Hardware Mapping of Critical Paths of
a GaAs Core Processor for
Solid Modelling Accelerator

by

Song Cui, B. E., M. E.

A thesis submitted for the degree of

Doctor of Philosophy

in the Centre for Gallium Arsenide VLSI Technology

Department of Electrical and Electronic Engineering

The University of Adelaide

January 1996
Abstract vii

Statement of Originality ix

Acknowledgements x

List of Author's Related Publications xi

1 Concepts of the Solid Modelling Accelerator ......................... 1

1.1 Solid Modelling ......................................................... 2
  1.1.1 Introduction ...................................................... 2
  1.1.2 The Primary Solid Representation Schemes .................... 3
  1.1.3 Characterization of a B-Rep Solid Modelling System ......... 5

1.2 Solid Modelling Accelerator Using Unified Technology ......... 8
  1.2.1 Unified CMOS/BiCMOS/GaAs Technology ...................... 8
  1.2.2 The Architecture of the Solid Modelling Accelerator ........ 9
  1.2.3 Partitioning ..................................................... 11

1.3 The Scope of this thesis .............................................. 11

2 GaAs Technology and Logic Circuit Design ......................... 14

2.1 Gallium Arsenide Technology ....................................... 15
  2.1.1 Introduction ...................................................... 15
  2.1.2 Advantages of GaAs ............................................ 15
  2.1.3 Comparison between GaAs and Silicon ....................... 15

2.2 GaAs Logic Families .................................................. 17
  2.2.1 Introduction ...................................................... 17
  2.2.2 Direct-Coupled FET Logic (DCFL) ............................ 20
  2.2.3 Source-follower DCFL (SDCFL) ............................... 22
  2.2.4 Super Buffer FET Logic (SBFL) ............................... 24
  2.2.5 Two-Phase Dynamic FET Logic (TDFL) ....................... 25

2.3 Design Considerations For Logic Families ....................... 28
  2.3.1 Definition of Design Parameters ............................. 28
    Noise Margin 28
2.3.2 DCFL Optimization .............................................. 31
2.3.3 SDCFL Optimization ........................................... 34
2.3.4 SBFL Optimization .............................................. 35
2.3.5 Performance Comparison ...................................... 36
2.3.6 TDFL Optimization .............................................. 37

2.4 Interconnections ..................................................... 38
2.4.1 Interconnection Analysis ...................................... 38
  coplanar waveguide (CPW) 39
  coplanar striplines (CPS) 42
  microstrip line 43
  stripline 44

2.4.2 Power Supply and Ground Considerations .................. 44
  Resistance: ir drop 44
  Electromigration: current density limitations 45
  Inductance: LdI/dt voltage variation 46

2.4.3 Propagation Delay .............................................. 46
2.4.4 Crosstalk ......................................................... 47

2.5 Summary ............................................................. 48

3 An 8-bit Serial Dynamic/Static Divider .......................... 50
3.1 The Algorithm of The Serial Divider ........................... 51
  3.1.1 Subtractive Division ........................................... 51
  Restoring Division 51
  Nonrestoring Division 53
  Higher-Radix Subtractive Division 54

  3.1.2 Multiplicative Division ...................................... 55
  Division by Series Expansion 55
  Newton-Raphson Iteration 56
3.2 Analysis of Adder Designs for GaAs VLSI .................60
  3.2.1 Ripple-carry Adder .....................................60
  3.2.2 Carry Look-ahead Adder .................................61
  3.2.3 Brent & Kung Algorithm (Binary Carry Look-ahead Adder) ......64
  3.2.4 Carry Select Adder .....................................67
  3.2.5 Carry-Skip Adder .......................................68
  3.2.6 Performance Comparison of Different Adders for GaAs VLSI ...70

3.3 An 8-bit Serial Dynamic/Static Divider ......................72
  3.3.1 The Structure of the 8-bit Divider .........................72
  3.3.2 The Adder/Subtractor ...................................72
  3.3.3 TDFL Based Registers ...................................74
      Why Use TDFL Registers? 74
      TDFL Shift Register A 77
      TDFL Shift Register P 78
      TDFL Shift Register B 79
  3.3.4 The Control Circuit and The Clock Generator ..............80
      The Control Signals 80
      The Two Nonoverlapping Clock Generator/Driver 82
  3.3.5 Simulated Results for the 8-bit Divider ..................83
  3.3.6 Process Spread ........................................85

3.4 Summary ....................................................88

4.4 A 32-bit IEEE Floating Point Multiplier .....................89

4.1 The IEEE Floating Point Standard ..........................90
  4.1.1 Introduction ...........................................90
  4.1.2 IEEE Floating Point Format .............................90
  4.1.3 IEEE Rounding Modes ................................92
  4.1.4 Floating Point Multiplication ..........................92

4.2 Integer Multiplication Algorithms ..........................93
  4.2.1 Simple array multiplier ................................94
  4.2.2 The carry save multiplier ...............................95
  4.2.3 Radix-4 Booth's algorithm ..............................96
  4.2.4 The Wallace tree multipliers ...........................105
4.3 Rounding Algorithms .................................................. 107
  4.3.1 A simple round to nearest/up algorithm ................. 108
  4.3.2 A parallel rounding algorithm ............................. 109
  4.3.3 Obtaining round to nearest/even by round to nearest/up .. 111

4.4 A 32-bit IEEE Floating Point Multiplier ....................... 113
  4.4.1 A modified carry save array ............................... 114
  4.4.2 The final adder .............................................. 115
  4.4.3 Rounding .................................................... 117
      The CSA method 117
      The T1P method 120
  4.4.4 The exponent block ........................................ 126

4.5 Summary .................................................................. 129

5 The Modified Ring Notation Approach ............................ 130

  5.1 The Original Ring Notation Approach ......................... 131
    5.1.1 Motivation .................................................. 131
    5.1.2 Ring Notation (Ring Diagram) ........................... 131

  5.2 The New Modified Ring Notation Approach ................... 135
    5.2.1 The Modified Ring Notation Layout ..................... 135
    5.2.2 The Improvement in Layout Area ....................... 135
    5.2.3 The Improvement in Speed .............................. 135

  5.3 Global Power Supply and Ground Arrangement ............... 137
    5.3.1 Local Power and Ground Arrangement ................. 137
    5.3.2 Global Power and Ground Arrangement .............. 138

  5.4 Implementing the 32-bit Floating Point Multiplier ......... 140
    5.4.1 Mantissa Multiplier ..................................... 140
        Half adder 141
        Full adder 142
        Multiplexer 144
        Booth encoder 145
    5.4.2 Sticky Bit Generator .................................... 146
    5.4.3 Final Adder/Rounding ................................... 147
An 8-bit CSA method adder/rounder 147
An 8-bit T1P method adder/rounder 149

5.4.4 Exponent Block ...................................................... 152
5.4.5 The floating point multiplier chip ................................ 155

Final Floorplan 155
Simulations 155
Performance comparison 158

5.5 Summary ........................................................................ 158

6 A 16\times16-bit Fixed Point Multiplier Chip ......................... 160

6.1 The 16\times16-bit Fixed Point Multiplier ......................... 161

6.1.1 The Architecture of the 16\times16-bit Multiplier .............. 161
6.1.2 The Global Power and Ground Buses ......................... 162

\text{ir drop} 164
Inductance 165

6.2 The TDFL Test Circuit ................................................... 168

6.2.1 The 24-bit TDFL Shift Register ................................. 168
6.2.2 On Chip Clock Generator ........................................ 169
6.2.3 Probe Pads ............................................................. 169
6.2.4 The low\_to\_high (lotohi) Input ............................... 169
6.2.5 The HSPICE Simulated Results ................................ 170

6.3 The Test Chip .............................................................. 171

6.3.1 The CMOS Compatible Pad Driver and Receiver .......... 171
6.3.2 The Chip Layout .................................................... 174
6.3.3 Packaging The Chip ................................................ 175
6.3.4 PCB Board ........................................................... 177
6.3.5 Test of the chip ...................................................... 182

6.4 Summary ....................................................................... 183

7 Conclusions and Recommendations ................................... 184

7.1 Conclusions .................................................................. 185

7.2 Recommendations for Future Work ............................... 187
Appendix A Procedure to Implement the Face Equation in GWB...189
Appendix B Design Tools Used in This Thesis...............................191
Appendix C HSPICE Simulation Files for Crosstalk..........................192
Appendix D Testing Equipment..................................................195
Bibliography..................................................................................200
Abstract

The field of solid modelling has been of great interest for many years. The ability to design, analyse and represent graphically three dimensional objects is highly desirable for all CAD/CAM systems. Several sophisticated solid modelling systems now exist, but none is able to process objects of useful complexity in real time [8][9]. A large class of problems share a common three-dimensional numerical structure and require numerous calculations on 3D vectors. The aim of this Ph.D thesis is to design and implement the hardware mapping of critical paths of a GaAs Core Processor for a Solid Modelling Accelerator. This solid modelling accelerator is to be designed using GaAs/CMOS/BiCMOS unified technology. High speed GaAs technology is used in the core processor to deal with floating point intensive calculations, while CMOS technology is used where high speed outputs are not required such as for frequent accesses to heavily interlinked high density data structures.

In this project, a solid modelling program called GWB was studied first to identify those operations in solid modelling systems which are most amenable to hardware acceleration. This study showed a requirement for a core processor with high speed arithmetic process elements, namely, floating point adder/subtractor, multiplier, divider and square root function. The design of a GaAs Core Processor commenced with characterization of suitable logic families and development of a design approach to produce high speed, high density and low power dissipation GaAs VLSI IC's. These have been achieved by:

- Evaluating and comparing logic families such as Direct Coupled Logic (DCFL), Source Follower DCFL (SDFL), Super Buffer (SBFL) and Two-phase Dynamic Logic (TDFL).

- Using a novel mixed dynamic/static approach to implement an 8-bit serial divider as a test bench to optimize the speed, power and area.

- Investigating various fixed and floating point multiplier algorithms in terms of delay, power and area to select a suitable architecture for the GaAs Core Processor implementation. Developing TIP (Trailing-1’s Predictor) rounding technique to speed up floating point multiplication.
• Developing a new layout approach, called Modified Ring Notation (MRN) to implement a 32-bit floating point multiplier, improving layout density and speed. The analysis of this new layout approach includes interconnections, power supply and ground considerations. The comparison between the MRN and the original Ring Notation showed that the MRN maintained the advantages of the original while improving the layout density by a factor of up to 2 to 3 and also improved the speed because of shorter interconnection lines. The MRN floating point multiplier had better performance than the other floating point multipliers reported in the literature.

• Implementing, fabricating and testing a 16x16-bit multiplier chip to test the arithmetic architecture and the MRN layout methodology. Unfortunately because of a bug in the foundry’s software, the chip has had to be sent for refabrication. Therefore, the test results will not be included here. We plan to report the test results later when the chip is available.