Table of Contents

TABLE OF CONTENTS ................................................................................................................................. 2
INTRODUCTION ............................................................................................................................................. 2
DISTRIBUTION DIRECTORY STRUCTURE .................................................................................................... 2
HARDWARE GETTING STARTED .................................................................................................................... 2
  Generic SoC .................................................................................................................................................... 3
  Spartan-3E Starter Kit ................................................................................................................................ 3
  Spartan-3 Starter Kit .................................................................................................................................... 5
  Memec Spartan-IIE LC Kit + P160 Communication module ........................................................................... 7
  Stratix1 S10 Development Kit .................................................................................................................... 8
SOFTWARE GETTING STARTED ................................................................................................................... 11
  Banner .......................................................................................................................................................... 11
  Timer ........................................................................................................................................................... 12
TROUBLE SHOOTING .................................................................................................................................. 13

Introduction

This guide is divided into two parts a) Hardware getting started, describes the predefined SoCs available with the distribution b) the Software getting started guide, describes the software examples provided with the distribution.

Distribution Directory structure

Hardware Getting started.

A typical MANIK SoC consists of the CPU and one or more peripheral. The MANIK CPU has one WISHBONE Master port, and each peripheral has one WISHBONE slave port. The peripherals are connected to the CPU via a SHARED Bus. Each peripheral (Slave port) is assigned an unique base address, it is recommended that I/O peripherals such as UART, EASYMac etc be assigned base addresses with the highest order bit set (0x80000000, 0x80000100 etc), this will ensure that they are placed in the un-cached memory region. The MANIK SoC uses a Registered feedback WISHBONE Cycle, Classic cycle option as described in Chapter 4 of the WISHBONE specification manual.
The figure above shows a small Manik SoC. The distribution contains four Prebuilt SoC.

**Generic SoC**

Generic, located in $(MANIK_BASE)/vhdl/socs/generic, this is a sample SoC, built for simulation and testing purposes.

a. **manik_soc.vhd** — contains the top level wiring of the peripherals & the processor.

b. **manikconfig.vhd** — has the configuration parameters for the CPU & other peripherals.

<table>
<thead>
<tr>
<th>Peripheral</th>
<th>Base Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-Chip RAM</td>
<td>0x00000000 - 0x1FFF</td>
</tr>
<tr>
<td>SRAM</td>
<td>0x10000000 - 0x1fffffff</td>
</tr>
<tr>
<td>UART</td>
<td>0x80000000 - 0x80000001</td>
</tr>
<tr>
<td>GPIO</td>
<td>0x80001000 - 0x80001003</td>
</tr>
<tr>
<td>EASYMAC</td>
<td>0x80020000 - 0x80020008</td>
</tr>
</tbody>
</table>

**Spartan-3E Starter Kit**

S3Estarterkit, Located in the directory $(MANIK_BASE)/vhdl/socs/s3Estarterkit. The SoC is for the Spartan-3E starter kit sold by Xilinx.
A manik_soc.vhd – contains the top level wiring of the peripherals & the processor.

B manikconfig.vhd – has the configuration parameters for the CPU & other peripherals.

C s3Estarterkit.ucf – contains Pin Assignments & clock net definitions.

D $(MANIK_BASE)/c_system_lib/s3EstarterKit/board_linker_script – contains the linker script that software programs can use to create executable images for this platform.

<table>
<thead>
<tr>
<th>Peripheral</th>
<th>Base Address</th>
<th>Block RAMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU Cache</td>
<td>0x00000000-0x00001FFF</td>
<td>6 (3-DCache, 3-ICache)</td>
</tr>
<tr>
<td>Debug ROM Monitor</td>
<td>0x04000000-0x05FFFFFF</td>
<td>1</td>
</tr>
<tr>
<td>DDR Controller</td>
<td>0x80000000-0x8000001</td>
<td></td>
</tr>
<tr>
<td>UART</td>
<td>0x80010000-0x8001003</td>
<td></td>
</tr>
<tr>
<td>GPIO</td>
<td>0x80020000-0x80020008</td>
<td>2</td>
</tr>
</tbody>
</table>

IMPLEMENTATION RESULTS

The SoC was synthesized, with 4K DCache & 4K ICache. No Hardware Break/Watch points. The implementation scripts for this board are available in the directory $(MANIK_BASE)/synth/s3Estarterkit/scripts. The bash script do_xst uses the Xilinx synthesis tool XST, and the script do_synplify uses Synplify (PRO).

Results using XST

Logic Utilization:
Number of Slice Flip Flops: 1,830 out of 9,312 19%
Number of 4 input LUTs: 2,973 out of 9,312 31%
Logic Distribution:
Number of occupied Slices: 1,999 out of 4,656 42%
Number of Slices containing only related logic: 1,999 out of 1,999 100%
Number of Slices containing unrelated logic: 0 out of 1,999 0%
*See NOTES below for an explanation of the effects of unrelated logic
Total Number 4 input LUTs: 3,138 out of 9,312 33%
Number used as logic: 2,973
Number used as a route-thru: 57
Number used for Dual Port RAMs: 64
(Two LUTs used per Dual Port RAM)
Number used as Shift registers: 44
Number of bonded IOBs: 72 out of 232 31%
IOB Flip Flops: 38
Number of ODDR2s used: 22
Number of DDR_ALIGNMENT = NONE 22
Number of Block RAMs: 9 out of 20 45%
Number of GCLKs: 6 out of 24 25%
Number of DCMs: 3 out of 4 75%
Number of MULT18X18SIOs: 3 out of 20 15%

Number of RPM macros: 5

Results using Synplify(PRO)

Logic Utilization:
- Number of Slice Flip Flops: 1,867 out of 9,312 20%
- Number of 4 input LUTs: 2,850 out of 9,312 30%

Logic Distribution:
- Number of occupied Slices: 1,975 out of 4,656 42%
  Number of Slices containing only related logic: 1,975 out of 1,975 100%
  Number of Slices containing unrelated logic: 0 out of 1,975 0%

Total Number 4 input LUTs: 2,971 out of 9,312 31%
- Number used as logic: 2,850
- Number used as a route-thru: 52
- Number used for Dual Port RAMs: 64
  (Two LUTs used per Dual Port RAM)
- Number used as Shift registers: 5
- Number of bonded IOBs: 73 out of 232 31%
- Number of ODDR2s used: 22
  Number of DDR_ALIGNMENT = NONE 22
- Number of Block RAMs: 9 out of 20 45%
- Number of GCLKs: 6 out of 24 25%
- Number of DCMs: 3 out of 4 75%
- Number of MULT18X18SIOs: 3 out of 20 15%

Number of RPM macros: 5

Spartan-3 Starter Kit

s3starterkit, Located in the directory $(MANIK_BASE)/vhdl/socs/s3starterkit. The SoC is for the Spartan-3 starter kit sold by Digilent/Xilinx. The implementation scripts for this board are available in the directory $(MANIK_BASE)/synth/s3starterkit/scripts. The bash script do_xst uses the Xilinx synthesis tool XST, and the script do_synplify uses Synplify (PRO).

- manik_soc.vhd – contains the top level wiring of the peripherals & the processor.
- manikconfig.vhd – has the configuration parameters for the CPU & other peripherals.
- s3starterkit.ucf – contains Pin Assignments & clock net definitions.
- $(MANIK_BASE)/c_system_lib/s3starterKit/board_linker_script – contains the linker script that software programs can use to create executable images for this platform.
### IMPLEMENTATION RESULTS

The SoC was synthesized, with **4K DCache & 4K ICache**. No Hardware Break/Watch points

#### Results using XST

<table>
<thead>
<tr>
<th>Logic Utilization:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slice Flip Flops:</td>
<td>1,239 out of 3,840 32%</td>
</tr>
<tr>
<td>Number of 4 input LUTs:</td>
<td>2,178 out of 3,840 56%</td>
</tr>
</tbody>
</table>

#### Logic Distribution:

| Number of occupied Slices: | 1,380 out of 1,920 71% |
| Number of Slices containing only related logic: | 1,380 out of 1,380 100% |
| Number of Slices containing unrelated logic: | 0 out of 1,380 0% |

*See NOTES below for an explanation of the effects of unrelated logic*

**Total Number 4 input LUTs:** 2,255 out of 3,840 58%

##### Results using Synplify(PRO)

<table>
<thead>
<tr>
<th>Logic Utilization:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Slice Flip Flops:</td>
<td>1,250 out of 3,840 32%</td>
</tr>
<tr>
<td>Number of 4 input LUTs:</td>
<td>1,980 out of 3,840 51%</td>
</tr>
</tbody>
</table>

#### Logic Distribution:

| Number of occupied Slices: | 1,315 out of 1,920 68% |
| Number of Slices containing only related logic: | 1,315 out of 1,315 100% |
| Number of Slices containing unrelated logic: | 0 out of 1,315 0% |

*See NOTES below for an explanation of the effects of unrelated logic*

**Total Number 4 input LUTs:** 2,071 out of 3,840 53%

Number used as logic: 2,178

Number used as a route-thru: 2

Number used for Dual Port RAMs: 64

(Two LUTs used per Dual Port RAM)

Number used as Shift registers: 11

Number of bonded IOBs: 70 out of 173 40%

IOB Flip Flops: 73

Number of Block RAMs: 7 out of 12 58%

Number of MULT18X18s: 3 out of 12 25%

Number of GCLKs: 4 out of 8 50%

Number of DCMs: 1 out of 4 25%

Number of RPM macros: 5
Memec Spartan-IIE LC Kit + P160 Communication module

MemecS2ELCKit, located in $(MANIK_BASE)/vhdl/soc/MemecS2ELCKit. This SoC is for the Memec Spartan-IIE LC Development Kit with a P160 Communication module. The implementation scripts for this board are available in the directory $(MANIK_BASE)/synth/MemecS2ELCKit/scripts. The bash script do_xst uses the Xilinx synthesis tool XST, and the script do_synplify uses Synplify (PRO).

- manik_soc.vhd – contains the top level wiring of the peripherals & the processor.
- manikconfig.vhd – has the configuration parameters for the CPU & other peripherals.
- MemecS2ELCKit.ucf – contains Pin Assignments & clock net definitions.
- $(MANIK_BASE)/c_system_lib/MemecS2ELCKit/board_linker_script – contains the linker script that software programs can use to create executable images for this platform.

<table>
<thead>
<tr>
<th>Peripheral</th>
<th>Base Address</th>
<th>Block RAMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debug Monitor ROM</td>
<td>0x00000000 – 0x1FFF</td>
<td>2</td>
</tr>
<tr>
<td>SDRAM Controller</td>
<td>0x4000000 – 0x5fffffff</td>
<td>0</td>
</tr>
<tr>
<td>UART (used by debug Monitor)</td>
<td>0x80000000 – 0x800000001</td>
<td>0</td>
</tr>
<tr>
<td>GPIO</td>
<td>0x80010000 – 0x80010003</td>
<td>0</td>
</tr>
<tr>
<td>EASYMAC</td>
<td>0x8020000 – 0x8020008</td>
<td>8</td>
</tr>
</tbody>
</table>

IMPLEMENTATION RESULTS

The Entire SoC was synthesized, without Hardware Breakpoints, Hardware Watch points, No Cache. The clock is set at 50 MHz.
Results using XST

Logic Utilization:
Number of Slice Flip Flops: 1,882 out of 6,144 30%
Number of 4 input LUTs: 2,965 out of 6,144 48%

Logic Distribution:
Number of occupied Slices: 1,933 out of 3,072 62%
Number of Slices containing only related logic: 1,933 out of 1,933 100%
Number of Slices containing unrelated logic: 0 out of 1,933 0%

*See NOTES below for an explanation of the effects of unrelated logic

Total Number 4 input LUTs: 3,109 out of 6,144 50%

- Number used as logic: 2,965
- Number used as a route-thru: 55
- Number used for Dual Port RAMs: 64
  (Two LUTs used per Dual Port RAM)
- Number used as 16x1 ROMs: 16
- Number used as Shift registers: 9
- Number of bonded IOBs: 83 out of 325 25%
- IOB Flip Flops: 64
- Number of Block RAMs: 10 out of 16 62%
- Number of GCLKs: 4 out of 4 100%
- Number of GCLKIOBs: 1 out of 4 25%

Number of bonded IOBs: 83 out of 325 25%

Number of bonded IOBs: 83 out of 325 25%

Results using Synplify(PRO)

Logic Utilization:
Number of Slice Flip Flops: 1,902 out of 6,144 30%
Number of 4 input LUTs: 2,539 out of 6,144 41%

Logic Distribution:
Number of occupied Slices: 1,786 out of 3,072 58%
Number of Slices containing only related logic: 1,786 out of 1,786 100%
Number of Slices containing unrelated logic: 0 out of 1,786 0%

*See NOTES below for an explanation of the effects of unrelated logic

Total Number 4 input LUTs: 2,684 out of 6,144 43%

- Number used as logic: 2,539
- Number used as a route-thru: 63
- Number used for Dual Port RAMs: 64
  (Two LUTs used per Dual Port RAM)
- Number used as 16x1 ROMs: 16
- Number used as Shift registers: 2
- Number of bonded IOBs: 84 out of 325 25%
- Number of Block RAMs: 10 out of 16 62%
- Number of GCLKs: 4 out of 4 100%
- Number of GCLKIOBs: 1 out of 4 25%

Number of bonded IOBs: 84 out of 325 25%

Number of bonded IOBs: 84 out of 325 25%

**Stratix1 S10 Development Kit**

Stratix1 S10 Development Kit, Located in the directory
$({MANIK_BASE})/vhdl/socs/stratix1S10kit. The SoC is for the Stratix-1 development kit sold by Altera.

- **manik_soc.vhd** – contains the top level wiring of the peripherals & the processor.
- **manikconfig.vhd** – has the configuration parameters for the CPU & other peripherals.
o $(MANIK_BASE)/c_system_lib/stratix1S10kit/board_linker_script – contains the linker script that software programs can use to create executable images for this platform.

<table>
<thead>
<tr>
<th>Peripheral</th>
<th>Base Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>On-Chip RAM</td>
<td>0x00000000 – 0x1FFF</td>
</tr>
<tr>
<td>SRAM</td>
<td>0x00100000 – 0x001FFFFF</td>
</tr>
<tr>
<td>SDRAM</td>
<td>0x04000000 – 0x04FFFFFF</td>
</tr>
<tr>
<td>UART</td>
<td>0x80000000 – 0x80000001</td>
</tr>
<tr>
<td>GPIO</td>
<td>0x80010000 – 0x80010003</td>
</tr>
</tbody>
</table>

**IMPLEMENTATION RESULTS**

The SoC was synthesized, with 4K DCache & 4K ICache. No Hardware Break/Watch points. The directory $(MANIK_BASE)/synth/stratix1S10kit contains Quartus-II project for implementation.

+--------------------------------------------------------------------+
| ; Flow Summary                                                      |
| +--------------------------+-----------------------------------------+|
| ; Flow Status             ; Analyzed – Wed Aug 16 21:46:46 2006          ;|
| ; Quartus II Version      ; 6.0 Build 178 04/27/2006 SJ Web Edition ;|
| ; Revision Name           ; manik2top                                  |
| ; Top-level Entity Name   ; manik_soc                                   |
| ; Family                  ; Stratix                                    |
| ; Device                  ; EP1S10F780C6                                 |
| ; Timing Models           ; Final                                      |
| ; Met timing requirements ; Yes                                       |
| ; Total logic elements    ; 2,805 / 10,570 ( 27 % )                         |
| ; Total pins              ; 126 / 427 ( 30 % )                             |
| ; Total virtual pins      ; 0                                       |
| ; Total memory bits       ; 111,616 / 920,448 ( 12 % )                         |
| ; DSP block 9-bit elements; 8 / 48 ( 17 % )                              |
| ; Total PLLs              ; 1 / 6 ( 17 % )                               |
| ; Total DLLs              ; 0 / 2 ( 0 % )                                |
+--------------------------------------------------------------------+

**Lattice HPE-MINI (ECP) Development Kit**

Lattice_HPE_mini_ECP, is located in $(MANIK_BASE)/vhdl/soc/Lattice_HPE_mini_ECP. This SoC is Lattice HPE mini (ECP) development Kit. The ispLever project for this SoC is present in the directory $(MANIK_BASE)/synth/Lattice_HPE_mini_ECP. The $(MANIK_BASE)/vhdl/soc/Lattice_HPE_mini_ECP directory contains the following files
- **manik_soc.vhd** – contains the top level wiring of the peripherals & the processor.

- **manikconfig.vhd** – has the configuration parameters for the CPU & other peripherals.

- **Lattice_HPE_mini_ECP.lpf** – contains Pin Assignments & clock net definitions.

- **$(MANIK_BASE)/c_system_lib/Lattice_HPE_mini_ECP/board_linker_script** – contains the linker script that software programs can use to create executable images for this platform.

- **mrem.vhd** - This file contains the **manikremote** debug ROM. This file is generated by the **ipExpress** tool using **$(MANIK_BASE)/manikremote/manikremote.mem** file as input.

### Peripheral | Base Address | Block RAMS
--- | --- | ---
Debug ROM Monitor | 0x00000000 – 0x1FFF | 2
SRAM Controller | 0x0100000 – 0x01fffff | 0
UART (used by debug Monitor) | 0x80000000 – 0x800000001 | 0
GPIO | 0x80010000 – 0x80010003 | 0
EASYMAC | 0x80020000 – 0x80020008 | 8

### IMPLEMENTATION RESULTS

The **Entire SoC** was synthesized, **without** Hardware Breakpoints, Hardware Watchpoints, **4K DCache & 4K ICache**. The clock is set at 50 MHz.

Number of registers: 1681
PFU registers: 1640
PIO registers: 41
Number of SLICEs: 1900 out of 16384 (12%)
SLICEs(logic/ROM): 1836 out of 12288 (15%)
SLICEs(logic/ROM/RAM): 64 out of 4096 (2%)
As RAM: 64
As Logic/ROM: 0
Number of logic LUT4s: 2709
Number of distributed RAM: 64 (128 LUT4s)
Number of ripple logic: 147 (294 LUT4s)
Number of shift registers: 0
Total number of LUT4s: **3131**
Number of external PIOs: 0 out of 360 (24%)
Number of PIO IDDR/ODDR: 0
Number of PIO FIXEDDELAY: 0
Software Getting started.

The MANIK distribution includes three example software projects.

Banner

The banner example illustrates the communication with the debugger. It also shows how to use the debug ROM Monitor to output text to the debugger console. To execute this sample application.

- Download Bitstream to the FPGA
- Start the debugger

`manik-elf-gdb banner.elf`

- Connect, Load and execute the program into the ram

```
(gdb) tar manikrem /dev/com4 115200
Remote manikrem connected to /dev/com4
(gdb) load
Loading section ram_section, size 0xc4b4 lma 0x4000000
Start address 0x4000000, load size 50356
Transfer rate: 80569 bits/sec, 508 bytes/write.
(gdb) continue
Continuing.
```

```
@@@     @@@       @      @@@    @@@ @@@@@@ @@@@   @@@@
@@     @@        @       @@     @    @@    @@     @
@@@   @@@       @@@      @@@    @    @@    @@    @
@@@   @@@       @@@      @ @@   @    @@    @@   @
@ @@ @ @@      @  @@     @ @@   @    @@    @@  @@
@ @@ @ @@      @  @@     @  @@  @    @@    @@ @ @@
@ @@@  @@     @    @@    @   @@ @    @@    @@@  @@
@  @@  @@     @@@@@@@    @   @@ @    @@    @@    @@
@  @   @@    @      @@   @    @@@    @@    @@    @@
@@@    @@@@ @@@@    @@@@ @@@    @@  @@@@@@ @@@@   @@@@
@@@@@@@@
@@   @@
@@    @@
@@     @@   @@@@@   @@@ @@ @@@      @@@
@@     @@  @     @   @@@ @@  @@   @@   @@
@@     @@ @@     @@  @@  @@  @@  @@     @@
@@     @@ @@@@@@@@@  @@  @@  @@  @@     @@
@@     @@ @@         @@  @@  @@  @@     @@
@@    @@  @@      @  @@  @@  @@  @@     @@
@@   @@    @@    @@  @@  @@  @@   @@   @@
```

```
The timer test program located in the directory `$MANIK_BASE/examples/timer` illustrates the usage of the built-in timer. The program sets the timer to generate an interrupt every 5 seconds, it also illustrates the usage of the power-down mode of the processor.

```
$ manik-elf-gdb ttest.elf
GNU gdb 6.3
Copyright 2004 Free Software Foundation, Inc.
GDB is free software, covered by the GNU General Public License, and you are
welcome to change it and/or distribute copies of it under certain conditions.
Type "show copying" to see the conditions.
There is absolutely no warranty for GDB.  Type "show warranty" for details.
This GDB was configured as "--host=i686-pc-cygwin --target=manik-elf"...
(gdb) tar manikrem /dev/com4 115200
Remote manikrem connected to /dev/com4
(gdb) load
Loading section ram_section, size 0x74b8 lma 0x4000000
Start address 0x4000000, load size 29880
Transfer rate: 79680 bits/sec, 506 bytes/write.
(gdb) c
Continuing.
in sig_handler()
5 more seconds elapsed. Total 5
in sig_handler()
5 more seconds elapsed. Total 10
in sig_handler()
5 more seconds elapsed. Total 15
in sig_handler()
5 more seconds elapsed. Total 20
in sig_handler()
5 more seconds elapsed. Total 25
in sig_handler()
5 more seconds elapsed. Total 30
Program received signal SIGINT, Interrupt.
main () at ttest.c:25
25   total += 5;
(gdb)
```

The debug session can be terminated by hitting a CRTL-C.

**LWIP (Light Weight IP).**

The MANIK port for LWIP can be downloaded separately from the core distribution. The lwip port for MANIK implements an echo server & a simple http server. Once downloaded the source has to be built, port will work for any SoC with an EASYMAC Ethernet MAC. The core LWIP stack has NOT been modified. The MANIK related files can be found in the directory `$(lwip_dir)/contrib./ports/manik`. 
- easymacif.c & easymacif.h: contains the low level interface to the EasyMac Ethernet MAC.

- Lwipopts.h – contains the lwip configuration options.

The Makefile in this directory can be used to build the lwip application. Default configuration the software sets the IP address to (192.168.2.99), Gateway IP Address to (192.168.2.88) & the network Mask to (255.255.255.0), these values are defined in the file main.c. The LWIP_DHCP options can be turned on in lwipopts.h.

Once the application is built the application (echop) can be downloaded to the target using the debugger (as described in the previous two examples). Use a web browser to connect to the IP address (192.168.2.99) to get a display of the webpage.

**Trouble shooting**

The debugger (manik-elf-gdb) will connect to the target (FPGA) via the serial port. The serial port communication problem is frequently the reason for problems. To verify that the serial connection to the target is working correctly,

- Start a terminal communication program (such as Teraterm or Hyperterm). Set the communication parameters for the terminal emulation program to the serial port on the PC that is connected to the serial port of the FPGA development board (COM1, COM2 .. etc). Set the baud rate to 115200 (For the XILINX boards) (38400 for the Altera board), data bits to 8bits & Stop bits to 1.

- Download the bitstream to the FPGA.

- Once the download is complete the letter ‘X’ should appear on the terminal communication window.

- Press ‘o’ the Debug Rom Monitor should respond with a ‘k’.

- Now disconnect/close the terminal emulation window and start the debugger (manik-elf-gdb). At the prompt type in the command `target manikrem /dev/comX <baudrate>` where /dev/comX is /dev/com1 if the serial port is COM1, /dev/com2 if serial port is COM2 and so on. The <baudrate> is 115200 for the Xilinux boards & 38400 for the Altera board.