Chapter 7

Performance Analysis

7.1 Design Verification

A simulator was made to verify the microprocessor and the fitness evaluator. The verification was done by executing a program in the simulator, then the simulated result was compared to the result from the actual hardware. A seed, used to produce random numbers, was fixed. The program performed one generation of GA, then a sum of memory was calculated. The sum of memory in the simulator must be equal to the sum of memory in the actual hardware. This is to ensure that the "state" of the computation of the simulator is equivalent to that of the actual hardware so that the performance analysis could be done in the simulator.

7.2 Performance Analysis

A profile of software-based GA, running on 200 MHz PentiumPro with Linux OS, is shown in Table 7.1. It can be seen that to mimic a practical sequential circuit, we must use a great number of long input/output sequences. As a result, the evaluation time increased drastically with the circuit size due to the large number of sequences needed to yield high correctness percentage.

The profile in Table 7.1 shows that the fitness evaluation was a major bottleneck of GA. Accordingly, the hardware contributed to speedup the fitness evaluation is reasonable. A comparison of software and hardware evaluator is shown in Table 7.2. The evaluation time of serial adder, in software, was profiled. In hardware, the evaluation time was calculated from the fitness evaluator (EV) operating at its maximum frequency (8 MHz).

The result in Table 7.2 shows that the fitness evaluator was little slower than the software running on PentiumPro. The fitness evaluator is not very fast due to two main

Table 7.1: Percentage of evaluation time.

Circuit	Number of States	Number of Sequences	Sequence Length	Evaluation Time
Serial Adder	2	10	100	8.9%
0101 Detector	4	100	100	45.0%
Modulo-4 Counter	4	100	100	53.6%
Reversible 8-counter	8	1,000	100	69.1%
Reversible 8-counter	8	10,000	100	95.6%

Table 7.2: A comparison of software and hardware evaluator (serial adder).

Number of Sequences	Sequence Length	Evaluation Time (ms)	
		Software	Hardware
10	100	0.06	0.25
100	100	0.65	2.50
1,000	100	13.50	25.00
10,000	100	136.94	250.00

reasons. First, the PentiumPro operates at very high frequency (200 MHz) while the maximum frequency of the FPGA in this experiment is 20 MHz. Second, the bottleneck of the memory limits the operational frequency at 8 MHz. Actually, the evaluator can operate at higher frequency. This is not surprising since the evaluator uses very little resources (about 5,000 gates) whereas the commercial CPU uses millions of transistors. Although the performance of the fitness evaluator is moderate, for large problem the evaluator can be parallelised. By using a number of evaluators in parallel, the linear speedup is achievable.

Next, the performance of the microprocessor will be analysed. In the design stage, the number of registers was set at 8. In the synthesis stage, number of registers was reduced to 4 due to the reason that the size of FPGA was not sufficient. The first assembly program was well-optimised for 8-register processor, then the 8-register program was simply translated to the 4-register program instruction by instruction. We did not put much effort to optimise the 4-register program. The translation of 8-register to 4-register program drastically dropped the performance (see Table 7.3).

The result of executing 4-register and 8-register programs, done in the simulator, is

shown in Table 7.3. In the 4-register program, the number of load/store instructions increased due to the small number of registers. For the same reason, the number of push/pop instructions, used to load/store the program variables to CPU stack, increased. It can be seen that the number of registers greatly affected the number of executed instructions. By using a little larger FPGA, the performance of the microprocessor can be significantly improved.

The comparison of PentiumPro and the custom microprocessor is shown in Table 7.4. The microprocessor and the fitness evaluator were separately analysed, therefore the execution time did not included the fitness evaluation. Table 7.3 was used to calculate the execution time of the custom processor. For PentiumPro, the execution time was profiled, then subtracted by the evaluation time. The result shows that PentiumPro is 200 times faster than the 8-register processor and 1,400 times faster than the 4-register processor. The execution time of GA on the custom processor was slower than the PentiumPro due to the following reasons.

- The custom processor executes the instructions in sequence while the commercial CPU uses an aggressive pipeline.
- The number of registers is too small. This causes the extensive load/store instructions.
- Our processor executes at 6 MHz whereas PentiumPro executes at 200 MHz. The FPGA cannot operate at high frequency as ASIC technology.
- The register allocation is not optimal. It can be done better by using a good compiler.

The performance of the custom processor is lower than PentiumPro; however, the performance depends on the the available resources. The custom microprocessor is considerably efficient according to the FPGA sizing of 10,000 equivalent gates.

The performance of the microprocessor and the fitness evaluator can be significantly improved by using the latest FPGA technology. The Virtex FPGA (Xilinx, 2000), which is a high-speed, high-density FPGA sizing of 3.2M equivalent gates and operating at 200 MHz, could be used. The memory bandwidth can be increased to 200 MHz using ZBT SRAM (Zero Bus Turnaround SRAM) – a next generation of SyncBurst SRAM specifically used for PC cache applications such as Pentium and PowerPC.

Table 7.5 shows the overall performance of three versions: SW, HW(XC4010), and HW(Virtex). It can be seen that SW took 10 min. 50 sec. while the HW(XC4010) took 24 min. 40 sec. to accomplish the same task. Although the fitness evaluation is a major bottleneck consuming about 90% of total time, the HW(XC4010) containing a fitness evaluator cannot outperform the SW. The use of Virtex device and ZBT SRAM can speedup both the microprocessor and the fitness evaluator. It can be seen that the total execution time of HW(Virtex) was 10.77 times faster than SW.

Indeed the enormous size of Virtex device could provide a pipeline and an adequate number of registers for the microprocessor. This considerably enhance the performance of the microprocessor. However, Table 7.1 shows that the evaluation time increased dramatically with the size of input/output sequences. Accordingly, the contribution to improve the microprocessor does not yield much benefit for larger problems. A parallel of fitness evaluators will helpfully reduce the evaluation time. Supposing the Virtex device operates at 200 MHz, and the evaluation time decreases linearly with the number of fitness evaluators. The comparison of the use of fitness evaluators in parallel is shown in Table 7.6. It can be seen that the evaluation time (EV) of HW(Virtex) with 8 EVs was 8.0 times faster than the HW(Virtex) with single evaluator. This reduced the overall performance (GA+EV) to 18 sec. that was 3.4 times faster than the unparalleled version.

By using the state-of-the-art FPGA, the HW (Virtex) with 8 EVs could perform 36 times faster than the software(SW) running on a conventional workstation.

Table 7.3: A comparison of 8-register and 4-register program.

Instruction	The number of instructions executed	
	8-register program	4-register program
JEQ	1,563,564	1,563,564
JNE	12,032	2,569,992
JGR	266,530	266,530
JLE	1,803,465	1,803,465
JMP	896	896
JSR	373,057	373,057
CIJ	2,557,960	0
RES	373,057	373,057
LDC	4,119,586	45,969,859
LDD	0	0
STD	0	0
LDR	2,800,706	30,321,047
STR	66,249	19,875,840
LDX	5,460,812	5,460,812
STX	1,294,820	1,294,820
SEV	896	896
REV	897	897
LFH	896	896
LFN	896	896
HLT	1	1
SED	1	1
MOV	317,385	104,407
CMP	2,082,761	4,640,721
COM	257	257
SFL	174,525	174,525
SFR	3,747,799	3,747,799
PSH	2,322,942	24,461,722
POP	2,322,942	24,461,722
POT	2	2
INC	1,499,715	4,057,675
DEC	0	0
CLR	0	0
ADD	1,395,780	1,395,780
AND	468,628	468,628
ORR	910,792	910,792
XOR	911,360	911,360
STI	7,168	7,168
RND	10,900	10,900
AD3	0	0
Total	36,869,277	175,229,984

Table 7.4: A comparison of PentiumPro and the custom microprocessor (serial adder).

PentiumPro	Custom microprocessor	
	(8 registers)	(4 registers)
0.28 sec.	56 sec.	6 min. 32 sec.

Table 7.5: A comparison of overall performance
(serial adder, sequence length = 100, number of sequences = 10,000).

Version	GA	EV	GA + EV
SW	0:29 min.	10:21 min.	10:50 min.
HW(XC4010)	6:32 min.	18:08 min.	24:40 min.
HW(Virtex)	0:12 min.	0:49 min.	1:01 min.

GA denoted the execution time of GA except the fitness evaluation
EV denoted the evaluation time.
SW denoted the software version of GA.
HW(XC4010) denoted the custom hardware running on XC4010 device.
HW(Virtex) denoted the custom hardware running on Virtex device.

Table 7.6: A comparison of the use of fitness evaluators in parallel
(serial adder, sequence length = 100, number of sequences = 10,000).

Version	GA	EV	GA + EV
HW(Virtex) with 1 EV	0:12 min.	0:49 min.	1:01 min.
HW(Virtex) with 2 EVs	0:12 min.	0:25 min.	0:37 min.
HW(Virtex) with 4 EVs	0:12 min.	0:12 min.	0:24 min.
HW(Virtex) with 8 EVs	0:12 min.	0:06 min.	0:18 min.

HW(Virtex) denoted the custom hardware running on Virtex device.