

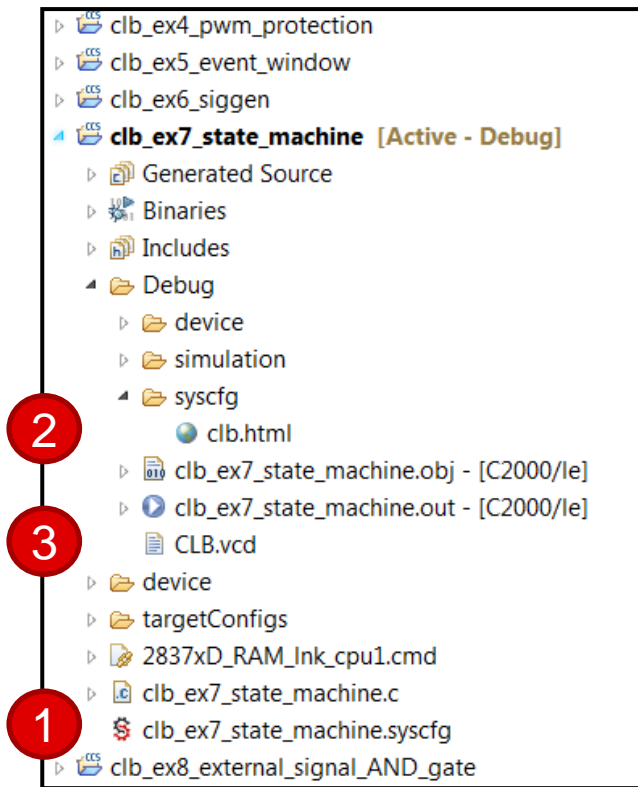
C2000™ CLB
configurable logic block
programming tool

CLB tool

Built into Code Composer Studio

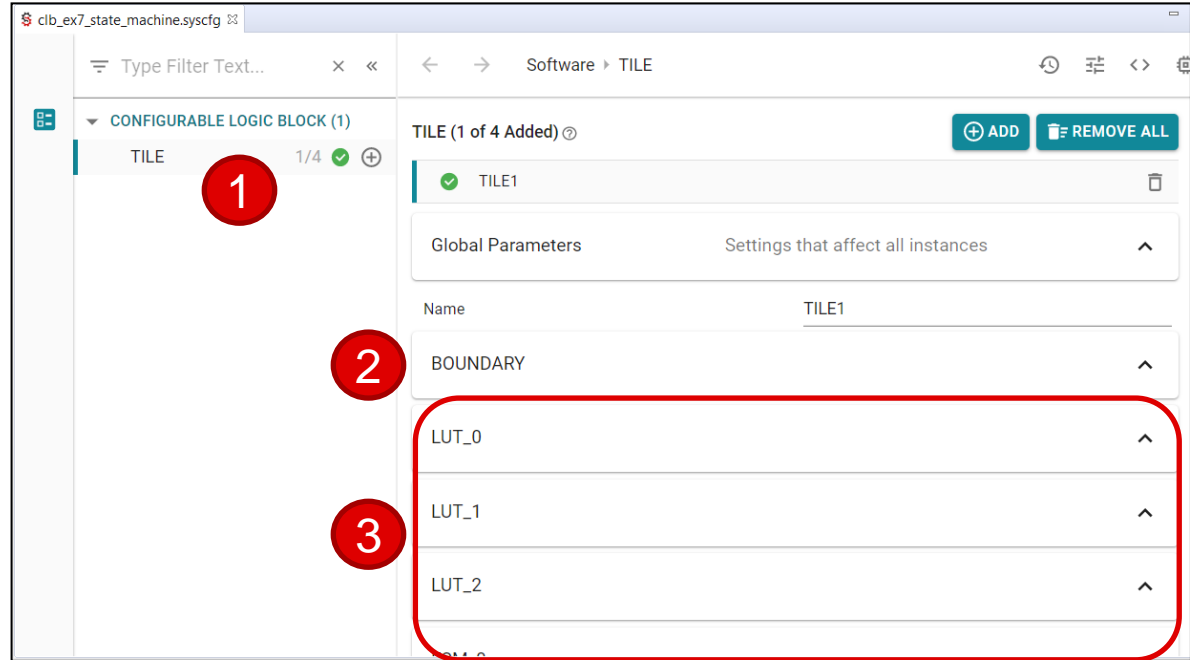
1. Simple and intuitive GUI tool to implement your hardware design for the CLB Tiles
2. Block diagram overview of the design
3. SystemC simulation for debugging

Examples available for all CLB enabled devices



CLB tool GUI

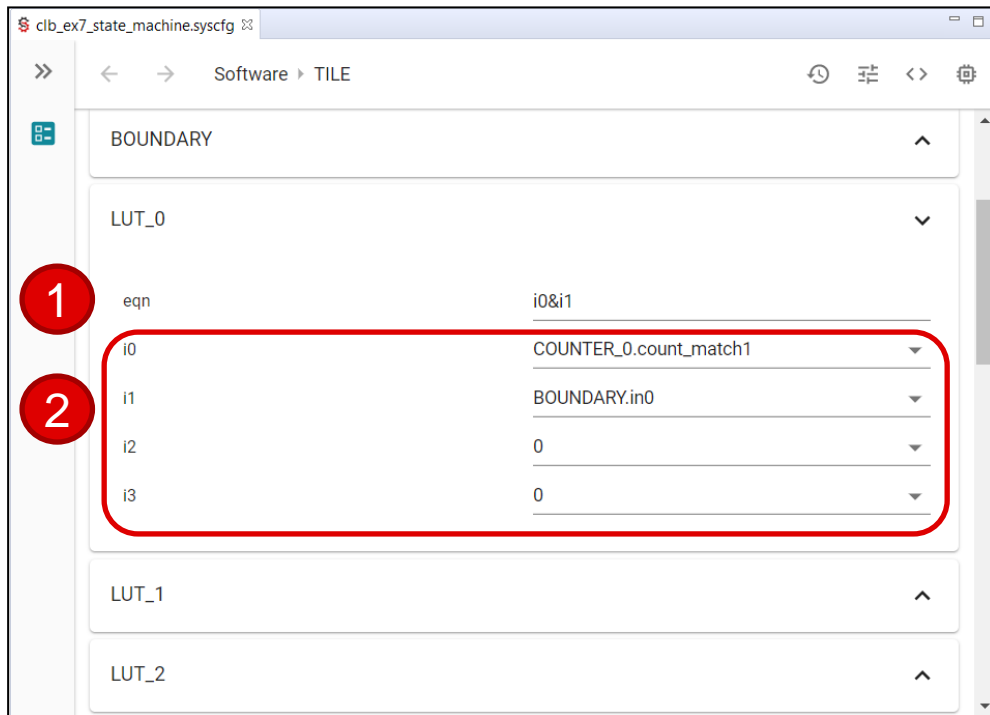
1. Keep track of available resources (number of CLB Tiles used)
2. BOUNDARY input used for simulating input signals for the TILE
3. All submodules inside the each CLB tile configurable through the tool



CLB tool – LUT configurable options

Each LUT has the following configurable options:

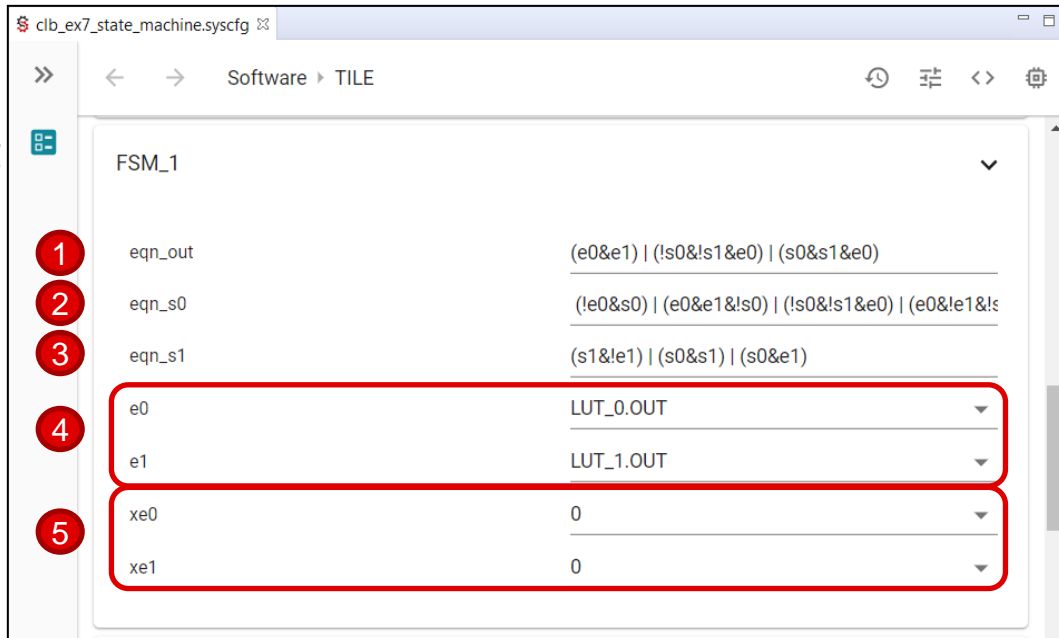
1. The logic equation for the output
2. i0 to i3 input signals



CLB tool – FSM configurable options

Each FSM has the following configurable options:

1. The logic equation for the output
2. The logic equation for the S0 NEXT state
3. The logic equation for the S1 NEXT state
4. e0 and e1 input signal
5. Extra input signal for when the FSM is used as a 2-state machine



The screenshot shows the CLB tool interface for configuring an FSM. The window title is "clb_ex7_state_machine.syscfg". The breadcrumb navigation shows "Software > TILE". The main content area displays the configuration for "FSM_1".

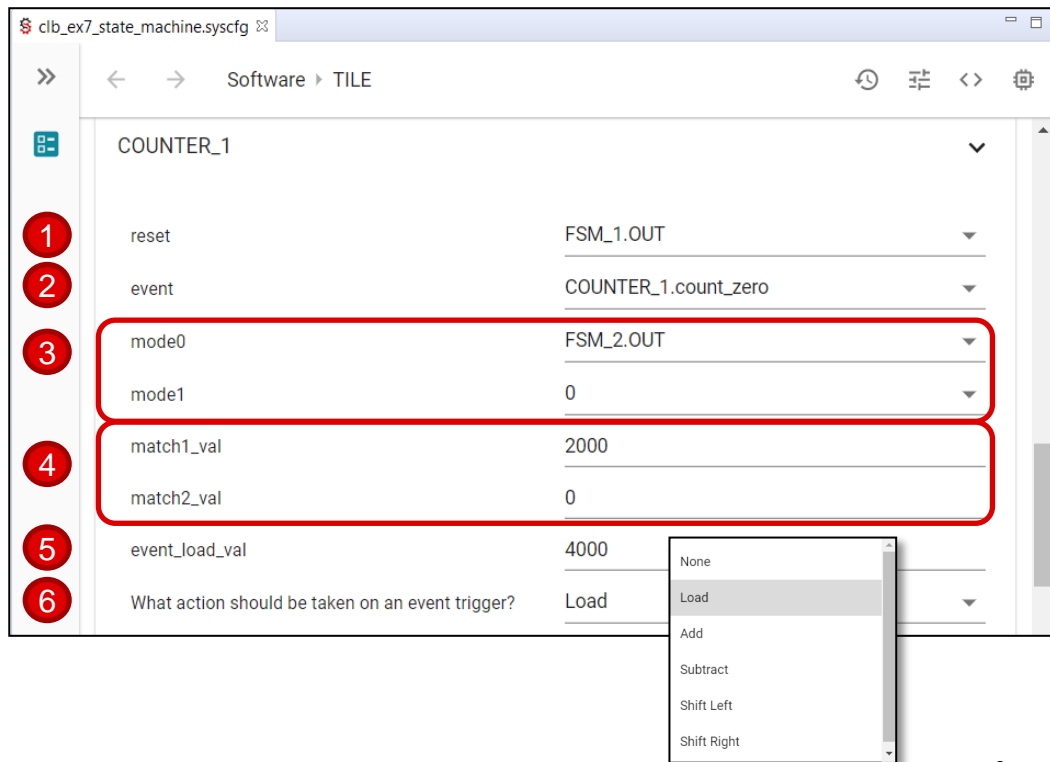
1	eqn_out	$(e0 \& e1) \mid (!s0 \& !s1 \& e0) \mid (s0 \& s1 \& e0)$
2	eqn_s0	$(!e0 \& s0) \mid (e0 \& e1 \& !s0) \mid (!s0 \& !s1 \& e0) \mid (e0 \& !e1 \& !s1)$
3	eqn_s1	$(s1 \& !e1) \mid (s0 \& s1) \mid (s0 \& e1)$
4	e0	LUT_0.OUT
	e1	LUT_1.OUT
5	xe0	0
	xe1	0

The configuration options are listed in a table. The first three rows (eqn_out, eqn_s0, eqn_s1) are numbered 1, 2, and 3 respectively. The next two rows (e0, e1) are numbered 4. The last two rows (xe0, xe1) are numbered 5. The input signals e0 and e1 are highlighted with a red box, and the extra input signals xe0 and xe1 are also highlighted with a red box.

CLB tool – COUNTER configurable options

Each COUNTER has the following configurable options:

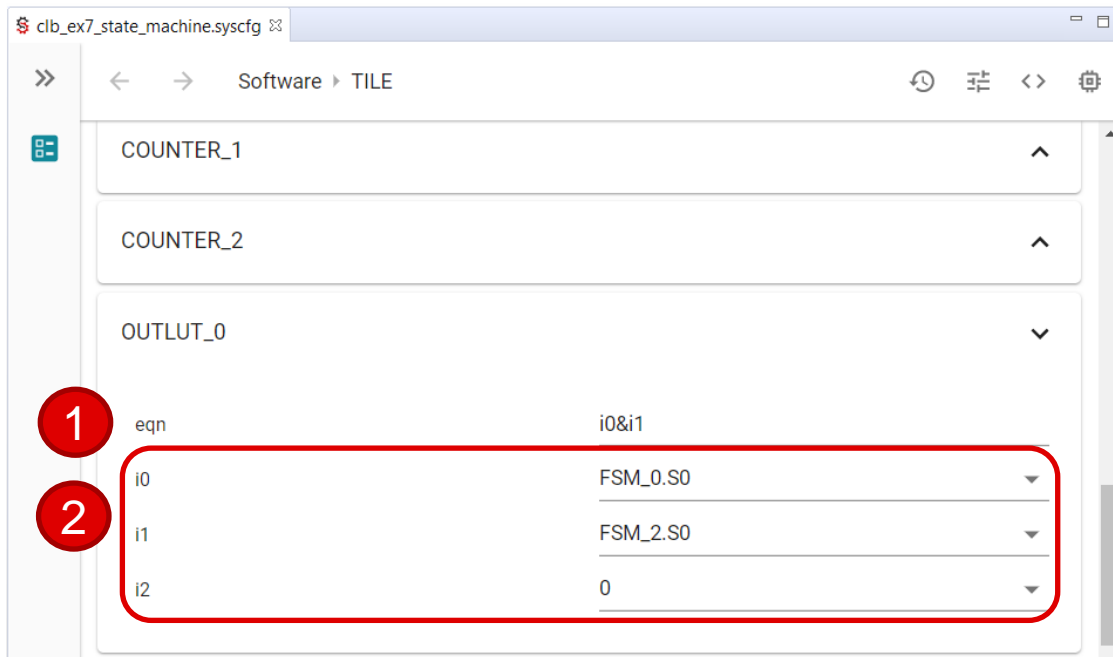
1. The reset input signal
2. The event input signal used along with the action option and the event load value option
3. mode0 and mode1 input signals
4. match1 and match2 values
5. Load value
6. The action specified when the event signal is triggered
 - configure the counter as adder, x2, /2, etc.



CLB tool – OUTLUT configurable options

Each OUTLUT has the following configurable options:

1. The logic equation for the output
2. i0 to i2 input signals



The screenshot shows the CLB tool interface for configuring an OUTLUT. The window title is "clb_ex7_state_machine.syscfg". The breadcrumb navigation shows "Software > TILE". The main content area displays a list of components: COUNTER_1, COUNTER_2, and OUTLUT_0. The OUTLUT_0 component is expanded, showing its configuration options. A red box highlights the "eqn" and "i0" to "i2" input fields. A red circle with the number "1" points to the "eqn" field, and a red circle with the number "2" points to the "i0" field.

Option	Value
eqn	i0&i1
i0	FSM_0.S0
i1	FSM_2.S0
i2	0

CLB tool – HLC configurable options

Each HLC has the following configurable options:

1. The event signals used to trigger the execution of up to 16 instructions
2. The general purpose registers R0-R3's initial values
3. The list instructions for each event input

HLC

Event 0 (e0)	LUT_2.OUT
Event 1 (e1)	0
Event 2 (e2)	0
Event 3 (e3)	0

R0_init	0
R1_init	0
R2_init	0
R3_init	0

Other Dependencies

program0	
instruct0	PULL C1
instruct1	PULL R1
instruct2	MOV_T1 R1,C1

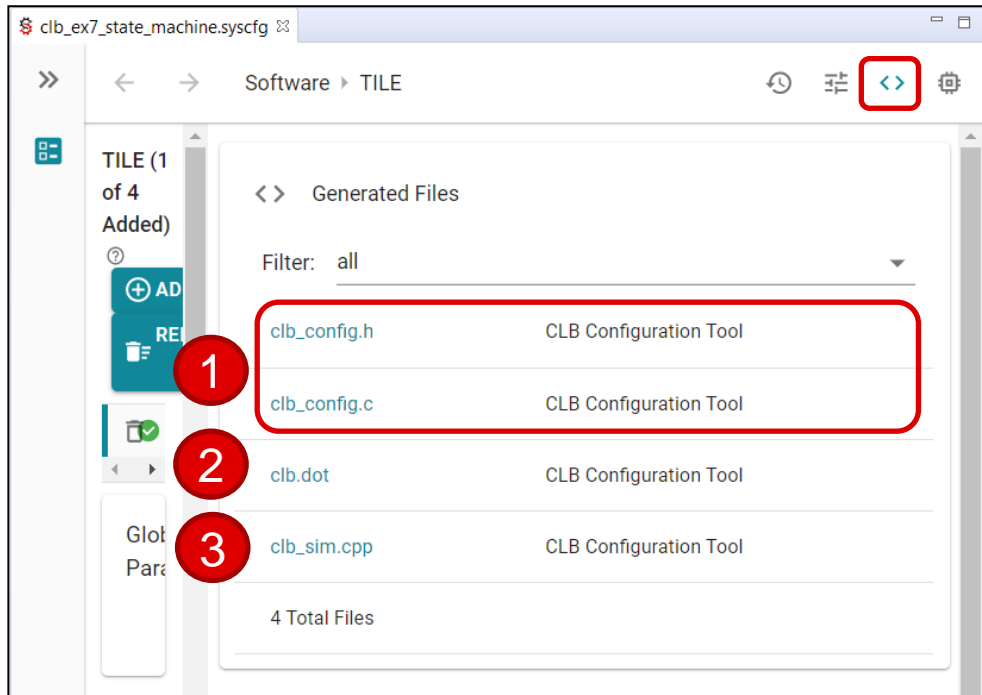
CLB tool auto-generated files

The CLB Tool will auto-generate:

1. The files needed to configure the CLB
2. The debugging simulation source file
3. The design block diagram source file

These outputs are then used by CCS to generate:

- MCU binary: .OUT file
- The simulation waveforms: .VCD file
- The CLB block diagram: .HTML file



Resources

- www.ti.com/c2000clb
- [C2000Ware](#) including the [CLB Tool User's Guide](#) [SPRUIR8]
- Application Note: [How to Design with the CLB](#) [SPRACL3]
- Application Note: [How to Migrate from FPGA/CPLD to CLB](#) [SPRACO2]

***Integrate* custom logic and *Augment* peripheral capability in your real-time MCU applications**