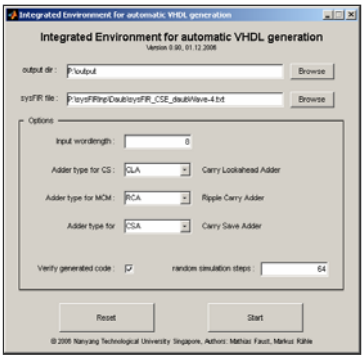
An integrated environment for automatic VHDL generation for FIR filters

In cooperation with Nanyang Technological University, Singapore

Mathias Faust

Markus Rähle

Diplomanden	Mathias Faust, Markus Rähle
Examinator	Prof. Erwin Brändle Assoc. Prof. Chip-Hong Chang
Experte	Theo Scheidegger
Themengebiet	High Performance Embedded Systems
Projektpartner	Nanyang Technological University, Singapore

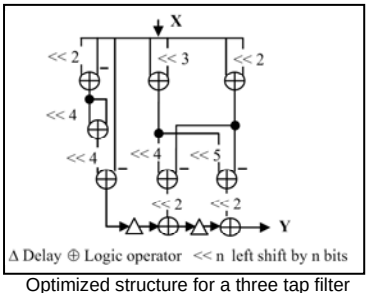


User Interface for VHDL generator

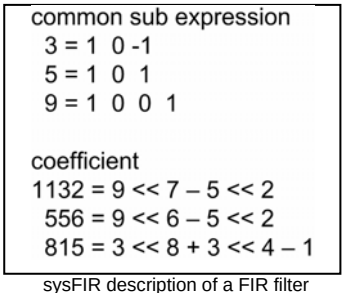
Problem: Creation of a unified platform within MatLab® that streamlines the process from filter coefficient to VHDL code and prepares for ASIC implementation. The platform should also provide an option to verify the VHDL codes by running a simulation in ModelSim®. Besides integrating the present parts, the platform should be modular so new parts can be plugged in easily.

Project Goal: The goal is, to build an integrated environment to ease the process getting from filter coefficient to running VHDL code and prepare the implementation in an integrated circuit. Several algorithms for optimizing the multiple constant multiplication of a transposed direct form FIR filter have been developed at Nanyang Technological University. Prior to our project there was no simple and fast way to compare the performance of these algorithms. Comparisons use area, delay and power estimations of the resulting integrated circuit, so it is essential to generate these data fast and always in the same way. Parts of the result generation exist but have not been integrated before. To integrate and improve these parts is the main focus of our work.

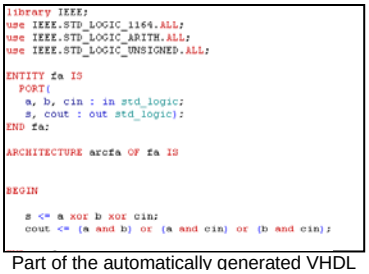
Solution: The process in the new integrated environment starts with typing in the filter coefficient and choosing the algorithm for optimization. The algorithm then reduces the cost of implementation by eliminating common subexpressions. The output of the algorithm is transformed to a standard format, called sysFIR. This format was designed to be very flexible and robust. Next, the newly developed VHDL code generator creates the VHDL file according to the sysFIR input. Several options are available that influence VHDL code generation, e.g. adder type. To verify the generated VHDL code it is simulated in ModelSim® and the result is compared to the calculation of MatLab®. The last step in the process is to generate scripts to run a Unix based Design Compiler in an automated way. Now the process runs fully automated and finishes within a few minutes rather than days as before when it was done manually.



Optimized structure for a three tap filter

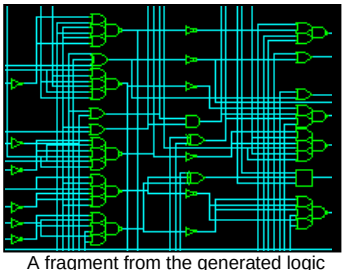


sysFIR description of a FIR filter



```
library IEEE;
use IEEE.STD_LOGIC_1164.ALL;
use IEEE.STD_LOGIC_ARITH.ALL;
use IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY fa IS
  PORT(
    a, b, cin : in std_logic;
    s, cout : out std_logic);
END fa;

ARCHITECTURE arcfa OF fa IS


BEGIN

   s <= a xor b xor cin;
   cout <= (a and b) or (a and cin) or (b and cin);
```

Part of the automatically generated VHDL



A fragment from the generated logic