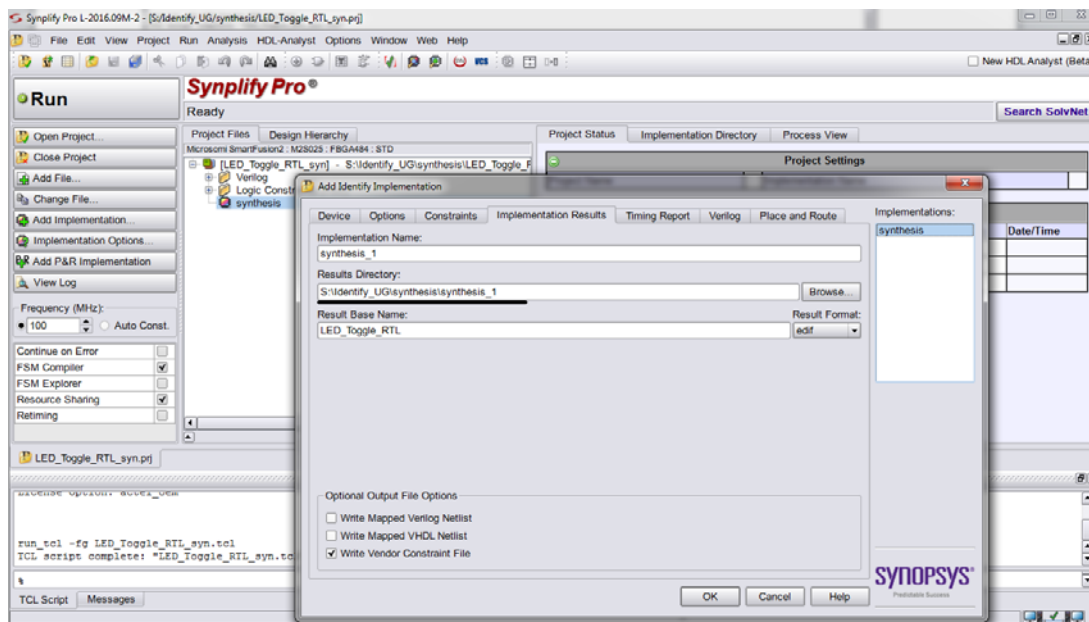
- Right-click on the existing synthesis implementation and select **New Identify Implementation** as shown in the following figure.

Figure 7 • Select New Identify Instrumentor



SynplifyPro creates the new implementation at default location:
 <Libero_Project_Location>\synthesis\synthesis_1 as shown in the following figure.

Figure 8 • Identify Implementation under Synthesis Folder.



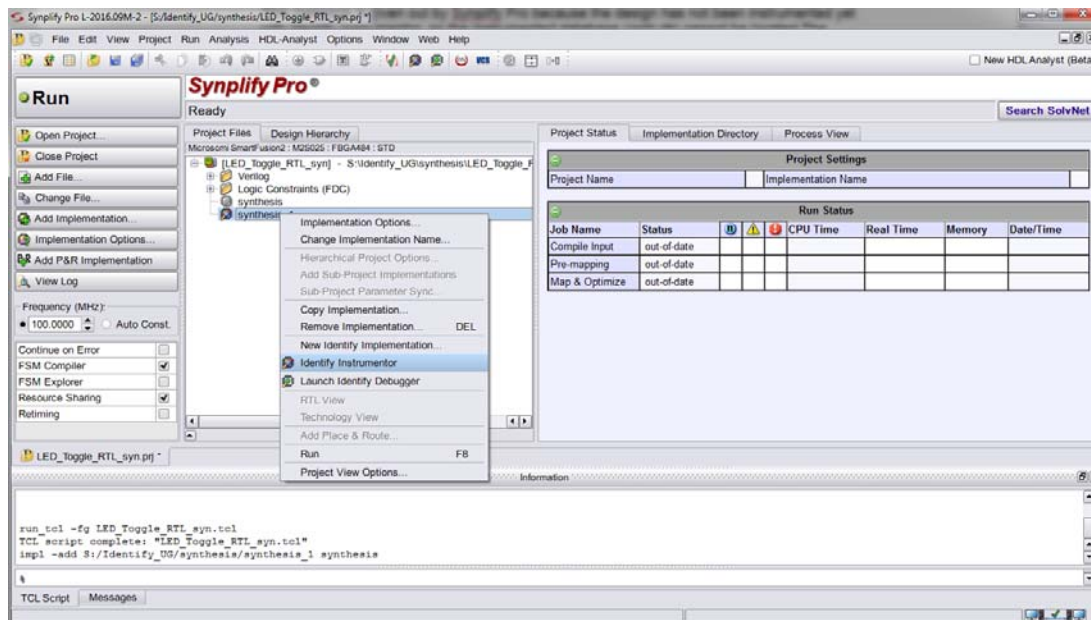
Note: You can edit **Implementation Name** in the **Implementation Results** tab. Must not change the defined Results Directory path as shown in the preceding figure.

Note: The Identify implementation folder must be under synthesis folder of Libero project. This is required for Libero to pick the correct identify instrumented synthesis netlist.

3.3 Step 3: Launch Identify Instrumentor

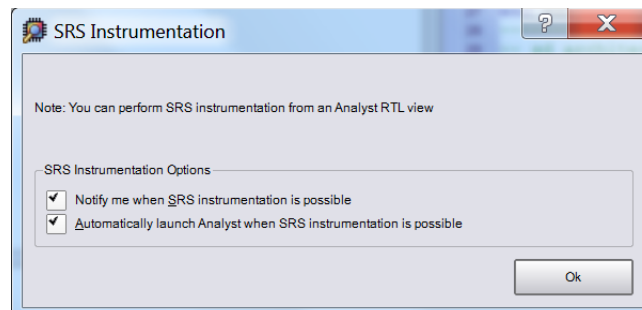
1. After creating a new Identify implementation, right-click on **synthesis_1** and select **Identify Instrumentor** as shown in the following figure.

Figure 9 • Launching Identify Instrumentor



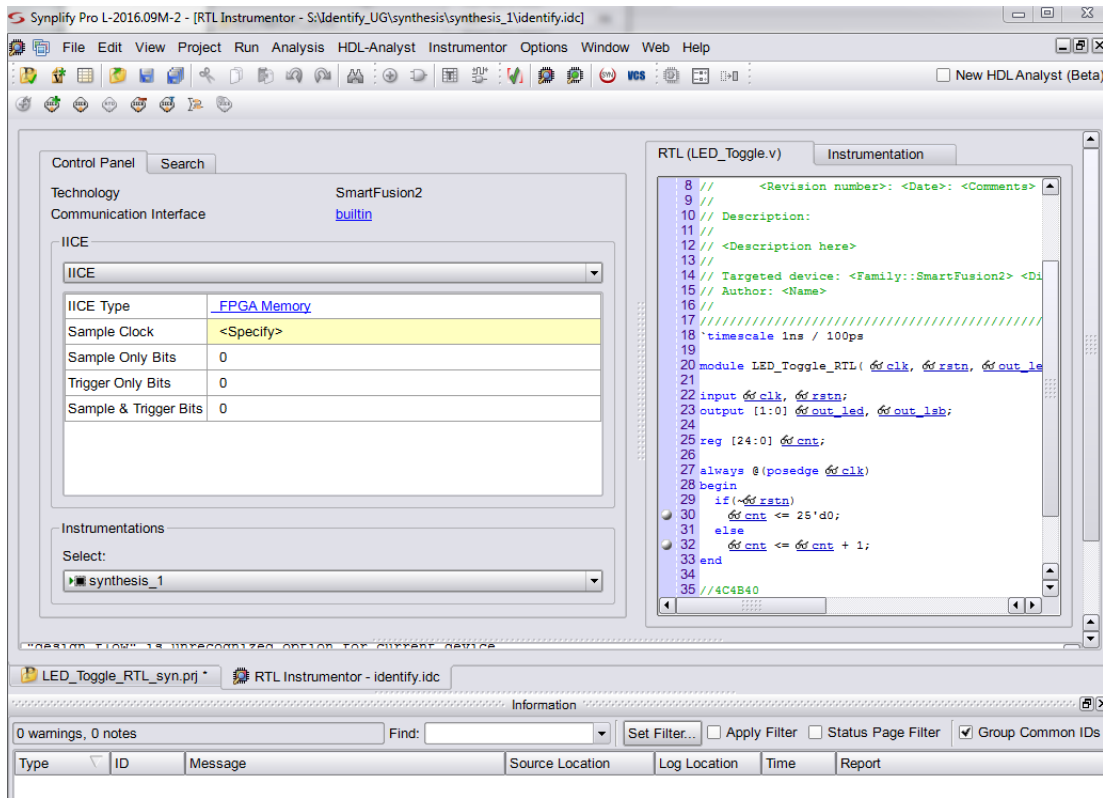
2. Click **Ok** in the SRS Instrumentation dialog box.

Figure 10 • SRS Instrumentation



Identify Instrumentor is integrated with SynplifyPro and is launched as shown in the following figure.

Figure 11 • Launching Identify Instrumentor

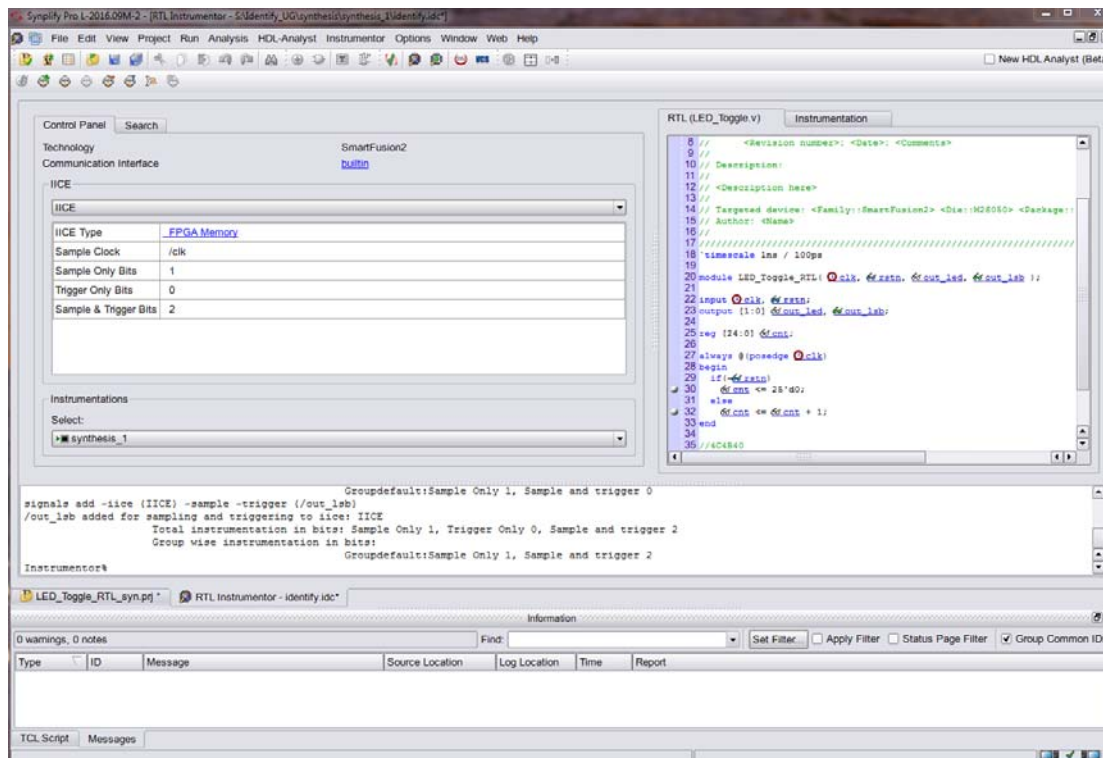


3.4 Step 4: Instrument the Design

1. You can instrument the design in Identify Instrumentor by selecting a sample clock and sample trigger signals.

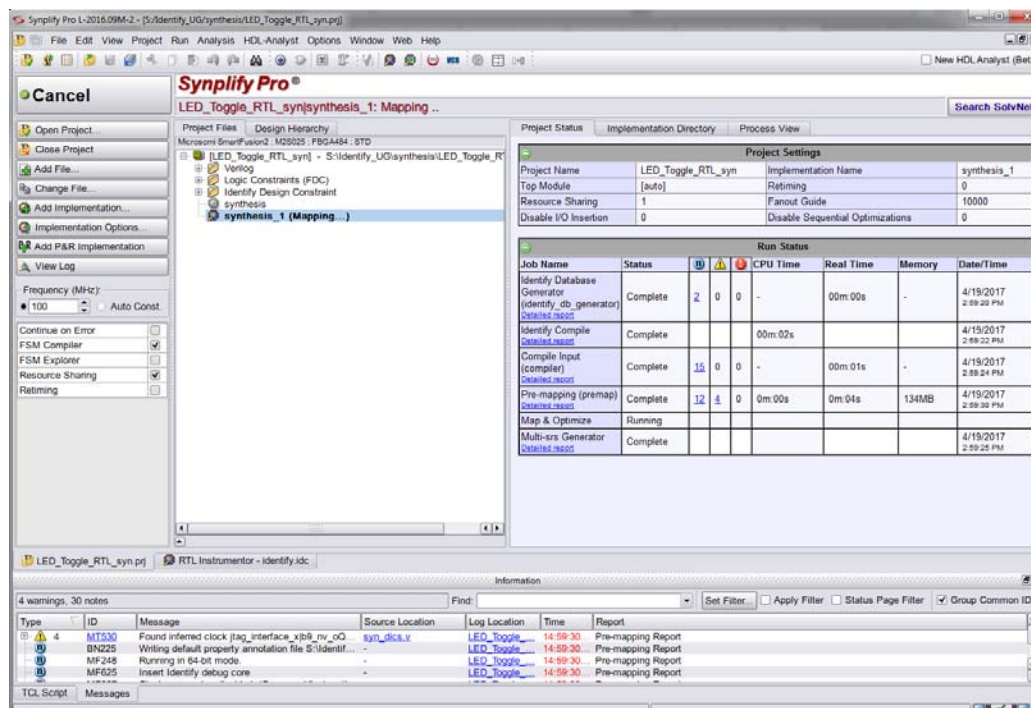
Note: In the Identify Instrumentor window, the signals available for instrumentation will have an icon in front of it as shown in the following figure. To choose the sample clock, right-click on the signal name and select **Sample Clock**. In the same way, select sample and trigger signals.

Figure 12 • Instrumentation with Identify Instrumentor



2. From the **File** menu, select **Save Project Instrumentation** to save and instrument the design.
3. Close the Instrumentor and run the final synthesis in Synplify Pro as shown in the following figure.

Figure 13 • Running Final Synthesis on Instrumented Design



4. Synplify Pro maps the design into logic gates.

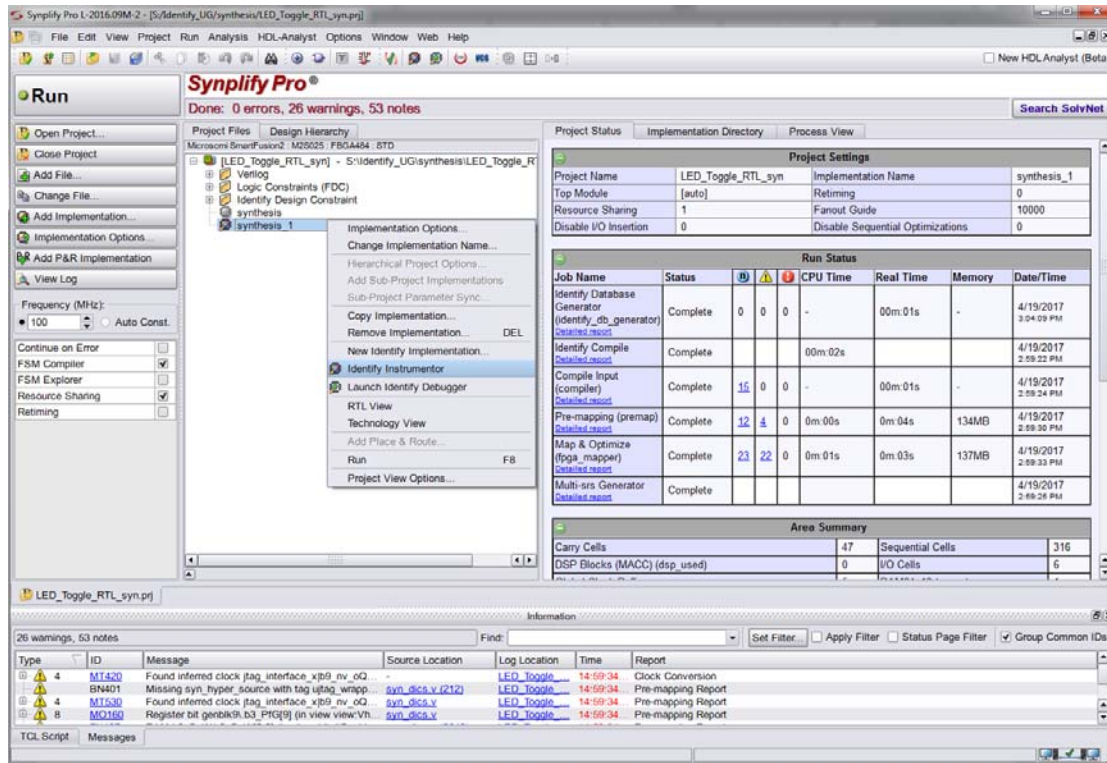
3.5 Step 5: Change the Implementation View (Optional)

You can change the implementation view in Synplify Pro directly by clicking on the implementation. Synplify Pro allows multiple implementation views at the same time.

3.6 Step 6: Launch and View the New Implementation (Optional)

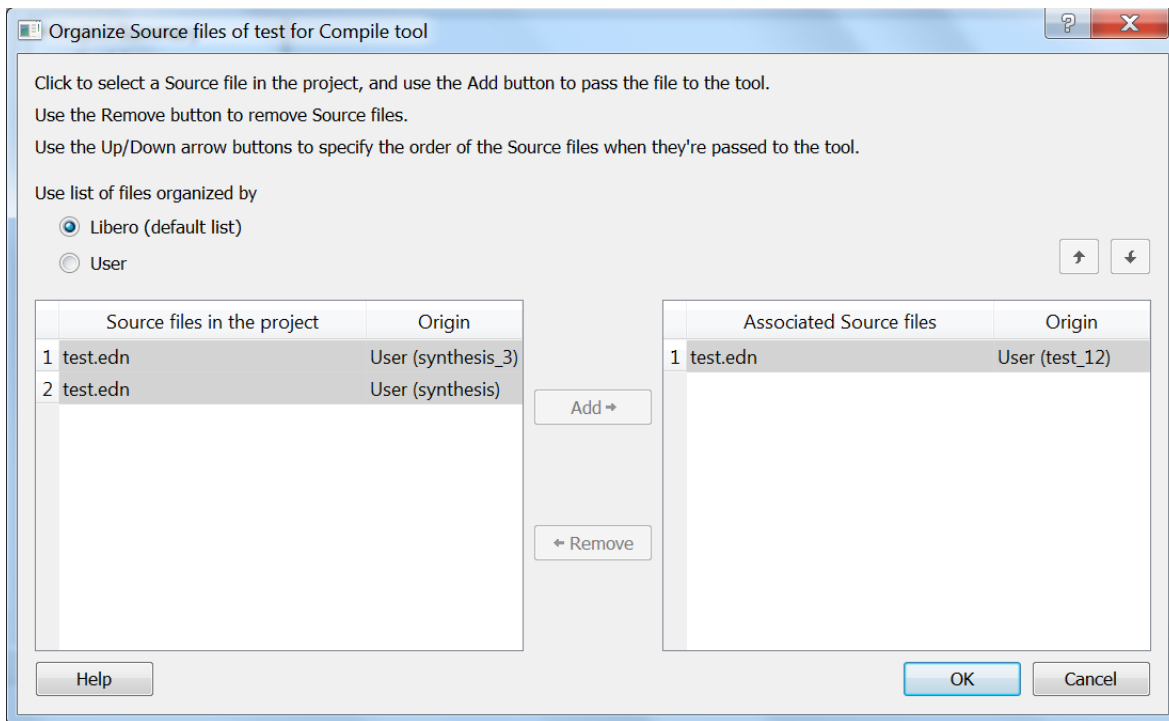
You can re-launch Identify from Synplify Pro by right-clicking on **Identify implementation** and selecting **Launch Identify** from the list as shown in the following figure.

Figure 14 • Re-Launching Identify in Synplify Pro



3.7 Step 7: Organize Instrumented EDIF in Libero SoC

1. The Libero SoC tool automatically selects the *.edn netlist from the Identify implementation folder (synthesis_1) and you can run **Compile** and Place and Route for it.
2. Alternatively, if you have multiple netlist in the design, they can be organized as shown in the following figure.
In the Libero Design Flow window, right-click on **Compile** and select Organize input files and Organize Source files.

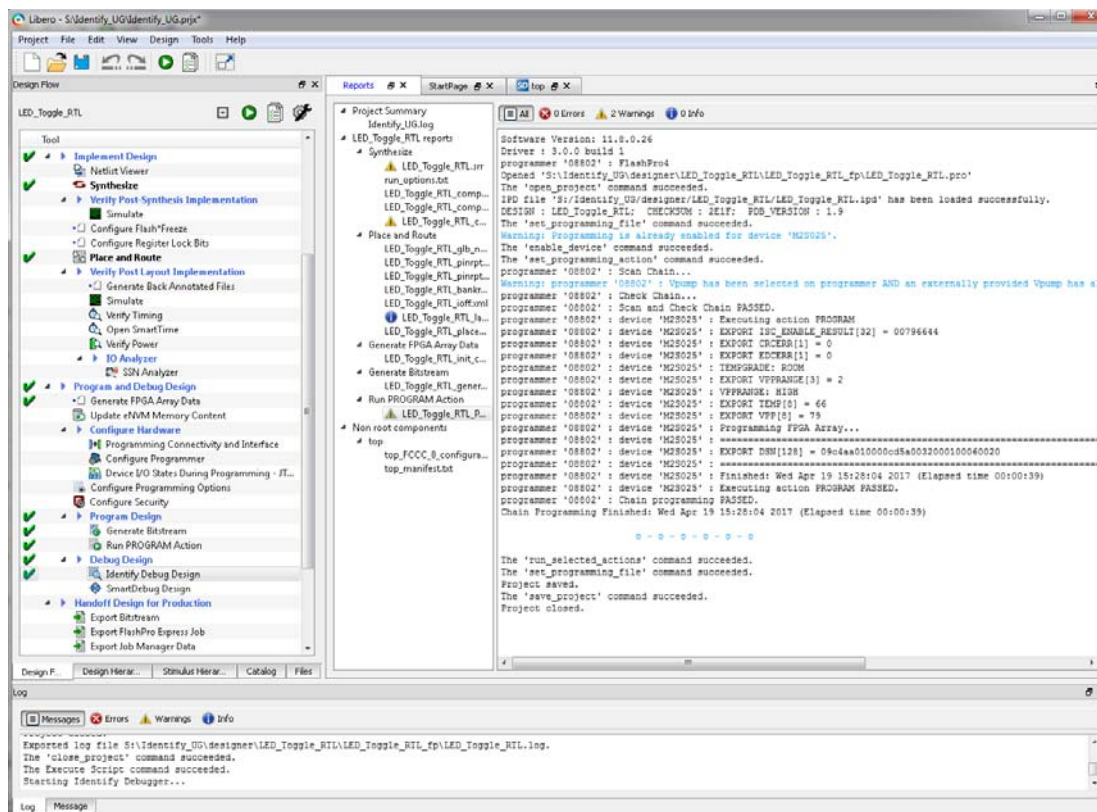
Figure 15 • Importing .EDN File in Libero SoC

As shown in the preceding figure, you can check/add an instrumented *.edn file.

3. After assigning/checking the input file, you can generate the programming file by clicking the **Generate programming** file.

4. Program the part by running Programming action as shown in the following figure.

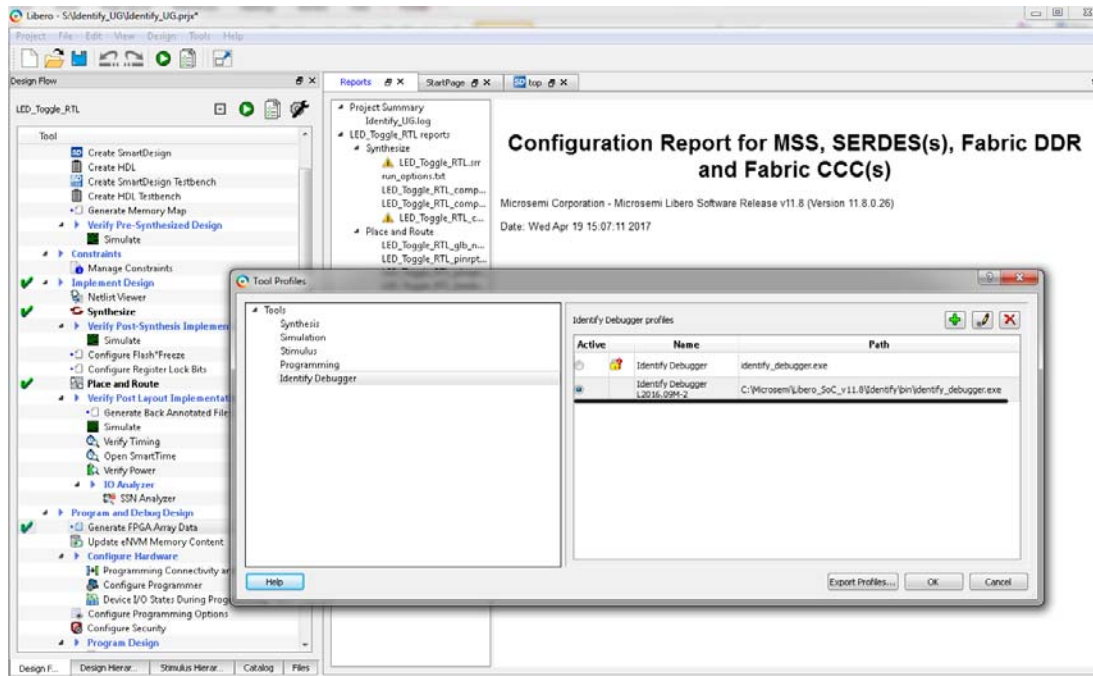
Figure 16 • Libero Compile, Place and Route, Generate Programming, and Run Program Action



3.8 Step 8: Launch Identify Debugger

1. After programming the device with the instrumented design, you can invoke the Identify Debugger through the Libero Design Flow.
2. Click **Identify Debug Design** and click **Edit Profiles** as shown in the following figure.

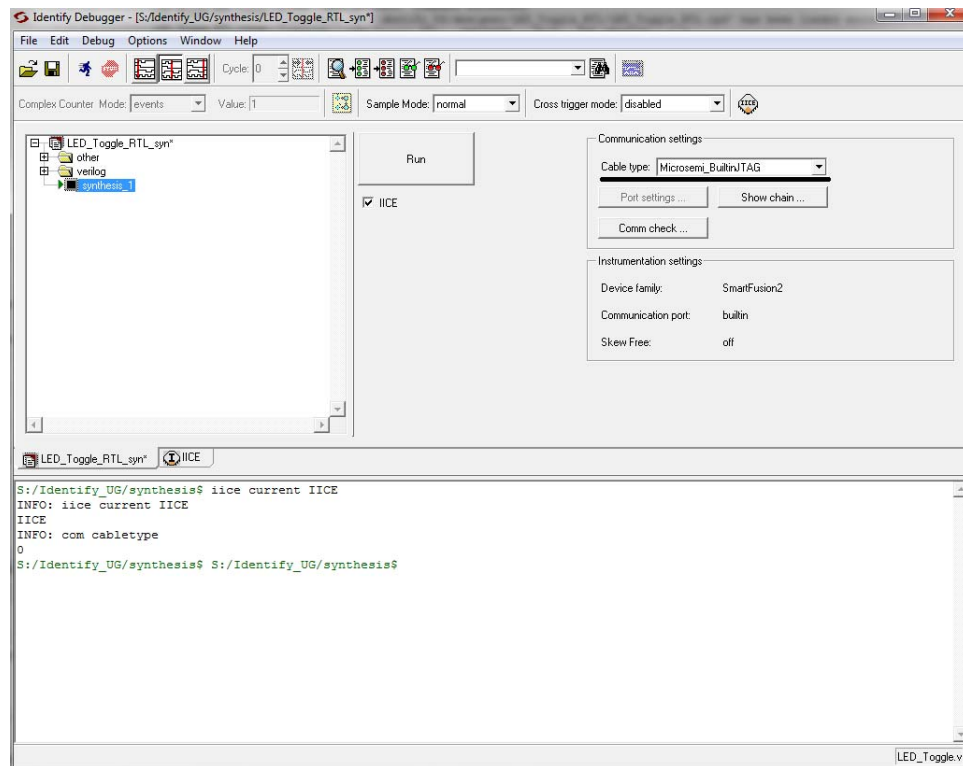
Figure 17 • Set Identify Debugger Profile within Libero



3. The created *.prj file by SynplifyPro/Identify Instrumentor is opened in the Identify Debugger

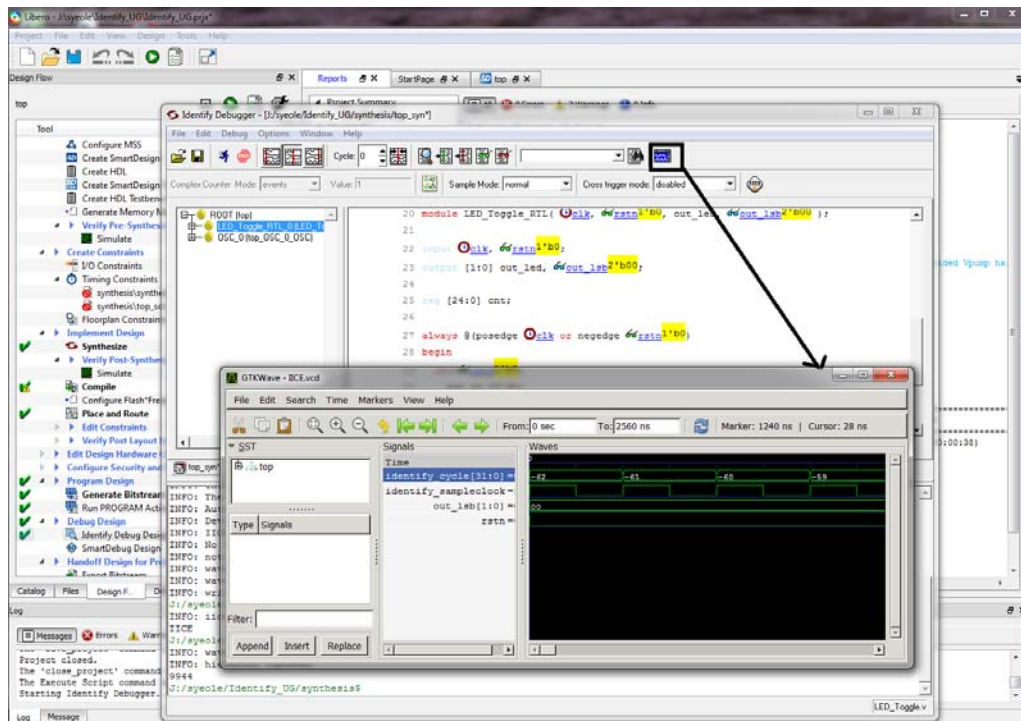
4. Set the cable type as **Microsemi_BuiltinJTAG** as shown in the following figure.

Figure 18 • Set Identify Debugger Profile within Libero



- Click the **Run** button. The Identify debugger stops when a trigger condition occurs and displays the signal values in the GTKWaveform Viewer as shown in the following figure.

Figure 19 • Identify Debugger



Note: For using different features and options within the Identify Instrumentor and Debugger tool, see [Synopsys Identify ME Tutorial](#).