

# Hardware Synthesis from SCCharts

Deprecated since 0.12

This article is deprecated. The described features are no longer available in current releases.

## Hardware Synthesis from SCCharts

SCCharts not only can be transformed into C-Code to run it on a  $\mu\text{C}$ , it is also interesting to synthesize SCCharts to hardware. There are several aspects to do that.

- On hardware there is per se a parallel execution, so parallel SCCharts regions can be executed in real parallel
- A synthesised hardware circuit is deterministic.
- Once the system clock is calculated in that way that all hardware units are executable, you have found your tick signal

But there are also problems in hardware synthesis.

- special handling for sequential variable assignment is needed

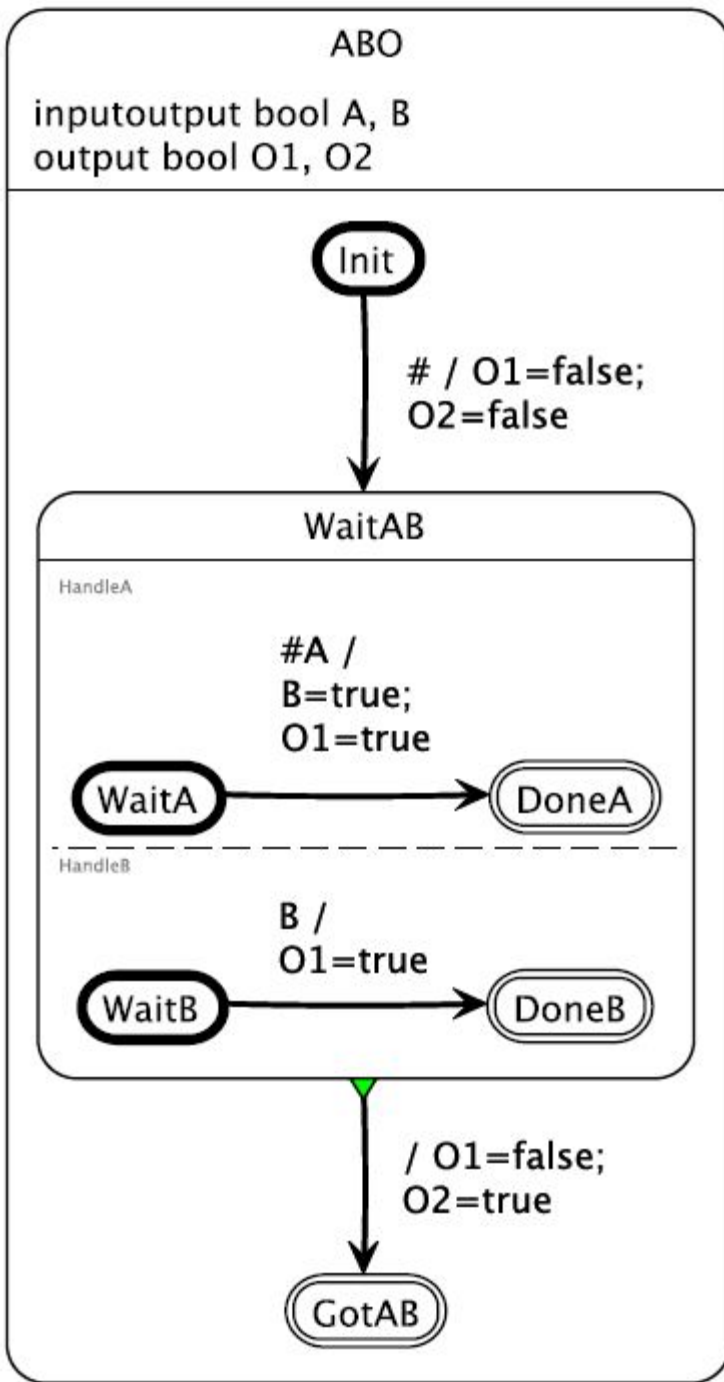
Where we are:

graphic coming soon.

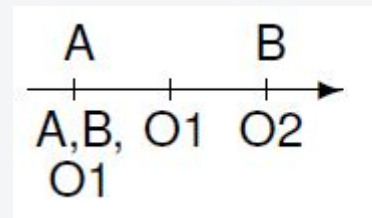
### First (naive) Implementation

In the first naive implementation the example ABO-ssa.c was taken and directly implemented in VHDL. The ABO-ssa.c example is a hand optimized ABO example which is written in synchronous C. The tick function was taken and implemented in VHDL. The "only" thing that was to do is, to change syntax differences between C and VHDL. The used variable must be adjusted too.

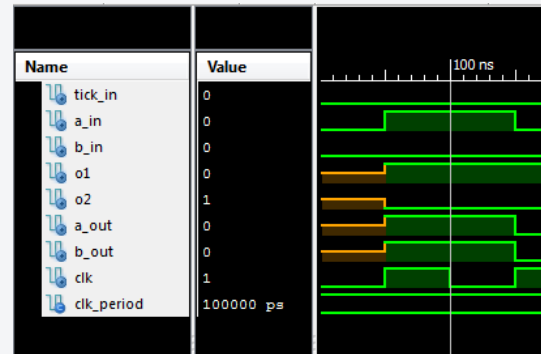
The picture shows the ABO SCChart and two possible execution traces. The signal graphs are showing the simulated ABO-hardware circuit. As you can see, it does the same 🍌.



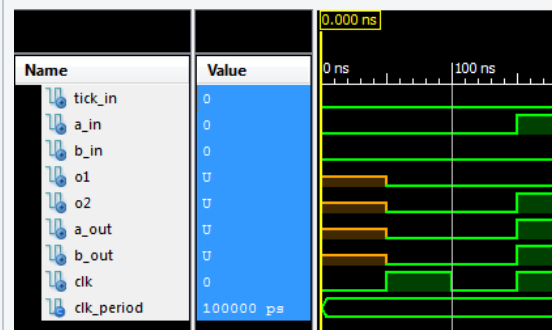
ABO example



two possible executi



signal trace from ex



signal trace from ex

**i** Here clk is tick\_in.  
A tick starts with every rising edge from clk.  
Do not note tick\_in.

After the execution from ABO the hardware signals still remain.

From this we can see that it is possible to synthesize hardware from SCCharts (Synchronous C).

