DES encryption core on NCL asynchronous logic

Asynchronous implementation of the DES encryption algorithm using 'Null Convention Logic'

Alberto Romero, Jordi Carrabina
Universitat Autònoma de Barcelona
{aromero, bordoll}@microelec.uab.es

Christopher Heer, Holger Bock
Infineon Technologies AG
{christoph.heer,holger.bock}@infineon.com

Jaafar Karker
ENS EIRB
karker@enseirb.fr

The DES algorithm: Sync-Assync Partition

- DES algorithm
  - Uses a 56 bit key length to encrypt 64 bit data blocks
  - 16 rounds where 48 bit long sub-keys are mixed with the data using permutations, S-boxes (tables) and XOR's

- Block partition
  - Clocked block with 8 selectable keys and data storage
  - Fully asynchronous core:
    - Key processing and sub-key generation
    - Encryption core
    - 5 Control FSMs and a 16- or 8 bits counter
Design methodology

- Standard tools
  - Synopsys
  - Modelsim
  - Avant
- VHDL + NCL packages and libraries
- Two-pass synthesis:
  - 1st pass generates a standard gate netlist from using a generic cell library
  - 2nd pass translates the standard gate netlist to a NCL library

Design flow evaluation

- NCL flow is a viable asynchronous flow
- Great advantage on using standard EDA tools
  - Smaller learning curve
  - Can be adapted on pre-existing design flows
- Uses VHDL language
- Different signal types and testbenches for pre-synthesis and post-synthesis simulation makes it difficult to compare results
- Some major flaws on synthesis from VHDL
  - Constants fixed to DATA value: If a signal is reduced to a constant value, it will be synthesized as a 'fixed' NCL (no DATA-NULL toggling)
  - Useful on data-driven designs, less on FSM based designs
  - The VHDL code tends to be "too structural"