



Name		Description	Note
clk	I	clock input	
rst	I	reset input	starts at IP=0
sysFs	I	timer clock enable	1/16 prescaler
intr	I	H/W interrupt	pulse
wdt	I	Watch-dog interrupt	pulse
break	I	release from DOZE state	
busy	O	becomes high if uP is running	!DOZE
tstin	I	test condition lines	
pulse	O	single-line output port	
outputport	O	output port	
inport	I	input port	
portAdr	O	port address	
wen	O	output enable	
ren	O	input enable	

Features:

- * Configurable 16b micro-sequencer
- * Optimized for Virtex-II FPGA architecture
- * Dual stack (data, prog) architecture
- * Supports hardware interrupt
- * register window for argument passing
- * up to 5 levels of function call
- * 59 instructions (including 19 optional inst's)
- * MUL / DIV / MOD instructions (16b op 16b)
- * up to 8KW (1W = 9b) code space
- * rooms for application specific instructions
- * timer logic included
- * Supported by C-like high-level language

Deliverables:

- * Design Provided in Synthesizable Verilog
- * Development tools (Assembler, Compiler)
- * Debugging tools
(JTAG debugger, dis-asm, UART)
- * Complete test bench
- * Documentation (in pdf, in HTML)

Applications:

- * FPGA SoC project
- * Interface to Ethernet, USB 1.1, IrDA
- * Sequence control
- * Servo control (arithmetic operations required)

The *scc-II* is a user-configurable 16b micro-sequencer, which is compact, yet user programs can be developed using a C like high-level language (the language 'SC').

The *scc-II* takes very small foot-print, thanks to its stack-based architecture. By changing a configuration file, several optional features, such as supporting MUL instruction, can be turned on / turned off independently. This allows users to choose the optimal configuration for their applications.

The *scc-II* uses 9b instruction code (18b for two byte instruction). 9b wide memories are available in many FPGAs, such as Xilinx Virtex-II FPGA family. Thanks to this extra bit, the *scc-II* supports larger memory and I/O spaces as well as extra instructions, such as multiply instruction. There are several instruction code spaces left vacant which can be used for users' application specific instructions. The program size can be up to 8KW which is adequate for even a very complex system.

The *scc-II* provides a single interrupt line to handle external hardware events. In addition to hardware interrupt, a 'reboot' instruction is provided to allow software to escape to a designated address location with stack pointers initialized.

The biggest incentive to use the *scc-II* is ease of software programming. The Assembler, 'scc2asm', and the Compiler, 'scc2', and other software tools are provided along with the HDL design. Most of *scc-II* program can be written in high-level language. Low level functions (such as I/O, timer) are natively supported in the language SC. Although Assembler code might be required to write an interrupt handler, for example, such code can be included in an SC program using an asm directive.