Cluster-wise State Encoding

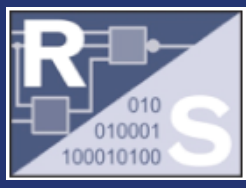
Mini Task Report

Supervisor: Dennis Wolf M.Sc.

Start: 05/11/2017 | Submission: 06/14/2017

Institute for Computer Engineering | Computer Systems Group | Prof. Dr.-Ing. Christian Hochberger

TECHNISCHE UNIVERSITÄT DARMSTADT



1 Introduction

1.1 Cluster Search

In order to encode the states cluster-wise, a good partition into clusters must be found. In the following, the criteria for well-formed clusters concerning the previously presented encoding strategy are described:

Available LUT Inputs

In the any case, each cluster must fit into one slice respectively lookup table. Therefore, all inputs and necessary signals for the state transitions inside the cluster and from other clusters must be fed to the LUT inputs. The amount of signals is calculated as shown in eqn. . One input is occupied for the cluster's own active bit; the states inside the cluster are encoded binarily, therefore $ld(number_of_states)$ bits are needed foir the states. From the external inputs, each bit which is not a shared dc in any of the contained transitions must be fed into the LUT inputs. At last, each other clusters input which is associated with any of the contained states must be considered as one input. All of these conditions must be met simultaneously.

Since the clustering problem is a combinatorial one, especially for large state machines, no ideal clusters can be searched. A heuristical approach is necessary. In this project, the application of a simulated annealing has been chosen because of its well-known effectiveness in hardware synthesis and the problem-independent implementation. The problem reduces to finding a suitable fitness function and mutation scheme. Due to the timed scope of this project, no further tuning of the other parameters like starting and end temperature could be included. Both fitness function and the mutation scheme are described in the next sections.

1.2 Usage of the BLIF format

The BLIF format for finite state machines as described in [**?**] has some advantages which qualify it over a HDL representation. Among these are the the good parseability, which is especially useful in small projects, and its independence of a specific platform. Additionally, the input files describing state machines in the scope of this project are .kiss files as well. Therefore, only the circuit mapping has to be written to the output file, and no conversion from BLIF to a HDL description of the functional behaviour has to be pursued. As a disadvantage, the .kiss format does not allow a more detailed circuit description than the mapping of latches. In our case, this disadvantage could be neglected, so that BLIF was the output format of choice.

2 Literaturverzeichnis