# 1

# Logic Design With Soft Modules

## Objectives

- Learn how to start a FastChip project.
- Become familiar with the FastChip library of soft modules.
- Learn how to instantiate and customize soft modules.
- Learn how to interconnect soft modules to form larger logic circuits.
- Learn how to connect a logic circuit to external signals through programmable I/O.
- Learn how to bind a logic circuit to a Triscend CSoC.
- Learn how to download a logic circuit to the XESS CSoC Board.
- Learn how to test a logic circuit with the XESS CSoC Board.

## Programmable Logic Resources in the CSoC

Your introduction to the Triscend CSoC begins by doing some combinatorial and sequential logic designs. The areas of the CSoC you will use are:

- the *configurable system logic* (CSL) matrix,
- *the programmable input/output* (PIO) pins,
- the *clock control* and distribution.

These areas are highlighted in Figure 1.

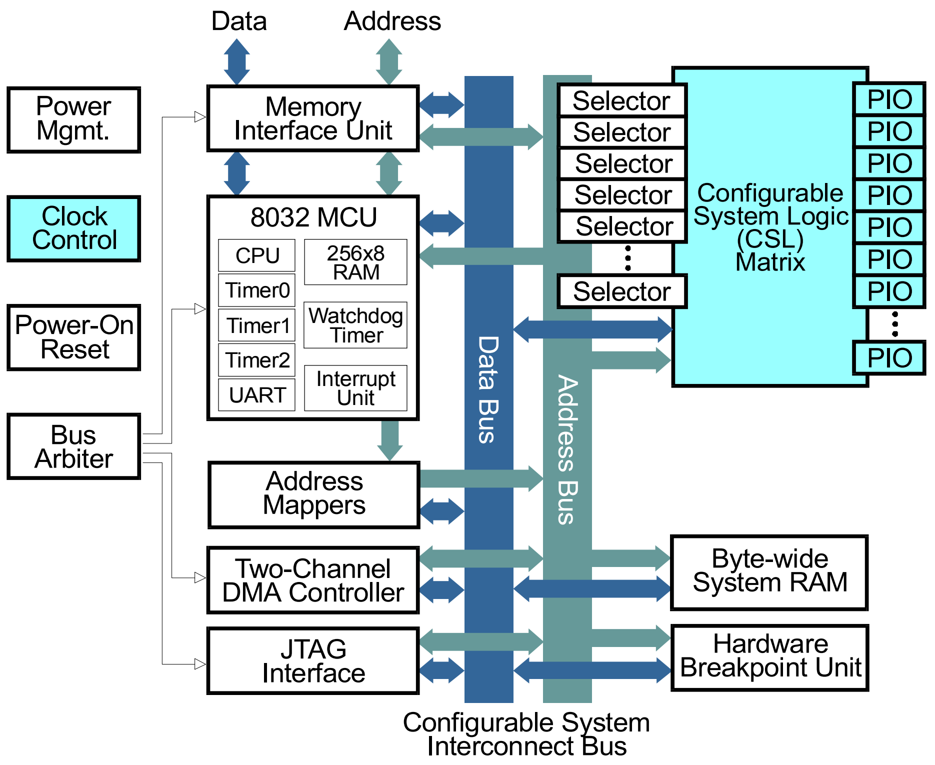


**Figure 1: Areas of the Triscend CSoC used in the designs in this chapter.**

## *The CSL*

The CSL consists of a matrix of cells interconnected by wiring and switches (Figure 2). Each cell contains combinatorial and sequential logic circuitry. The cells are configured to perform various logic operations and then the switches are programmed to interconnect the cells into a larger logic circuit.

Each cell contains a *look-up table* (LUT) for implementing combinatorial logic and an edge-triggered D flip-flop for use in sequential logic. These elements can be used independently or in concert.

The LUT implements any four-input logic function. The LUT can be envisioned as a 16×1 RAM that is loaded with the truth-table for a given logic function. The four inputs serve as an address that selects the location in the RAM that stores the logic value from the same location in the truth-table. The LUT is augmented with dedicated carry-chain circuitry so it can also act as a single-bit adder, subtractor, or multiplier.

The RAM in the LUT can also be used as a 16×1 RAM or ROM when you need more than the single bit of storage provided by the flip-flop. It also has a mode where it can be used as an eight-bit shift register.

Each cell contains additional circuitry that lets you efficiently combine cells to build larger functions. Any five-input logic function and many functions of six to nine inputs can be implemented with just two cells. You can also build 32×1 RAMs or ROMs with just two cells, or implement a 16×1 dual-port RAM.

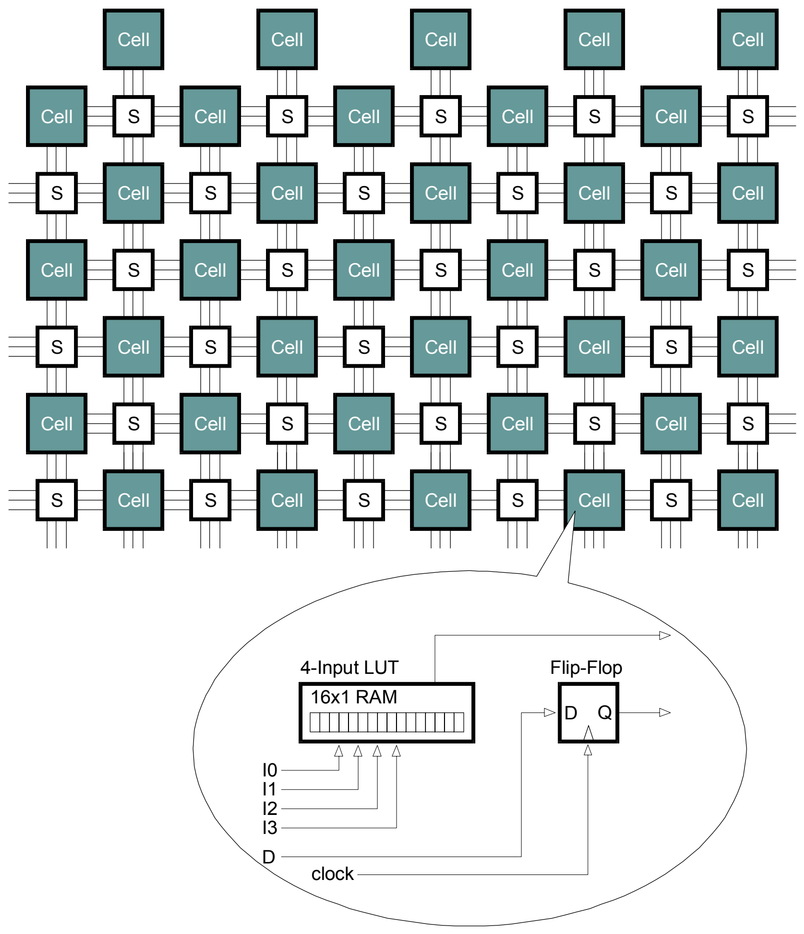


**Figure 2: A simplified view of the CSL.**

## *The PIO Pins*

The PIO pins let the circuits built in the CSL communicate with the environment outside the CSoC. The structure of an individual PIO is shown in Figure 3. Each PIO provides both an input and an output channel. The input channel carries logic signals from the pad into the CSL. The output channel transmits signals from the CSL to the outside world under the control of a signal that enables the output driver to the pad. The input, output, and output-enable signals can change freely or they can be registered so they change only on a clock edge.

In addition to the plain-vanilla I/O circuitry, the PIO also contains circuitry for the following functions:

- Although the Triscend CSoC is powered by +3.3V, the PIO can safely drive and accept logic signals to and from 5V logic chips (i.e., it is *5V-tolerant*).
- The polarity of the output and output-enable signals can be reversed with programmable inverters.
- The strength of the output driver can be configured in a low-power mode which sources a maximum of 4 mA, or a higher-power mode that sources up to 12 mA.
- A programmable pull-up or pull-down resistor can be connected to the PIO pad to hold the input at a stable logic 1 or 0 level. You can also activate a BusMinder™ circuit which maintains the last valid logic level on the input pad.
- Programmable hysteresis on the input channel can filter out up to 150 mV of noise on an input signal.
- A small delay can be switched into the path of the input register to reduce the hold time for the flip-flop to zero or less.

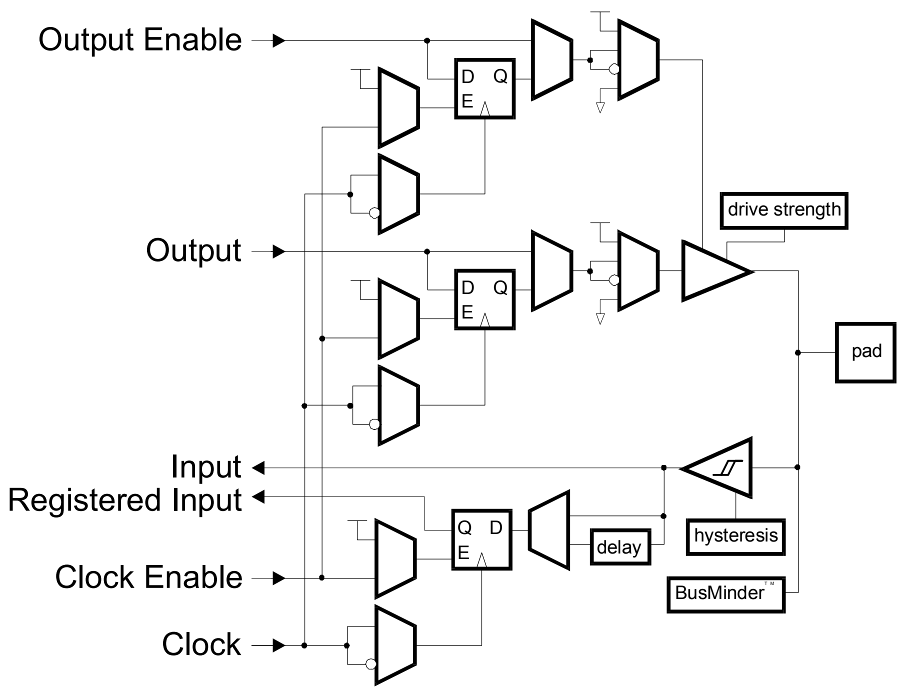


**Figure 3: Programmable I/O circuitry.**

### *Clock Control and Distribution*

The flip-flops in the CSL cells and the PIO pins can receive their clock signals over a network of low-skew wiring. Low-skew clocks are necessary to avoid timing violations. When a timing violation exists, a clock edge can change a flip-flop output before it is clocked into the next flip-flop by a delayed version of the clock edge. If the clock skew is small, the flip-flops will all switch at the same time and the logic values will be reliably transferred from one flip-flop to another.

The clock network can be driven from several sources:

- A ring oscillator within the CSoC which generates a frequency in the range 5–20 MHz,
- An external oscillator that drives the BCLK input of the CSoC,
- An external crystal attached between the BCLK and XTALIN pins of the CSoC,
- One or more of six global clock buffers that accept clocks from the CSoC pins or which are generated by logic circuits within the CSL.

You have seen the various features and capabilities of the CSL, PIO pins, and clock control circuitry. Now you will use some of these elements to construct some simple logic designs. You will find the FastChip software makes it easy to configure the various options while you build your logic circuits.

## Design 1.1 - A Simple Seven-Segment LED Decoder

You will start with a simple design: a decoder which displays a four-bit hexadecimal value on a seven-segment LED digit (Figure 4). The four inputs are driven by four of the DIP switches on the CSoC Board and the seven outputs drive the LED digit on the same board. The decoder circuit is programmed into the CSL matrix of the CSoC chip.

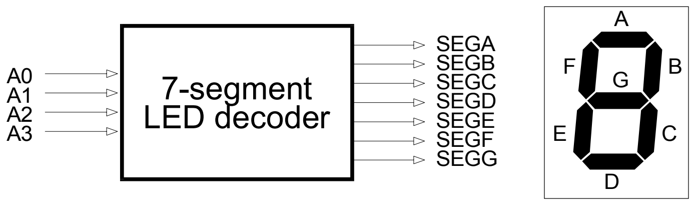
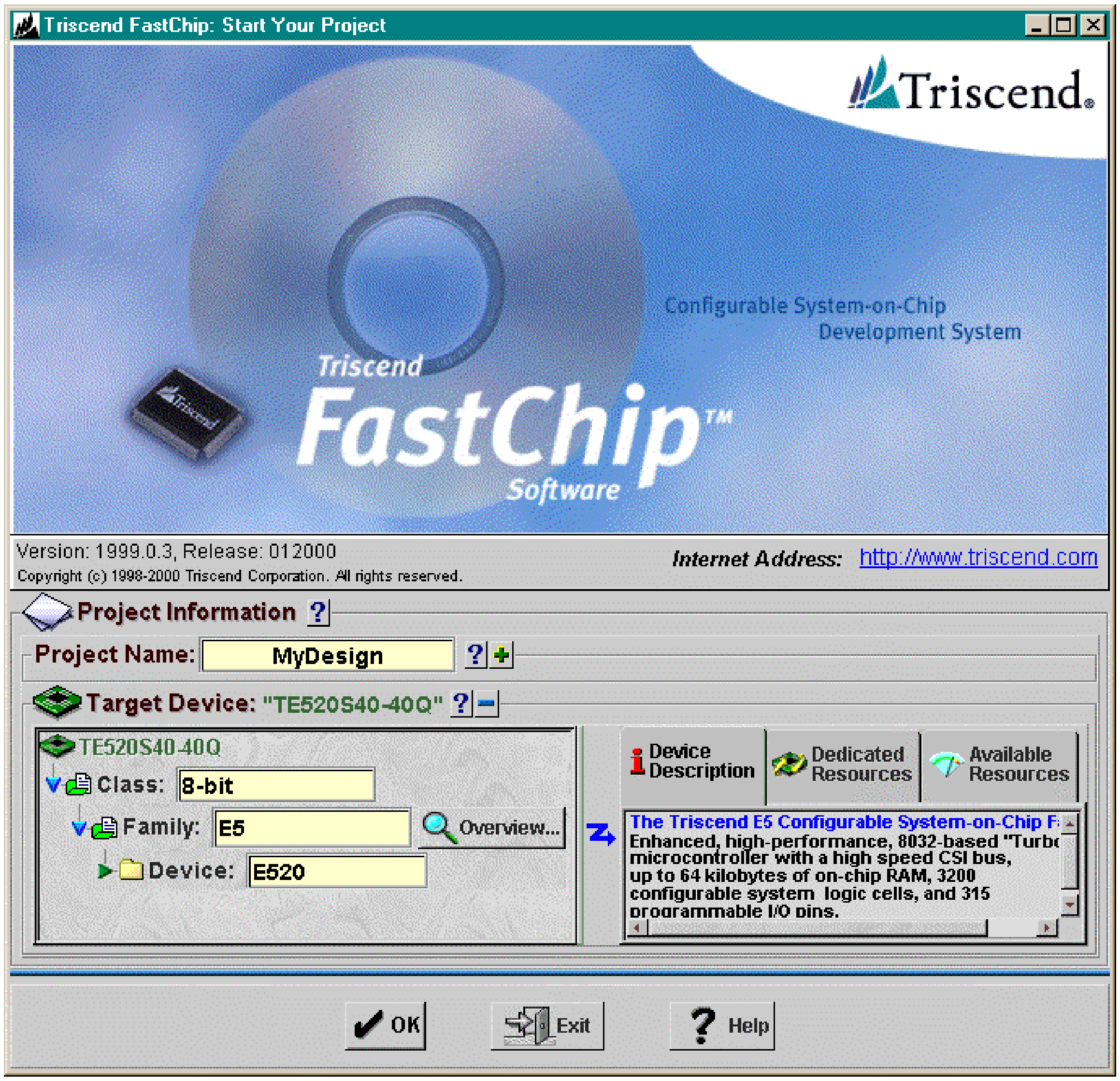


**Figure 4: Block diagram of a seven-segment LED decoder and the labeling of LED segments in an LED digit.**

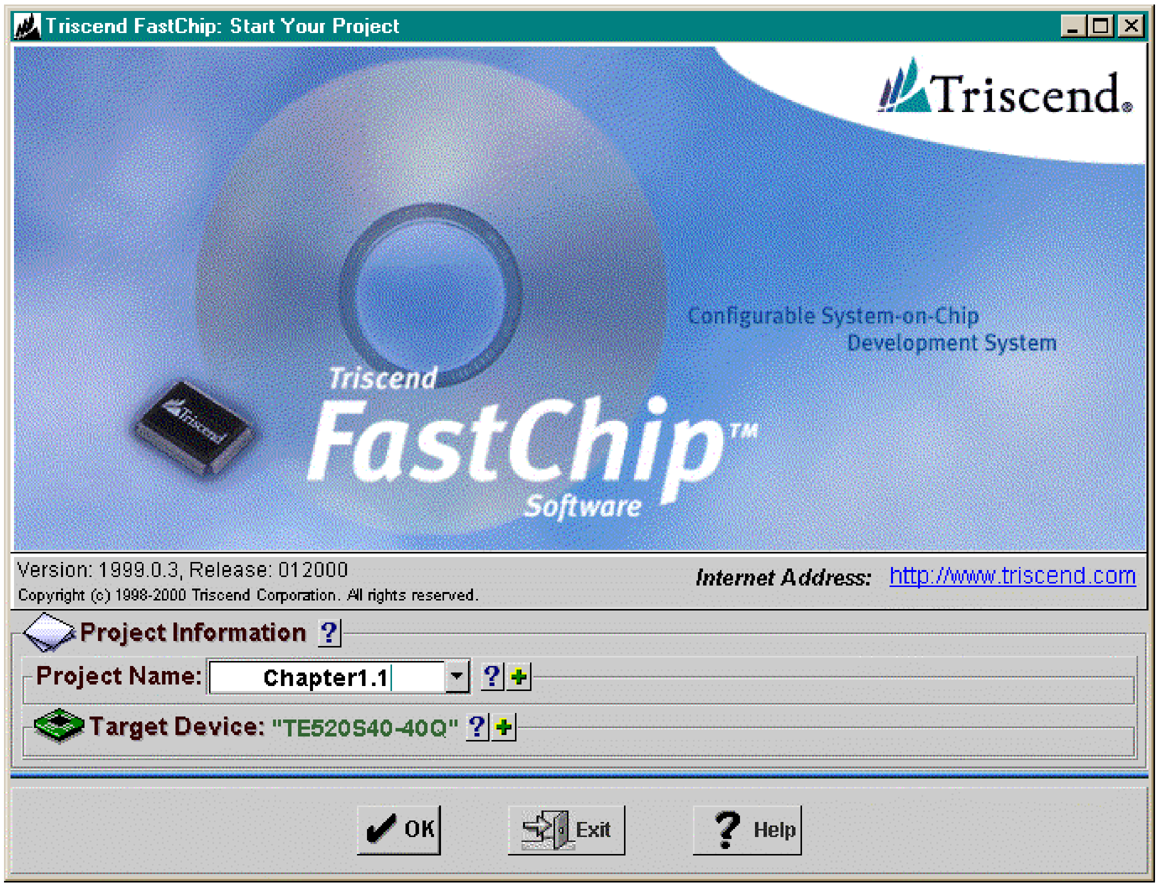
***Starting the LED Decoder Project***

Start the FastChip software by double-clicking the Triscend FastChip icon on your desktop. The main screen appears as shown below.


Triscend FastChip: Start Your Project
Triscend.
Configurable System-on-Chip Development System
Triscend FastChip™ Software
Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.
Internet Address: http://www.triscend.com
Project Information
Project Name: MyDesign
Target Device: "TE520S40-40Q"
TE520S40-40Q
Class: 8-bit
Family: E5
Overview...
Device: E520
Device Description
Dedicated Resources
Available Resources
OK
Exit
Help

The Triscend E5 Configurable System-on-Chip F
Enhanced, high-performance, 8032-based "Turbo
microcontroller with a high speed CSI bus,
up to 64 kilobytes of on-chip RAM, 3200
configurable system logic cells, and 315
programmable I/O pins.

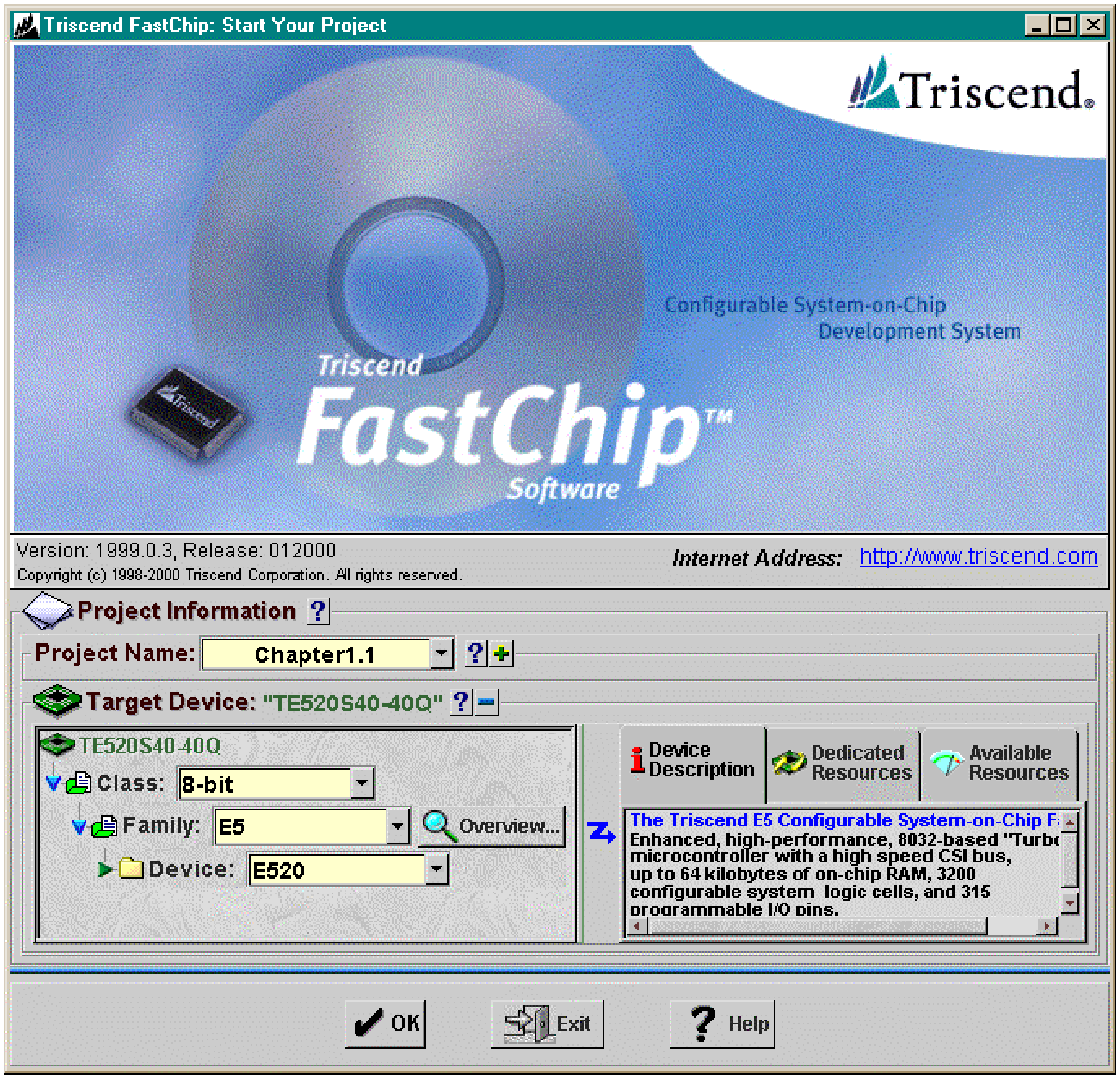
In the **Start Your Project** window, type the name for your project in the Project Name field. I typed the name `Chapter1.1` to signify this is the first design in Chapter 1. You can select a different name if you choose.


Triscend FastChip: Start Your Project

Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.
Internet Address: http://www.triscend.com

Project Information
Project Name: Chapter1.1
Target Device: "TE520S40-40Q"

After setting the project name, you have to select the type of Triscend CSoC chip you plan to use for this design. The XESS CSoC Board uses a TE505 chip in a 128-pin LQFP package that runs at a maximum speed of 25 MHz. You will enter this information into the Target Device area of the FastChip project window. If the fields for entering this information are not visible, click on the ➕ icon in the Target Device area. This will expand the Target Device area so the individual fields are visible as in the window shown below.


Triscend FastChip: Start Your Project
Triscend.
Configurable System-on-Chip Development System
Triscend FastChip™ Software
Internet Address: http://www.triscend.com
Project Information
Target Device: "TE520S40-40Q"
Device Description
Dedicated Resources
Available Resources
Overview...
OK
Exit
Help

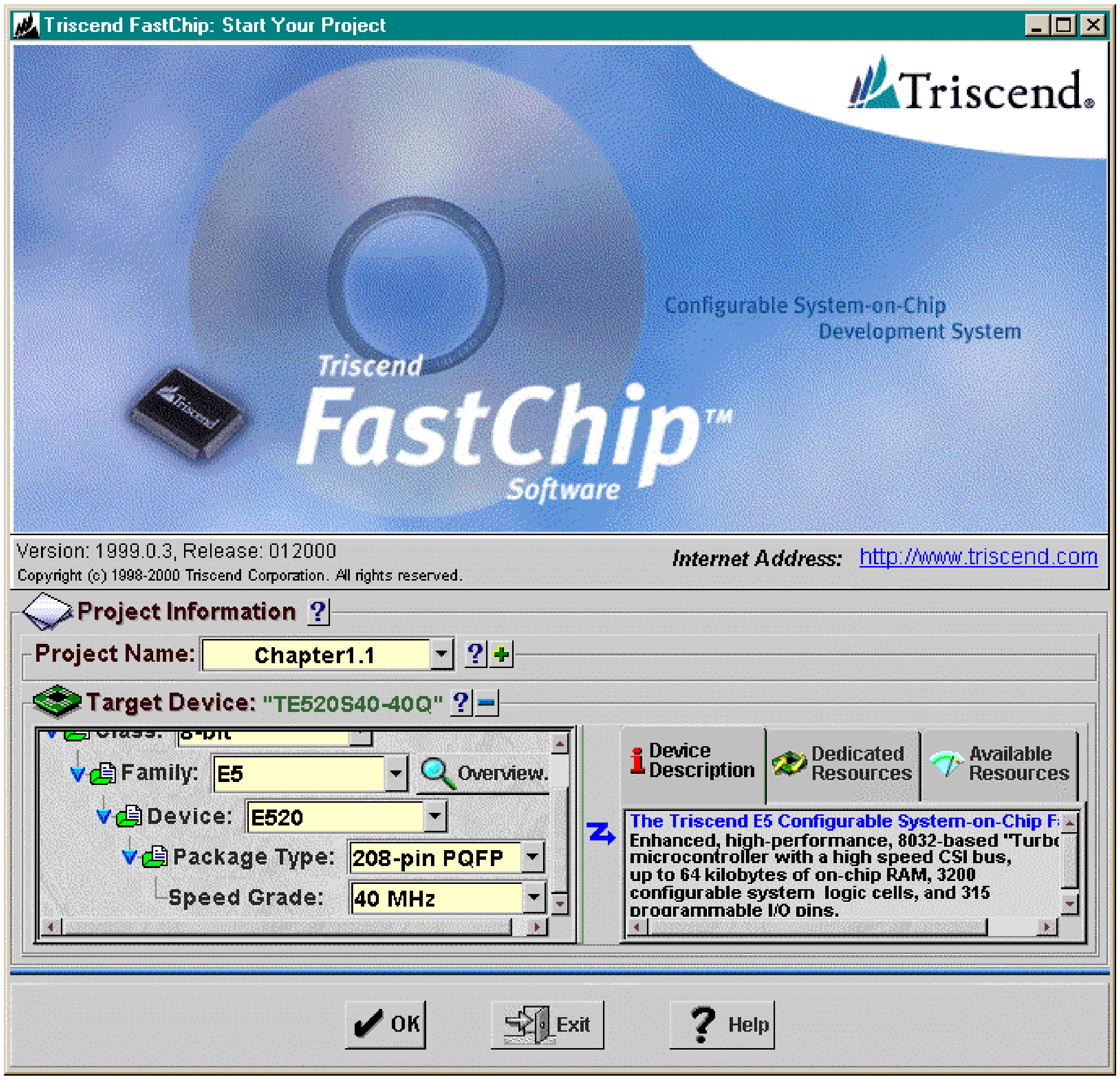
Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.

Project Name: Chapter1.1

TE520S40-40Q
Class: 8-bit
Family: E5
Device: E520

The Triscend E5 Configurable System-on-Chip F
Enhanced, high-performance, 8032-based "Turbo
microcontroller with a high speed CSI bus,
up to 64 kilobytes of on-chip RAM, 3200
configurable system logic cells, and 315
programmable I/O pins.

To further expand the Target Device area, click on the ▶ arrows so the window appears as shown below.

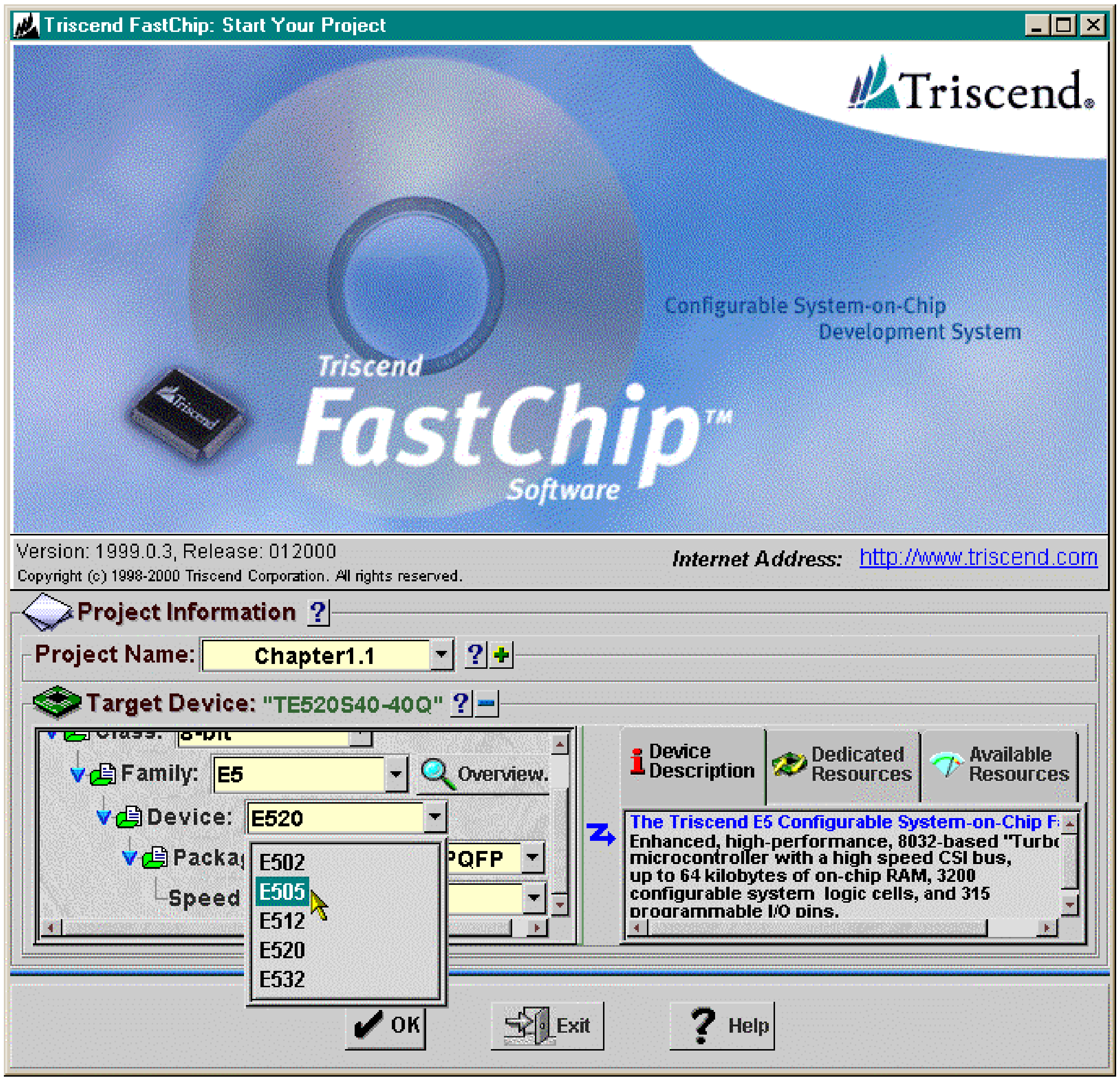

Triscend FastChip: Start Your Project
Triscend
Configurable System-on-Chip Development System
Triscend FastChip™ Software
OK
Exit
Help
Device Description
Dedicated Resources
Available Resources
Overview.

Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.
Internet Address: http://www.triscend.com

Project Information
Project Name: Chapter1.1
Target Device: "TE520S40-40Q"
Class: 8-bit
Family: E5
Device: E520
Package Type: 208-pin PQFP
Speed Grade: 40 MHz

The Triscend E5 Configurable System-on-Chip F
Enhanced, high-performance, 8032-based "Turb
microcontroller with a high speed CSI bus,
up to 64 kilobytes of on-chip RAM, 3200
configurable system logic cells, and 315
programmable I/O pins.

Click on the ▾ at the right of the Device field to display a scrollable list of potential target devices. Click on the E505 entry in the list to select the TE505 CSoC chip as your target device.



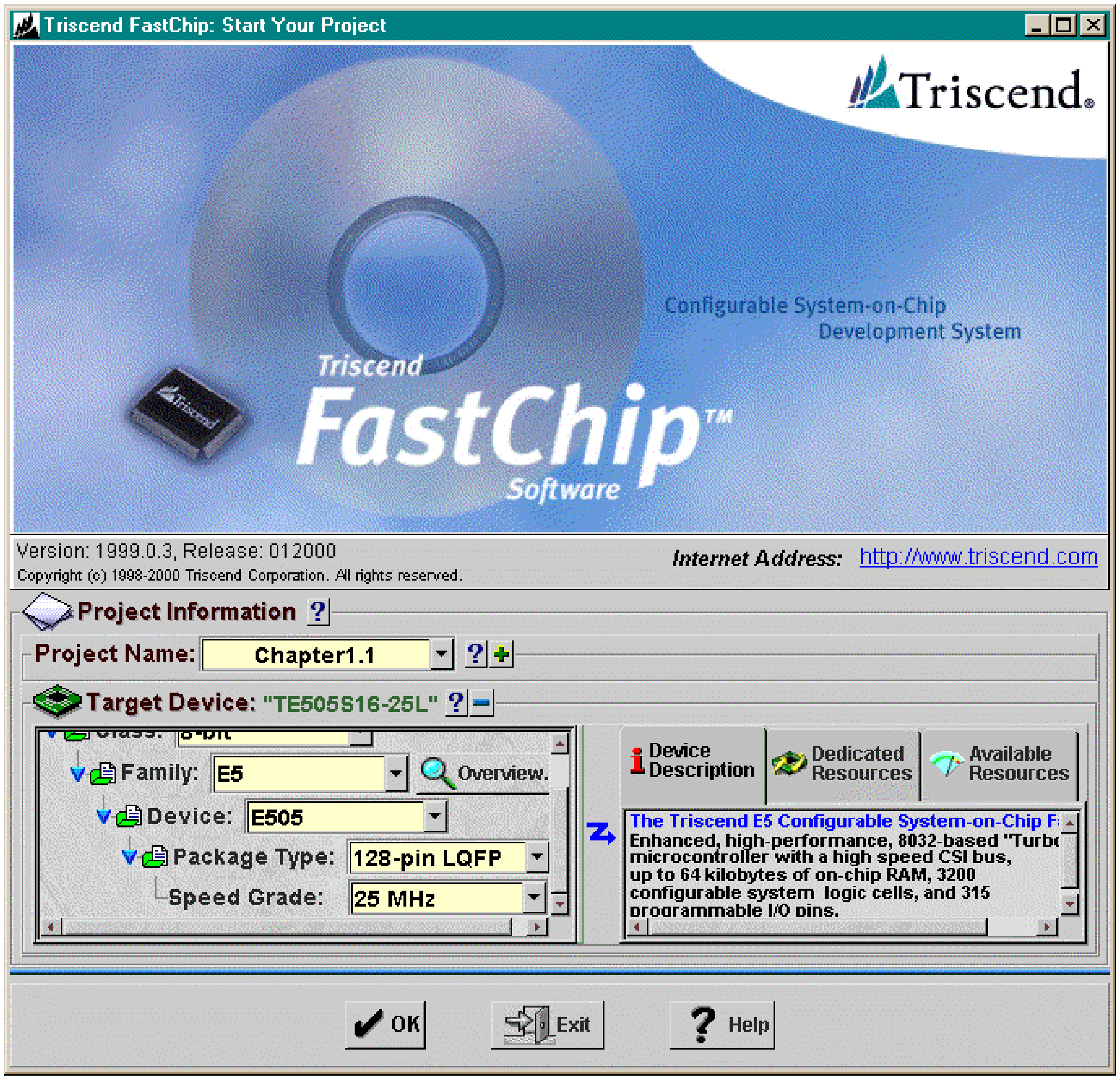
Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.
Internet Address: http://www.triscend.com

Project Name: Chapter1.1
Target Device: "TE520S40-40Q"
Class: 8-bit
Family: E5
Device: E520
Packa… …PQFP

E502
E505
E512
E520
E532

The Triscend E5 Configurable System-on-Chip F…
Enhanced, high-performance, 8032-based "Turb… microcontroller with a high speed CSI bus, up to 64 kilobytes of on-chip RAM, 3200 configurable system logic cells, and 315 programmable I/O pins.

Repeat this procedure to set the Package Type field to 128-pin LQFP and the Speed Grade field to 25 MHz as shown below.


Triscend FastChip: Start Your Project
Triscend.
Configurable System-on-Chip Development System
Triscend FastChip™ Software
Device Description
Dedicated Resources
Available Resources
Overview.
OK
Exit
Help

Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.
Internet Address: http://www.triscend.com

Project Information

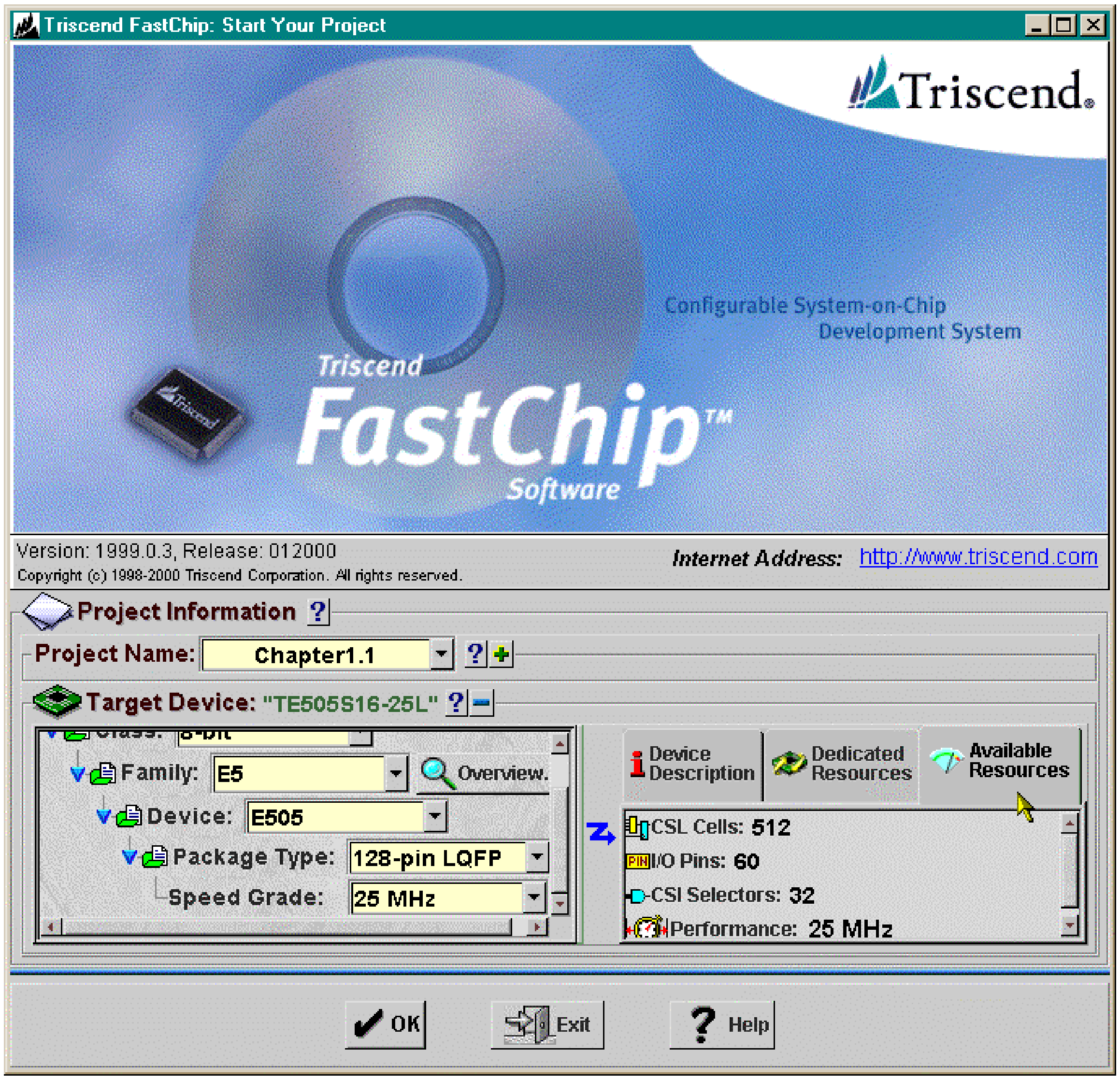
Project Name: Chapter1.1

Target Device: "TE505S16-25L"

Class: 8-bit
Family: E5
Device: E505
Package Type: 128-pin LQFP
Speed Grade: 25 MHz

The Triscend E5 Configurable System-on-Chip F
Enhanced, high-performance, 8032-based "Turb
microcontroller with a high speed CSI bus,
up to 64 kilobytes of on-chip RAM, 3200
configurable system logic cells, and 315
programmable I/O pins.

Now that you have specified your target device, you can look at what resources you have available for your design. Click on the Available Resources tab on the right-hand side of the Target Device area. The resources in the TE505 chip will be displayed below the tab. The TE505 has 512 cells in the CSL matrix. This is more than enough for your simple LED decoder. (A single CSL cell can compute any single-output function with four inputs or less.) There are also 60 I/O pins available for use in your design. The LED decoder only needs 4 inputs and 7 outputs, so there is plenty of I/O to go around. You also have 32 CSL selectors, but you won't need them for this design. (You probably don't even know what they are!) Finally, the performance is listed as 25 MHz which is just the speed grade for the chip.


Triscend FastChip: Start Your Project
Triscend.
Configurable System-on-Chip Development System
Triscend FastChip™ Software
OK
Exit
Help

Version: 1999.0.3, Release: 012000
Copyright (c) 1998-2000 Triscend Corporation. All rights reserved.
Internet Address: http://www.triscend.com

Project Information

Project Name: Chapter1.1

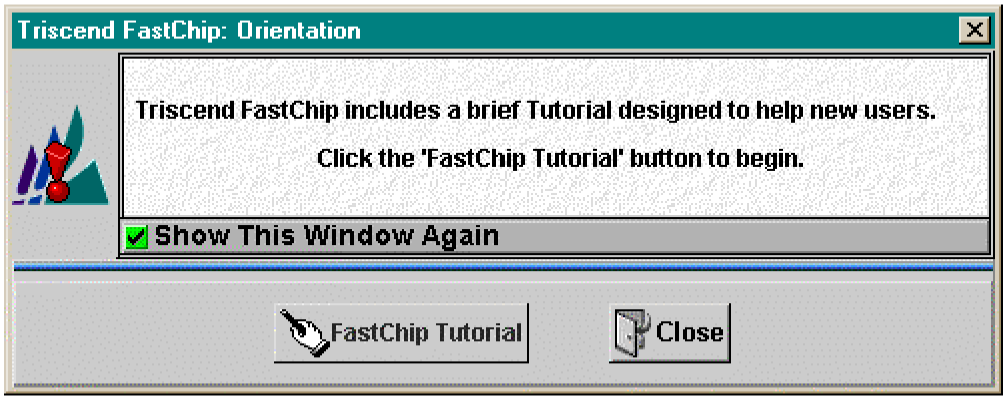
Target Device: "TE505S16-25L"

Family: E5
Device: E505
Package Type: 128-pin LQFP
Speed Grade: 25 MHz

Device Description | Dedicated Resources | Available Resources

CSL Cells: 512
I/O Pins: 60
CSI Selectors: 32
Performance: 25 MHz

Now that the target device information is specified, click on OK to proceed with your design. The window shown below may be displayed. Uncheck the box next to Show This Window Again and then click Close if you don't want to bother with the online tutorial.


Triscend FastChip: Orientation

Triscend FastChip includes a brief Tutorial designed to help new users.

Click the 'FastChip Tutorial' button to begin.

✓ Show This Window Again

At this point, your project design window should appear. The name of your design and the target device will be listed in the window title bar. There are five main areas within the window:

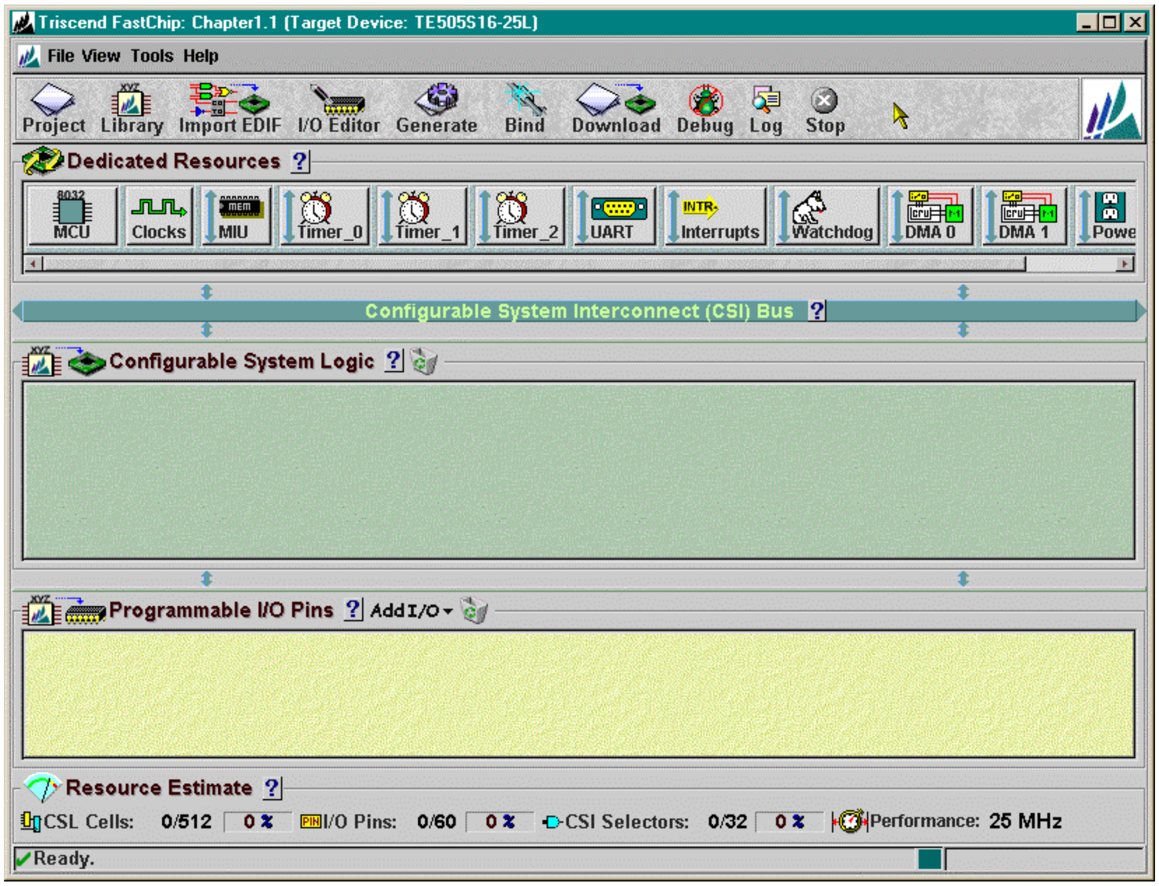
**Toolbar**: This is a row of icons which activate various tools and processes in the FastChip software.

**Dedicated Resources**: This is a row of icons that are used to set parameters controlling the fixed functional blocks of the CSoC.

**Configurable System Logic**: This is an area where the icons representing soft modules are placed to show they will be implemented using cells in the CSL matrix.

**Programmable I/O Pins**: This is an area where icons representing soft modules are placed to show they will implemented using PIO cells.

**Resource Estimate**: This area gives you a real-time estimate of how much of each CSoC resource remains for use in your design.



### *The Library of Soft Modules*

To begin actually designing the LED decoder, click on the Library toolbar icon. This will display a Triscend Library pane with a hierarchical list of soft modules along the left-hand side of the FastChip project window. The entries in the list partition the soft modules into the following categories:

**Archive**: Soft modules which have been replaced by improved versions are stored here. These modules are provided for backward compatibility with old designs.

**Displays**: Soft modules that format information for display are stored here. At this time there is only a single entry: a seven-segment LED decoder. (How lucky for us!)

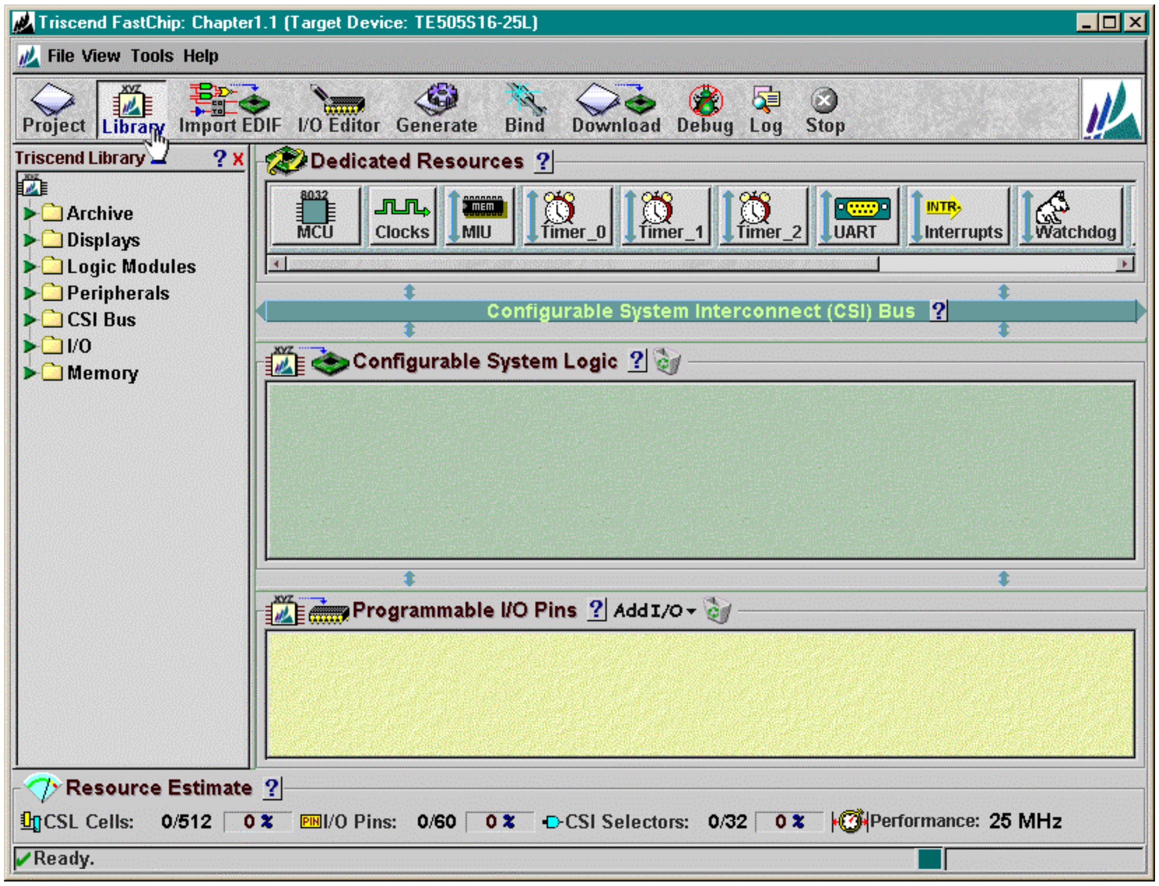
**Logic Modules**: ALUs, constants, counters, registers, FIFOs, and arbitrary four-input logic functions are some of the soft modules stored here.

**Peripherals**: Soft modules that act as peripherals to the 8032 microcontroller in the CSoC are stored here. This category is further divided into categories of soft modules such as UARTs, control registers, I/O ports, and pulse-width modulators.

**CSI Bus**: Soft modules for interfacing to the configurable system interconnect bus are stored here.

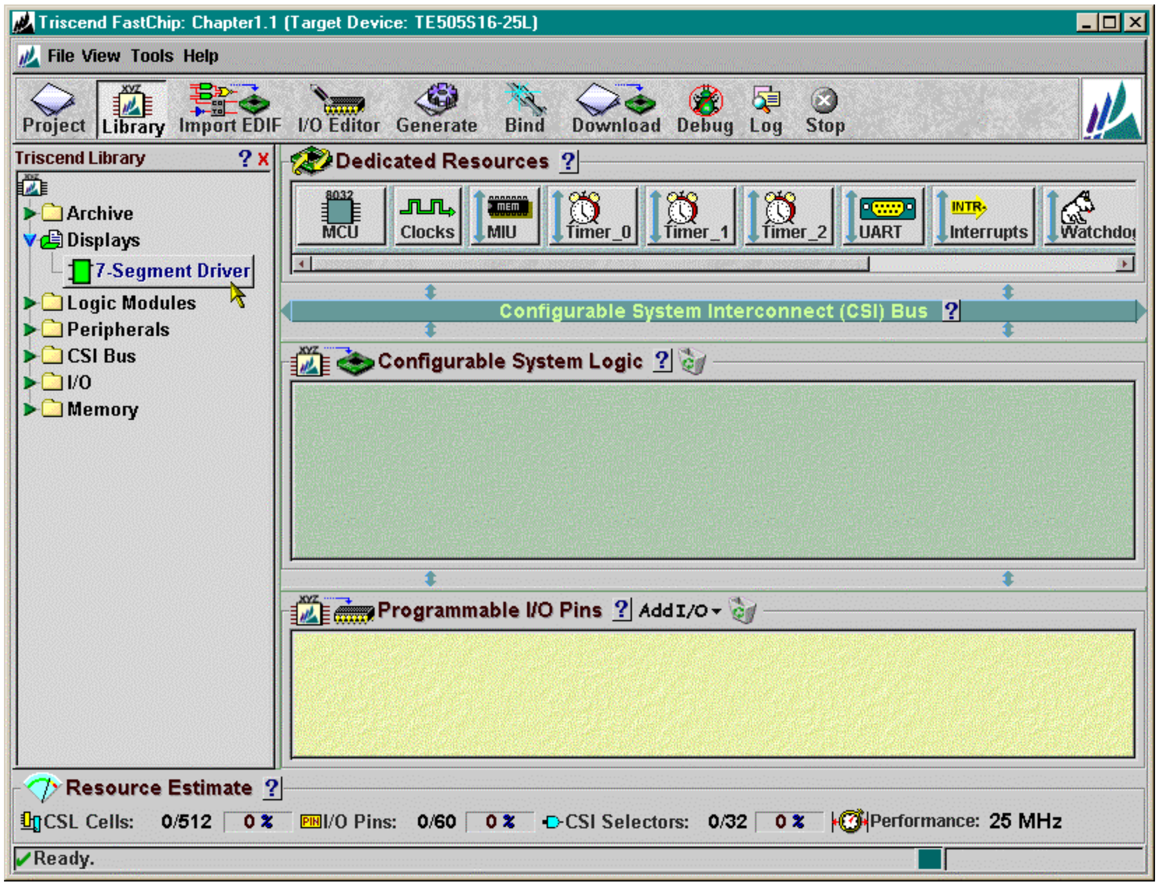
**I/O**: Soft modules for input, output, bidirectional, and three-state output pins are stored here.

**Memory**: Soft modules for memory elements are stored here. At this time, only a single 16-deep ROM element is available.

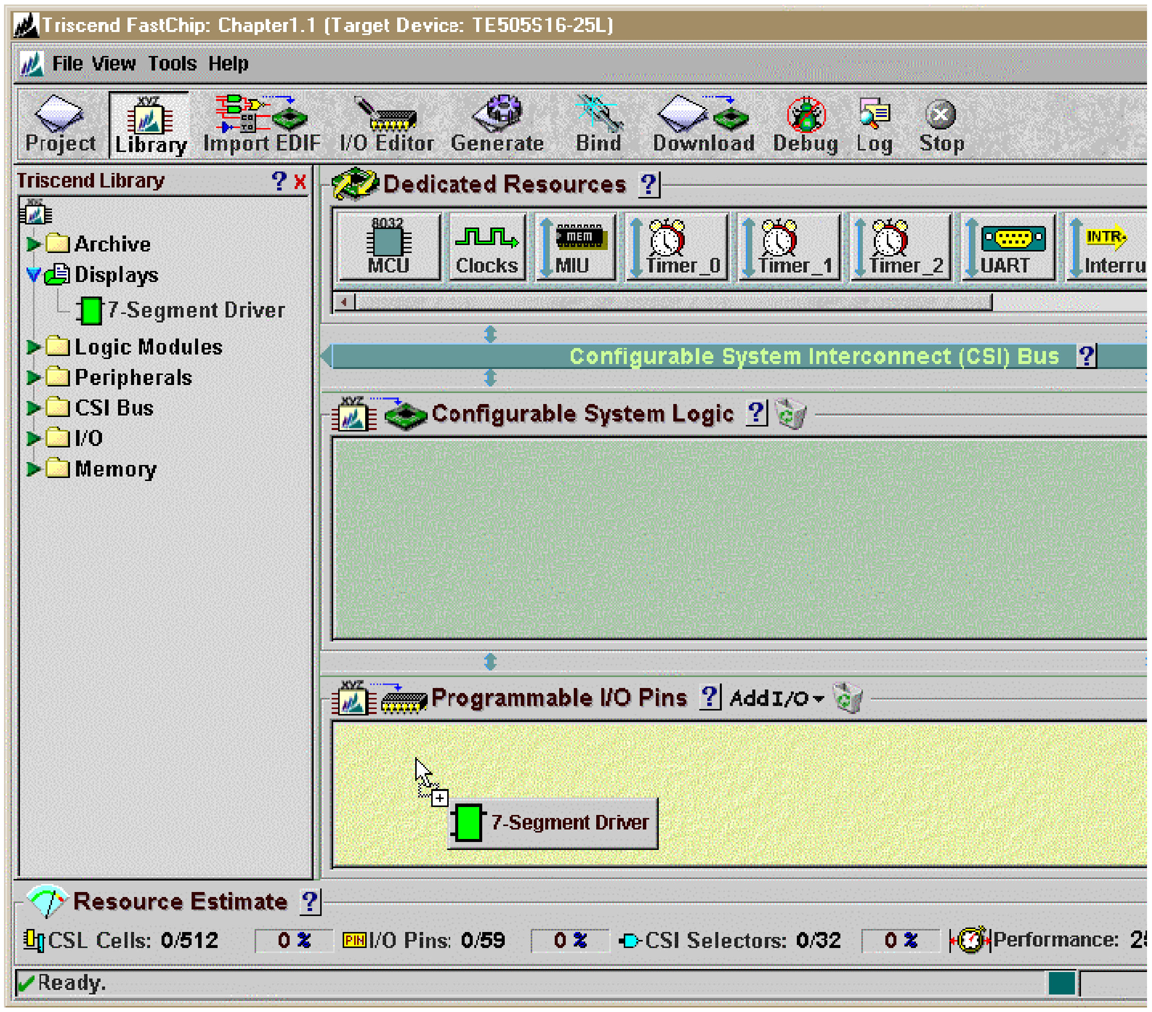


### *Instantiating Soft Modules*

Start by clicking on the ▶ arrow next to the Displays entry in the Triscend Library pane. This will expand the list entry and display the icon for the seven-segment LED decoder soft module.

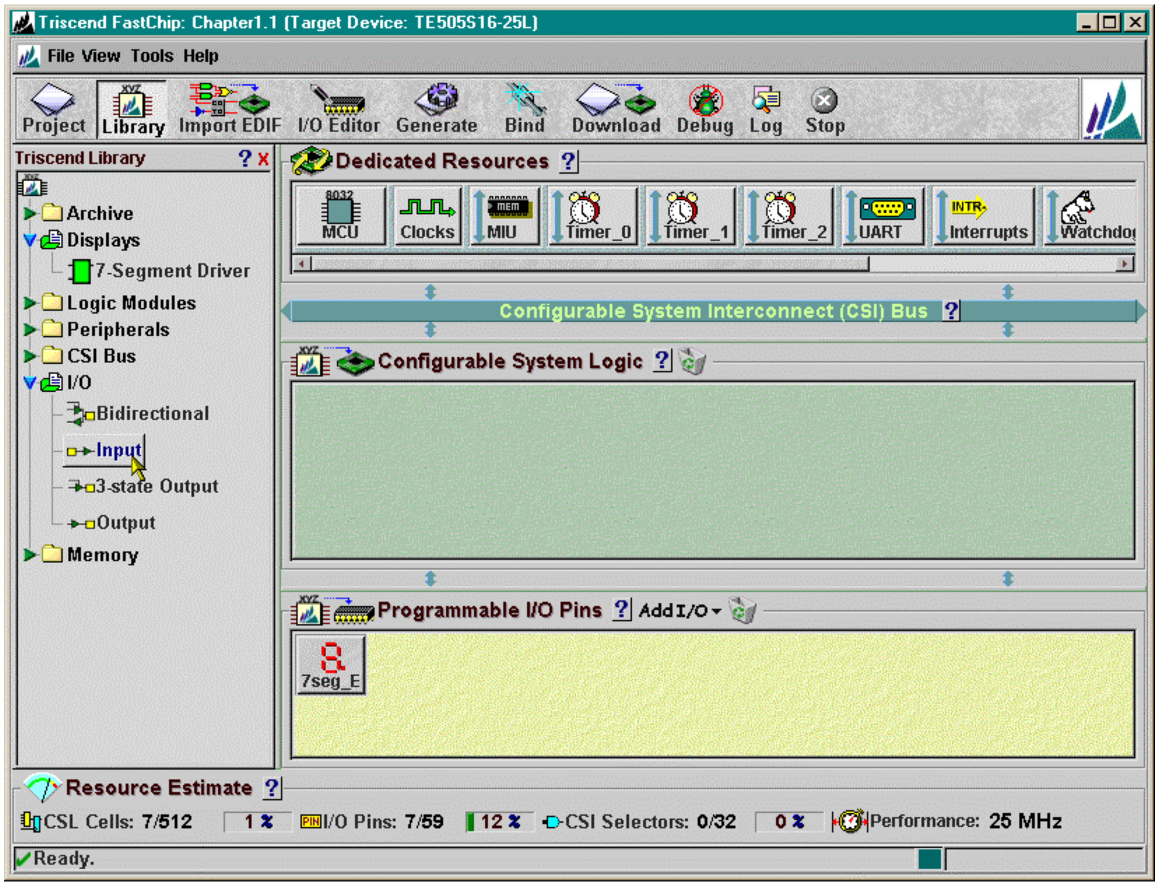


To instantiate the LED decoder, click-and-drag the 7-Segment Driver icon into the Programmable I/O Pins area as shown below. A ⊞ will appear whenever your mouse pointer passes over an area where the soft module can be dropped. (The LED decoder can be placed in the Programmable I/O Pins area, but not in the Configurable System Logic area.)



Once you release the mouse button over the Programmable I/O Pins area, the LED decoder will appear. After you do this, take note of the CSL Cells counter in the Resource Estimate area of the project design window. It now shows that seven of the 512 cells are used. These seven cells are the ones used by the LED decoder. The number of I/O pins used has also increased to seven for the same reason.

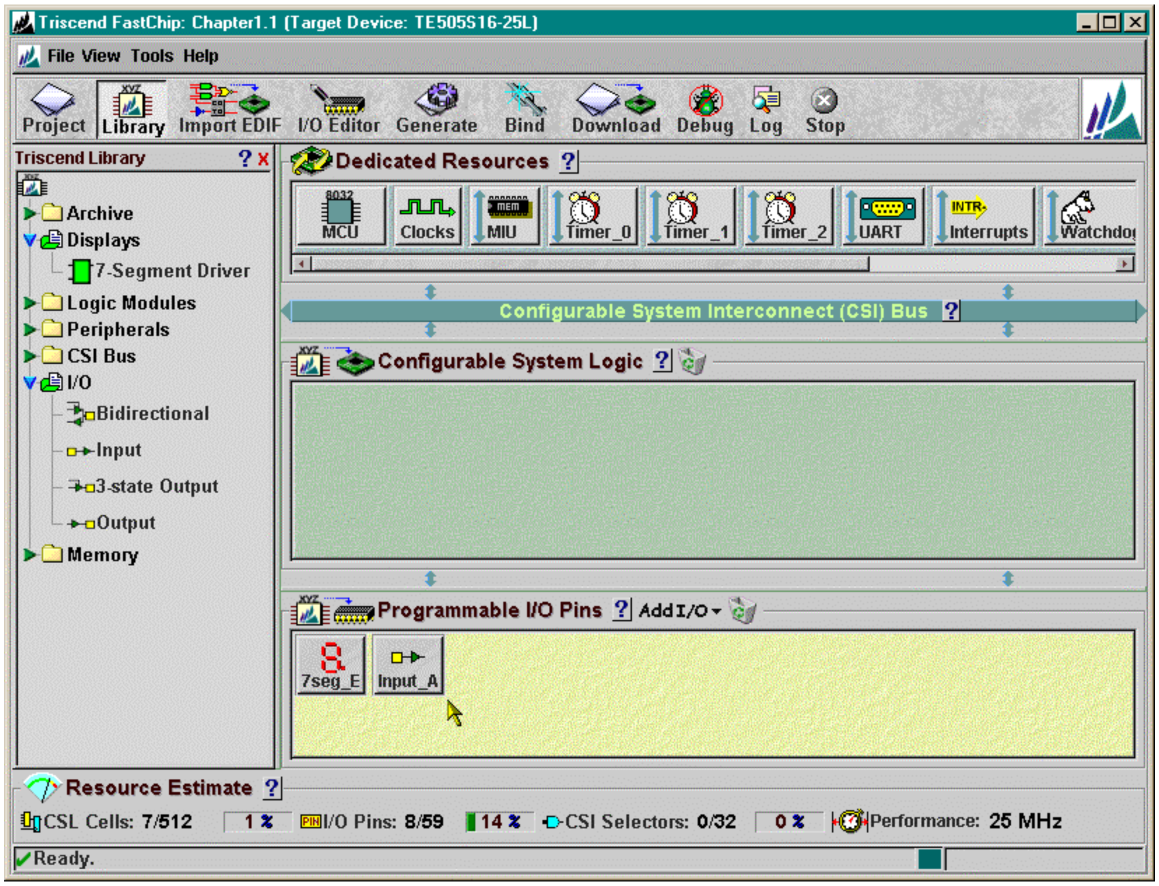
The LED decoder will drive the seven outputs to the LED digit on the CSoC Board, but you also need to get four inputs into your circuit. To do this, expand the I/O section of the Triscend Library pane and click-and-drag the Input soft module into the Programmable I/O Pins area.



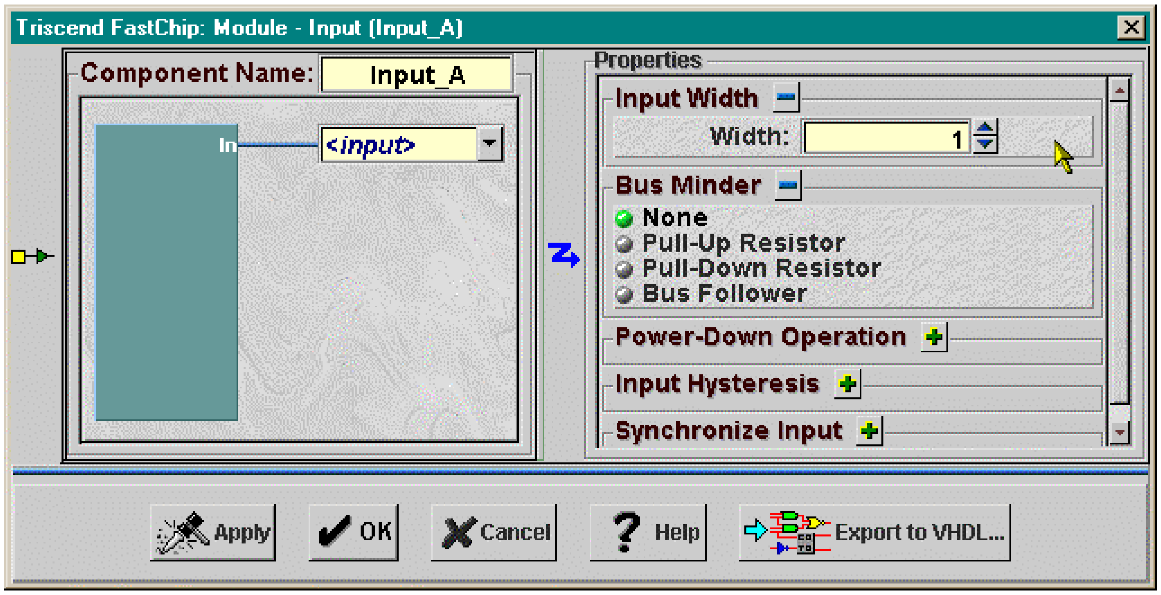
CSL Cells: 7/512 1 %  I/O Pins: 7/59 12 %  CSI Selectors: 0/32 0 %  Performance: 25 MHz

Ready.

Now your project design window should look like the one shown below. The LED decoder and input modules are instantiated as **7seg_E** and **Input_A**, respectively. But how do you connect them together so the inputs get to the LED decoder?



To start interconnecting the soft modules, click on the Input_A icon. The following window will appear. Using this window, you can change the name of the **Input_A** instantiation of the soft module, its properties, and the name of the signals coming out of it.


Triscend FastChip: Module - Input (Input_A)

Component Name: Input_A

In <input>

Properties

Input Width

Width: 1

Bus Minder

None [✓]

Pull-Up Resistor

Pull-Down Resistor

Bus Follower

Power-Down Operation

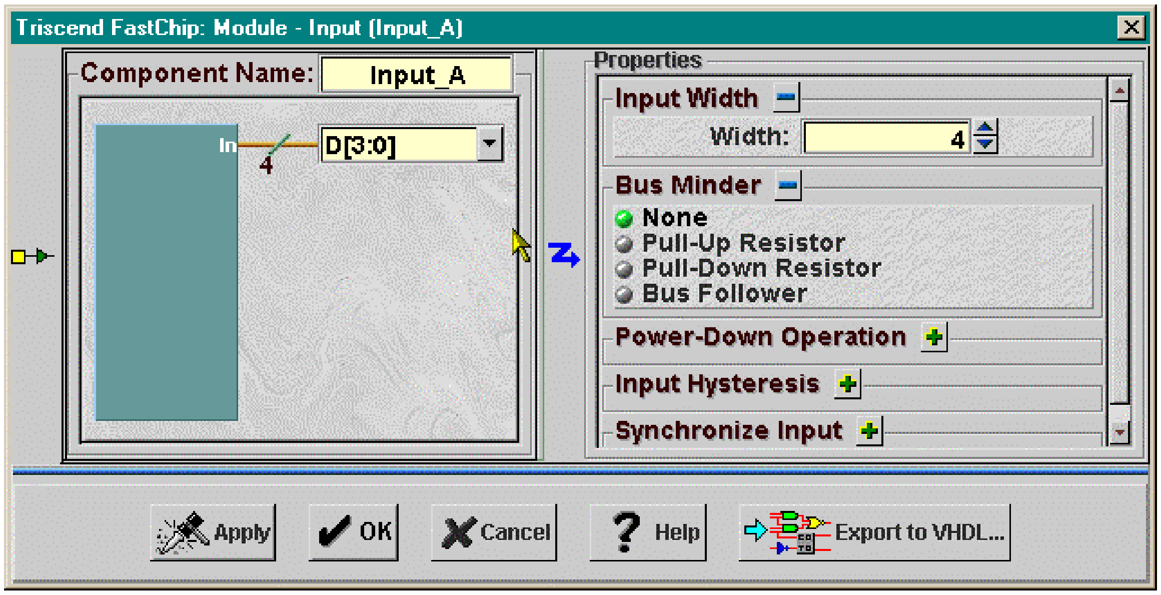
Input Hysteresis

Synchronize Input

First you need to expand the width of the input bus which is currently set at one. Click the ▲ up-arrow on the Input Width box until it is 4.

To name the inputs coming out of the **Input_A** module, click in the field that displays <input> and type D. Click anywhere outside the input name box and you will see the name you entered is expanded into a four-bit wide vector, **D[3:0]**. This is actually the name of the *outputs* coming from the **Input_A** module. The actual inputs to the **Input_A** module are driven by signals on the pins of the CSoC. Then these signals are passed out through the **D** outputs of the **Input_A** module and into the rest of the CSL where they can be used by other soft modules.

You are finished making modifications to the **Input_A** module, so click OK.


Triscend FastChip: Module - Input (Input_A)

Component Name: Input_A

D[3:0]

Properties

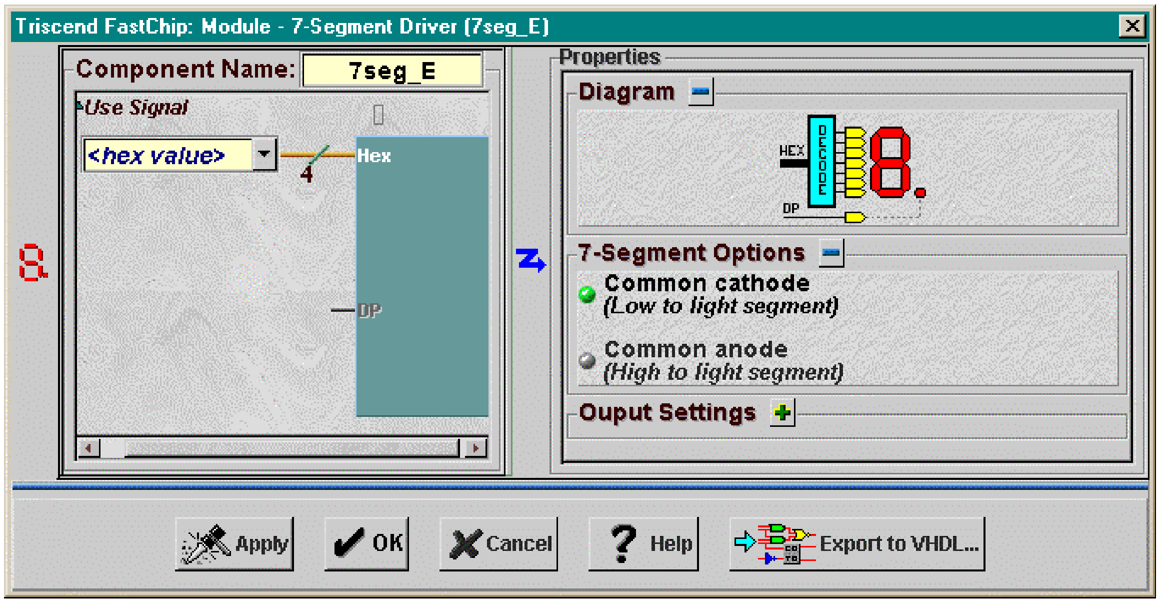
Input Width

Width: 4

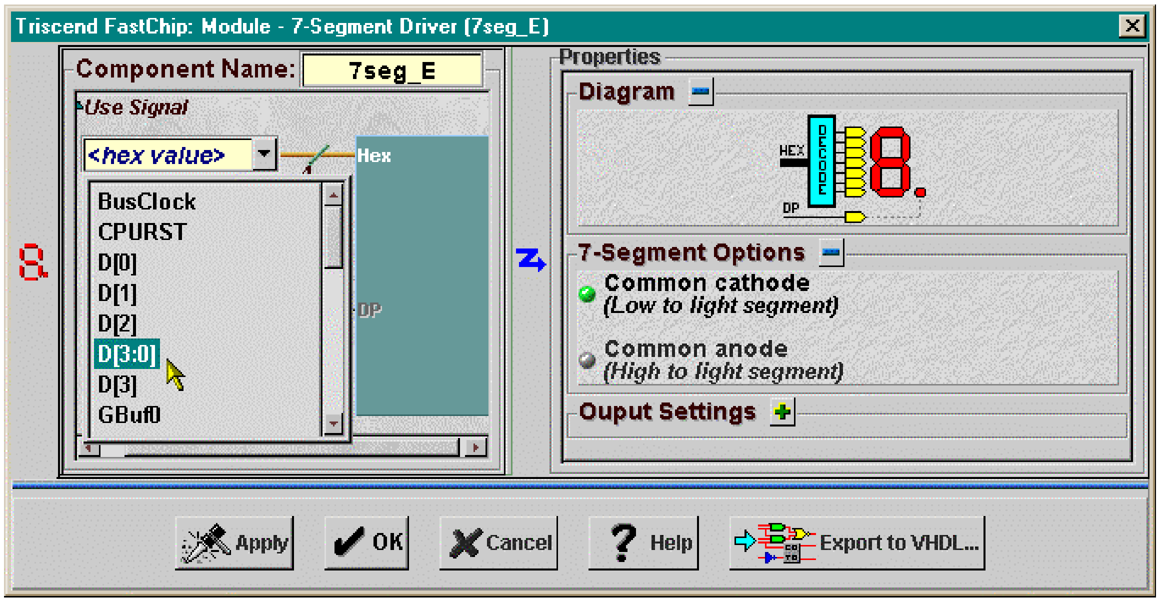
Bus Minder

None [✓]
Pull-Up Resistor
Pull-Down Resistor
Bus Follower

Now you have to make some modifications to the LED decoder. Click on the 7seg_E icon in the Programmable I/O Pins area so the following window appears.

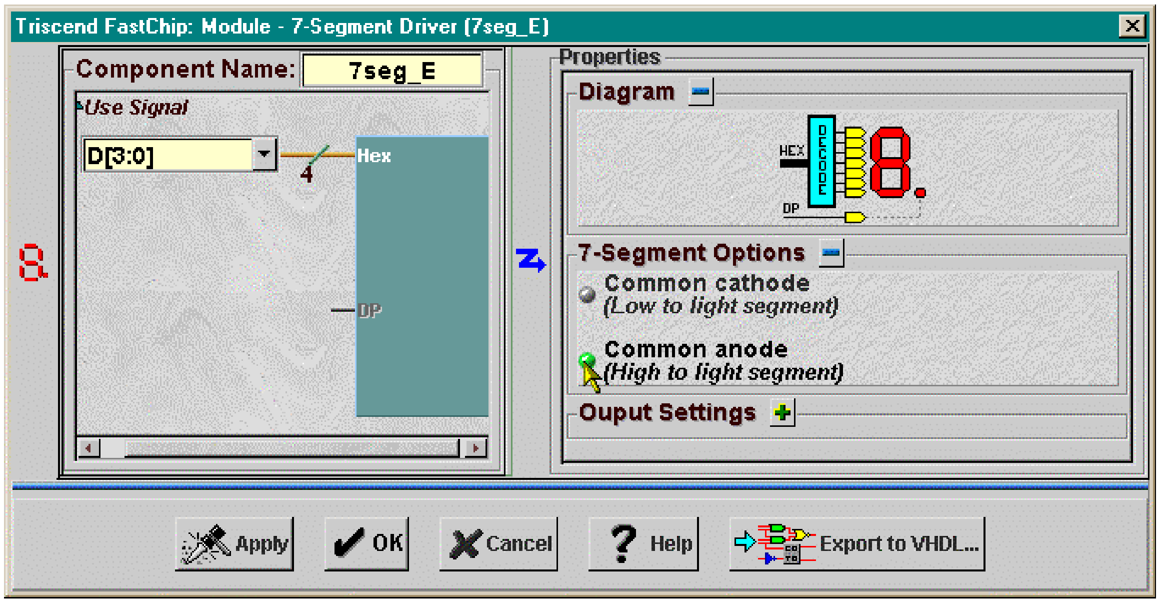


First, you want to connect the **D[3:0]** outputs from the **Input_A** module to the inputs of the **7seg_E** module. You could type `D[3:0]` into the <hex value> field, but there is an easier way. Click on the ▾ arrow on the right-side of the <hex value> field to display a scrollable list. This list contains every currently defined signal for your design. There are many global signals in this list that are always available as soon as you start your design. New signals are added to the list as you add soft modules and specify names for their inputs and outputs. Move down the list and click on the D[3:0] entry. That's all you have to do to connect the four outputs of the **Input_A** module to the **7seg_E** module.



You also have to set the polarity for the **7seg_E** module outputs. The segments of the LED digit on your CSoC Board glow when a positive voltage is applied to them. This is a *common anode* arrangement. (It would be a *common cathode* arrangement if the segments glowed when they were grounded.) Click on the Common anode button in the 7-Segment Options area to set the proper polarity for the CSoC Board.

You are finished making modifications to the **7seg_E** module, so click OK.


Triscend FastChip: Module - 7-Segment Driver (7seg_E)

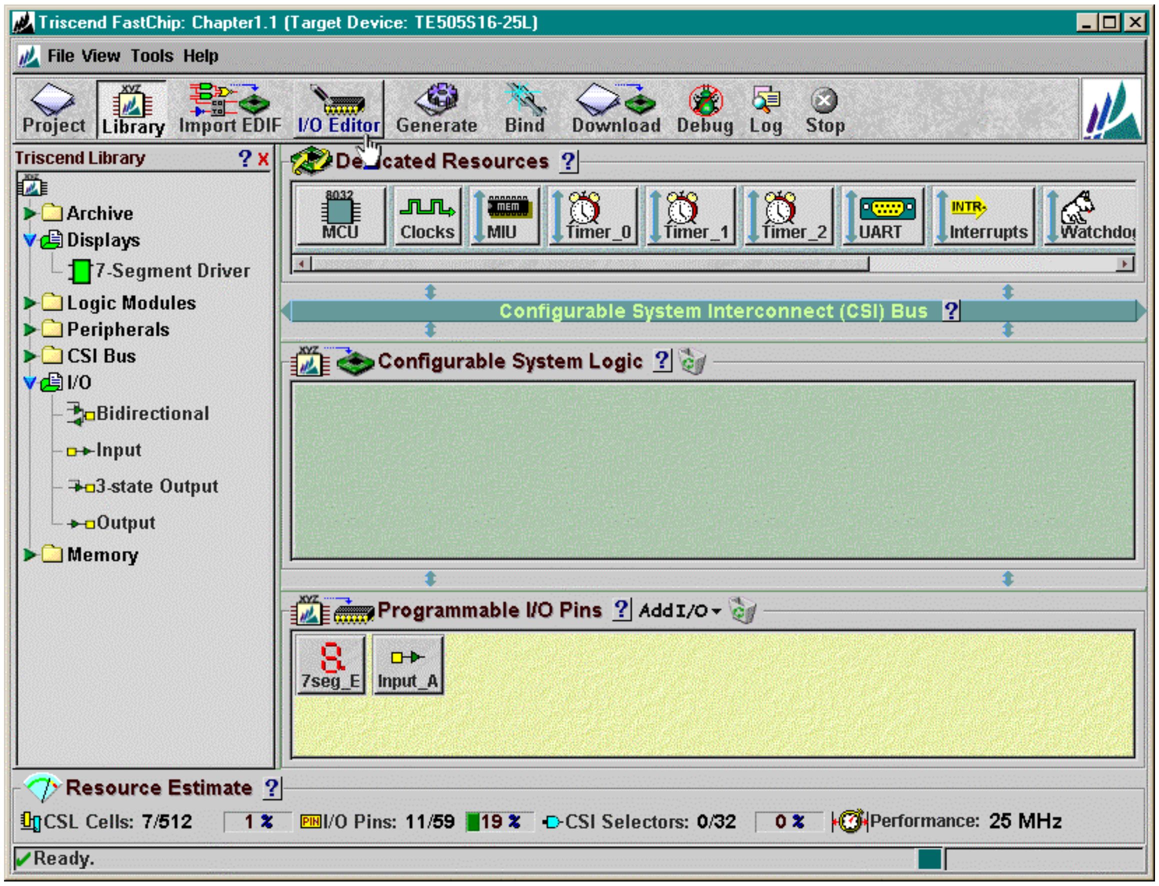
Component Name: 7seg_E

Use Signal: D[3:0]

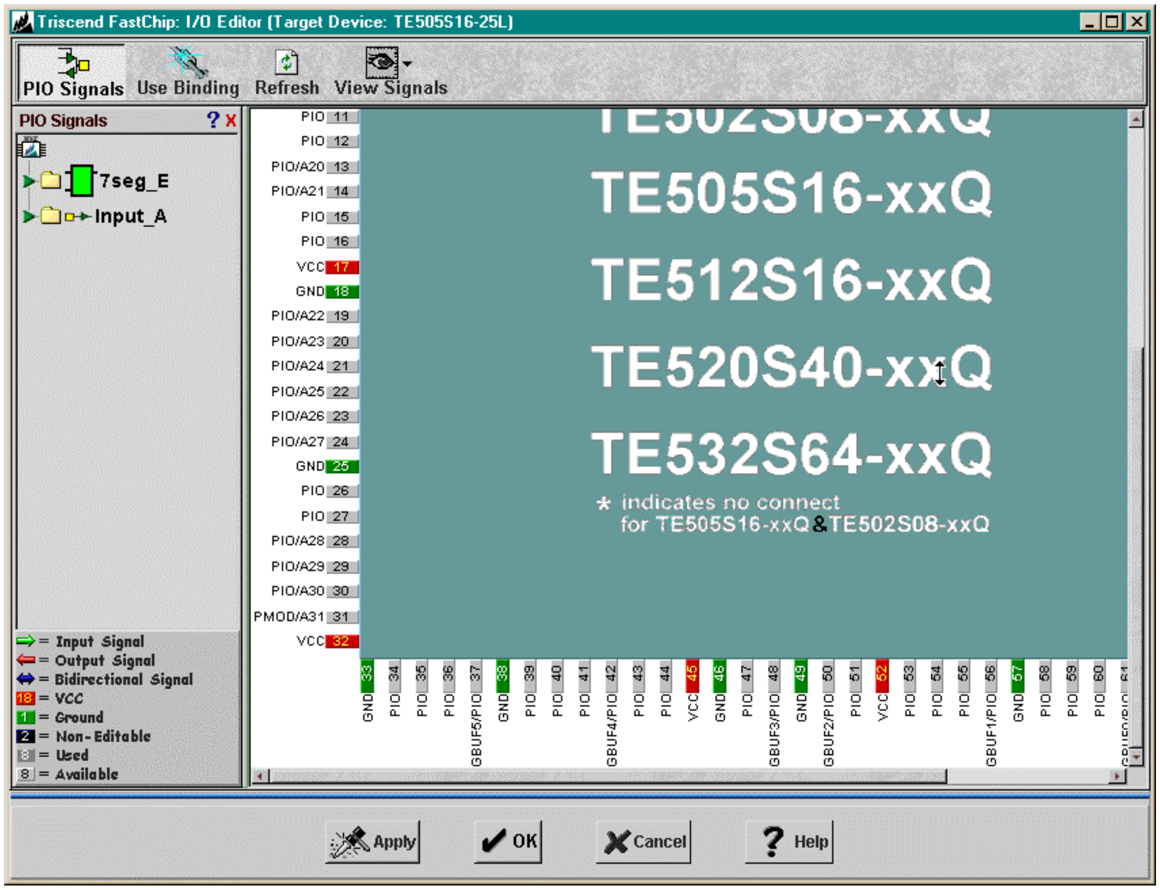
7-Segment Options:
( ) Common cathode (Low to light segment)
(•) Common anode (High to light segment)

### *Assigning Pins to Input and Output Signals*

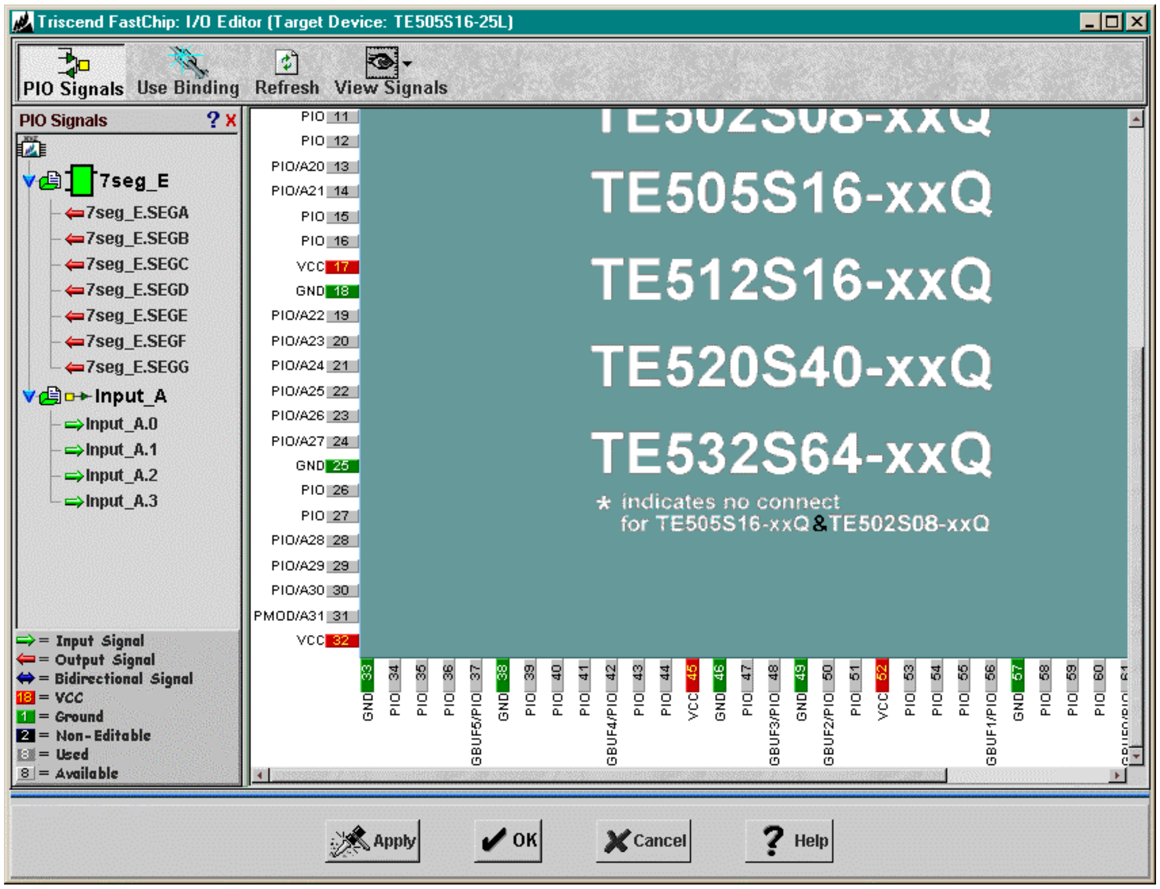
At this point the signals internal to the CSoC that connect the soft modules are defined. Now it is time for you to connect the inputs and outputs of your circuit to the I/O pins of the chip. Click on the I/O Editor icon in the toolbar to begin this process.



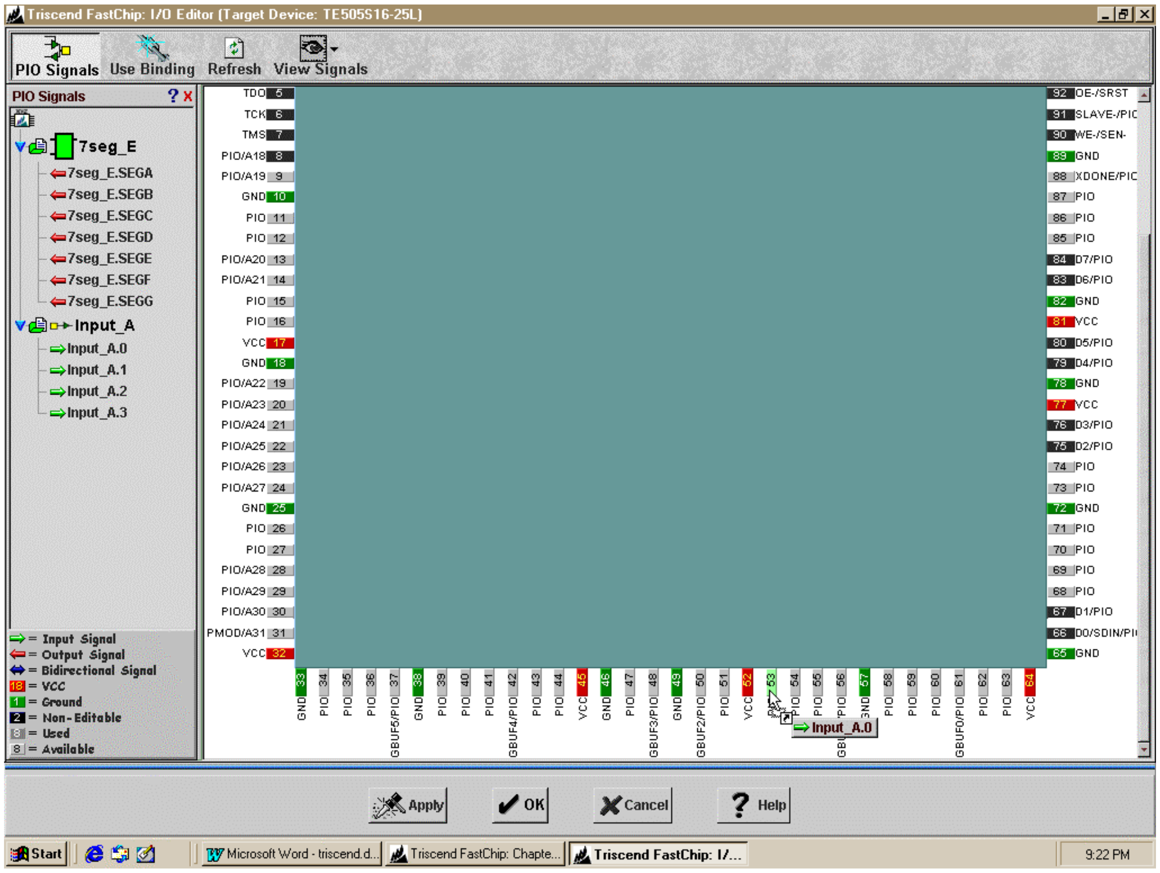
The **I/O Editor** window shown below will appear. The left-hand pane shows a list of the soft modules you dropped into the Programmable I/O Pins area of the project design window. The right-hand pane depicts the pins on the 128-pin LQFP package of the Triscend TE505 CSoC.



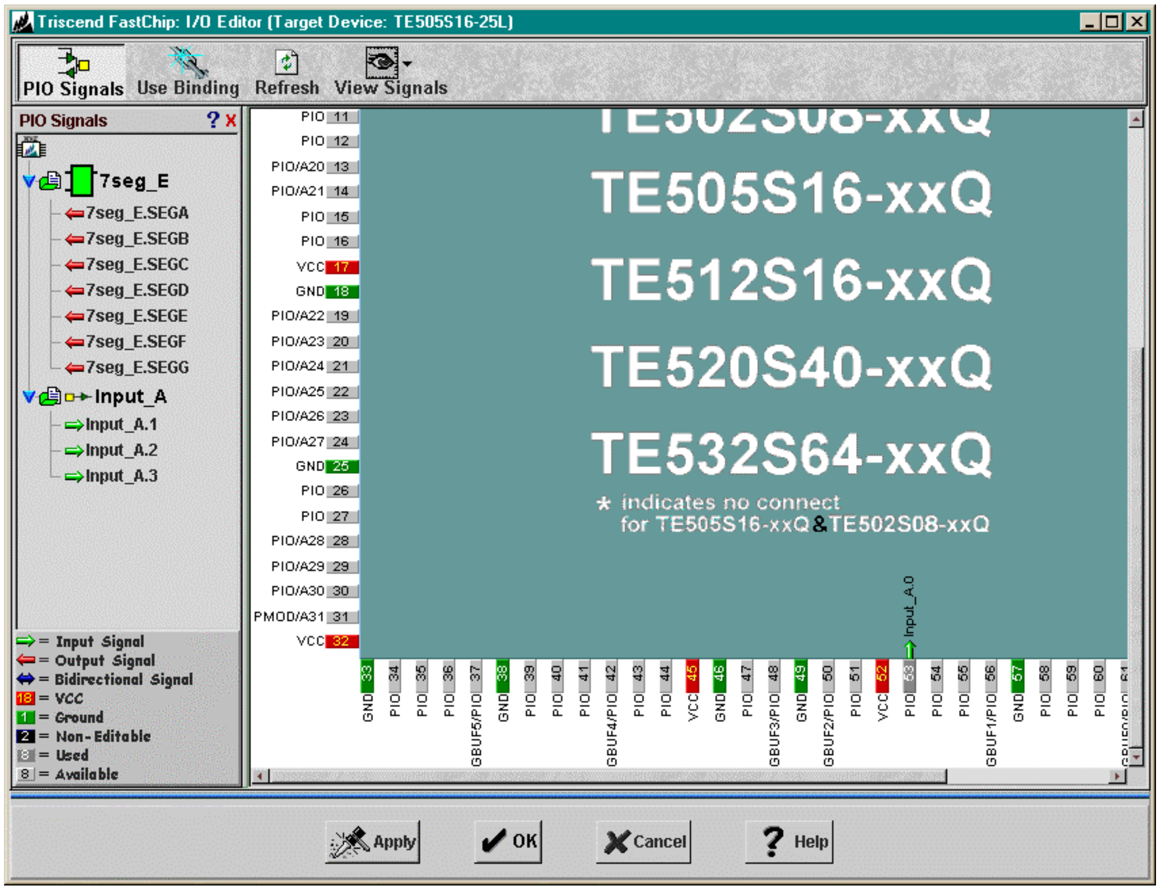
To begin assigning I/O pins to your circuit, you need to expand the I/O for the soft modules. Click on the ▶ arrow by the 7seg_E icon. The list of seven outputs from the LED decoder will appear. Click the ▶ arrow by the Input_A icon and the list of four inputs will appear. (The legend at the bottom of the left-hand pane lists the various icons that are used to identify inputs, outputs, etc.)



In order to assign one of the inputs (e.g. **Input_A.0**) to an I/O pin (e.g. 53), just click-and-drag the Input_A.0 icon from the left-hand pane into the right-hand pane. Move your mouse over pin 53 and release the button.



After you let go of the left-mouse button, the Input_A.0 icon disappears from the list in the left-hand pane and reappears above pin 53 in the right-hand pane. Signal **Input_A.0** is now attached to I/O pin 53 of the CSoC.



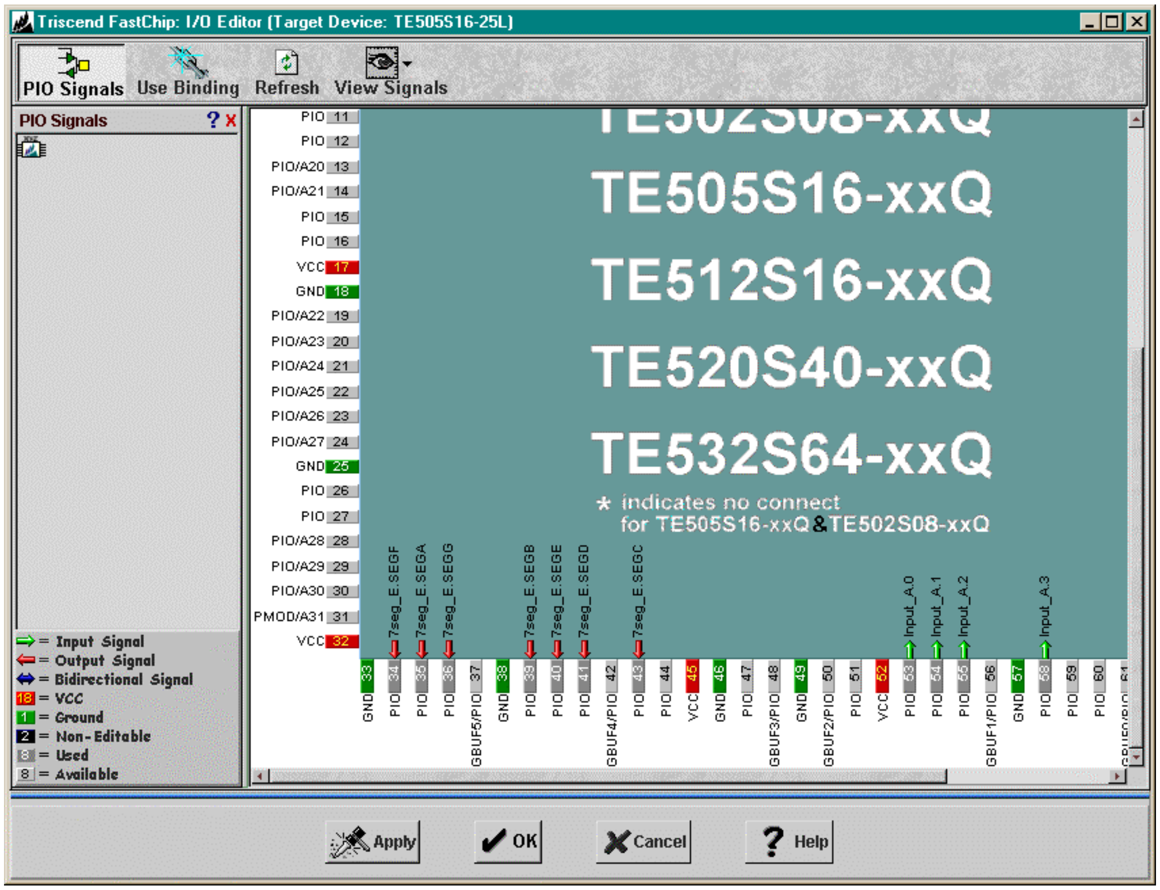
If you want to change the pin assignment for **Input_A.0**, just click on the Input_A.0 icon in the right-hand pane and move it to another pin. Or drag it back to the left-hand pane to undo the pin assignment completely.

Why did you assign **Input_A.0** to pin 53? Because the CSoC Board has a DIP switch attached to pin 53 that can drive the input to either a logic 1 or 0. Table 1 is a list of the relevant CSoC pins for this design and what they are connected to on the board:

**Table 1: Pin assignments and functions for the LED decoder design.**

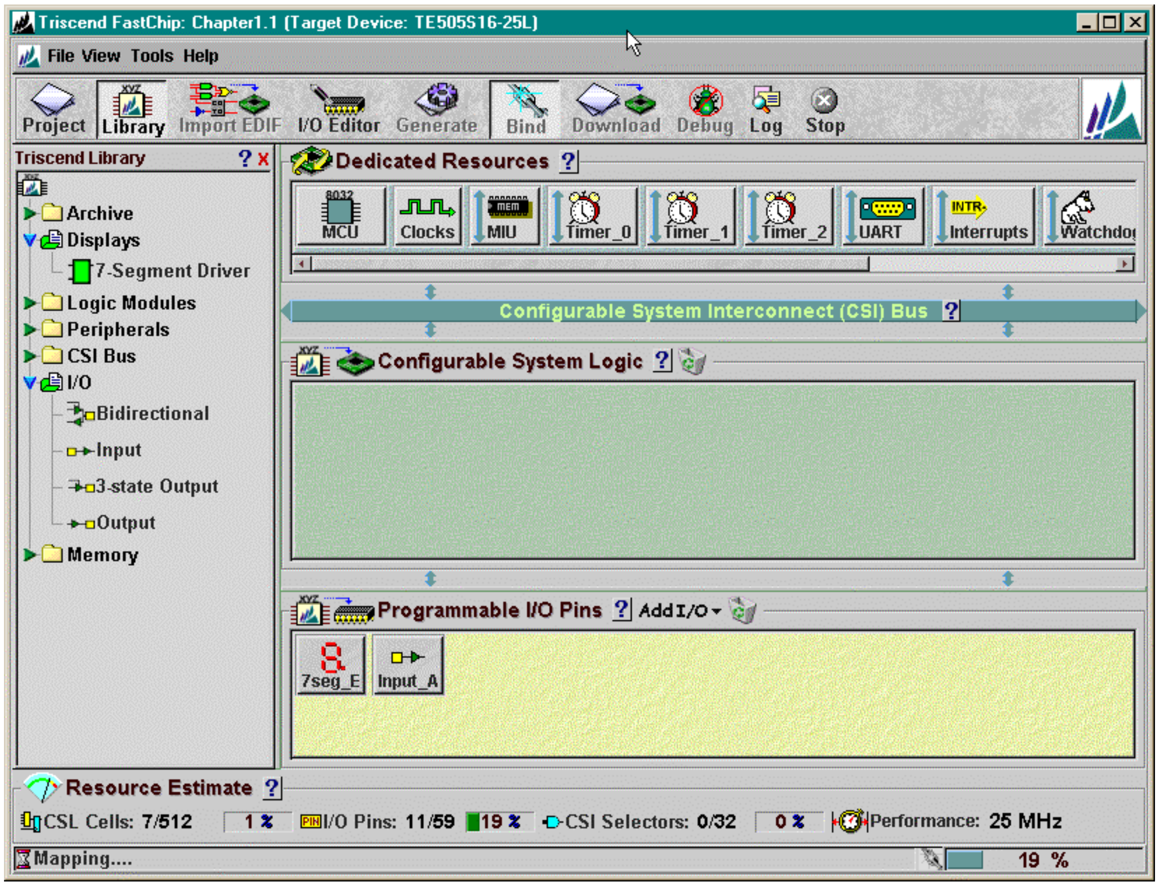
| Signal | Pin | CSoC Board Resource |
|---|---|---|
| **Input_A.0** | 53 | DIP switch position #1 |
| **Input_A.1** | 54 | DIP switch position #2 |
| **Input_A.2** | 55 | DIP switch position #3 |
| **Input_A.3** | 58 | DIP switch position #4 |
| *not used* | 59 | DIP switch position #5 |
| *not used* | 60 | DIP switch position #6 |
| *not used* | 62 | DIP switch position #7 |
| *not used* | 63 | DIP switch position #8 |
| **7seg_E.SEGA** | 35 | LED digit segment A |
| **7seg_E.SEGB** | 39 | LED digit segment B |
| **7seg_E.SEGC** | 43 | LED digit segment C |
| **7seg_E.SEGD** | 41 | LED digit segment D |
| **7seg_E.SEGE** | 40 | LED digit segment E |
| **7seg_E.SEGF** | 34 | LED digit segment F |
| **7seg_E.SEGG** | 36 | LED digit segment G |

Using the pin assignments listed in the table, click-and-drag the rest of the inputs and outputs from the left-hand pane to the appropriate pins. The **I/O Editor** window should now appear like the one below. After making all the pin assignments, click on OK to go back to the FastChip project design window.



## *Binding Your Design to the CSoC*

Now the soft modules for your circuit are instantiated and the signals which tie them together and to the I/O pins are specified. The circuitry in each module must now be *mapped* into the four-input LUTs contained in each CSL cell. Then the cells have to be *placed* into the matrix with an arrangement that lets them connect to each other and to the I/O pins. Then the signals between the modules and also to the I/O pins must be *routed* using the wiring within the CSL. Finally, a programming file must be generated that configures the internal circuitry of the CSoC so it physically implements the circuitry according to the mapping, placement, and routing phases. These operations are carried out by the FastChip software when you click on the Bind icon in the toolbar. You can see the progress of the binding operation on the right-hand side of the status bar at the bottom of the project design window.

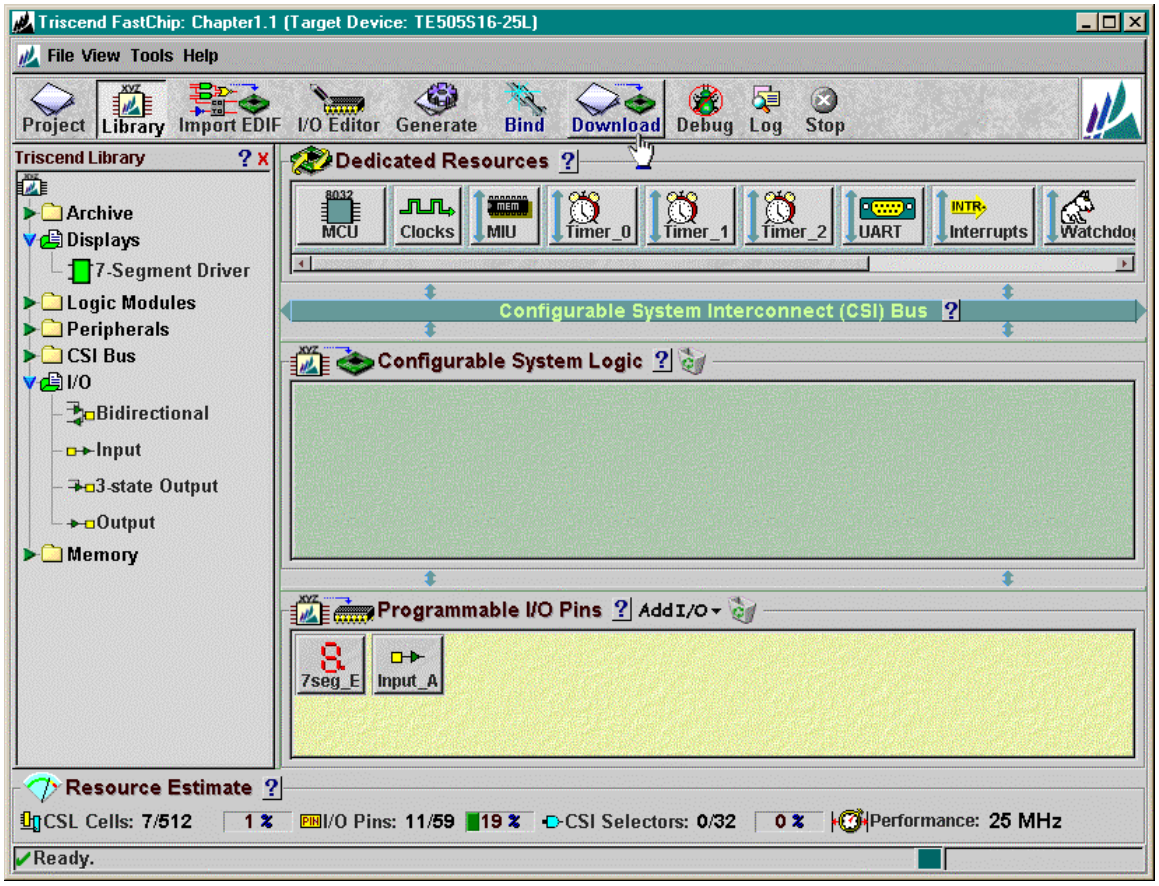


CSL Cells: 7/512 1 % I/O Pins: 11/59 19 % CSI Selectors: 0/32 0 % Performance: 25 MHz

Mapping.... 19 %

### *Downloading Your Design to the CSoC Board*

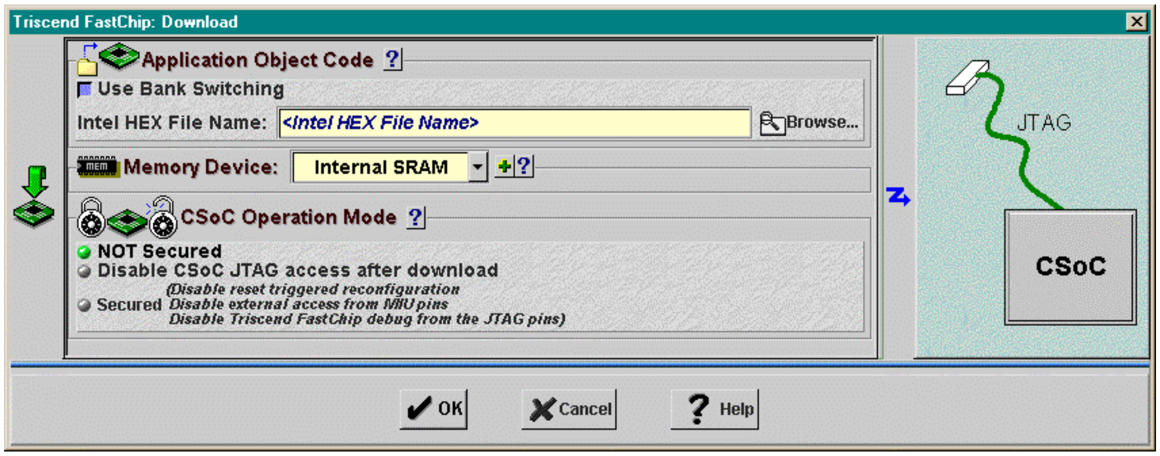
Once the bind operation is complete, you have to download the configuration file into the TE505. Click on the Download icon to begin this operation. *Make sure your CSoC Board is attached to the 9V DC power supply and it is connected to the parallel port of your PC with the downloading cable.*


Triscend FastChip: Chapter1.1 (Target Device: TE505S16-25L)

CSL Cells: 7/512 1 % I/O Pins: 11/59 19 % CSI Selectors: 0/32 0 % Performance: 25 MHz

Ready.

When the **Download** window appears, just click on OK to start downloading the configuration file to the CSoC Board. (Set the fields of the **Download** window to those shown below if your window appears differently. Make sure the Not Secured button is activated in the CSoC Operation Mode area or the CSoC Board may not operate correctly.)


Triscend FastChip: Download

Application Object Code ?

Use Bank Switching

Intel HEX File Name: <Intel HEX File Name> Browse...

Memory Device: Internal SRAM

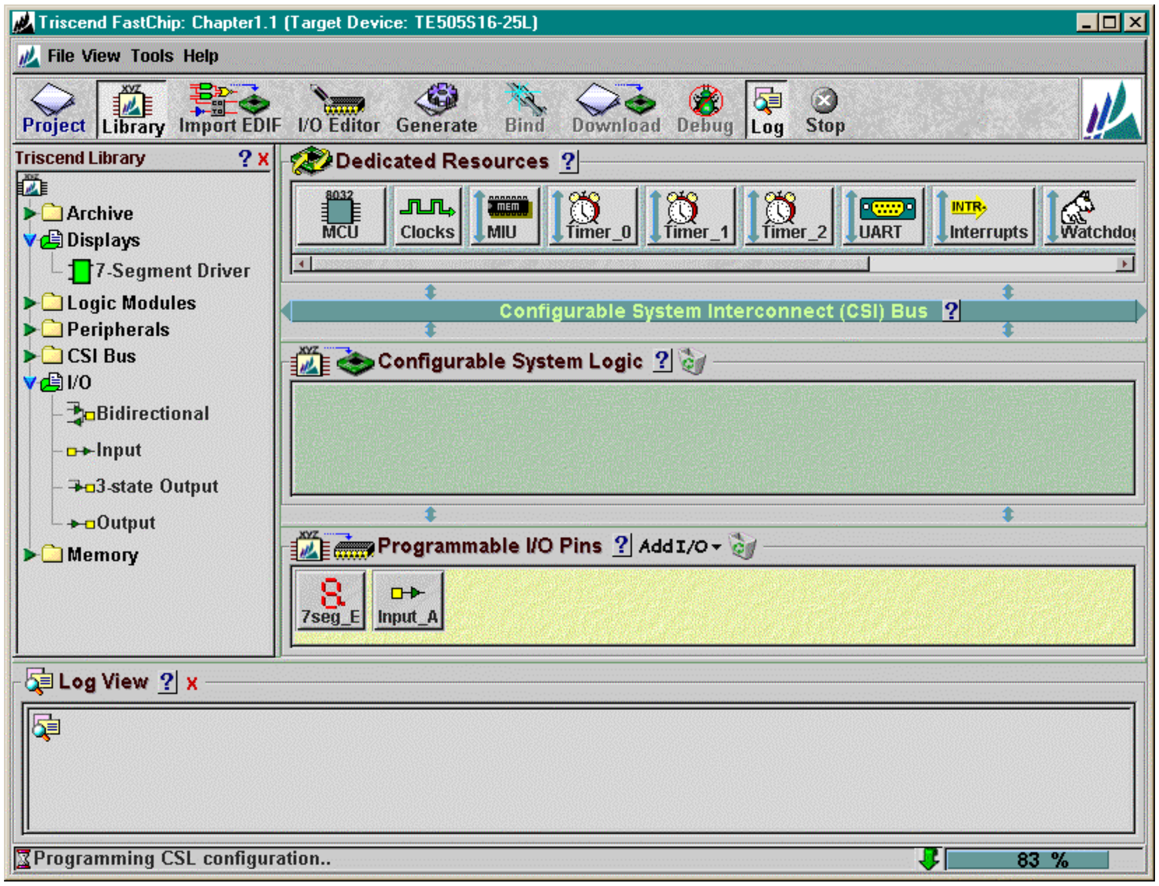
CSoC Operation Mode ?

NOT Secured

Disable CSoC JTAG access after download

Secured (Disable reset triggered reconfiguration
Disable external access from MIU pins
Disable Triscend FastChip debug from the JTAG pins)

Once you click on OK, the **Download** window will disappear and you can view the progress of the configuration file download process on the status bar at the bottom of the project design window. If the download process aborts, try checking the power supply and download cable connections to your CSoC Board.


Triscend FastChip: Chapter1.1 (Target Device: TE505S16-25L)

File View Tools Help
Project Library Import EDIF I/O Editor Generate Bind Download Debug Log Stop
Triscend Library
Archive
Displays
7-Segment Driver
Logic Modules
Peripherals
CSI Bus
I/O
Bidirectional
Input
3-state Output
Output
Memory
Dedicated Resources
MCU Clocks MIU Timer_0 Timer_1 Timer_2 UART Interrupts Watchdog
Configurable System Interconnect (CSI) Bus
Configurable System Logic
Programmable I/O Pins Add I/O
7seg_E Input_A
Log View

Programming CSL configuration.. 83 %

### Testing Your Design with the CSoC Board

After the configuration file is loaded into the CSoC Board, the LED digit should display a hexadecimal numeral (0–9 or A–F). Changing the settings of DIP switches 1–4 will change the numeral shown on the LED digit. Figure 5 shows the activation of the LED digit segments for each setting of the DIP switches. (Note that a given DIP switch forces a logic 1 on the CSoC input pin when it is pushed downward into the OFF position. It forces a logic 0 when it is pushed upward into the ON position.)

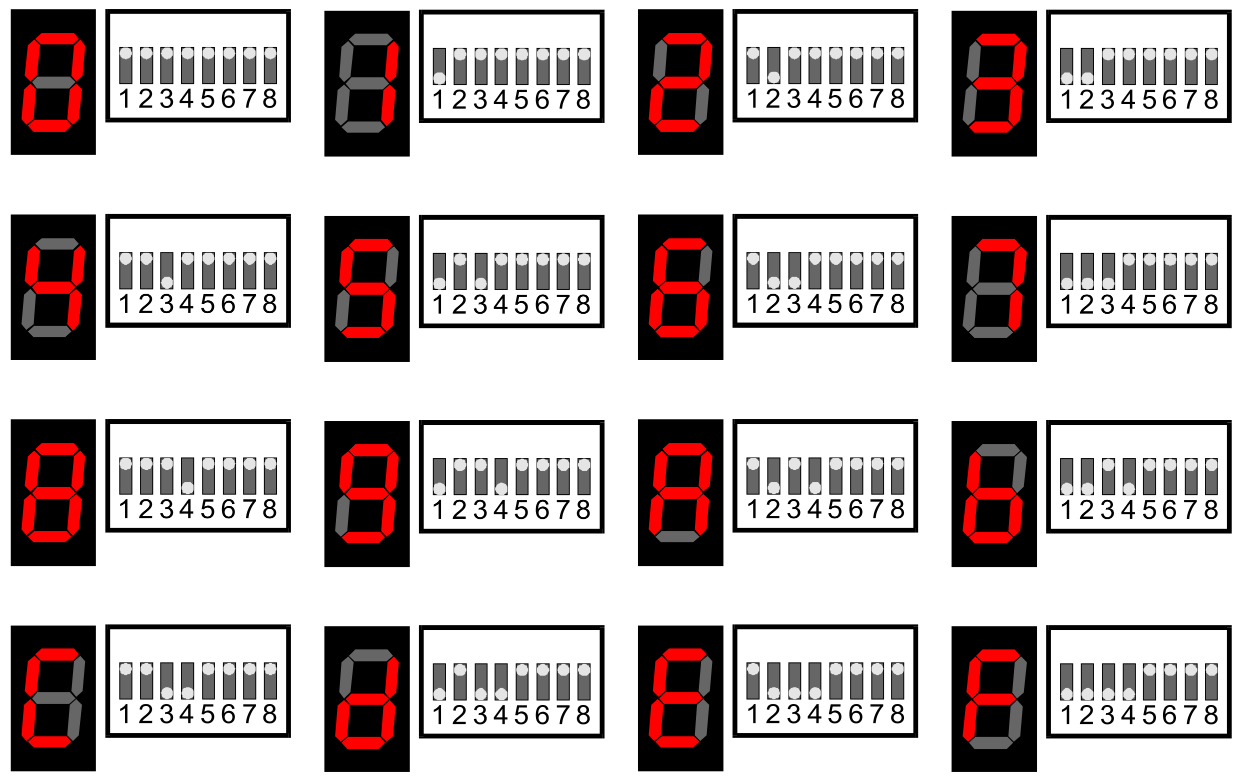


**Figure 5: The LED digit displayed for each DIP switch setting.**

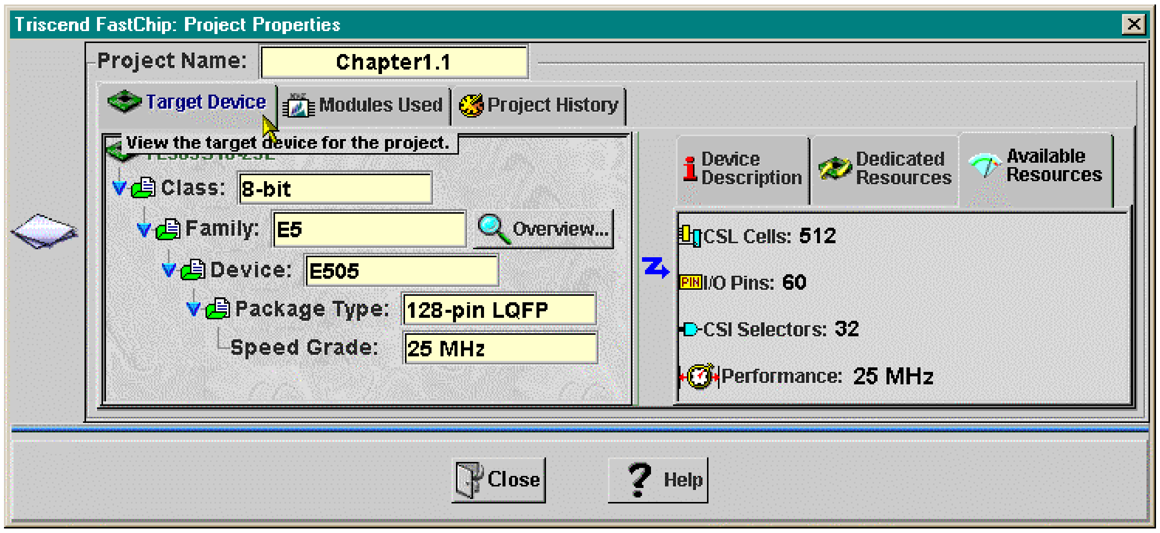
If your LED digit doesn't display the same patterns for the given switch settings shown in Figure 5, check the following:

- Check the LED decoder options and make sure it is in the common anode mode.
- Check your input and output pin assignments and make sure they match those in Table 1.

### Examining Project Properties

After testing and verifying your design, you can check the final statistics for your project. Click on the Project icon in the toolbar of the project design window. A **Project**

**Properties** window will appear that displays the information on the device targeted by your design.



Triscend FastChip: Project Properties

Project Name: Chapter1.1

Target Device | Modules Used | Project History

View the target device for the project.

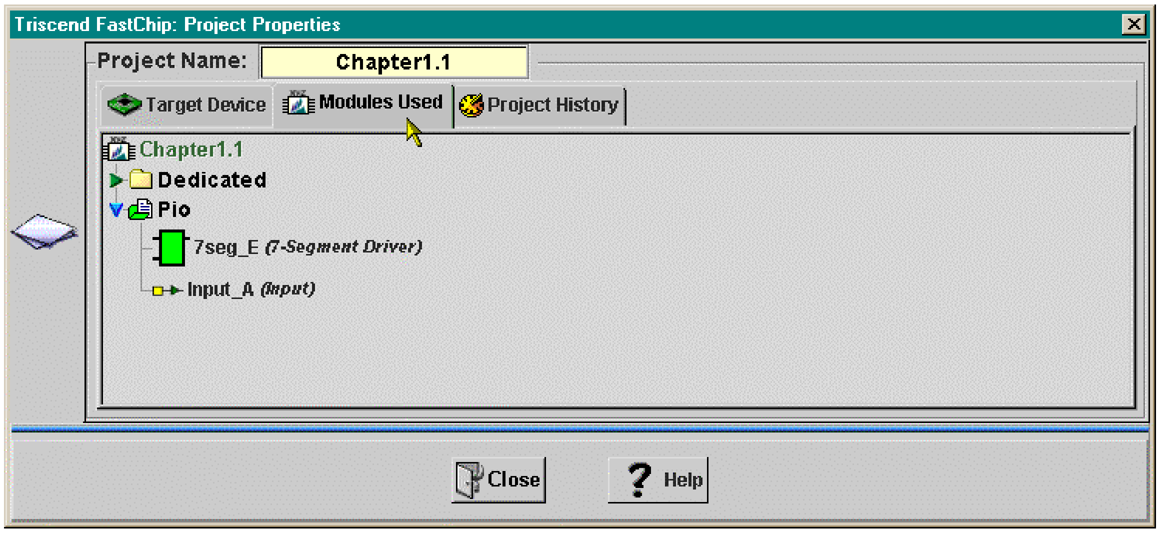
Class: 8-bit
Family: E5 Overview...
Device: E505
Package Type: 128-pin LQFP
Speed Grade: 25 MHz

Device Description | Dedicated Resources | Available Resources

CSL Cells: 512
I/O Pins: 60
CSI Selectors: 32
Performance: 25 MHz

Close | Help

Clicking on the Modules Used tab displays the soft modules you have instantiated in your design. There should only be the LED decoder and input modules in this example.



Triscend FastChip: Project Properties

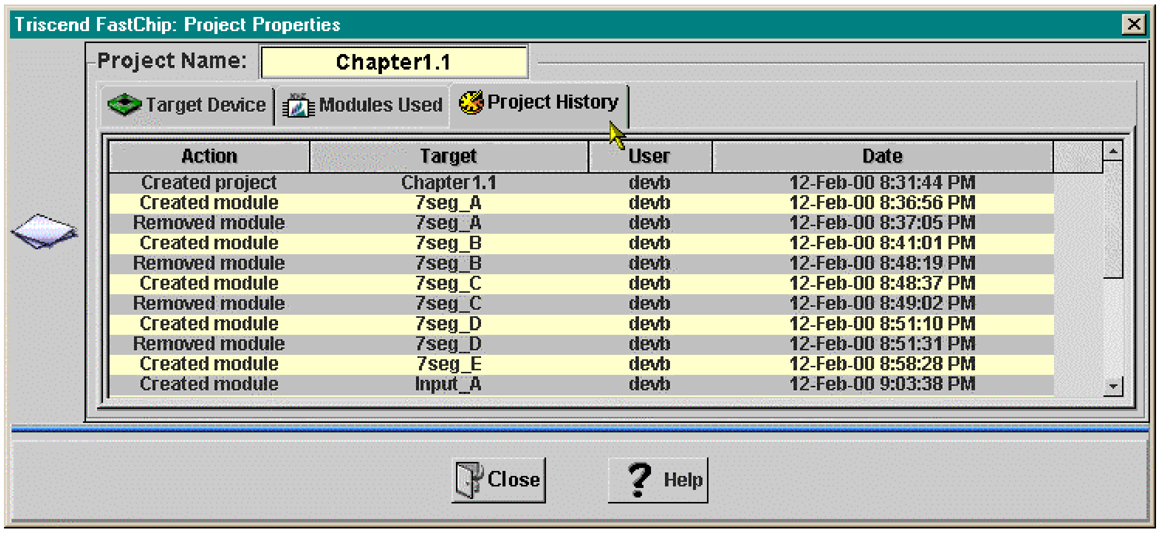
Project Name: Chapter1.1

Target Device | Modules Used | Project History

```
Chapter1.1
  Dedicated
  Pio
    7seg_E (7-Segment Driver)
    Input_A (Input)
```

Close | Help

Finally, clicking on the Project History tab displays a log of all the actions you performed while working on your design and the times when you performed them.

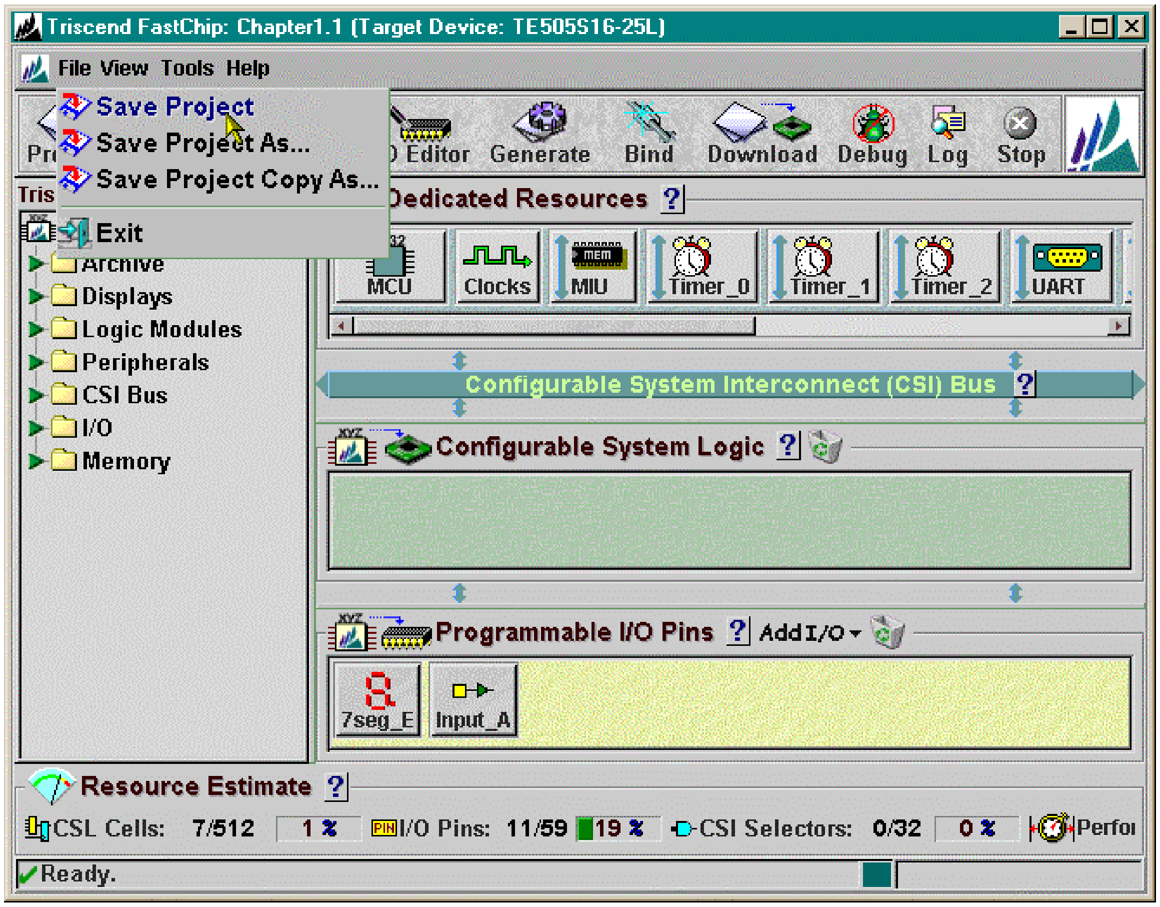


Triscend FastChip: Project Properties

Project Name: Chapter1.1

| Action | Target | User | Date |
|---|---|---|---|
| Created project | Chapter1.1 | devb | 12-Feb-00 8:31:44 PM |
| Created module | 7seg_A | devb | 12-Feb-00 8:36:56 PM |
| Removed module | 7seg_A | devb | 12-Feb-00 8:37:05 PM |
| Created module | 7seg_B | devb | 12-Feb-00 8:41:01 PM |
| Removed module | 7seg_B | devb | 12-Feb-00 8:48:19 PM |
| Created module | 7seg_C | devb | 12-Feb-00 8:48:37 PM |
| Removed module | 7seg_C | devb | 12-Feb-00 8:49:02 PM |
| Created module | 7seg_D | devb | 12-Feb-00 8:51:10 PM |
| Removed module | 7seg_D | devb | 12-Feb-00 8:51:31 PM |
| Created module | 7seg_E | devb | 12-Feb-00 8:58:28 PM |
| Created module | Input_A | devb | 12-Feb-00 9:03:38 PM |

After you look at the log, click on File⇒Save Project to save your LED decoder design into the C:\CSoC_Examples\Chapter1.1 folder. The click on File⇒Exit to terminate the FastChip program.



Triscend FastChip: Chapter1.1 (Target Device: TE505S16-25L)