VLSI TECHNOLOGY, INC.

VGT USERS GUIDE

ASIC Division

$15.00

VLSI TECHNOLOGY, INC.

VGT USERS GUIDE

VLSI TECHNOLOGY, INC.

VGT USERS GUIDE

ASIC Division

VLSI TECHNOLOGY, INC.

CONTENTS

CHAPTER 1

OVERVIEW

Introduction

This User's Guide describes the methodology for designing with VGT10 and VGT100 series gate arrays, using VLSI Technology's software tools and portable library.

For further information on using the software tools described in this guide, refer to the corresponding VLSI tool manual.

VGT Series Description

Process

The VGT10 and VGT100 series are advanced, high-performance CMOS gate array families in VLSI Technology's proprietary Continuous Gate^{TM} technology. These series, fabricated using a dual metal, N-well, implanted CMOS process, have high layout efficiency. Effective channel lengths are typically 1.5μ for VGT10 and 1.1μ for VGT100. The base arrays for the current VGT10 and VGT100 series are listed in Tables 1 and 2, respectively, together with pertinent information; other bases may be added in the future.

Table 1. Standard VGT10 Series Arrays

Marketing Part #	Nominal Array Size	Actual # Raw Gates [1]	Estimated Usable Gates [2]	# Sites	# I/O Pads [3]
VGT10-016	1600	1590	1200	3179	56
VGT10-024	2400	2425	1800	4850	68
VGT10-037	3700	3718	2800	7436	84
VGT10-064	6400	6358	4800	12716	108
VGT10-084	8400	8365	6300	16731	124
VGT10-106	10600	10648	8000	21296	140

Table 2. Standard VGT100 Series Arrays

Marketing Part #	Nominal Array Size	Actual # Raw Gates [1]	Estimated Usable Gates [2]	# Sites	# I/O Pads [3]
VGT100122	12000	12149	9000	24299	152
VGT100160	16000	16038	12000	32076	172
VGT100205	20000	20465	15000	40931	196
VGT100270	27000	26950	20000	53900	224
VGT100397	40000	39700	30000	79475	272
VGT100528	53000	52800	40000	105644	312
VGT100665	66500	66550	50000	133100	348

Notes:

1. A *gate* is defined as one 2-input NAND equivalent. *Raw* refers to total gates available before routing.

2. The *Estimated Usable Gates* value represents approximately 75% of the raw gate count. Maximum usage varies depending upon design.

3. Every array has eight additional dedicated VSS pads available to the user: four for internal VSS and four for output buffer VSS.

Array Core

The VGT10 and VGT100 cores consist of rows of uncommitted P and N transistors laid out at regular intervals. Key features are high gate density arising from a gate isolation technique and a unique global routing scheme.

Gate isolation turns off P and N transistors to isolate diffusion nodes from one another. This technique allows array area normally consumed by oxide isolation to be used as raw transistors.

The VGT gate array architecture eliminates open areas for routing channels; instead, the routing channels run over active cells.

Features

The VGT10 and VGT100 Series Gate Arrays provide these features:

- An advanced high-density architecture with typical 75% gate utilization.
- A flexible I/O structure, including:
  - TTL, CMOS, and Schmitt trigger inputs
  - pullup, pulldown, and hold circuits

- output drive programmable from 2 to 12 mA

- VLSI's extensive Portable Library, which allows future migration of your design to other gate array and cell-based technologies.

- Full support by VLSI Technology's comprehensive integrated IC design tools.

- An advanced dual metal, N-well, implanted CMOS process, with 2.0μ drawn channel lengths for VGT10 and 1.5μ for VGT100.

- High immunity to noise, latchup and electrostatic discharge (ESD).

Package Availability

The packages available for the VGT10 and VGT100 series gate array bases, as of the date of this publication, are listed in Appendix A.

I/O Buffers

I/O Locations

Any I/O location can be programmed as input, output, bidirectional, VSS or VDD.

Output

The output buffers have been designed to sink and source up to 12mA across the industrial temperature range. In addition, the VGT10 and VGT100 I/O buffers have minimal susceptibility to latchup and electrostatic discharge (ESD). Appendix E discusses output buffer AC characterization for the VGT family.

Input

By selecting appropriate library I/O cells, you can program each input location as TTL, CMOS, or Schmitt Trigger. In addition, pullup and pulldown resistors and hold circuits are available in combination with inputs, to prevent floating signals.

Power

A dual power bus structure isolates the output buffer power supply from that of the array core, to achieve high noise immunity. A complete set of rules for power pads is given in Chapter 3, SPECIFYING YOUR DESIGN.

There are no dedicated pads in the padring other than the eight VSS pads in the corners: four for internal VSS and four for output buffer VSS.

Guide Description

This User's Guide contains six chapters:

- This Chapter, OVERVIEW, gives a brief overview of the VGT gate array series and its primary features.
- Chapter 2, DESIGN CYCLE, lists what you, and the VLSI Technology Design Center, need to do to complete your chip, and the VLSI software design tools needed to perform these tasks.
- Chapter 3, SPECIFYING YOUR DESIGN, describes how to specify the physical characteristics of your chip, such as array size, power pads, packaging, and bonding diagrams.
- Chapter 4, IMPLEMENTING YOUR DESIGN, discusses the considerations that are unique to VGT gate array design.
- Chapter 5, PRE-ROUTE DESIGN REVIEW, describes the design tasks that must be completed before VLSI can place and route your chip.
- Chapter 6, POST-ROUTE SIMULATION/SIGNOFF, describes the design tasks that must be completed before VLSI can build prototypes of your chip.

Terms And Definitions

This section defines the specialized terms used in this guide.

Base Array - A fixed-size gate array, consisting of the non-custom mask layers.

Base Count - The number of transistor pairs necessary to implement a particular macro function. The base count includes one end isolation transistor pair.

Cell Count - The total number of transistors in an array base divided by two.

Gate Utilization - The number of active gates used in a design as a percentage of total available gates. Gate utilization with the VGT series is typically 75%.

Option - A design implemented on a gate array base, including the custom mask layers.

Portable Library - VLSI's cell library, which is supported in several design and process technologies, including VGT10 and VGT100 arrays, and VSC10 and VSC100 standard cells.

Raw Gates - The total number of available gates in the core of a particular base array. A gate is defined as four transistors.

Sites - Refers to the core transistor pairs in which macro functions may be implemented. The number of sites is approximately equal to one half the number of raw gates. Also referred to as Cell Sites.

Site Utilization - The number of transistor pairs that are used in a design, including isolation transistors, as a percentage of total available sites. Transistor pairs are called "placement sites" or "sites." The typical maximum upper limit for site utilization with the VGT series is 90%. VTIscreen

reports this value as SITE UTILIZATION PERCENTAGE. The number of sites in each base array is given in Tables 1 and 2.

Usable Gates - The number of gates that is estimated to be routable. In the VGT Series, the maximum number of usable gates is approximately 75% of the raw gates, but is dependent on the particular design and may be even higher. The difference between raw and usable gates is due to isolation transistors and routing inefficiency.

VDDI - VDD pad for the internal core, input logic, and pad drivers.

VDDO - VDD pad for the external output and I/O buffers.

VGT - Refers to two series of gate array bases first introduced by VLSI Technology, Inc. These bases use Continuous Gate^{TM} technology to achieve dense arrays and high cell utilization.

VGT10 - The 2μ (drawn) VGT gate array series.

VGT100 - The 1.5μ (drawn) VGT gate array series.

VSSI - VSS pad for the internal core, input logic, and pad drivers.

VSSO - VSS pad for the external output and I/O buffers.

CHAPTER 2

DESIGN CYCLE

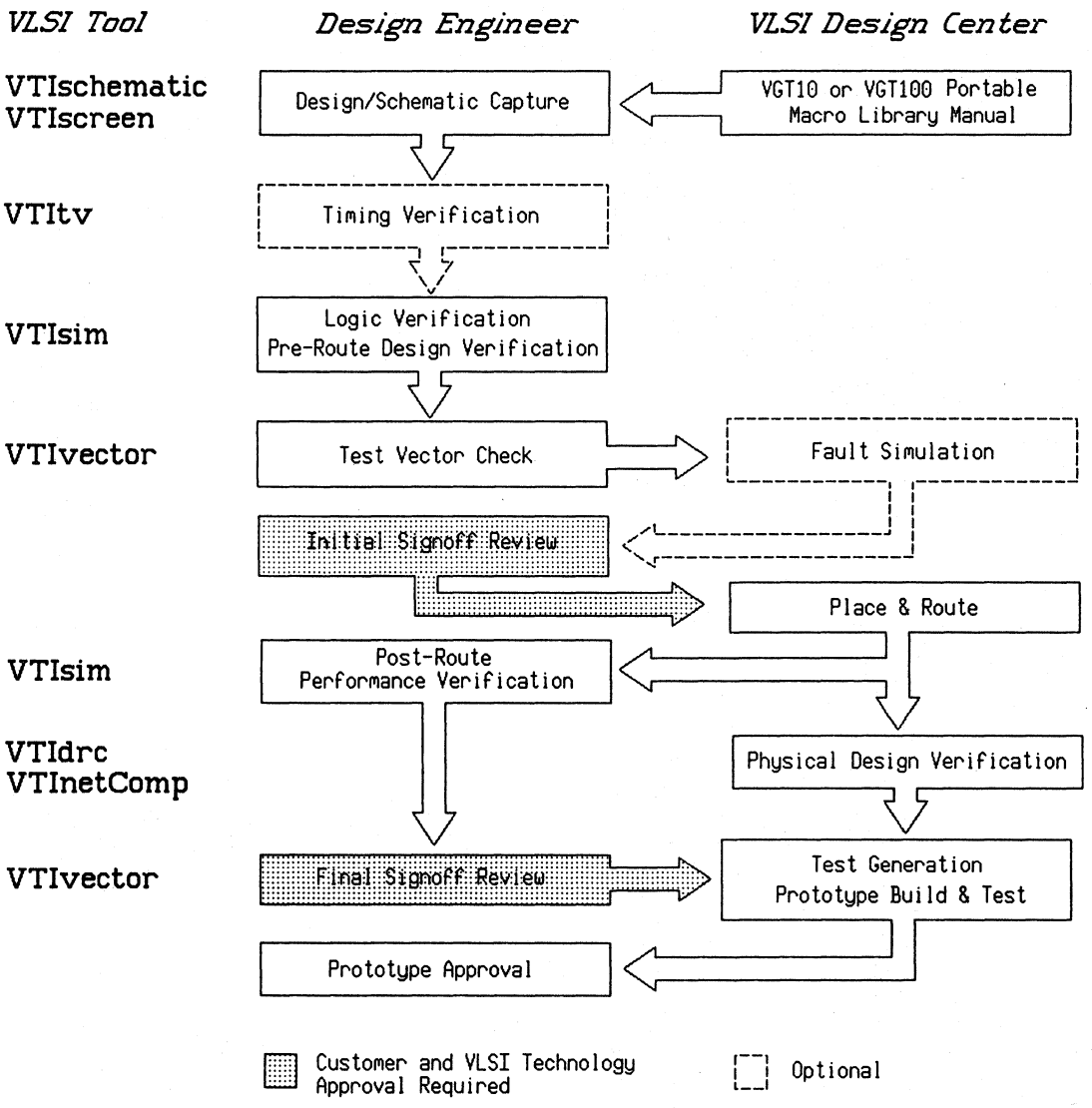
Introduction

This chapter gives a top-level look at how to plan, implement, and check your gate array design. It also describes the VLSI software tools that are used during the VGT design process. It assumes that you already have a working knowledge of the applicable tools. Detailed information concerning the use of a particular tool can be found in the appropriate user's manual; for example, details on using the VTIsim logic/timing simulator can be found in the VTIsim Users Manual.

Planning Your Design

Figure 1 shows the standard design flow using the VGT Portable Library. The steps are described in more detail in the text following the figure.

Figure 1. Standard VGT Design Flow



Preliminary Specifications

Before you begin your design, you need to:

- Determine the gate array base to be used in the design.
- Define the I/O ring and pinouts.
- Select the device package.
- Describe performance specifications.

You can then fill out the **Initial Design Specification Form** and begin a draft of the **AC/DC Specification Form.**

If necessary, an Initial Specification Acceptance review can be held at a VLSI Design Center to determine design feasibility, review the specifications, and answer and resolve any technical questions.

Design/ Schematic Capture

Schematic Capture

After the base and package are selected, enter your schematic using VTIschematic on your workstation. Either the VGT10 or VGT100 macro library is used, depending upon which gate array family you choose to implement your design. Wire up the appropriate macros, in a hierarchical fashion, to create the desired circuit.

For detailed instructions on schematic capture, refer to your VTIschematic manual.

Netlist Screening

Next, run VTIscreen to identify simple errors and other potential problems in the logic design. It also provides a utilization summary of the particular array you have chosen. Refer to the VTIscreen manual for more details.

When you run VTIscreen to read the `[hns]` netlist created by VTIschematic, it:

- Calculates the internal site and I/O pad utilization and produces a summary report.
- Provides a Design Statistics report listing information on the number of nets and FromTos - connections between two pins - in your design.
- Provides a Design Complexity report listing gate equivalents per macro.
- Checks for certain design and connectivity errors and other potential problems in your design.
- Prints a short form of the report to your terminal as well as producing a disk file, `[scr]`, containing the entire report.
- Includes in the `[scr]` disk file a summary listing of the number of each type of macro used as well as the instance names of each placement.
- Optionally writes an `[lds]` file to be used by VTIvector. This file is required for test program generation; be sure to specify this file by using the SAVE command in VTIscreen.

A summary of the reports, warnings, and errors provided by VTIscreen can be found in Appendix B of this document, SCREEN REPORTS AND ERRORS.

Timing Verification

After the schematic is entered and checked, it is useful to check critical paths for timing verification. Timing verification with VTItv allows you to detemine if your design functions properly at your clock speed and that all critical speed paths meet their specified timing. This step is optional.

Logic Verification And Pre-Route Design Verification

Logic Verification

Prepare an evaluation logic driver file and run a detailed logic simulation of your circuit with VTIsim to verify the functionality of your design. VTIsim creates waveform plots and a trace file, `[trc]`.

Pre-Route Design Verification

The customer design engineer prescribes a set of timing and test program engineering simulations, where the pre-route performance of the chip is predicted by simulating with predicted wire capacitances. These simulations are run and documented by the customer engineer, or at the VLSI Design Center.

Test Vector Check

Once the initial logic is verified, use VTIvector and its CHECK command to assure that the simulation vectors you have chosen adhere to the test vector guidelines described in the application note called **Test Generation Guidelines.** When there are no errors in the resulting `[ver]` error report file, VLSI can use your simulation vectors to produce Sentry test vectors.

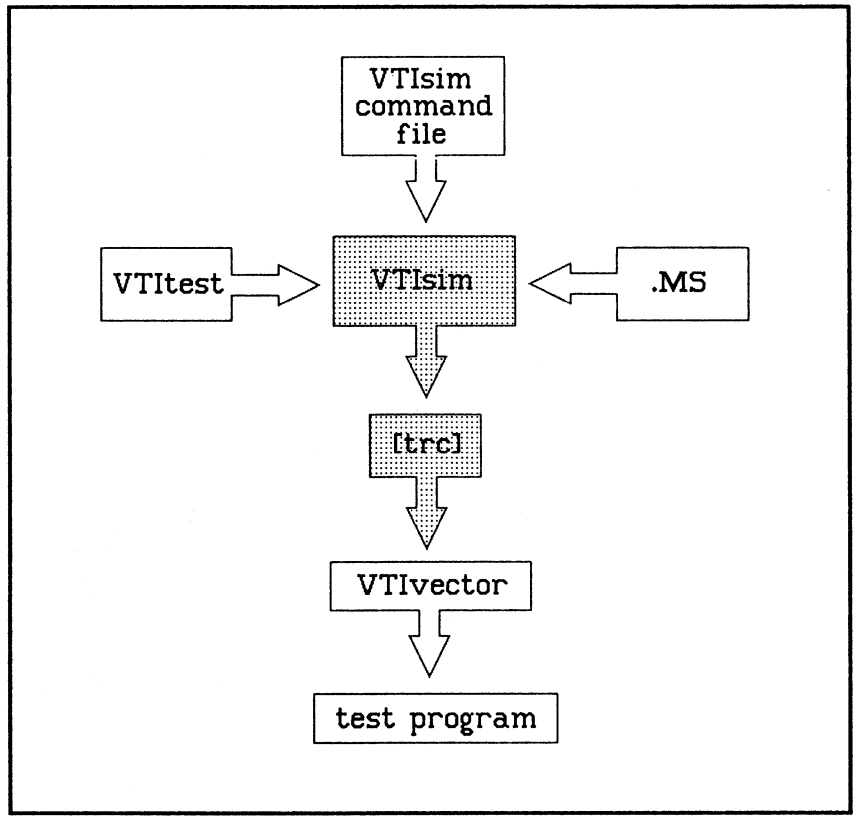
To create the simulation vectors, prepare a standard 1MHz test program driver file and run a simulation to generate a `[trc]` trace output file. This file is used to create the test program for the specified tester.

There are three ways to create the required `[trc]` file, as shown in Figure 2:

- Use VTItest to drive VTIsim as described in the VTItest manual.
- Use VTIsim commands entered manually or from a command file.
- Write a Mainsail .MS file to drive VTIsim as described in the VTIsim manual.

In all cases, the resulting output `[trc]` file is used by VTIvector, run by a VLSI Design Center, to create the test program.

Figure 2. Trace File Generation



VLSI uses your simulation trace file, `[trc]`, and the `[lds]` file produced by VTIscreen, to produce the correct test vectors. These vectors are mapped from the simulation vectors that were used in your VTIsim simulation.

You also create an Input Timing Generator `[itg]` file and, if you want a critical path check, an optional Critical Path Check `[cpc]` file. See the application note entitled **Test Generation Guidelines** for more information on creating these files.

Fault Simulation

Fault simulation provides an automated algorithmic measure of a test program's ability to exercise nodes in your circuit. VLSI Technology Design Centers, as a service, offers assistance in performing fault simulation.

Using your design netlist and simulation vectors, VLSI Technology runs the GenRad HILO3 fault simulator, and provides you with a detailed report of the results. The report includes fault coverage, percentage of faults dropped, and potential coverage. Additionally, VLSI provides a file of undetected, undetectable and potentially detected faults, a cross-reference file, a netlist screen file, and documents to help evaluate and interpret the results.

Assistance in analyzing the results is also provided to help you determine if the fault coverage is acceptable. If not, then additional vectors are needed. VLSI can also assist you to enhance your fault coverage.

Fault simulation is an optional step.

Initial Signoff Review

When simulation is complete, you can fill out the initial VLSI forms detailing the pinouts and pad selections. A completed **User Design Information Form, AC/DC Specification Form,** and **Customer Package Marking Form** are required. In addition, the **Pad Placement File**, on floppy disk or magnetic tape, is required. A joint customer-and-VLSI pre-route design review is then held at which the **Initial Signoff Review Checklist** is filled out and signed off.

Place and Route

The VLSI Technology Design Center automatically places and routes your design in accordance with your Pad Placement File and the optional Critical Placement File. The Design Center checks the resulting CIF data base file with VTIdrc and verifies it against the original schematics with VTInetComp. A post-route `[pst]` file containing the actual routing interconnect wire capacitance is returned to you for use in post-route timing analysis.

Post Route Performance Verification

After routing, the VLSI Design Center provides you with a `[pst]` file containing the metal interconnect routing capacitances. Use this file to rerun the simulation, with VTIsim in the timing mode, to get accurate timing information about your circuit. You can then document the prescribed set of design timing and test program performance simulations, to be included in the final signoff review package. Based upon this information, you decide whether or not to proceed with prototyping the array.

Physical Design Verification

After the circuit is routed, VLSI does a final verification on your circuit by:

- Comparing the `[hns]` netlist from VTIschematic to the extracted netlist from the physical routed array and
- Performing a design rule check of the physical array.

These are the final checks performed before prototype build can begin.

Final Signoff Review

A joint customer-and-VLSI post-route design review is held, during which the **Final Timing Signoff Review Checklist** is filled out and signed off.

Test Generation and Prototype Build and Test

Test Generation

During the Test Generation phase, the VLSI Technology Design Center uses VTIvector to convert your simulation trace file, `[trc]`, to a test program for wafer sort and final device test of your prototypes.

Prototype Build and Test

The VLSI Technology Design Center submits the fully checked and verified mask data base to the foundry for design code mask generation. VLSI fabricates the wafers, assembles the prototype parts, tests them, and delivers them to the customer.

Prototype Approval

When you approve the tested prototypes by signing the **Prototype Approval Report**, the device is transferred to manufacturing and is ready for volume production.

Design Assistance

VLSI Technology Design Centers, as a service, offer assistance in performing any step in the design flow. Design Center services include test vector generation, functional/timing simulation, fault simulation and special testing services.

CHAPTER 3

SPECIFYING YOUR DESIGN

Introduction

This chapter introduces the recommended methodology for beginning a VLSI Technology gate array design.

Design Specifications

Preparing your design specifications requires you to make several decisions and calculations about your circuit. These specifications are listed on the **User Design Information Form**: the array size, the package type, and other physical design specifications.

Prepare the initial specifications based on planning done at your site, and submit them to VLSI Technology. The initial specifications include:

- User Design Information Form
- AC/DC Specification Form
- Customer Package Marking Form
- Special requirements (optional)

These are your first performance goal statements for the circuit. The initial specifications identify key requirements and estimates such as gate count, I/O buffer count, AC characteristics, power, and packaging.

Submit the initial specifications before implementing the actual design so that VLSI Technology can advise you of the feasibility of your design plan. Because this phase occurs early in the design process, and because estimates are used, initial specifications should be considered only as guidelines.

Initial Specifications

User Design Information Form

Use this form to provide the VLSI Design Center with information and general specifications for your design, such as:

- The array size
- Prototype and production packages for your design
- Maximum operating frequency
- Engineering contact
- Estimated design schedule

AC/DC Specification Form

Use this form to specify the operating conditions for your chip:

- Ambient operating temperature
- Supply voltage

- Power dissipation
- Output load capacitance
- Input capacitance
- Output three-state leakage current
- Input leakage current
- VIL, VIH, VOL, VOH, IOL, IOH
- Critical path information

Optional Files

Performance requirements can affect many of your design decisions. VLSI Technology's VGT series of gate arrays is specified for high performance design and offers some very high drive output buffers. Even so, your design may have one or more critical paths requiring special design consideration.

VLSI Technology addresses this problem in two ways. For placement and routing, VLSI allows you to specify critical placements in a **critical placement file**, CPF (Chapter 5). To test a critical path, you can submit a **critical path check file**, CPC. Refer to the application note called **Test Generation Guidelines** for information on creating the CPC file.

Both of these files are optional, and need only be submitted in the case of some critical timing that even the VGT series' overall high performance specifications cannot guarantee without special attention.

While you do not submit these files until the end of the design process, the critical path should be identified and analysed at the **beginning** of the design specification.

Completing The User Design Information Form

This section describes the information required to complete the **User Design Information Form.**

Selecting Package Type

You need to specify the package type for both the prototype and production versions of your chip on the **User Design Information Form.** You can choose from a variety of plastic and ceramic packages for your device, including:

- Plastic dual in-line packages, PDIP
- Side-brazed ceramic packages, S/B
- Leaded ceramic chip carriers, LDCCC
- Leadless ceramic chip carriers, LLCCC
- Plastic leaded chip carriers, PLCC
- Ceramic pin-grid arrays, CPGA
- Plastic pin-grid arrays, PPGA

Decide the package type and size you want, using the charts in Appendix A. Contact your VLSI Technology Design Center for information on availability of packages not listed in these charts.

Determining the Number of Package Pins

One package pin is required for each input, output, bidirectional, power and ground signal. The number of package pins is the number of pins needed for the design plus power and ground pins. The section entitled **How To Determine the Number and Placement of Power Pads,**

later in this chapter, tells how to determine the number of power pins. Write the number of input, output, bidirectional, power, and ground pins on the **User Design Information Form** as part of your initial specifications.

This information is also required for preparing a bonding diagram, which is covered in the section entitled **Preparing a Bonding Diagram**, later in this chapter.

Estimating Gate Count

Estimate the number of gates your design will contain. If this number exceeds 75% of the raw gate count of the base array you selected, please contact your VLSI Design Center for approval.

Selecting Array Size

Each array size has a specific number of gates, I/O pads, and fixed VSS pads, as shown in Tables 1 and 2 in Chapter 1. To choose the device which best suits your needs, you must determine the following factors about your circuit:

- Percentage of array used (gate utilization)
- Number of I/O signals
- Number of power pads (VSSI, VSSO, VDDI, VDDO)

Determining these factors will aid you in properly filling out the **User Design Information Form.** More importantly, accurately determining these factors helps you choose the package type most appropriate for your gate array design.

Power Calculations

Background Information

When you complete the **Power Calculations Worksheet** according to the instructions at the end of this section, you will be using typical power dissipation figures supplied by VLSI for each array series. These figures are based on the following considerations:

- Causes of Power Dissipation: The amount of heat generated within the silicon chip, known as power dissipation, is low in CMOS technology, compared to other technologies. Power dissipation causes temperature to rise, increasing a circuit's propagation delay. The three causes of power dissipation in CMOS technology are:

  1. The charging and discharging of the internal capacitance of a circuit. Known as *AC power dissipation*, the charging and discharging -- switching -- of circuit capacitance is responsible for more than 90% of a circuit's total power dissipation. The power dissipation in a CMOS circuit is essentially a function of the frequency of the logic switching. The charging of a capacitor (C) to a voltage (V) through a P-channel device builds up a charge (CV) and stores energy (CV*V). This energy is later discharged through the N-channel device which is paired with the P-channel device. When such switching takes place at a frequency (F), the resulting power dissipation can be expressed as $P=FCV^2$, where P is power dissipation.

  2. DC current. There are two types of DC power consumption: static DC current, which flows through ON transistors; and leakage DC current, which

continues to flow when transistors are OFF. Because DC leakage in a properly functioning CMOS circuit is very low, you need to set up a test condition in which all static DC current can be turned off, so DC leakage (static Idd) can be measured easily.

The total power dissipation of a gate array option can be accurately estimated by adding the estimated power dissipation for the macros you have used in your design, and multiplying by a correction factor to account for the percentage of gates which switch simultaneously. Statistically, 0.20 (20%) has been found to be a useful correction factor, reflecting the percentage of simultaneously switching gates commonly found in VGT designs. Your own estimate may be somewhat higher or lower, depending on the characteristics of your design.

The power dissipation for each VGT macro is given in the library manual. Unused I/O macro locations dissipate no power. Maximum allowable power dissipation, of course, depends on the package and cooling system used.

3. Transient currents. Transient currents occur when the P- and N- transistors switch from the High to Low state, or vice-versa, in the period when:

$$VTH(N) > VIN < VDD-VTH(P)$$

where VTH = input threshold loads, N = N-transistor, VIN = input, VDD = supply voltage, and P = P-transistor. Transient currents are responsible for less than 10% of the total power dissipation.

Completing the Power Calculations Worksheet

The **Power Calculations Worksheet**, illustrated in Figure 3, helps you calculate your circuit's junction temperature, which is the temperature of the die inside the package. It is also used to arrive at certain values required on the **AC/DC Specifications Form**. VLSI encourages you to include a copy of the worksheet as part of the initial signoff review package, although it is not required.

Completing the **Power Calculations Worksheet** requires the following steps, described in detail in the text that follows:

- Complete the circuit data section.
- Calculate the Register Percentage (R).
- Calculate the Internal Power Dissipation (Pint).
- Calculate the External Power Dissipation (Pext).
- Calculate the Total Power Dissipation (Ptot).
- Calculate the Junction Temperature (Tj).

Figure 3. Power Calculations Worksheet

POWER CALCULATIONS WORKSHEET

DESIGN NAME _____

VLSI TECHNOLOGY PART NUMBER _____

ARRAY SERIES MILLIWATTS/GATE	P =	0.020 mW/MHZ/GATE	
AVG. OPERATING FREQUENCY (F)	F =	_____	MHZ
EST. FRACTION OF GATES SWITCHING SIMULTANEOUSLY (TYPICALLY 0.20)	S =	_____	
AMBIENT OPERATING TEMPERATURE	TA =	_____	C
NUMBER OF GATES	G =	_____	
NUMBER OF OUTPUT PINS	B =	_____	
AVG. OUTPUT LOAD CAPACITANCE	C =	_____	PF
INTERNAL POWER DISSIPATION PINT = P * F * S * G	PINT =	_____	mW
EXTERNAL POWER DISSIPATION PEXT = 0.035 * F * B * .2 * C	PEXT =	_____	mW
TOTAL POWER DISSIPATION PTOT = 0.001 * (PINT + PEXT)	PTOT =	_____	W
PACKAGE TYPE (INCL. # PINS)		_____	
THETA JA OF PACKAGE		_____	deg. C
JUNCTION TEMPERATURE TJ = (PTOT * THETA JA) + TA	TJ =	_____	C

REPORT GENERATED BY: _____

Complete the Circuit Data Section

Complete each part of the circuit data section as follows:

- ARRAY SERIES MILLIWATTS/GATE (P). Typical power dissipation in milliwatts/MHz/gate. The typical power dissipation for VLSI's VGT arrays is provided in the datasheets in the **VGT10** and **VGT100 Macro Library Manuals.**

- AVERAGE OPERATING FREQUENCY (F). Write the circuit's operating frequency in megahertz in the blank.

- AMBIENT OPERATING TEMPERATURE (Ta). Write the circuit's maximum ambient operating temperature in degrees centigrade in the blank.

- NUMBER OF GATES (G). Write the number of gates required by the circuit in the blank.

- NUMBER OF OUTPUT PINS (B). Write the number of outputs the circuit contains in the blank.

- AVERAGE OUTPUT LOAD CAPACITANCE (C). Estimate the output load capacitance using the input capacitance specifications for the interfacing chips and the interconnect capacitance. Write your estimate of the average, in picofarads, in the blank.

Calculate the Internal Power Dissipation (Pint)

Using the values you noted in the circuit data section, and the Register Percentage you calculated, solve the equation shown and write the result in the blank.

Calculate the External Power Dissipation (Pext)

Using the values you noted in the circuit data section, solve the equation shown, and write the result in the blank.

Calculate the Total Power Dissipation (Ptot)

Add the Internal Power Dissipation to the External Power Dissipation and multiply the sum by .001. The result is the circuit's Total Power Dissipation in watts. Write the results in the blank.

Write In The Package Type

Write the type of package you have chosen, including the number of pins, in the blank provided.

Write The Thermal Impedance

Write the thermal impedance (Theta JA) of the package you have chosen in the blank provided. This value is given in the Semiconductor Package Selection Guide which you obtain from the Design Center.

The junction-to-ambient (JA) thermal resistance data is based on a 10,000 square mil die, with the board mounted in still air. The thermal resistance varies with the materials used, die size, process technology, air circulation, and heat dissipation characteristics of the device. Values listed in the Package Selection Guide are meant to serve as guidelines and are believed to be on the high side. For larger die, the values are typically lower.

Calculate the Junction Temperature (Tj)

Multiply the Total Power Dissipation by the thermal impedance (Θ_{ja}) of the package you have chosen. When the thermal impedance is given as a range, use the highest value in the range. Add the result to the Ambient Operating Temperature (T_a) you noted in the circuit data section. Write the sum in the blank. This is the estimated junction temperature in your circuit.

Specifying Power Pads

The rules, tables, and procedures in this section must be closely followed to ensure reliability of your VGT design. Various design examples are presented at the end to help you determine the right number of power pins for your design.

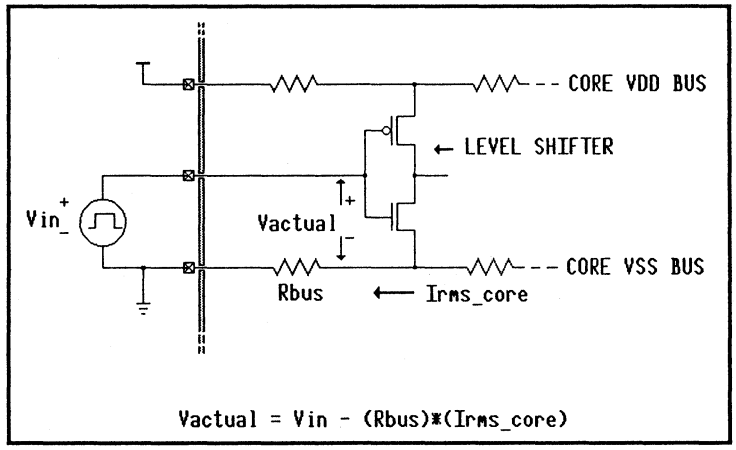
Background Information

The VGT family I/O buffers are designed for high output drive capability. The switching current of an output buffer with this drive level can be as high as 95 mA. When a number of output buffers switch simultaneously, within a window of 100ns (worst case), the peak switching current at a VSS or VDD pin is high enough to cause a voltage spike on the internal VSS and VDD buses. This spike might result in erroneous switching of the input level shifters. TTL input level shifters are exceptionally sensitive to these spikes since they are designed to operate within narrow VIL and VIH levels.

Power pin spikes are caused mainly by two factors. One source is the inductive voltage drop/surge caused by the lead inductance of the package pin bonded to the power pad. The other factor affecting the switching performance of input level

shifters is the ohmic voltage drop across the VSS power bus between the virtual VSS of the level shifter and the VSS pin of the device. This voltage drop causes the input of the level shifter to see an actual voltage of VIN reduced by the voltage drop across the VSS bus (Figure 4).

Figure 4. Level Shifter Model



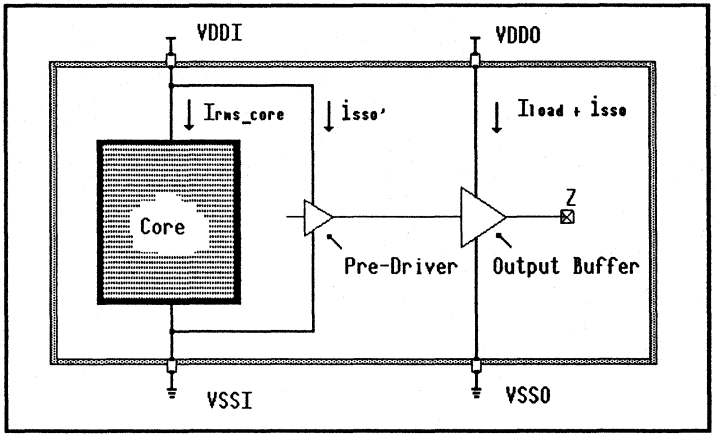
Additionally, enough power must be applied to handle the DC requirements of the output buffers and the average operating current of the array core.

To Split or Not to Split

The VGT family power structure is designed for maximum flexibility of power pad arrangement. Power macros can be selected to:

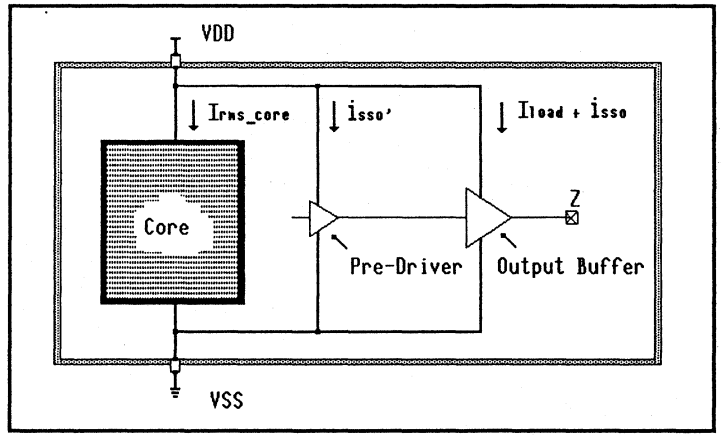
- Have a separate VSS and VDD bus structure for the output buffers and the input buffers/array core (Figure 5). This arrangement requires a minimum of two VSS pins and two VDD pins, and is by far the best arrangement for avoiding input level sensitivity problems. Although using two separate power supplies is the ultimate in performance optimization, running two separate power traces from a single external power supply is the next best thing. This requires bonding out internal and external power pairs separately. However, the noise immunity performance obtained from such a split is much better than using a non-split power configuration. For the purpose of the next sections, treat the case of internally split power pairs powered from two separate traces of a single supply as a case of two separate supplies.

Figure 5. Split Power Supplies



- Have no separate VSS and VDD power buses (Figure 6). The internal and external VSS and VDD buses are shorted at the power pad level. This is a less favorable arrangement than the separate power buses described above, but it is maintained since some applications cannot manage to have two separate power traces on the board housing the part.

Figure 6. Single Power Supply/Joint Power Pads



Regardless of the choice of a joint or separate power bus arrangement, noise problems are minimized by choosing the proper package pins for power. This is due to the fact that different pins of the same package have a wide range of lead inductance, ranging from less than 5 nH to over 20 nH. For the same current level, a 20 nH pin produces a spike four times greater than the voltage spike caused by a 5 nH pin. Thus, choosing the appropriate power pin assignment for a selected package can by itself reduce the number of power pins required, and the associated noise, by a large factor.

Output buffers' DC currents, IOL and IOH, establish high current demand on the supplies. Thus, the output buffers are designed with four levels of drive capability: 2 mA, 4 mA, 8 mA, and 12 mA. The buffer with the lowest drive capability should be used, unless higher speed or drive capability is required.

As seen in Figures 5 and 6, there are four primary current paths in your design:

- `Irms_core` is the RMS current required by the internal core of the array. This current is primarily a function of operating frequency and array utilization.

- `isso` is the instantaneous current associated with the simultaneously switching output buffers.

- `isso'` is the instantaneous current caused by the simultaneously switching outputs. This current is associated with the internal core since the power for the pre-drivers is tapped from the internal power bus.

- `Iload` is the DC current that the output buffers sink when they output a low state.

With split supplies, shown in Figure 5, `isso` is steered away from the input level shifters, which greatly improves noise margins. Designs using a single supply must effectively handle all current components through one VSS bus. This deteriorates the input noise margins, and usually requires more power pairs.

How to Determine the Number and Placement of Power Pads

Use the steps given in this section to determine the number and placement of power pairs in your design. Determining power pairs and placement can be an iterative process if your design operates at a very high frequency and has many simultaneously switching outputs.

- **Step One: Determine the power pin locations.**

If the pin assignment is predetermined, then the pin numbers used for power are known and their inductances can be looked up in the package inductance table, Table 3. If the bonding is not preassigned, consult the package inductance table for the package pins with the least inductance. If only partial bonding flexibility is available, try to assign the pin with the lowest inductance to VSS.

Table 3. Package Pin Inductance

S/B PKG.	Pin Inductance - S/B Packages			
	Low (0 - 5 nH)	Medium Low (5 - 10 nH)	Medium (10 - 15 nH)	High (15 - 20 nH)
22-pin S/B	2, 3, 7-10, 13, 14, 18-21	1, 4, 5, 11, 12, 15, 16, 22	6, 17	
24-pin S/B	2-11, 14-23		1, 12, 13, 24	
28-pin S/B 250x250	4-11, 18-25	2, 3, 12, 13, 16, 17, 26, 27	1, 14, 15, 28	
28-pin S/B 310x310	6-9, 20-23	1-5, 10-19, 24-28		
28-pin S/B 350x350	4-11, 18-25	1-3, 12-17, 26-28		
40-pin S/B	8-13, 28-33	2-7, 14-19, 22-27, 34-39	1, 20, 21, 40	
48-pin S/B	10-15, 34-39	4-9, 16-21, 28-33, 40-45	1-3, 22-27, 46-48	
64-pin S/B		10-23, 42-55	2-9, 24-31, 34-41, 56-63	1, 32, 33, 64 (23.4 nH)

Table 3. Package Pin Inductance (Cont'd)

CPGA PKG.	Pin Inductance - CPGA Packages			
	Low (0 - 5 nH)	Medium Low (5 - 10 nH)	Medium (10 - 15 nH)	High (15 - 20 nH)
68-pin CPGA	9, 26, 43, 60	All pins except 9, 26, 43 and 60		
84-pin CPGA	1, 22, 43, 64	All pins except 1, 22, 43 and 64		
120-pin CPGA		1, 2, 4, 5, 7-26, 28, 29, 31, 32, 34, 35, 37-56, 58, 59, 61, 62, 64, 65, 67-86, 88, 89, 91, 92, 94, 95, 97-116, 118, 119	3, 6, 27, 30, 33, 36, 57, 60, 63, 66, 87, 90, 93, 96, 117, 120	
180-pin CPGA	23, 68, 113, 158	1, 10, 11, 16-18, 22, 24, 28, 31, 46, 63, 64, 66, 70, 71, 91, 100, 101, 106-108, 112, 114, 118, 121, 136, 153, 154, 156, 160, 161	2-9, 12-15, 19-21, 23, 25-27, 29, 30, 32-45, 47-62, 65, 67, 69, 72-82, 84-90, 92-99, 102-105, 109-111, 115-117, 119, 120, 122-135, 137-152, 155, 157, 159, 162-172, 174-180	83, 173

Table 3. Package Pin Inductance (Cont'd)

PDIP PKG.	Pin Inductance - PDIP Packages			
	Low (0 - 5 nH)	Medium Low (5 - 10 nH)	Medium (10 - 15 nH)	High (15 - 20 nH)
20-pin PDIP	2-9, 12-19	1, 10, 11, 20		
40-pin PDIP 180x180	8-13, 28-33	2-7, 14-19, 22-27, 34-39	1, 20, 21, 40	
40-pin PDIP 200x200	6-15, 26-35	2-5, 16-19, 22-25, 36-39	1, 20, 21, 40	
40-pin PDIP 260x266	2-19, 22-39	1, 20, 21, 40		

PLCC PKG.	Pin Inductance - PLCC Packages			
	Low (0 - 5 nH)	Medium Low (5 - 10 nH)	Medium (10 - 15 nH)	High (15 - 20 nH)
28-pin PLCC	All pins			
44-pin PLCC	1-4, 9-15, 20-26, 31-37, 42-44	5-8, 16-19, 27-30, 38-41		
68-pin PLCC		All pins		
84-pin PLCC		All pins		

Table 3. Package Pin Inductance (Cont'd)

PKG.	Pin Inductance - LLCCC and LDCCC Packages			
	Low (0 - 5 nH)	Medium Low (5 - 10 nH)	Medium (10 - 15 nH)	High (15 - 20 nH)
44-pin LDCCC	1-5, 8-16 19-27, 30-38, 41-44	6, 7, 17, 18, 28, 29, 39, 40		
68-pin LLCCC		All pins		
84-pin LLCCC	1, 21, 22, 43, 44, 64, 65, 84	2-10, 13-20, 23-31, 34-42, 45-52, 55-63, 66-73, 76-83	11, 12, 32, 33, 53, 54, 74, 75	

In general, arrange the power pairs for maximum current distribution. However, in arrays larger than 16,000 with high utilization and operating at frequencies greater than 10 MHz, internal power requirements can be in the hundreds of milliamperes.

The power pads must be arranged so that those located on the sides adjacent to the internal buses, the two opposite sides from which internal VSS/VDD buses are tapped, can fully handle all the internal power needs of the array plus that of the output buffers located on those sides as well. The power pads should alternate (VSS, VDD) and should be evenly distributed. Refer to the section on **Preferred Power Pad Locations**, in this chapter.

- **Step Two: Determine the power pairs required for core, `Irms_core`.**

Table 4 gives the total current, `Irms_core`, for various array internals at 100% utilization as a function of f-clock, the highest clocking frequency of the circuit. `Irms_core` is in mA and f (f-clock) is in MHz. For lower utilization:

```
Irms_core = Irms_core(table) * %utilization
```

This table also gives the maximum `Irms_core` at which an array can be operated before the maximum current handling capability of internal buses is reached.

Table 4. Current Requirements

DEVICE SIZE	TOTAL Irms_core (IN MA) AS A FUNCTION OF F(IN MHZ) AND AT 100% UTILIZATION	MAXIMUM Irms_core (IN MA) FOR INTERNAL BUS SAFETY
1590	1.0 mA * f-clock (MHz)	--
2425	1.4 * f	--
3718	2.2 * f	--
6358	3.4 * f	--
8365	4.45 * f	--
10648	6.45 * f	--
12149	7.75 * f	435
16038	8.8 * f	435
20465	10.0 * f	500
26930	13.1 * f	550
39700	18.8 * f	600
52800	23.9 * f	700
66550	29.4 * f	700

At the upper limit of `Irms_core`, the maximum current handling capability of an internal bus might be approached, and the proper operation of the circuit is no longer guaranteed.

Using the `Irms_core` current derived from Table 4, the number of power pads needed for the VGT10 internal array is:

$$\text{Number of VSSI pads needed (VGT10)} = \frac{\text{Irms\_core (in mA)}}{\text{40 (Ma/Power Pad)}}$$

$$\text{Number of VDDI pads needed (VGT10)} = \frac{\text{Irms\_core (in mA)}}{\text{50 (Ma/Power Pad)}}$$

For VGT10, 40 mA is the maximum current rating of the internal VSSI pad, and 50 mA is the maximum current rating of the internal VDDI pad.

Using the `Irms_core` current derived from Table 4, the number of power pads needed for the VGT100 internal array is:

$$\text{Number of VSSI pads needed (VGT100)} = \frac{\text{Irms\_core (in mA)}}{\text{100 (Ma/Power Pad)}}$$

$$\text{Number of VDDI pads needed (VGT100)} = \frac{\text{Irms\_core (in mA)}}{\text{110 (Ma/Power Pad)}}$$

For VGT100, 100 mA is the maximum current rating of the internal VSSI pad, and 110 mA is the maximum current rating of the internal VDDI pad.

- **Step Three: Determine the power pairs required for SSO noise, `isso'`.**

Based on the lead inductances to be used for power pins, calculate an average inductance for all of the VSS pins.

Now calculate the average SSO drive capability of your design. Use the rated drive capability, even if your application does not use the full current handling capability of the driver.

If you have split power supplies, use Table 5 to calculate the number of SSO's per power pair.

Table 5. Maximum SSO's For Split Supplies

Power Drive (mA) / L (nH)	2 mA	4 mA	8 mA	12 mA
2 nH	17	16	15	15
3 nH	15	14	14	13
4 nH	12	12	12	11
5 nH	10	10	10	9
7 nH	8	8	7	7
10 nH	6	5	5	5
15 nH	5	4	4	4
20 nH	3	3	3	3

NOTE: With separate power buses, the number of SSO's per power pair for a certain inductance does not diminish with increased drive requirements.

If you have a single power supply, use Table 6 to calculate the number of SSO's per power pair.

Table 6. Maximum SSO's For Single Power Supply

Power Drive (mA) / L (nH)	2 mA	4 mA	8 mA	12 mA
2 nH	11	9	7	5
3 nH	9	7	4	4
4 nH	8	6	3	3
5 nH	7	4	3	3
7 nH	5	3	2	2
10 nH	3	2	1	1
15 nH	2	2	1	1
20 nH	1	1	1	1

The total number of internal power pairs required for SSO noise margins can be calculated as follows:

```
[Number of SSO's] / [SSO's per power pair]
```

- **Step Four: Determine the number of external power pairs for DC loads, `Iload`.**

Calculate your total DC load requirement, **`Iload`**, by summing the individual drive capability of each output buffer. Divide this number by the current rating per pad to obtain total power pairs required for the external power bus.

The calculation is:

```
                                                    Iload (in Ma)
Number of external power pairs needed (VGT10):      -------------
                                                   85 (Ma/Power Pad)

                                                    Iload (in Ma)
Number of external power pairs needed (VGT100):     -------------
                                                   130 (Ma/Power Pad)
```

Where 85 mA is the maximum current rating of the VGT10 power pad and 130 mA is the maximum current rating of the VGT100 power pads.

- **Step Five: Determine the total number of power pairs.**

Add the total number of power pairs calculated in Steps Two and Four, and compare this number to the number obtained in Step Three. The total number of required pairs is the larger of these numbers.

- **Step Six: Determine the distribution.**

FOR SPLIT SUPPLIES -- If the sum of the values obtained in Steps Two and Four is greater than the value obtained in Step Three, then the total number of power pairs needed is the sum of Steps Two and Four. Otherwise, use the Step Three value for the total number needed.

Bias the distribution of the internal power pairs to have the majority on the *vertical* sides of the array, where pad 1 is in the lower left corner and the VLSI logo is in the upper left corner.

If the Step Three value is greater than the Steps Two and Four value, place all the required internal and external power pairs first; you can then add any extra power pairs in the ratio of 3 to 1, favoring external pairs.

FOR NON-SPLIT SUPPLIES -- There is no need to treat internal pairs differently, but be sure that at least the number of internal pairs calculated in Step Two is located on the vertical sides of the array.

Example I - Separate Power Sources/Buses Structure

This example will demonstrate the steps used to calculate the number of required power pairs for an 8K VGT10 design. The design has a total of 80 outputs:

- 40 I/Os have 2 mA drive capability
- 20 I/Os have 4 mA drive capability
- 20 I/Os have 12 mA drive capability
- Three power pairs are available on pins 20, 40, 69, 80, 100, and 119

The clock frequency going into the array is 15 MHz, and the package used is a 120-pin CPGA. Utilization of the array is 55%. 24 I/O's switch simultaneously within a window of 100 ns and are utilized as follows:

- 10 with 12 mA drive capability
- 14 with 2 mA drive capability

The chip will have separate power supplies. These are the steps used to determine the number of power pairs required for this example:

- **Step One: Determine the power pin locations.**

Using Table 3, we find that power pins 20, 40, 69, 80, 100 and 119 in the CPGA package all have lead inductance of 5 - 10 nH. Since the pinout is fixed, all we can do is alternate the VDD and VSS pins around the array. Also, the power pins we did choose have the lowest range of inductances for the 120-pin CPGA package.

- **Step Two: Determine the power pairs required for core, `Irms_core`.**

For a VGT10 8K array, the internal core requirement from Table 4 is 4.45 (mA/MHz)*f (MHz) at 100% utilization. The value from the table is corrected for 55% utilitzation, and this results in an `Irms_core` of 36 mA.

```
Irms_core = 4.45 (mA/Mhz) * 15 (Mhz) * .55 = 36 mA.
```

Since the number of pads needed for the internal array is the `Irms_core` value in mA divided by either 40 mA for VSSI or 50 mA for VDDI, the total number of power pairs required for the core current can then be calculated as follows:

```
Number of VSSI pads needed (VGT10) = 36/40 = 0.9

Number of VDDI pads needed (VGT10) = 36/50 = 0.72
```

This design example requires one internal power pair.

- **Step Three: Determine the power pairs required for SSO noise, `isso'`.**

Since all of the power pins are located in the Medium Low inductance range (5 to 10 nH), we can assume that the average VSS pin inductance is 7 nH.

Of the 24 IOs that switch simultaneously, 10 have 12 mA drive and 14 have 2 mA drive. Now calculate the average drive capability of the SSO's:

$$\frac{(10)*(12mA) + (14)*(2mA)}{10 + 14} = 6.17\ mA$$

The closest drive value to 6.17 mA in Table 5 is 8 mA. Using 7 nH for the inductance of the VSS pins, we can consult Table 5 for the number of power pairs per SSO, which is seven. The total number of power pairs required for SSO noise can then be calculated by dividing the number of SSO's by the number of SSO's per power pair:

$$\frac{24\ (SSO)}{7\ (Power\ Pairs/SSO)} = 3.4$$

The result of this last calculation can be rounded to the nearest integer; In this case we would round 3.4 down to 3. Therefore the number of internal power pairs required for SSO noise is 3.

- **Step Four: Determine the number of external power pairs for DC loads, `Iload`.**

Worst case output buffer current load requirements are calculated by summing the individual drive capability of each output buffer:

```
Iload = (40)*(2mA) + (20)*(4mA) + (20)*(12ma) = 400 mA
```

The number of power pairs required can be calculated by dividing the load requirement, `Iload`, by the current rating per pad for VGT10, 85:

$$\text{External Power Pairs Required (VGT10)} = \frac{400\text{Ma}}{85\ \text{(mA/Power Pair)}} = 5 \text{ power pairs}$$

- **Step Five: Determine the total number of power pairs.**

Since the sum of the numbers found in Steps Two and Four (1 + 5 = 6) is greater than the number found in Step Three (3 pairs), the total number of power pairs needed is 6.

- **Step Six: Determine the distribution.**

The total number of power pairs is determined by adding the internal and external requirements. Therefore, the total number of power pairs for this design is (1 + 5) = 6. At first this requirement of 6 power pairs may seem large, but careful analysis of the design gives a smaller figure, as seen below.

Analysis. Although current requirements for the array core required only one power pair, noise requirements dictate using three power pairs for the internal VSSI/VDDI bus. The reason for the requirement of three power pairs is the 7 nH lead inductance on the VSS pins. If there were, for example, a package with 4 nH lead inductance for the same power pins, the number of power pairs for internal switching would be reduced to two.

Adding the external power requirement for the output buffer, five pairs, to the internal requirement of one pair produces a total of six power pairs. This is because it is assumed that all 80 outputs are either in the High or the Low state at the same time. All that is needed, however, is to determine the maximum number of outputs that could be in one state at one time.

For this example, a maximum number of 30 outputs are active in any one cycle, and, of those 30, 16 are Low and 14 are High.

Then, assuming that the average drive is 8 mA, all that is needed is:

$$\frac{16 \text{ (Outputs in LOW state)} * 8 \text{ (mA)}}{85 \text{ (mA/Power Pair)}} = 2 \text{ power pairs}$$

This lowers the total power pair requirement to a reasonable value of three, one internal and two external.

Example II - Non-Separate Power Sources/Buses Structure

This is the same example, only with non-separate power supplies:

- **Step One: Determine the power pin locations.**

The analysis is the same as for split supplies, but it is included here for convenience.

Using Table 3, we find that the power pins have lead inductance of 5 - 10 nH. Since the pinout is fixed, all we can do is alternate the VDD and VSS pins around the array. Also, the power pins we did choose have the lowest range of inductances for the 120-pin CPGA package.

- **Step Two: Determine the power pairs required for core, `Irms_core`.**

The internal power requirement has not changed, so we can use the value calculated in Step Two of the split supply example. The number of power pairs required for handling `Irms_core` is one.

- **Step Three: Determine the power pairs required for SSO noise, `isso'`.**

Since all of the power pins are located in the Medium Low inductance range (5 to 10nH), we can assume that the average VSS pin inductance is 7 nH.

Now determine the average drive capability of the SSO's. This can be calculated as follows:

$$\frac{(10)*(12\text{mA}) + (14)*(2\text{mA})}{10 + 14} = 6.17\ \text{mA}$$

The closest value to 6.17 mA in Table 6 is 8 mA. Using 7 nH for the inductance of the VSS pins, we can consult Table 6 for the number of power pairs per SSO, which is two. The number of power pairs required for SSO noise can then be calculated by dividing the number of SSO's by the number of SSO's per power pair:

$$\frac{24 \text{ (SSO)}}{2 \text{ (Power Pairs/SSO)}} = 12$$

This step is where the split versus non-split power pair requirements differ greatly. Calculating the number of SSO's per power pair for split supplies requires using Table 6. If you compare the numbers in Table 5 with the numbers in Table 6, you'll notice a significant decrease in the ability of a single supply to handle SSO's.

- **Step Four: Determine the number of external power pairs for DC loads: `Iload`.**

Assuming that only 16 of the outputs are in the Low state during a cycle, and that the average drive is 8 mA, the number of external power pairs can be calculated as follows:

$$\frac{16 \text{ (Outputs in LOW state)} * 8 \text{ (mA)}}{85 \text{ (mA/Power Pair)}} = 2 \text{ power pairs}$$

- **Step Five: Determine the total number of power pairs.**

Since the number found in Step Three (12 pairs) is greater than the sum of the numbers found in Steps Two and Four (1 + 2 = 3), the total number of power pairs needed is 12.

• **Step Six: Determine the distribution.**

The total number of power pairs is determined by adding the internal and external requirements. Therefore the total number of power pairs for this design is (1 + 2) = 3. However, the requirement for SSO's is 12 pairs, so you need a total of 12 pairs. Start with one internal pair and two external pairs. Then split the remaining nine pairs into six external pairs and three internal pairs.

Analysis. While the external power requirements for the whole array could be handled by using only two power pairs, the noise requirement dictates using 12 power pairs to ensure that the input level shifters operate at normal TTL levels.

This demonstrates both the desirability of separate power supplies when possible and the neccessity of choosing low-lead-inductance pins for power pad bonding.

Preferred Power Pad Locations

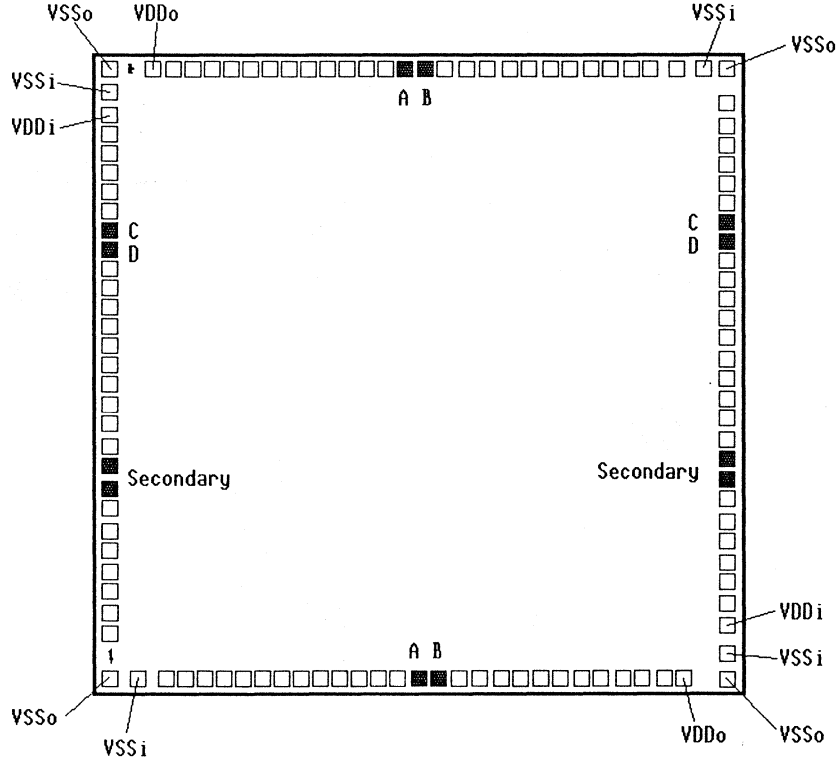
VLSI has adopted preferred power pad guidelines to reduce your cost of manufacturing. Following these guidelines can reduce your design cost and shorten prototype delivery time. If you chose not to use the preferred power pad locations, you should notify your Design Center so that specialized wafer sort hardware can be built. This hardware build increases your NRE charges and delays tested prototype delivery.

Preferred Power Pad Locations

Place one power pair, shown as A and B in Figure 7, in each of the centers of the horizontal sides. These are the two primary power pairs to the chip and are mandatory. These preferred power pad locations are different for each array. Refer to Table 7 for the pad numbers of the array you selected.

If you are using a single power bus, use power macro PCVDD3 in the A position and PCVSS3 in the B position. Also place PCVDD3 and PCVSS3 macros in the corners as indicated.

Figure 7. VGT10 And VGT100 Preferred Power Pad Locations



If you choose the recommended split power bus arrangement, place a PCVDD1 macro in the A position and a PCVSS1 macro in the B position. Use PCVDD2 and PCVSS2 on the vertical sides of the array, as shown, in locations C and D respectively. Additionally, for VGT100 arrays, place PCVDD1 and PCVDD2 macros in the two indicated corners of the array.

For large arrays, VGT10-6400 and up, additional power pairs may be added on the vertical sides of the array, about 1/3 of the way from the top and bottom. These secondary power pairs can be PCVDD1 and PCVSS1 types.

The pad numbers of these locations for all VGT bases is given in Table 7. Pads are numbered clockwise, starting at the arrow.

Table 7. Pad Location Pin Numbers

VGT BASE	PRIMARY				SECONDARY	VDDI	VDDO	VSSI	VSSO
	A	B	C	D					
VGT10-016	23, 56	24, 55	8, 39	7, 40				11, 23, 35, 47	12, 24, 36, 48
VGT10-024	28, 66	29, 65	10, 46	9, 47				18, 37, 56, 75	19, 38, 57, 76
VGT10-037	34, 80	35, 79	12, 56	11, 57				22, 45, 68, 91	23, 46, 69, 92
VGT10-064	43, 101	44, 100	21, 65	20, 66	8, 9, 77, 78			28, 57, 86, 115	29, 58, 87, 116
VGT10-084	49, 115	50, 114	23, 75	22, 76	10, 11, 87, 88			32, 65, 98, 131	33, 66, 99, 132
VGT10-106	55, 129	56, 128	26, 84	25, 85	11, 12, 99, 100			36, 73, 110,131	37, 74, 111,148
VGT100122	59, 140	60, 139	29, 107	28, 106	12, 13, 90, 91	38, 118	41, 121	39, 119	40, 120
VGT100160	67, 157	68, 156	31, 122	30, 121	12, 13, 102, 103	43, 133	46, 136	44, 134	45, 135
VGT100205	76, 178	77, 177	35, 137	34, 136	15, 16, 117, 118	49, 151	52, 154	50, 152	51, 153
VGT100270	87, 202	88, 201	40, 156	39, 155	17, 18, 133, 134	56, 172	59, 175	57, 173	58, 174
VGT100397	103, 246	104, 245	48, 189	47, 188	21, 22, 160, 161	68, 208	71, 211	69, 209	70, 210
VGT100528	119, 280	120, 279	55, 213	54, 214	25, 26, 183, 184	78, 238	81, 241	79, 239	80, 240

Preparing a Bonding Diagram

Preparation

A preliminary bonding diagram is a very important part of your initial design specifications. This section offers guidelines for preparing your bonding diagram. After manually preparing the bonding diagram, give it to your VLSI Design Center engineer, who checks the electrical and mechanical connections.

Before you prepare the bonding diagram, you must:

- Choose an array size and package.
- Establish your preferred pinout.
- Obtain blank build sheets for the package you selected from your VLSI Design Center.

Prepare the bonding diagram after you have chosen a package and preferred pinout but **before** you create and simulate the design. Please keep in mind that, although most designs can be implemented with your preferred pinout, this is not always possible; VLSI cannot guarantee the exact pinout until the placement and routing phase has been completed.

After you have chosen your device, established the preferred pinout, and selected a package, ask your VLSI Technology Design Center for blank build sheets.

Completing the Bonding Diagram

Bonding diagram guidelines for the VGT series are the same for all array sizes. To complete the bonding diagram:

1. Obtain the footprint of your base array from the Technology Center, cut it out, and place it in the center

of your chosen build sheet form. Be sure that the footprint is oriented correctly within the build sheet, and that *the build sheet and footprint are the same scale.*

2. Locate the power pads. Dedicated ground pads, VSSI and VSSO, are located around the perimeter of each array. Use these pads as power pads only; do not use them as signal pads. VSSI pads connect to the ground bus associated with the internal core and input level shifters of the array. VSSO pads connect to the ground bus associated with the output buffers. VLSI has isolated the output buffer and internal ground bus structures to prevent current spikes from causing erroneous switching.

3. Use preferred power pads first. Review the section entitled **Specifying Power Pads**, beginning on page 32 in this chapter.

4. Draw bonding wires. When you have placed all input and output buffers, use a pencil to draw bonding wires connecting each input or output pad on the footprint with its corresponding package pin on the build sheet. Do not allow two wires to cross. Wire angles cannot be so acute that they cross over either adjacent pads or adjacent package pins.

For quick turnaround, prototypes are always packaged in ceramic. However, the empty blanks sent to you are for plastic packages, because such build sheets are more critical. In addition, whatever we plan in plastic can be built in ceramic with identical pin assignments.

Observe these basic guidelines:

- Do not crosswire a pad from one side to a pin on an adjacent side.
- Try to keep the wires as short as possible and as concentric to the center of the die as possible.
- To avoid confusion, a wire must start at the center of the pad and go to the center of the pin leg. Any ambiguity can cause errors and confusion.

IMPORTANT

All bonding diagrams must be approved by your VLSI Technology Design Center engineer prior to placement and routing.

CHAPTER 4

IMPLEMENTING YOUR DESIGN

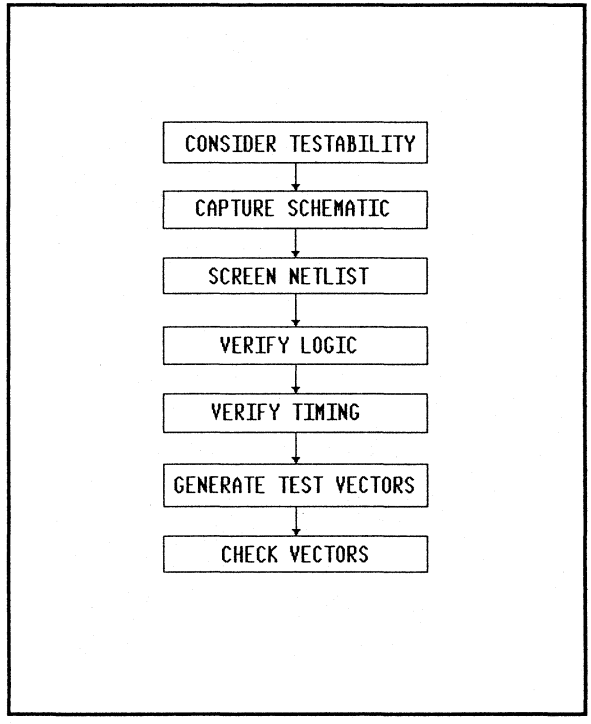
Introduction

This chapter describes how to implement your design using the VGT portable library and VLSI design tools. Each section addresses important and necessary steps you need to take while implementing your design. This does not cover every aspect of the design process, but tries to highlight the critical areas.

Design Implementation Flow

Figure 8 outlines the steps necessary to create a Gate Array design with the VLSI tools. This flow is not rigid, but instead shows the necessary steps in their relative order. Each of the steps are treated in more detail in the following text.

Figure 8. Recommended Design Implementation Sequence



Consider Testability

You can greatly reduce the amount of work involved in simulating and creating test patterns for some designs by adding a small amount of testability logic. For this reason, the question of circuit testability should be addressed at the very beginning of the design process. For highly sequential circuits, testability issues can be complicated, and test logic can consume a significant portion of the complete logic design. Even for simple circuits, however, there are a few fundamental testability issues to keep in mind when designing a circuit.

Circuit Initialization

The circuit must be initializable to a known state before testing can proceed. Even though a circuit may function correctly regardless of the initial state of storage elements upon power-up, this is not acceptable for circuit testing. In order for a test program to be generated from a simulation trace file, *no internal nodes of the circuit may be stimulated at any time during the simulation.*

Test Modes

Test modes are a convenient way to add simple bypass multiplexers to key inputs and outputs of the circuit, and to directly access important internal nodes. If no I/O pin is available for use as a test mode input pin, it is often possible to define a test mode as some illegal combination of inputs.

Counters and Sequencers

A large counter or sequencer that cycles through its course hundreds of times per second in an actual system can take hours to simulate. Counters and sequencers are often very important candidates for test logic because a simple bypass multiplexer can allow a multi-bit counter or sequencer circuit to be tested in a fraction of the time required by normal operation.

Additional Circuitry vs. Long Simulations

Designers are sometimes reluctant to add test logic because they are trying to minimize the logic. It is important to consider the cost of long simulations during the design cycle, and to make the appropriate trade-offs regarding testability logic at the beginning of the design phase.

VLSI Design Center engineers are available to assist with testability considerations at the beginning of the design phase.

For more details about guidelines for generating test patterns, see the application note called **Test Generation Guidelines.**

Capture Schematic

Before capturing your design schematic with the VLSI schematic editor, you should read the VTIschematic manual and attend one of VLSI's tools classes. This section presents some guidelines to follow when entering your schematics on VLSI tools.

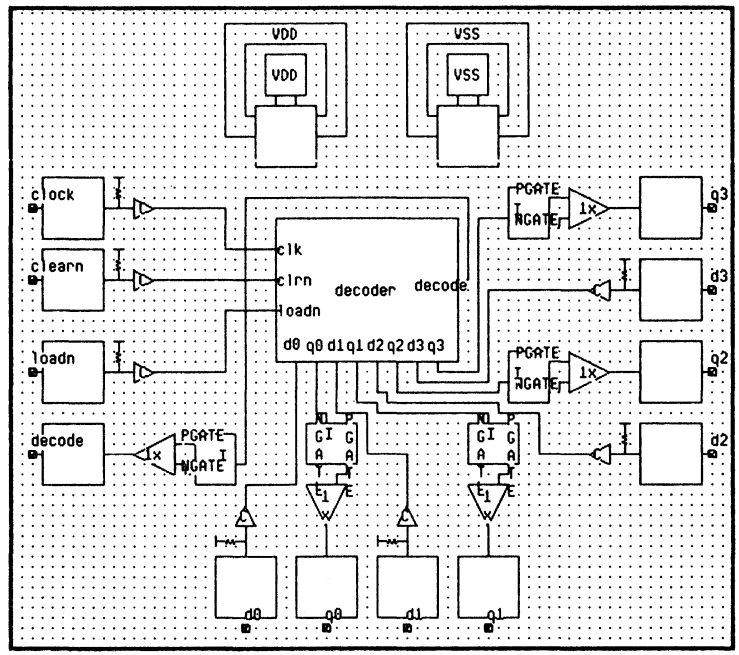
Partitioning Considerations

You should think about partitioning your design before you enter your schematic. Partitioning the design at the beginning helps you enter the schematic faster and helps you keep track of cell utilization. Screen the netlist frequently while entering the design so that you can catch VLSI netlist rule violations and measure your cell utilization. See the section entitled **Screen Netlist**, later in this chapter, or read the VTIscreen manual for more details.

Top Level Schematic Considerations

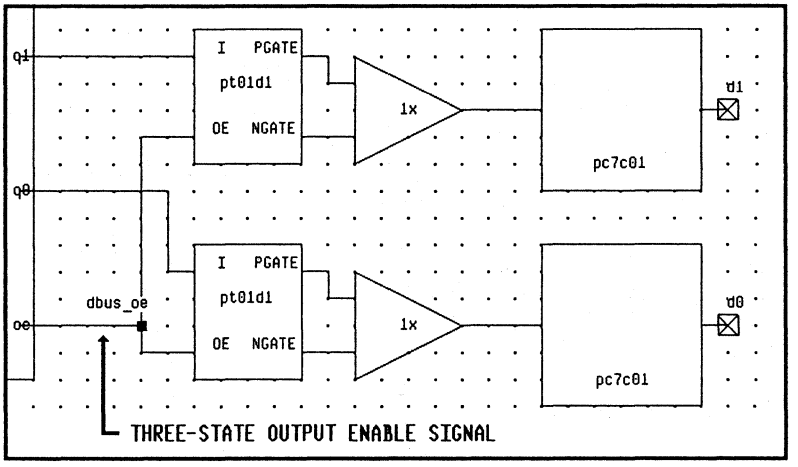
The top level of the design hierarchy should contain all of the I/O pads and a core block (Figure 9). Pad signals should have a connector on the pad node. Only I/O pad signals should have connectors on them in the top level. The VLSI netlist screener will report an error otherwise.

Figure 9. Top Level Of Design



Three-state, bidirectional, open drain and open source pad control signals should be explicitly named (Figure 10). Refer to the VTIschematic manual for the syntax of legal signal names. Also, all three-state, bidirectional, open drain and open source output buffers must be driven by a pad driver. The output pad's NGATE and PGATE pins must be connected to a single driver cell's NGATE and PGATE pins respectively.

Figure 10. Pad Drivers



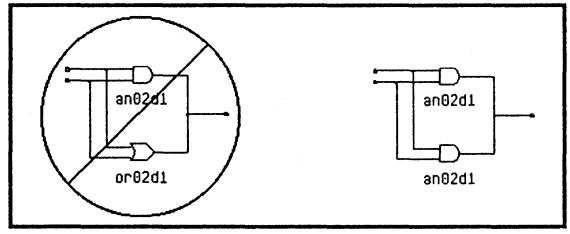
Provide Adequate Signal Drive

To provide adequate signal drive:

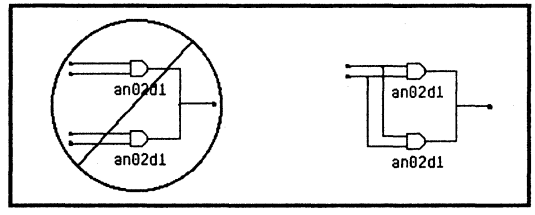
- Buffer all internal signal outputs sufficiently.
- Internal signal rise/fall delays should not exceed 10ns.
- Clock signal rise/fall delay should not exceed 2ns.
- Be aware of output diffusion capacitance on three-state gates.
- Create a clock "tree" to adequately buffer a clock signal and reduce clock skew.

Parallel Logic

Do not parallel macros with dissimiliar logic functions. The VLSI netlist screener produces an error message if a signal is driven by more than one type of cell unless the macros are three-state. This restriction prevents the possibility of contention:



If a signal is driven by macros in parallel but whose input signals are not common, the VLSI netlist screener produces an error message unless the macros are three-state. This restriction prevents contention:



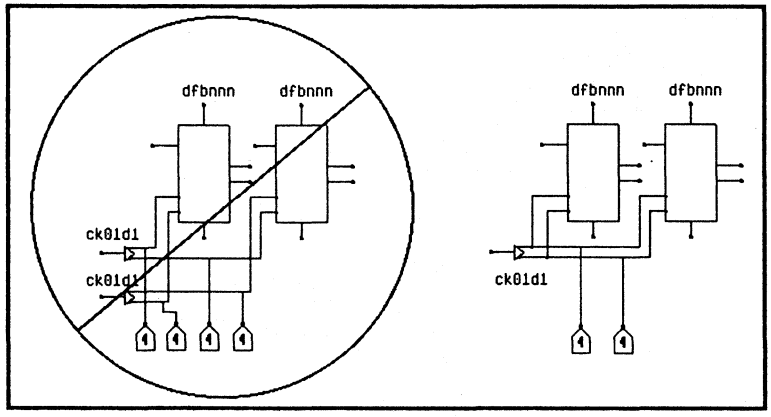
Bus Repeater Cells For Three-State Nodes

Bus repeater cells prevent excess power consumption by floating three-state nodes driving other logic. The bus repeater cell, RP01D1, holds the last value that was driven onto the node. Only one bus repeater cell should be used for each three-state node. Bus repeater cells should not be used on nodes that are not three-state. If the circuit is such that the three-state node never floats, a bus repeater is not required.

Clock Buffers

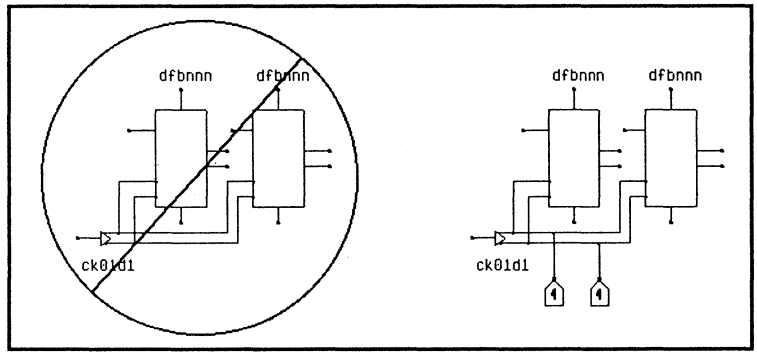
All unbuffered flip-flops and latches must be driven by a clock buffer cell in order to minimize clock skew.

A clock buffer cell's output pins must drive the same cells in order to reduce the amount of skew between output pins C and CN:



Weights on Clock Buffers

The clock buffer cell's output pins need to be weighted in order to minimize wire length and clock skew. The weight on the clock buffer cell's output pins must be balanced in order to minimize wire length and clock skew, as shown in the illustration which follows.



VLSI Reserved Signal Names

Do not use VDD or VSS to refer to anything other than power or ground. This includes names that have "vdd" or "vss" in them, such as "myvdd."

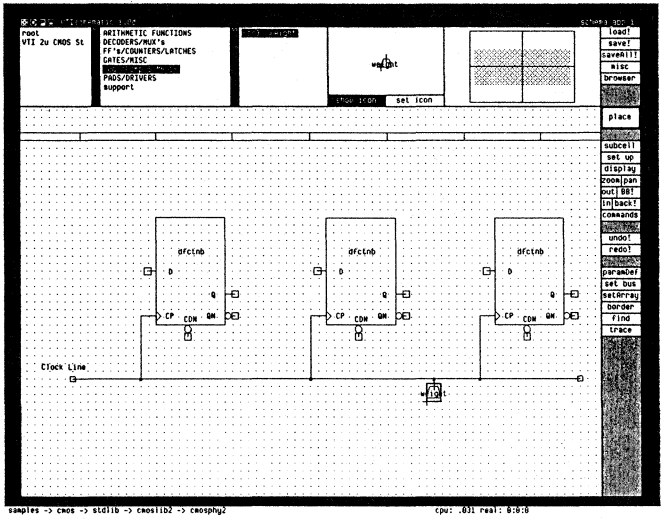
Place and Route Information in Your Schematic

You can choose to specify pad placement or routing priority by using net weights in your schematic, instead of in a critical placement file.

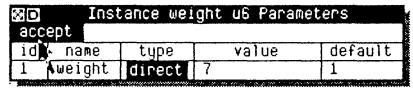
Assigning Routing Priority In VTIschematic

Optionally, routing priority may be assigned in VTIschematic by placing a weighting icon on the net. You may wish to refer to the description of the WEIGHT macro in the appropriate **VGT Macro Library Manual.**

To assign routing priority in VTIschematic, view a section of the net you are going to weight and select the WEIGHT cell in the browser. This cell is located in the library tree under `logicComp SYMBOLS`. Then click on [▮▯▯] → [`place`], and place the icon near the line. Wire from the weight icon to the selected line as you would any other cell:

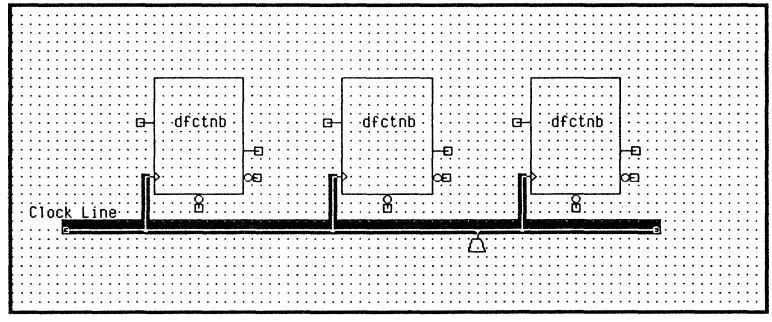


To set the value of the weight, click on commands → SetParam to bring up the weight property sheet:



The value set in this box becomes the routing priority for that net. All nets have a default weight of 1. A net with a higher weight is considered more important than other nets at placement time. A net assigned a weight of 5 is considered 5 times as important as a normal net.

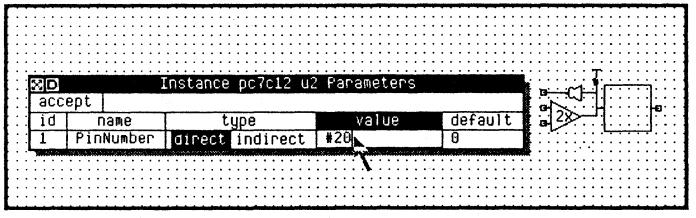
Only one weight has to be attached to a net. That is, everything traced in a single net by VTIschematic has the same routing priority. The entire net highlighted in the illustration below has the same priority:



Specifying Pad Placement in your Schematic

Optionally, you can assign pad placement in VTIschematic by modifying the property sheet associated with a specific pad.

You can modify the property sheet for a pad by clicking on [commands] → [SetParam], marking the pad, and pressing [□□■] to bring up the parameter sheet. The value assigned to PinNumber is used for the die pad number for this pad. Put a pound sign, (#), **before** the pad number, as shown. Any number not preceded by a pound sign is interpreted as the package pin number, not the intended die pad number:



Screen Netlist

When your schematic is complete, check it with VTIscreen. VTIscreen identifies simple errors and other potential problems in your logic design. It also provides useful information about the utilization of the array you have chosen. VTIscreen should be run before sending a design to VLSI for placement and routing. It is recommended that you run VTIscreen routinely during the design process in order to keep track of your array utilization and to find any errors that exist at that time.

How to Run VTIscreen

You run VTIscreen from VTIshell or in a VTIterminal window running `shell!`.

- **Type `utility screen` to the `VTI>` prompt.**

You get a `VTIscreen>` prompt.

- **Type ? to see the functions available in VTIscreen:**

```
VTIscreen> ?
  ARRAY <array name>     name of gate array, such as VGT10600
  EXIT                   exit VTIscreen
  READ <cell>            read an HNS or FNS netlist
  OUTPUT <cell name>     cell name of the VTIscreen report
  QUIT                   exit VTIscreen
  SAVE                   save the netlist as a LDS
  SCREEN                 screen the netlist using the selected values
  STATUS                 show current status
  VDD <number of vdd's>  number of extra vdd power pads needed
                           (only for VGC and VGT10 using the Gate
                            Array Macro Library)
  VSS <number of vss's>  number of extra vss power pads needed
                           (only for VGC and VGT10 using the Gate
                            Array Macro Library)
```

- **Type `read` *cell***

The *cell* is the name of an HNL netlist, such as `[hns]`*design*.

- **Type `array` *array name***

The *array name* is the name of the gate array family, such as VGT10600. Valid array names are:

If you have this array:	Use this VTIscreen array name:	If you have this array:	Use this VTIscreen array name:
VGT10-016	VGT1600	VGT100122	VGT12000
VGT10-024	VGT2400	VGT100160	VGT16000
VGT10-037	VGT3700	VGT100205	VGT20000
VGT10-064	VGT6400	VGT100270	VGT27000
VGT10-084	VGT8400	VGT100397	VGT40000
VGT10-106	VGT10600	VGT100528	VGT53000
		VGT100665	VGT67000

- **Type** `output` *cell name*

The *cell name* is the name of the VTIscreen output file, such as `[scr]`*design*. The cell type is SCR.

- **Type** `vdd` *number of vdd's*

The *number of vdd's* is the number of extra VDD power pads in the design. This command should only be used when screening VGC designs or VGT10 designs using the Gate Array Macro Library. When screening other products, the VDD power pads should be entered into the design using VTIschematic.

- **Type** `vss` *number of vss's*

The *number of vss's* is the number of extra VSS power pads in the design. This command should only be used when screening VGC designs or VGT10 designs using the Gate Array Macro Library. When screening other products, the VSS power pads should be entered into the design using VTIschematic.

- **Type `screen` to screen the netlist using the selected values.**

- **Type `save` to save the netlist as an `[lds]` file.**

This command should be used when your design does not contain errors and you want to run VTIvector. The cell name of the output LDS cell will be the same as the name of the HNL cell that was read in. For example, if `[hns]`*design* is the name of the top level HNL cell, then `[lds]`*design* will be created. Using the SAVE command is the only way to have VTIscreen produce the LDS cell.

- **Type `quit` to return to `VTIshell`.**
- **Type `status` to find out the current status.**

Sample Screener Output

This section shows a representative sample of a VTIscreen report:

```
VTI> utility screen


VTIscreen 1.2
VTIscreen> read [hns]vgt10            read your design into VTIscreen
VTIscreen> array vgt10600             indicate the array size of your design
VTIscreen> screen                     start the screening process


     VTIscreen - vgt10 design.

*****************************************************************************
*                                                                           *
*       VTIscreen Utilization Summary for vgt10 - VGT10600 Gate Array       *
*                                                                           *
*   Number of Input Pads                                          47        *
*   Number of Output Pads                                         38        *
*   Number of Bidirectional Pads                                   4        *
*   Number of VDD                                                  0        *
*   Number of VSS                                                  0        *
*   Number of VDD Pads                                            12        *
*   Number of VSS Pads                                            12        *
*   Number of VDD Core                                            12        *
*   Number of VSS Core                                            12        *
*                                                                           *
*   Number of Available Pads                                     140        *
*   Number of Pads Used                                          137        *
*   Percent of Pads Used                                       97.86%       *
*                                                                           *
*   Number of Available Placement Sites                        21296        *
*   Number of Placement Sites Used                             17883        *
*   Site Utilization Percentage                                83.97%       *
*                                                                           *
*   Number of Gate Equivalents Used                          5790.00        *
*                                                                           *
*   Number of Non-Primitive Blocks Used                            7        *
*                                                                           *
*****************************************************************************

Design Statistics:
  Number of Nets                  : 1754
  Number of FromTos               : 6158
  Average Number of Pins per Net  :  3.5
  Maximum Number of Pins per Net  :   50
```

```
Design Complexity:
```

Cell Name	Number of Occurrences	Number Gate Equivalents Per Cell	Total Gate Equivalents	Number Sites Per Cell	Total Sites
ao01d1	48	2.00	96.00	5	240
dfntnb	26	7.00	182.00	19	494
in01d1	152	.50	76.00	2	304
in01d2	396	1.00	396.00	3	1188
in01d3	4	1.50	6.00	4	16
it01d1	768	1.50	1152.00	5	3840
it01d2	12	2.00	24.00	6	72
it02d1	4	1.50	6.00	4	16
it02d3	24	3.00	72.00	9	216
lantnb	456	4.50	2052.00	14	6384
mx21d1	72	3.00	216.00	8	576
nd02d1	69	1.00	69.00	3	207
ND03D1	4	1.50	6.00	4	16
nr02d1	65	1.00	65.00	3	195
NR03D1	3	1.50	4.50	4	12
NR04D1	32	2.00	64.00	5	160
NR05D1	16	2.50	40.00	6	96
nt01d1	360	2.00	720.00	6	2160
nt01d2	12	2.50	30.00	8	96
nt01d3	24	3.50	84.00	11	264
OA02D1	1	3.00	3.00	7	7
OA04D1	3	1.50	4.50	4	12
pc7c00	47	.00	.00	0	0
pc7c02	4	.00	.00	0	0
pc7o01	14	.00	.00	0	0
pc7o02	24	.00	.00	0	0
pcvdd1	12	.00	.00	0	0
pcvdd2	12	.00	.00	0	0
pcvss1	12	.00	.00	0	0
pcvss2	12	.00	.00	0	0
pt04d1	28	5.00	140.00	16	448
xn02d1	12	2.50	30.00	9	108
xo02d1	84	3.00	252.00	9	756
Totals:	2812		5790.00		17883

```
****   Summary of design warnings and errors   ***

WARNING>  the three-state signal u143.DATA originating from
          the following location should be connected to
          1 or more three-state cells:
```

Cell name	Pin name	Instance name
nt01d3	Z	u2.m22901_1.ybuf.b23.u2

```
ERROR>  Illegal parallel drives; the signal u2.QMSB
        originating from the following locations is
        driven by more than one type of cell:
```

Cell name	Pin name	Instance name
it02d1	ZN	u41.spec3
nt01d1	Z	u2.m22901_1.qshift.b23.u8

```
WARNING>  the three-state signal u45.DATA originating from
          the following locations is not connected to
```

```
          a bus repeater cell:

            Cell name     Pin name     Instance name
            ---------     --------     -------------
            1t02d1        ZN           u45.spec3
            nt01d1        Z            u2.m22901_1.ramshift.b23.u8

ERROR>  The following signals are not driven by any cells:

          u207.u2.PGATE

WARNING>  The following signals do not drive any cells:

            u2.m22901_1.rama.b23.u1.QN          u2.m22901_1.rama.b22.u1.QN
            u2.m22901_1.rama.b21.u1.QN          u2.m22901_1.rama.b20.u1.QN
            u2.m22901_1.rama.b19.u1.QN          u2.m22901_1.rama.b18.u1.QN

End of screener check
```

Simulation

Before simulating your design netlist on the VLSI simulator, you should read the VTIsim manual and attend one of VLSI's tools classes. This section discusses what you need to do while simulating your netlist. In general, you should:

- Verify the functionality of your design.
- Verify the timing requirements of your design.
- Create test vectors for prototype and production testing.

Logic Verification

The logic verification step determines if the design or sub-circuit of the design is functioning as specified. VLSI suggests that you thoroughly simulate all subcircuits in your design. Don't wait for top level simulations to find a logic error. Tracking down a netlist error at the final simulation stage can be very time-consuming and difficult.

Timing Verification

Timing verification allows you to detemine if your design functions properly at your clock speed and that all critical speed paths meet their specified timing.

Estimate Routing Capacitance

Use estimated routing capacitance while simulating your design. To add estimated routing capacitance to your netlist, use the proper SIMPARMS command and parameters in the simulator. The easiest way to enter the SIMPARMS command is to use the command file supplied with your library. The following example illustrates how to account for predictive routing capacitance in the simulator:

```
VTIsim> load [sim]predcap
VTIsim> load [hns]design
```

Provide Output Load Capacitance

Include the external load capacitance that the output buffers will see in your design application. To load your output buffers, use the SET CAPACITANCE command in VTIsim. The following example loads output RAS with 50pF and output CAS with 150pF. Default units for the simulator are picofarads.

```
VTIsim> set capacitance 50  RAS
VTIsim> set capacitance 150 CAS
```

SET CAPACITANCE overrides any previous capacitance value assigned or computed for a node. If the load on RAS is currently 5pF, and you give the SET CAPACITANCE 50 command, the load on RAS becomes 50pF, not 55pF. If you wish to increase the existing load, SET CAPACITANCE to the total value desired.

Worst Case Timing Can Hide Best Case Hazards

All of VLSI's portable library models use worst case process timing in the models. You should analyze your circuit for race conditions caused by best case processing. One possibility is to use the VTIsim SET SIMPARMS DELAYFACTOR n statement, where n is the best case derating factor for process, temperature or voltage factors. Derating factors are given in the **General Information** in the **VGT10 Macro Library** manual, in this binder. If necessary, contact the VLSI Technology Design Center for assistance.

Generate Test Vectors

The application note called **Test Generation Guidelines** describes VLSI's requirements for developing test vectors. This application note should be read and understood before you create your test vectors. This section provides a short synopsis of the test vector requirements; you should refer to the application note for more information.

Run Test Generation Simulation At 1MHz

The purpose of the test generation simulation is to generate **functional** test patterns, *not to verify system timing constraints.* Check the system timing constraints carefully during the system simulation phase. The fact that a circuit functions within the timing range predicted by the simulation can be guaranteed by a critical path check.

Limit Test Vectors To 10,000

You should submit no more than 10,000 vectors with your design. VLSI Technology provides testing services at an

additional cost for customers who request more than 10,000 vectors.

Watch All Primary I/O and Control Signals

All I/O signals and all control signals on bidirectional and three-state output signals must appear in the test vector trace file. The reason that control signals must be included is that the vector conversion software uses the value of the control signal to determine whether a bidirectional is acting as an input or an output, and whether or not a three-state output is driving out.

Separate Edge Times By 100ns

Defining edge times at 100ns intervals ensures that the device has time to settle before the next transition. This helps prevent race conditions while functionally testing your device.

Check Vectors

VTIvector provides a vector checking command, CHECK, that scans a set of trace vectors for adherence to VLSI's standard vector guidelines. These guidelines ensure that the vectors are easily convertible to a variety of testers. While checking the trace file for conformance to standard vector guidelines, CHECK creates the utility files that are required to convert the vectors.

The CHECK command requires only a trace file and netlist for inputs, and produces not only an error report, VER, but a complete trace definition file, TDF, as well.

How to Check Vectors

The VTIvector CHECK command examines vectors for conformity to an expected timing format. The timing format can be specified explicity, by providing an ITG or TDF file, or the timing information can be derived from the trace vectors as they are checked.

If an LDS netlist is supplied, the CHECK command creates a TDF file. Otherwise, a TDF file, either with or without explicit timing information, must be supplied as input to the CHECK command. If a TDF file without timing information is supplied, the TDF file is modified to include timing information.

Files Required

Input files required by the CHECK command are:

- Simulation Trace File, `[trc]` -- The simulation trace file is the vector input file.
- Logical Netlist Data File, `[lds]` -- The LDS file provides information about the netlist I/Os.
- Input Timing Generator File, `[itg]` -- Optional. Specifies explicit input timing for which to check.
- Trace Definition File, `[tdf]` -- If an LDS file cannot be provided, a TDF file must be supplied to identify the I/O characteristics of the circuit. The TDF input need not have timing information in it.

Cycle duration and trace type must also be specified for the CHECK command.

The output files created by CHECK are:

- VIF Error Report, `[ver]` -- The VIF error reports lists all variations from expected timing.
- TDF Trace Definition File, `[tdf]` -- If a TDF file is not provided as an input, it is created by the CHECK command. If a TDF file is provided as an input, it is modified to include correct timing information.
- ITG Input Timing Generator File, `[itg]` -- If an ITG file is specified as an input, it is not modified by CHECK.

Example

The most common use of VTIvector is checking trace vectors to make sure they pass the guidelines recommended by VLSI Technology for creating prototype test programs. In the process of successfully checking your vectors for correctness, VTIvector creates an ITG file and a TDF file from the trace file. If for some reason, such as the unavailability of a netlist, it is not possible to create an LDS file, the vector checker can still be used, but you must create a TDF file by hand.

Be sure that your trace file and LDS file are available, then call the VTIvector checker by typing:

```
VTIvector> check
VTIvector.check> input [trc]counter
VTIvector.check> lds [lds]counter
VTIvector.check> itg [itg]counter (optional)
VTIvector.check> duration 1000 (defaults to 1000 ns)
VTIvector.check> source vlsi (defaults to VLSI)
VTIvector.check>                                          output counter
([ver],[itg], and [tdf] outputs all share the same cellname)
VTIvector.check> create
##########################################################################
####        No Errors - Vectors are Super Synchronous                 ####
##########################################################################
VTIvector.check> quit
```

CHAPTER 5

PRE-ROUTE DESIGN REVIEW

Design Preparation

After you specify the physical characteristics of your chip, and create and simulate the design schematics on your workstation, prepare a design package for formal acceptance by VLSI so the Design Center can place and route your chip. This chapter provides guidelines for deriving specifications and test patterns from your design, as well as background information for filling out the forms required at this phase of the design cycle.

During this preparation phase, you:

- Prepare the netlist
- Screen the netlist for schematic errors
- Prepare the 1MHz test patterns
- Prepare the Critical Placement File (optional)
- Prepare bonding diagrams
- Prepare the Pad Placement File

- Initiate the signoff review checklist

These steps are described in detail in the sections that follow.

Preparing the Netlist

HNS Format

Your design must be in VLSI's HNS format to submit it for placement and routing. Schematic graphic files, created using VTItools, are also acceptable. Follow the directions in the VTIschematic manual for creating the HNS netlist.

Netlist Screening

VTIscreen is a tool that can be used at any point during the design process to check for schematic entry errors. The screener points out floating nodes and unconnected outputs, as well as giving you cell counts and utilization factors. Once the netlist is finalized and no more logic changes are expected, do a final screening. The resulting report file, `[scr]`, with the netlist, is submitted to VLSI Technology during the design review.

Preparing Test Patterns

VLSI Technology requires that each gate array netlist be submitted with a 1MHz test pattern to be used in the manufacturing process. Please refer to the application note, **Test Generation Guidelines**, for detailed information about generating your test vectors.

The test vectors VLSI requires for test program generation are output directly from your simulation files. However, you will

want to be selective in choosing test vectors, since the vectors used by VLSI to generate test programs must conform to certain restrictions. In order to clarify this situation, two different types of simulation are defined here:

- **Engineering Simulation -** The engineering simulation is the simulation that the design engineer deems adequate assurance that his circuit will operate correctly in his system. This can include an unlimited number of vectors, simulated at the expected system speed, with any kind of timing the engineer chooses to provide. These simulations are generally the basis for both the pre-route logic verification simulation and signoff, and the post-route performance simulation and signoff.

- **Test Generation Simulation -** This simulation must be run at a 1MHz synchronous clock speed and can consist of no more than 10,000 vectors. The timing on the input patterns must conform to the guidelines outlined in the application note, **Test Generation Guidelines.**

VLSI requests that you submit, along with your test generation simulation file, an Input Timing Generator (ITG) file, created with the file name *design*.ITG. The purpose of the ITG file is to specify the timing behavior of the input signals of your device during your test generation simulation. This helps generate the proper stimulation of the circuit.

The optional Critical Path Check (CPC) file specifies the timing at which the outputs are tested in order to check for proper performance of critical paths on the circuit. Refer to the application note referenced above for information on generating the ITG and CPC files.

Creating a Critical Placement File

Introduction

The Critical Placement File (CPF) is an optional file which is used to fix macros to a specific location on the placement grid. This is useful when one or more critical paths exist in a circuit whose performance depends on their relative placement.

The Critical Placement File is a text file which you submit to VLSI Technology along with your netlist. The information in the critical placement file is used by the VLSI placement tools to preferentially place the instances you specify. The CPF file name has the form *design*.CPF.

Additionally, you can specify net weighting to have the router preferentially route certain nets to help minimize the wiring capacitance on critical nodes. Both the critical placement and net weighting options can be used simultaneously to help optimize the circuit layout.

Optionally, pad placement and net weighting can be specified in your schematic, as shown in the **Place and Route Information in Your Schematic** section, in Chapter 4, IMPLEMENTING YOUR DESIGN. This information must be consistent with any information given in your pad placement file and critical placement file, or delays in the place-and-route cycle may result.

CPF File Format

The critical placement file contains all the fixed initial macro placements. This file is usually in the top level of your design hierarchy. During the place and route operation, VLSI Technology engineers add I/O information to this file.

- **General Information**

The CPF begins with the following information, which may appear in any order:

$SUBMITTER *name*

Where *name* is the name of the person or company submitting the design to VLSI. This statement is required for all designs.

$DESIGN *design*

Where *design* is the name you have given your design. This statement is optional.

$PARTNUMBER *partNumber*

Where *partNumber* is the VLSI-assigned option number for your design. This statement is required for all designs.

$ARRAY *arrayName*

Where *arrayName* is the name of the array base used for your design, such as VGT1600. Valid array names are the same as those used for VTIscreen, listed on page 73. This statement is required for all designs.

$PACKAGE *packageName*

Where *packageName* is the VLSI bond diagram form number of the prototype package to be used for your design. This statement is required for all designs.

$PRODUCTION_PACKAGE *packageName*

Where *packageName* is the VLSI bond diagram form number of the production package to be used for your design. This statement is optional.

$NETLIST *fileName*

Where *fileName* is the name of the top level of the netlist for your design. This statement is required.

comment text

Comments begin with the pound sign.

• Placements Section

After these statements there is an optional placements section. This section begins with the following keyword:

$PLACEMENTS

after which you can specify net weights or fixed placements as described below.

• Net Weights

To specify a net weight, add a statement of the form:

W *signalName weight*

Where *signalName* is the signal name assigned to the critical signal in your netlist file and *weight* is a number between 1 and 100. For example, the statement W RESET 15 assigns a weight of 15 to the signal named RESET.

All nets have a default weight of 1. A net with a higher weight is considered more important than other nets at placement time. A net assigned a weight of 5 is considered 5 times as important as a normal net.

The net weight feature should be used in moderation. It is recommended that only critical nets be weighted and that common nets, such as reset lines and other signals that are used widely in the circuit, not be weighted.

• Fixed Placement - Internal

To fix an internal macro or cell, specify:

I *instanceName row column*

Where *instanceName* is the instance name of the component in your schematic or netlist, *row* is the row number and *column* is the column number of the site to be used. For VGT10 and VGT100 arrays, row 1 is the bottom row and column 1 is the leftmost column. For example, the statement `I U105 1 1` places instance U105 at the bottom left corner of the array core.

Every macro is wholly contained on one row. The origin of each macro is the bottom left corner, with the macro occupying as many columns to the right of the origin as it has cells. The cell count for each macro is given under "Base Count" in its data sheet. Take this into account when placing macros to avoid accidentally overlapping them on a particular row. The base count includes isolation transistors, and adjacent macros share these isolators. The space on a row required for one cell is equivalent to one column. The VLSI place and route software checks proper placements and a netlist is returned to you if it detects any placement errors.

Table 8 lists the number of rows and columns for each of the VGT arrays.

Table 8. VGT Site Rows And Columns

BASE	ROWS	COLUMNS
VGT10-016	17	187
VGT10-024	21	231
VGT10-037	26	286
VGT10-064	34	374
VGT10-084	39	429
VGT10-106	44	484
VGT100122	47	517
VGT100160	54	594
VGT100205	61	671
VGT100270	70	770
VGT100397	85	935
VGT100528	98	1078
VGT100665	110	1210

• Fixed Placement - Pad

To specify a fixed placement for a pad, the syntax is:

P *instanceName—or—signalName* *pkgPinNumber* #*padNumber*

Where *instanceName—or—signalName* is the instance name of the pad or a defined signal name in your schematic or netlist, *PkgPinNumber is the package pin number to* be assigned to this instance or signal, and *padNumber* is the specific pad number of the pad to be used. The specification of the package pin number or pad number is optional.

Sample Critical Placement File

Here is an sample critical placement file which includes net weights, as well as fixed placement of macros and pads.

```
# A sample critical placement file
#
$Design testdouble
$PartNumber 0333
$Array vgt01600
$netlist [fns]testdouble
$placements
P OUTPUT_SIGNAL  #19
P u19.pad #20
I u24 4 20
I u25 6 1
W signal_name 2
W another_signal_name 2
W one_more_signal_name 3
$end
```

Your Final Bonding Diagram

The final bonding diagram must be included with your design package. You may wish to review the guidelines for preparing this diagram in Chapter 3. Submit a bonding diagram for both the prototype package and the production package if they are different.

Creating a Pad Placement File

Determine Pinouts

Before submitting a pad placement file, determine the final pinout for your design. It is important to ensure that a bonding diagram can be made for your selected pinout before finalizing your design specification. It is not always possible to bond to any arbitrary pin ordering for every package/VGT-base combination. These factors require care in determining pad placement for certain desired pinouts:

- VGT base array die size vs. package cavity size
- Locations of fixed VSSI and VSSO pins
- Power pad requirements for minimizing I/O switching noise
- Preferred package power and ground pin locations

Filling Out the Pad Placement File

After generating your bonding diagram, fill out the Pad Placement File. This file is used to place and route your circuit, and it is also used by the test generation software. The Pad Placement File is a text file on your workstation. To enter your pad placements into the file, use your system text editor to open the file and mark your selections.

Enter Package Pin Numbers

The first step is to enter the package pin numbers into the file. Using your bonding diagram and the footprint for the base you have chosen, locate the die pad number for each of the package pins connected. Opposite the die pad number in the Pad Placement File form, enter the package pin number in the appropriate row and column location. This is done for all of

the package pins that are connected to the die. Omit any unused or NC pins.

If the package pin numbers for the prototype package are different than the pin numbers for the production package, submit two Pad Placement Files, one for each package type. This won't affect the placement and routing of the circuit but is necessary for proper generation of the test programs.

Identify Pin Function

Next, identify the function of each pin. In the FUNCTION section of the Pad Placement File, enter an X under the appropriate column. For example, enter an X under the TTL column for TTL input buffers, an X under the PU + PD RES column if the input has an input pull-up or pull-down resistor, and so on.

Enter Signal And Macro Names

Finally, enter the signal name and macro name for each of the pins. For any power pins which do not use the dedicated VSSO or VSSI pins, enter the appropriate power macro name from the macro library.

The Pad Placement File is included with the rest of the files that are written on magnetic media when the design is transferred to a VLSI Design Center for placement and routing.

Customer Marking Form

The Customer Marking form provides VLSI with the marking information for your design. The marking information becomes part of the VLSI marking spec, which includes a VLSI logo, date code, and lot number.

Contact your VLSI Design Center if your marking requirements are not covered on this form.

Pre-Route Design Review and Signoff

When the design is finished and all of the pre-route requirements are completed, an engineering sign-off review is held at the VLSI Design Center. During this review, all of the items listed in the **Initial Signoff Review Checklist** are reviewed and handed over to a VLSI engineer. It is important that all of the items required on the checklist are complete, and that printed copies of them are available for the review.

After the checklist is reviewed and signed off, the design is transferred to the VLSI Design Center for placement and routing of the circuit.

CHAPTER 6

POST-ROUTE SIMULATION/SIGNOFF

Introduction

After you submit your design package for pre-route design review, VLSI places and routes it, then returns a data file containing actual interconnect capacitances. Use this information to reverify and resimulate your chip. If the chip performs correctly, you and a VLSI Technology design engineer sign the **Final Timing Signoff Review Checklist**, releasing the design for prototype fabrication. This chapter describes the post-route steps in detail.

How to Simulate After Layout

The data file containing actual interconnect capacitance is named *design*.PST. The *design*.PST file contains VTIsim capacitance statements for each node in your design that was routed. This file is loaded into the simulator after the netlist is loaded.

To back-annotate the netlist in the simulator, type these commands:

```
VTIsim> load [hns]design
VTIsim> load design.pst
```

where `design` is the name of your top-level netlist.

Final Timing Signoff Review

The Final Timing Signoff Review is very important, as it is the go-ahead for mask generation from your design. When you are completely satisfied that all simulations show your design to be correct, fill out the **Final Timing Signoff Review Checklist**. VLSI Technology will then generate tests and build and test prototypes for your approval.

APPENDIX A

VGT PACKAGE AVAILABILITY

Table 9 lists the packages available for the VGT10 series gate array bases, and Table 10 lists the packages available for VGT100, as of the date of this publication.

These packaging tables are subject to continual change as customers require new package pin counts and families. Please consult with VLSI marketing if a package you need is not listed. Packages not listed here may become available in the future.

Table 9. VGT10 Series Package Availability

PKG TYPE	#PINS	Θ_{ja} [1]	VGT10 ARRAY BASE					
			1600	2400	3700	6400	8400	10600
Plastic DIP[2]	16	65/120						
	20	55/110						
	24	50/110	●	●	●	●		
	28	45/105	●	●	●	●		
	40	45/100	●	●	●	●	●	●
	48	45/100		●	●	●	●	●
PDIP	64	45/100				●	●	
Ceramic Sidebrazed DIP	16	70						
	20	65						
	24	55	●	●	●	●	●	●
	28	50	●	●	●	●	●	
	40	40	●	●	●	●	●	●
	48	40	●	●	●	●	●	●
S/B	64	40	●			●	●	
Leaded Ceramic Chip Carrier JEDEC C	20							
	28	40		●				
	44	40		●	●	●	●	
	52	40			●	●	●	
	68	40	●	●	●	●	●	●
LDCCC	84	40	●	●	●	●	●	●
Leadless Ceramic Chip Carrier JEDEC B	20							
	28	40						
	44	40						
	52							
	68	40	●	●	●	●	●	●
LLCCC	84	40	●	●	●	●	●	●
Plastic Leaded Chip Carrier	20	50	●	●				
	28	45	●	●	●			
	44	45	●	●	●	●	●	●
	52	45	●		●	●	●	
	68	45		●	●	●	●	●
PLCC	84	45				●	●	●

[1] Approximate value (°C/W) in still air

[2] Θ_{ja} values given as Copper/AL42

Table 9. VGT10 Series Package Availability (Cont'd)

PKG TYPE	#PINS	Θ_{ja} [1]	VGT10 ARRAY BASE					
			1600	2400	3700	6400	8400	10600
Ceramic Pin Grid Array	68	40		●	●	●	●	●
	84	40		●	●	●	●	●
	100	35				●	●	●
	120	35				●	●	●
	132	35					●	●
	144	35						●
	149	35						●
	172	35						
CPGA	180	35						
	224	35						
Plastic Pin Grid Array	68							
	84	48	●	●	●	●		
	100	64	●			●	●	●
	120	29				●	●	●
	132							
	144							
	149	24						
	172							
PPGA	180							
	224							

[1] Approximate value (°C/W) in still air

Table 10. VGT100 Series Package Availability

PKG TYPE	#PINS	Θ_{ja} [1]	VGT100 ARRAY BASE						
			12100	16000	20500	27000	39700	52800	66500
Leaded Ceramic Chip Carrier JEDEC C	20								
	28	40							
	44	40							
	52	40		●					
	68	40	●	●	●				
LDCCC	84	40	●	●	●	●			
Leadless Ceramic Chip Carrier JEDEC B	20								
	28	40							
	44	40							
	52								
	68	40	●	●	●				
LLCCC	84	40	●	●	●				
Plastic Leaded Chip Carrier	20	50							
	28	45							
	44	45							
	52	45		●					
	68	45	●	●	●				
PLCC	84	45	●	●	●	●			
Ceramic Pin Grid Array	68	40	●	●	●				
	84	40	●	●	●				
	100	35	●	●	●				
	120	35	●	●	●				
	132	35	●	●	●				
	144	35		●	●				
	149	35			●				
	172	35		●	●				
CPGA	180	35		●	●	●	●	●	
	224	35		●	●				
Plastic Pin Grid Array	68								
	84	48							
	100	64	●	●	●	●			
	120	29	●	●	●	●			
	132								
	144								
	149	24			●	●			
	172								
	180								
PPGA	224								

[1] Approximate value (°C/W) in still air

APPENDIX B

SCREEN REPORTS AND ERRORS

Screen Report

VLSI's screener performs a check of your design and outputs a report detailing specific types of errors. This report helps to find errors easily and quickly, while assuring some correctness of design. A sample of the VTIscreen output can be found on page 75.

NOTE: This appendix describes **Version 1.1** of the VTIscreen tool. If you have a later version, please contact your Design Center for assistance, if necessary.

Cell Utilization Summary

VTIscreen first prints a cell usage summary, which breaks down the total cell count into individual cell usage for each macro.

The cell utilization summary gives the following information:

- Number of Input Pads
- Number of Output Pads
- Number of Bidirectional Pads

- Number of VDD
- Number of VSS
- Number of VDD Pads
- Number of VSS Pads
- Number of VDD Core
- Number of VSS Core
- Number of Available Pads
- Number of Pads Used
- Percent of Pads Used
- Number of Available Placement Sites
- Number of Placement Sites Used
- Site Utilization Percentage
- Number of Gate Equivalents Used
- Number of Non-Primitive Blocks Used

Design Statistics

The design statistic report gives the following information about your design:

- Number of Nets. A net is another name for signal.
- Number of FromTos. A FromTo is defined as the connection between two pins.

- Average Number of Pins per Net.
- Maximum Number of Pins per Net.

Design Complexity Summary

The Design Complexity summary lists the different VLSI macros used in your design, the number of instances of each macro, the number of gate equivalents per cell and the total gate equivalents.

Screen Errors

The screener performs an important function in the design process beyond that of generating cell statistics. It also performs a preliminary check of the design to ensure smooth netlisting and transfer of the design to VLSI. The following is a list of the checks and associated error messages that the screener produces.

Warning messages are to alert you to potential problems. Please check each message against your schematic to decide whether a real problem exists.

Error messages are to alert you that a problem exists and that corrective action on your part is necessary before you proceed to simulate or place and route your design.

`WARNING - The following signals do not drive any macros:`

A list of signal names follows the message. For each signal, the net *signal name* is connected to one or more output ports but not connected to an input port.

`WARNING - The following primary input signals can not connect to other macros:`

A list of signal names follows the message. For each signal, this message indicates that another macro has been directly connected to an input pad. This usually happens when an internal signal has the same name as a primary input signal.

`WARNING - The following primary output signals can not connect to other macros:`

A list of signal names follows the message. For each signal, another macro has been directly connected to an output pad. This usually happens when an internal signal has the same name as a primary output signal.

`ERROR - The following macros can not be used with the` *product name* `product:`

A list of macros follows the message. These macros belong to a product other than the one being screened. Check to be sure the correct `[lpd]` cells are on the search path.

`ERROR - The following signals are not driven by any macros:`

A list of signal names follows the message. For each signal, the net *signal name* connects to one or more input ports but does not originate from an output port, therefore the net *signal name* will never carry a signal.

ERROR - Signal *signal name* originates from the following locations:

A table giving the macro name, instance name and port name follows this message. The net *signal name* that is connected to the output port *port name* is also connected to the output port of an instance of a different macro type. The output ports of two different macro types cannot be tied together.

ERROR - The following primary input signals should go thru connectors:

A list of signal names follows the message. For each signal, the net *signal name* that connects to an input buffer's input port must originate from a connector.

ERROR - The following primary output signals should go thru connectors:

A list of signal names follows the message. For each signal, the net *signal name* that connects to an output buffer's output port must originate from a connector.

ERROR - The following primary input signals should go thru input buffers:

A list of signal names follows the message. For each signal, the net *signal name* connects directly from a connector to an input port of an internal macro such as ND03D1 or DFPTNB. A net from an input connector can only be connected to an input buffer's input port.

ERROR - The following primary output signals should go thru output buffers:

A list of signal names follows the message. For each signal, the net *signal name* connects directly from a connector to an output port of an internal macro such as ND03D1 or DFPTNB. A net from an output connector can only be connected to an output buffer's output port.

WARNING - The following signals should connect to only one bus repeater cell:

A list of signal names follows the message. These signals should only connect to one bus repeater.

WARNING - The following signals should not connect to a bus repeater cell:

A list of signal names follows the message. These signals are not three-state signals and therefore should not connect to a bus repeater cell.

ERROR - Each of the following output pads' ngate/pgate pins must connect to a single pad driver cell's ngate/pgate pins respectively:

A table giving the macro name and instance name follows this message. The output pad *macro name's* ngate and pgate ports must connect to a pad driver's ngate and pgate ports respectively. The output pad should connect to only one pad driver.

`ERROR - The following pads should connect to a level shifter:`

A table giving the macro name, instance name, port name and signal name follows this message. The signal name(s) in the table should connect the pad macro with a level shifter.

Macro Usage Table

The macro usage table lists the different VLSI macros used in your design, the number of instances of each macro used and each flattened instance name of each macro used. This table is printed only in the disk file, `[scr]`.

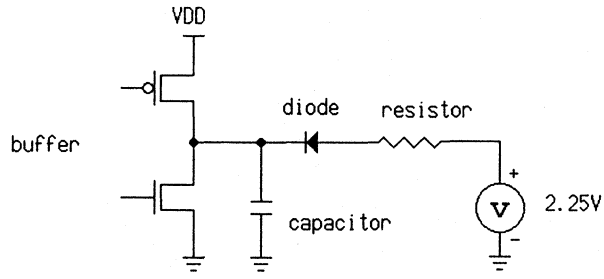
APPENDIX C

OUTPUT BUFFERS

The VGT output buffers are characterized with special load circuitry. This load circuit is intended to duplicate actual operating conditions and is used for both TTL and CMOS output buffer characterization.

For output buffers associated with a TTL input buffer, outputs are characterized to TTL levels 0.8V and 2.0V. In addition to the capacitive load, the traditional diode-resistor network is hung on the pad to simulate a TTL load. A resistor of the appropriate size is used to ensure that the buffer sinks or sources the specified amount of current.

For output buffers associated with a CMOS input buffer, outputs are characterized to CMOS levels 0.85V and 3.65V. In addition, a diode-resistor-voltage supply network is hung on the pad, as shown:



The resistor and voltage network causes the output buffers to sink or source the appropriate amount of current during AC performance simulations. For sourcing current, the diode in the above figure points the opposite direction.

The addition of the resistor slows down the AC performance from that of a purely capacitive load.

INDEX

W

VLSI TECHNOLOGY, INC.

VLSI Technology, Inc.
ASIC Division
1109 McKay Drive
San Jose, CA 95131
408-434-3100
TLX: 278807
FAX: 408-263-2511