SOFTWARE PRODUCT DESCRIPTION

FOR DK DESIGN SUITE VERSION 5.0

SPD number 02.05.0408

April 2008

DK DESIGN SUITE VERSION 5.0

DESCRIPTION

DK is a tool for the design and creation of electronic systems using the Handel-C language. It provides facilities for the creation, debugging, simulation and compilation of the Handel-C language and has capabilities for simulation of mixed C++, ANSI-C and Handel-C code and for co-simulation with other simulation and debugging tools.

The DK software supports two products families, DK Design Suite and the various platform developer's packages (PDP). Specific features available to the user vary according to the product purchased and are as agreed in the quotation. Table 2 shows the basic product feature blocks and their availability.

Both DK Design Suite and the PDP packages are designed so that they may be used with the Agility Platform Developer's Kit. The functionality of the Platform Developer's Kit is described separately in its own SPD.

FEATURES

The DK Design Suite provides an Integrated Development Environment (IDE) which allows a user to edit and compile Handel-C source code files to produce EDIF netlists, RTL VHDL or Verilog. A user can simulate and debug a Handel-C application within the IDE using a cycle accurate symbolic debugger. The IDE also provides project configuration tools to manage building applications with many files.

The IDE is composed of the following components:

Component	Description
Graphical User Interface (GUI)	Interface through which the user interacts with the IDE to create and edit code and to initiate compilation, simulation and debugging.
Handel-C compiler	Translates Handel-C source code files to a hardware description (EDIF/VHDL/Verilog) or a C++ /ANSI-C simulation which is then further compiled into executable code. Translation into hardware includes: **Technology mapping:** The DK Technology Mapper performs technology mapping of general logic into device specific logic blocks. Technology mapping is available for EDIF output for the following devices: • Xilinx: Virtex, VirtexE, VirtexII, VirtexII-Pro, VirtexII-Pro-X, Virtex-4, Virtex-5, Spartan-II, Spartan-IIE, Spartan-3, Spartan-3E, Spartan-3A and Spartan-3A-DSP. • Altera: Apex 20K, 20KE and 20KC, Apex II, Stratix, Stratix-GX, Stratix II, Stratix II-GX, Cyclone, Cyclone II and Cyclone III. **Logic estimation:** The Handel-C compiler can give information on logic usage and

Component	Description

	depth to help optimize designs. (Note that as this information is based on estimates, full place and route is needed to get exact logic and area information.)
	
	Optimization: Certain compiler optimization techniques can be selected by the user.
	
	Retiming
	
	The retiming option moves flip-flops around a circuit to try and meet the required clock period.
	
	It preserves the timing for logic between clock domains and the timing adjacent to interfaces, but moves flip-flops in other parts of the circuit until the respective clock periods are met. The retimer also moves flip-flops at interfaces in order to meet the intime and outtime specifications. After this the flip-flops are moved again to minimize the number in the circuit, whilst conserving the specified clock period.
	
	Automatic Mapping to Embedded ALUs
	
	Some FPGA devices possess embedded ALU primitives, which the Handel-C compiler has the ability to target automatically. Rather than leave it up to the user to specify where special ALU units should be used, the compiler intelligently uses them where they will provide the most improvement in performance over the equivalent logic.
	
	Currently the following ALU primitives and configurations are supported.
	
	• Xilinx VirtexII, VirtexII-Pro, VirtexII-pro-X, Spartan-3, Spartan-3E and Spartan-3A:
	  ○ 18-bit multiplier blocks[1]
	• Xilinx Virtex-4, Virtex-5 and Spartan-3A-DSP: DSP blocks in the following modes
	  ○ Multiplier[1]
	  ○ Multiply add/subtract
	  ○ Multiply-accumulate
	• Altera Cyclone II and Cyclone III:
	  ○ 18-bit multiplier blocks[1]
	• Altera Stratix, Stratix-GX, Stratix II, Stratix II-GX: DSP blocks in the following modes
	  ○ Multiplier[1]

Component	Description
	○ Multiply add/subtract ○ Multiply-accumulate Notes: 1- The compiler is able to tile multiple blocks to form wider multipliers, using dedicated routing where available. Automatic pipelining of RAM access By default the DK compiler uses an inverted version of the main Handel-C clock to drive on-chip synchronous memories. This allows it to conform to Handel-C's timing semantics of 1 clock cycle per assignment. This can potentially halve the maximum clock rate in some designs. The compiler can pipeline the accesses to on-chip SSRAMs if code is written in an appropriate manner. The effect is that the memory is driven by the main (non-inverted) Handel-C clock, potentially doubling the clock rate for the design. The transform applies to certain devices and configurations • Altera: EAB on Apex20K, Apex20KE and Apex20KC; EAB on Apex II except for single-port RAMs and true dual-port RAMs; M512 on Stratix and Stratix GX, except for true dual-port RAM's; M4K on Stratix, Stratix, Stratix GX, Stratix II, Cyclone, Cyclone II and Cyclone III, except for single-port RAMs and true dual-port RAMs. • Xilinx: BlockRAM on Virtex-II, Virtex-II-Pro, Virtex-4, Virtex-5, Spartan-3, Spartan-3E, Spartan-3A and Spartan-3ADSP. The Handel-C compiler may be used within the DK IDE or as a command line tool.
Handel-C simulator	Loads and executes a Handel-C / C++ / ANSI-C simulation.
Handel-C debugger	Provides interactive control of Handel-C simulations.
Online help system	Provides indexed, searchable information on DK and Handel-C

Table 1, IDE features

HANDEL-C DEBUGGER

The Handel-C debugger provides windows that show information when the operation of a compiled Handel-C program is simulated. The simulation steps the program through clock cycles, and allows the contents of any variables that are in scope to be viewed in the Variables window. Variables can also be selected for display in the Watch window. The call stack (the route by which a function has been called) is displayed in the Call Stack window. Clock cycles and current executing threads can be observed in the Clocks/Threads window.

Debug commands can be used to:

- Step through every line of code in a Handel-C program
- Set and remove breakpoints to segment the simulation
- Follow a selected processing thread or clock
- View the clock cycle count
- See how a function has been called
- Examine variables

CO-SIMULATION

Co-simulation with external HDL simulators, Instruction Set Simulators and other languages and tools is supported through the use of a co-simulation API. This API supports the development of a bridge between the DK Design Suite and any third party product that can also support this type of simulation.

The co-simulation API can also be used to create external test harnesses in any suitable programming language.

SUPPORTED FPGAS/PLDS

The following device families can be targeted using the EDIF output from the DK tool. The appropriate external place and route or fitter tool will be required.

- Altera APEX II
- Altera APEX 20K
- Altera APEX 20KE
- Altera APEX 20KC
- Altera Stratix and Stratix-GX
- Altera Stratix-II and Stratix-II-GX
- Altera Cyclone
- Altera Cyclone-II
- Altera Cyclone-III
- Xilinx Spartan
- Xilinx Spartan-II
- Xilinx Spartan-XL
- Xilinx Spartan-II E
- Xilinx Spartan-III

- Xilinx Spartan-III E
- Xilinx Spartan-III-A and Spartan-III-ADSP
- Xilinx Virtex
- Xilinx Virtex-E
- Xilinx Virtex-II
- Xilinx Virtex-II Pro
- Xilinx Virtex-4
- Xilinx Virtex-5

3RD PARTY MODELING TOOLS

HDL output is provided suitable for use by the following 3rd party simulation tools

- Model Technology ModelSim SE 6.1a
- Aldec Active-HDL 7.2

HDL OUTPUT

HDL output from the DK Design Suite is provided in generic style which can be used as input for 3rd party synthesis tools.

Note: Separate editing of the generated HDL output may be necessary to suit the user's tools, product or design flow. There may be situations where such modification is not feasible.

In addition HDL output styles are provided to specifically target the following synthesis tools

- Mentor Graphics' Precision Synthesis 2006a .92
- Xilinx XST 9.1
- Quartus II 6.1 HDL synthesis

PREREQUISITE SOFTWARE

The following third party software is required for the use of the DK Design Suite.

- One of
  - Microsoft Windows XP SP2
  - SuSE Linux 10.2
  - RedHat Linux Enterprise 4
- One or more of (depending on target FPGA/PLD devices)
  - Altera Quartus II 6.1
  - Xilinx ISE 9.1 (32 or 64 bit under Linux RedHat)

PREREQUISITE HARDWARE

The DK design suite requires the following hardware as a minimum:

- Pentium 3 600 MHz (or equivalent)
- 512Mb RAM,
- 500Mb of hard disk space
- CD-ROM drive
- Ethernet card (for software license)
- 1024 x 768 graphics (recommended)

However, we recommend an enhanced specification as follows:

- Pentium 4 3GHz (or equivalent)
- 2Gbyte RAM
- 1Gbyte hard disk space
- CD-ROM drive
- Ethernet card (for software license)
- 1024 x 768 graphics (recommended)

Note that large or complex designs may require a faster processor or increased RAM over and above the recommended specification.

PRODUCT PACKAGE FEATURES

Product Feature Group	DK Design Suite	Platform Development Package
DK Base – Graphical user interface, symbolic debugger and simulator	Yes	Yes
Handel-C compiler for simulation	Yes	Yes
Handel-C synthesis to hardware (See below for available outputs)	Yes	Yes
EDIF FPGA Synthesis output for supported Xilinx devices	Optional	No – (Except for device(s) on selected development boards(s))
EDIF FPGA Synthesis output for supported Altera devices	Optional	No – (Except for device(s) on selected development boards(s))
VHDL Output	Optional	No
Verilog Output	Optional	No

Table 2, Comparison of product package features

LICENSING

The DK Design suite is licensed under the terms of an End User Software License. Details of this may be obtained from the Agility.

WARRANTY

Agility warrants that for a period of 90 days from the delivery of the Software and the Key to the Licensee, the Software will operate substantially in accordance with this specification and that the media on which the Software is delivered will be free from defects on delivery. If, within 90 days of receipt of the Software by the Licensee the medium on which it is supplied is faulty and returned to Agility, Agility shall replace it without charge.

SUPPORT

Support is provided through the Agility website at http://www.agilityds.com/support or email at support@agilityds.com. Support is included with 1 year and 3 year licenses and is priced separately for perpetual licenses.

Agility
Algorithm to Implementation. *Fast*.

Corporate Headquarters

Agility Design Solutions Inc
1076A East Meadow Circle,
Palo Alto,
Calif. 94303.
United States.
Tel: + 1 650 846-2555
Fax: + 1 650 846-2557

US Sales Office

Agility Design Solutions Inc
Suite 185, Lakeview Plaza,
4516 Seton Center Parkway,
Austin, TX 78759,
United States.
Tel: +1 512 795 8170
Fax: +1 512 795 8167

UK Development & Sales Office

Agility Design Solutions
66 Milton Park,
Abingdon,
OX14 4RX,
United Kingdom.
Tel: +44 1235 863656
Fax: +44 1235 863648

Japan Sales Office

Agility Japan KK
YBP West Tower, 11F,
134 Godo-cho, Hodogaya-ku,
Yokohama,
240-0005 JAPAN.
Tel: +81 (0) 45 331 0218
Fax: +81 (0) 45 331 0433

Copyright 2008 Agility Design Solutions. All rights reserved.