Mentor Graphics and Gould AMI
Design Manual

TABLE OF CONTENTS

TABLE OF CONTENTS [continued]

TABLE OF CONTENTS [continued]

Section 7 APPENDICES [continued]

LIST OF FIGURES

1. INTRODUCTION

This manual provides a tutorial for designing GOULD AMI gate arrays using the Mentor Graphics engineering work station. The work station offers different levels of support in the gate array design cycle, allowing you to choose the amount of control that you have over your design. This manual addresses every level of support from schematic capture through physical layout of the design.

This manual explains the design cycle and the various Mentor Graphics tools which are used to complete a design. This manual also explains how you may alter the design stream and tailor it to your needs. An example circuit serves to demonstrate the design stream.

This manual does not describe all of the capabilities of the Mentor Graphics tools such as NETED, SYMED, etc. Some familiarity with these applications is assumed. Separate reference manuals for each of these tools are available. This manual will refer you to pertinent manuals during the course of the design cycle explanation. The final section of this manual lists the reference manuals that are useful for learning more about the design tools. It is suggested that you take the Mentor Graphics IDEA STATION course before using the GOULD AMI design package.

The first 5 sections describe the design cycle and the tools that are used for the design. Section 6 shows you how to run MAGIC and gives the results from each design step for comparison.

2. GATE ARRAY DESIGN STREAM

2.1 How a Gate Array is Designed

The process of designing a gate array can be broken into 6 basic steps -

1. Schematic Capture
2. Simulation
3. Package assignment
4. Physical layout
5. Back annotation and resimulation
6. Test vector generation

The flowchart in figure 2-1 shows how these steps fit together and the data generated at each step.

2.1.1 Schematic Capture

The first step in the design cycle is to define the gate array logic. This is done by selecting the symbols representing the logical functions available in the GOULD AMI gate array family and then connecting the pins together with nets. The symbols contain the information necessary to drive simulation and physical layout. To facilitate the testing and debugging of logic, the design is usually implemented in a hierarchical manner.

2.1.2 Simulation

Simulation uses the connectivity data generated from the schematics entered in step 1 to model the behavior of the design. Simulation includes timing information as well as the functional logical simulation. Simulation recognizes the delay time required for changing the states of the output pins on each macro, and uses this information to model the behavior of the circuit. Delay time includes the internal gate delay and the delay due to the capacitive loading of gate inputs and predicted wiring. The delay of a signal also varies with voltage, temperature and fabrication process. These factors can be controlled when running the simulation.

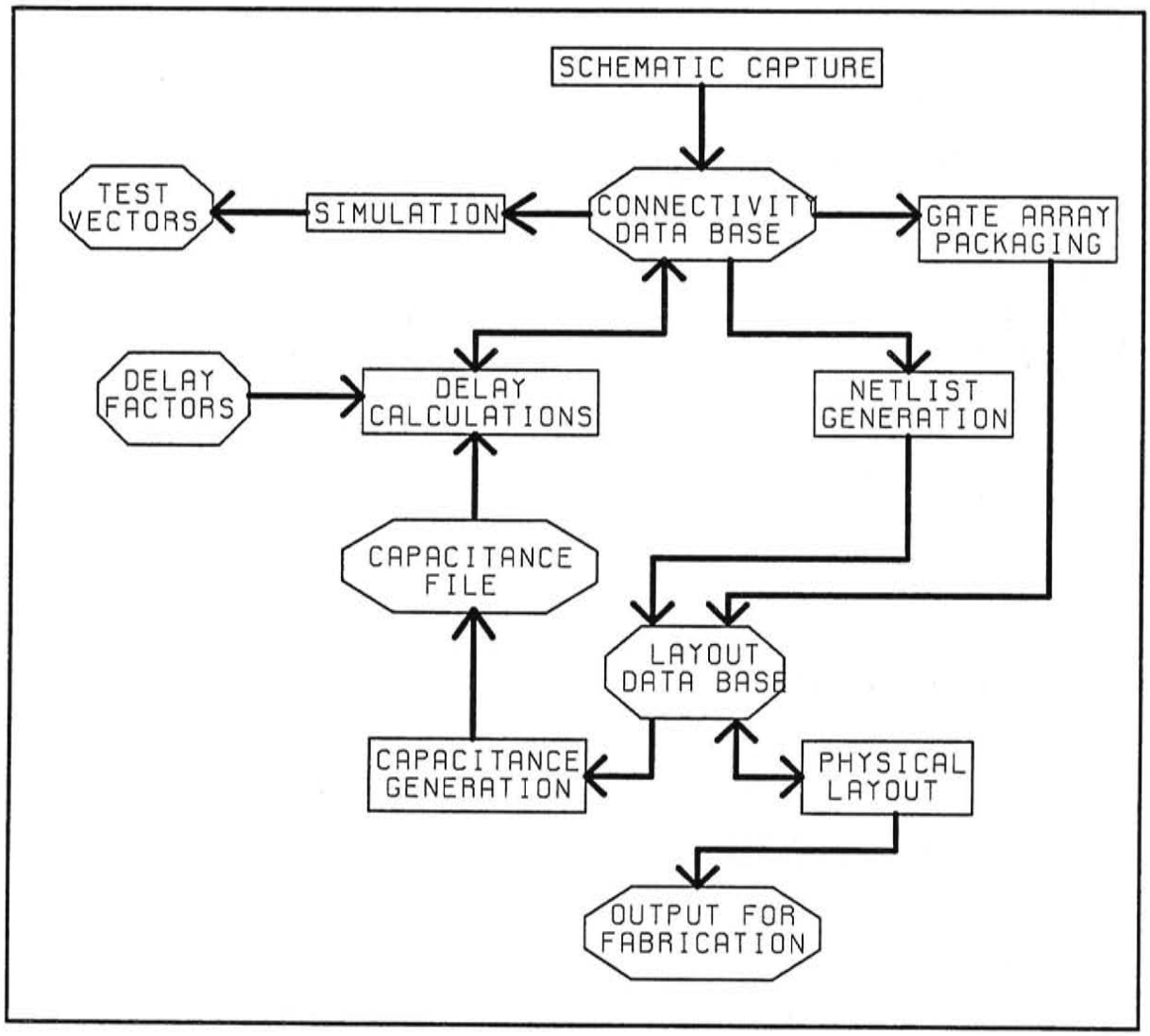


Figure 2-1. GATE ARRAY DESIGN FLOW

2.1.3 Package Assignment

There are a large number of package types that can be used for implementing a gate array. The type of package that you choose depends on the number of output pins on the circuit and the conditions under which the gate array must function. The uncommitted gate array (the gate array before any logic has been assigned) has a fixed number of locations for input and output signals to the chip. Not every location is acceptable for any given design. When a subset of the I/O locations are used, there are very specific rules

concerning the use of the locations. The package definition step guides you through the choice of the I/O locations. These assignments must be made before physical layout, because these decisions affect how the signal paths will be routed to the I/O locations.

2.1.4 Physical Layout

The physical layout step uses the connectivity and package assignment information to implement the design. Starting with an uncommitted chip, the GOULD AMI macros used in the design are optimally placed. Placement defines the functions of transistors on the array by 'placing' predefined patterns of connectivity over the transistors. These patterns of connectivity are called macros and are used to create a variety of logical functions, ranging from inverters to flip-flops and multiplexers. When the transistors have been programmed into a logical function, signal routing is added to connect the logical functions. Both placement and routing can be performed automatically or through interactive graphics.

2.1.5 Back Annotation and Simulation

Once the physical implementation of the design has been determined, a more accurate model of signal delays can be generated. The length of all signal wires is known and the capacitive load of each net can be calculated. Back annotation extracts the capacitance values and updates the simulation data to include these delays. Simulation can now be run with a higher confidence since the true delays are being modeled.

2.1.6 Test Vector Generation

It is necessary to provide a set of input stimuli to test the circuit when it has been produced. These patterns are generated from simulation values, but are formatted to create the commands for the chip test hardware.

2.2 GATE STATION Support of Gate Array Design

GATE STATION can support the entire gate array design process described above. It is also possible to use a subset of GATE STATION and the GOULD AMI libraries to support different levels of the design process. The following list shows the 3 levels of support that are offered :

- Schematic Capture - Design Stream A (Schematic Interface)
- Schematic Capture, Simulation and Back Annotation - Design Stream B (Validated Netlist Interface)
- Schematic Capture, Simulation, Physical Layout and Back Annotation - Design Stream C (Data Base Interface)

Figure 2.2 shows the the grouping of the design flow into design stream support.

2.2.1 Design Stream A (Schematic Interface)

This is the minimum level of support. GOULD AMI provides the symbol library for schematic capture. The library does not support simulation or layout. The Mentor Graphics CAPTURE STATION is the minimum configuration to support this type of design. You would give GOULD AMI a netlist built from the schematic describing the connectivity of the circuit. GOULD AMI would perform the simulation and layout at the factory. Design stream A is highlighted in Figure 2-2.

2.2.2 Design Stream B (Validated Netlist Interface)

This stream provides extensive simulation capability in addition to schematic capture. The Mentor Graphics IDEA STATION supports this design stream. GOULD AMI provides a logic library which contains symbols with functional and timing simulation data. You receive data that allows you to test the signal delays based on a variety of different parameters. The signal delays for a design can vary with operating temperature and voltage, fabrication process, and capacitive loading on the signal driver. The capacitive loading is broken into the fanout loading due to input gates and predicted wiring in the net. The GOULD AMI library contains the data necessary to simulate these conditions. The signal delays are automatically calculated when any of these parameters are changed and the simulation files updated.

In addition to simulation, package definition capability is offered. Package definition is a series of programs that help you create a correct bonding diagram based on the package type that you have selected. The bonding diagram is the guide for GOULD AMI that shows the connections between the pads on the chip and the pins on the package. You may, for example, use only a subset of the possible I/O pads on the uncommitted chip. Package definition will ensure that you use the appropriate locations on the chip for this subset to create the correct bonding diagram. This information is used in the layout of the design.

After you are satisfied with the design, it is sent to GOULD AMI for layout. When they complete the layout, a capacitance file is generated and sent back for resimulation. The capacitance values are converted into delay values on your system and the resimulation is performed with these accurate delay values.

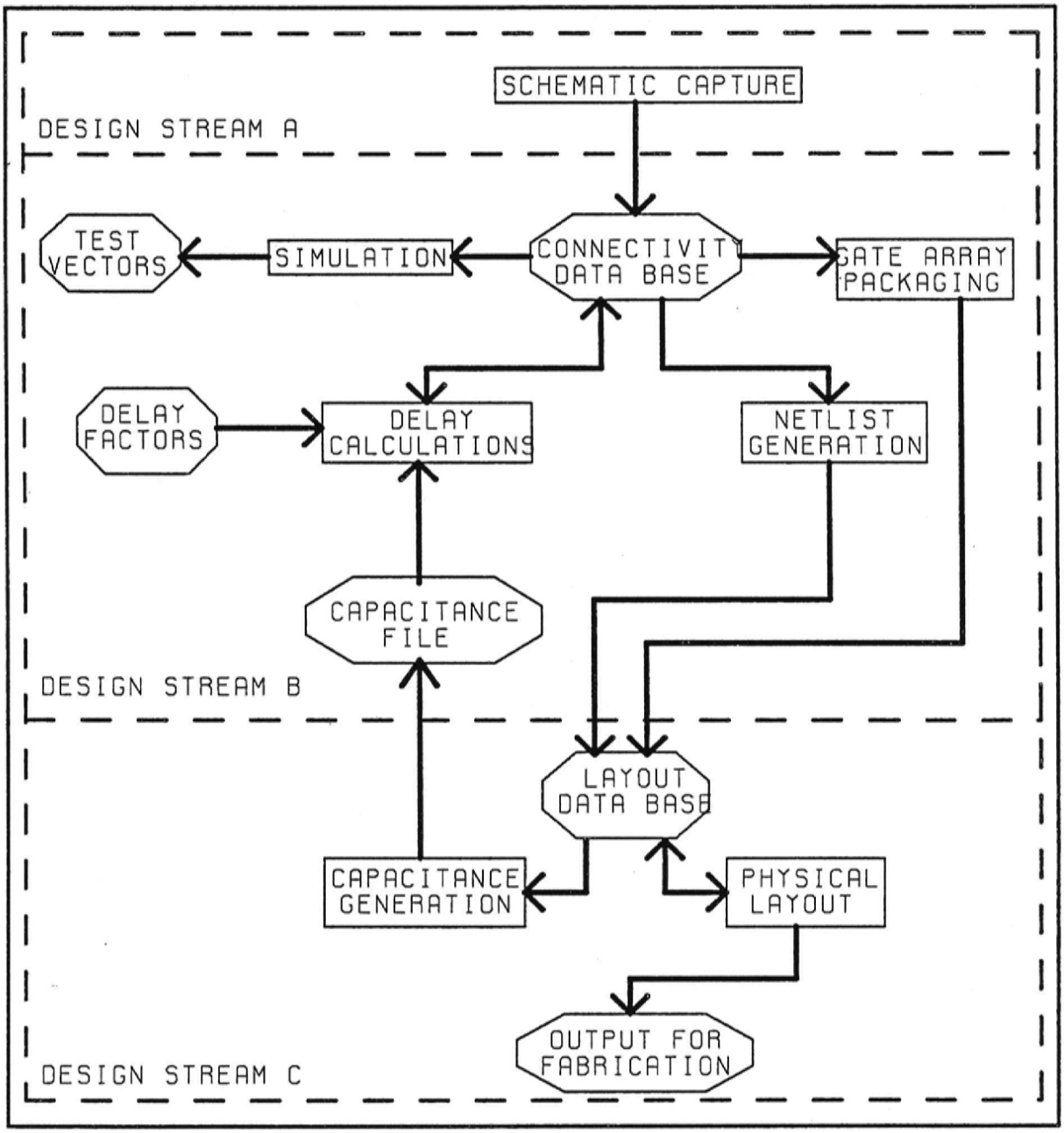


Figure 2-2. Mentor Design Stream Support

You would return to GOULD AMI a netlist for the physical layout and a set of test patterns for testing the final chip. Design stream B is highlighted in Figure 2-2.

2.2.3 Design Stream C (Date Base Interface)

Design stream C offers all the capability of B plus physical layout. The Mentor Graphics GATE STATION supports this design stream. In addition to the simulation data, you receive the physical data describing the uncommitted chip and the macros. When the layout is complete you can back-annotate the true net delays due to wire length for resimulation.

The test vectors, netlist, and physical layout data are returned to GOULD AMI and the circuit would be ready for fabrication. This design stream gives you complete control over the design process. Stream C is highlighted in Figure 2-2.

3. MENTOR GRAPHICS TOOLS FOR GATE ARRAY DESIGN

Section 2 discussed gate array design and how the design is supported. Section 3.1 provides more detail about the data and programs you will need at each design stream level. Section 3.2 is a description of the directory organization for the GOULD AMI design package.

While this section describes all the data and programs that you will need, it does not describe how to run the programs. Section 4 addresses design methodology and section 7 lists relevant reference manuals.

3.1 Design stream descriptions

The GOULD AMI design includes data, programs and documentation. The amount of information you receive will vary depending on the level of design support that you choose. This data will be used as input to the Mentor Graphics tools. You will send back to GOULD AMI the data necessary to complete the design. The steps performed at GOULD AMI to complete the design will vary depending on the design stream option that you use. The following lists describe each design stream in detail.

DESIGN STREAM A - SCHEMATIC CAPTURE (Schematic Interface)

WHAT GOULD AMI SENDS TO YOU

1. Parts library containing a symbol for each macro. The symbols do not have sheets associated with them for simulation, but the symbols will contain the following properties for netlist generation :

   a. COMP name corresponding to the macro names in the GOULD AMI library

   b. PIN name corresponding to the pin names in the GOULD AMI library

2. The BOLT netlist generator to generate the netlist for simulation and layout at GOULD AMI.

MENTOR GRAPHICS TOOLS YOU WILL NEED

1. NETED for schematic capture

2. SYMED for hierarchical design

3. EXPAND for creating the flattened design to generate the netlist

WHAT YOU WILL SEND TO GOULD AMI

1. BOLT netlist containing all the connectivity data for the design.

2. Logic simulation table of stimuli and reponses

3. Package data

4. Device Specification

WHAT STEPS ARE PERFORMED AT GOULD AMI

1. Simulate the design

2. Determine and specify layout constraints (I/O preplacement, critical nets)

3. Perform physical layout

4. Re-simulate using physical layout results for calculating delays

5. Determine test vectors

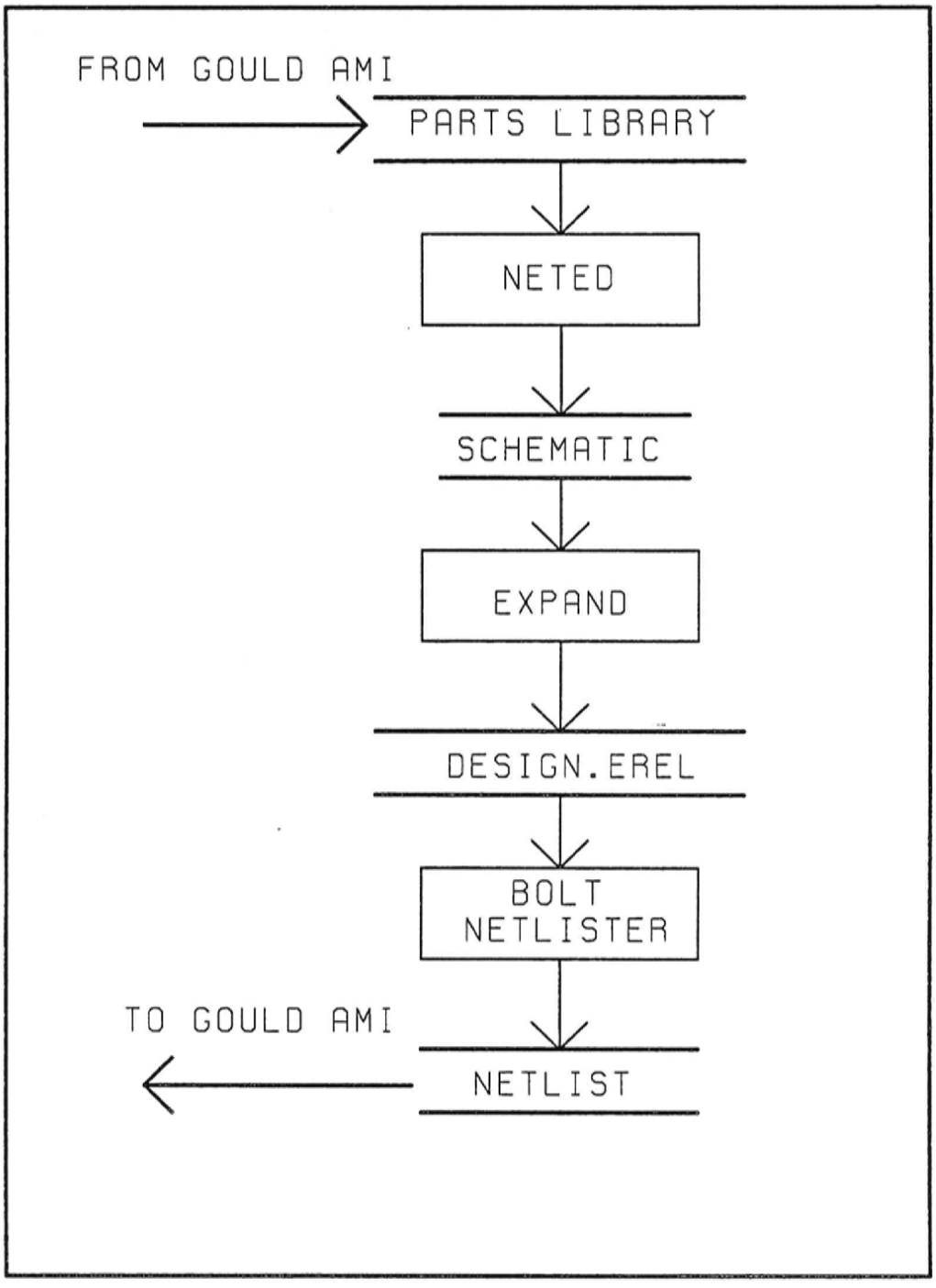


Figure 3-1. Schematic Capture

DESIGN STREAM B - SCHEMATIC CAPTURE AND SIMULATION
(Validated Netlist Interface)

WHAT GOULD AMI SENDS TO YOU

1. All the data needed for Design Stream A
2. Sheets describing the functional simulation for each macro
3. Timing information for each macro
4. Technology file describing the timing information for
   a. different operating temperatures
   b. different operating voltages
   c. process variation
   d. predicted wiring delays based on fanout
5. AMI_ADD_DELAY program for modifying the simulation timing based on the factors specified in the technology file and GOULD AMI delay equations
6. Package Definition program for specifying the gate array packaging arrangement
7. Description of each package configuration for use in package definition

MENTOR GRAPHICS TOOLS YOU WILL NEED

1. NETED for schematic capture
2. SYMED for doing hierarchical design
3. EXPAND for creating the flattened design to generate the netlist
4. SIM for simulation

WHAT YOU WILL SEND TO GOULD AMI

1. BOLT netlist
2. Test vectors
3. I/O placement for the selected package type
4. Device specifications

WHAT STEPS ARE PERFORMED AT GOULD AMI

1. Perform physical layout
2. Generate the capacitance file from the layout

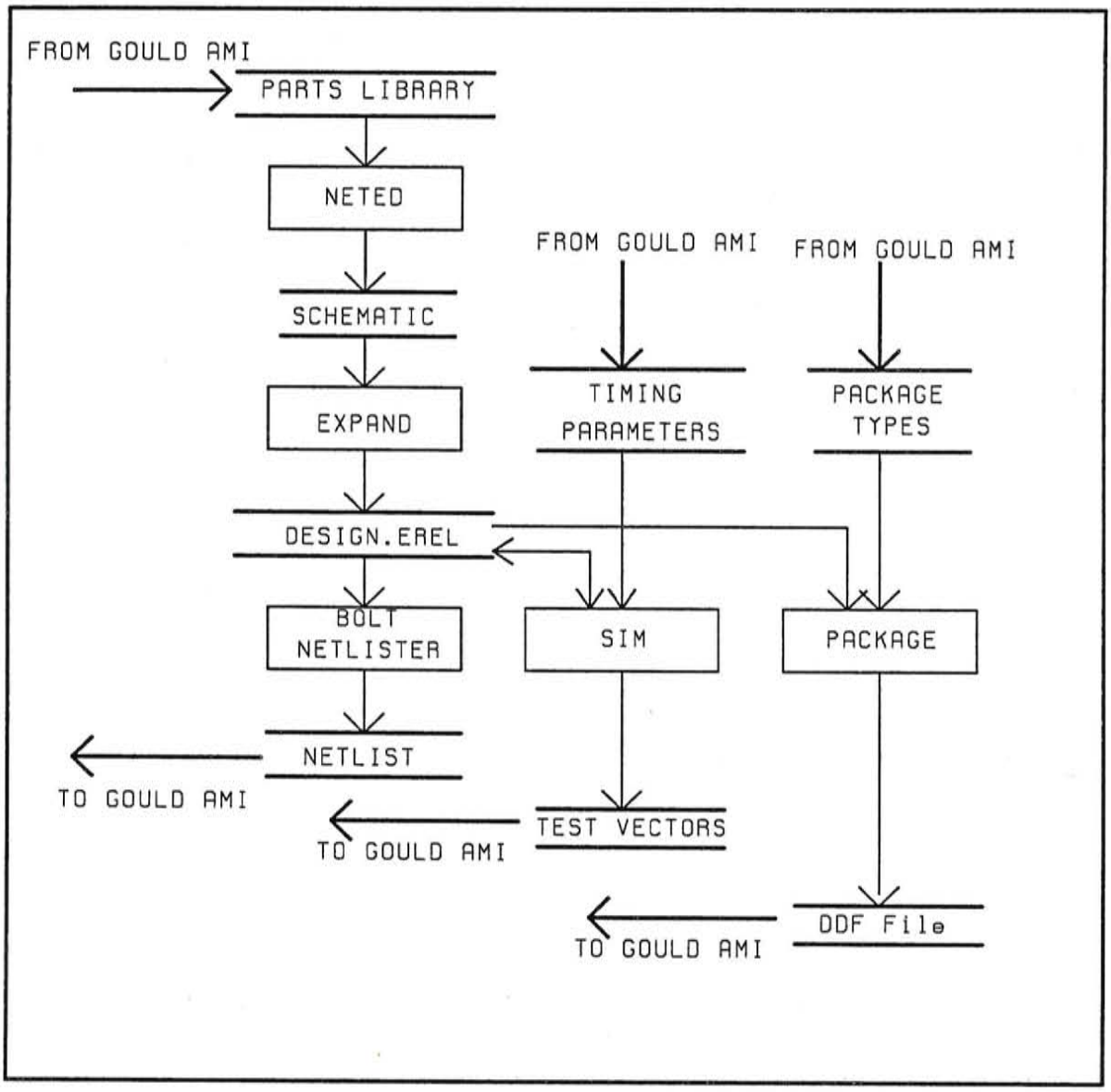


Figure 3-2. Schematic Capture, Simulation and Package Constraints

DESIGN STREAM C - SCHEMATIC CAPTURE, SIMULATION, BACK ANNOTATION AND PHYSICAL LAYOUT (Data Base Interface)

WHAT GOULD AMI SENDS TO YOU

1. All the data and programs in design streams A and B
2. Data bases for physical layout

MENTOR GRAPHICS TOOLS YOU WILL NEED

1. NETED for schematic capture
2. SYMED for hierarchical design
3. EXPAND for creating the flattened design to generate the netlist
4. SIM for simulation
5. CADISYS for doing physical layout

WHAT YOU WILL SEND TO GOULD AMI

1. BOLT netlist
2. Test vectors
3. DDF file describing the macro placements and routing data
4. Device specification

WHAT STEPS ARE PERFORMED AT GOULD AMI

Build the chip

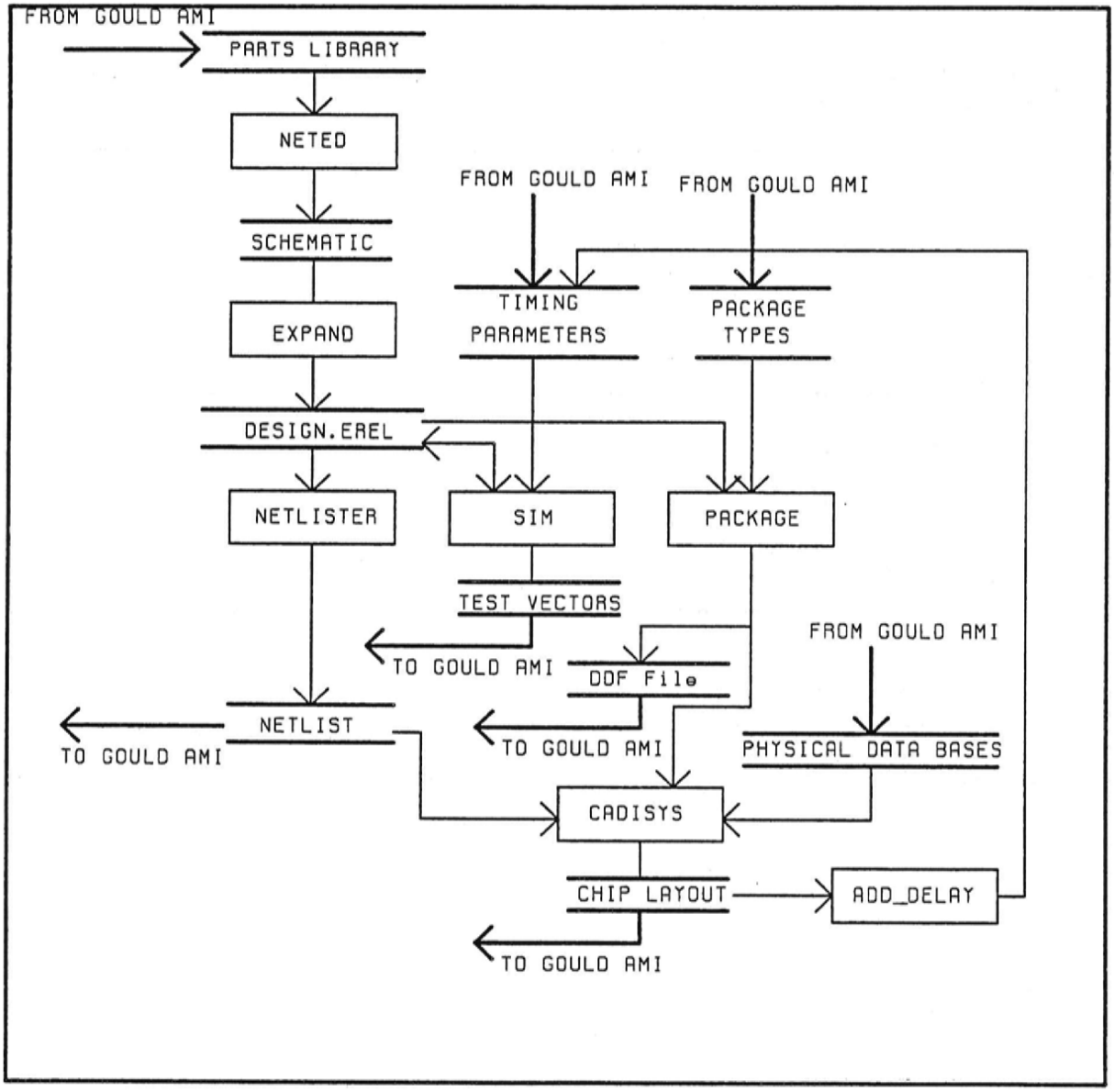


Figure 3-3

3.2 Directory Structure

Understanding the directory structure of the GOULD AMI design kit is valuable for doing designs. The naming conventions for the directory structure will be used to describe program execution.

The directory structure is designed to be flexible enough to contain new technologies as they become available and to be expandable as new capabilities are added to the kit. It is also designed to be modular to allow you to delete portions as they become outdated.

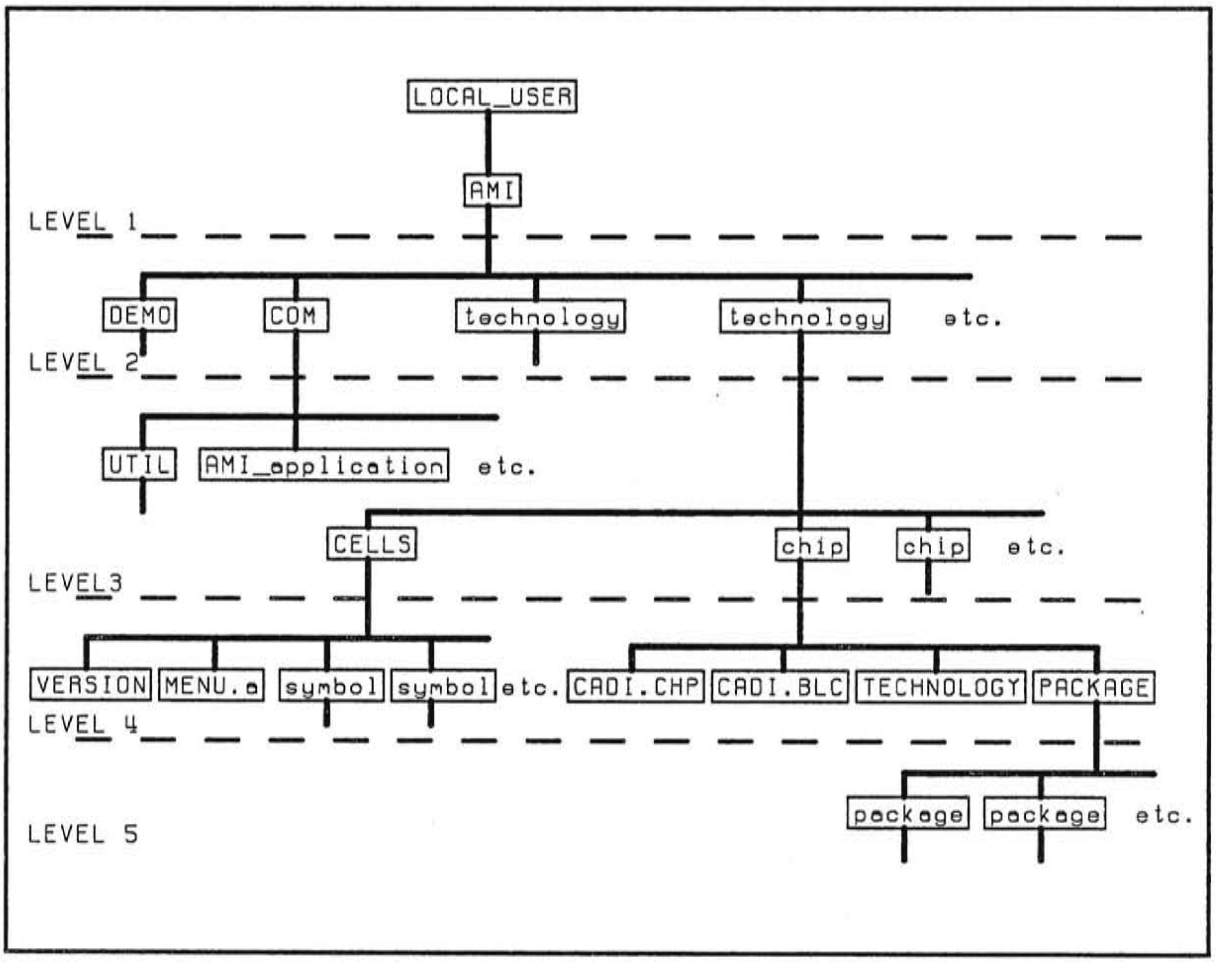


Figure 3-4. Directory Structure

Generic names are used to explain the directory structure. A final diagram will contain the specific GOULD AMI names. In these examples, a simple naming convention is used for directory names: a capitalized name is the name you would see if you looked into the directory. A lower case name in the example will be replaced by a more descriptive name in the actual design kit.

Level 1

- ~AMI

  As figure 3-4 shows, the vendor name appears at the top level. In this case it is AMI. This normally resides in the LOCAL_USER directory with a pointer to this name in the USER directory. In other words a 'user' should be created for this directory. (For more information on the CREATE_USER command, type HELP CREATE_USER or review the IDEA System Manager's Manual chapter 7.) This will allow all users on the network to access this directory with the command WD ~AMI. CREATE_USER is performed automatically with the AMI_INSTALL command described in section 5 of this manual.

  Level 2

- ~AMI/DEMO

  Under the directory DEMO are the demo circuits for each technology. The actual demo circuit is discussed later in this manual. This directory is for training purposes and can be deleted without affecting designs.

- ~AMI/COM

  The COM directory contains the programs and command files for the GOULD AMI design kit. This includes the command files for AMI_ADD_DELAY, AMI_SIMOUT and so on. This directory should be included in your CSR list (command search rules - type HELP CSR in any AEGIS window for additional information). It is also possible include this CSR in your startup file automatically. Section 5 describes this process.

- ~AMI/technology

  At the second level you will find the directories containing the different types of technologies. For example, you may have a double-level 2 micron gate array library and a two-level 3 micron standard cell library. There may be other technologies at this level as well. Each of these technology directories contains the data specific to that particular technology. This would include the logic symbols, simulation data and so on.

  Level 3

- ~AMI/COM/UTIL

  This directory contains specific GOULD AMI programs and utility command files.

- ~AMI/COM/AMI_application

  These are the command files for invoking applications with AMI specific start up sequences. These are automatically activated if ~AMI/COM is included in the CSR list.

- ~AMI/technology/CELLS

  This directory contains all the symbols for schematic capture. These symbols contain the information used to generate netlists. They also contain the information for running a functional and "baseline" timing simulation. A baseline simulation only takes into account the gate delay in the macro. The information for wiring delay and other operating parameters reside in another location. The symbols can be used to build any size chip in the technology family. The timing parameters will vary based on the size of the uncommitted chip.

- ~AMI/technology/chip

  This directory contains all the data specific to a particular uncommitted chip. For example you may have a family of uncommitted chips with gate counts of 1K, 2K, 3K and 4K. Each of these chips would have a separate directory.

  Level 4

- ~AMI/technology/CELLS/VERSION

  This file contains the version number of this technology and the date the technology was created. This will be used to time stamp your design.

- ~AMI/technology/cells/MENU.application

  There may be several menu files for the different application programs. The menus will be invoked at the beginning of the application. These are ASCII files and can be viewed

- ~AMI/technology/CELLS/symbol

  These are the individual components for schematic capture.

- ~AMI/technology/chip/CADI.CHP

  This is the physical data base describing the uncommitted gate array for this chip size.

- ~AMI/technology/chip/CADI.BLC

  This is the physical data base describing the macro library.

- ~AMI/technology/chip/TECHNOLOGY

  This is an ASCII file containing the information for running simulations based on chip dependent parameters. Timing, voltage, temperature and fanout parameters are included in this file.

- ~AMI/technology/chip/PACKAGE

  This directory contains the sheets for package assignment. Each different package type will be represented by a sheet in this directory

  Level 5

- ~AMI/technology/chip/PACKAGE/package

  The sheet for the package type to be used in the package definition program. Each package type is created using NETED.

Figure 3-5 shows an example using the names specific to GOULD AMI. In this example the only technology that is represented is the GA_M3D which stands for gate array 3 micron double metal. There are two chip types in the example, the AMI1K and AMI2K.

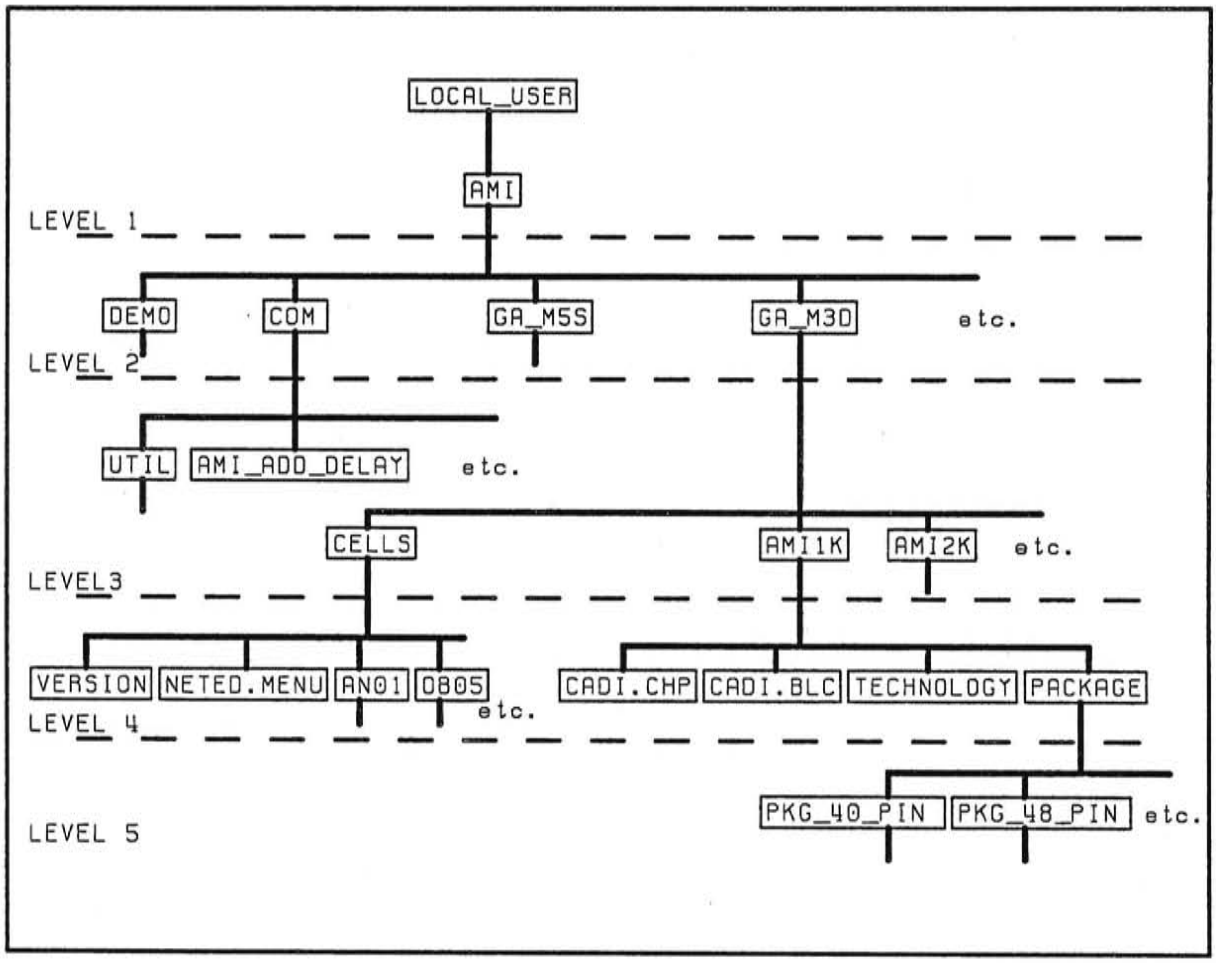


Figure 3-5. GOULD AMI Directory Structure

4. DESIGN METHODOLOGY

This section provides more detail about the data and programs in your design kit. There are specific features which can simplify gate array design, and these features are highlighted in this section.

4.1 Schematic Capture

4.1.1 About the symbols

For each macro in a technology there is a corresponding symbol. The symbols reside in the 'technology/CELLS' directory. Symbols can be activated for NETED from this directory. A file containing the NETED commands for creating a menu is located in the file 'technology/CELLS/MENU.NETED'. When this file is executed in NETED, a menu will appear under the COMPONENT LIBRARIES menu allowing you to activate the GOULD AMI components for instantiation in NETED.

The symbol contains information for the programs that access the DESIGN.EREL file. This information is stored in the properties on the symbol. The following list shows each property name and a brief description of its function. When possible, a reference indicates where you can find more information about the property.

You should not use these properties in higher level symbols. This will cause a conflict with some of the design kit programs because they assume that these properties are associated with symbols representing macros.

- COMP

  This is the name of the macro in the physical data base that corresponds to this symbol. This is the property that links each logical symbol with its physical description in the physical data bases. When the design is expanded for layout it is expanded to the COMP level. For this reason it is important that the COMP property is not used on higher level symbols in the design. The physical layout programs expect to find a macro matching the COMP name.

- KRISE and KFALL

  These are factors used by ADD_DELAY for converting the loading capacitance into delay times. These properties are attached to the output pins of every symbol. For more information about this property, see the ADD_DELAY documentation at the end of this manual.

- LOAD

  This property is used by ADD_DELAY to compute the new delay due to fanout. LOAD is attached to every input pin and represents the capacitive loading added to the net by the pin. More information about LOAD can be found in the ADD_DELAY documentation at the end of this manual.

- MODEL

  The model name indicates to the SIM program which simulator primitive should be used for a given symbol. This property appears only on a primitive symbol or a symbol that uses a behavior language model. The SIM section of the IDEA SYSTEM REFERENCE MANUAL contains more information on this property.

- PHY_PIN

  This property is attached to a pin and its value indicates the name of the pin on the physical macro. This property links the pins on the symbol to the pins of the macro in the physical data bases. If this property is not attached to a pin, the PIN property is used to generate the pin name for the physical layout netlist.

- PIN

  This is the pin name. More information on the PIN property can be found in the SIM section of the IDEA SYSTEM REFERENCE MANUAL.

- PINTYPE

  This property indicates the direction of the pin to the simulator. More information on the PINTYPE property appears in the SIM section of the IDEA SYSTEM REFERENCE MANUAL.

- RISE and FALL

  These properties are delay times through the macro. If AMI_ADD_DELAY is not run on the design, these are the delay values that will be reflected in the simulation of the circuit. For more information, see the SIM section of the IDEA SYSTEM REFERENCE MANUAL.

- SC_IO

  This property is attached to the I/O macros of the library. The value of the property indicates the type of I/O macro. PKG_DEF uses

this information to assign I/O macros correctly.

4.1.2 Software Macros

Software macros are created by GOULD AMI to simplify circuit design. These hierarchical macros are represented by a single symbol in the library, but reference the other hardware macros in the library.

Any macro that represents a single physical macro in the GOULD AMI physical data bases is considered a hardware macro. When you use one of these macros you are using a function that is placed on the array as a single entity. This entity ranges from a single inverter to flip flop's and latches. Complete descriptions of all the hardware macros available for design are found in the 3μ Single and Double-Metal CMOS Gate Arrays Databook.

Software macros combine these hardware macros on a NETED sheet to specify a logical function. A symbol representing this function is then created and added to the library. When you instantiate this symbol into your design, you are simply adding another level of hierarchy. The functionality of the software macro has been tested at GOULD AMI. The information for layout that is generated from designs using software macros contains the references to the hardware macros in the physical data base. The layout software will optimize the layout of this macro. When back annotation is performed, the correct delay values will be inserted into the DESIGN.EREL file for each instance of a software macro.

You can create a library of software macros. Once you create a circuit that performs the desired function, you create a symbol to represent it. This symbol can then appear in any other design you have that uses the same GOULD AMI library. Software macros generated from the 3μ single level library cannot be used in a 2μ double level design. The simulation and the back annotation program will handle these hierarchical designs correctly so long as you do not use the properties described in section 4.1.1 on your symbol. These properties are reserved for the hardware macros.

4.1.3 NETED Features for Gate Array Design

Three features of NETED facilitate gate array design. The following paragraphs explain these features briefly. More detailed information concerning each of these features can be found in the IDEA SYSTEM USERS MANUAL and IDEA SYSTEM REFERENCE MANUAL.

- Hierarchical design

This is a feature of the entire Mentor Graphics system, but it must be considered when entering the schematic.

Symbols represent circuits. A given symbol may be repeated in your design, or it may be used only once. Representing a circuit with a symbol makes the overall design easier to understand. It is possible to create designs that are several levels deep in symbols, with the lowest level of symbols pointing to components from the GOULD AMI library. You may also mix hierarchical symbols and library components on the same sheet.

Each occurrence of a symbol points to the sheet containing the circuit description, so all occurrences of the symbol specify the same function. The circuit in the symbol can be tested independently of the entire design, making the testing of the design easier.

The important concept to note here is that all the tools in the GOULD AMI design kit will work on these hierarchical designs. You can expand each portion of the design and run AMI_ADD_DELAY on it independently and predict the behavior of the circuit. When EXPAND is run on the entire design the design is flattened into the DESIGN.EREL file, but the hierarchical information is retained. Back annotation works on this file, so all the true delay values will be represented from the physical implementation.

- Busses

  By using bus naming conventions and bus rippers you can represent multiple signals compactly on your schematic. This makes the schematic easier to understand and easier to draw. Section 4.4 of the IDEA SYSTEM USERS MANUAL gives a tutorial on creating busses and using bus rippers. The sample circuit MAGIC has some simple examples that illustrate the benefits of using busses.

- FOR Frames

  FOR frames allow you to replicate components using a simple expression. You only instantiate the component or group of components only once and include them in a frame. You then give the frame an expression declaring the number of replications. This has the same benefits as the bus structures. Your schematic is easier to read and more compact. Section 4.5 of the IDEA SYSTEM USERS MANUAL gives a tutorial on using FOR frames. FOR frames used in conjunction with busses are shown in MAGIC.

The command :

will execute NETED with the GOULD AMI menus included in the component libraries menu.

4.1.4 Defining the I/O's on the Chip

I/O's are the macros that transfer signals between the internal gate array and the outside world. There are special considerations when including these symbols in your schematic.

All the I/O's are prefixed with the letters IB, IO, or OB. The 'I' prefix is for input signals and the 'O' prefix is for output buffers (IO05 is a bidirectional driver). More information concerning the behavior of all the I/O macros can be found in the 3µ Single and Double-Metal CMOS Gate Arrays Databook.

It is common practice to assign a net name to the side of the I/O that goes off the chip. This net name is reflected on the root symbol (discussed in section 4.1.6). These net names are normally the pin names for the finished chip. These net names will not appear in the layout because the net names change when they go through the I/O buffer. For ease of use in CADIGRAPH, you should assign an associated net name to the net going to the internal area of the array. As long as this net name is on the top level schematic, the net name will pass into CADISYS unchanged. If the net name is down in the hierarchy of the circuit, a hierarchical name will be assigned to the net. This name is of the form 'instance/instance/assigned_net_name'.

4.1.5 Defining Critical Nets

Nets can be specified as critical to the layout system. When nets are identified as critical, they are given special consideration in both placement and routing. Macros containing the critical net will be placed closer together and the routing of the critical net is done before other nets. There are some ramifications of making nets critical. It can affect the performance of the placement and routing software. Completion rates may be reduced as accomodations are made for the critical nets. For this reason critical nets should be used with discretion.

To assign a critical net you would perform these steps in NETED :

1. Define the owner of the PRIO property: 'OWNER PRIO -NET'
2. Select the critical net
3. Add the PRIO property to the net: 'ADD PROP PRIO value'

The PRIO property indicates that this is a critical or priority net. The values for PRIO range from 0 to 16. Zero specifies no priority; 16 is the highest priority. A given priority value is not unique for a design. Several nets can have the same value because it is a weighting factor, not a net identifier.

There are three factors to be considered when assigning critical nets.

1. Critical nets should be used sparingly because of the reasons listed above.

2. Assigning priority to a large net (greater than 20 pins in the net) may not produce good results. In this case, it may be wise to consider preplacement and prerouting (see section 4.4).

3. If a priority value is assigned to a net that is repeated in the hierarchy of the design, every occurrence of that net will have a priority value. For example :

   a. Net N$1 is assigned priority 10 in component A

   b. Component A is instantiated in component B four times

   c. Component B is instantiated in the top sheet 2 times

   There are 8 occurrences of the net with a priority of 10

   This does not cause an error, but will affect placement and routing as described in the beginning of this section.

4.1.6 Creating a Root Symbol

It is necessary to create a root symbol for your chip. This provides information to the layout programs and is used when you generate the data to transfer your design back to GOULD AMI.

The symbol can be as simple as a box with a pin and a name for each pin on the chip. In general these pin names correspond to the output net names on the I/O buffers. **Each pin on the symbol must have a PINTYPE property**. The value of the PINTYPE property can be :

1. IN for input signals

2. OUT for output signals

3. IO for bidirectional signals

If you do not have a PINTYPE property you will not be able to generate correct test vector files. If the PINTYPE property is not one of the 3 listed values you will get incorrect test vector files.

MAGIC demonstrates the correspondence between the root symbol and top level schematic.

4.2 Simulation

When you use the GOULD AMI libraries, SIM and AMI_ADD_DELAY you can achieve three results :

1. a functional and timing analysis of the design before any layout has been performed

   The symbols contain the necessary information for functional and base line timing simulation. The AMI_ADD_DELAY program in conjunction with the technology data supplied for each array, allows you to vary the operating conditions for the circuit.

2. a timing analysis of the circuit once layout has been performed

   The AMI_ADD_DELAY program will read the capacitance values from layout and modify the DESIGN.EREL file so that the simulation reflects these accurate models.

3. test vectors generated from your simulation

   You will be able to deliver a set of test vectors to GOULD AMI for testing your circuit once it has been fabricated. GOULD AMI will be able to automatically generate the commands for their test equipment from the test vectors.

4.2.1 How to Design for Useful Simulation

You can increase the testability of your circuit by considering three factors :

1. Initialization - The ability to set the circuit to a known state at the beginning of the test cycle.

2. Controlability - The ability to set the internal nodes of the circuit to a given state with external signals.

3. Observability - The ability to observe at an output pin, the state of an internal node.

More information concerning these concepts can be found in section 6.3 of the 3µ Single and Double-Metal CMOS Gate Arrays Databook. This section appears as an appendix to this manual.

4.2.2 The Different Conditions for Pre-Layout Simulation

Pre-layout simulation can be performed under 6 conditions and several combinations of these 6 conditions. This is achieved using the AMI_ADD_DELAY program. The appendix of this manual contains the AMI_ADD_DELAY User Manual. This document explains the input data and equations used by the program in some detail.

In this section, the six simulation conditions are described and the setup for each condition is demonstrated.

4.2.2.1 Gate Delay Only

When you run AMI_EXPAND_DESIGN on a design, a new DESIGN.EREL file for SIM is created. The only delay values that are associated with each net are the RISE and FALL delays associated with each macro. If you run SIM at this point you will get the gate delays only.

To run a simulation using only gate delays you perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology** - this will create the new DESIGN.EREL file for the design, circuit.

2. **SIM circuit** - you will now be ready to run the simulator.

The command AMI_EXPAND_DESIGN runs the Mentor Graphics EXPAND program and creates the design file that is expanded to the primitive level. The design file contains the same information as file generated using EXPAND and the SIM SETUP command. AMI_EXPAND_DESIGN will not take over the display when it is executed.

4.2.2.2 Fanout Loading Delay

When you run AMI_ADD_DELAY it normally calculates the delay due to fanout loading and adds this to the total delay for each net. The only exception occurs when you ask for only layout delay (described in section 4.2.2.3). Also, when you run AMI_ADD_DELAY you get the delay from the predicted wiring added to the total delay for the net, unless you specify only fanout delay. The default for AMI_ADD_DELAY adds both the fanout and predicted wiring delays to the DESIGN.EREL file.

To run a simulation using both fanout and layout delays perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology**- if you have a current DESIGN.EREL file this is not necessary

2. **AMI_EXPAND_COMP circuit technology**- this will create a COMP.EREL file

3. **AMI_ADD_DELAY circuit technology array** - this will modify the DESIGN.EREL file to contain the new delay values

4. **SIM circuit** - ready for simulation

The AMI_EXPAND_COMP command again runs the Mentor Graphics program EXPAND, but now expands the design to the component level. The values for fanout calculation reside at the component level. Again, you will not see EXPAND take control of the display. This expansion will run faster than AMI_EXPAND_DESIGN because you are not expanding to the primitive level.

The parameters for AMI_ADD_DELAY are

1. circuit - the design name

2. technology - the GOULD AMI technology; GA_M3D, GA_M2D etc.

3. array - the array that is being used; AMI1K, AMI2K etc.

The technology and array names are described in more detail in section 3.2 of this manual and a list of all possible technology and array combinations is included as an appendix of this manual.

To run a simulation using fanout delay only, perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology** - if you have a current DESIGN.EREL file this is not necessary

2. **AMI_EXPAND_COMP circuit technology**- this is not necessary if you have a current COMP.EREL file

3. **AMI_ADD_DELAY circuit technology array -FO** - the format of this command is the same as above, but now the switch -FO indicates that this is fanout only

4. **SIM circuit** - ready for simulation

4.2.2.3 Predicted Wiring Delay

As described in section 4.2.2.2, the default for AMI_ADD_DELAY is to calculate and include the predicted wiring delay for simulation. The method for simulating with both fanout and predicted wiring delays appears in section 4.2.2.2.

To run simulation with predicted wiring delay only, perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology** - if you have a current DESIGN.EREL file this is not necessary

2. **AMI_EXPAND_COMP circuit technology** - if you have a current COMP.EREL file this is not necessary

3. **AMI_ADD_DELAY circuit technology array -LO** - the switch -LO will update the delay values in the DESIGN.EREL file with predicted layout only

4. **SIM circuit** - ready to simulate

The parameters and programs are the same as those described in section 4.2.2.2.

4.2.2.4 Varying the Delay Based on Temperature

The operating temperature of the chip affects its performance. You can simulate thes operating conditions by specifying an operating temperature to AMI_ADD_DELAY. AMI_ADD_DELAY reads the equation describing temperature affects from a file. This file is created for each technology/array combination. From this file a multiplicative factor is generated and used to modify the delay values of the entire array. This process is described in more detail in the AMI_ADD_DELAY USERS MANUAL appears as an appendix to this document.

A word of caution - Though the AMI_ADD_DELAY USERS MANUAL describes the method for building files that contain the coefficents for temperature equations, you should not change these coefficents. The equation they describe was derived from experimental data on test arrays. Changing the equation will decrease the accuracy of the procedure for modeling the behavior of GOULD AMI arrays.

To run a simulation with a specific operating temperature perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology** - if you have a current DESIGN.EREL file this is not necessary

2. **AMI_EXPAND_COMP circuit technology** - if you have a current COMP.EREL file this is not necessary

3. **AMI_ADD_DELAY circuit technology array -TEMP number** - the -TEMP switch followed by a value (in degrees Centigrade) will modify the delays for circuit.

4. **SIM circuit** - ready for simulation

In this example, the fanout and predicted wiring delays will also be included in the new delay values. The -TEMP switch can be used with the -LO and -FO switches as well. The -TEMP switch can also be used with the -V and process switches described in sections 4.2.2.5 and 4.2.2.6.

The current range for the temperature value is -55 to 135 degrees Centigrade. If you enter a value outside this range, an error message will appear in the output report file and the delay values will not be modified. The multiplicative factor for the delay values created by the new operating temperature is in the output report. This report is called 'circuit/ADD_DELAY.LOG'. The operating temperature that gives a multiplicative factor of 1 (nominal operating temperature) is 25 degrees. A curve displaying the affect of temperature on the performance of the array is included at the end of the 3µ Single and Double-Metal CMOS Gate Arrays Databook.

4.2.2.5 Varying the Delay Based on Voltage

As described in section 4.2.2.4, the operating voltage will affect the performance of the design. The method for altering the delay values based on operating voltage is the same as described for operating temperature. The word of caution also applies; do not change these files.

To run a simulation with a specific operating voltage perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology** - if you have a current DESIGN.EREL file this is not necessary

2. **AMI_EXPAND_COMP circuit technology** - if you have a current COMP.EREL file this is not necessary

3. **AMI_ADD_DELAY circuit technology array -V number** - the -V switch followed by a value will modify the delay values

4. **SIM circuit** - ready for simulation

In this example the fanout and predicted wiring delays will be calculated and used as well. The -V switch can be used with the -LO, -FO, -TEMP and process switches.

The current range for the voltage value is 4.5 to 5.5 volts. If you enter a value outside this range, an error message will appear in the output report file ('circuit/ADD_DELAY.LOG') and the delay values will not be modified. The multiplicative factor for the delay values created by the new operating voltage (and temperature if applicable) is in the output report. The operating voltage that gives a multiplicative factor of 1 (nominal operating voltage) is 5 volts. A curve displaying the affect of voltage on the performance of the array is included at the end of the 3µ Single and Double-Metal CMOS Gate Arrays Databook.

4.2.2.6 Varying the Delay Based on Process

There are three process conditions that are defined for modifying the delay values. The default value is the nominal process, and multiplies all the delay values by 1. You can also choose best and worst case process conditions for simulation.

To run a simulation with either best or worst case process, perform the following steps :

1. **AMI_EXPAND_DESIGN circuit technology** - if you have a current DESIGN.EREL file this is not necessary

2. **AMI_EXPAND_COMP circuit technology** - if you have a current COMP.EREL file this is not necessary

3. **AMI_ADD_DELAY circuit technology array -BEST (-WORST)** - the -BEST (-WORST) switch indicates that nominal process is not being used, and the multiplicative factor for the delays should be modified accordingly.

4. **SIM circuit** - ready for simulation

The -BEST and -WORST switches cannot be used at the same time, but they can be used independently any of the other switches described. A curve displaying the affect of process on the performance of the array is included at the end of the 3µ Single and Double-Metal CMOS Gate Arrays Databook.

SIM operates normally under any of these conditions. Section 4.6 describes how to generate test vectors from the SIM data.

Every time AMI_ADD_DELAY is executed, it will generate a report file in the design directory (same level where sheets reside) called ADD_DELAY_LOG. This report file contains the multiplicative factors used to modify the delays. It also contains all the old and new delay values for each net.

MAGIC demonstrates several simulation runs using different operating parameters.

A word of caution - Each time you run AMI_EXPAND_DESIGN you will set the design file to the **gate delay only** condition. It will be necessary to rerun AMI_ADD_DELAY to get to any other simulation condition.

4.3 Package Definition

The package definition program assigns the pads of the gate array to pins on a package. The pads of the gate array are the locations where the internal signals of the array access the pins of the package. The pads are configured into functions by the I/O macros that are described in section 4.1.4.

There are several packages for each array type. Information on choosing an appropriate package can be found in chapter 7 of the 3μ Single and Double-Metal CMOS Gate Arrays Databook. A picture of each package that is available in the library is included as an appendix to this document.

The following command executes the package assignment program.

AMI_PKG_DEF circuit technology array package

The parameters for AMI_PKG_DEF are

1. circuit - the design name
2. technology - the GOULD AMI technology; GA_M3D, GA_M2D etc.
3. array - the array that is being used; AMI1K, AMI2K etc.
4. package - the name of the package; PKG_40_PIN etc.

Information describing technology, array and package names can be found in section 3.2 of this document.

When you issue the AMI_PKG_DEF command the following sequence of events occurs.

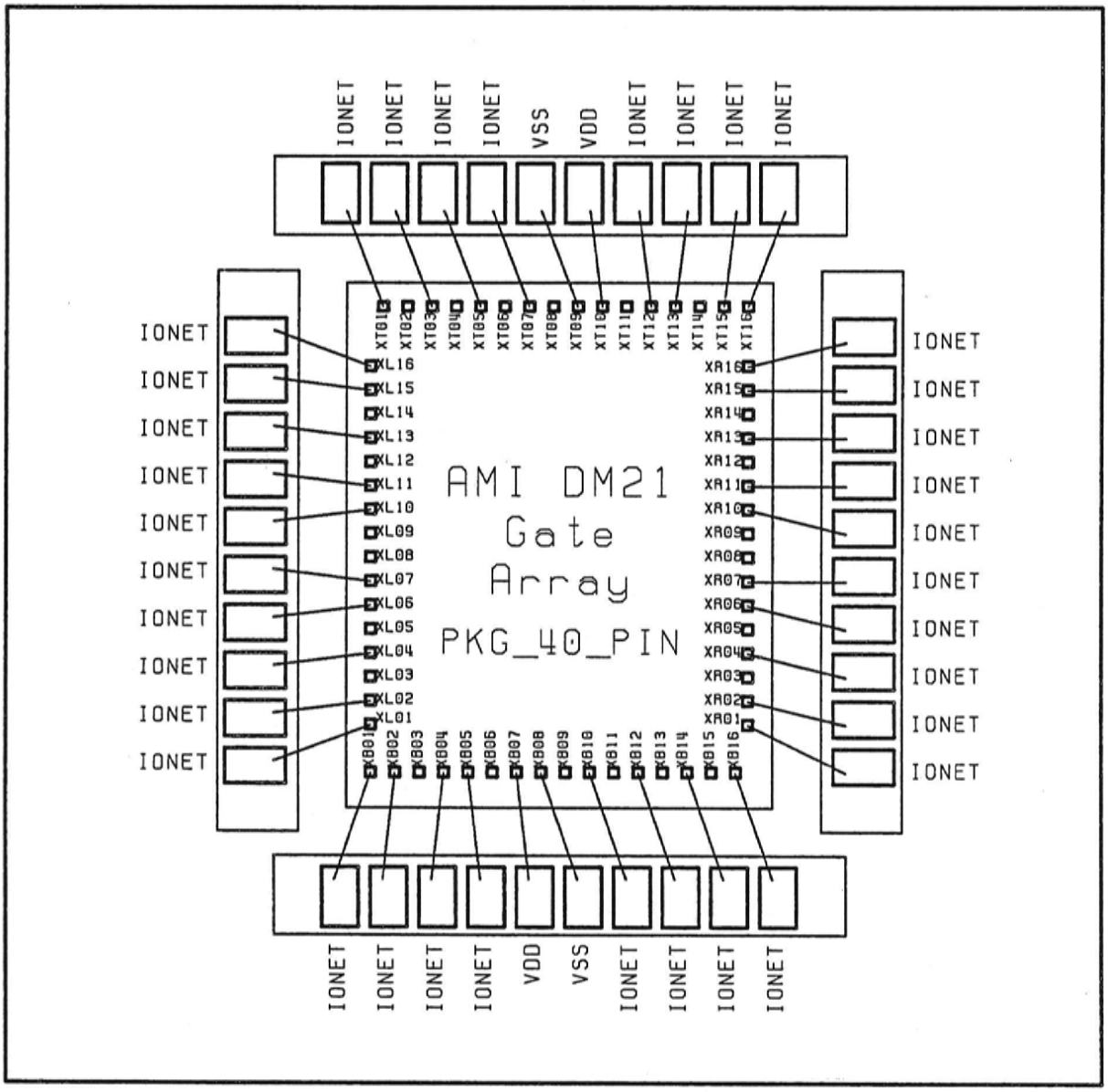


Figure 4-1. Example Package Symbol

1. The names of all the nets that go to pins of the package are extracted from the root symbol of the design and stored in a temporary file. (Review section 4.1.4 concerning I/O assignment on the schematic and section 4.1.6 on creating a root symbol)

2. The names of all these nets are used to generate a NETED menu.

3. NETED is invoked with a symbol representing the package. (Figure 4.1 shows an example package symbol)

4. You assign the nets to the package symbol using a NETED menu

5. When you exit NETED, a file containing all the I/O assignments for the layout software is generated

All the steps of this process are performed automatically, except for the net assignment to pin locations. To understand how to do this, let us first examine the package symbol.

The large rectangle in the center with the title '**AMI DM21 Gate Array PKG_40_PIN**' represents the gate array. The small squares on the periphery of the array represent all the possible locations for I/O macros. The medium rectangles surrounding the gate array represent the pins of the package. This example has 40 pins on the package. The lettering associated with the pins is the name of the net for that pin. In the example all net name are IONET (indicating they have not been assigned yet), VDD or VSS (indicating that these pins are reserved for these signals).

Figure 4-1 is the symbol that appears when NETED is executed. The menu that appears is a list of all the signals that can be assigned. To assign a signal to a pin you perform the following steps :

1. Place the cursor over the signal in the menu

2. Press the CMD key (or point stroke)

3. Place the cursor over the IONET label of the pin you wish to assign

4. Press <return>

If the selected signal can be assigned to the selected pin, the IONET label will change to the net name and the net name will disappear from the menu. If the signal cannot be assigned to the selected pin, a message will appear in the transcript window and the IONET label will not change. When you are satisfied with the assignments execute the menu item **$_EXIT_PKG_NETED** and you will exit NETED and continue with the program sequence.

It is not necessary to assign all the nets to pins or even any of the signals, but it is necessary to run AMI_PKG_DEF before any layout is performed. AMI_PKG_DEF performs the useful function of allowing you to specify pin assignment, and it also performs a function indispensable to the layout software.

Typically, all the possible I/O locations are not used in a package. When a package uses only a subset of the I/O locations, there are a fixed subset that can be used. The other I/O locations must be ignored by the layout software. AMI_PKG_DEF tells the layout software which locations can be used. If you do not assign the signals, the layout software will swap the I/O's in all the legal locations to achieve an optimum layout (this process will be described in section 4.4), but it will not use the illegal locations. If you do not run AMI_PKG_DEF, the layout software may swap I/O's in all I/O locations. This can create illegal placements. For this reason, AMI_PKG_DEF must always be run before layout.

MAGIC demonstrates AMI_PKG_DEF.

4.4 Physical Layout

The layout of the design is performed by the CADISYS gate array layout software. This is true whether the layout is done on GATE STATION or at GOULD AMI. For more information on CADISYS, see the Mentor Graphics GATE STATION Users Manual and the Mentor Graphics CADIGRAPH Users Manual.

4.4.1 How CADISYS Works

The CADISYS package consists of four functions: design creation, placement, routing and interactive editing.

- Design Creation

  When you are satisfied with your design and you have run AMI_PKG_DEF you can create the design file for layout. This is done by executing the command

  AMI_DDFNET circuit technology

  This will create a DDF netlist for the design 'circuit' using the GOULD AMI technology 'technology' (GA_M3D is an example) for the array 'array' (AMI1K is an example). The file that is created is an ASCII netlist and will be called 'circuit/CADI.DDFIN'.

  When the DDF netlist is completed you will create a binary design file for the placement and routing. This is done by executing the command

  AMI_LAYOUT LOGIC_ENTRY circuit technology array

  The command AMI_LAYOUT indicates that you are using the GOULD AMI design kit for layout. The argument LOGIC_ENTRY indicates that you want to build a design file. Any command described in Mentor

Graphics GATE STATION Users Manual can be executed under the AMI_LAYOUT command when it is submitted as the second argument. The AMI_LAYOUT command recognizes the GOULD AMI design kit directory structure and will log the execution of the program for future reference. (For more information concerning execution logging see section 5 of this manual.) The last three parameters are the same as described earlier.

The design file that is generated is binary and resides in 'circuit/PHYSICAL_DESIGN/DESIGN.DSN'.

o Placement

Automatic placement takes all the macros in the design and optimally arrange them on the uncommitted array. This is a difficult, computationally intensive problem.

Consider that there are potentially several hundred objects (N) and just as many locations (N) for these objects. To try all possible locations for all objects is an N! problem. It is, therefore, impossible to try every placement combination. Some method for estimating a good placement without trying every possible combination must be determined. This is what placement does.

It is also necessary to define what a 'good placement' is. The most reasonable definition is determined by the performance of the router. If you achieve a very high routing completion rate, the placement was good. Placement is creating an arrangement of macros that will achieve high routing completion rates.

For more information on the automatic placement program see the Mentor Graphics GATE STATION Users Manual.

To execute automatic placement you use the command :

AMI_LAYOUT CADIPLACE circuit technology array

The arguments are the same as described in Design Creation above. This command will create an updated design file 'circuit/PHYSICAL_DESIGN/DESIGN.PLC'.

o Routing

Automatic routing works on a placed design file and adds the metal paths to connect all the signal pins of the macros. Routing is also a computationally intensive job.

When there are a large number of connections that must fit in a finite area, it is beneficial to do some planning before doing any connections. By looking at the entire chip, problem areas where there are too many connections can be flagged by the automatic routing programs. The first portion of routing is performing this global analysis. It distributes the connections across the chip so most can be made, while still attempting to keep the shortest wire lengths possible.

After the global analysis is completed, a plan for routing all the nets is given to programs that put the connections on the chip. These programs optimize the routing so that as many connections as possible can be packed into the routing channels.

The scheme described above routes most of the connections. The remaining connections are attempted by more tenacious routers that will try more difficult paths. Nets will be moved to accomodate unmade connections when possible. More information about the automatic routing programs can be found in the Mentor Graphics GATE STATION Users Manual.

To execute automatic routing you use the command :

AMI_LAYOUT CADIROUTE circuit technology array

The arguments are the same as described in Design Creation. This command will update 'circuit/PHYSICAL_DESIGN/DESIGN.DSN'.

o Interactive Editing

CADIGRAPH is an interactive editor that allows you to change the placement and routing of your design. You can complete nets that were not completed by the automatic programs. CADIGRAPH can also be used for preplacement and prerouting as described in sections 4.4.2 and 4.4.3.

The Mentor Graphics CADIGRAPH Users Manual describes the operation of CADIGRAPH in detail. To execute CADIGRAPH you use the commands :

AMI_LAYOUT PREGRAPH circuit technology array

AMI_LAYOUT CADIGRAPH circuit technology array

The first command creates the graphics database for CADIGRAPH. Once this database has been created, CADIGRAPH can access it multiple times. It only needs to be recreated when the physical design file changes. A further description of the graphics work file and its

use can be found in the Mentor Graphics CADIGRAPH Users Manual.

CADIGRAPH is a user-friendly editor, but it is also very powerful. Just as it was suggested that you take the Mentor Graphics training course on IDEA STATION before using NETED and SIM, it is suggested that you take the Mentor Graphics training course on GATE STATION. This course focuses on the operation of CADIGRAPH as well as the other layout tools.

4.4.2 Preplacement Using CADIGRAPH

CADIGRAPH can be used for preplacing macros. Preplacement of macros should be used sparingly because it creates constraints on the automatic placement that can reduce routing completion. To do preplacement the layout commands are executed in the following sequence :

1. **AMI_LAYOUT LOGIC_ENTRY circuit technology array**
2. **AMI_LAYOUT PREGRAPH circuit technology array**
3. **AMI_LAYOUT CADIGRAPH circuit technology array**
4. **AMI_LAYOUT CADIPLACE circuit technology array**
5. **AMI_LAYOUT CADIROUTE circuit tehcnology array**

The Mentor Graphics CADIGRAPH Users Manual describes how you place unplaced macros using CADIGRAPH.

4.4.3 Prerouting Using CADIGRAPH

CADIGRAPH can be used to preroute nets once you have a completely placed design. Prerouting nets should be done sparingly because it will affect the performance of the automatic routing programs. To do prerouting, the layout commands are executed in the following sequence :

1. **AMI_LAYOUT LOGIC_ENTRY circuit technology array**
2. **AMI_LAYOUT CADIPLACE circuit technology array**
3. **AMI_LAYOUT PREGRAPH circuit technology array**
4. **AMI_LAYOUT CADIGRAPH circuit technology array**
5. **AMI_LAYOUT CADIROUTE circuit technology array**

The Mentor Graphics CADIGRAPH Users Manual describes how you route nets before automatic routing.

4.5 Back Annotation

After layout, the capacitance value of each net can be extracted from the design and a more accurate net delay can be calculated. The new delay is inserted into the DESIGN.EREL file by the AMI_ADD_DELAY program.

You can receive back annotation information from two sources :

1. By extracting it from the physical design file if you performed the layout on your work station.

2. By receiving a file from GOULD AMI containing the capacitance values after layout has been performed at GOULD AMI.

To back annotate the capacitance values from your design file you would execute the command

AMI_ADD_DELAY circuit technology array -BA

The -BA switch indicates that the data should be extracted from the physical design file. Once this data has been extracted it is not necessary to use the BA switch when running AMI_ADD_DELAY, unless the physical design file is changed. When this data has been extracted once, it is available every time AMI_ADD_DELAY is used. If the physical design file changes the BA switch should be used again. The BA switch can be used in conjunction with any of the other switches described in section 4.2.2.

To back annotate the capacitance values from GOULD AMI, you install the capacitance file from GOULD AMI (described in section 5 of this manual) and execute the command

AMI_ADD_DELAY circuit technology array

The first command installs the delay values in the correct file and AMI_ADD_DELAY is run to update the DESIGN.EREL file. You execute AMI_ADD_DELAY without the -BA switch. If the -BA switch is used, a warning will be issued stating that there is no physical design file for extracting capcitance values. Any of the other AMI_ADD_DELAY switches can be used.

4.6 Test Vector Generation

Test vectors are automatically generated from a SIM list using the AMI_SIMOUT command. The factors you should consider in generating test vectors are discussed in chapter 6.3 of the 3µ Single and Double-Metal CMOS Gate Arrays Databook; this chapter is also included as an appendix to this

document.

Before you execute AMI_SIMOUT you must run AMI_EXPAND_COMP. This creates a COMP.EREL file with the correct data for running the programs that create the GOULD AMI formatted files for test vector generation. You must also have the PINTYPE property attached to every pin on the root symbol. Review section 4.1.6 for more information on creating a root symbol. To execute AMI_SIMOUT you issue the command

AMI_SIMOUT circuit technology

AMI_SIMOUT performs two functions. When it is executed for the first time, it builds a template 'do' file for SIM. This 'do' file has the following characteristics :

1. All I/O signal names are included in the list window
2. All I/O signals are logged only on the clock cycle
3. A clock period is defined
4. A run length is defined
5. A write list command is included

All this data is necessary for generating the correct test vectors. When AMI_SIMOUT is executed it will prompt you for the following information :

- **Enter desired CLOCK period for SIM:**

  Respond with the operating clock period of the chip in nano seconds

- **Enter desired RUN TIME for SIM:**

  Respond with the total run time for this test in nano seconds

The 'do' file is named 'circuit_SIM' and placed in the current working directory. You will have to add your force statements to 'circuit_SIM' using the Aegis editor. After adding FORCE statements you execute this file by typing

DO circuit_SIM

from SIM. The list window will be written to 'circuit/SIM_LIST'.

You can change the order of the signals in the list window for convenience, but do not delete any of the signals or change the update period for the

signals. If either of these conditions is changed, the commands for the GOULD AMI test system can not be generated correctly. It is also possible to change the run time because the list file will be checked for this value before the output files are generated.

After SIM has been run and the 'circuit/SIM_LIST' file has been generated, the command

AMI_SIMOUT circuit technology

is again executed. This time AMI_SIMOUT will read the 'circuit/SIM_LIST' file and prompt you for the following information

- o Enter desired SNAP SHOT TIME:

  Respond with the time in the clock cycle at which the signals should be sampled. For example, a reasonable snap shot time for a clock cycle of 100 would be 90

AMI_SIMOUT creates two ASCII files: 'circuit/TESTFORM_INP' and 'circuit/SIMAD_INP'. These will be used at GOULD AMI to create the test programs for this design. Transfering these files as well as other data is discussed in section 5 of this manual.

5. DATA TRANSFER

This section describes how you will recieve data from GOULD AMI and how you will send design information back.

5.1 Receiving Data

The floppies containing the GOULD AMI design data have command procedures to facilitate the installation of the system. The AMI tree described in section 3 of this manual will be delivered on the floppies. The first file on floppy 1 is an installation command. To install the AMI tree the following actions must be performed.

1. Insert floppy 1

2. **RBAK -DEV F -F 1 -ALL** - this will install the file AMI_INSTALL into the current working directory

3. **AMI_INSTALL** - this command file will prompt you for the following information

   a. Enter desired NODE ID for design kit (name only)

      Enter the name of the node where the AMI tree will reside. For instance, if the node is //NODE_2846, enter NODE_2846.

4. If an AMI directory exists, the command file will rename the old directory and continue installing the new directory. The name given to the old AMI directory is 'AMI.date'.

5. You will be prompted to swap floppies until the entire tree is installed.

6. You will be prompted for installing the demo circuit. This is the MAGIC circuit described in section 6 of this manual. MAGIC occupies apporiximately 1.2 MB of disk space.

The procedure for installing incremental updates will be sent with the updates, but will be similar to the procedure described here.

5.2 AMI_INSTALL_CSR

The AMI_INSTALL_CSR command file has been provided in the ~AMI/COM directory to facilitate running GOULD AMI applications. This command file adds the string '~AMI/COM' to the CSR list of individual users. Before using AMI_INSTALL_CSR, please read this explanation of how it works. Some assumptions are made about the configuration of your network by this command

and it will change user startup files.

AMI_INSTALL_CSR accepts a valid user name as the first argument. This user name should be included as a link in the /USER directory of the network. The link should point to a /LOCAL_USER directory for that user. This relationship is described in the CREATE_USER documentation of the IDEA SYSTEM MANAGER'S MANUAL chapter 7.

AMI_INSTALL_CSR assumes that there is a 'user/USER_DATA' directory. If this directory does not exist, it will be created. In this directory it will attempt to modify the 'startup' command file. There is a startup file for login and separate startup files for each display type. The startup file for login is called 'STARTUP_LOGIN'. The process startup files are called 'STARTUP.DM', 'STARTUP.DM_19L', and 'STARTUP.DM_COLOR'. The 'STARTUP.DM' file is for the older 15 inch displays. 'STARTUP.DM_19L' is the file for the 19 inch screens and 'STARTUP.DM_COLOR' is for the series 600 color displays. If these files do not exist they will be created.

These startup files may contain key definitions. AMI_INSTALL_CSR adds the string '~AMI/COM' to an existing key definition for the SHELL key. If the new startup is created, it will contain the key definition for the SHELL key that executes 'CSR . /COM ~COM ~AMI/COM' every time a shell is invoked. If the startup files do not exist, this data is being read from the '/SYS/NODE_DATA' directory when a process was invoked.

Including this string in your CSR list will allow you to run all the GOULD AMI specific applications described in this manual without prefixing each command with '~AMI/COM/command'. Now you will be able to type the command. All of the GOULD AMI commands are prefixed with the string 'AMI_', so they should be unique to your system.

It is possible to edit your startup files by hand if you prefer. If you have some hesitation in using AMI_INSTALL_CSR, make a copy of your 'user/USER_DATA' directory (it is normally a small directory) and try it. If the results are not satisfactory, you can always go back to the old version.

To execute the command type

~AMI/COM/AMI_INSTALL_CSR user_name

5.3 Transmitting Data

All the files you will be sending to GOULD AMI, with the exception of the schematic tree, are relatively small ASCII files. Even the data for layout is transmitted in a compact ASCII format. Currently the only transmission method is by floppy. The following list summarizes all the files that will

be transmitted to GOULD AMI.

1. BOLT NETLIST DATA - ASCII netlist files for running the GOULD AMI simulator

2. DDF NETLIST - ASCII netlist for running GOULD AMI layout software

3. circuit_SIM - The file of SIM commands used to generate the test vectors

4. SIM_LIST - The list window from SIM when 'circuit_SIM' do file was executed

5. SIMAD_INP - Command statements for running a test pattern through the GOULD AMI simulator (this will be used to generate a command sequence for the GOULD AMI test equipment)

6. TESTFORM_INP - The expected results from the test pattern

7. DDF OUTPUT - Contains all the placement and routing information

8. DESIGN LOG - Contains version numbers for all the data used in the design stream; This file is explained in section 5.4.

9. SCHEMATIC - The entire schematic

The following sections describe how to transmit data for each design stream. The arguments listed below are used in the transmit command descriptions.

1. circuit - the design name

2. technology - the GOULD AMI technology; GA_M3D, GA_M2D etc.

5.3.1 Design Stream A

You will need the BOLT and DDF netlist data for design stream A. To generate the the BOLT netlist file you will need to issue the following command:

AMI_BOLTNET circuit

This will generate the 3 files 'circuit/BOLT.NETLIST', 'circuit/BOLT.ORDER' and 'circuit/BOLT.ORDER'. This program will not work if you do not have a root symbol (review section 4.1.6 of this manual).

The following command generates the DDF netlist file.

AMI_DDFNET circuit technology

This will create the file 'circuit/CADI.DDFIN'. The DDF netlist cannot be generated if there is no a root symbol (review section 4.1.6 of this manual). The netlist will not be correct if AMI_PKG_DEF has not been run on the design (review section 4.3 of this manual).

To generate the data for stream A execute the command:

AMI_STREAM_A circuit

This command checks to see that all the necessary data exists and copies it into the directory 'circuit.STRMA'. You will then write this directory to a floppy using the command:

WBAK circuit.STRMA -DEV F -F 1 -FID STRMA -OWN AMI

You also must include the schematic directory on the same floppy, or another floppy if you desire. If it is included on the same floppy you would use the following command:

WBAK circuit -DEV F -F 2 -FULL

The following list shows all the files that must be included in the stream A floppy.

File 1

1. BOLT.NETLIST
2. BOLT.ORDER
3. CADI.DDFIN
4. DESIGN_LOG

File 2

1. Entire schematic tree

5.3.2 Design Stream B

All the data required for design stream A is required for design stream B. In addition, design stream B must have the following data.

1. circuit_SIM

2. circuit/SIM_LIST

3. SIMAD_INP

4. TESTFORM_INP

For information on generating 'circuit/SIMAD_INP' and 'circuit/TESTFORM_INP' review section 4.6 of this manual. The 'circuit_SIM' and 'circuit/SIM_LIST' are included in case the simulation needs to be checked. These files are generated with the AMI_SIMOUT command that is used to generate SIMAD_INP and TESTFORM_INP.

To generate the data for stream B execute the command:

AMI_STREAM_B circuit

This command checks to see that all the necessary data exists and copies it into the directory 'circuit.STRMB'. You will then write this directory to a floppy using the command:

WBAK circuit.STRMB -DEV F -F 1 -FID STRMB -OWN AMI

You also must include the schematic directory on the same floppy, or another floppy if you desire. If it is included on the same floppy you would use the following command:

WBAK circuit -DEV F -F 2 -FULL

The following list shows all the files that must be included in the stream A floppy.

File 1

1. BOLT.NETLIST

2. BOLT.ORDER

3. CADI.DDFIN

4. SIM_CMDS

5. SIM_LIST

6. SIMAD_INP

7. TESTFORM_INP

8. DESIGN_LOG

   File 2

1. Entire schematic tree

The back annotation information from GOULD AMI is contained in an ASCII file that is sent to you via floppy. To install this file you would perform the following steps :

1. Insert the floppy

2. **WD circuit**

   This should locate you in the directory containing information for the design such as sheets, symbols and the DESIGN_LOG file.

3. **RBAK -DEV F -F 1 -ALL -AS NETPARM.DDF**

   This command will write the capacitance data into the file NETPARM.DDF.

The AMI_ADD_DELAY program will read capacitance data from the file 'circuit/NETPARM.DDF' if the file exists.

5.3.3 Design Stream C

All the data required for design stream B is required for design stream C. In addition, design stream C must have the following data.

1. DDF OUTPUT

The following command generates the DDF output file.

AMI_LAYOUT DDF_OUTPUT circuit technology array

This will create the file 'circuit/PHYSICAL_DESIGN/DESIGN.DDF'. The DDF output cannot be generated if there is no physical layout data (review section 4.4 of this manual).

To generate the data for stream C execute the command:

AMI_STREAM_C circuit

This command checks to see that all the necessary data exists and copies it into the directory 'circuit.STRMC'. You will then write this directory to a floppy using the command:

WBAK circuit.STRMC -DEV F -F 1 -FID STRMC -OWN AMI

You also must include the schematic directory on the same floppy, or another floppy if you desire. If it is included on the same floppy you would use the following command:

WBAK circuit -DEV F -F 2 -FULL

The following list shows all the files that must be included in the stream A floppy.

File 1

1. BOLT.NETLIST
2. BOLT.ORDER
3. CADI.DDFIN
4. SIM_CMDS
5. SIM_LIST
6. SIMAD_INP
7. TESTFORM_INP
8. DESIGN.DDF
9. DESIGN_LOG

File 2

1. Entire schematic tree

5.4 The DESIGN_LOG File

The 'circuit/DESIGN_LOG' file contains a log of programs and data used to generate this design. The version files that are kept in each '~AMI/technology/CELLS' directory and version information for the physical data bases are used to generate this data.

Any time an expand program or physical layout program is run, the time, date and version of the data is logged to this file. This is to help ensure that your design can be correctly processed at GOULD AMI. An example 'DESIGN_LOG' is included with MAGIC.

6. MAGIC

A demonstration circuit is delivered with the GOULD AMI design kit. This circuit is used to demonstrate the procedures described in this manual.

6.1 What Does MAGIC Do

MAGIC is a simple design consisting of two major functions. The first function is a shifter. The second function performs data selection. Figure 6-1 shows the MAGIC circuit.

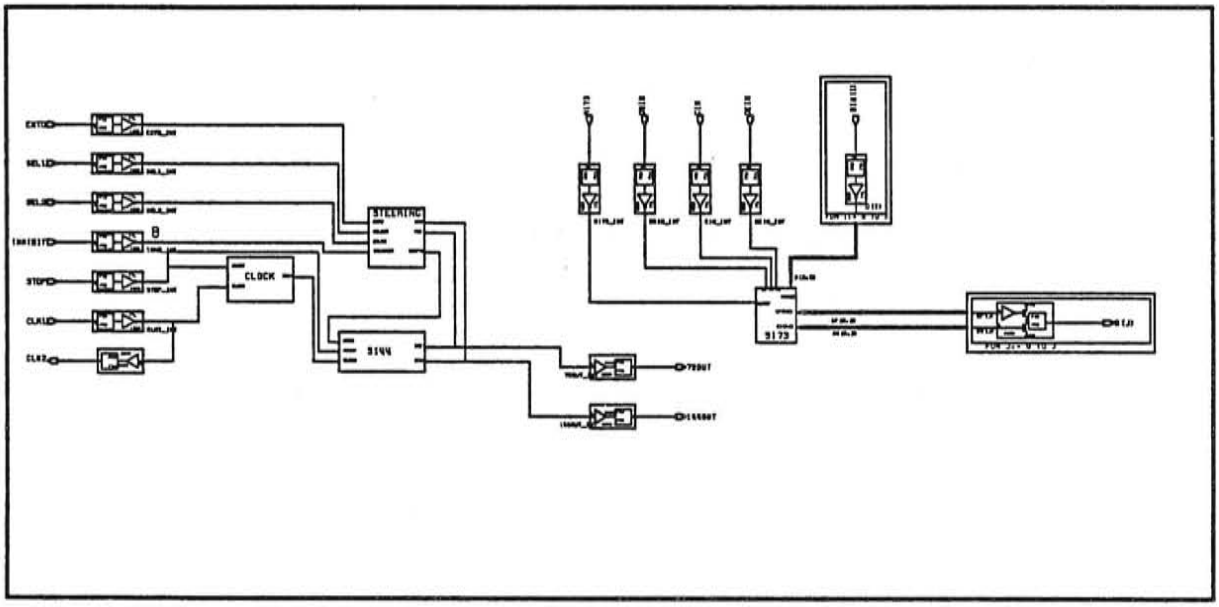


Figure 6-1. MAGIC

The three blocks in the left hand portion of MAGIC make up the shifter and the data selecter is the block located in the right hand portion of the schematic. The rectangular symbols that are connected to the blocks are I/O macros for the gate array.

Figure 6-2 shows the shifter in greater detail.

The CLOCK symbol divides the signal CLK1 in half. This causes the shifter S144 to run at half the speed of the external clock signal.

The STEERING symbol determines the signal that is sent to the S144. When the INHIBITP signal is low, SHIFTD (output from steering) is low and will remain low. When INHIBITP is high, signals are allowed through STEERING. The input signal SELEXT determines whether the signal EXTD is sent to S144

or one of the outputs from S144 is selected as the input to S144. When SELEXT is high EXTD is the input to the shifter. If SELEXT is low either Q72 or Q144 is the input into the shifter. SEL72 determines which signal will be the input. If SEL72, is high the input is Q72 and when SEL72 is low Q144 is the input.

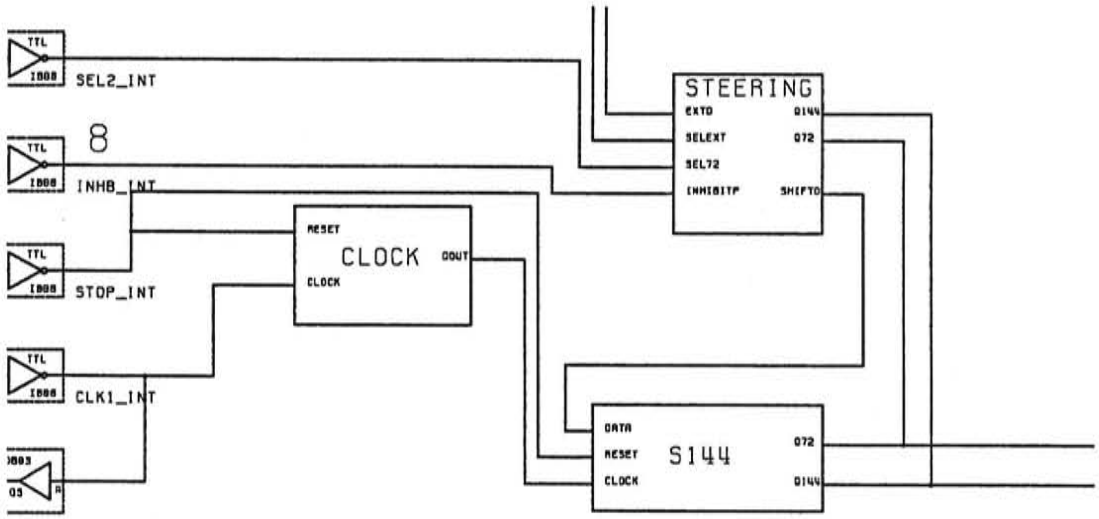


Figure 6-2. The Shifter

The S144 symbol takes an input signal in on DATA and shifts it it through a series of D flip flops. The signal appears on the output Q72 after 72 clock cycles. The same signal appears on Q144 144 clock cycles from when it appears on DATA.

The signal names with the suffix '_INT' are named so they can be easily viewed in CADIGRAPH. The '8' is a net priority of 8 on the inhibit line.

The data select circuit is shown in figure 6-3.

When R173P is low, the flip flops in S173 are set to 0. The QN and QP outputs of S173 drive a tristate buffer. When QP and QN have opposite values, the output from the buffers will be the QN . Both QP and QN cannot be high at the same time. When both QP and QN are low, the output buffer is at high impedence. When OE is low the output buffers will be at unknown high impedence.

The input bus D(0:3) is data. When DS is high the data from the D bus is clocked into S173 with the clock CN. The inverse of this bus will appear on the tristate buffer outputs if OE is high after the next clock cycle . When DS is changed to low the D bus is inhibited and the state of the output buffers is maintained at the state of the D bus that was received when DS and OE were high. Anytime OE goes low, the output buffers will go to high impedence.

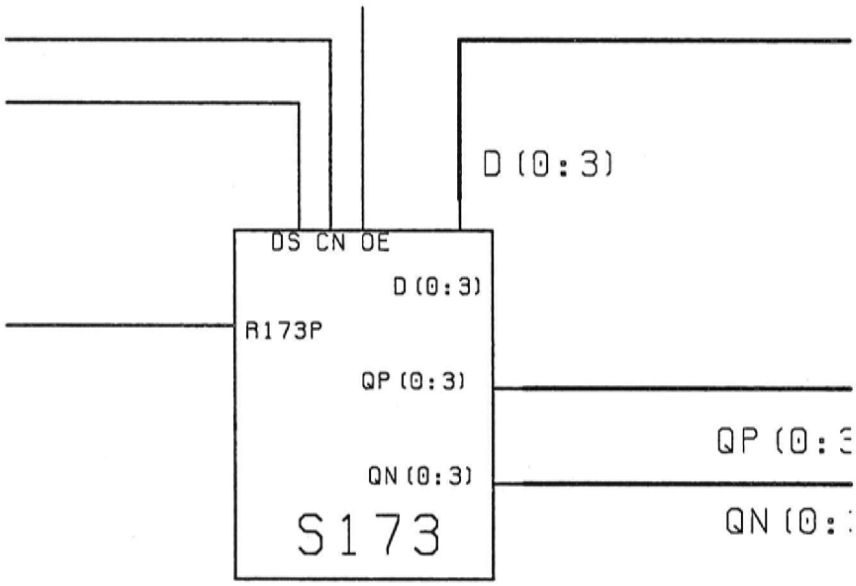


Figure 6-3. Data Select

The symbol for MAGIC is shown in figure 6-4. Notice that all the pins have a PINTYPE property. Reivew section 4.1.6 for an explanation of the PINTYPE property. The root symbol with PINTYPEs is necessary to ensure the correct operations of several of the design kit programs.

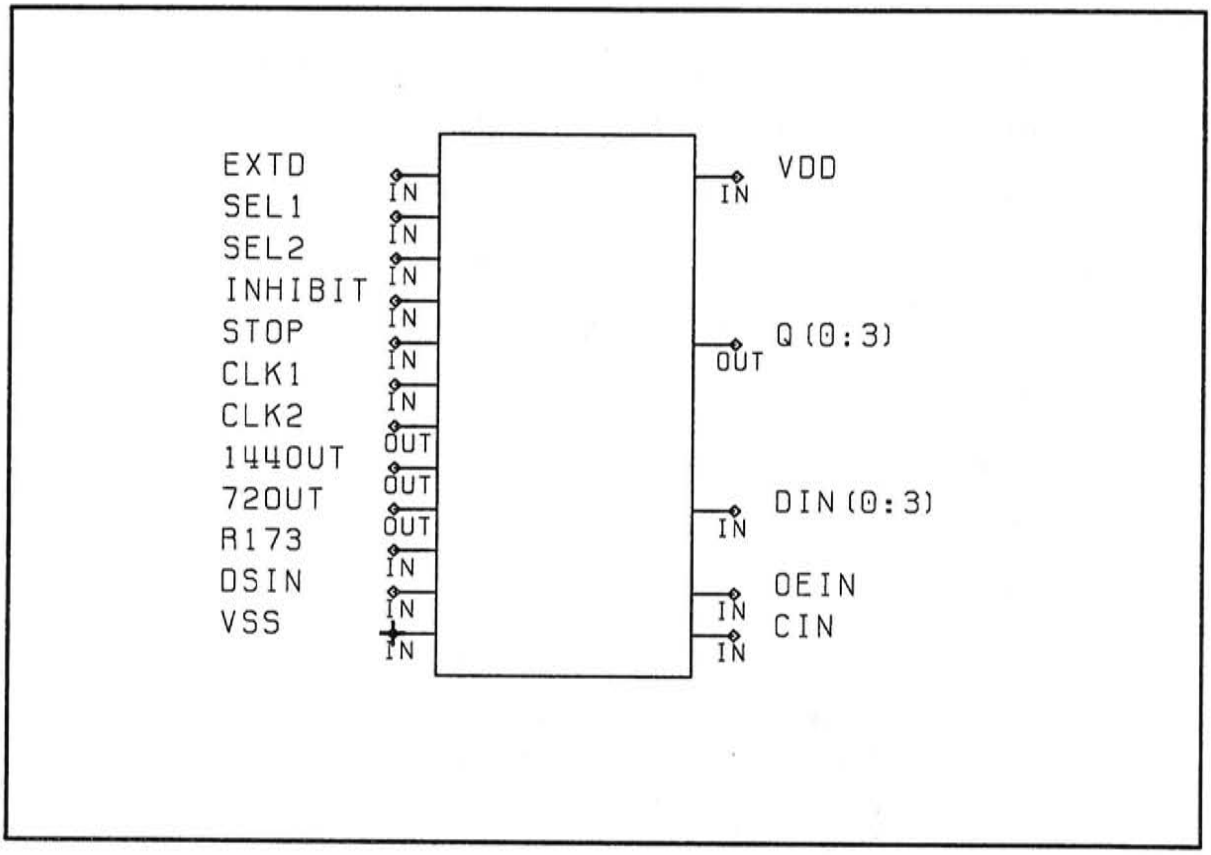


Figure 6-4. Root Symbol

6.2 What Comes with MAGIC

MAGIC is located in the directory '~AMI/DEMO/MAGIC'. This directory contains the sheets and symbols described in section 6.1. It also contains expamples of program inputs and corresponding outputs.

The directory 'CIRCUIT' is the design directory. The file DESIGN_LOG is an example log file. The remaining files are the program examples. The files with the .INP suffix are input examples. The input files contain the command line that is submitted for that program, any messages that are issued by the program during execution and pertinent ASCII input files that are used. The files with the .OUT suffix are the output logs. These files contain a list of the files that were created by the program, files that were modified by the program and output report files.

The following list briefly describes all the files and directories :

1. add_delay_best.inp, add_delay_best.out

   Executing add delay using best case operating conditions.

2. add_delay_default.inp, add_delay_default.out

   Executing add delay using default operating conditions.

3. add_delay_nominal.inp, add_delay_nominal.out

   Executing add delay using nominal operating conditions.

4. add_delay_worst.inp, add_delay_worst.out

   Executing add delay using worst case operating conditions.

5. boltnet.inp, boltnet.out

   Executing the BOLT netlister.

6. circuit

   This is the design directory. Execute AMI_NETED CIRCUIT to see the MAGIC circuit described in section 6.1.

7. ddfnet.inp, ddfnet.out

   Executing the DDF netlister.

8. ddfout.inp, ddfout.out

   Executing the DDF output program.

9. design_log

   An example design log file.

10. expand_comp.inp, expand_comp.out

    Expanding MAGIC to the component level.

11. expand_design.inp, expand_design.out

    Expanding MAGIC to the primitive level.

12. package_definition.inp, package_definition.out

Executing the package definition program.

13. physical_layout.inp, physical_layout.out

    Executing the CADISYS job stream for physical layout.

14. simout_after_sim.inp, simout_after_sim.out

    Executing the test vector generation programs after SIM.

15. simout_before_sim.inp, simout_before_sim.out

    Configuring SIM for test vector generation.

16. sim_short_run.inp, sim_short_run.out

    A short simulation run example.

17. sim_test_vectors.inp, sim_test_vectors.out

    The force statements and list output for generating the test vectors

18. stream_a.inp, stream_a.out

    Generating the output data for design stream A.

19. stream_b.inp, stream_b.out

    Generating the output data for design stream B.

20. stream_c.inp, stream_c.out

    Generating the output data for design stream C.

6.3 How to Use MAGIC

MAGIC comes with only the sheets and symbols. You will generate the rest of the data. It is meant as a learning tool for you.

In each .INP file the fist line is the command you should issue to execute the programs. Use the responses and messages in the .INP file as a comparison when you run the program. The corresponding .OUT files give information on the changes you should see occur.

If you run MAGIC through layout, the directory will occupy approximately 5 MB of disk space. Experiment with MAGIC before starting with your design and use it as a reference. When you are through with MAGIC, the entire directory can be deleted.

7. APPENDICES

7.1 Technologies Supported by this Design Kit 2-4-85

1. CMOS 3µ Double-Metal Gate Array

7.2 Packages Supported 2-4-85

CMOS 3µ Double-Metal Gate Array, 1K gate count

1. 24 pin - plastic
2. 28 pin - chip carrier, plastic
3. 40 pin - ceramic, chip carrier, plastic, TJF
4. 44 pin - chip carrier, MFP
5. 48 pin - ceramic, plastic, chip carrier
6. 64 pin - ceramic, plastic
7. 68 pin - MFP, pin grid array, chip carrier

CMOS 3µ Double-Metal Gate Array, 2K gate count

1. 24 pin - plastic
2. 28 pin - plastic
3. 40 pin - ceramic, plastic, MFP
4. 44 pin - MFP
5. 48 pin - ceramic, chip carrier
6. 64 pin - ceramic, plastic
7. 68 pin - chip carrier, PCC, MFP, pin grid array
8. 84 pin - chip carrier, PCC, pin grid array

CMOS 3µ Double-Metal Gate Array, 3K gate count

1. 40 pin - plastic, MFP
2. 44 pin - MFP
3. 68 pin - chip carrier, MFP
4. 84 pin - pin grid array, chip carrier
5. 100 pin - pin grid array

7.3 AMI_ADD_DELAY Users Manual

7.3.1 INTRODUCTION

This document describes the use of the add_delay program in conjuntion with Gould-AMI semi-custom libraries. Capabilities and limitations of the program will be discussed is this document while methods of program modifications will be discussed in the REFERENCE MANUAL.

Add_delay is a Design File Interface (DFI) program which modifies the rise and fall times in a gate array or standard cell design file to reflect additional delay due to net loading, net routing, operating temperature, operating voltage, and/or standard cell die size. In addition, add_delay modifies rise and fall times to reflect your choice of best, worst, or nominal case analysis. Any additional delay placed in the design file is reflected in subsequent simulation and timing verifier runs. It can be used both before and after physical layout of the chip and does not have to be used with the Cadisys layout software. The program can calculate additional delay for any of the five following cases.

- **Fanout only delay** - Only loading capacitance is added to the net delay values.
- **Layout only delay** - Only the delay due to layout (capacitance and possibly resistance) is added to the net delay values.
- **Expected layout delay** - An expected delay due to layout is calculated based on the net loading and this value is added to the net.
- **Fanout and layout delay** - Both actual layout delay and fanout delay are calculated and added to the net.
- **Fanout and expected layout delay** - Expected layout delay and fanout delay are calculated and added to the net.

7.3.2 RUNNING THE PROGRAM

The program is invoked by using the following command line and switch arguments. Direct invocation by typing in the complete command line will be rather lengthy, so it is suggested that a shell program be created which invokes the program with the site-standard switch values. Switches enclosed in brackets "[ ]" are optional. Switch order is not important, but the command must be submitted on a single line and must include the design root, component library pathname, and the name of the semi-custom array that the design will be implemented on.

```
AMI_ADD_DELAY design_root component_library semi-custom_array
                    [-log logical_design_file]
                    [-phy physical_design_file]
                    [-t technology_file]
                    [-v operating_voltage]
                    [-temp operating_temperature]
                    [-syn synonym_file]
                    [-cap capacitance_file]
                    [-ddf DDF_file]
                    [-best|worst]
                    [-fo|lo]
                    [>output_file]
```

- DESIGN_ROOT - This is the name of the root component directory. This directory must contain the logical and physical design files.

- COMPONENT_LIBRARY - This is the full pathname or link to the component library used to create the logical design specified in the first argument. There will generally be one component library for each standard cell or gate array family.

- SEMI-CUSTOM_ARRAY - This is the name of the array the logical design will be implemented on. This argument selects the delay constants and equations specific to each design.

- LOGICAL_DESIGN_FILE (OPTIONAL) - This is the name of the design file which feeds the Mentor analysis tools. This design file is updated with the additional delay. The .EREL extension must be included in the design file name. (Note - The logical and physical design file may be the same.) If the logical design file is not specified, DESIGN.EREL is used.

- PHYSICAL_DESIGN_FILE (OPTIONAL) - This is the name of the design file which feeds the physical layout tools. This design file is read to obtain net names and parameters used to calculate the additional delay. The .EREL extension must be included in the design file name. (Note - The logical and physical design file may be the same.) If the physical design file is not specified, COMP.EREL is used and may be obtained by using the EXPAND_COMP shell script.

- TECHNOLOGY_FILE (OPTIONAL) - This is the name of a user provided data file containing constants used in delay calculation. These constants include pfs_per_load, default krise and kfall values, and expected layout information. See section 3.2 for the file format. If the technology file switch is not used, the technology file for the specified semi-custom array is used.

- OPERATING_VOLTAGE - This is the operating voltage (in volts) for the circuit. Legal values are defined in the accompanying technology file. Values outside the legal range will result in an error message and program termination.

- OPERATING_TEMPERATURE - This is the operating temperature (in degrees Centigrade) for the circuit. Legal values are defined in the accompanying technology file. Values outside the legal range will result in an error message and program termination.

- SYNONYM_FILE - This file contains Mentor net names and the associated synonym used to feed the physical layout tools. See section 4.

- CAPACITANCE FILE - This file contains the net names and the associated capacitance (and optionally resistance) extracted from physical layout. See section 5.

- DDF FILE - This is a standard DDF output file generated from the Cadisys data base containing net routing information. It can be read to extract the layout capacitance and resistance.

- -FO - The presence of this switch indicates that only fanout delay will be calculated and added to the design file.

- -LO - The presence of this switch indicates that only layout delay will be calculated and added to the design file. Note that the layout delay can be either expected or actual based on the presence or absence of the DDF or capacitance file.

- -BEST - The presence of this switch indicates that the shortest possible delays for the given technology and process are to be used. In the absence of this switch and the "-WORST" switch, nominal delay values are assumed.

- -WORST - The presence of this switch indicates that the longest possible delays for the given technology and process are to be used. In the absence of this switch and the "-BEST" switch, nominal delay values are assumed.

- OUTPUT_FILE - The program generates a report of the original and added delay for each net which is written to standard output. This can be redirected to a file with the >.

7.3.3 THE DELAY EQUATION AND ITS PARAMETERS

The delay equation currently implemented in the add_delay program is designed for two layer metal CMOS arrays and considers capacitive delay due both to fanout and interconnect distance. The add_delay program has the capability to read and use other parameters including resistance or number of polysilicon underpasses on a net. See the Reference Manual for instructions on how to modify the program to include these parameters. The following equation is used for computing both the rise and the fall delay. K represents either KRISE or KFALL. St and Sv are scalars calculated from the temperature and voltage specified at runtime. Sp is a scalar that is related to the technology and the choice of best, worst or nominal case analysis.

added_delay = K{(metal_capacitance) + [(#loads/size)(pfs_per_load)]}StSvSp

The following two sections explain the parameters and how they are extracted. NOTE that some parameters are extracted but are not currently used by the delay equation above.

7.3.3.1 Parameters From the Design File

Parameters extracted from the design file are component or net specific. For instance, the number of loads on a net must be obtained from the design file instead of the technology file. Following is a list of the properties considered and how they are interpreted.

The following parameters are extracted from the physical design file.

- PINTYPE property - The pintype property is used to determine the sense of the signal. *Owner is PIN, Value is IN, OUT, or IO.*

- KRISE or KFALL property - This property is extracted from the source pin of the net in the physical design file. Its value is used to convert capacitance into nanoseconds. If this property is not present on the source pin of the net, the default value is used from the technology file. If more than one source pin (O or IO) is present, the largest value of Krise or Kfall is used. *Owner is PIN, value is double precision real.*

- LOAD property - The LOAD property can be attached to any pin and is used to total all the loads on a particular net. The LOAD property value is a real number, so fractional loads can be reflected. If this property is not present on a pin, the default value is 1 for IN or IO pins and 0 for all output pins. *Owner is PIN, value is double precision real.*

- SIZE property - The size property is used to determine the number of driving gates or the relative size of the driving gate. The SIZE property is a real number, so fractional transistor sizes can be reflected. If this property is not present on an output pin, a value of 1 is assumed. Size properties attached to input pins are ignored. *Owner is PIN, value is double precision real.*

- NET.TID property - This property is used to store netname synonyms in the design file. The Namechecker program available with some netlisters creates this property in the design file. If the Namechecker program is not available, this property is not meaningful. *Owner is NET, value is string.*

The following parameters are updated in the logical design file.

- RISE and FALL properties - The RISE and FALL values are used by the Mentor analysis tools to reflect circuit delays. The add_delay program adds delay to any existing values on all the output pins of the net. For example, if the value of the RISE property on the net source pin is "1 2 3" and the additional computed delay is 3.45, the new RISE property will be "4.45 5.45 6.45".

7.3.3.2 Parameters From the Technology File

Parameters extracted from the technology file are designed to be component or net independent. The format of the technology file is as follows:

1. Any line with a # in column is considered a comment and is ignored by the program.

2. Any non-comment line has the form

   <parameter name><delim.><real>{<delim.><integer>}

   where parameter name is one of the parameters in the following list, delimiter is ' ', ',', or '=', real is a double precision real value, and an optional integer value can be included for expected_net parameters.

Currently the possible parameters are as follows:

- **KRISE_DEFAULT** - Contains the default KRISE value for a net if one is not found in the design file.
  KRISE_DEFAULT real_number

- **KFALL_DEFAULT** - Contains the default KFALL value for a net if one is not found in the design file.

KFALL_DEFAULT real_number

- **PFS_PER_LOAD** - Contains a constant reflecting the capacitance due to one load.
  PFS_PER_LOAD real_number

- **EXPECTED_NET_CAP** - Contains the expected layout capacitance due to routing based on the net loading information. (Integer is the number of loads and the real number is the capacitance).
  EXPECTED_NET_CAP real_number integer

- **TEMPERATURE_MAX, TEMPERATURE_MIN, VOTAGE_MAX, VOLTAGE_MIN** - These parameters set the legal values for the voltage and temperature arguments specified with the "-v" and "-temp" switches. The limits are inclusive (min <= value <= max).

- **PROCESS_BEST, PROCESS_WORST** - These parameters specify Sp for the best and worst process case.

- **TEMPERATURE_COEF, VOLTAGE_COEF** - These parameters are used to describe a polynomial of arbitrary order to calculate St and Sv from the user provided values for voltage and temperature. To describe, for example, the polynomial 2.1(V**2) - 1.005V + .25, the technology file would contain:

```
VOLTAGE_COEF=2.1 2;
VOLTAGE_COEF=-1.005 1;
VOTAGE_COEF=0.25 0;
```

For standard cell applications, the following parameters may also be included:

- **STANDARD CELL** - For Standard cell applications set this value to "TRUE". All other values are ignored. If the standard cell option is selected, all of the remaining parameters must be defined.

- **MAX_DIE_SIZE, MIN_DIE_SIZE** - The maximum and minimum die edge size allowed for the standard cell technology described. Die size units are arbitrary but must be consistent with units used in subsequent parameter definitions.

- **DIE_SIZE_INTERVAL** - The die size interval between standard cell table entries. Estimated die sizes from gate count and gate area equations will be rounded *UP* to the closest legal die size.

- **COMBINATIONAL_GATE_DENSITY** - Estimated average density of combinational logic in gates per unit area.

- **SEQUENTIAL_GATE_DENSITY** - Estimated average density of sequential logic in gates per unit area.

- **COMBINATIONAL_PERCENTAGE** - Estimated percentage of combinational logic usage. The sum of this value and sequential_percentage should be 1.0.

- **SEQUENTIAL_PERCENTAGE** - Estimated percentage of sequential logic usage. The sum of this value and combinational_percentage should be 1.0.

- **INTERCONNECT_FACTOR** - Estimated proportion of interconnect area area to component area.

- **PERIPHERAL_BUS_SIZE** - Width of external bus connections to the die.

- **SCRIBE_LINE_SIZE** - Width of scribe lines.

- **CORE_LIMITED_PAD_SIZE** - Width of pads dedicated to input and output buffers.

- **STANDARD_CELL_TABLE** - Full pathname to standard cell interconnect capacitance matrix. This file contains average capacitances for nets based upon die size and the number of pins on each net. The effect of these capacitances supercedes capacitances defined with the EXPECTED_NET_CAP parameter.

7.3.4 NET NAMES AND SYNONYMS

Since the Mentor-generated netnames are not always legal in all layout software, synonyms are sometimes required to feed the layout. (Synonyms are not required for Cadisys). There are two ways to keep track of the synonym information. Note that add_delay can only read synonyms generated by another program, not generate new synonyms. The two method of reading synonyms are described in tyhe following sections.

7.3.4.1 External Synonym File

Mentor supplied netlisters have the capability of generating a synonym file used to cross reference Mentor-generated names to Netlister-generated legal names. If this netlister is used to feed physical layout tools, a method must be supplied for keeping track of the new name when delay information is fed back to the system. To handle this condition, the add_delay program can read netlister-generated synonym file and perform the cross reference. This

file is of the form:

<Mentor netname>' '<synonym used to feed layout>

7.3.4.2 Synonyms in the Design File

An alternate method of handling synonyms is possible by storing new synonyms in the design file. The program which creates the synonyms and stores them into the design file is called the Namechecker, and is available with some netlisters. If synonyms in the design file must be considered by the add_delay program, the argument to the -SYN switch is the physical design file name. Add_delay then extracts the value of the NET.TID property to use as the net synonym name.

7.3.5 LAYOUT DELAY

Layout delay parameters can read in from one of two files (but not both files in the same run). Currently the delay parameters extracted from these files include capacitance and possibly resistance. The following sections discuss the details of these files.

7.3.5.1 The Capacitance File

The capacitance file is designed primarily for delay values coming back from layout software other than Cadisys. This is an Ascii file with the following format:

<Net name><' '><net capacitance(real)>{<' '><net resistance(real)>}

Note that the netname will be a synonym if synonym capability was used to create the netlist used to feed the layout software. Both net capacitance and net resistance are double precision real numbers.

7.3.5.2 The DDF File

The DDF file is the standard input and output interface to the Cadisys layout software. On input to the layout software, logic connectivity and preplacement is included and on output connectivity, placement, routing, and net parameter information is included. See the Cadisys documentation for instructions on how to run this program.

7.3.6 LIMITATIONS

There are two limitations that the user of this program should be aware of involving the delay model used and use of add_delay with behavioral language models.

The current model used to calculate delay is done on a per net basis in which specific pin to pin delays are not handled. This is a reasonably accurate model when only net capacitance is considered, but can give pessimistic delay values when various pin to pin paths have differing numbers of polysilicon underpassses. A modification of the program could be made to add different delay values to each of the net's load (pintype IN) pins to handle the individual pin to pin delays.

Behavioral Language models can be written so that any delays on their symbol pins are ignored by the simulator. They can also be written so that the delays are considered in simualtion. If a BL model ignores the delays on symbol pins, the added delay will not be available since add_delay adds the RISE and FALL values to the pins of the symbols.

7.4 Designing for Testability

This is a reprint from the GOULD AMI 3μ Single and Double-Metal CMOS Gate Arrays Databook.

6.3 Designing For Testability

Design for testability involves three main concepts:

Initialization - The ability to set the state of the device under test (DUT) at the beginning of the test cycle.

Controlability - The ability to set internal nodes of the DUT in a given state by external stimulus.

Observability - The ability to observe at a DUT output, the state of an internal node.

Here are some suggestions/solutions to overcome the implementation problem for each testability concept.

1. Initialization - It is essential to set the state of the DUT at the beginning of a test cycle. This reduces both testing time and the risk of invalid failures of good devices. If the DUT cannot be initialized, then a routine known as 'match mode' must be used on the Sentry to apply a pattern to the DUT until it 'matches' a pattern supplied by the test engineer. When a match occurs, the DUT is in a known state. There are two types of resets used in the implementation of Semicustom products. These are the reset pin and reset by a combination of stimuli on one or several other pins.

    a. Reset pin - Designing a reset pin into the circuitry is the recommended way to guarantee initialization. This pin is connected internally to all of the storage elements and provides an immediate initialization of the DUT. The reset signal is set by the test system to intialize the DUT and start the test sequence. A disadvantage of the use of a reset pin is that it requires an additional pin and reset; it adds a substantial amount of circuitry to the device. The reset pin becomes especially costly when the designer has to go into the next larger pin count package to provide the reset pin. However, these disadvantages are often outweighed by the advantages of the fastest possible reset sequence and the easiest initialization method. The tester only needs to send only pulse to the DUT rather than a whole sequence of reset commands.

b. Reset by a Combination (Sequential reset) - The variety of combinational rest techniques is limited only by the imagination and creativity of the designer. Two widely used techniques are discussed here. The first uses a reset pin, but in order to save some internal circuitry, the internal reset pin goes only to a limited number of devices. A complete reset of the DUT requires a certain number of clock cycles during which known signals are propagated through the entire circuit. The implementation of this technique starts by the the identification of serial synchronous delay lines and the study of the signals required to initialize them in the chosen state. The length of the reset sequence is the length of the longest such line. The second example avoids using a additional reset pin, generally for package considerations. Here the reset sequence must be initialized by a combination of the input signals that cannot occur during the normal opperation of the circuit. The advantage of the above techniques are that they:

   i. use standard circuitry

   ii. allow the designer the flexibility of deciding how to implement the reset function

   iii. allow the test engineer to precondition the DUT into a known state at a known time

2. Controlability - The first element of controlability is the ability to unconditionally reset the DUT. Another important controlability technique is the ability to externally place the DUT in any state required to simplify testing. Several apporaches can be used. One of the most popular methods of increasing controlability is the implementation of a 'test mode'. This can be achieved externally and requires a number of exclusive combinations of the inputs at least equal to the number of states in the test mode. This can also be achieved by internally storing the test mode sequence of events and starting it by the reset pin.

3. Observability - Observability of the outputs defines how easily the internal behaviior of the device can be observed on the external pins. Numerous techniques exist that improve the observability of a given circuit, and almost all of them are based on the tranferring of the internal states to an external pin. The main limitation is the pin count or the extra circuitry required. However, the access to normanlly inaccessible internal nodes can greatly increase the test coverage while at the same time, reduce the overall test time.

3UM SINGLE METAL GATE ARRAY DATA SHEETS

The Gould-AMI 3um Single Metal Gate Array Macro Library is a collection of high performance digital integrated circuit building blocks. It is intended for use by logic, circuit, and MOS IC designers. Thus, anyone familiar with standard logic design methods can successfully design a gate array using these macros.

The 3um Single Metal Macro Cells are implemented with a p-well, CMOS process. They are intended primarily for 5 Volt (+10%) operation but will operate at voltages as low as 2.5 volts with reduced performance. Below is a description of the data and terms used in the data sheets.

- **PRE-ROUTED MACROS** are the simpler functional blocks such as the basic gates, latches, and flip-flops. Since these macros have a pre-defined layout they can be accurately characterized for propagation delays.

- **SOFTWARE MACROS** are macros which incorporate several pre-routed macros to form more complicated functional blocks such as a counters or shift registers. These macros are not pre-defined with respect to layout, so the automatic placement and routing software maintains optimum layout flexibility. Since layout is not pre-defined, estimates of propagation delays through these macros are arrived at by adding the delays of the individual pre-routed macros used in the software macros.

- **THE CELL NAME** for each functional macro is displayed in the upper right hand corner of each data sheet. This is the same name that appears in the Bolt invocation.

- **THE LOGIC SYMBOL** for each macro is displayed as it appears in the gate array library.

- **A TRUTH TABLE** description of the logical functions for the cells is provided.

- **THE TABLE OF INPUT LOADING** gives the number of equivalent unit loads for each logical input of the pre-routed macros.

- **THE EQUIVALENT GATE COUNT** is the number of equivalent two input NAND gates in each cell.

- **BOLT SYNTAX** is the invocation syntax of each macro cell. Notice these statements always end with a semi-colon, and the order of these inputs and outputs must be maintained.

- **THE SWITCHING CHARACTERISTICS** give the propagation delay as a function of unit load. The unit load is the capacitance associated with a gate pair (i.e. the gate capacitance of an N-channel and a P-channel core transistor), plus an associated metal interconnection capacitance. All data is given for typical process, temperature, and power supply conditions. Both low-to-high transitions are shown. To find delays under other operating conditions derating curves must be used.

- **THE PROPAGATION DELAY EQUATION** is used for calculating propagation delays when the load is different from those explicitly given in the table. The intrinsic propagation delay, tdx, is used when there is no output loading, and ktdx, is a capacitive multiplication factor. As an example consider the two input NAND Gate with a fan-out of 5. To calculate the propagation delay for the high-to-low transition use tdx = Ø.1 and ktdx = 1.8 so,

$$tPHL = Ø.1 + (1.8)(.25)(5) = 2.35 \text{ ns.}$$

- **A LOGIC SCHEMATIC** is shown for the more complicated pre-routed macros and the software macros not shown.

- **POWER SUPPLY MACROS** (PPØ1,PPØ2) must be shown on each circuit, while no connections to these macros are necessary one macro must be shown per power or ground pin.

- **DERATING CURVES** are provided to account for changes in propagation delay as process, temperature and power supply vary.

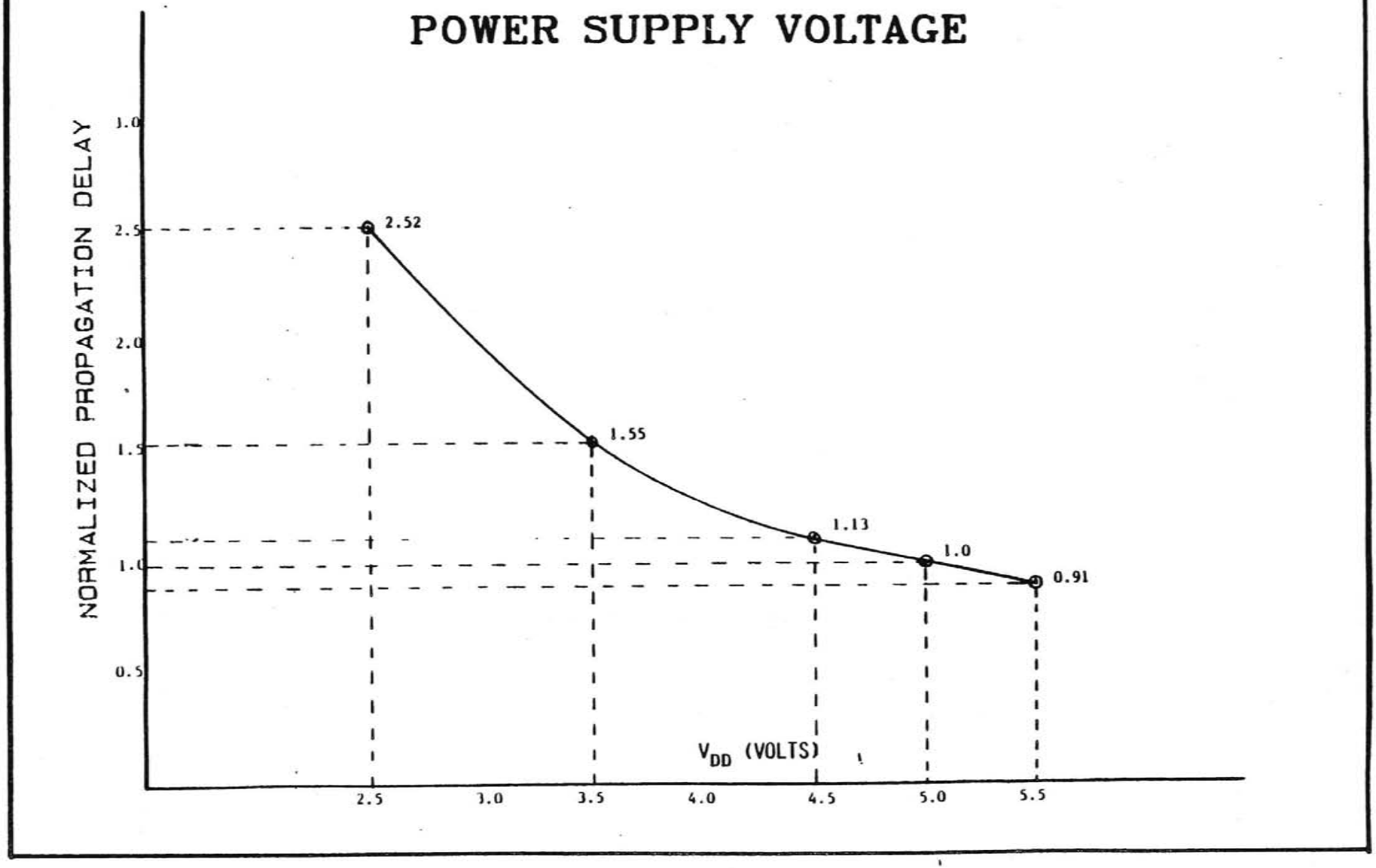
3u SINGLE METAL GATE ARRAY MACROS

1.	IN01	Single Pre-Routed Inverter
2.	IN02	Two Pre-Routed Inverters in Parallel
3.	IN03	Three Pre-Routed Inverters in Parallel
4.	IN04	Four Pre-Routed Inverters in Parallel
5.	IN05	Five Pre-Routed Inverters in Parallel
6.	NA02	Pre-Routed Two Input Nand Gate
7.	NA03	Pre-Routed Three Input Nand Gate
8.	NA04	Pre-Routed Four Input Nand Gate
9.	NA05	Pre-Routed Five Input Nand Gate
10.	NO02	Pre-Routed Two Input Nor Gate
11.	NO03	Pre-Routed Three Input Nor Gate
12.	NO04	Pre-Routed Four Input Nor Gate
13.	NO05	Pre-Routed Five Input Nor Gate
14.	AA02	Pre-Routed Two Input And Gate
15.	AA03	Pre-Routed Three Input And Gate
16.	AA04	Pre-Routed Four Input And Gate
17.	OR02	Pre-Routed Two Input Or Gate
18.	OR03	Pre-Routed Three Input Or Gate
19.	OR04	Pre-Routed Four Input Or Gate
20.	EN01	Pre-Routed Exclusive Nor Gate
21.	EO01	Pre-Routed Exclusive Or Gate
22.	DF01	Pre-Routed D Flip Flop
23.	DF02	Pre-Routed D Flip Flop With Asynchronous Active Low Set
24.	DF03	Pre-Routed D Flip Flop With Asynchronous Active Low Reset
25.	DF04	Pre-Routed D Flip Flop With Asynchronous Active Low Set And Active Low Reset
26.	DF05	Pre-Routed D Flip Flop With Asynchronous Set
27.	DF06	Pre-Routed D Flip Flop With Asynchronous Reset
28.	DF07	Pre-Routed D Flip Flop With Asynchronous Set and Reset
29.	DL01	Pre-Routed Latch with Q and QN
30.	DL18	Pre-Routed Latch With Set and Reset
31.	DL19	Pre-Routed Latch With Set
32.	DL1A	Pre-Routed Latch With Reset
33.	DL1B	Pre-Routed Latch With Active Low Set
34.	DL1C	Pre-Routed Latch With Active Low Reset
35.	DL1D	Pre-Routed Latch With Acive Low Set and Reset
36.	AN01	Pre-Routed And Nor Gates
37.	AN02	Pre-Routed And Nor Gate
38.	AN03	Pre-Routed And Nor Gate
39.	AN04	Pre-Routed And-Nor-Nor Gate
40.	ON01	Pre-Routed Or-Nand Gate
41.	ON02	Pre-Routed Or-Nand Gate
42.	ON03	Pre-Routed Or-Nand Gate
43.	ON04	Pre-Routed Or-Nand-Nand Gate
44.	AO01	Pre-Routed And-Or Gate
45.	RS00	Pre-Routed Set-Reset Nand Gate Latch
46.	RS01	Pre-Routed Nor S R Latch
47.	IT02	Pre-Routed Inverter Driving A Transmission Gate

48. II11 Pre-Routed Clock Driver With A Single Inverter Followed By A Single Inverter
49. II12 Pre-Routed Clock Driver With A Single Inverter Followed By Two Inverters in Parallel
50. II13 Pre-Routed Clock Driver With A Single Inverter Followed By Three Inverters In Parallel
51. II21 Pre-Routed Clock Driver With Two Inverters In Parallel Followed By A Single Inverter
52. II22 Pre-Routed Clock Driver With Two Inverters In Parallel Followed By Two Inverters In Parallel
53. II31 Pre-Routed Clock Driver With Three Inverters In Parallel Followed By A Single Inverter
54. IO05 Tri-State Input-Output Buffer
55. IIF3 Pre-Routed TTL Lever Translator
56. IIF0 Pre-Routed Inverting Schmitt Trigger
57. IB13 Input Pad With Protection Diode
58. IB14 Input Pad With Protection Diode With P-Channel
59. IB15 Input Pad With Protection Diode With N-Channel
60. OB03 CMOS Output Buffer
61. OB0D Tri-State Output Buffer
62. OB07 Open Drain Output Buffer
63. PP01 VSS Ground Pin
64. PP02 VDD Power Pin

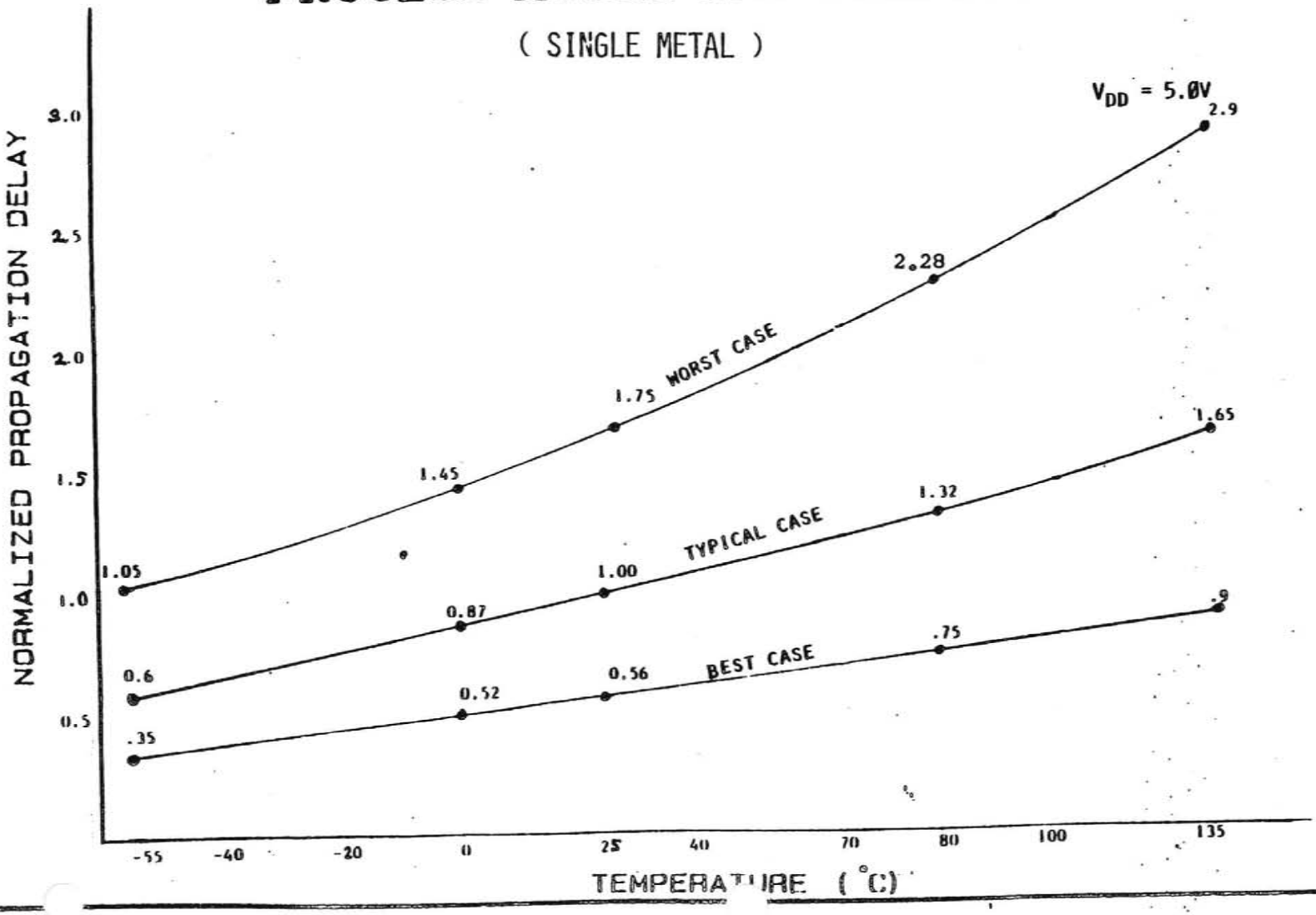
Power Supply Voltage Derating Curve
Process And Temperature Derating Curve

NORMALIZED DELAYS VS. POWER SUPPLY VOLTAGE



NORMALIZED DELAYS VS. PROCESS MODEL AND TEMPERATURE

(SINGLE METAL)



Secrets of CMOS gate-array delay specs

Understanding how process parameters, temperature, fanout, and power-supply variations affect the operation of CMOS gate arrays is essential for design success

Michael Friedman
Senior Applications Engineer
American Microsystems Inc.
Santa Clara, CA

The recent proliferation of gate-array vendors and the lack of industry standards for gate arrays can easily mislead the potential customer. Especially disturbing are the inadequate ways that propagation delays are often specified. Some vendors specify merely ill-defined, so-called typical values, which could mean almost anything—often a value that is just a nominal starting point from which deratings should be made. Others may give only maximum delays at some nominal power supply and temperature, and an often unspecified load.

But circuits should always be designed to meet electrical performance requirements under the most stringent conditions. Accordingly, meaningful propagation-delay data should include the effects of the maximum expected operating temperature, the load (or fanout) on the gate (or other logic element), and the minimum power-supply voltage—in other words, worst-case operating conditions. Since CMOS represents the bulk of today's semicustom gate-arrays (see "CMOS revisited"), a 4-bit binary counter gate-array design (see Fig. 1) is analyzed with the help of the manufacturer's properly supplied propagation-delay data.

In the counter, propagation delays of a series of logic circuits determine the critical timing of signals transferred among the D-type flip-flops (see Fig. 2). The most critical path for this counter is from the output of the least significant bit (LSB) flip-flop, D1BAR, to the input of the most significant bit (MSB) flip-flop, D4IN. The signal must propagate along this path, and set up flip-flop FF_4 in one clock cycle. The path is shown isolated in Fig. 2a.

Note that inverter A, the first logic element in the critical path, has a fanout of four, or a load of

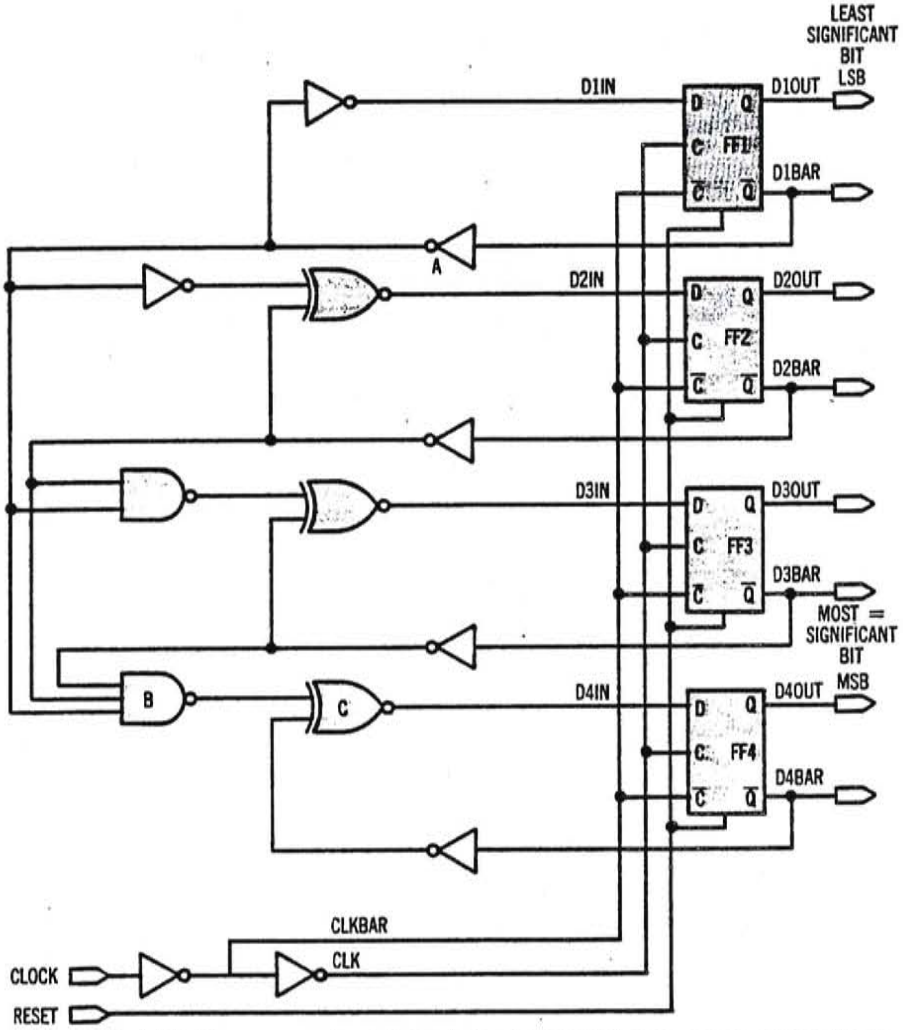
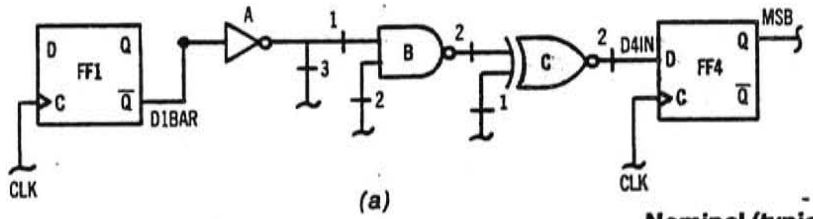
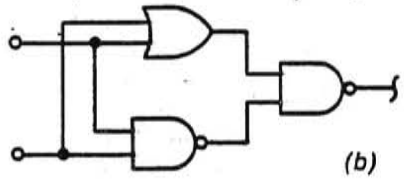


Fig. 1. A 4-bit binary counter fabricated with CMOS gate-array elements is analyzed for its critical propagation-delay paths. The worst-delay in the circuit termines the circuit's maximum counting speed.





four logic elements, which is indicated by vertical line number three, plus an input to the next element, a three-input NAND gate. This gate drives an XNOR gate's inputs, which constitutes a load of two because the gate is implemented with several simpler gates (see Fig. 2b). Also, the FF_4 flip-flop presents a four-unit load.

When each element's average propagation delay (see Fig. 2c)—which accounts for the required load at 25°C operation—and a 5-V power supply are added, the overall delay of 13.50 ns translates into a maximum allowable clock frequency of 74.07 MHz. This calculation, however, is an oversimplification. Dramatic changes in propagation delay occur as process, temperature, and power-supply vary—common events in the real world.

To quickly evaluate changes in propagation delay as process temperatures, and power-supply voltage vary, American Microsystems Inc., of Santa Clara, CA developed a set of derating curves for its gate-array products. These are based on data gathered from measurements made on large numbers of production samples of logic elements. The measurements were corrolated and normalized into two derating plots—one for delay as a function of process and temperature (at 5-V power supply) and the other as a function of power-supply voltage (at 25°C).

As an example of how the derating curves are used, reconsider the critical path of the four-bit counter, where the propagation delays provided were at a supply voltage, V_{DD}, of 5 V at 25°C and totaled 13.50 ns.

Nominal (typical) Propagation Delays for Individual Logic Elements

DEVICE	LOAD	TYPICAL PROPAGATION DELAY (ns)
Clock to Q	1	3.36
Inverter	4	1.90
3-Input NAND	2	2.51
Exclusive NOR	4	3.21
Set-up (MSB)		2.52

(c)

Fig. 2. The counter's most critical path is from CLK input of the least significant-bit flip-flop to the output setup time of the most-significant-bit flip-flop(a). Because an XNOR is made up of several gates (b), it presents a standard fanout load of two on each of its inputs. Similarly, FF4's D input represents four loads, as noted in the propagation-delay table (c).

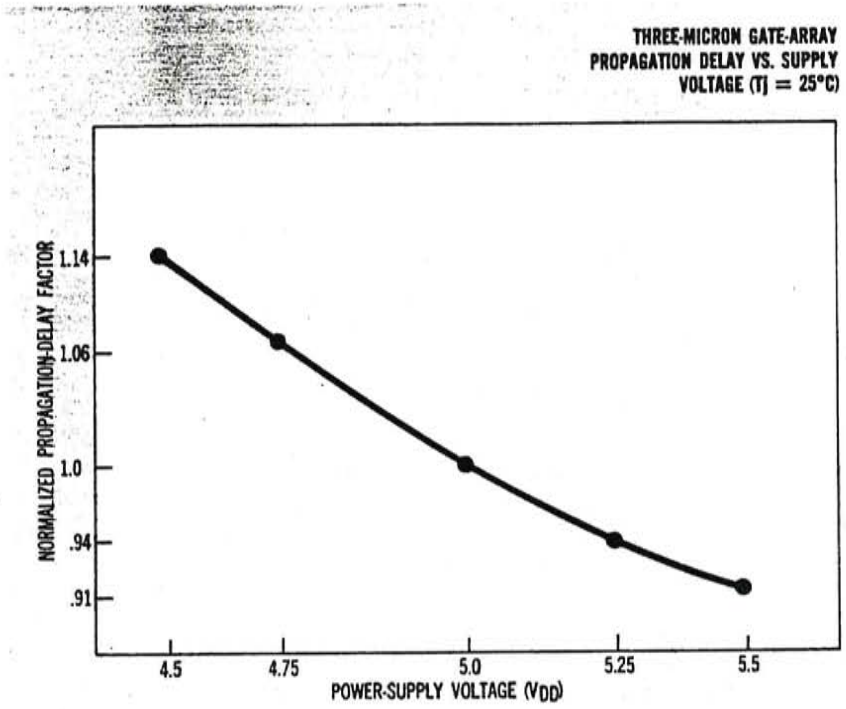


Fig. 3. The normalized propagation-delay derating curve shows that delay increases as the power-supply voltage, V_{DD}, decreases.

But if V_{DD} is only 4.5 V, what is the delay along this path? Figure 3 can provide the ratio of the propagation delay at 4.5 V normalized to 5 V as follows:

$$\frac{V_{DD}\text{ at }4.5\text{ V}}{V_{DD}\text{ at }5.0\text{ V}} = \frac{1.14}{1.00} = 1.14$$

Accordingly, the new propagation delay is derated to $t_{PD} = 1.14 \times 13.50 = 15.39$ ns, and the new maximum operating frequency is 64.98 MHz.

But an operating temperature of 25°C is unrealistic. Moreover, since so-called typical process parameters are seldom clearly defined and a reliable design is desired, derating to worst-case process parameters should be the next step. For the propagation delay of the worst-case process parameters, and a maximum temperature of, say, 70°C, the normalized propagation delay ratio is shown in Fig. 4:

$$\frac{\text{Worst case at }70°\text{C}}{\text{Typical case at }25°\text{C}} = \frac{2.3}{1.0} = 2.3$$

With this additional derating factor, the new propagation delay becomes $t_{PD} = 2.3 \times 15.39 = 35.40$ ns, and the maximum operating frequency is 28.25 MHz. Under these worst-case conditions, the maximum operating frequency is more than 2.5 times slower than under typical conditions (74.07 MHz versus 28.25 MHz.

The nominal propagation delays in Fig. 2c came from the manufacturer. AMI supplies data sheets for all logic elements (called functional macros), compiled as a gate-array cell library in a form similar to Fig. 5.

The data includes a table of propagation delays versus capacitive loading. Data for fanouts range from one to four loads and often to six loads. Additionally, data is supplied so that propagation delays can be calculated for fanouts that exceed the provided range:

Propagation delay (t_{PD}) = tdx + ktdx•C_L

tdx = intrinsic propagation delay with zero fanout

ktdx = capacitive multiplication factor

C_L = load capacitance (typically 0.25 pf per gate input)

This equation results from the fact that the dynamic behavior of a CMOS gate can be modeled by a series RC networks, in which the resistance is simply the on-impedance of the p- or n-channel transistors and the output load is a function of the fan-

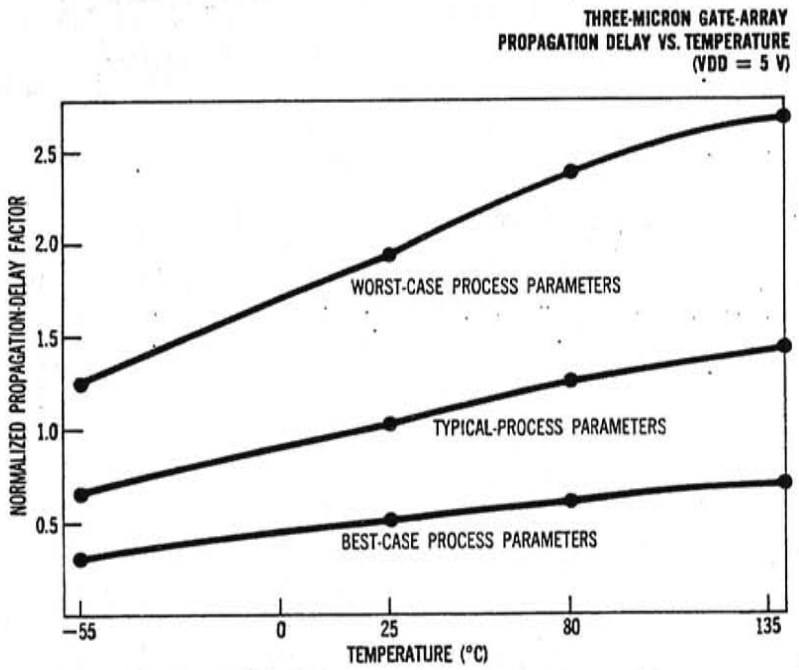


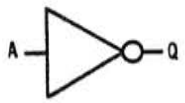
Fig. 4. Increasing temperature also derates (increases) a gate-array element's propagation delay. Of course, the worst-case process curve is usually used when checking a design.

out's load and wiring capacitance.

Note in Fig. 5 that the propagation delays differ for rising and falling signal edges. To explain this phenomenon, consider the output of a CMOS circuit which typically is like that of an inverter (see Fig. 6). Under quiescent conditions, one of the transistors is cut off, whereas the operating point for the complementary transistor lies in its linear, or nonsaturated, portion of the current-voltage characteristic curve. But it does so at zero current (P_1) because the first transistor is cut off.

Thus when the n-channel transistor is off and the p-channel is on, the output load capacitance charges to the supply-rail voltage V_{DD}. Should the gate voltage, V_G, suddenly be raised to V_{DD}, the operating point will first move to P_2 as current flows in the n-channel transistor. This happens because V_{DD} is still across the n transistor, since capacitor C cannot change its voltage instantaneously. Simultaneously, the p-channel transistor turns off, and the capacitor starts discharging. When the voltage across the drain-source terminals of the turned-on n-channel device reaches a point where $V_{DS} = V_G - V_T$ (V_T is the

INVERTER



A	Q
1	0
0	1

INPUT LOADING: 1-STANDARD LOAD (0.25 pF)

DELAY VS CAPACITANCE

FANOUT (pF)	1 (.25)	2 (.5)	3 (.75)	4 (1.0)	Tdx	Ktdx
T = 25°C	Typical Propagation Delays (nS)					
rise — 0 to 1	3.5	5.1	6.6	8.1	2	6.12
fall — 1 to 0	0.9	1.7	2.6	3.5	0	3.47
(logic levels)						

Fig. 5. A gate-array manufacturer should supply comprehensive data sheets for each logic element in its library. While this example may not be current data, it is representative of the kind of information that should be supplied.

CMOS REVISITED

The basic logic circuit in CMOS technology consists of one p-channel and one n-channel transistor connected in series between the V_{DD} and the V_{SS} power-supply rails (Fig. a). The n-channel transistor is fabricated in a p-well substrate and connected to the most negative supply in the circuit; the p-channel transistor is connected to the most positive supply.

When the gate voltage, V_G, of either transistor is equal to its substrate voltage, the device turns off. But it becomes conductive as the gate voltage moves away from the substrate voltage (to the opposite polarity) and exceeds the threshold voltage, V_T, to form a conducting channel between drain and source.

The transistor's geometry dopant concentration, gate-oxide thickness, and substrate conductivity determine the threshold voltage. An increase in either the thickness of the oxide over the channel region or the impurity concentration in the channel region, increases the gate voltage required to form a conducting channel. An increase in threshold voltage lowers both the transistor's speed and drain-source current. Thus a manufacturer's goal is to keep V_T as low as possible.

In the conductive region, initially, the current varies almost linearly as the voltage between the transistor's source and drain, V_{DS}, increases (Fig. b). But soon, when $V_{DS} + V_T = V_G$, the electric field developed in the oxide near the drain end of the channel is insufficient to attract additional current carriers, and the channel becomes pinched-off, or saturated. The drain current thereafter remains almost constant with increasing drain voltage.

Accordingly, the three operating regions of a MOS transistor and the current, I_{DS}, that flows are determined by the relationships among V_G, V_{DS}, and V_T:

- **Cut-off**
  $|V_G| < |V_T|$
  $I_{DS} = 0$
- **Non-linear**
  $0 < |V_{DS}| < |V_G - V_T|$
  $$I_{DS} = \frac{m \epsilon_O \epsilon_{OX} W}{t_{OX} \quad L} \left[(V_G - V_T) V_{DS} - \frac{V_{DS}^2}{2} \right]$$
- **Saturation**
  $|V_T| < |V_G| < |V_{DS}|$
  $$I_{DS} = \frac{m \epsilon_O \epsilon_{OX} \quad W}{2 t_{OX} \quad L} (V_G - V_T)^2$$

m = average surface mobility of holes or electrons
t_{OX} = thickness of oxide over the channel
ϵ_{OX} = permittivity of the oxide
ϵ_O = permittivity of free space
L = channel length W = channel width

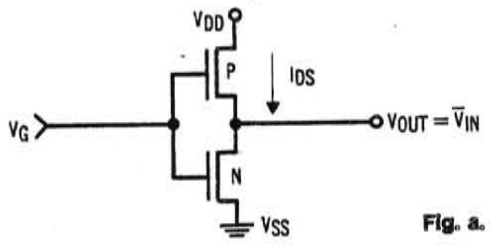


Fig. a.

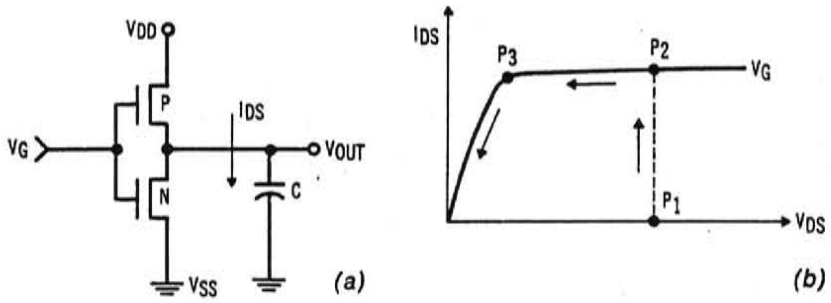


Fig. 6. The output of a CMOS inverter with its capacitive load (a) is typical of all CMOS logic elements, as is the switching cycle (b) through which it passes.

transistor's threshold voltage) the transistors enters the nonsaturated region, P_3, until the capacitor is fully discharged.

The low-to-high transitions follow a similar pattern, but the delays are longer than for the high-to-low transitions just described. The main factors affecting the switching-cycle speed of a CMOS output circuit are the amount of charge stored in the output capacitance and the drain-to-source current-voltage characteristics of the specific switching transistor.

Clearly then, it must be the different n- and p-channel characteristics that account for the different propagation speeds; the load switched is the same in both cases. The conductance of n-doped silicon is about 2.5 times that of equally doped p-type material. Therefore, internal capacitances (and leakage current) are not equal for transistors having the same on-resistance. The p-channel device must be about 2.5 times larger to equalize the on-resistance, and correspondingly, it also has more capacitance. As a result, the low-to-high switching sequence, when the p-channel turns on is slower because of its larger capacitance.

Thus, with the typical propagation delays corrected for fanout in accordance with the data sheets the manufacturer should supply for each macro function in its library, and a set of derating curves, the worst-case propagation delays over operating temperature, voltage, and process parameter ranges can be realistically evaluated. Then the actual throughput of a prospective semicustom implementation can be reliably predicted before the circuit is committed to silicon. □

The channel length, L, is the tiny spacing between the drain and source regions—in the order of 1 to 5 μm—and is the important geometrical dimension used to characterize FETs and established by the manufacturer's processing capability. A decrease in this dimension increases both I_{DS} and speed and, of course, reduces the die area.

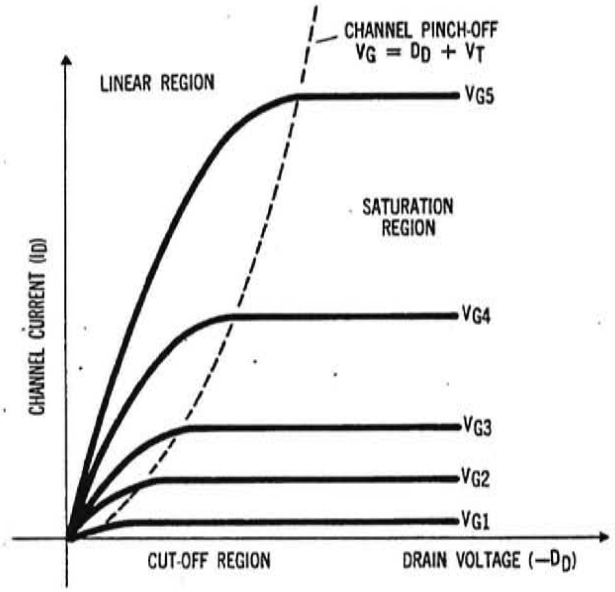


Fig. b. Drain characteristics of a PMOS transistor

Capacitive loading on a transistor slows down switching transition time. Therefore, the increased I_{DS} has an added advantage of enabling the charging and discharging of the capacitance load more quickly, thereby counteracting the slowing effect of the capacitance.

Another processing and speed factor that strongly affects I_{DS} is the mobility (m) of the current carriers in the doped silicon. The mobility is a measure of the average drift speed of the carriers through the silicon lattice. It is heavily dependent on the type of carrier (electron or hole) and the level of doping, and it varies with temperature. Close control of the doping level is important for a uniform, reliable product.

Less controllable is the effect of temperature on speed. The energy levels in the band gap of a semiconductor are temperature dependent. Thus, the threshold voltage also varies with temperature. Generally, an increase in temperature results in a decrease in the absolute value of the threshold voltage which, by itself, would increase switching speed and I_{DS}. However, a rise in temperature also reduces carrier mobility, which then strongly reduces both I_{DS} and speed. Mobility has a dominant effect on device speed, and despite a decreasing threshold voltage, device speed decreases as temperature rises.

Finally, the effects of the power-supply voltage can be established from the supply's influence on V_G. The gate voltage of a CMOS inverter can swing across the full-supply range (from V_{SS} to V_{DD}). Therefore, any increase in power-supply voltage also increases V_G. This change corresponds to an increase in I_{DS} (see equations) and also increases device speed.

Reprinted From ELECTRONIC PRODUCTS Magazine — June 18, 1984
645 Stewart Ave., Garden City, NY 11530 • © 1984 Hearst Business Communications Inc. PRINTED IN U.S.A.

CMOS gate-array circuits yield low-cost programmable logic

For applications requiring reduced power consumption, small size and high reliability, CMOS gate arrays yield devices that compare favorably in cost and performance with programmable array logic.

Saeed Kazmi, *VLSI Technologies Inc, and*
Michael Friedman, *American Microsystems Inc*

When designing logic for fuse-programmable circuits, consider the power and cost savings that CMOS gate arrays can offer. For instance, in a video-controller example, a single array replaces eight programmable-array-logic devices, cuts power consumption by a factor of 30 and reduces by half the controller's cost. For other applications, gate arrays can replace considerably more programmable-logic devices, as long as your speed specifications don't rule out CMOS.

Consider first the tradeoffs between gate arrays and programmable logic. If you desire low system cost, low power consumption, design flexibility and high reliability, then CMOS gate arrays are preferable to fuse-programmable logic arrays (FPLAs) or Monolithic Memories' PAL devices. In cases where high speed, fast turnaround or low volume are the main considerations, bipolar programmable logic is likely to offer the better solution.

PAL circuits and FPLAs are standard, off-the-shelf products, requiring as an initial investment only a PAL/FPLA programmer. Once you have simulated the logic and implemented it in the proper form, the time required to customize a chip by blowing fuses is relatively short.

For gate arrays, you must send out for masks and wait several weeks for prototypes. Advances in CAD software, however, are continually reducing development time. Manufacturers currently offer turnaround times, from logic diagrams to working prototypes, of six to eight weeks. The turnaround time includes logic conversion, logic simulation, layout, metal-mask generation, processing and testing. Unlike fuse-programmable devices, whose logic must be transformed into AND/OR logic, gate arrays have uncommitted transistor pairs, and the logic can be implemented in its original form.

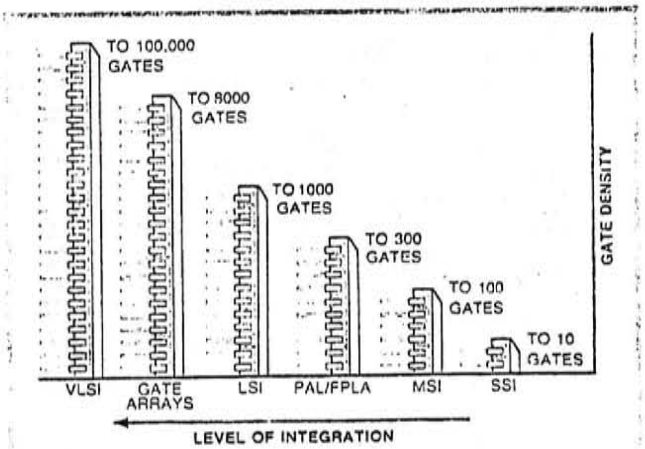


Fig 1—The density of gates on silicon devices *has increased steadily. The complexity of PAL/FPLAs fits in between MSI and LSI, while gate arrays approach VLSI.*

Gate arrays have advantages

Fig 1 shows increasing levels of silicon integration in various types of chips. PALs and FPLAs typically have 50 to 250 gates, while gate arrays generally contain 200 to 8000 gates. Just as one fuse-programmable device can replace four to six SSI or MSI chips, a single gate array can take the place of several PALs or FPLAs.

Custom logic isn't expensive

The average silicon utilization of a gate array is 80%, but only 50% for PALs or FPLAs. With greater numbers and densities of gates, the level of gate-array integration is an order of magnitude beyond that of PAL devices or FPLAs. If your application requires 4000 gates, for example, you'll need 20 to 30 fuse-programmable arrays, but only one gate array.

In addition to a higher level of integration, gate arrays have several other advantages over fuse-programmable logic. Gate arrays provide maximum pinout flexibility, which eases board layout. You can define each pad and its associated peripheral cell and use it as an input, output, 3-state buffer or power-ground pin. PAL circuits and FPLAs, on the other hand, usually have fixed input, output and power-ground pins, although some models have programmable I/O pins. Also, PALs and FPLAs have 28 pads at most; gate arrays can have as many as 180 pads.

The core structures of fuse-programmable devices and gate arrays are significantly different: PAL devices and FPLAs use global connectivity; gate arrays use local connectivity. With global connectivity, you can connect any input line to one or more product lines, and vice versa. Unless you blow the fuse, the intersection of each input and product line is a contact. Although this approach is flexible, you waste real estate, because only a few outputs are connected to each input. Because you can't move inputs, outputs and such devices as flip flops, fuse-programmable logic requires global connectivity.

The local connectivity used in gate arrays results in efficient use of real estate (**Fig 2**). Because you may place devices anywhere on a gate array, you can optimize the interconnect area. For arrays of less than

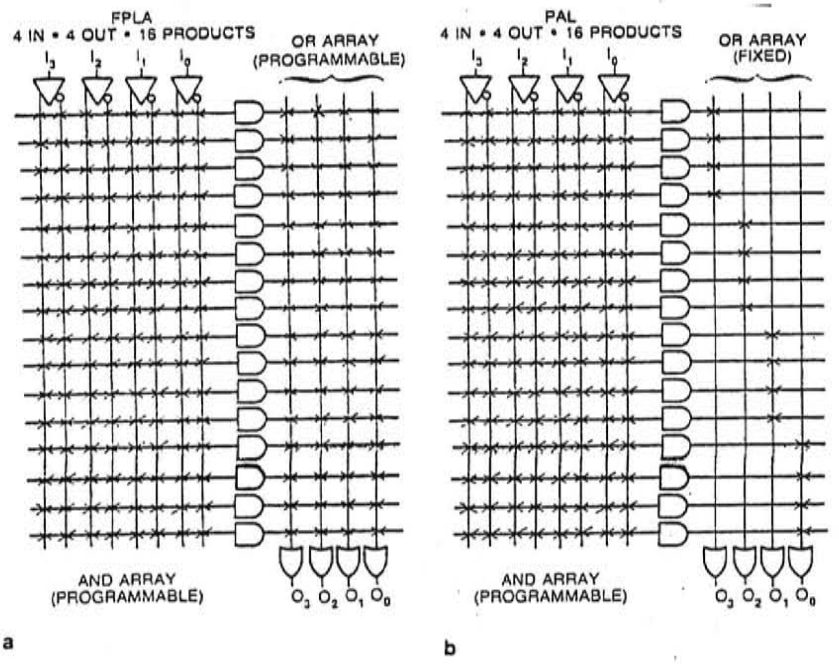
What do gate arrays, PALs and FPLAs do?

Standard cells, gate arrays and fuse-programmable logic each let you tailor a chip without starting from scratch. When using standard cells, you must customize the entire mask set to your requirements. Although this approach offers great design flexibility, the engineering costs and design time aren't much less than they would be for a fully customized layout. You can cut the design cycle and reduce engineering expense by using either gate arrays or fuse-programmable logic.

Fuse-programmable logic (also called field-programmable logic) comprises two subcategories: programmable array logic (PAL circuits) and field-programmable logic arrays (FPLAs). Both employ similar design philosophies, but they differ in architecture. In both cases, you must configure Boolean equations in a sum-of-products form, because PAL devices and FPLAs consist of AND and OR arrays. A language like PALASM can convert logic equations directly into fuse patterns.

In PAL circuits, the AND array is programmable, but the OR array is fixed (see figure, part a). Both the AND and the OR arrays are programmable in FPLAs (b). PAL circuits and FPLAs are standard off-the-shelf products. You can customize them by selectively blowing fuses with a PAL/FPLA programmer. Both device types are generally fabricated with bipolar technology.

Gate arrays consist of uncommitted transistor-pair cells (c), ar-



You selectively blow fuses to program PALs and FPLAs. For PALs, you program only the AND array; for FPLAs, you specify both AND and OR arrays. Uncommitted transistor pair cells are used in gate arrays for maximum flexibility in pinout assignments.

300 gates, the connectivity type makes little difference. Beyond 1000 gates, global connectivity becomes impractical.

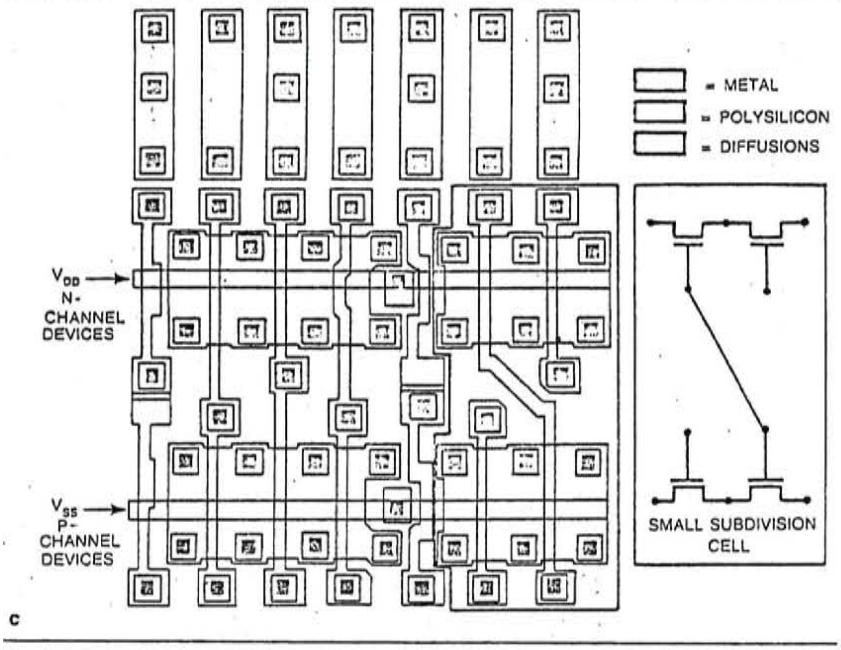
With local connectivity, gate arrays have a limited number of routing tracks between rows of cells. Because in most cases more interconnects exist at the center of an array than at the periphery, routing-track densities often vary within an array. For example, the 3-μm, double-metal CMOS gate array used in a video controller may have 12 routing tracks in the middle of the core but only six tracks around the periphery.

PAL devices and FPLAs can't continue to use global connectivity if gate densities are to be increased. Chip designers are developing devices that use a combination of local and global connectivity, but at the cost of design flexibility.

Another constraint of PAL architectures is the fact that a common input clock serves all registers, rendering PALs unsuitable for asynchronous logic operations. PAL circuits also lack an asynchronous preset and clear function. One other drawback of PAL circuits and FPLAs is that each flip flop is associated with an output; if you make a chain of flip flops, you'll be forced to waste the associated outputs. The gate-array-based circuit, however, can perform any type of synchronous or asynchronous logic. The number of inputs to any gate is limited only by speed considerations.

CMOS technology conserves power

Because PAL devices and FPLAs are typically produced with bipolar technology, they are fast and have high drive capability, but they dissipate a considerable amount of power. CMOS gate arrays are not as fast and their drive capability is lower, but the power dissipated



c

ranged in a matrix of rows and columns that are separated by routing channels. Most of the pads are uncommitted, so you have maximum flexibility in pinout assignments.

Vendors fabricate stock gate arrays by processing wafers to the first metal deposition. In your layout, you connect the uncommitted cells and route the channels according to your specifications. The vendor generates a mask with your layout information and then customizes the prefabricated wafers by etching the metal according to your pattern. The current standard technology in gate arrays is 3-μm silicon CMOS, with either single- or double-layer metal options.

Several programs create a prototype gate array from a logic diagram. First, you must convert the bipolar PAL into the equivalent CMOS gate-array logic. When you enter the schematic, you generate a net list that creates AMI CAD Technology's BOLT (Block Oriented Logic Translator) database. With this database, you can verify the logic with the company's SIMAD program and verify the timing with its Path program. You can also use the BOLT database to place and route the layout automatically. After layout, you calculate the line capacitance with AMI's Capacitance program. By entering the line capacitances into AMI's Delay program, you can calculate path delays. You can also run the Sentry test program with test vectors from SIMAD.

After you are satisfied with a layout, you produce a pattern-generation tape for fabricating the metal mask. Total development time, from submission of your logic diagrams to reception of your first silicon prototypes, should be six to 10 weeks.

Gate-array densities are increasing

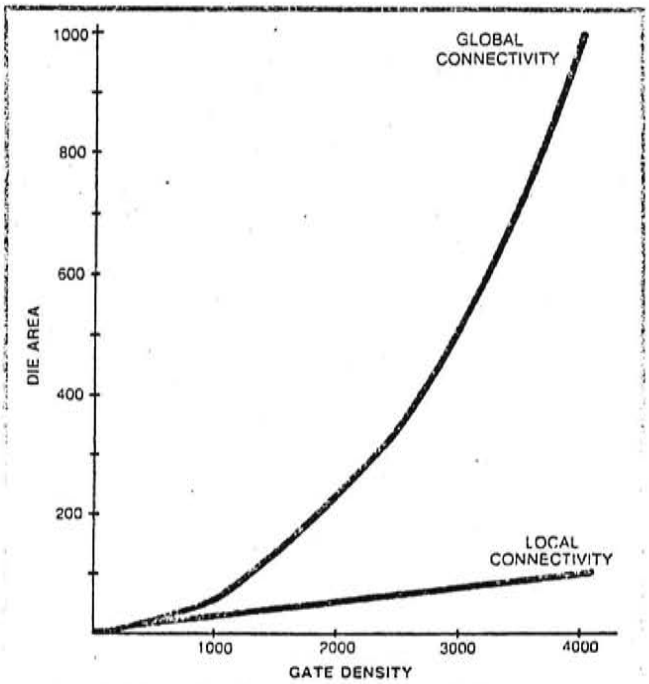


Fig 2—The global connectivity used in programmable-logic devices *consumes increasingly large amounts of real estate as the gate density rises. The local-connectivity design approach of gate arrays requires much less silicon for complex circuits.*

is in the microwatt range. You can occasionally overcome the speed problem of gate arrays, however. Their architectural freedom allows the design of circuits with faster logic than is possible in PALs and FPLAs. For example, in sequential PAL circuits and FPLAs, a signal must go through a fixed number of levels, ie, through the input buffer, AND/OR gates, flip flops and output stage. In gate arrays, you determine the number of logic levels. Also, PALs and FPLAs have fixed output drive; in gate arrays, you can join several outputs in parallel to achieve higher drive capability.

The static power consumption of CMOS gate arrays is low because all the transistors are off. The gate power used under dynamic conditions is given by the equation

$$\text{power consumption} = (\text{capacitance/gate area}) \times (\text{maximum operating voltage})^2 \times (\text{operating frequency}).$$

The dynamic power dissipation of CMOS gate arrays is relatively low because, on average, only about 20 to 30% of the gates are switching at a given time. Furthermore, only a small percentage of those gates are switching at the highest frequency. Power dissipation in bipolar fuse-programmable logic devices is the same under static or dynamic conditions—about an order of magnitude more than that of CMOS gate arrays. Each product term, whether or not it's used, draws 0.5 to 1 mA, which can limit high-density PAL devices or FPLAs.

You personalize PAL devices and FPLAs by blowing fuses with a programmer. Losses usually occur during the programming operation, so you should structure these chips for ease of testing when you design a state machine. Because of the flexible architecture of gate arrays, you can implement additional circuitry at little cost.

Design a video controller

If your speed specifications don't rule out CMOS technology, you'll find that you can replace several PALs or FPLAs with a gate array. You can, for example, implement a video controller using a gate array instead of eight PAL circuits.

When you replace the eight PAL circuits with a gate array, the number of external pins decreases from 160 to 31, board space shrinks from 24.75 to 8.75 in.^{2}, and power consumption under dynamic conditions decreases from 7.56 to 0.11W. The maximum frequency remains 8 MHz, and the cost per gate falls by about a factor of two (see nearby **table**).

Fig 3a shows a video controller designed with fuse-programmable logic. It consists of the eight PAL devices, a 2k×8-bit static RAM (the HM6116P), and a 5×7 character generator (the MM6055). The video-controller board is divided into two subsystems that share the RAM—in effect providing a multiport RAM. An ASCII code is entered into the system with an external device like a keyboard through a Write Only RS-232C port. The pointers Scrol and Cursor indicate where the code resides. Code is read out of the RAM from locations labeled Line and Character and transferred through a Read Only port to the character generator. The MM6055 generates characters to the screen. Reading occurs every cycle, even when nothing is written. The Read operation is continuous, so the picture stays stable. The Write operation is triggered by receipt of a special signal.

The dot-generator chip (a 20×8 PAL) is a shift register for the dots in a dot line of each character. DTCNT counts eight dots for each character. The character/cursor generator (a 20×10 PAL) sends the five least significant bits of the address to the RAM. The outputs called Character when reading from the RAM are renamed Cursor when writing into the RAM.

The scanner/line generator (a 20×10 PAL) produces the four most significant bits of the address to the RAM. The outputs that are identified as Lines when reading from the RAM are known as Scrol when writing to the RAM. The baud-rate generator (a 20×8 PAL) is a shift register. It loads the ASCII bits in serial form from the RXD line. The UART control (a 20×10

PAL) generates the Sample pulses and the Ready signal when a code for a character is in the UART. It also detects a false Start signal. The RAM control (a 20×10 PAL) governs other functions of the RAM, such as the Read and Write timings, swapping, and end-of-line detection.

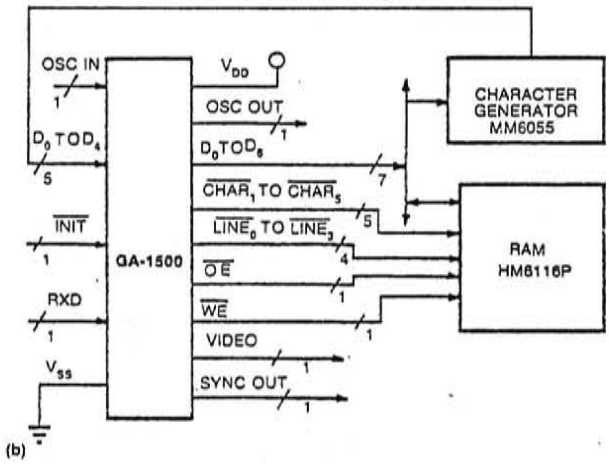
Convert the PAL design to a gate array

The redesigned video controller (Fig 3b) consists of a gate array and the same HM6116P RAM and MM6055 character generator used in the PAL design. All functions carried out by the PAL circuits are implemented in a single gate array.

The first steps in converting a PAL circuit to a gate array are partitioning the system and determining the pin count. Because of their complexity, the RAM and the character generator in the PAL design aren't included in the gate array, and you must provide a pinout option to connect the gate array with these two external components. Because you are integrating the eight

VIDEO-CONTROLLER DESIGN COMPARISON

	PALs	GATE ARRAYS
PART TYPES	6 20 × 10 PALs 2 20 × 8 PALs	GA-1500
NUMBER OF ICs	8	1
NUMBER OF PINS	160	31
POWER	7.56W	0.26 mW (STATIC) 0.11W (DYNAMIC)
SPEED	8 MHz	8 MHz
BOARD AREA	24.75 IN.²	8.75 IN.²
APPROXIMATE COST (HIGH VOLUME)	$0.02/GATE	$0.01/GATE
DEVELOPMENT STEPS	LOGIC CONVERSION LOGIC SIMULATION PROGRAMMING TESTING	LOGIC CONVERSION LOGIC SIMULATION LAYOUT TO PG TAPE PROCESSING TESTING
COST	DATA NOT AVAILABLE	APPROX $10,000 TO $20,000
TIME	DATA NOT AVAILABLE	APPROX 3 TO 10 WEEKS



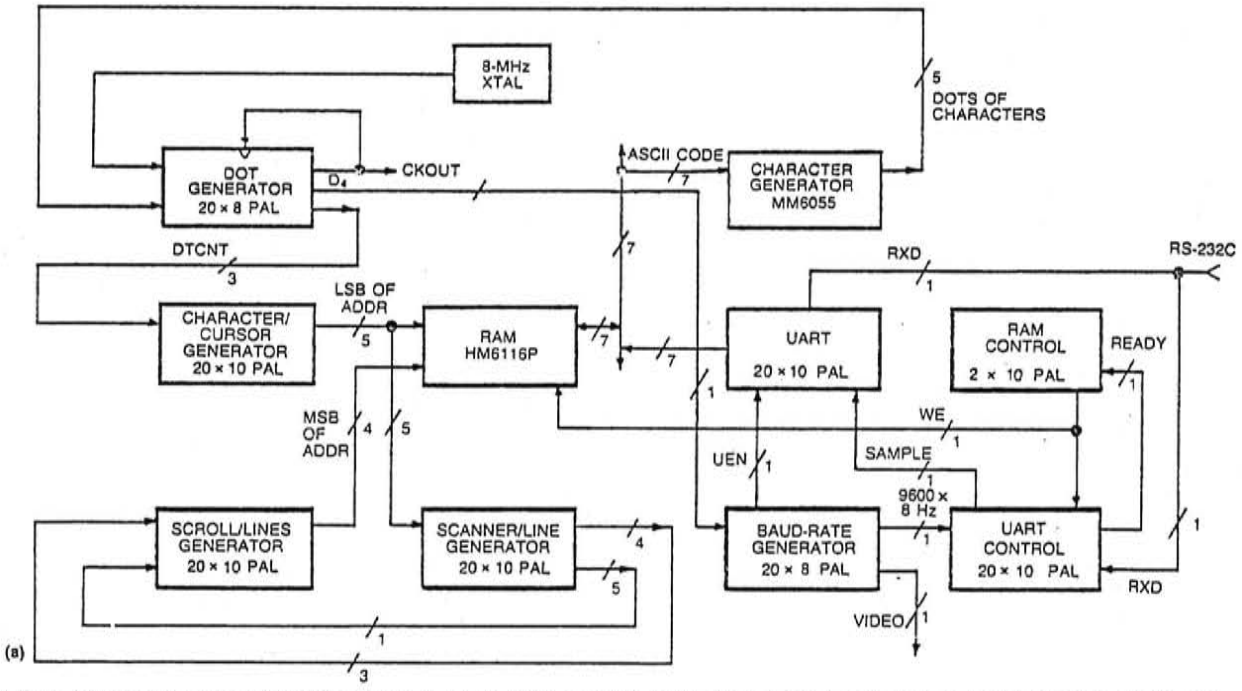


Fig 3—You can implement a video controller *with PALs or gate arrays. The GA-1500 gate array in* **b** *simplifies the circuit and replaces the eight PALs in* **a**.

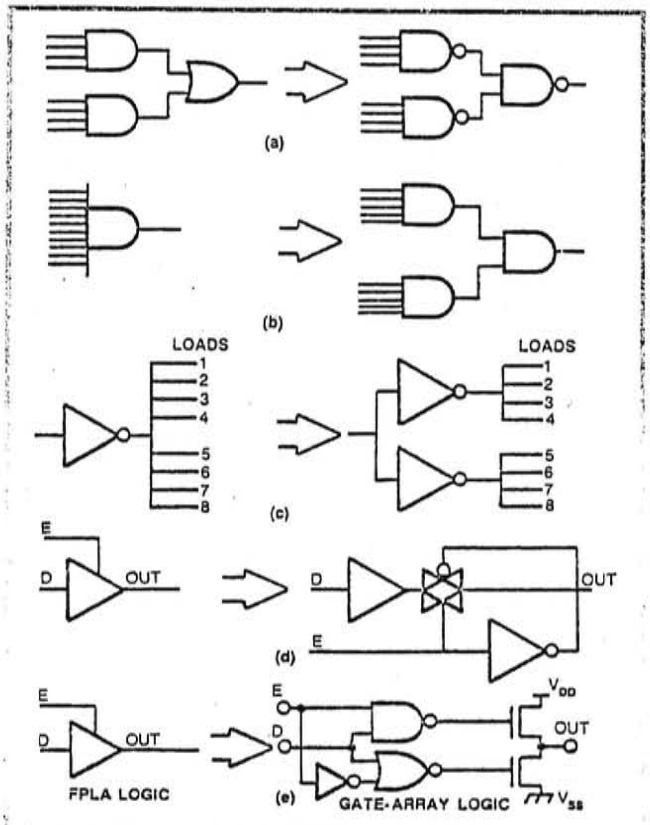


Fig 4—PALs consist of AND/OR logic; *CMOS gate arrays use a NAND/NOR design, with NAND preferred to NOR. In a typical example of converting PALs to gate-array logic, two AND gates going into an OR become three NANDs (a). Because CMOS has less fan-in and fan-out capability than bipolar, gates in PAL designs that have more than five inputs or outputs must be broken down for gate-array logic (b and c). Three-state and I/O buffers are slightly more complex in their gate-array versions than in their PAL implementations (d and e).*

PAL devices onto a chip, all the interconnections between the PAL circuits will be unnecessary in the gate array. You will need at least two pins for power and ground, depending on current and power-dissipation requirements. Extra peripheral cells might be required if you need higher output drive.

The next step is to determine the gate count. Before you can determine the actual gate count, you must convert the bipolar TTL of the PAL design into its equivalent CMOS gate-array logic. The internal structure of PAL devices consists of AND/OR logic, which can be directly translated into NAND/NOR logic in gate arrays (Fig 4a). NAND is the logic preferred in gate arrays.

As a result of their high drive capability, bipolar PAL devices can have AND gates with as many as 20 inputs. CMOS gates, with their large fan-in, must be subdivided into smaller gates to optimize performance (Fig 4b). As a rule of thumb, you should restrict to five the maximum number of inputs for a NAND/NOR gate in a CMOS array. The loading of each gate requires careful checking. Normally, internal gates drive four loads at most, although there is no limit to the number of loads that a gate can drive in applications where speed is unimportant. If a gate's fan-out is more than four loads, as is usually the case with clock drivers, you must provide proper buffering to increase the drive capability and meet the speed specifications. You can have higher drives simply by connecting devices in parallel (Fig 4c).

To convert TTL input levels to CMOS, you can use a structure similar to a 3-input NOR gate. A D-type flip flop in a PAL device can be implemented directly in a gate array by adding a Set or Reset signal. To make an internal 3-state buffer, you can simply use transmission gates (Fig 4d). You can design I/O buffers connecting the data bus with external devices by using output buffers and six internal gates (Fig 4e). If high drive is required at the output, you can connect two or more buffers in parallel. After you have completed the logic conversion from bipolar to CMOS, you are ready to make an accurate gate count. In the video-controller example, the equivalent gate count for the eight PAL devices is 1340 gates. Once you have determined the gate count and the pinout pattern, you can select a gate array that meets your ac and dc specifications.

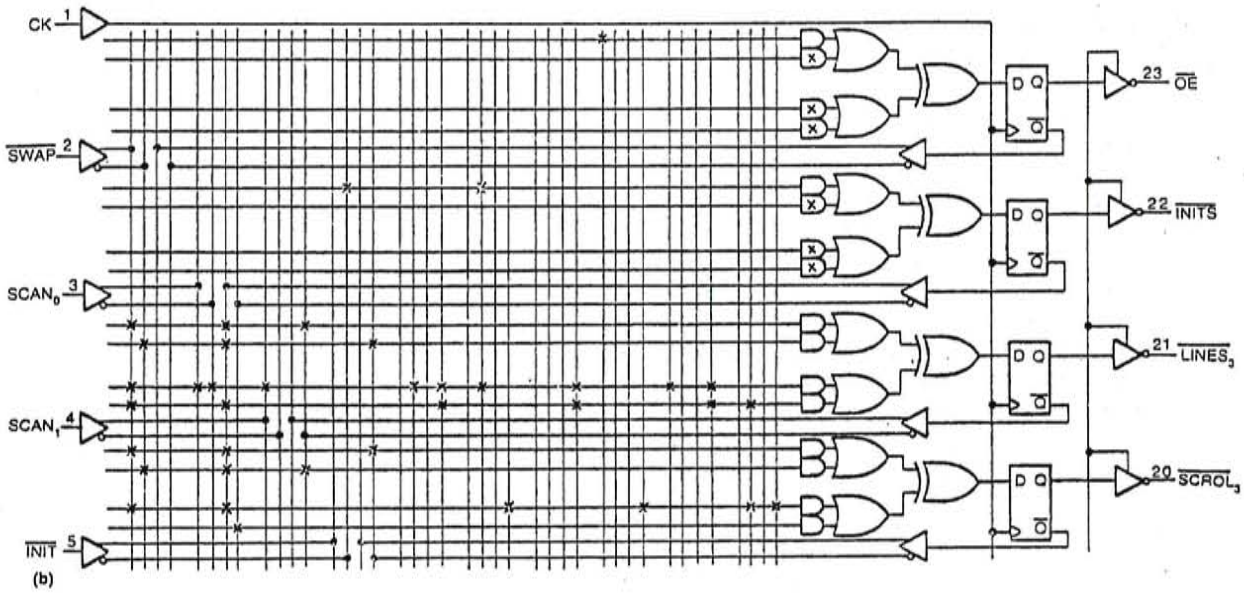
Check your design parameters

To interface with the outside world, the video controller must convert from TTL to CMOS. You can achieve a logical low-input voltage (V_{IL}) of 0.8V and a logical high-input voltage (V_{IH}) of 2.0V on the CMOS gate array by using a 3-input NOR structure. Another difference between bipolar and CMOS devices is that the 20×10 PAL has a logical low-output current (I_{OL}) of 24 mA at 0.5V, while the gate array furnishes 3.2 mA at 0.4V. If a higher I_{OL} figure is needed in the gate array, you should join multiple outputs in parallel. The supply current (I_{CC}) for each 20×10 PAL is 180 mA at 5.25V and is fixed. In gate arrays, the dynamic supply current depends on gate utilization. In any case, the supply current (I_{DD}) in CMOS gate arrays is an order of magnitude less than that of the equivalent PAL circuit and is not an issue when converting PAL devices into gate arrays.

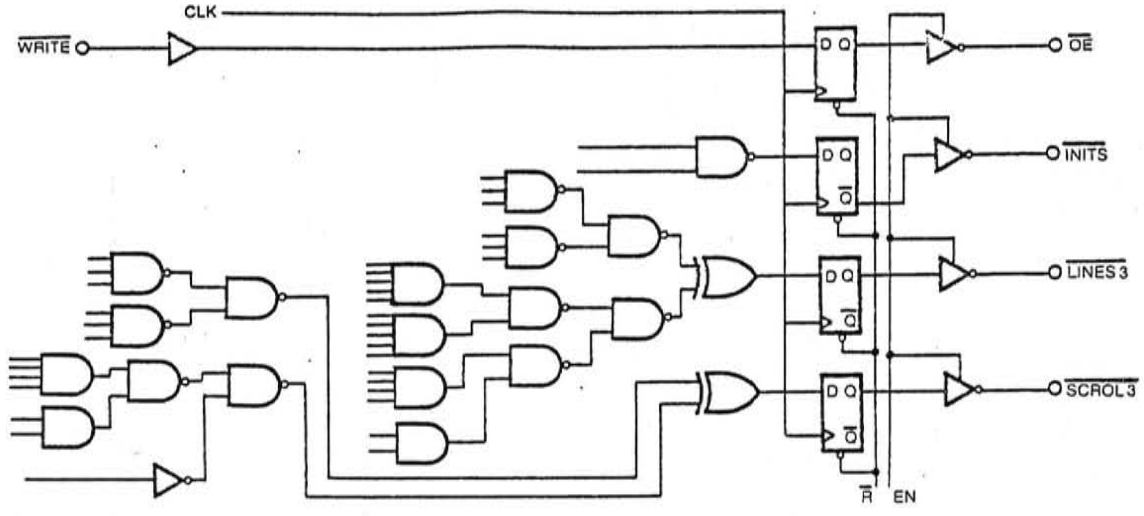
PAL devices have a fixed input-to-output delay of 40 nsec. Gate arrays have a flexible I/O structure and can be designed so that the input-to-output delay is lower or higher than that of PAL devices. Maximum clock rate in the video controller is 3 MHz. This speed requirement is easily met in both PAL devices and gate arrays. Current 3-μm CMOS single-metal gate-array

```
PAL20X10                                        [illegible] SPECIFICATION
VP4                                             BIRKNER/DUDI        7/14/81
LINES/SCROL GENERATOR
MMI SUNNYVALE, CALIFORNIA
  CK / SWAP     SCAN0     SCAN1     /INIT     [illegible]     NC     /WRITE     /LINE4
/INCSR GND /OC     /SCROL0     /LINES0  /SCROL1     /LINES1  /SCROL2     /LINES2     /SCROL3
/LINES3     /INITS   /OE // VCC

 LINES3    :=   /SWAP*/INITS*LINES 3                          ; HOLD
                +  SWAP*/INITS*SCROL3                         ; SWAP WITH SCROL
               :+:  /SWAP*/INITS*H255*/LINE 4                 ; INC (12 DOT LINES)
                        *  SCAN3*SCAN1*SCAN0
                        *  LINES2*LINES1*LINES0               ; INC (LF)
                +   /SWAP*/INITS*INCSCR
                        *  LINES2*LINES1*LINES0

 SCROL3    :=   /SWAP*/INITS*SCROL3                           ; HOLD
                +  SWAP*/INITS*LINES3                         ; SWAP WITH LINES
               :+:  /SWAP*/INITS*INSCR                        ; INC (LF OR CHAR = 47)
                        *SCROL2*SCROL1*SCROL0
                +          INITS                              ; INITIALIZE

 INITS     =    INIT*H255                                     ; INITIALIZATION SIGNAL

 OE        :=   /WRITE                                        ; ENABLE THREE STATE OF RAM
```

(a)



(b)



(c)

Fig 5—For a PAL-based design, *start with a specification* **(a)** *that you can convert into a logic diagram* **(b).** *You can remove the unused logic of the programmable-logic circuit during gate-array layout* **(c),** *thus offsetting the increased levels of logic required by CMOS's lower fan-in and fan-out capability.*

families can operate with a 30-MHz clock rate under worst-case commercial and process conditions; 3-μm CMOS double-metal families work at frequencies to 37 MHz. PAL circuits and gate arrays are fairly compatible for other ac and dc parameters.

Fig 5a shows a PAL design specification as a demonstration of how to convert from PALs to a gate array. Line 1 in the specification shows the PAL type. The pin list starts on line 5 with the logic equations written in the sum-of-products form. Note that the active-Low PAL devices have an inversion at the output, but the equations are written before the inversion. Consequently, the polarity of the output pins in the pin list and in the equations is always opposite. In gate arrays, you can write the equations in active High or Low form because the restriction of inverted outputs doesn't occur.

Bipolar PAL devices can have many inputs to a gate without degrading the gate's ac performance. In CMOS gate arrays, you'll slow the circuit if you increase the number of inputs to a gate. An increase in series resistance that slows the charging and discharging of the output node causes this decline in performance. The standard procedure is to limit to five the number of inputs to a CMOS device. One product term of the PAL device in the video controller has a 10-input AND gate. When implemented in the gate array, two 5-input gates replace the 10-input device (Fig 4b). This increases the number of logic levels, but the total delay stays within the range of the specified clock rate.

Compare PAL and gate-array logic

Fig 5b shows a portion of the PAL diagram. The horizontal lines ending in AND gates are product terms and the vertical lines are inputs. The Xs mark intact fuses. An X inside a gate indicates an unused gate. All flip flops have a common clock, and all outputs have a common Chip Enable. Inputs are fed into the array as true and complement. The output of the flip flop is also fed back into the array as true and complement.

Fig 5c gives the gate-array version of the logic. By removing the unused logic in the case of output $\overline{OE}$, you can simplify three levels of logic. Because NAND logic is preferred for CMOS gate arrays, you should implement the $\overline{INITS}$ output with a NAND gate. As discussed earlier, you must break the 10-input AND gate into two 5-input AND gates on output $\overline{Lines3}$. You must also change the overall AND/OR structure to NAND/NOR logic, increasing the path delay by one logic level. Similarly, for $\overline{Scrol3}$, you replace the 6-input AND gate with a 4-input gate and a 2-input gate.

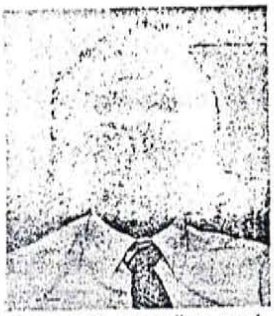
PAL devices use one clock signal and transfer data at the rising edge of the clock. Gate arrays aren't limited to a single clock and can transfer data on the rising or falling edge of a clock. The same is true of the Output Enable pin. PAL circuits don't have asynchronous preset and clear functions, but you can implement such functions in gate arrays without adding gates. These functions were used in the video controller.

PAL design specifications include testing information in the form of a function table that contains input stimuli and output test vectors. The same information can be used by AMI CAD Technology's (Santa Clara, CA) gate-array logic-verification program, called SIMAD (Logic Simulator with Assignable Delays) to generate test programs.

The gate-array implementation of the controller requires 1340 gates and 31 I/O pins. You can implement this logic on the GA-1500 array, which has 1500 gates and 64 I/O pins. The gate utilization is 89% (gate utilizations average approximately 80%). The next level of array complexity is provided by the GA-2000, which contains 2000 gates and 74 I/O pins. Both arrays are made with 3-μm, oxide-isolated CMOS technology. EDN

Authors' biographies

Saeed Kazmi is an applications manager at VLSI Technologies Inc (San Jose, CA), where he concentrates on technical communication between designers and customers of gate arrays. After graduating with a BA in electrical engineering from the University of Karachi, Pakistan, in 1972, he received his MS degree in electrical engineering from Oregon State University in 1976. Before joining VLSI Technologies, Saeed worked for American Microsystems and Monolithic Memories. He likes to travel.



Michael Friedman is an applications engineer at American Microsystems Inc (Santa Clara, CA), where he specializes in technical-feasibility studies and marketing support. Mike came to AMI from Intersil. He is a graduate of the University of Michigan, where he studied engineering physics, and he is presently studying for an MBA from the University of Santa Clara. Mike enjoys golf, traveling and camping.

GOULD

AMI Semiconductors

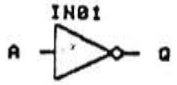
IN01

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Single Pre-Routed Inverter

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 1

Bolt Syntax: Q .IN01 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.6	2.4	3.1	3.8	6.7
		t_{PHL}	Ø.1	Ø.5	Ø.9	1.2	2.6

Intrinsic Parameters tdx	ktdx
Ø.9	2.92
Ø.Ø	1.43

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

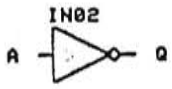
IN02

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Two Pre-Routed Inverters In Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 1

Bolt Syntax: Q .IN02 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.Ø	1.4	1.8	2.2	3.7
		t_{PHL}	Ø.1	Ø.3	Ø.6	Ø.8	1.7

Intrinsic Parameters tdx	ktdx
Ø.6	1.55
Ø.Ø	Ø.9Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

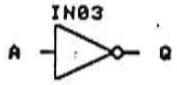
IN03

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Three Pre-Routed Inverters in Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: 1.5

Bolt Syntax: Q .IN03 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 4	8	12	16	20
Any Input	Q	t_{PLH}	2.0	3.1	4.3	5.5	6.7
		t_{PHL}	0.2	0.9	1.7	2.4	3.1

Intrinsic Parameters tdx	ktdx
0.8	1.20
0.0	0.72

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

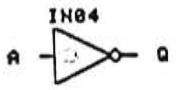
IN04

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Four Pre-Routed Inverters in Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	4

Equivalent Gate Count: 2.5

Bolt Syntax: Q .IN04 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 4	8	12	16	20
Any Input	Q	t_{PLH}	1.4	2.3	3.2	4.1	5.0
		t_{PHL}	0.2	0.8	1.4	2.0	2.7

Intrinsic Parameters tdx	ktdx
0.5	0.90
0.0	0.62

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

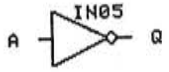
IN05

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Five Pre-Routed Inverters In Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	5

Equivalent Gate Count: 2.5

Bolt Syntax: Q .IN05 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 4	8	12	16	20
Any Input	Q	t_{PLH}	1.5	2.2	3.0	3.8	4.6
		t_{PHL}	0.1	0.6	1.2	1.7	2.3

Intrinsic Parameters tdx	ktdx
0.7	0.80
0.0	0.54

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

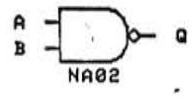
NAØ2

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Two Input NAND Gate

Logic Symbol



Truth Table

A	B	Q
L	X	H
X	L	H
H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1

Bolt Syntax: Q .NAØ2 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.2	2.9	3.6	4.4	7.3
		t_{PHL}	Ø.5	1.Ø	1.5	1.9	3.8

Intrinsic Parameters tdx	ktdx
1.4	2.95
Ø.1	1.84

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

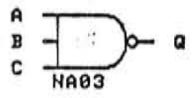
NAØ3

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Three Input NAND Gate

Logic Symbol



Truth Table

A	B	C	Q
L	X	X	H
X	L	X	H
X	X	L	H
H	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1.5

Bolt Syntax: Q .NAØ3 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.8	3.5	4.3	5.Ø	8.Ø
		t_{PHL}	1.2	1.8	2.4	3.Ø	5.4

Intrinsic Parameters tdx	ktdx
2.Ø	2.97
Ø.6	2.43

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

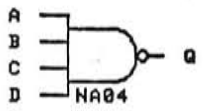
NAØ4

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Four Input NAND Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	X	X	X	H
X	L	X	X	H
X	X	L	X	H
X	X	X	L	H
H	H	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .NAØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.4	4.2	5.Ø	5.7	8.8
		t_{PHL}	2.2	3.Ø	3.8	4.6	7.9

Intrinsic Parameters tdx	ktdx
2.6	3.Ø8
1.3	3.3Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

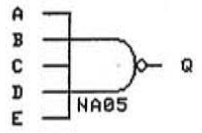
NAØ5

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Five Input NAND Gate

Logic Symbol



Truth Table

A	B	C	D	E	Q
L	X	X	X	X	H
X	L	X	X	X	H
X	X	L	X	X	H
X	X	X	L	X	H
X	X	X	X	L	H
H	H	H	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .NAØ5 A B C D E;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.9	4.7	5.5	6.3	9.4
		t_{PHL}	3.3	4.3	5.3	6.3	1Ø.2

Intrinsic Parameters tdx	ktdx
3.1	3.15
2.3	3.99

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

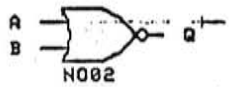
NØØ2

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Two Input NOR Gate

Logic Symbol



Truth Table

A	B	Q
L	L	H
X	H	L
H	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1

Bolt Syntax: Q .NØØ2 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.8	4.Ø	5.2	6.4	11.3
		t_{PHL}	Ø.4	Ø.7	1.1	1.4	2.8

Intrinsic Parameters tdx	ktdx
1.6	4.84
Ø.1	1.37

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

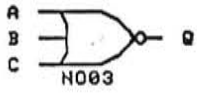
NØØ3

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Three Input NOR Gate

Logic Symbol



Truth Table

A	B	C	Q
L	L	L	H
X	X	H	L
X	H	X	L
H	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1.5

Bolt Syntax: Q .NØØ3 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.2	6.9	8.7	1Ø.4	17.2
		t_{PHL}	Ø.6	1.Ø	1.3	1.7	3.1

Intrinsic Parameters tdx	ktdx
3.5	6.82
Ø.2	1.44

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

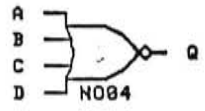
NØ04

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Four Input NOR Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	L	L	L	H
X	X	X	H	L
X	X	H	X	L
X	H	X	X	L
H	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .NØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	8.3	1Ø.7	13.2	15.6	25.4
		t_{PHL}	Ø.7	1.Ø	1.4	1.8	3.3

Intrinsic Parameters tdx	ktdx
5.8	9.8Ø
Ø.3	1.5Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

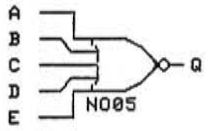
NØ5

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Five Input NOR Gate

Logic Symbol



Truth Table

A	B	C	D	E	Q
L	L	L	L	L	H
X	X	X	X	H	L
X	X	X	H	X	L
X	X	H	X	X	L
X	H	X	X	X	L
H	X	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .NØ5 A B C D E;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	12.4	15.3	18.2	21.1	32.9
		t_{PHL}	Ø.9	1.2	1.5	1.9	3.2

Intrinsic Parameters tdx	ktdx
9.4	11.72
Ø.5	1.36

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

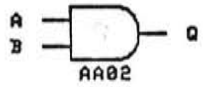
AA02

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Two Input AND Gate

Logic Symbol



Truth Table

A	B	Q
L	X	L
X	L	L
H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1.5

Bolt Syntax: Q .AA02 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.8	2.5	3.2	3.9	6.7
		t_{PHL}	2.2	2.5	2.8	3.1	4.4

Intrinsic Parameters tdx	ktdx
1	2.85
1.9	1.20

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

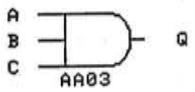
AAØ3

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre Routed Three Input AND Gate

Logic Symbol



Truth Table

A	B	C	Q
L	X	X	L
X	L	X	L
X	X	L	L
H	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .AAØ3 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.5	3.2	4.Ø	4.7	7.6
		t_{PHL}	2.9	3.2	3.5	3.8	5.1

Intrinsic Parameters tdx	ktdx
1.8	2.92
2.5	1.3Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

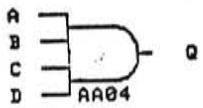
AA04

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre Routed Four Input AND Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	X	X	X	L
X	L	X	X	L
X	X	L	X	L
X	X	X	L	L
H	H	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .AA04 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.4	4.1	4.9	5.6	8.6
		t_{PHL}	3.2	3.6	3.9	4.2	5.6

Intrinsic Parameters tdx	ktdx
2.7	2.97
2.9	1.33

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

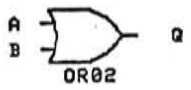
OR02

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Two Input OR Gate

Logic Symbol



Truth Table

A	B	Q
L	L	L
H	X	H
X	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1.5

Bolt Syntax: Q .OR02 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.5	2.3	3.0	3.7	6.5
		t_{PHL}	2.7	3.0	3.3	3.6	4.9

Intrinsic Parameters tdx	ktdx
0.8	2.85
2.3	1.30

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

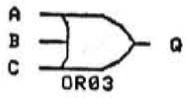
OR03

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Three Input OR Gate

Logic Symbol



Truth Table

A	B	C	Q
L	L	L	L
H	X	X	H
X	H	X	H
X	X	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .OR03 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.7	2.4	3.1	3.8	6.7
		t_{PHL}	3.8	4.2	4.6	5.Ø	6.6

Intrinsic Parameters tdx	ktdx
Ø.9	2.88
3.4	1.62

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

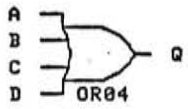
OR04

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Four Input OR Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	L	L	L	L
H	X	X	X	H
X	H	X	X	H
X	X	H	X	H
X	X	X	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .OR04 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.1	2.8	3.6	4.3	7.4
		t_{PHL}	7.5	8.Ø	8.6	9.1	11.2

Intrinsic Parameters tdx	ktdx
1.4	2.88
7.Ø	2.1Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

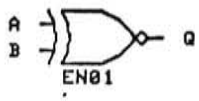
EN01

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Two Input Exclusive - NOR Gate

Logic Symbol



Truth Table

A	B	Q
L	L	H
L	H	L
H	L	L
H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	2

Equivalent Gate Count: 2.5

Bolt Syntax: Q .EN01 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.0	2.7	3.4	4.2	7.1
		t_{PHL}	2.6	3.1	3.5	4.0	5.8

Intrinsic Parameters tdx	ktdx
1.3	2.90
2.2	1.84

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

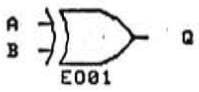
E001

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Two Input Exclusive - Or Gate

Logic Symbol



Truth Table

A	B	Q
L	L	L
L	H	H
H	L	H
H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	2

Equivalent Gate Count: 2.5

Bolt Syntax: Q .EØ1 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	4.Ø	5.4	6.9	8.3	14.1
		t_{PHL}	2.7	3.Ø	3.4	3.7	5.1

Intrinsic Parameters tdx	ktdx
2.6	5.75
2.4	1.35

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

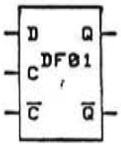
DFØ1

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop

Logic Symbol



Truth Table

C	CN	D	Q	QN
L	H	X	No Change	
↑	↓	L	L	H
↑	↓	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ1 D C CN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	3	3.4	3.9	4.3	6.Ø
		t_{PHL}	2.4	2.6	2.9	3.1	4.1
C	QN	t_{PLH}	4.1	4.8	5.6	6.3	9.2
		t_{PHL}	3.2	3.5	3.8	4.1	5.3
Set-Up		To H	1.8				
Set-Up		To L	2.2				

Intrinsic Parameters tdx	ktdx
2.6	1.7Ø
2.2	1.ØØ
3.4	2.9Ø
2.9	1.2Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

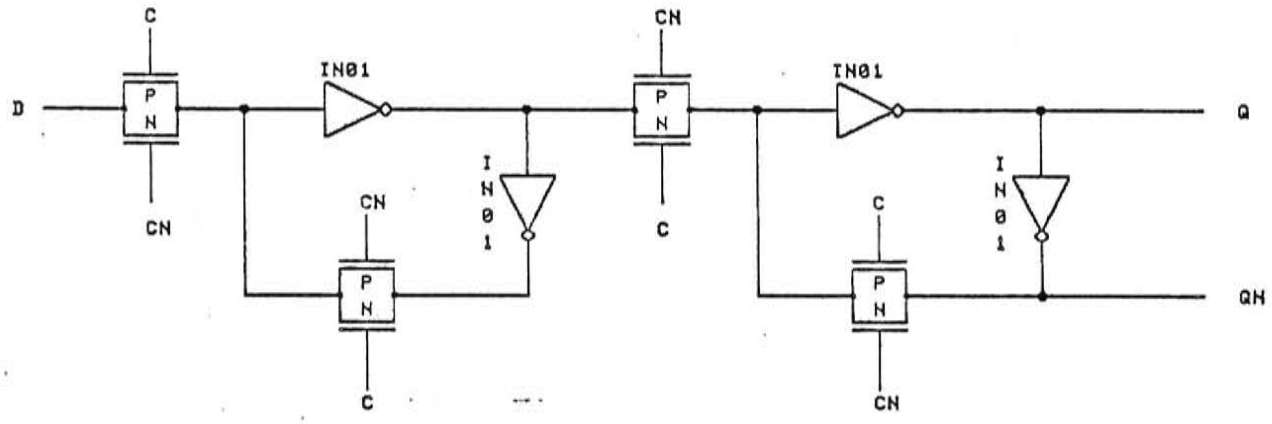
GOULD

AMI Semiconductors

DF01

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

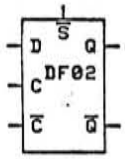
DFØ2

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Active Low Set

Logic Symbol



Truth Table

SN	C	CN	D	Q	QN
H	L	H	X	No Change	
H	↑	↓	L	L	H
H	↑	↓	H	H	L
L	X	X	X	H	L

Input Loading

Logic Input	Equivalent Unit Loads
D	2.5
All Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ2 D C CN SN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	3.4	4.2	4.9	5.7	8.6
		t_{PHL}	3.1	3.5	4.Ø	4.5	6.5
C	QN	t_{PLH}	4.6	5.4	6.1	6.8	9.6
		t_{PHL}	3.6	3.9	4.2	4.5	5.7
SN	Q	t_{PLH}	2.6	3.3	4.Ø	4.8	7.6
SN	QN	t_{PHL}	3.Ø	3.3	3.6	3.8	5.Ø
Set-Up		To H	2.8				
Set-Up		To L	3.0				

Intrinsic Parameters tdx	ktdx
2.7	2.99
2.6	1.97
3.9	2.85
3.3	1.17
1.9	2.85
2.7	1.11

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

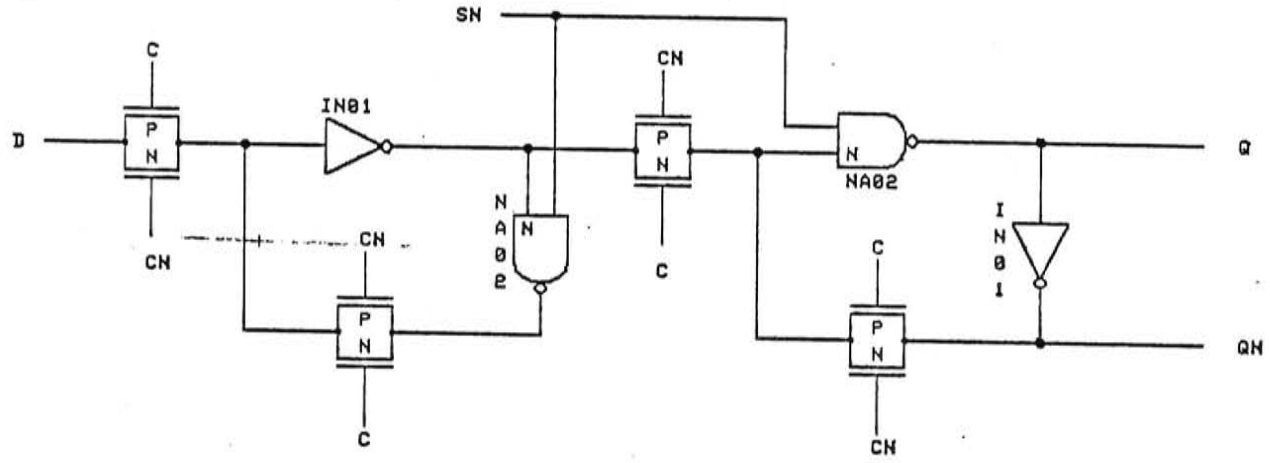
GOULD

AMI Semiconductors

DFØ2

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

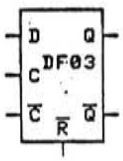
DFØ3

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Active Low Reset

Logic Symbol



Truth Table

RN	C	CN	D	Q	QN
H	L	H	X	No Change	
H	↑	↓	L	L	H
H	↑	↓	H	H	L
L	X	X	X	L	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ3 D C CN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	3.8	4.6	5.4	6.1	9.2
		t_{PHL}	2.7	3.1	3.5	3.8	5.3
C	QN	t_{PLH}	4.7	5.4	6.2	6.9	9.8
		t_{PHL}	4.1	4.5	5.Ø	5.4	7.1
RN	QN	t_{PLH}	3.5	4.2	4.9	5.5	8.2
RN	Q	t_{PHL}	3.8	4.2	4.6	4.9	6.4
Set-Up		To H	2.7				
Set-Up		To L	3.1				

Intrinsic Parameters tdx	ktdx
3.Ø	3.1Ø
2.3	1.5Ø
4.Ø	2.9Ø
3.7	1.7Ø
2.8	2.7Ø
3.4	1.5Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

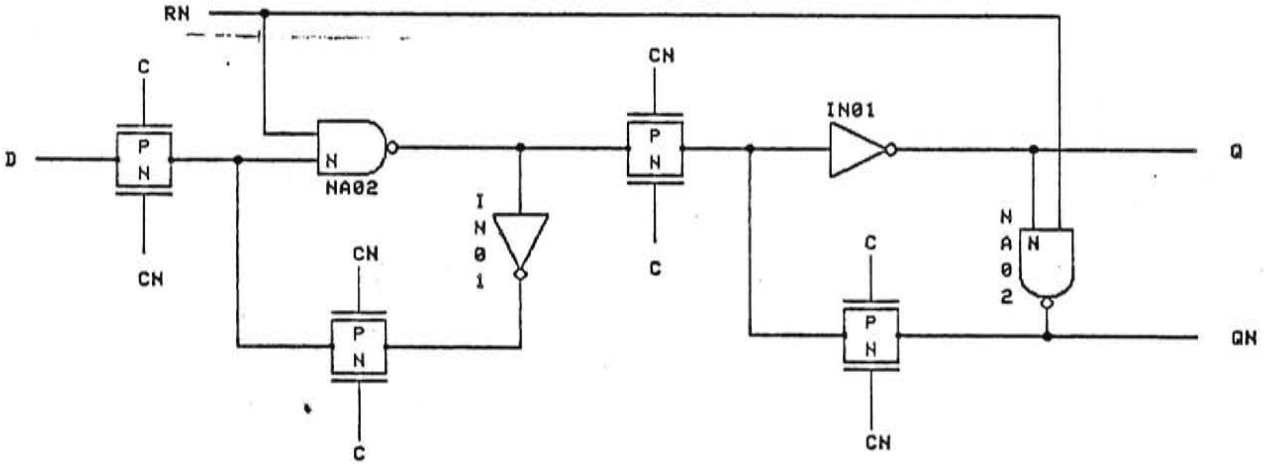
GOULD

AMI Semiconductors

DFØ3

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

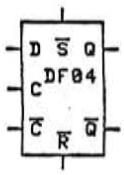
DFØ4

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Active Low Set And Active Low Reset

Logic Symbol



Truth Table

SN	RN	C	CN	D	Q	QN
H	H	L	H	X	No Change	
H	H	↑	↓	L	L	H
H	H	↑	↓	H	H	L
H	L	X	X	X	L	H
L	H	X	X	X	H	L
L	L	X	X	X	Illegal	

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 7

Bolt Syntax: Q QN .DFØ4 D C CN SN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	4.5	5.2	6.Ø	6.7	9.8
		t_{PHL}	3.1	3.5	4.Ø	4.5	6.5
C	QQ	t_{PLH}	5.3	6.1	6.8	7.5	1Ø.4
		t_{PHL}	4.8	5.3	5.7	6.2	7.9
SN	Q	t_{PLH}	2.2	2.9	3.6	4.4	7.3
SN	Q	t_{PHL}	4.6	5.1	5.7	6.2	8.3
RN	QN	t_{PLH}	3.6	4.3	5.Ø	5.7	8.4
RN	Q	t_{PHL}	4.9	5.4	6.Ø	6.5	8.6
Set-Up		To H	3.4				
Set-Up		To L	3.6				

Intrinsic Parameters tdx	ktdx
3.7	3.Ø5
2.6	1.95
4.6	2.89
4.4	1.74
1.4	2.95
4.1	2.1Ø
3.Ø	2.71
4.4	2.1Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

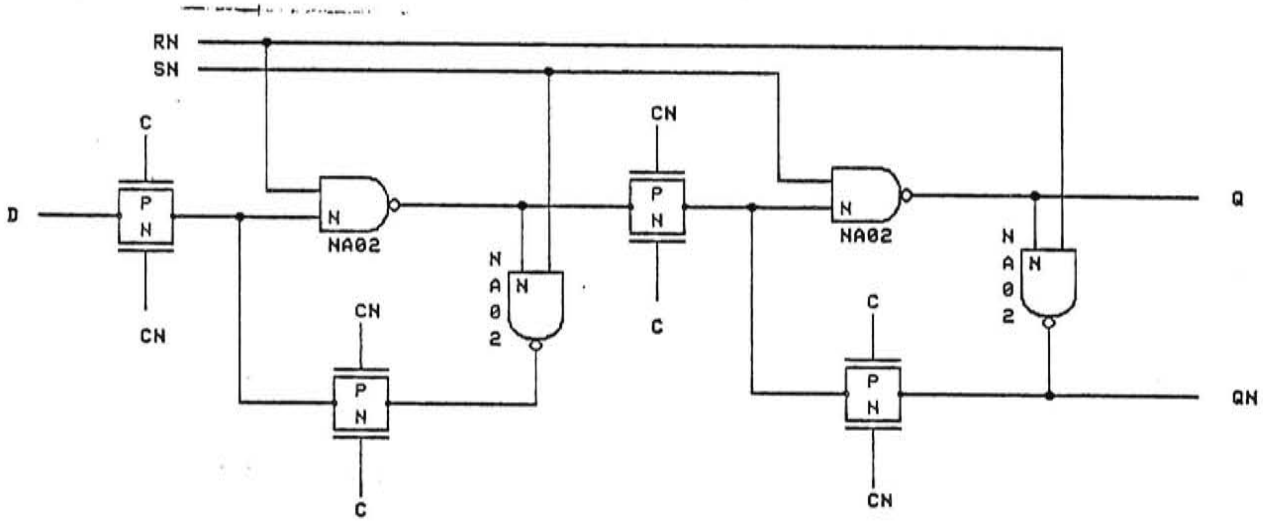
GOULD

AMI Semiconductors

DFØ4

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

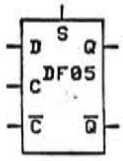
DFØ5

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Flip Flop With Asynchronus Set.

Logic Symbol



Truth Table

S	C	C̄N	D	Q	Q̄N
L	L	H	X	No Change	
L	↑	↓	L	L	H
L	↑	↓	H	H	L
H	X	X	X	H	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Others	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ5 D C CN S;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	2.9	3.7	4.5	5.2	8.3
		t_{PHL}	2.5	2.9	3.4	3.8	5.6
C	QN	t_{PLH}	5.4	6.6	7.8	9.Ø	13.8
		t_{PHL}	2.8	3.1	3.4	3.7	4.8
S	Q	t_{PLH}	3.3	4.1	4.8	5.6	8.5
S	QN	t_{PHL}	1.3	1.6	1.9	2.2	3.5
SET UP TIME		TO H	2.3				
SET UP TIME		TO L	3.6				

Intrinsic Parameters tdx	ktdx
2.2	3.07
2.1	1.74
4.2	4.8Ø
2.5	1.15
2.6	2.97
1.Ø	1.25

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

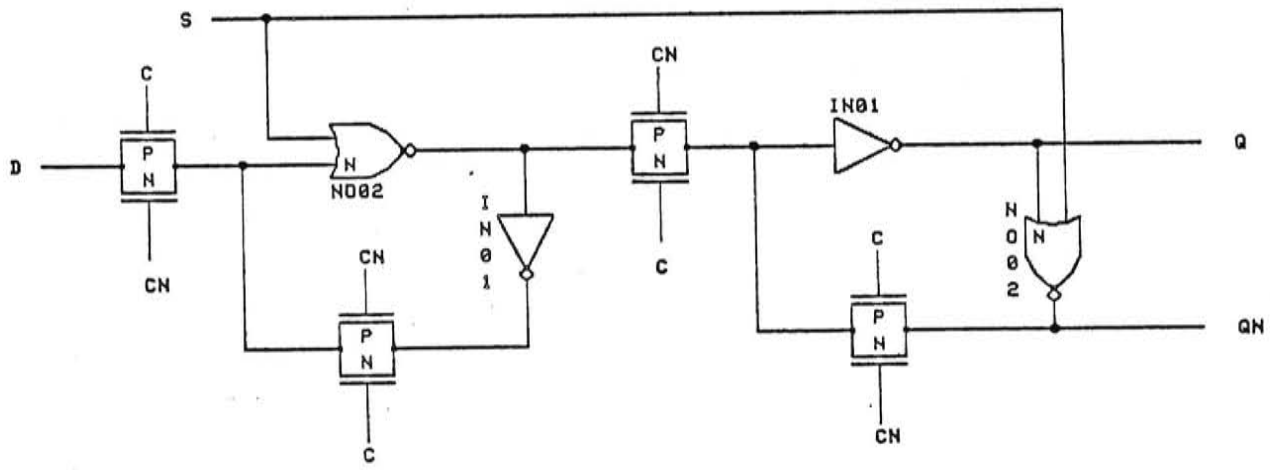
GOULD

AMI Semiconductors

DF05

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

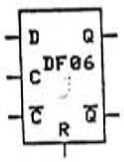
DFØ6

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Reset.

Logic Symbol



Truth Table

R	C	CN	D	Q	QN
L	L	H	X	No Change	
L	↑	↓	L	L	H
L	↑	↓	H	H	L
H	X	X	X	L	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
Any Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DGØ6 D C CN R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH} t_{PHL}	4.4 2.4	5.7 2.7	6.9 3.1	8.1 3.5	13.2 5.Ø
C	QN	t_{PLH} t_{PHL}	3.9 3.5	4.7 3.8	5.4 4.2	6.1 4.5	9.Ø 5.8
R	Q	t_{PHL}	Ø.6	1.Ø	1.3	1.7	3.1
R	QN	t_{PLH}	3.5	4.2	4.9	5.7	8.6
SET-UP-TIME		To H	3.6				
SET-UP-TIME		To L	2.8				

Intrinsic Parameters tdx	ktdx
3.2 2.Ø	5.ØØ 1.5Ø
3.2 3.2	2.9Ø 1.3Ø
Ø.3	1.4Ø
2.7	2.95

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

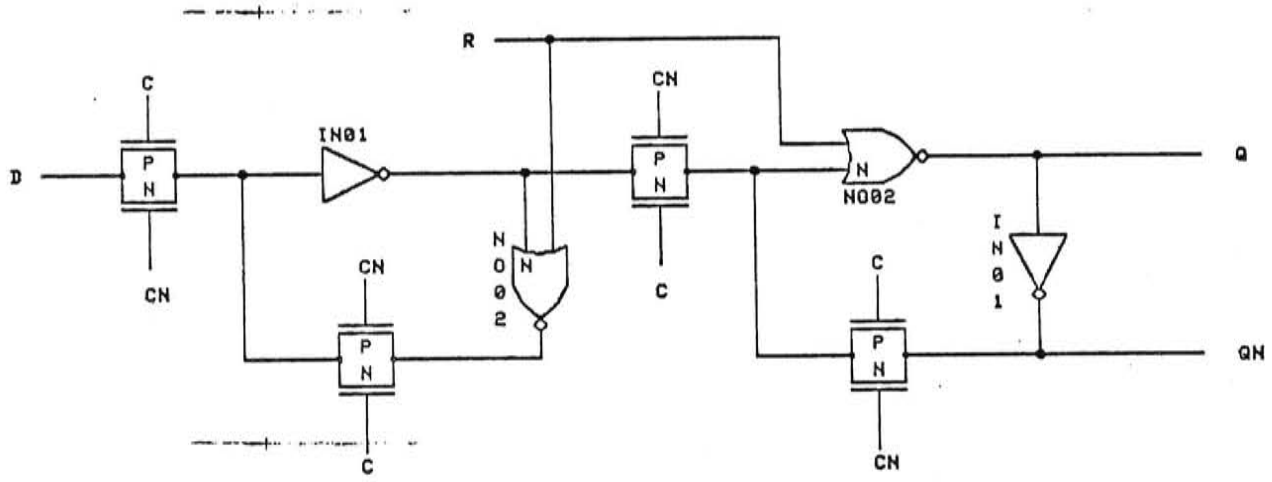
GOULD

AMI Semiconductors

DF06

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

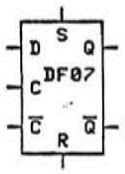
DF07

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronus Set And Reset.

Logic Symbol



Truth Table

S	R	C	CN	D	Q	QN
L	L	L	H	X	No Change	
L	L	H	L	L	L	H
L	L	H	L	H	H	L
L	H	X	X	X	L	H
H	L	X	X	X	H	L
H	H	X	X	X	ILLEGAL	

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Others	2

Equivalent Gate Count: 6

Bolt Syntax: Q QN .DFØ7 D C CN S R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	4.4	5.6	6.8	8.1	13.Ø
		t_{PHL}	2.5	2.9	3.4	3.8	5.5
C	QN	t_{PLH}	5.8	7.Ø	8.2	9.4	14.3
		t_{PHL}	3.7	4.Ø	4.3	4.6	5.9
S	Q	t_{PLH}	4.6	5.9	7.1	8.4	13.3
S	QN	t_{PHL}	1.4	1.7	2.Ø	2.3	3.6
R	Q	t_{PHL}	Ø.7	1.Ø	1.3	1.6	2.9
R	QN	t_{PLH}	6.4	7.5	8.7	9.9	14.7
SET UP TIME		TO L	3.3				
SET UP TIME		TO H	3.8				

Intrinsic Parameters tdx	ktdx
3.1	4.95
2.1	1.7Ø
4.6	4.85
3.4	1.25
3.4	4.96
1.Ø	1.28
Ø.4	1.25
5.2	4.75

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

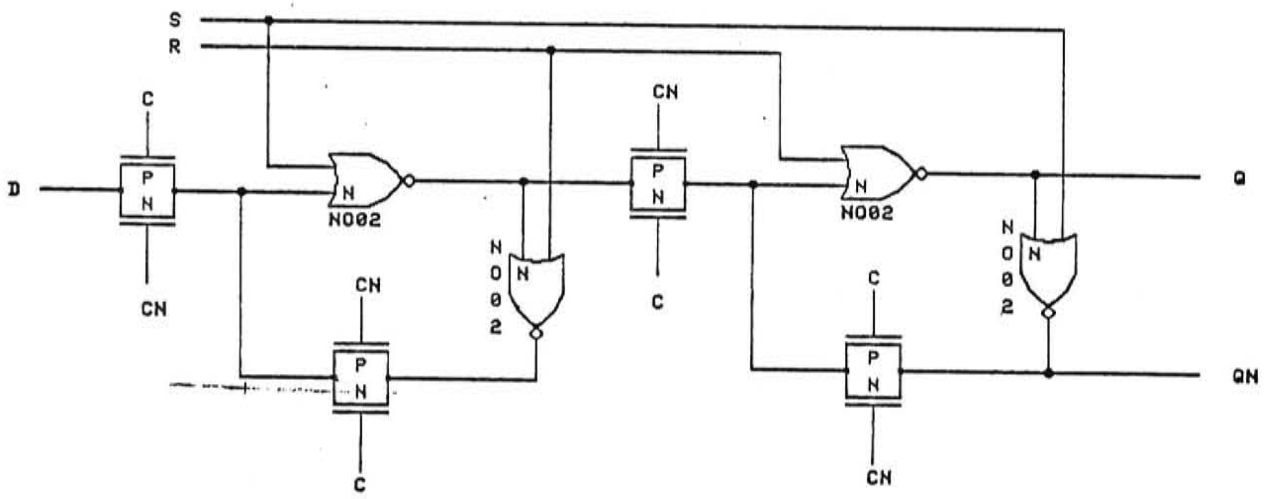
GOULD

AMI Semiconductors

DF07

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

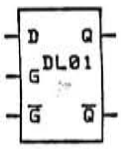
DLØ1

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Q and QN

Logic Symbol



Truth Table

G	GN	D	Q	QN
L	H	X	No Change	
H	L	D	D	DN

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q QN .DLØ1 D G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{LH}	4.3	5.Ø	5.7	6.4	9.3
		t_{HL}	3.1	3.4	3.7	3.9	5.1
G	QN	t_{LH}	2.8	3.2	3.6	4.Ø	5.7
		t_{HL}	2.5	2.8	3.Ø	3.3	4.4
D	Q	t_{PLH}	2.1	2.9	3.6	4.3	7.2
D	Q	t_{PHL}	2.2	2.5	2.8	3.Ø	4.1
D	QN	t_{PLH}	1.9	2.3	2.7	3.2	4.8
D	QN	t_{PHL}	Ø.5	Ø.8	1.Ø	1.3	2.4

Intrinsic Parameters tdx	ktdx
3.6	2.85
2.8	1.12
2.4	1.65
2.2	1.Ø6
1.4	2.9Ø
1.9	1.1Ø
1.5	1.67
Ø.2	1.11

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

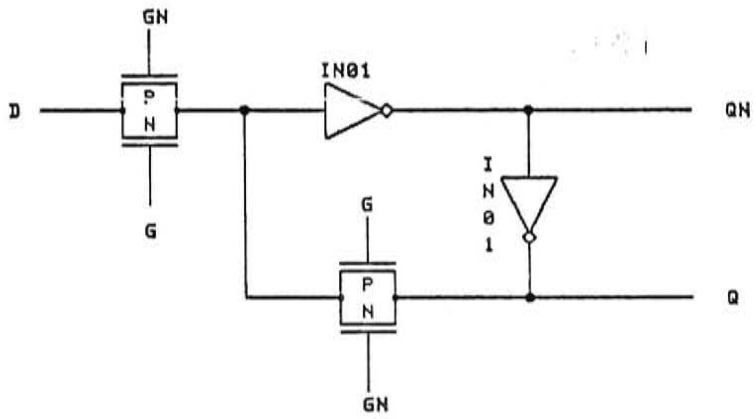
GOULD

AMI Semiconductors

DLØ1

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

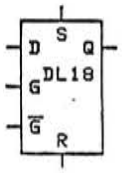
DL18

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Asynchronous Set And Reset

Logic Symbol



Truth Table

S	R	G	GN	D	Q
L	L	L	H	X	No Change
L	L	H	L	D	D
L	H	X	X	X	L
H	L	X	X	X	H
H	H	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
D	2.5
All Other	1

Equivalent Gate Count: 3.5

Bolt Syntax: Q .DL18 D G GN S R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	5.7	6.9	8.1	9.4	14.3
		t_{PHL}	4.Ø	4.3	4.7	5.Ø	6.3
D	Q	t_{PLH}	4.Ø	5.2	6.5	7.7	12.6
		t_{PHL}	3.3	3.6	4.Ø	4.3	5.6
S	Q	t_{PLH}	6.Ø	7.2	8.5	9.7	14.6
R	Q	t_{PHL}	1.2	1.5	1.9	2.2	3.5

Intrinsic Parameters tdx	ktdx
4:5	4.9Ø
3.7	1.3Ø
2.8	4.9Ø
3.Ø	1.3Ø
4.8	4.9Ø
Ø.9	1.3Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

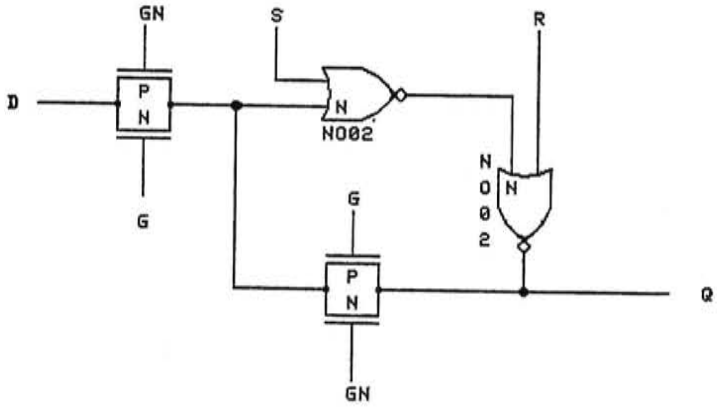
GOULD

AMI Semiconductors

DL18

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

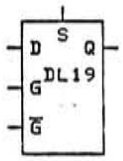
DL19

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Set

Logic Symbol



Truth Table

S	G	GN	D	Q
L	L	H	X	No Change
L	H	L	D	D
H	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
D	2.5
All Other	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .DL19 D G GN S;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	4.3	5.Ø	5.7	6.5	9.4
		t_{PHL}	3.9	4.2	4.6	4.9	6.2
D	Q	t_{PLH}	2.5	3.2	3.9	4.7	7.6
		t_{PHL}	3.2	3.5	3.9	4.2	5.5
S	Q	t_{PLH}	3.4	4.1	4.9	5.6	8.5

Intrinsic Parameters tdx	ktdx
3.6	2.9Ø
3.6	1.3Ø
1.8	2.9Ø
2.9	1.3Ø
2.7	2.9Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

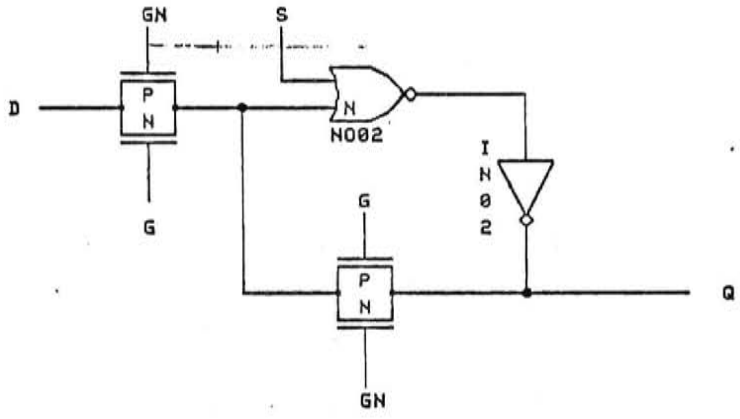
GOULD

AMI Semiconductors

DL19

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

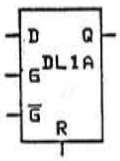
DL1A

August 1984

Description: Pre-Routed Latch With Reset

3U SINGLE - METAL HCMOS GATE ARRAYS

Logic Symbol



Truth Table

R	G	GN	D	Q
L	L	H	X	No Change
L	H	L	D	D
H	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .DL1A D G GN R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	5.6	6.9	8.1	9.3	14.2
		t_{PHL}	3.3	3.5	3.8	4.1	5.2
D	Q	t_{PLH}	4.Ø	5.2	6.4	7.7	12.5
		t_{PHL}	2.5	2.8	3.1	3.4	4.6
R	Q	t_{PHL}	1.2	1.5	1.8	2.2	3.4

Intrinsic Parameters tdx	ktdx
4.4	4.88
3.Ø	1.11
2.8	4.87
2.2	1.20
Ø.9	1.27

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

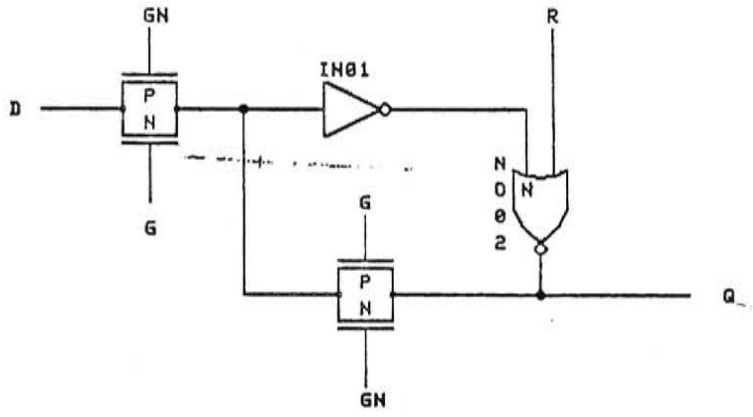
GOULD

AMI Semiconductors

DL1A

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

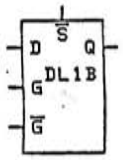
DL1B

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Active Low Set

Logic Symbol



Truth Table

SN	G	GN	D	Q
H	L	H	X	No Change
H	H	L	D	D
L	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .DL1B D G GN SN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	4.3	5.Ø	5.7	6.4	9.2
		t_{PHL}	3.7	4.1	4.6	5.Ø	6.7
D	Q	t_{PLH}	2.6	3.3	4.Ø	4.8	7.7
		t_{PHL}	3.Ø	3.5	3.9	4.3	6.1
SN	Q	t_{PLH}	3.9	4.5	5.2	5.8	8.5

Intrinsic Parameters tdx	ktdx
3.6	2.8Ø
3.3	1.7Ø
1.9	2.9Ø
2.6	1.8Ø
3.2	2.6Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

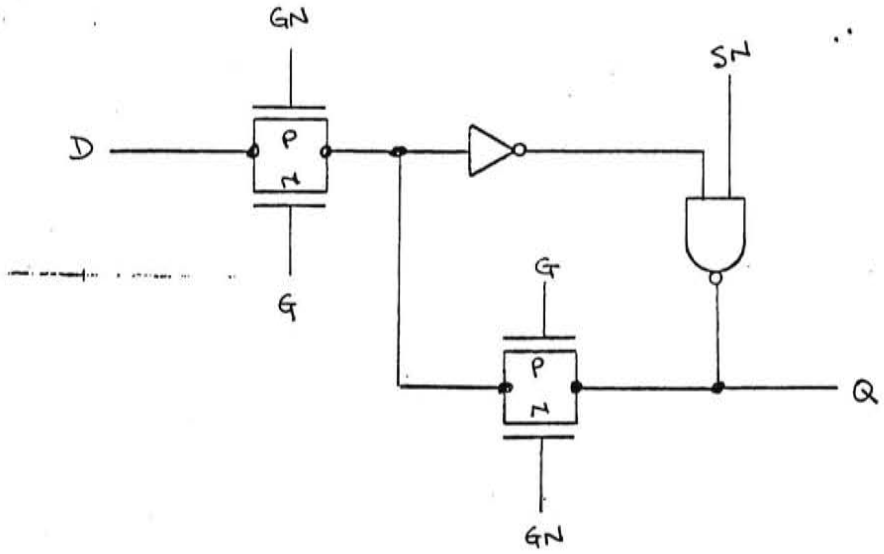
GOULD

AMI Semiconductors

DL1B

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

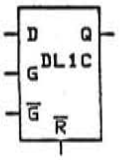
DL1C

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Active Low Reset

Logic Symbol



Truth Table

RN	G	GN	D	Q
H	L	H	X	No Change
H	H	L	D	D
L	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .DL1C D G GN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	4.5	5.3	6.Ø	6.7	9.6
		t_{PHL}	3.4	3.6	3.9	4.2	5.3
D	Q	t_{PLH}	3.Ø	3.7	4.4	5.1	8.Ø
		t_{PHL}	2.6	2.9	3.2	3.5	4.6
RN	Q	t_{PHL}	3.Ø	3.3	3.6	3.9	5.Ø

Intrinsic Parameters tdx	ktdx
3.8	2.87
3.1	1.12
2.2	2.88
2.3	1.15
2.8	1.13

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

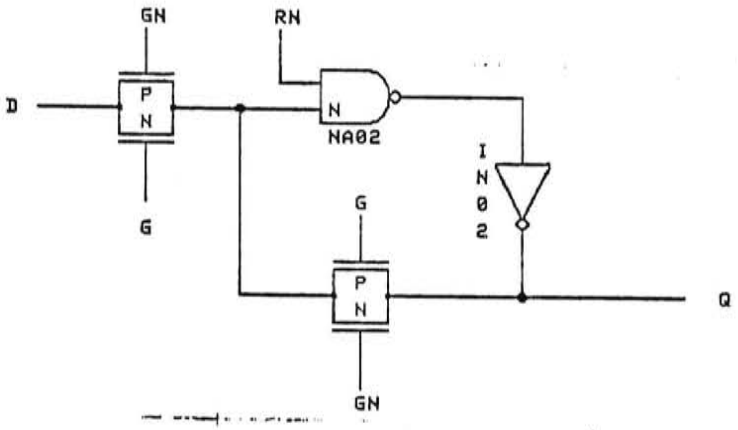
GOULD

AMI Semiconductors

DL1C

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

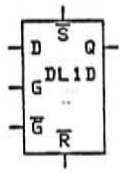
DL1D

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Active Low Set And Reset

Logic Symbol



Truth Table

SN	RN	G	GN	D	Q
H	H	L	H	X	No Change
H	H	H	L	D	D
H	L	X	X	X	L
L	H	X	X	X	H
L	L	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 3.5

Bolt Syntax: Q .DL1D D G GN SN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
D	Q	t_{PLH}	4.8	5.5	6.3	7.Ø	9.9
		t_{PHL}	4.1	4.6	5.2	5.7	7.9
D	Q	t_{PLH}	3.2	3.9	4.7	5.4	8.3
		t_{PHL}	3.5	4.Ø	4.6	5.1	7.2
SN	Q	t_{PLH}	4.1	4.8	5.5	6.1	8.8
RN	Q	t_{PHL}	4.6	5.1	5.7	6.2	8.3

Intrinsic Parameters tdx	ktdx
4.1	2.9Ø
3.5	2.2Ø
2.5	2.9Ø
3.Ø	2.1Ø
3.4	2.7Ø
4.1	2.1Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

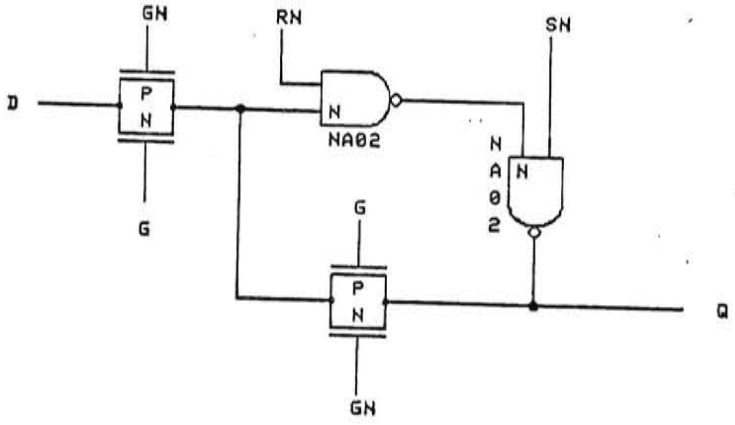
GOULD

AMI Semiconductors

DL1D

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

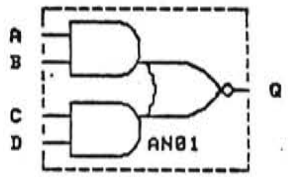
AN01

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed AND-NOR Gates

Logic Symbol



Truth Table

A	B	C	D	Q
H	H	H	H	L
H	H	X	X	L
X	X	H	H	L
All Other				H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .AN01 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	6.6	8.1	9.5	11	16.8
		t_{PHL}	1.3	1.8	2.3	2.7	4.6

Intrinsic Parameters tdx	ktdx
5.2	5.80
0.9	1.84

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

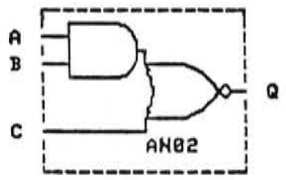
AN02

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed AND-NOR Gate

Logic Symbol



Truth Table

A	B	C	Q
H	H	H	L
H	H	X	L
X	X	H	L
All Other			H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1.5

Bolt Syntax: Q .AN02 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	4.6	6.Ø	7.5	8.9	14.7
		t_{PHL}	Ø.8	1.3	1.7	2.2	4.1

Intrinsic Parameters tdx	ktdx
3.1	5.8Ø
Ø.3	1.86

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

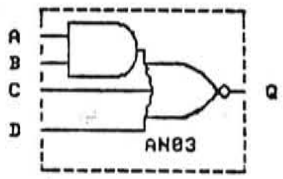
AN03

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed AND-NOR Gate

Logic Symbol



Truth Table

A	B	C	D	Q
H	H	X	X	L
X	X	H	X	L
X	X	X	H	L
X	L	L	L	H
L	X	L	L	H

Input Loading

Logic Input	Equivalent Unit Loads
ANY INPUT	1

Equivalent Gate Count: 2.0

Bolt Syntax: Q .AN03 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	7.7	9.7	11.6	13.6	21.4
		t_{PHL}	1.1	1.6	2.1	2.5	4.4

Intrinsic Parameters tdx	ktdx
5.8	7.80
0.6	1.90

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

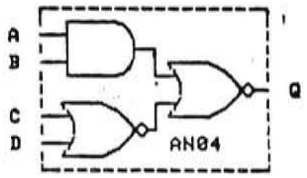
AN04

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed AND-NOR-NOR Gate

Logic Symbol



Truth Table

A	B	C	D	Q
H	H	X	X	L
X	X	L	L	L
X	L	X	H	H
X	L	H	X	H
L	X	X	H	H
L	X	H	X	H

Input Loading

Logic Input	Equivalent Unit Loads
ANY INPUT	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .AN04 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.9	3.2	3.6	3.9	5.3
		t_{PHL}	4.2	5.7	7.1	8.5	14.4

Intrinsic Parameters tdx	ktdx
2.6	1.35
2.8	5.80

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

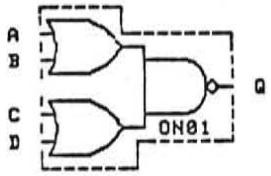
ONØ1

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed OR-OR-NAND Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	L	X	X	H
X	X	L	L	H
L	L	L	L	H
All Other				L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .ONØ1 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.5	6.7	7.9	9.1	14.Ø
		t_{PHL}	1.9	2.5	3.Ø	3.6	5.9

Intrinsic Parameters tdx	ktdx
4.3	4.82
1.3	2.28

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD
AMI Semiconductors

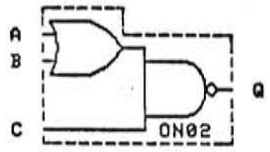
ONØ2

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed OR-NAND Gate

Logic Symbol



Truth Table

A	B	C	Q
X	X	L	H
H	X	H	L
X	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1.5

Bolt Syntax: Q .ONØ2 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	4.Ø	5.2	6.4	7.7	12.6
		t_{PHL}	Ø.8	1.3	1.8	2.3	4.2

Intrinsic Parameters tdx	ktdx
2.8	4.91
Ø.3	1.93

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

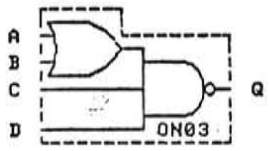
ON03

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed OR-NAND Gate

Logic Symbol



Truth Table

A	B	C	D	Q
X	X	X	L	H
X	X	L	X	H
L	L	H	H	H
X	H	H	H	L
H	X	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.Ø

Bolt Syntax: Q .ONØ3 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.7	6.9	8.1	9.4	14.3
		t_{PHL}	2.Ø	2.7	3.4	4.Ø	6.7

Intrinsic Parameters tdx	ktdx
4.4	4.95
1.3	2.7Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD
AMI Semiconductors

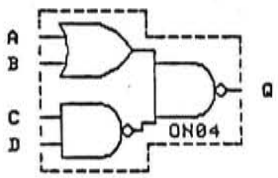
ONØ4

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed OR-NAND-NAND Gate

Logic Symbol



Truth Table

A	B	C	D	Q
X	H	X	L	L
H	X	X	L	L
X	H	L	X	L
H	X	L	X	L
L	L	X	X	H
X	X	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q .ONØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.Ø	2.8	3.5	4.2	7.1
		t_{PHL}	3.Ø	3.5	3.9	4.4	6.5

Intrinsic Parameters tdx	ktdx
1.3	2.9Ø
2.5	2.ØØ

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

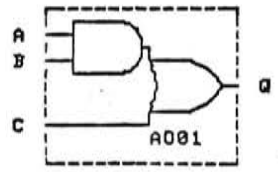
AØØ1

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed AND-OR Gate

Logic Symbol



Truth Table

A	B	C	Q
H	H	H	H
H	H	X	H
X	X	H	H
All Other			L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.Ø

Bolt Syntax: Q .AØØ1 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.2	2.9	3.6	4.4	7.3
		t_{PHL}	4.1	4.4	4.8	5.2	6.7

Intrinsic Parameters tdx	ktdx
1.5	2.9Ø
3.7	1.5Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

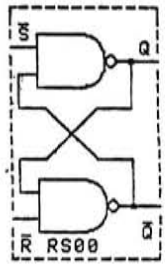
RSØØ

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Set-Reset NAND Gate Latch

Logic Symbol



Truth Table

SN	RN	Q	QN
L	L	H	H
L	H	H	L
H	L	L	H
H	H	No Change	

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q QN .RSØØ RN SN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.6	3.3	4.Ø	4.7	7.4
		t_{PHL}	2.8	3.Ø	3.2	3.4	4.3

Intrinsic Parameters tdx	ktdx
2.Ø	2.72
2.6	Ø.82

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

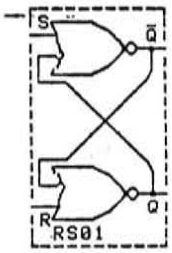
RSØ1

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Set-Reset NOR Gate Latch

Logic Symbol



Truth Table

S	R	Q	QN
H	H	L	L
H	L	H	L
L	H	L	H
L	L	No Change	

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2.5

Bolt Syntax: Q QN .RSØ1 R S;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.4	4.6	5.8	7.1	12.Ø
		t_{PHL}	Ø.6	Ø.9	1.2	1.5	2.8

Intrinsic Parameters tdx	ktdx
2.1	4.92
Ø.2	1.26

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

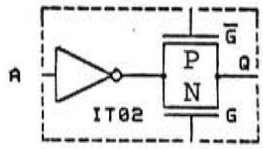
IT02

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Inverter Driving A Transmission Gate.

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	AN

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .IT02 A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	2.3	3.3	4.4	5.4	9.5
		t_{PHL}	1.2	1.7	2.3	2.8	5.0
A	Q	t_{PLH}	3.4	4.4	5.5	6.5	10.6
		t_{PHL}	1.1	1.6	2.1	2.6	4.6

Intrinsic Parameters tdx	ktdx
1.3	4.10
0.6	2.20
2.4	4.10
0.6	2.00

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

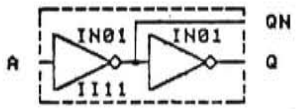
III1

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With A Single Inverter Followed By One Inverter in Parallel

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 1

Bolt Syntax: Q QN .III1 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	2.Ø	3.1	4.2	5.3	9.8
		t_{PHL}	2.5	3.5	4.5	5.5	9.5
A	QN	t_{PLH}	2.1	2.8	3.6	4.3	7.3
		t_{PHL}	Ø.4	Ø.7	1.1	1.4	2.8

Intrinsic Parameters tdx	ktdx
.9	4.47
1.5	4.ØØ
1.4	2.93
Ø.1	1.35

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

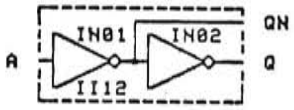
II12

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With A Single Inverter Followed By Two Inverter in Parallel

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 1

Bolt Syntax: Q QN .II12 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.6	2.3	3.1	3.8	6.8
		t_{PHL}	2.5	3.2	3.9	4.7	7.7
A	QN	t_{PLH}	2.8	3.5	4.2	4.9	7.7
		t_{PHL}	Ø.6	Ø.9	1.3	1.6	2.9

Intrinsic Parameters tdx	ktdx
Ø.8	3.ØØ
1.7	3.ØØ
2.1	2.78
Ø.3	1.32

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

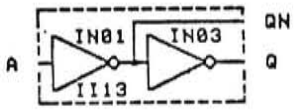
II13

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With A Single Inverter Followed By Three Invertes In Parallel.

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 2

Bolt Syntax: Q QN II13 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.9	2.6	3.2	3.9	6.5
		t_{PHL}	2.9	3.6	4.3	5.Ø	7.9
A	QN	t_{PLH}	3.6	4.1	4.8	5.5	8.4
		t_{PHL}	Ø.8	1.1	1.5	1.8	3.1

Intrinsic Parameters tdx	ktdx
1.3	2.6Ø
2.2	2.85
2.7	2.85
Ø.5	1.32

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

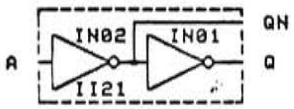
II21

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Dirver With Two Inverters In Parallel Followed By a Single Inverter.

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 2

Bolt Syntax: Q QN II21 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.3	2.2	3.2	4.1	7.9
		t_{PHL}	1.7	2.4	3.Ø	3.7	6.3
Q	A	t_{PLH}	1.2	1.6	2.Ø	2.4	3.9
		t_{PHL}	Ø.2	Ø.4	Ø.6	Ø.8	1.6

Intrinsic Parameters tdx	ktdx
Ø.3	3.8Ø
1.Ø	2.65
Ø.8	1.55
Ø.Ø	Ø.8Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

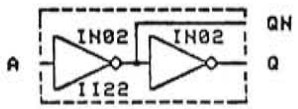
II22

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With Two Inverters In Parallel Followed By Two Inverters In Parallel.

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II22 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.2	1.8	2.4	3.Ø	5.4
		t_{PHL}	1.8	2.3	2.8	3.3	5.4
A	QN	t_{PLH}	1.5	1.9	2.3	2.7	4.2
		t_{PHL}	Ø.2	Ø.4	Ø.6	Ø.9	1.7

Intrinsic Parameters tdx	ktdx
Ø.6	2.42
1.3	2.Ø5
1.1	1.55
Ø.Ø	Ø.87

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

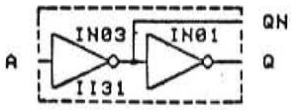
II31

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Drivers With Three Parallel Inverters Followed By a Single Inverter.

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II31 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.4	2.3	3.2	4.1	7.7
		t_{PHL}	1.8	2.4	3.Ø	3.6	5.9
A	QN	t_{PLH}	1.3	1.5	1.8	2.1	3.4
		t_{PHL}	Ø.2	Ø.4	Ø.5	Ø.7	1.3

Intrinsic Parameters tdx	ktdx
Ø.4	3.65
1.2	2.35
1.Ø	1.2Ø
Ø.Ø	Ø.66

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

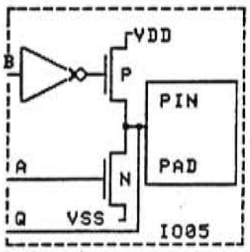
I005

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Tri-State Input - Output Buffer

Logic Symbol



Truth Table

A	B	Q
L	H	H
L	L	Z
H	L	L
H	H	Illegal

Input Loading

Logic Input	*Equivalent Unit Loads
B	3
A	4

* .25 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used.

Bolt Syntax: Q .IOØ5 A B;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

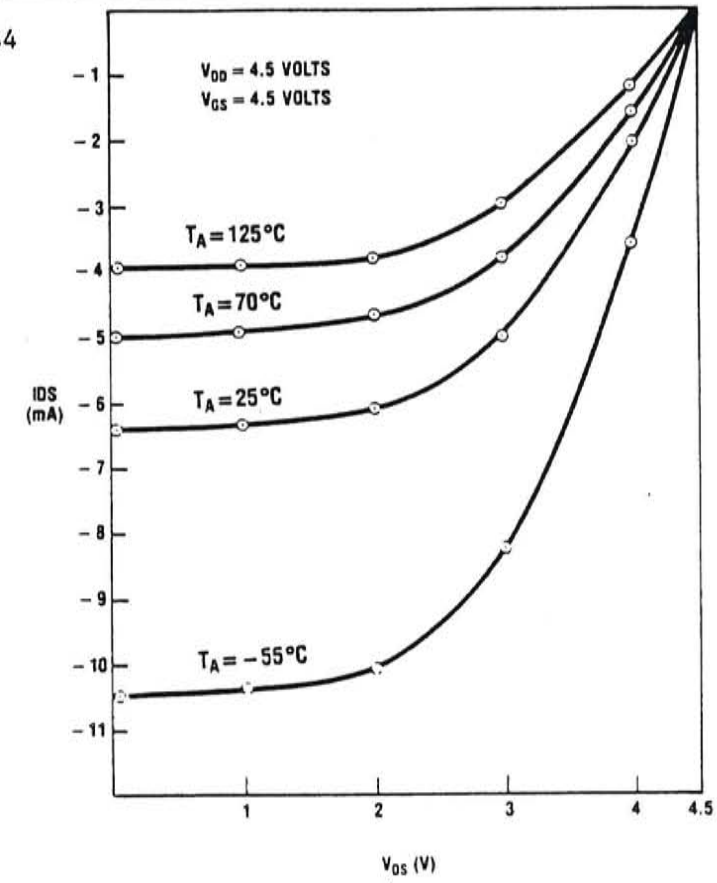
Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf	Delay Coefficients tdx	ktdx
A	Q	t_{PLH}	9.3	16.2	23.1	2.5	Ø.14
		t_{PHL}	5.6	1Ø.4	15.2	Ø.8	Ø.1Ø
ENABLE	Q	t_{PZH}	8.4	15.2	22.1	1.5	Ø.14
		t_{PZL}	5.7	11.4	17.Ø	Ø.1	Ø.11
DISABLE	Q	t_{PLZ}		Ø.6		Ø.6	Ø.Ø
		t_{PHZ}		1.8		1.8	Ø.Ø

Symbol	Parameters	Conditions	Min.	Typ.	Max.
IDSS	Output Leadage Current		-1Ø µA		+1Ø µA
VOL	Low Level Output Voltage	IOL=4.Ø mA			.4V
	High Level Output Voltage	IOH=5.Ø mA	2.4V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

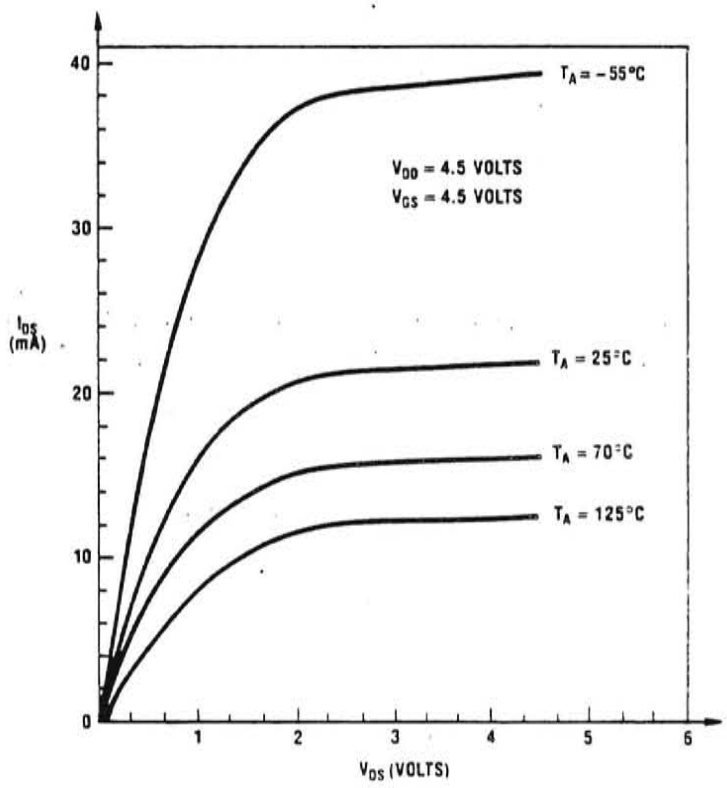
August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



P-CHANNEL DRIVER

SOURCING CHARACTERISTICS



N-CHANNEL DRIVER

SINKING CHARACTERISTICS

GOULD

AMI Semiconductors

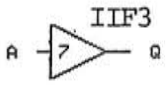
IIF3

August 1984

3μ SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed TTL Lever Translator

Logic Symbol



Truth Table

A	Q
H	H
L	L

Input Loading

Logic Input	Equivalent Unit Loads
A	4

Equivalent Gate Count: 2.5

Bolt Syntax: Q .IIF3 A;

Switching Characteristics:

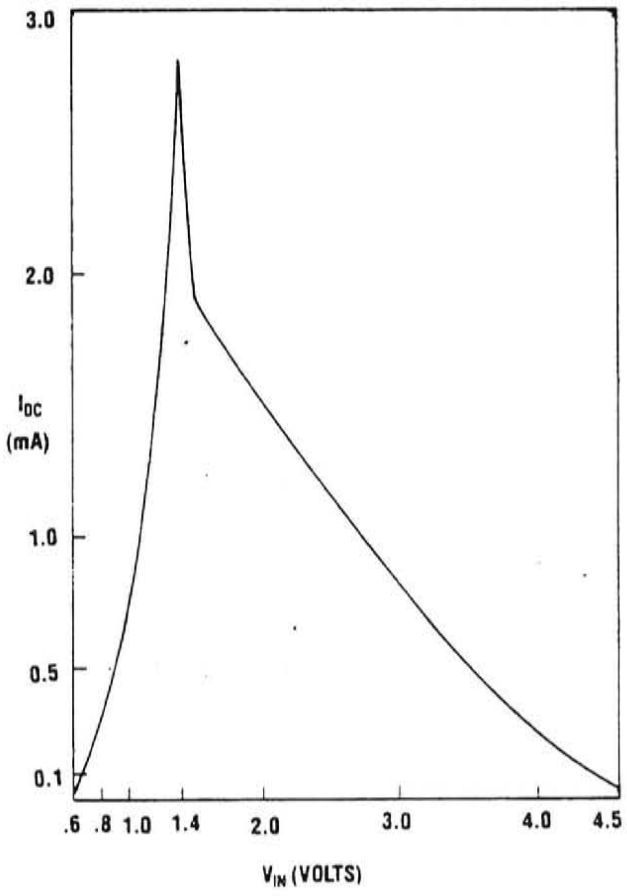
Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.3	6.0	6.7	7.5	10.4
		t_{PHL}	3.9	4.2	4.6	4.9	6.3

Intrinsic Parameters tdx	ktdx
4.6	2.9Ø
3.5	1.4Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance



I_{DD} vs. V_{IN} For TTL Level Translators

GOULD

AMI Semiconductors

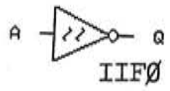
IIFØ

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Inverting Schmitt Trigger

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 1.5

Bolt Syntax: Q.IIFØ A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	4.6	5.9	7.3	8.6	12.4
		t_{PHL}	2.2	2.7	3.3	3.8	4.6

Intrinsic Parameters tdx	ktdx
3.3	5.4Ø
1.7	2.1Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .25 pf including statistical wiring capacitance

	VIH	VIL	HYSTERISIS
Trip Points	3.ØV	1.6V	1.4V

GOULD

AMI Semiconductors

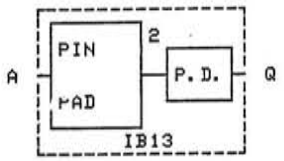
IB13

August 1984

3W SINGLE - METAL HCMOS
GATE ARRAYS

Description: Input pad with Protection Diode.

Logic Symbol



Truth Table

A	Q
H	H
L	L

Input Loading

Logic Input	Equivalent Unit Loads
N.A.	

Equivalent Gate Count: Contained within peripheral cells, no core devices used.

Bolt Syntax: Q .IB13 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH} t_{PHL}					

Intrinsic Parameters tdx	ktdx

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

NOTE: Delay not applicable because of external drive.

GOULD

AMI Semiconductors

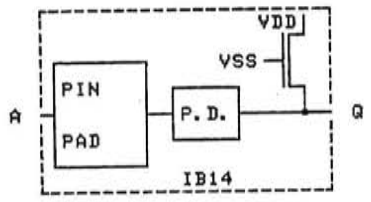
IB14

August 1984

3W SINGLE - METAL HCMOS
GATE ARRAYS

Description: Input Pad With Protection Diode And P-Channel Transistor

Logic Symbol



Truth Table

A	Q
H	H
L	L

Input Loading

Logic Input	Equivalent Unit Loads
N.A.	

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used

Bolt Syntax: Q .IB14 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

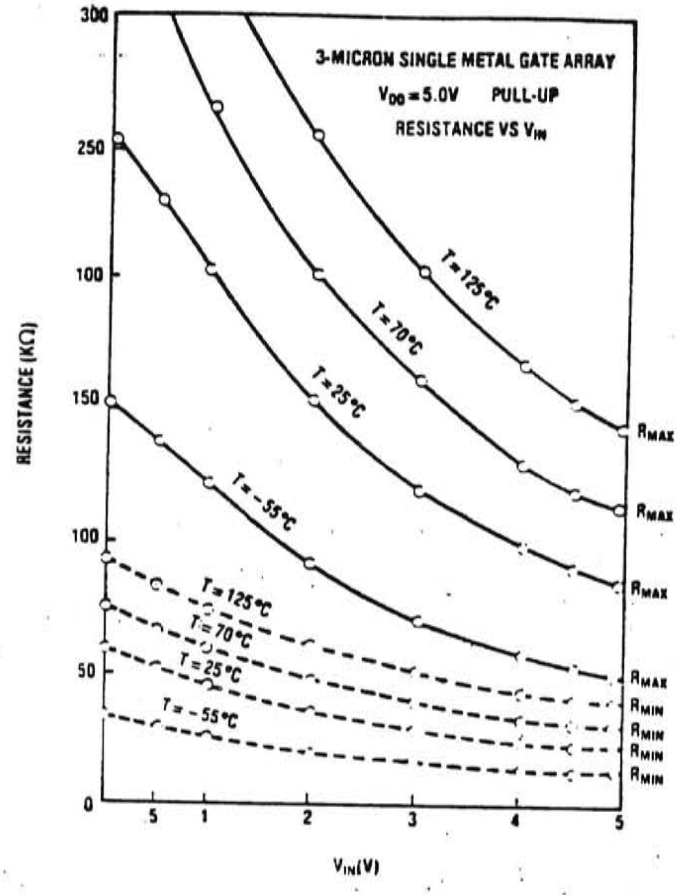
Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH} t_{PHL}					

Intrinsic Parameters tdx	ktdx

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

NOTE: Delay not applicable because of external drive.



Gate-Array Pull-Up Characteristics

GOULD

AMI Semiconductors

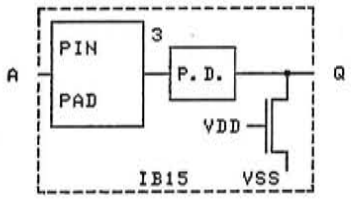
IB15

August 1984

3W SINGLE - METAL HCMOS
GATE ARRAYS

Description: Input Pad With Protection Diode And N-Channel Transistor

Logic Symbol



Truth Table

A	Q
H	H
L	L

Input Loading

Logic Input	Equivalent Unit Loads
N.A.	

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used

Bolt Syntax: Q .IB15 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

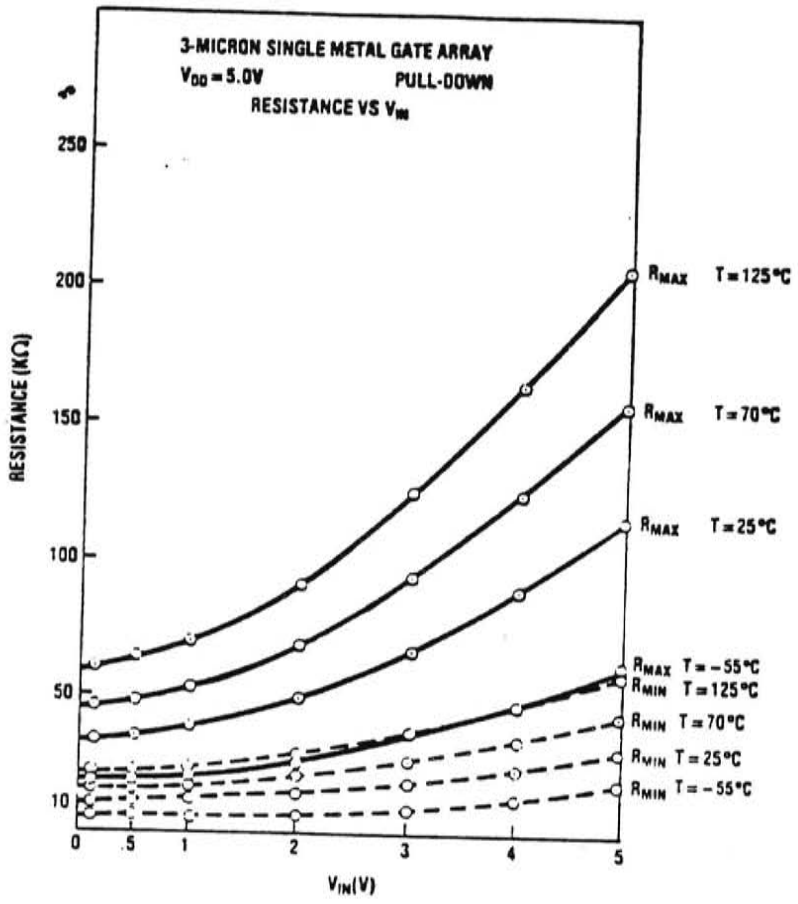
Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH} t_{PHL}					

Intrinsic Parameters tdx	ktdx

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .25 pf including statistical wiring capacitance

NOTE: Delay not applicable because of external drive.



Gate Array Pull-Down Characteristics

GOULD

AMI Semiconductors

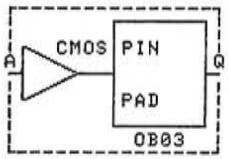
OBØ3

August 1984

Description: CMOS Output Buffer

3U SINGLE - METAL HCMOS GATE ARRAYS

Logic Symbol



Truth Table

A	Q
L	L
H	H

Input Loading

Logic Input	*Equivalent Unit Loads
A	3

*.25 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used.

Bolt Syntax: Q .OBØ3 A;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf	Delay Coefficients tdx	ktdx
A	Q	t_{PLH}	9.Ø	15.6	22.2	2.3	Ø.13
		tPHL	6.8	11.2	15.7	2.3	Ø.Ø9
A (I TTL LOAD)	Q	t_{PLH}	5.9	9.8	13.8	1.9	Ø.Ø8
A (I TTL LOAD)	Q	t_{PHL}	1Ø	17.3	24.6	2.7	Ø.15

Symbol	Parameters	Conditions	Min.	Typ.	Max.
LDSS	Output Leakage Current		-1Ø µA		+1Ø µA
VOL	Low Level Output Voltage	IOL=1 µA			.Ø5V
VOH	High Level Output Voltage	IOH=1 µA	4.95V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

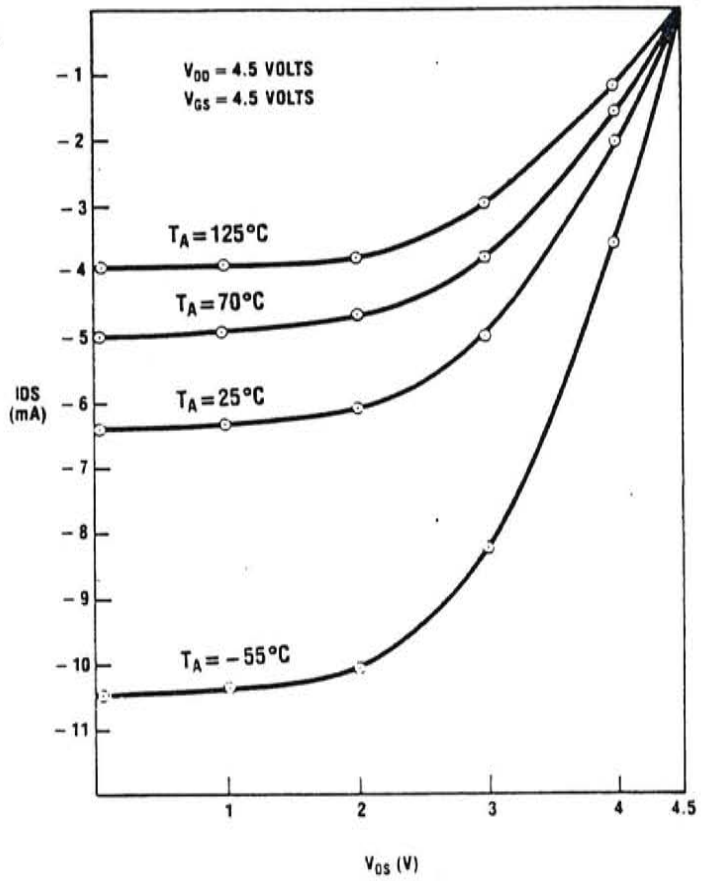
GOULD

AMI Semiconductors

OBØ3

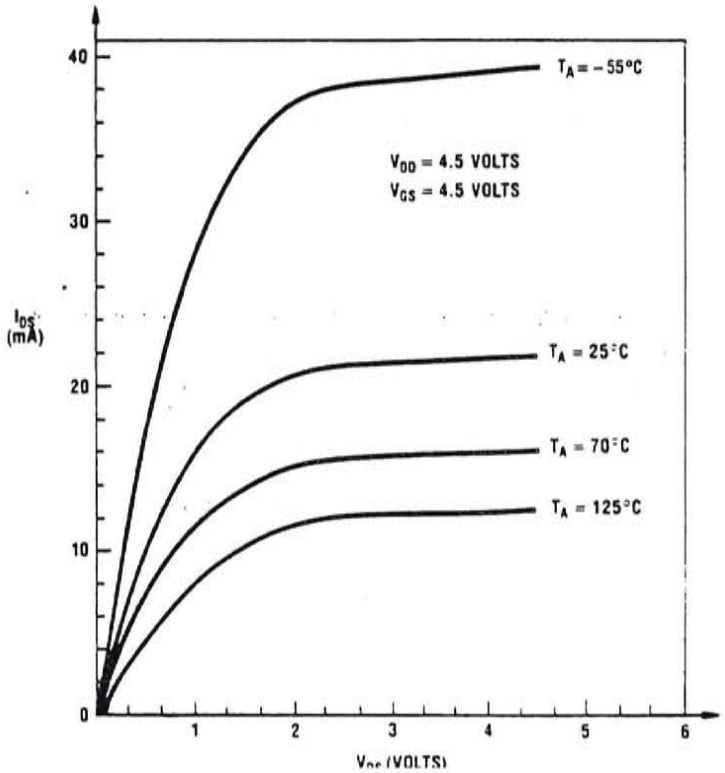
August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



P-CHANNEL DRIVER

SOURCING CHARACTERISTICS



N-CHANNEL DRIVER

SINKING CHARACTERISTICS

GOULD

AMI Semiconductors

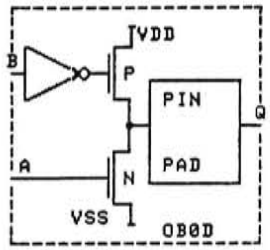
OBØD

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Tri-State Output Buffers

Logic Symbol



Truth Table

A	B	Q
L	H	H
L	L	Z
H	L	L
H	H	Illegal

Input Loading

Logic Input	*Equivalent Unit Loads
B	3
A	4

*.25 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used.

Bolt Syntax: Q .OBØD A B;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf	Delay Coefficients tdx	ktdx
A	Q	t_{PLH}	9.3	16.2	23.1	2.5	Ø.14
		t_{PHL}	5.6	1Ø.4	15.2	Ø.8	Ø.1Ø
ENABLE	Q	t_{PZH}	8.4	15.2	22.1	1.5	Ø.14
		t_{PZL}	5.7	11.4	17.0	Ø.1	Ø.11
DISABLE	Q	t_{PLZ}		Ø.6		Ø.6	Ø.Ø
		t_{PHZ}		1.8		1.8	Ø.Ø

Symbol	Parameters	Conditions	Min.	Typ.	Max.
LDSS	Output Leakage Current		-1Ø µA		+1Ø µA
VOL	Low Level Output Voltage	IOL=4.Ø mA			.4V
VOH	High Level Output Voltage	IOH=5.Ø mA	2.4V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

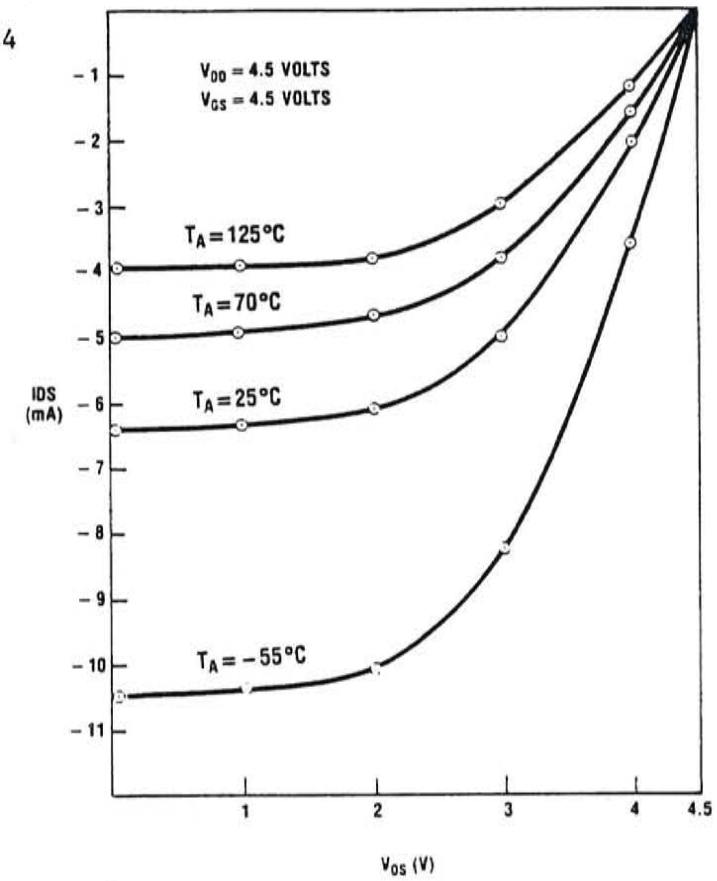
GOULD

AMI Semiconductors

OBØD

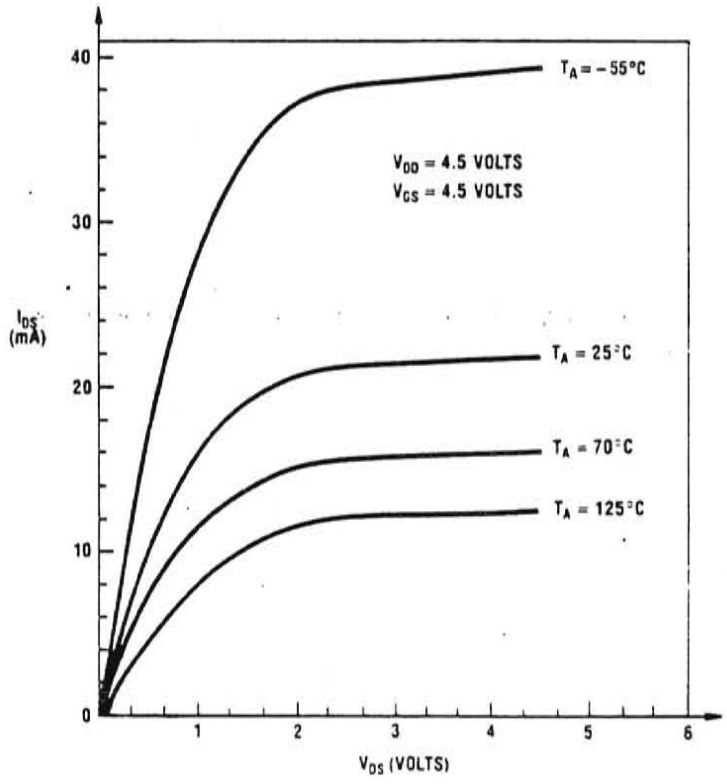
August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



P-CHANNEL DRIVER

SOURCING CHARACTERISTICS



N-CHANNEL DRIVER

SINKING CHARACTERISTICS

GOULD

AMI Semiconductors

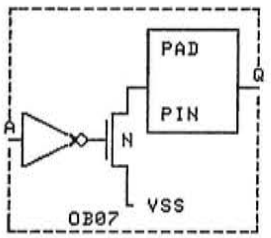
OBØ7

August 1984

Description: Open Drain N-Channel Output Buffer

3U SINGLE - METAL HCMOS GATE ARRAYS

Logic Symbol



Truth Table

A	Q
H	Z
L	L

Input Loading

Logic Input	*Equivalent Unit Loads
A	3

*.25 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cell, No Core Devices Used.

Bolt Syntax: Q .OBØ7 A;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf
A (IK RESISTOR)	Q	t_{PHL}	5.8	1Ø.4	15.Ø
A (10K RESISTOR)	Q	t_{PHL}	5.6	1Ø.Ø	14.4
A	Q	t_{PLZ}		Ø.6	

Delay Coefficients tdx	ktdx
1.3	Ø.Ø9
1.2	Ø.Ø9
Ø.6	

Symbol	Parameters	Conditions	Min.	Typ.	Max.
IDSS	Output Leakage Current		-1Ø µA		+1Ø µA
VOL	Low Level Output Voltage	IOL-4.Ø mA			.4V
VOH	High Level Output Voltage	IOL=5.Ø mA	2.4V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

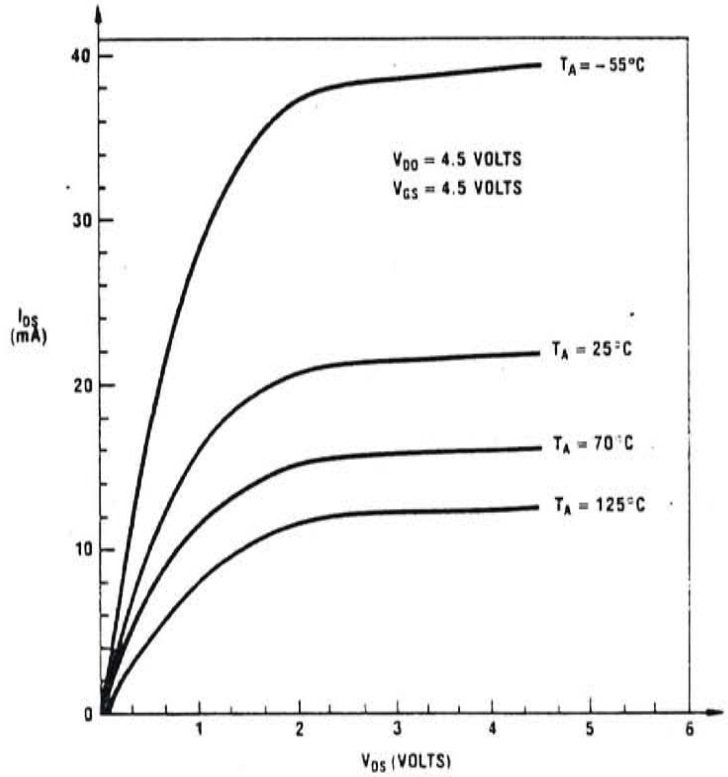
GOULD

AMI Semiconductors

0B07

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



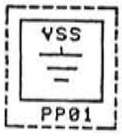
N-CHANNEL DRIVER

SINKING CHARACTERISTICS

PPØ1

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



.PP01

NOTE: ONE PPØ1 MUST BE USED PER GROUND (VSS) PIN.

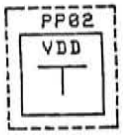
GOULD

AMI Semiconductors

PPØ2

August 1984

3U SINGLE - METAL HCMOS
GATE ARRAYS



.PP02

NOTE: ONE PPØ2 MUST BE USED PER POWER (VDD) PIN.

GOULD

AMI Semiconductors

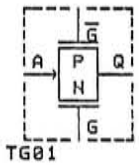
TGØ1

August 1984

3U SINGLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Transmission Gate

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	A

Input Loading

Logic Input	Equivalent Unit Loads
A	2
G,GN	1

Equivalent Gate Count: 0.5

Bolt Syntax: Q. TGØ1 A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.4	1.9	2.4	3.0	4.56
		t_{PHL}	0.9	1.3	1.7	2.1	3.74
G,GN	Q	t_{PLH}	3.8	5.2	6.6	8.1	13.78
		t_{PHL}	2.4	2.8	3.3	3.7	5.38

Intrinsic Parameters tdx	ktdx
0.3	2.13
0.5	1.62
2.3	5.74
2.0	1.69

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

3UM DOUBLE METAL GATE ARRAY DATA SHEETS

The Gould AMI 3um Double Metal Gate Array Macro Library is a collection of high performance digital integrated circuit building blocks. It is intended for use by logic, circuit, and MOS IC designers. Thus, anyone familiar with standard logic design methods can successfully design a gate array using these macros.

The 3um Double Metal Macro Cells are implemented with a p-well, CMOS process. They are intended primarily for 5 Volt (+10%) operation but will operate at voltages as low as 2.5 volts with reduced performance. Below is a description of the data and terms used in the data sheets.

- **PRE-ROUTED MACROS** are the simpler functional blocks such as the basic gates, latches, and flip-flops. Since these macros have a pre-defined layout they can be accurately characterized for propagation delays.

- **SOFTWARE MACROS** are macros which incorporate several pre-routed macros to form more complicated functional blocks such as a counters or shift registers. These macros are not pre-defined with respect to layout, so the automatic placement and routing software maintains optimum layout flexibility. Since layout is not pre-defined, estimates of propagation delays through these macros are arrived at by adding the delays of the individual pre-routed macros used in the software macros.

- **THE CELL NAME** for each functional macro is displayed in the upper right hand corner of each data sheet. This is the same name that appears in the Bolt invocation.

- **THE LOGIC SYMBOL** for each macro is displayed as it appears in the gate array library.

- **A TRUTH TABLE** description of the logical functions for the cells is provided.

- **THE TABLE OF INPUT LOADING** gives the number of equivalent unit loads for each logical input of the pre-routed macros.

- **THE EQUIVALENT GATE COUNT** is the number of equivalent two input NAND gates in each cell.

- **BOLT SYNTAX** is the invocation syntax of each macro cell. Notice these statements always end with a semi-colon, and the order of these inputs and outputs must be maintained.

- **THE SWITCHING CHARACTERISTICS** give the propagation delay as a function of unit load. The unit load is the capacitance associated with a gate pair (i.e. the gate capacitance of an N-channel and a P-channel core transistor), plus an associated metal interconnection capacitance. All data is given for typical process, temperature, and power supply conditions. Both low-to-high transitions are shown. To find delays under other operating conditions derating curves must be used.

- **THE PROPAGATION DELAY EQUATION** is used for calculating propagation delays when the load is different from those explicitly given in the table. The intrinsic propagation delay, tdx, is used when there is no output loading, and ktdx, is a capacitive multiplication factor. As an example consider the two input NAND Gate with a fan-out of 5. To calculate the propagation delay for the high-to-low transition use tdx = Ø.6 and ktdx = 2.1 so,

  $$tPHL = Ø.6 + (2.1)(.21)(5) = 2.8 \text{ ns.}$$

- **A LOGIC SCHEMATIC** is shown for the more complicated pre-routed macros and the software macros not shown.

- **POWER SUPPLY MACROS** (PPØ1,PPØ2) must be shown on each circuit, while no connections to these macros are necessary one macro must be shown per power or ground pin.

- **DERATING CURVES** are provided to account for changes in propagation delay as process, temperature and power supply vary.

3u DOUBLE METAL GATE ARRAY MACROS

1.	IN01	Single Pre-Routed Inverter
2.	IN02	Two Pre-Routed Inverters in Parallel
3.	IN03	Three Pre-Routed Inverters in Parallel
4.	IN04	Four Pre-Routed Inverters in Parallel
5.	IN06	Six Pre-Routed Inverters in Parallel
6.	NA02	Pre-Routed Two Input Nand Gate
7.	NA03	Pre-Routed Three Input Nand Gate
8.	NA04	Pre-Routed Four Input Nand Gate
9.	NA06	Pre-Routed Six Input Nand Gate
10.	NO02	Pre-Routed Two Input Nor Gate
11.	NO03	Pre-Routed Three Input Nor Gate
12.	NO04	Pre-Routed Four Input Nor Gate
13.	NO06	Pre-Routed Six Input Nor Gate
14.	AA02	Pre-Routed Two Input And Gate
15.	AA03	Pre-Routed Three Input And Gate
16.	AA04	Pre-Routed Four Input And Gate
17.	AA05	Pre-Routed Five Input And Gate
18.	OR02	Pre-Routed Two Input Or Gate
19.	OR03	Pre-Routed Three Input Or Gate
20.	OR04	Pre-Routed Four Input Or Gate
21.	OR05	Pre-Routed Five Input Or Gate
22.	EN01	Pre-Routed Exclusive Nor Gate
23.	EO01	Pre-Routed Exclusive Or Gate
24.	DF01	Pre-Routed D Flip Flop
25.	DF02	Pre-Routed D Flip Flop With Asynchronous Active Low Set
26.	DF03	Pre-Routed D Flip Flop With Asynchronous Active Low Reset
27.	DF04	Pre-Routed D Flip Flop With Asynchronous Active Low Set And Active Low Reset
28.	DF05	Pre-Routed D Flip Flop With Asynchronous Set
29.	DF06	Pre-Routed D Flip Flop With Asynchronous Reset
30.	DF07	Pre-Routed D Flip Flop With Asynchronous Set and Reset
31.	DL01	Pre-Routed Latch with Q and QN
32.	DL18	Pre-Routed Latch With Set and Reset
33.	DL19	Pre-Routed Latch With Set
34.	DL1A	Pre-Routed Latch With Reset
35.	DL1B	Pre-Routed Latch With Active Low Set
36.	DL1C	Pre-Routed Latch With Active Low Reset
37.	DL1D	Pre-Routed Latch With Acive Low Set and Reset
38.	AN01	Pre-Routed And Nor Gates
39.	AN02	Pre-Routed And Nor Gate
40.	AN03	Pre-Routed And Nor Gate
41.	AN04	Pre-Routed And-Nor-Nor Gate
42.	ON01	Pre-Routed Or-Nand Gate
43.	ON02	Pre-Routed Or-Nand Gate
44.	ON03	Pre-Routed Or-Nand Gate
45.	ON04	Pre-Routed Or-Nand-Nand Gate
46.	AO01	Pre-Routed And-Or Gate
47.	RS00	Pre-Routed Set-Reset Nand Gate Latch
48.	RS01	Pre-Routed Nor S R Latch

49. IT02 Pre-Routed Inverter Driving A Transmission Gate
50. Two Pre-Routed Inverters In Parallel Driving A Transmission Gate
51. Three Pre-Routed Invertes In Parallel Driving A Transmission Gate
52. Four Pre-Routed Inverters In Parallel Driving A Transmission Gate
53. Six Pre-Routed Inverters In Parallel Driving A Transmission Gate
54. II11 Pre-Routed Clock Driver With A Single Inverter Followed By A Single Inverter
55. II12 Pre-Routed Clock Driver With A Single Inverter Followed By Two Inverters in Parallel
56. II13 Pre-Routed Clock Driver With A Single Inverter Followed By Three Inverters in Parallel
57. II21 Pre-Routed Clock Driver With Two Inverters In Parallel Followed By A Single Inverter
58. II22 Pre-Routed Clock Driver With Two Inverters In Parallel Followed by Two Inverters In Parralel
59. II31 Pre-Routed Clock Drive WIth Three Inverters In Parallel Followed By A Single Inverter
60. IB02 Pre-Routed CMOS Inverting Input Buffer
61. IB04 Pre-Routed CMOS Inverting Input Buffer With Pull-Up
62. IB06 Pre-Routed CMOS Inverting Input Buffer With Pull-Down
63. IB08 Pre-Routed TTL Inverting Input Buffer
64. IB0A Pre-Routed TTL Inverting Input Buffer With Pull-Up
65. IB0C Pre-Routed TTL Inverting Input Buffer With Pull-Down
66. IO05 Tri-State Bidirectional Buffer
67. OB01 TTL Output Buffer
68. OB03 CMOS Output Buffer
69. OB0D Tri-State Output Buffer
70. OB07 Open Drain Output Buffer
71. OB05 Open Drain Output Buffer
72. PP01 VSS Ground Pin
73. PP02 VDD Power Pin

Process And Temperature Derating Curves
Power Supply Derating Curves

GOULD

AMI Semiconductors

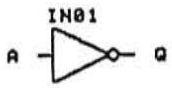
INØ1

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description : Single Pre-Routed Inverter

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 1

Bolt Syntax: Q .INØ1 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.3	1.9	2.4	2.9	5.Ø
		t_{PHL}	Ø.6	Ø.9	1.3	1.6	3.0

Intrinsic Parameters tdx	ktdx
Ø.8	2.52
Ø.3	1.58

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

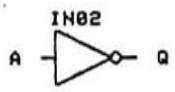
INØ2

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Two Pre-Routed Inverters In Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 1

Bolt Syntax: Q .INØ2 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	Ø.8	1.1	1.4	1.6	2.7
		t_{PHL}	Ø.2	Ø.4	Ø.6	Ø.8	1.6

Intrinsic Parameters tdx	ktdx
Ø.5	1.31
Ø.Ø	.96

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

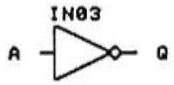
IN03

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Three Pre-Routed Inverters In Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: 2

Bolt Syntax: Q.IN03 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 4	8	12	16	20
A	Q	t_{PLH}	1.4	2.1	2.8	3.5	4.3
		t_{PHL}	0.6	1.1	1.6	2.2	2.7

Intrinsic Parameters tdx	ktdx
0.6	0.9
0.0	0.6

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

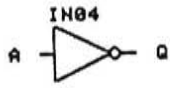
IN04

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description : Four Pre-Routed Inverters In Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	4

Equivalent Gate Count: 2

Bolt Syntax: Q .IN04 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 4	8	12	16	20
A	Q	t_{PLH}	1.1	1.7	2.3	2.9	3.5
		t_{PHL}	0.4	0.9	1.4	1.9	2.4

Intrinsic Parameters tdx	ktdx
0.5	0.71
0.0	0.58

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

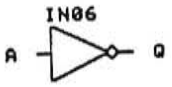
IN06

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description : Six Pre-Routed Inverters In Parallel

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	6

Equivalent Gate Count: 3

Bolt Syntax: Q .IN06 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 4	8	12	16	20
A	Q	t_{PLH}	Ø.9	1.4	1.8	2.2	2.7
		t_{PHL}	Ø.2	Ø.5	Ø.8	1.1	1.5

Intrinsic Parameters tdx	ktdx
Ø.5	Ø.52
Ø.Ø	Ø.39

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

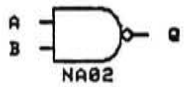
NAØ2

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Two Input Nand Gate

Logic Symbol



Truth Table

A	B	Q
L	X	H
X	L	H
H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1

Bolt Syntax: Q . NAØ2 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.8	2.3	2.8	3.4	5.4
		t_{PHL}	1.0	1.5	1.9	2.4	4.2

Intrinsic Parameters tdx	ktdx
1.3	2.49
0.6	2.14

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

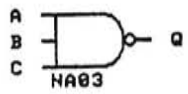
NAØ3

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Three Input Nand Gate

Logic Symbol



Truth Table

A	B	C	Q
L	X	X	H
X	L	X	H
X	X	L	H
H	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .NAØ3 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.5	3.0	3.6	4.1	6.3
		t_{PHL}	2.1	2.8	3.4	4.1	6.7

Intrinsic Parameters tdx	ktdx
1.9	2.62
1.5	3.1Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

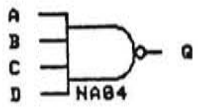
NAØ4

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Four Input Nand Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	X	X	X	H
X	L	X	X	H
X	X	L	X	H
X	X	X	L	H
H	H	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	.1

Equivalent Gate Count: 2

Bolt Syntax: Q .NAØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.7	3.3	3.9	4.4	6.7
		t_{PHL}	3.0	3.9	4.7	5.5	8.8

Intrinsic Parameters tdx	ktdx
2.2	2.69
2.2	3.95

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD
AMI Semiconductors

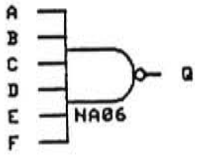
NAØ6

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Six Input Nand Gate

Logic Symbol



Truth Table

A	B	C	D	E	F	Q
L	X	X	X	X	X	H
X	L	X	X	X	X	H
X	X	L	X	X	X	H
X	X	X	L	X	X	H
X	X	X	X	L	X	H
X	X	X	X	X	L	H
H	H	H	H	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .NAØ6 A B C D E F;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.3	4.0	4.6	5.2	7.7
		t_{PHL}	5.9	7.2	8.4	9.6	14.5

Intrinsic Parameters tdx	ktdx
2.7	2.96
4.7	5.81

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

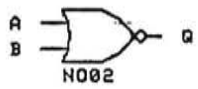
NØØ2

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description : Pre-Routed Two Input Nor Gate

Logic Symbol



Truth Table

A	B	Q
L	L	H
X	H	L
H	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 1

Bolt Syntax: Q .NØØ2 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.4	3.4	4.3	5.2	8.9
		t_{PHL}	0.8	1.1	1.4	1.8	3.1

Intrinsic Parameters tdx	ktdx
1.5	4.39
0.5	1.55

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

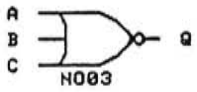
NØØ3

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Three Input Nor Gate

Logic Symbol



Truth Table

A	B	C	Q
L	L	L	H
X	X	H	L
X	H	X	L
H	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .NØØ3 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	4.7	6.1	7.5	8.9	14.6
		t_{PHL}	1.0	1.3	1.6	1.9	3.1

Intrinsic Parameters tdx	ktdx
3.3	6.72
0.7	1.42

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

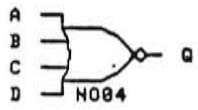
NØØ4

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Four Input Nor Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	L	L	L	H
X	X	X	H	L
X	X	H	X	L
X	H	X	X	L
H	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .NØØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	7.5	9.3	11.1	12.8	2Ø.Ø
		t_{PHL}	1.1	1.4	1.8	2.1	3.3

Intrinsic Parameters tdx	ktdx
5.7	8.55
0.8	1.5Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

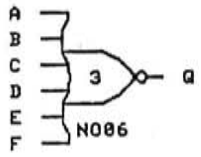
NØØ6

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Six Input Nor Gate

Logic Symbol



Truth Table

A	B	C	D	E	F	Q
L	L	L	L	L	L	H
X	X	X	X	X	H	L
X	X	X	X	H	X	L
X	X	X	H	X	X	L
X	X	H	X	X	X	L
X	H	X	X	X	X	L
H	X	X	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .NOØ6 A B C D E F;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	14.6	17.3	20.0	22.7	33.6
		t_{PHL}	1.3	1.6	1.9	2.2	3.4

Intrinsic Parameters tdx	ktdx
11.9	12.9
1.0	1.4

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

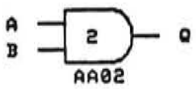
AA02

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Two Input And Gate

Logic Symbol



Truth Table

A	B	Q
L	X	L
X	L	L
H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .AA02 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.2	2.7	3.2	3.7	5.8
		t_{PHL}	2.2	2.5	2.8	3.1	4.3

Intrinsic Parameters tdx	ktdx
1.7	2.5
1.9	1.4

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

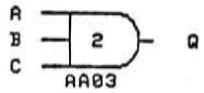
AAØ3

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description Pre-Routed Three Input And Gate

Logic Symbol



Truth Table

A	B	C	Q
L	X	X	L
X	L	X	L
X	X	L	L
H	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .AAØ3 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.3	3.8	4.4	4.9	7.1
		t_{PHL}	2.8	3.1	3.5	3.8	5.1

Intrinsic Parameters tdx	ktdx
2.8	2.6
2.5	1.6

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

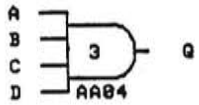
AAØ4

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description : Pre-Routed Four Input And Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	X	X	X	L
X	L	X	X	L
X	X	L	X	L
X	X	X	L	L
H	H	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .AAØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	4.2	4.8	5.3	5.9	8.2
		t_{PHL}	3.1	3.4	3.7	4.1	5.4

Intrinsic Parameters tdx	ktdx
3.6	2.7
2.7	1.6

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

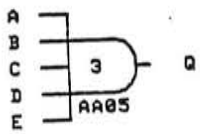
AAØ5

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Five Input And Gate

Logic Symbol



Truth Table

A	B	C	D	E	Q
L	X	X	X	X	L
X	L	X	X	X	L
X	X	L	X	X	L
X	X	X	L	X	L
X	X	X	X	L	L
H	H	H	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .AAØ5 A B C D E;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.8	6.4	7.0	7.7	10.1
		t_{PHL}	3.4	3.8	4.1	4.5	6.0

Intrinsic Parameters tdx	ktdx
5.2	2.9
3.0	1.8

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

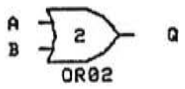
OR02

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Two Input Or Gate

Logic Symbol



Truth Table

A	B	Q
L	L	L
X	H	H
H	X	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .OR02 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	1.9	2.4	2.9	3.5	5.5
		t_{PHL}	2.7	3.1	3.5	3.8	5.2

Intrinsic Parameters tdx	ktdx
1.4	2.5
2.4	1.7

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

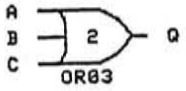
OR03

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Three Input Or Gate

Logic Symbol



Truth Table

A	B	C	Q
L	L	L	L
X	X	H	H
X	H	X	H
H	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .OR03 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.2	2.7	3.3	3.8	5.8
		t_{PHL}	4.9	5.3	5.7	6.2	7.8

Intrinsic Parameters tdx	ktdx
1.7	2.5
4.5	1.9

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

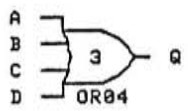
OR04

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Four Input Or Gate

Logic Symbol



Truth Table

A	B	C	D	Q
L	L	L	L	L
X	X	X	H	H
X	X	H	X	H
X	H	X	X	H
H	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .OR04 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.3	2.8	3.4	3.9	5.9
		t_{PHL}	6.6	7.1	7.6	8.2	10.3

Intrinsic Parameters tdx	ktdx
1.8	2.5
6.1	2.5

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

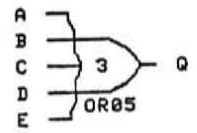
OR05

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Five Input Or Gate

Logic Symbol



Truth Table

A	B	C	D	E	Q
L	L	L	L	L	L
X	X	X	X	H	H
X	X	X	H	X	H
X	X	H	X	X	H
X	H	X	X	X	H
H	X	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .OR05 A B C D E;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.4	2.9	3.5	4.0	6.2
		t_{PHL}	9.3	9.9	10.5	11.0	13.3

Intrinsic Parameters tdx	ktdx
1.9	2.6
8.8	2.7

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

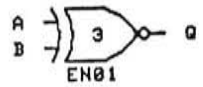
EN01

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Exclusive Nor Gate

Logic Symbol



Truth Table

A	B	Q
L	L	H
L	H	L
H	L	L
H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .EN01 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.4	4.0	4.5	5.1	7.4
		t_{PHL}	3.2	3.7	4.1	4.6	6.6

Intrinsic Parameters tdx	ktdx
2.9	2.7
2.7	2.3

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

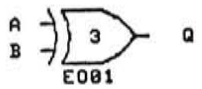
EØØ1

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Exclusive Or Gate

Logic Symbol



Truth Table

A	B	Q
L	L	L
L	H	H
H	L	H
H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	2

Equivalent Gate Count: 3

Bolt Syntax: Q .EØ1 A B;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.6	4.7	5.7	6.7	10.0
		t_{PHL}	2.9	3.3	3.6	4.0	5.3

Intrinsic Parameters tdx	ktdx
2.6	5.0
2.6	1.6

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

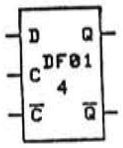
DFØ1

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop

Logic Symbol



Truth Table

C	CN	D	Q	QN
L	H	X	No Change	
↑	↓	L	L	H
↑	↓	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 4

Bolt Syntax: Q QN .DFØ1 D C CN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	2.8	3.3	3.9	4.4	6.6
		t_{PHL}	2.3	2.7	3.Ø	3.4	4.9
C	QN	t_{PLH}	3.7	4.3	4.8	5.3	7.5
		t_{PHL}	3.Ø	3.3	3.6	3.8	4.9
SET-UP		TO H	2.5				
SET-UP		TO L	2.7				

Intrinsic Parameters tdx	ktdx
2.2	2.6
1.9	1.8
3.2	2.6
2.8	1.3

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

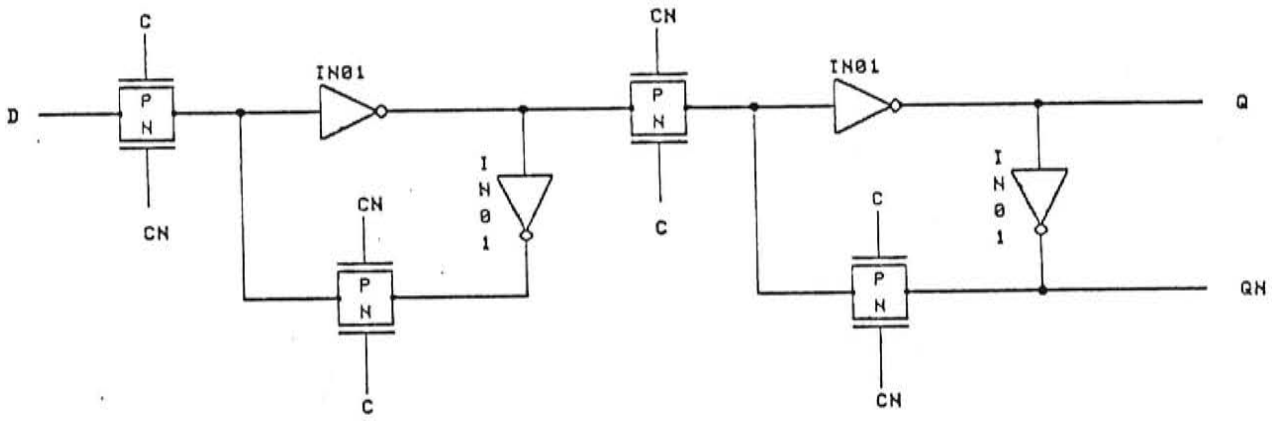
GOULD

AMI Semiconductors

DFØ1

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

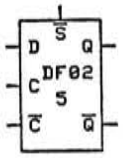
DFØ2

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Active Low Set

Logic Symbol



Truth Table

SN	C	CN	D	Q	QN
H	L	H	X	No	Change
H	↑	↓	L	L	H
H	↑	↓	H	H	L
L	X	X	X	H	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ2 D C CN SN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8	Intrinsic Parameters tdx	ktdx
C	Q	t_{PLH}	2.9	3.5	4.1	4.6	6.9	2.3	2.8
		t_{PHL}	3.Ø	3.5	4.1	4.6	6.6	2.5	2.5
C	QN	t_{PLH}	4.4	5.Ø	5.5	6.1	8.3	3.9	2.7
		t_{PHL}	3.2	3.5	3.8	4.1	5.2	3.9	1.4
SN	Q	t_{PLH}	2.3	2.9	3.4	4.Ø	6.3	2.3	1.4
SN	QN	t_{PHL}	2.6	2.9	3.2	3.5	4.7	2.3	1.4
SET-UP		TO H	2.6						
SET-UP		TO L	3.3						

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

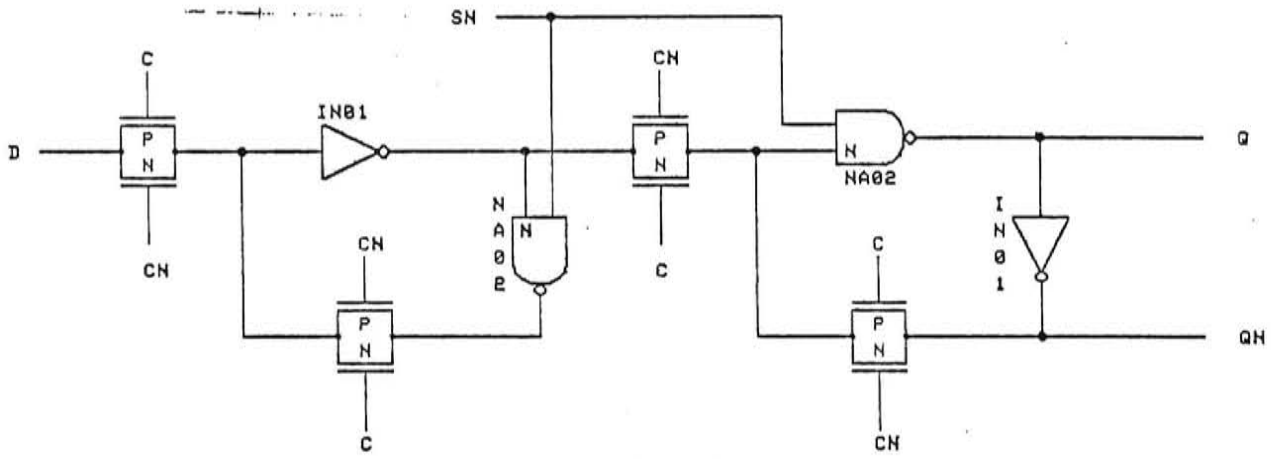
GOULD

AMI Semiconductors

DF02

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

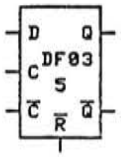
DFØ3

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Active Low Reset

Logic Symbol



Truth Table

RN	C	CN	D	Q	QN
H	L	H	X	No Change	
H	↑	↓	L	L	H
H	↑	↓	H	H	L
L	X	X	X	L	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ3 D C CN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	3.0	3.6	4.2	4.8	7.1
		t_{PHL}	2.4	2.7	3.1	3.4	4.8
C	QN	t_{PLH}	3.8	4.4	5.Ø	5.5	7.8
		t_{PHL}	3.9	4.3	4.8	5.2	7.1
RN	QN	t_{PLH}	3.4	4.Ø	4.5	5.Ø	7.2
RN	Q	t_{PHL}	4.1	4.4	4.8	5.2	6.7
SET-UP		TO H	3.1				
SET-UP		TO L	2.8				

Intrinsic Parameters tdx	ktdx
2.4	2.8
2.Ø	1.7
3.2	2.7
3.4	2.2
2.9	2.6
3.7	1.8

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

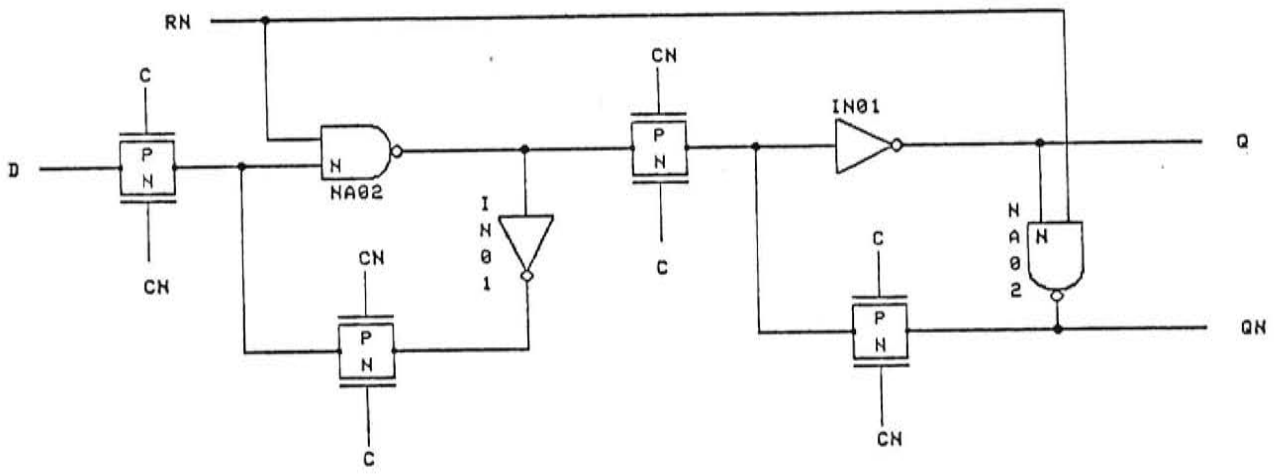
GOULD

AMI Semiconductors

DFØ3

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

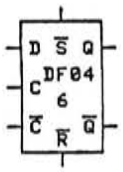
DFØ4

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop with Asynchronous Active Low Set and Active Low Reset

Logic Symbol



Truth Table

SN	RN	C	CN	D	Q	QN
H	H	L	H	X	No Change	
H	H	↑	↓	L	L	H
H	H	↑	↓	H	H	L
H	L	X	X	X	L	H
L	H	X	X	X	H	L
L	L	X	X	X	Illegal	

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 6

Bolt Syntax: Q QN .DFØ4 D C CN SN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8	Intrinsic Parameters tdx	ktdx
C	Q	t_{PLH}	3.1	3.7	4.3	4.9	7.3	2.5	2.9
		t_{PHL}	3.0	3.5	4.0	4.6	6.7	2.5	2.5
C	QN	t_{PLH}	4.4	5.0	5.5	6.1	8.2	3.9	2.6
		t_{PHL}	4.0	4.5	4.9	5.4	7.3	3.5	2.2
SN	Q	t_{PLH}	2.3	2.8	3.4	4.0	6.3	1.7	2.7
SN	QN	t_{PHL}	3.2	3.7	4.1	4.6	6.4	2.7	2.2
RN	QN	t_{PLH}	2.4	3.Ø	3.5	4.1	6.2	1.9	2.6
RN	Q	t_{PHL}	5.1	5.6	6.2	6.7	8.9	4.5	2.6
SET-UP		TO H	3.3						
SET-UP		TO L	3.3						

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

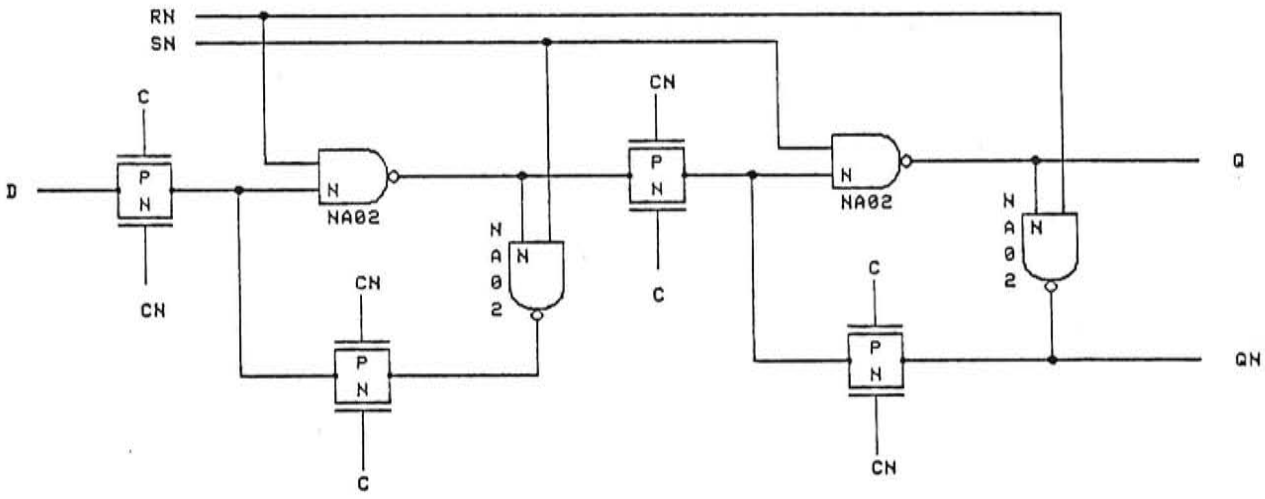
GOULD

AMI Semiconductors

DFØ4

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

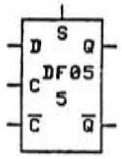
DFØ5

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Set

Logic Symbol



Truth Table

S	C	CN	D	Q	QN
L	L	H	X	No Change	
L	↑	↓	L	L	H
L	↑	↓	H	H	L
H	X	X	X	H	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Others	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ5 D C CN S;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	2.9	3.5	4.Ø	4.6	6.8
		t_{PHL}	2.7	3.1	3.5	4.Ø	5.7
C	QN	t_{PLH}	5.6	6.6	7.6	8.6	12.6
		t_{PHL}	3.2	3.5	3.8	4.1	5.2
S	Q	t_{PLH}	3.7	4.3	4.9	5.5	7.9
S	QN	t_{PHL}	1.1	1.4	1.7	2.Ø	3.3
SET-UP		TO H	2.6				
SET-UP		TO L	4.Ø				

Intrinsic Parameters tdx	ktdx
2.4	2.7
2.3	2.Ø
4.6	4.8
2.9	1.3
3.1	2.8
Ø.8	1.5

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

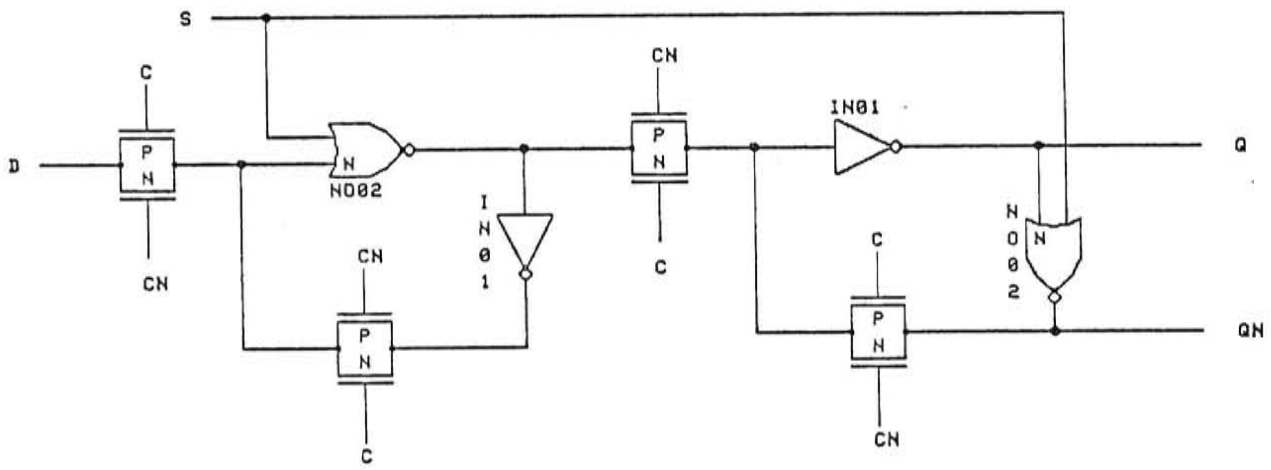
GOULD

AMI Semiconductors

DF05

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

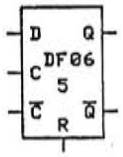
DFØ6

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Reset

Logic Symbol



Truth Table

R	C	CN	D	Q	QN
L	L	H	X	No	Change
L	↑	↓	L	L	H
L	↑	↓	H	H	L
H	X	X	X	L	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 5

Bolt Syntax: Q QN .DFØ6 D C CN R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	4.2	5.2	6.2	7.3	11.4
		t_{PHL}	2.4	2.7	3.1	3.4	4.8
C	QN	t_{PLH}	3.8	4.4	5.Ø	5.5	7.8
		t_{PHL}	3.9	4.2	4.5	4.9	6.2
R	QN	t_{PLH}	3.6	4.2	4.7	5.3	7.6
R	Q	t_{PHL}	Ø.9	1.2	1.6	1.9	3.4
SET-UP		TO H	3.5				
SET-UP		TO L	2.8				

Intrinsic Parameters tdx	ktdx
3.1	4.9
2.0	1.7
3.2	2.7
3.6	1.6
3.Ø	2.7
Ø.5	1.7

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

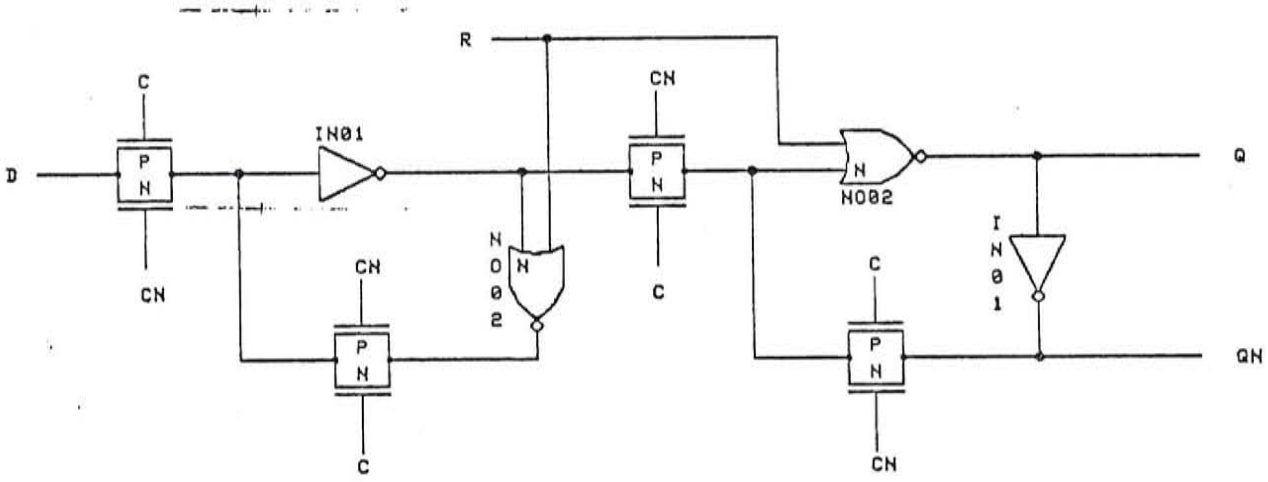
GOULD

AMI Semiconductors

DF06

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD AMI Semiconductors

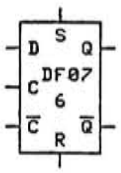
DFØ7

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed D Flip Flop With Asynchronous Set And Reset

Logic Symbol



Truth Table

S	R	C	CN	D	Q	QN
L	L	L	H	X	No Change	
L	L	↑	↓	L	L	H
L	L	↑	↓	H	H	L
L	H	X	X	X	L	H
H	L	X	X	X	H	L
H	H	X	X	X	Illegal	

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	2

Equivalent Gate Count: 6

Bolt Syntax: Q QN .DFØ7 D C CN S R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
C	Q	t_{PLH}	4.2	5.3	6.3	7.3	11.5
		t_{PHL}	2.6	3.1	3.5	3.9	5.7
C	QN	t_{PLH}	5.6	6.6	7.6	8.6	12.6
		t_{PHL}	4.Ø	4.3	4.6	4.9	6.2
R	QN	t_{PLH}	4.Ø	5.Ø	6.Ø	7.Ø	11.Ø
R	Q	t_{PHL}	Ø.9	1.3	1.6	1.9	3.2
S	Q	t_{PLH}	4.8	5.8	6.8	7.8	11.8
S	QN	t_{PHL}	1.1	1.4	1.7	2.Ø	3.3
SET-UP		TO H	3.6				
SET-UP		TO L	4.Ø				

Intrinsic Parameters tdx	ktdx
3.2	5.Ø
2.2	2,Ø
4.6	4.8
3.7	1.5
3.Ø	4.8
Ø.6	1.5
3.8	4.7
Ø.8	1.5

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

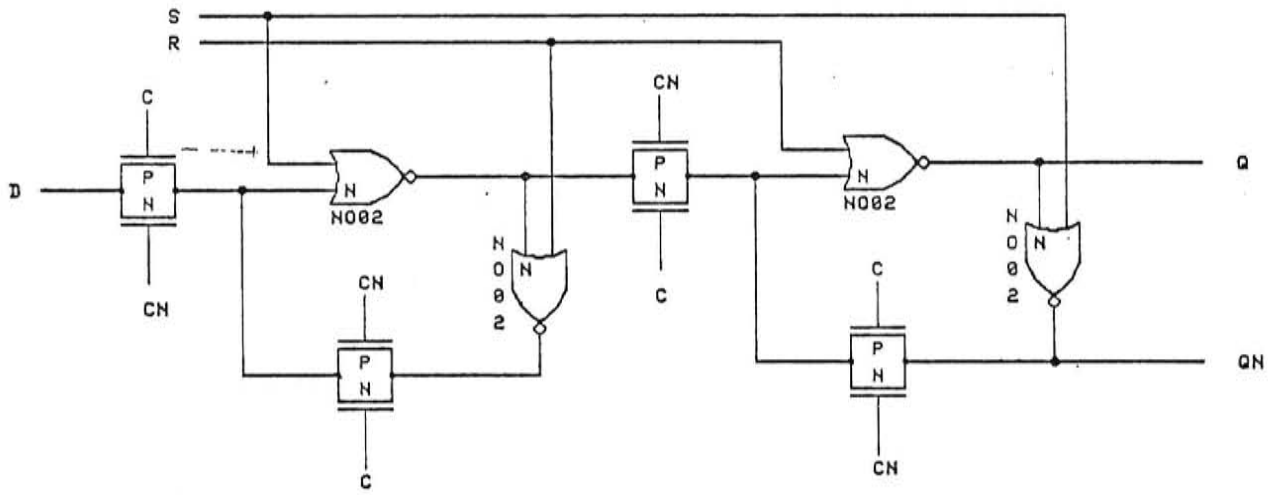
GOULD

AMI Semiconductors

DF07

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD
AMI Semiconductors

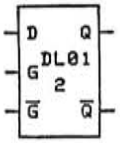
DLØ1

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Q and QN

Logic Symbol



Truth Table

G	GN	D	Q	QN
L	H	X	No Change	
H	L	D	D	DN

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 2

Bolt Syntax: Q QN .DLØ1 D G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	3.9	4.4	5.0	5.5	7.8
		t_{PHL}	3.2	3.5	3.8	4.0	5.2
G	QN	t_{PLH}	2.9	3.4	4.0	4.6	6.9
		t_{PHL}	2.4	2.8	3.1	3.5	4.9
D	Q	t_{PLH}	2.7	3.2	3.8	4.3	6.5
D	Q	t_{PHL}	2.6	2.9	3.2	3.4	4.6
D	QN	t_{PLH}	2.3	2.9	3.4	4.0	6.4
D	QN	t_{PHL}	1.2	1.6	2.0	2.3	3.8

Intrinsic Parameters tdx	ktdx
3.3	2.7
2.9	1.4
2.3	2.7
2.1	1.7
2.1	2.6
2.3	1.4
1.7	2.8
0.9	1.7

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

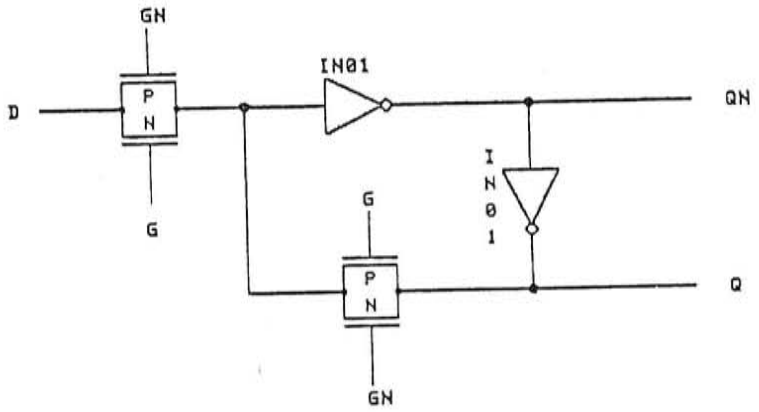
GOULD

AMI Semiconductors

DLØ1

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

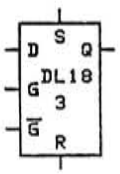
DL18

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Set and Reset

Logic Symbol



Truth Table

S	R	G	GN	D	Q
L	L	L	H	X	No Change
L	L	H	L	D	D
L	H	X	X	X	L
H	L	X	X	X	H
H	H	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .DL18 D G GN S R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	5.0	6.0	7.0	8.0	12.0
		t_{PHL}	3.8	4.1	4.5	4.8	6.1
D	Q	t_{PLH}	4.1	5.1	6.1	7.1	11.1
		t_{PHL}	3.3	3.6	3.9	4.2	5.9
S	Q	t_{PLH}	6.2	7.2	8.3	9.3	13.3
R	Q	t_{PHL}	1.5	1.8	2.1	2.4	3.6

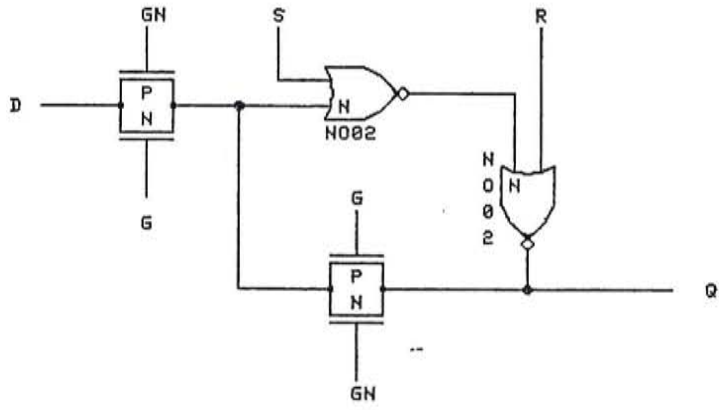
Intrinsic Parameters tdx	ktdx
4.0	4.8
3.5	1.5
3.1	4.8
3.0	1.4
5.2	4.8
1.2	1.4

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS



GOULD

AMI Semiconductors

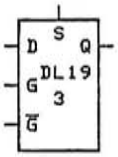
DL19

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Set

Logic Symbol



Truth Table

S	G	GN	D	Q
L	L	H	X	No Change
L	H	L	D	D
H	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .DL19 D G GN S;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	4.0	4.6	5.2	5.7	8.0
		t_{PHL}	4.1	4.4	4.8	5.1	6.4
D	Q	t_{PLH}	2.6	3.2	3.7	4.3	6.4
		t_{PHL}	3.4	3.7	4.0	4.3	5.5
S	Q	t_{PLH}	3.7	4.2	4.8	5.4	7.7

Intrinsic Parameters tdx	ktdx
3.5	2.7
3.8	1.6
2.1	2.5
3.1	1.5
3.1	2.7

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

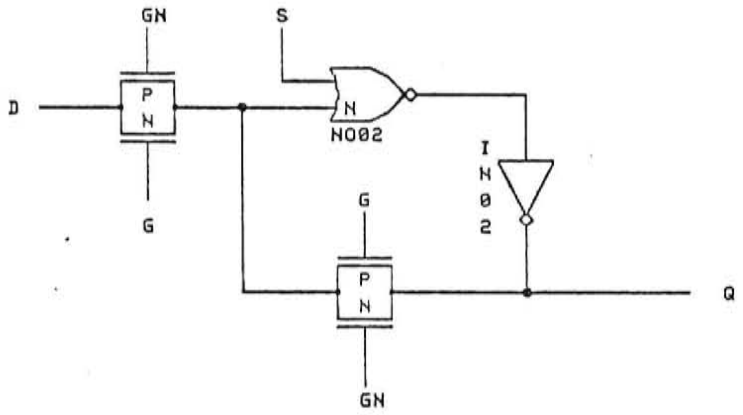
GOULD

AMI Semiconductors

DL19

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

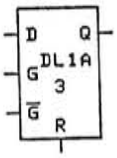
DL1A

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Reset

Logic Symbol



Truth Table

R	G	GN	D	Q
L	L	H	X	No Change
L	H	L	D	D
H	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .DL1A D G GN R;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	5.5	6.5	7.5	8.4	12.4
		t_{PHL}	3.2	3.5	3.8	4.1	5.3
D	Q	t_{PLH}	4.2	5.2	6.2	7.3	11.3
		t_{PHL}	2.6	2.9	3.2	3.4	4.5
R	Q	t_{PHL}	1.7	2.0	2.3	2.6	3.8

Intrinsic Parameters tdx	ktdx
4.5	4.7
2.9	1.4
3.2	4.8
2.4	1.3
1.4	1.4

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

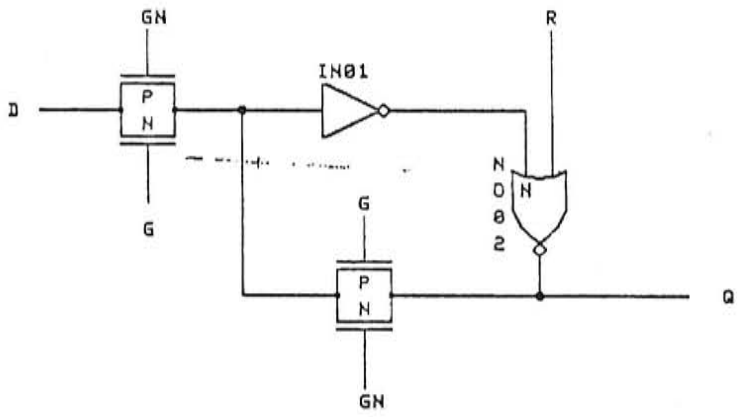
GOULD

AMI Semiconductors

DL1A

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

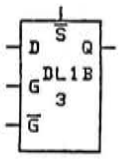
DL1B

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Active Low Set

Logic Symbol



Truth Table

SN	G	GN	D	Q
H	L	H	X	No Change
H	H	L	D	D
L	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .DL1B D G GN SN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	4.1	4.7	5.2	5.7	7.9
		t_{PHL}	3.9	4.4	4.9	5.4	7.4
D	Q	t_{PLH}	2.7	3.3	3.8	4.3	6.5
		t_{PHL}	3.2	3.7	4.2	4.6	6.5
SN	Q	t_{PLH}	3.6	4.2	4.7	5.2	7.3

Intrinsic Parameters tdx	ktdx
3.6	2.6
3.4	2.3
2.2	2.6
2.8	2.2
3.1	2.5

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

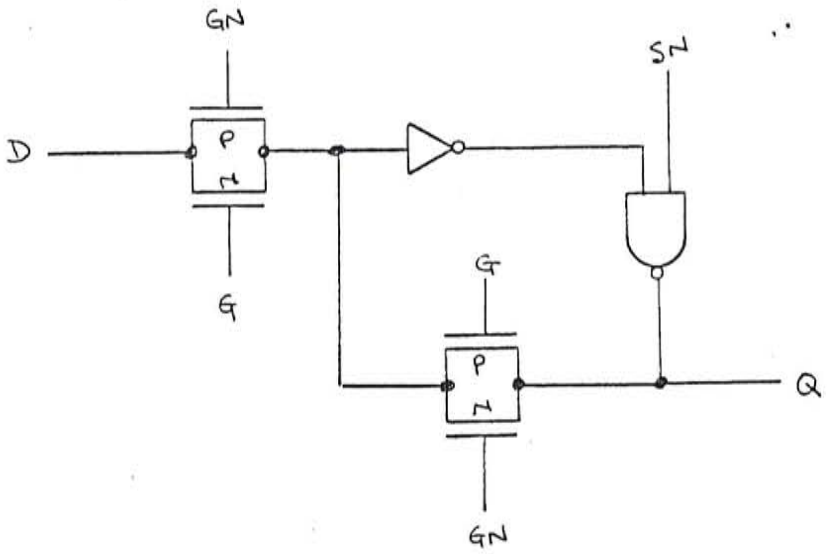
GOULD

AMI Semiconductors

DL1B

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

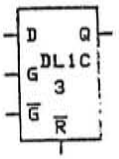
DL1C

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Active Low Reset

Logic Symbol



Truth Table

RN	G	GN	D	Q
H	L	H	X	No Change
H	H	L	D	D
L	X	X	X	L

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .DL1C D G GN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	4.4	5.0	5.5	6.1	8.3
		t_{PHL}	3.3	3.5	3.8	4.1	5.2
D	Q	t_{PLH}	3.2	3.8	4.3	4.9	7.1
		t_{PHL}	2.6	2.9	3.2	3.5	4.6
RN	Q	t_{PHL}	3.1	3.4	3.7	4.0	5.1

Intrinsic Parameters tdx	ktdx
3.9	2.7
3.0	1.3
2.7	2.7
2.3	1.4
2.8	1.4

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

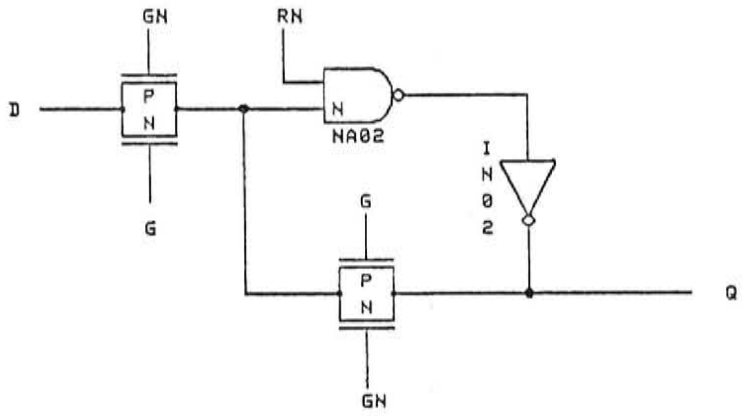
GOULD

AMI Semiconductors

DL1C

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

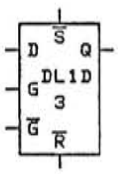
DL1D

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Latch With Active Low Set And Reset

Logic Symbol



Truth Table

SN	RN	G	GN	D	Q
H	H	L	H	X	No Change
H	H	H	L	D	D
H	L	X	X	X	L
L	H	X	X	X	H
L	L	X	X	X	H

Input Loading

Logic Input	Equivalent Unit Loads
D	3
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .DL1D D G GN SN RN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	5.0	5.5	6.0	6.6	8.8
		t_{PHL}	4.1	4.6	5.0	5.5	7.3
D	Q	t_{PLH}	3.6	4.1	4.7	5.2	7.4
		t_{PHL}	3.5	3.9	4.3	4.8	6.5
SN	Q	t_{PLH}	3.7	4.3	4.8	5.3	7.5
RN	Q	t_{PHL}	4.5	4.9	5.4	5.8	7.7

Intrinsic Parameters tdx	ktdx
4.4	2.6
3.6	2.2
3.0	2.6
3.0	2.1
3.2	2.6
4.0	2.2

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

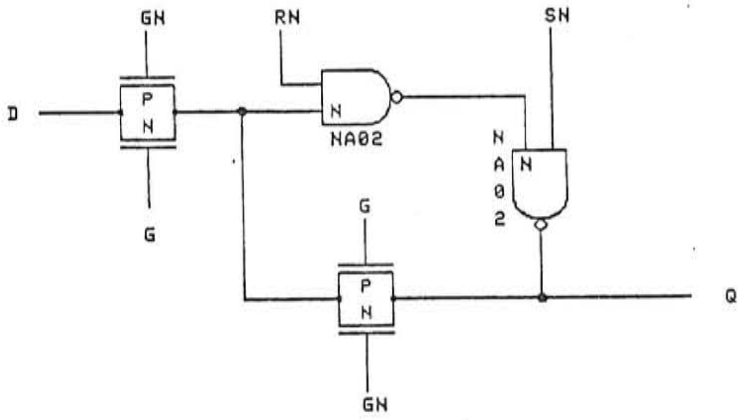
GOULD

AMI Semiconductors

DL1D

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



GOULD

AMI Semiconductors

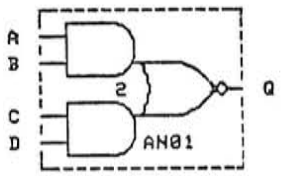
AN01

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed And-Nor Gates

Logic Symbol



Truth Table

A	B	C	D	Q
L	X	L	X	H
L	X	X	L	H
X	L	L	X	H
X	L	X	L	H
H	H	X	X	L
X	X	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .AN01 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.3	6.3	7.3	8.2	12.2
		t_{PHL}	2.1	2.5	2.9	3.4	5.1

Intrinsic Parameters tdx	ktdx
4.3	4.7
1.6	2.0

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

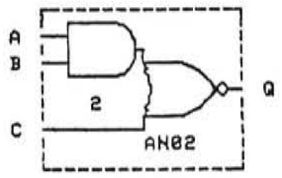
AN02

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed And-Nor Gate

Logic Symbol



Truth Table

A	B	C	Q
L	X	L	H
X	L	L	H
H	H	X	L
X	X	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .AN02 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.9	4.8	5.7	6.7	10.5
		t_{PHL}	1.6	2.1	2.5	3.0	4.8

Intrinsic Parameters tdx	ktdx
2.9	4.5
1.1	2.2

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

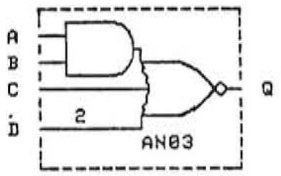
AN03

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed And-Nor Gate

Logic Symbol



Truth Table

A	B	C	D	Q
X	X	X	H	L
X	X	H	X	L
H	H	X	X	L
X	L	L	L	H
L	X	L	L	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q . AN03 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	6.4	7.8	9.2	10.6	16.3
		t_{PHL}	1.8	2.3	2.7	3.1	4.9

Intrinsic Parameters tdx	ktdx
5.0	6.7
1.4	2.1

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

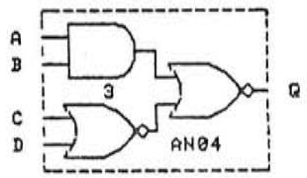
AN04

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description : Pre-Routed And-Nor-Nor Gate

Logic Symbol



Truth Table

A	B	C	D	Q
X	X	L	L	L
H	H	X	X	L
X	L	X	H	H
X	L	H	X	H
L	X	X	H	H
L	X	H	X	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .AN04 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.6	4.6	5.6	6.7	10.8
		t_{PHL}	3.0	3.3	3.6	3.9	5.2

Intrinsic Parameters tdx	ktdx
2.6	4.9
2.6	1.6

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

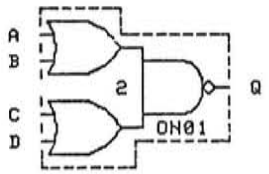
ONØ1

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Or-Nand Gate

Logic Symbol



Truth Table

A	B	C	D	Q
H	X	H	X	L
H	X	X	H	L
X	H	H	X	L
L	L	X	X	H
X	X	L	L	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .ONØ1 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	5.1	6.0	7.0	7.9	11.7
		t_{PHL}	2.1	2.6	3.0	3.4	5.2

Intrinsic Parameters tdx	ktdx
4.1	4.52
1.7	2.07

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

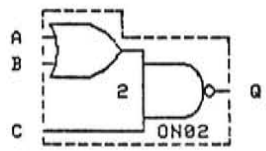
ON02

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Or-Nand Gate

Logic Symbol



Truth Table

A	B	C	Q
X	X	L	H
H	X	H	L
X	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .ONØ2 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	4.1	5.1	6.1	7.1	11.0
		t_{PHL}	1.6	2.1	2.6	3.0	4.9

Intrinsic Parameters tdx	ktdx
3.1	4.7
1.1	2.2

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

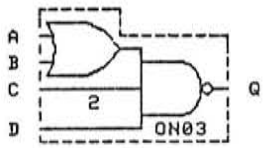
ON03

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed Or-Nand Gate

Logic Symbol



Truth Table

A	B	C	D	Q
X	X	X	L	H
X	X	L	X	H
L	L	X	X	H
X	H	H	H	L
H	X	H	H	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .ONØ3 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.8	3.7	4.7	5.6	9.4
		t_{PHL}	1.9	2.6	3.3	3.9	6.5

Intrinsic Parameters tdx	ktdx
1.9	4.5
1.3	3.1

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

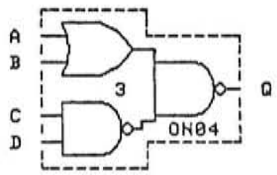
ON04

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description : Pre-Routed Or-Nand-Nand Gate

Logic Symbol



Truth Table

A	B	C	D	Q
X	H	X	L	L
H	X	X	L	L
X	H	L	X	L
H	X	L	X	L
L	L	X	X	H
X	X	H	H	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 3

Bolt Syntax: Q .ONØ4 A B C D;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.3	2.8	3.3	3.8	5.9
		t_{PHL}	2.6	3.1	3.6	4.1	6.0

Intrinsic Parameters tdx	ktdx
1.7	2.5
2.1	2.3

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

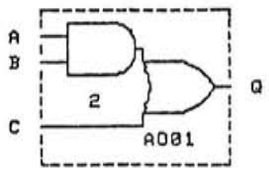
AØØ1

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description : Pre-Routed And-Or Gate

Logic Symbol



Truth Table

A	B	C	Q
X	X	H	H
L	X	L	L
X	L	L	L
H	H	X	H

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .AØ1 A B C;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	2.7	3.3	3.8	4.3	6.5
		t_{PHL}	4.1	4.5	4.9	5.3	6.9

Intrinsic Parameters tdx	ktdx
2.2	2.6
3.7	1.9

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

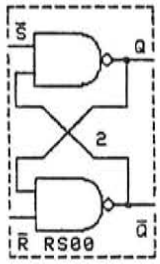
RSØØ

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Set-Reset Nand Gate Latch

Logic Symbol



Truth Table

SN	RN	Q	QN
L	L	H	H
L	H	H	L
H	L	L	H
H	H	No Change	

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q QN .RSØØ RN SN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
SN	Q	t_{PLH}	2.2	2.7	3.2	3.8	5.9
RN		t_{PHL}	3.0	3.4	3.9	4.3	6.1

Intrinsic Parameters tdx	ktdx
1.6	2.5
2.6	2.1

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

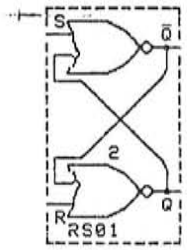
RSØ1

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed Nor S R Latch

Logic Symbol



Truth Table

S	R	Q	QN
L	L	No change	
L	H	L	H
H	L	H	L
H	H	L	L

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q QN .RSØ1 R S;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
S	Q	t_{PLH}	3.5	4.4	5.4	6.3	1Ø.Ø
R	Q	t_{PHL}	1.Ø	1.3	1.6	1.9	3.2

Intrinsic Parameters tdx	ktdx
2.6	4.4
Ø.7	1.5

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

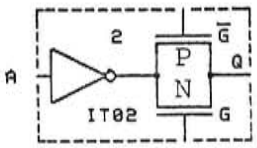
ITØ2

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Inverter Driving a Transmission Gate

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	AN

Input Loading

Logic Input	Equivalent Unit Loads
Any Input	1

Equivalent Gate Count: 2

Bolt Syntax: Q .ITØ2 A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
G	Q	t_{PLH}	1.8	2.4	3.1	3.8	6.5
		t_{PHL}	1.0	1.4	1.8	2.3	4.0
A	Q	t_{PLH}	2.5	3.2	3.8	4.5	7.1
		t_{PHL}	1.2	1.6	2.0	2.4	4.1

Intrinsic Parameters tdx	ktdx
1.1	3.2
0.5	2.0
1.9	3.1
0.8	1.9

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

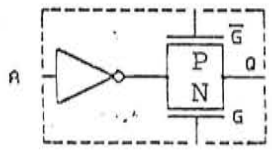
IT06

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description: Two Pre-Routed Inverters In Parallel Driving A Transmission Gate

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	AN

Input Loading

Logic Input	Equivalent Unit Loads
A	2
All Other	1

Equivalent Gate Count: 2

Bolt Syntax: Q .IT06 A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	2.1	2.6	3.1	3.6	5.8
		t_{PHL}	Ø.9	1.2	1.6	1.9	3.4
G	Q	t_{PLH}	1.4	2.1	2.7	3.3	5.8
		t_{PHL}	Ø.9	1.3	1.7	2.1	3.9

Intrinsic Parameters tdx	ktdx
1.6	2.5
0.5	1.7
Ø.8	3.Ø
Ø.5	2.Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

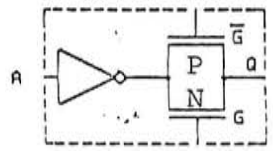
ITØ8

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Three Pre-Routed Inverters In Parallel Driving A Transmission Gate

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	AN

Input Loading

Logic Input	Equivalent Unit Loads
A	3
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q . ITØ8 A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.9	2.3	2.8	3.2	5.1
		t_{PHL}	Ø.8	1.1	1.4	1.7	2.9
G	Q	t_{PLH}	1.3	1.8	2.4	3.0	5.2
		t_{PHL}	Ø.9	1.2	1.6	1.9	3.4

Intrinsic Parameters tdx	ktdx
1.4	2.2
Ø.5	1.4
Ø.7	2.7
Ø.5	1.7

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

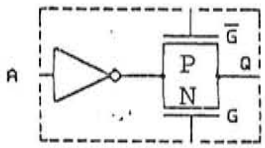
ITØA

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Four Pre-Routed Inverters In Parallel Driving A Transmission Gate

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	AN

Input Loading

Logic Input	Equivalent Unit Loads
A	4
All Other	1

Equivalent Gate Count: 3

Bolt Syntax: Q .ITØA A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.6	2.Ø	2.5	2.9	4.7
		t_{PHL}	Ø.5	Ø.8	1.1	1.4	2.6
G	Q	t_{PLH}	1.2	1.8	2.3	2.8	4.9
		t_{PHL}	Ø.8	1.2	1.5	1.9	3.4

Intrinsic Parameters tdx	ktdx
1.2	2.1
Ø.2	1.4
Ø.7	2.5
Ø.5	1.7

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

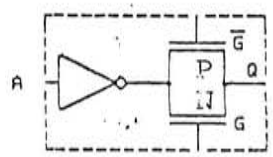
ITØC

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Six Pre-Routed Inverters In Parallel Driving A Transmission Gate

Logic Symbol



Truth Table

A	G	GN	Q
A	L	H	Z
A	H	L	AN

Input Loading

Logic Input	Equivalent Unit Loads
A	6
All Other	1

Equivalent Gate Count: 4

Bolt Syntax: Q .ITØC A G GN;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.5	1.9	2.3	2.6	4.1
		t_{PHL}	Ø.5	1.7	1.Ø	1.3	2.4
G	Q	t_{PLH}	1.2	1.7	2.1	2.6	4.6
		t_{PHL}	Ø.8	1.2	1.5	1.9	3.4

Intrinsic Parameters tdx	ktdx
1.1	1.8
Ø.2	1.3
Ø.7	2.3
Ø.5	1.7

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

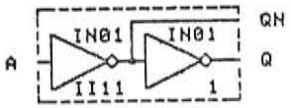
II11

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With a Single Inverter Followed By a Single Inverter

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 1

Bolt Syntax: Q QN .II11 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	2.6	3.7	4.9	6.Ø	1Ø.6
		t_{PHL}	2.9	4.Ø	5.Ø	6.1	1Ø.3
A	QN	t_{PLH}	2.2	3.Ø	3.7	4.4	7.4
		t_{PHL}	Ø.9	1.3	1.7	2.1	3.8

Intrinsic Parameters tdx	ktdx
1.4	5.4
1.9	5.Ø
1.5	3.5
Ø.5	2.Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

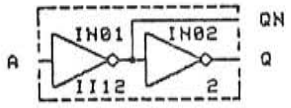
II12

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With A Single Inverter Followed By Two Inverters In Parallel

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II12 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	2.2	3.Ø	3.8	4.6	7.7
		t_{PHL}	2.9	3.8	4.6	5.5	9.Ø
A	QN	t_{PLH}	2.9	3.6	4.3	4.9	7.6
		t_{PHL}	1.2	1.6	2.Ø	2.3	3.9

Intrinsic Parameters tdx	ktdx
1.4	3.8
2.Ø	4.1
2.2	3.2
Ø.8	1.8

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

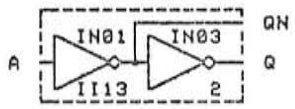
II13

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With A Single Inverter Followed By Three Inverters In Parallel

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	1

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II13 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.9	2.9	3.9	4.8	8.7
		t_{PHL}	2.1	2.9	3.6	4.3	7.3
A	QN	t_{PLH}	1.3	1.7	2.Ø	2.4	3.9
		t_{PHL}	Ø.2	Ø.4	Ø.7	1.Ø	2.1

Intrinsic Parameters tdx	ktdx
Ø.9	4.7
1.4	3.5
Ø.9	1.8
Ø.Ø	1.3

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

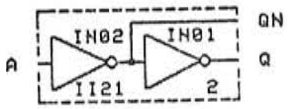
II21

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With Two Inverters In Parallel Followed By A Single Inverter

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II21 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.4	2.0	2.7	3.3	5.8
		t_{PHL}	1.9	2.5	3.1	3.6	5.9
A	QN	t_{PLH}	1.6	1.9	2.3	2.6	4.1
		t_{PHL}	Ø.4	Ø.6	Ø.9	1.1	2.1

Intrinsic Parameters tdx	ktdx
Ø.8	3.Ø
1.4	2.7
1.2	1.7
Ø.1	1.1

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

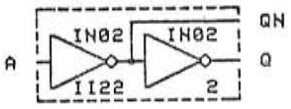
II22

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With Two Inverters In Parallel Followed By Two Inverters In Parallel

Logic Symbol



Truth Table

A	Q N	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	2

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II22 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	2.5	3.2	3.8	4.5	7.1
		t_{PHL}	3.2	4.Ø	4.8	5.5	8.6
A	QN	t_{PLH}	3.5	4.2	4.9	5.5	8.3
		t_{PHL}	1.4	1.9	2.3	2.7	4.3

Intrinsic Parameters tdx	ktdx
1.9	3.1
2.5	3.6
2.8	3.2
1.Ø	1.9

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

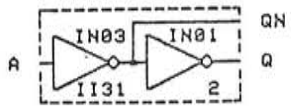
II31

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed Clock Driver With Three Inverters In Parallel Followed By A Single Inverter

Logic Symbol



Truth Table

A	QN	Q
H	L	H
L	H	L

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: 2

Bolt Syntax: Q QN .II31 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
A	Q	t_{PLH}	1.7	2.7	3.6	4.5	8.2
		t_{PHL}	2.Ø	2.6	3.2	3.8	6.3
A	QN	t_{PLH}	1.2	1.4	1.7	1.9	2.9
		t_{PHL}	Ø.1	Ø.3	Ø.5	Ø.7	1.5

Intrinsic Parameters tdx	ktdx
Ø.8	4.4
1.4	2.9
Ø.9	1.2
Ø.Ø	Ø.9

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

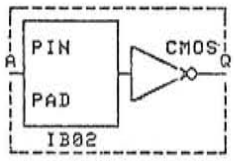
IBØ2

August 1984

3U DOUBLE - METAL HCMOS GATE ARRAYS

Description: Pre-Routed CMOS Inverting Input Buffer

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	6

Equivalent Gate Count: Ø

Bolt Syntax: Q .IBØ2 A;

Switching Characteristics:

Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	Ø.5	Ø.6	Ø.7	Ø.8	1.3
		t_{PHL}	Ø.2	Ø.3	Ø.4	Ø.5	Ø.8

Intrinsic Parameters tdx	ktdx
Ø.4	Ø.5
Ø.2	Ø.4

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance

GOULD

AMI Semiconductors

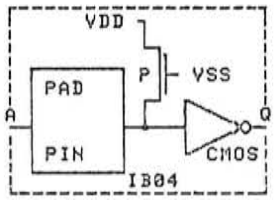
IBØ4

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre- Routed CMOS Inverting Input Buffer With Pull-Up

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	6

Equivalent Gate Count: Ø

Bolt Syntax: Q .IBØ4 A;

Switching Characteristics:

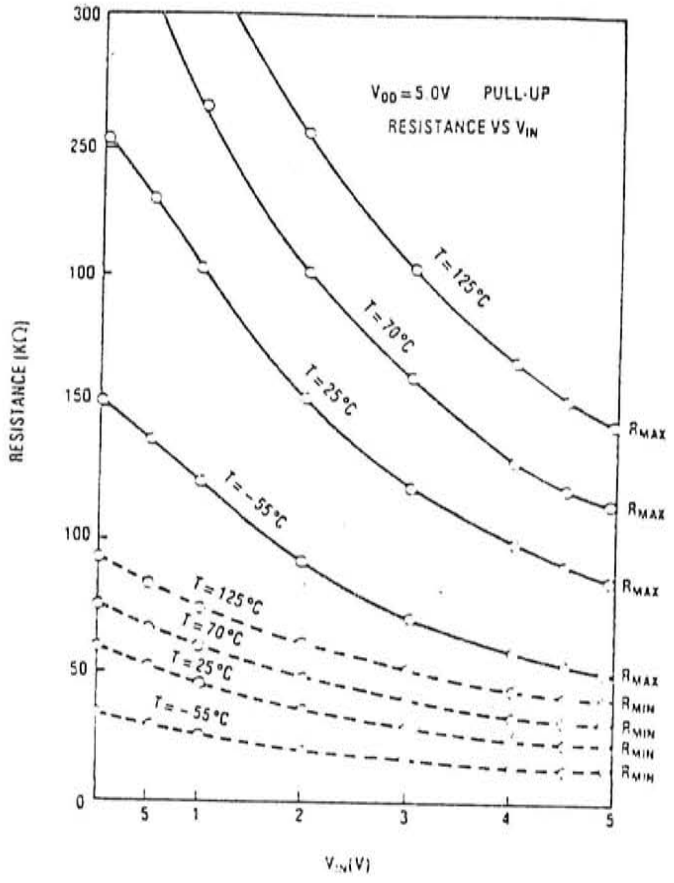
Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	Ø.5	Ø.6	Ø.7	Ø.8	1.3
		t_{PHL}	Ø.2	Ø.3	Ø.4	Ø.5	Ø.8

Intrinsic Parameters tdx	ktdx
Ø.4	Ø.5
Ø.2	Ø.4

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance



Gate Array Pull-Up Characteristics

NOTE: The values on this curve must be multiplied by 0.78 for double metal resistance values

GOULD

AMI Semiconductors

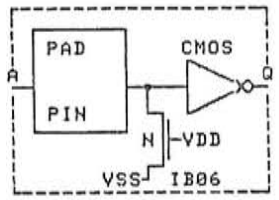
IBØ6

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed CMOS Inverting Input Buffer With Pull Down

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	6

Equivalent Gate Count: Ø

Bolt Syntax: Q .IBØ6 A;

Switching Characteristics:

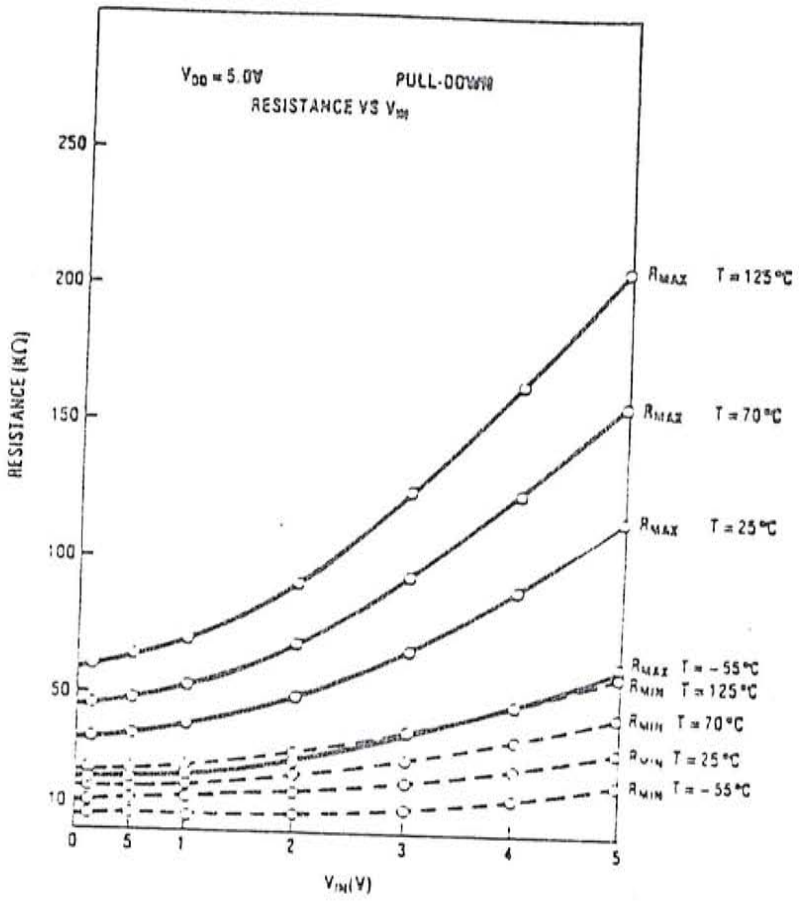
Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	Ø.5	Ø.6	Ø.7	Ø.8	1.3
		t_{PHL}	Ø.2	Ø.3	Ø.4	Ø.5	Ø.8

Intrinsic Parameters tdx	ktdx
Ø.4	Ø.5
Ø.2	Ø.4

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance



Gate Array Pull-Down Characteristics

NOTE: The values on this curve must be multiplied by 0.74 for double metal resistance values

GOULD

AMI Semiconductors

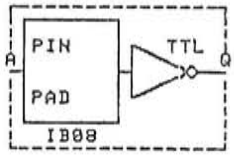
IBØ8

August 1984

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed TTL Inverting Input Buffer

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: Ø

Bolt Syntax: Q .IBØ8 A;

Switching Characteristics:

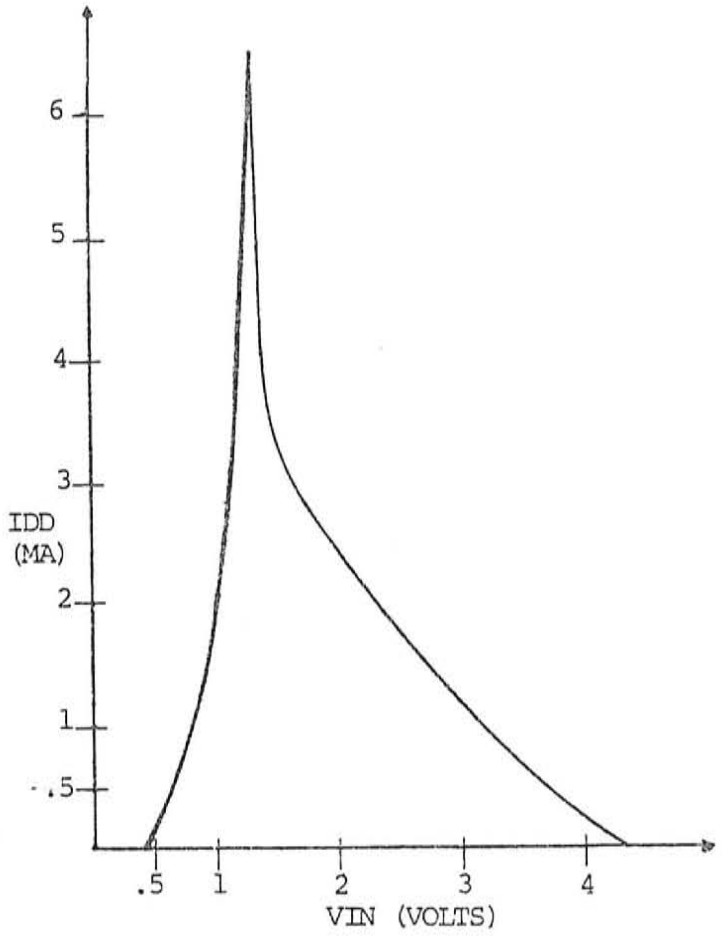
Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.3	3.6	3.9	4.2	5.6
		t_{PHL}	2.5	2.8	3.Ø	3.2	4.1

Intrinsic Parameters tdx	ktdx
2.9	1.6
2.3	1.Ø

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

Unit Load = .21 pf including statistical wiring capacitance



IDD vs. VIN FOR TTL LEVEL TRANSLATOR

GOULD

AMI Semiconductors

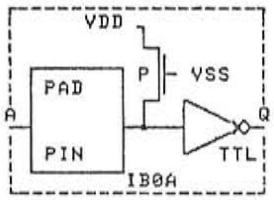
IBOA

August 1984

3μ DOUBLE - METAL HCMOS
GATE ARRAYS

Description: Pre-Routed TTL Inverting Input Buffer With Pull-Up

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: Ø

Bolt Syntax: Q . IBØA A;

Switching Characteristics:

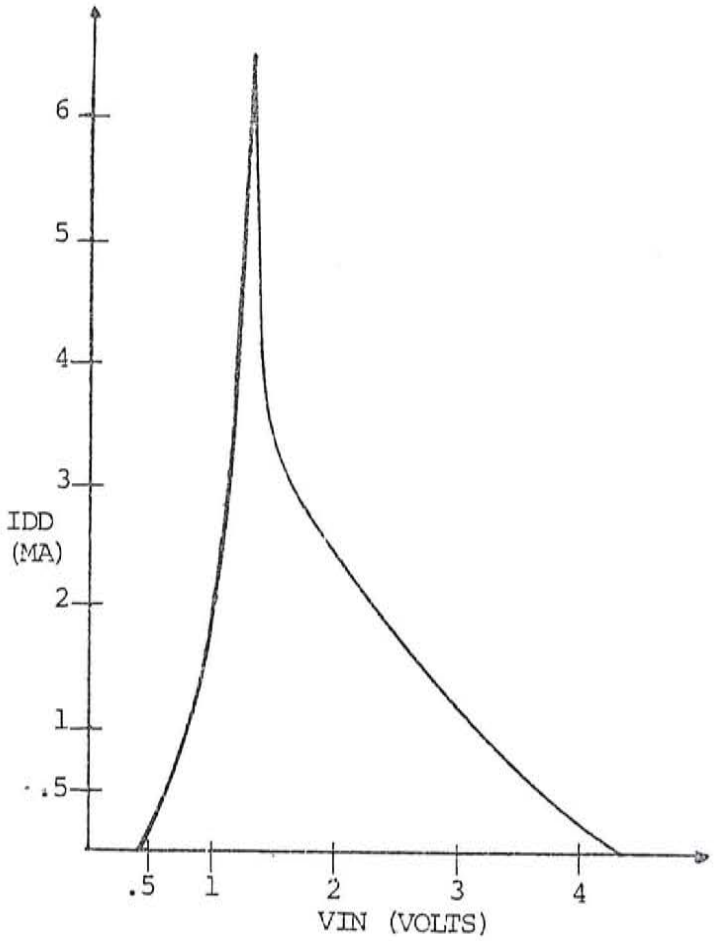
Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.3	3.6	3.9	4.2	5.6
		t_{PHL}	2.5	2.8	3.0	3.2	4.1

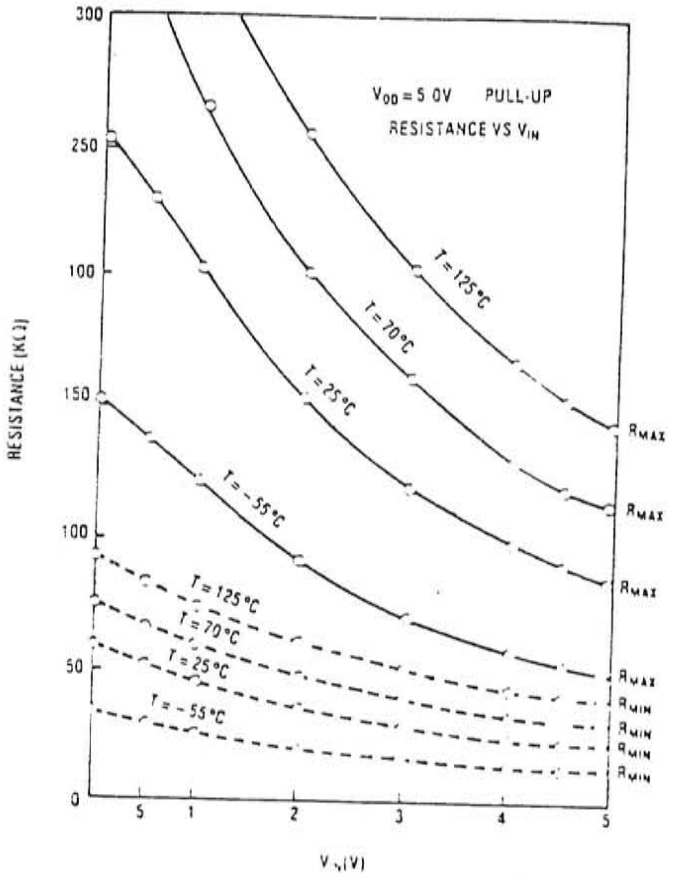
Intrinsic Parameters tdx	ktdx
2.9	1.6
2.3	1.Ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance



IDD vs. VIN FOR TTL LEVEL TRANSLATOR



Gate Array Pull-Up Characteristics

NOTE: The values on this curve must be multiplied by 0.78 for double metal resistance values

GOULD

AMI Semiconductors

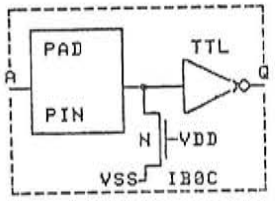
IBØC

August 1984

3µ DOUBLE - METAL HCMOS GATE ARRAYS

Description : Pre-Routed TTL Inverting Input Buffer With Pull Down

Logic Symbol



Truth Table

A	Q
H	L
L	H

Input Loading

Logic Input	Equivalent Unit Loads
A	3

Equivalent Gate Count: ø

Bolt Syntax: Q .IBØC A;

Switching Characteristics:

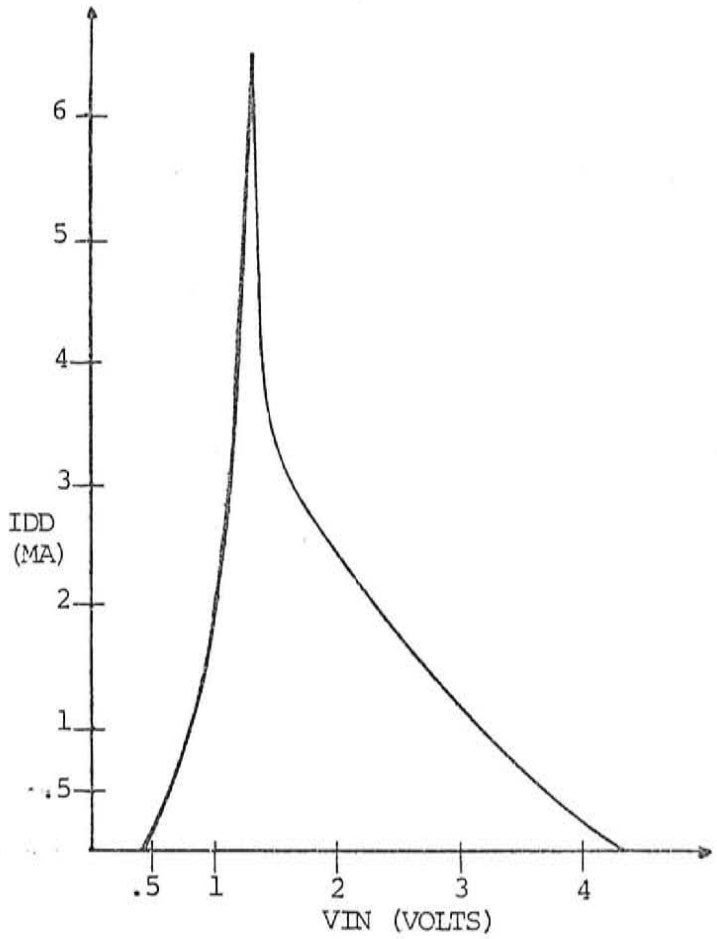
Conditions: V_{DD} = 5V, T_J = 25 °C, Typical Process

Max. Delay (ns) FROM	TO	Parameter	Number of Unit Loads 1	2	3	4	8
Any Input	Q	t_{PLH}	3.3	3.6	3.9	4.2	5.6
		t_{PHL}	2.5	2.8	3.ø	3.2	4.1

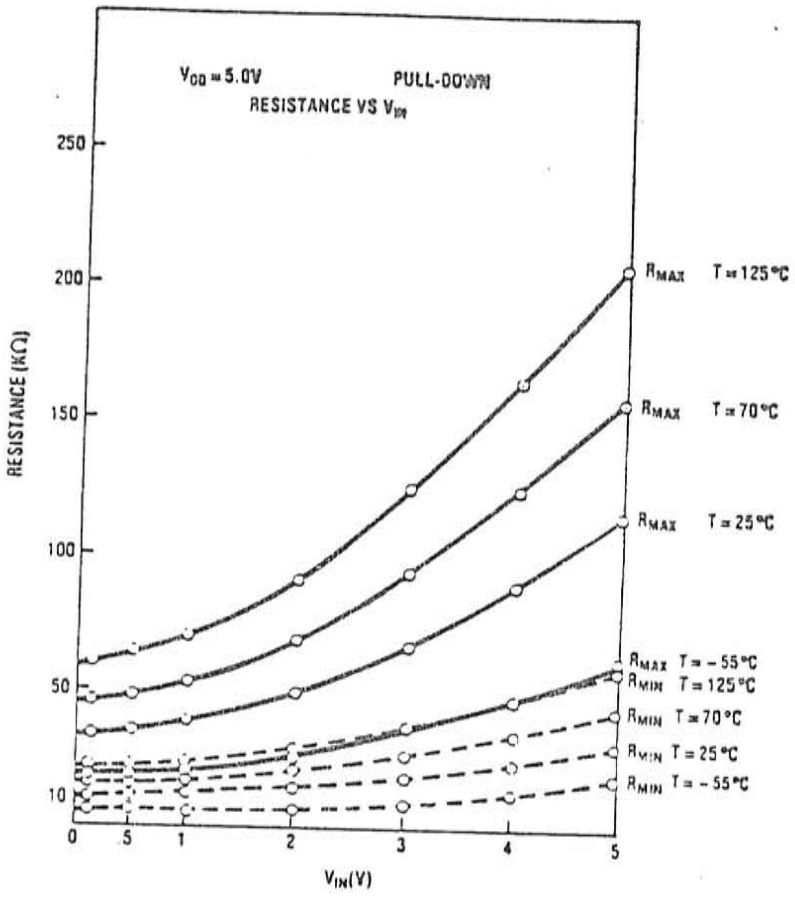
Intrinsic Parameters tdx	ktdx
2.9	1.6
2.3	1.ø

Propagation Delay Equation: tpx = tdx + ktdx * C_L

Unit Load = .21 pf including statistical wiring capacitance



IDD vs. VIN FOR TTL LEVEL TRANSLATOR



Gate Array Pull-Down Characteristics

NOTE: The values on this curve must be multiplied by 0.74 for double metal resistance values

GOULD

AMI Semiconductors

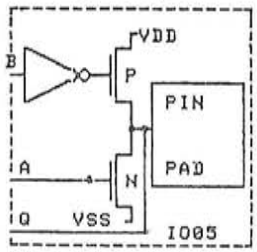
I0Ø5

August 1984

Description: Tri-State Bidirectional Buffer

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Logic Symbol



Truth Table

A	B	Q
L	L	Z
L	H	H
H	L	L
H	H	Illegal

Input Loading

Logic Input	*Equivalent Unit Loads
B	5
A	6

*.21 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used

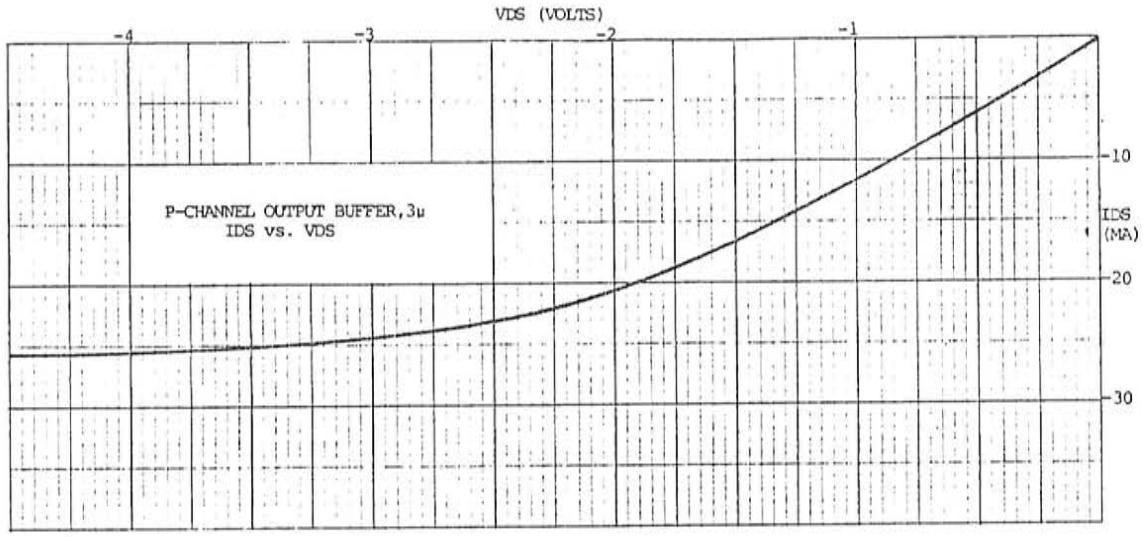
Bolt Syntax: Q .I0Ø5 A B;

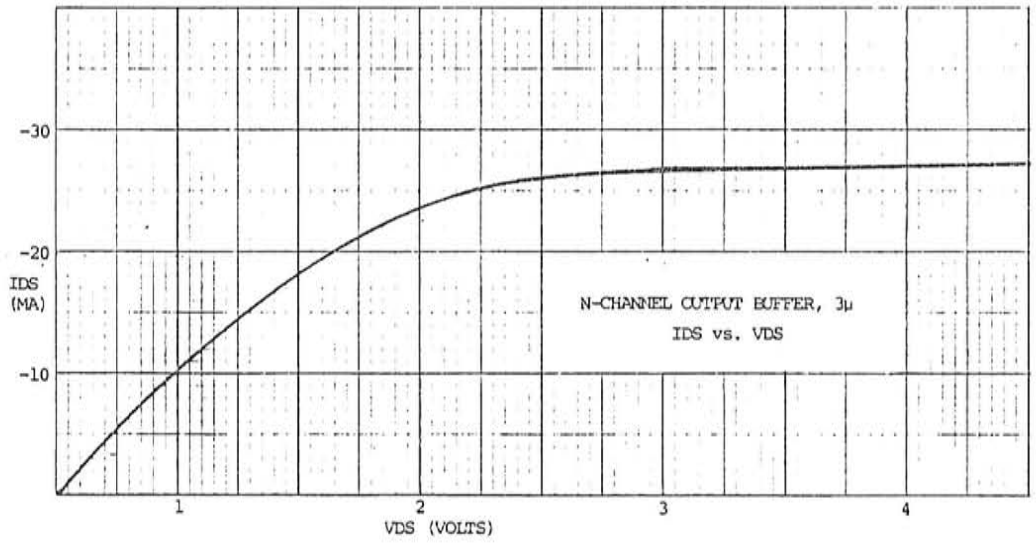
Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf	Delay Coefficients tdx	ktdx
A	Q	t_{PLH}	7.9	13.2	18.5	2.5	Ø.1
		t_{PHL}	4.4	7.4	1Ø.4	1.4	Ø.1
B	Q	t_{PZH}	6.7	12.1	17.6	1.2	Ø.1
		t_{PZL}	4.1	7.6	11.1	Ø.6	Ø.1
ITTL LOAD		t_{PLH}	5.5	9.Ø	12.4	2.1	Ø.1
A	Q	t_{PHL}	6.8	11.8	16.7	1.9	Ø.1
B	Q	t_{PZH}	4.1	7.5	1Ø.9	Ø.7	Ø.1
		t_{PZL}	6.6	12.2	17.9	Ø.9	Ø.1

Symbol	Parameters	Conditions	Min.	Typ.	Max.
IDSS	Output Leakage Currents	VOUT = VDD or VSS	-1Ø µA		1Ø µA
VOL	Low Level Output Voltage	IOL = 4.Ø mA			Ø.4 V
VOH	High Level Output Voltage	IOH = -5.Ø mA	2.4 V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$


VDS (VOLTS)
-4
-3
-2
-1
P-CHANNEL OUTPUT BUFFER, 3μ
IDS vs. VDS
IDS
(MA)
-10
-20
-30


-30
-20
-10
IDS
(MA)
N-CHANNEL OUTPUT BUFFER, 3μ
IDS vs. VDS
1
2
3
4
VDS (VOLTS)

GOULD

AMI Semiconductors

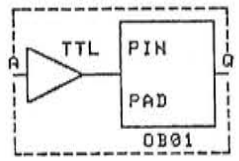
OBØ1

August 1984

Description: TTL Output Buffer

3U DOUBLE - METAL HCMOS GATE ARRAYS

Logic Symbol



Truth Table

A	Q
H	H
L	L

Input Loading

Logic Input	*Equivalent Unit Loads
A	5

*.21 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cell, No Core Devices Used

Bolt Syntax: Q .OBØ1 A;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

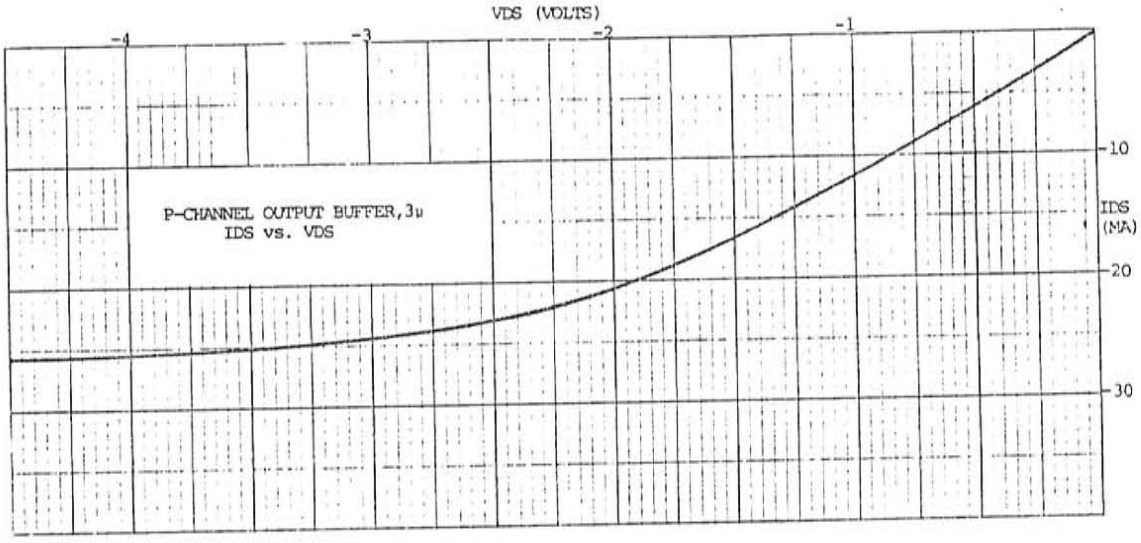
Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf	Delay Coefficients tdx	ktdx
D	Q	t_{PLH}	11.5	21.6	31.7	1.4	Ø.2
		t_{PHL}	5.Ø	8.4	11.7	1.7	Ø.1
ITTL LOAD		t_{PLH}	7.Ø	12.7	18.5	1.2	Ø.1
D	Q	t_{PHL}	5.Ø	8.4	11.7	1.7	Ø.1

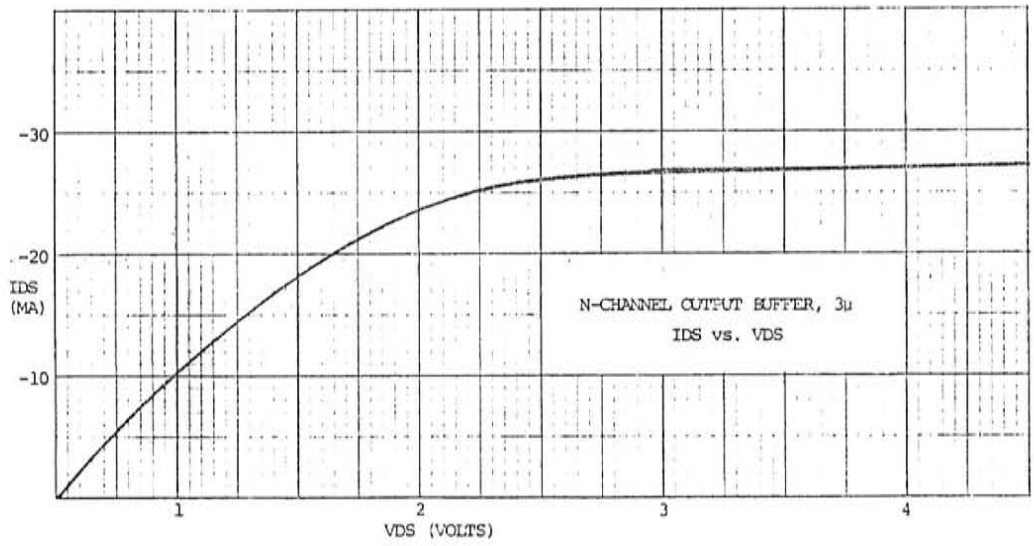
Symbol	Parameters	Conditions	Min.	Typ.	Max.
VOL	Low Level Output Voltage	IOL = 4.0 mA			0.4V
VOH	High Level Output Voltage	IOH = -5.0 mA	2.4V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

August 1984

3μ DOUBLE - METAL HCMOS
GATE ARRAYS





GOULD

AMI Semiconductors

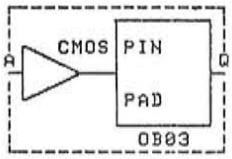
OBØ3

August 1984

Description: CMOS Output Buffer

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Logic Symbol



Truth Table

A	Q
L	L
H	H

Input Loading

Logic Input	*Equivalent Unit Loads
A	5

*.21 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cell, No Core Devices Used.

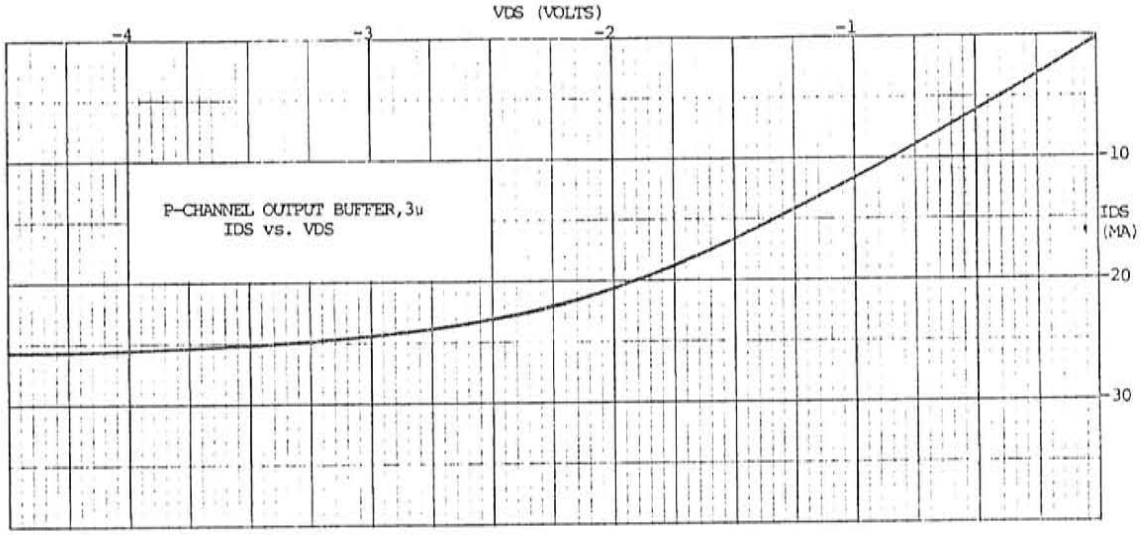
Bolt Syntax: Q .OBØ3 A;

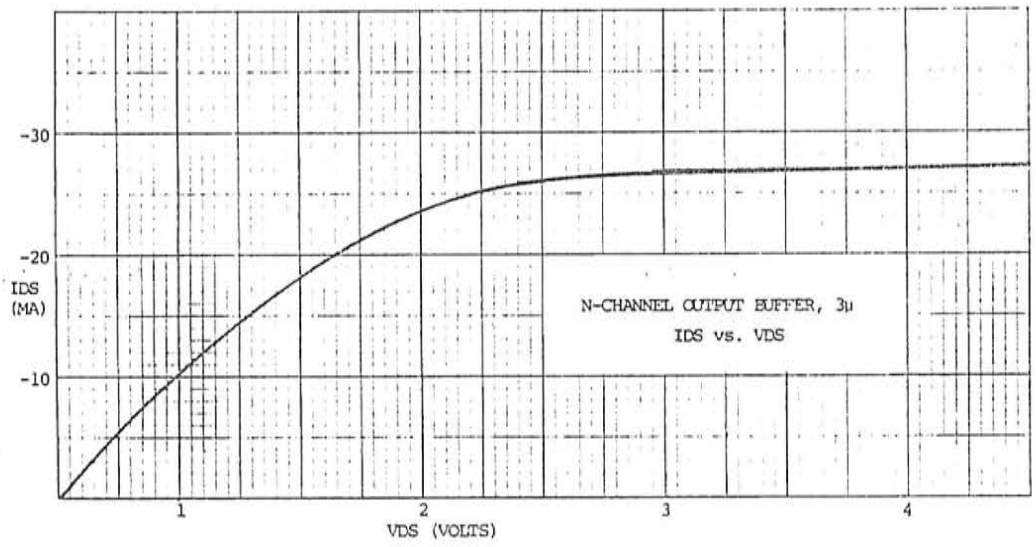
Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf	Delay Coefficients tdx	ktdx
D	Q	t_{PLH}	6.8	11.6	16.5	2.Ø	Ø.1
		t_{PHL}	5.3	8.4	11.6	2.2	Ø.1
ITTL LOAD		t_{PLH}	4.9	8.Ø	11.2	1.7	Ø.1
D	Q	t_{PHL}	8.1	13.6	19.1	2.6	Ø.1

Symbol	Parameters	Conditions	Min.	Typ.	Max.
VOL	Low Level Output Voltage	IOL = 1µA			Ø.Ø5V
VOH	High Level Output Voltage	IOH = 1µA	4.95V		

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$


VDS (VOLTS)
-4
-3
-2
-1
P-CHANNEL OUTPUT BUFFER, 3μ
IDS vs. VDS
-10
-20
-30
IDS
(MA)


-30
-20
-10
IDS
(MA)
N-CHANNEL OUTPUT BUFFER, 3μ
IDS vs. VDS
1
2
3
4
VDS (VOLTS)

GOULD

AMI Semiconductors

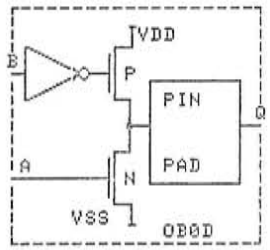
OBØD

August 1984

Description: Tri-State Output Buffers

3U SINGLE - METAL HCMOS
GATE ARRAYS

Logic Symbol



Truth Table

A	B	Q
L	H	H
L	L	Z
H	L	L
H	H	Illegal

Input Loading

Logic Input	*Equivalent Unit Loads
B	3
A	4

*.25 Pf PER UNIT LOAD

Equivalent Gate Count: Contained Within Peripheral Cells, No Core Devices Used.

Bolt Syntax: Q .OBØD A;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf
A	Q	t_{PLH}	9.3	16.2	
		t_{PHL}	5.6	1Ø.4	
ENABLE	Q	t_{PZH}	8.4	15.2	
		t_{PZL}	5.7	11.4	
DISABLE	Q	t_{PLZ}		Ø.6	
		t_{PHZ}		1.8	

Delay Coefficients tdx	ktdx
2.5	Ø.1
Ø.8	Ø.1
1.5	Ø.1
Ø.1	Ø.1
Ø.6	Ø
1.8	Ø

Symbol	Parameters	Conditions	Min.	Typ.	Max.
LDSS	Output Leakage Current		-1Ø µA		+1Ø µA
VOL	Low Level Output Voltage	IOL=4.Ø mA			.4V
VOH	High Level Output Voltage	IOH=5.Ø mA			

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

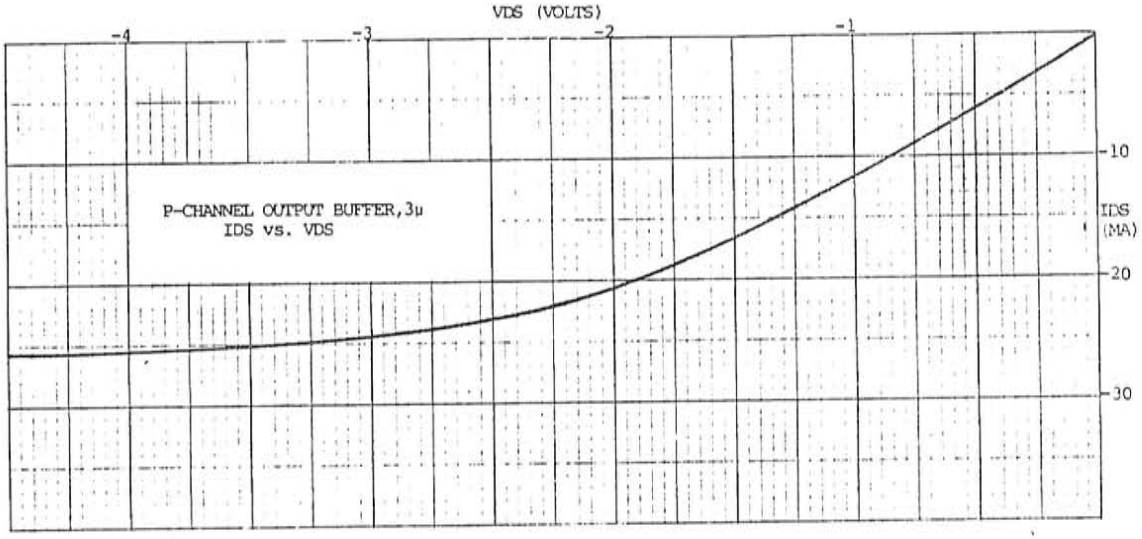
GOULD

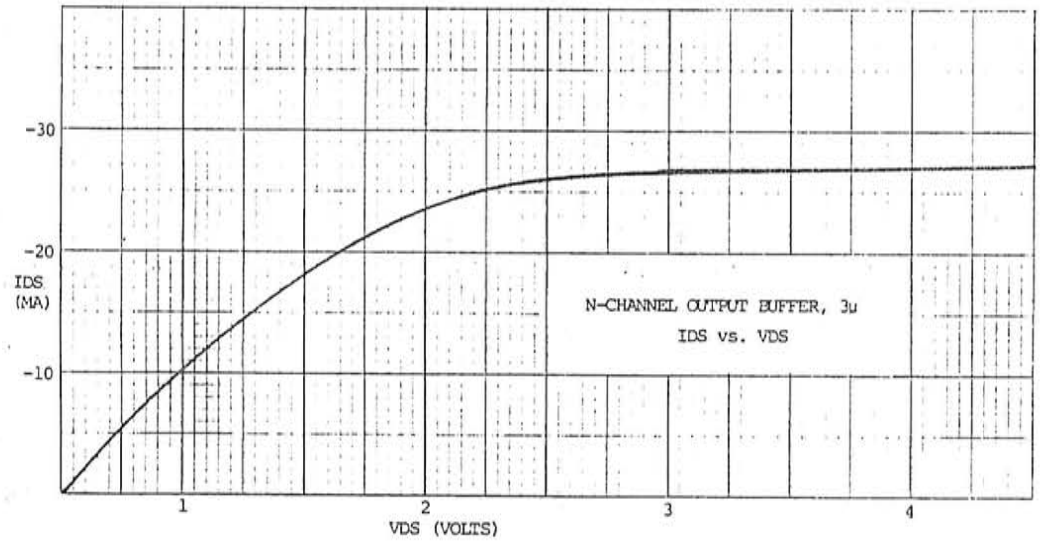
AMI Semiconductors

OBØD

August 1984

3u DOUBLE - METAL HCMOS
GATE ARRAYS





GOULD

AMI Semiconductors

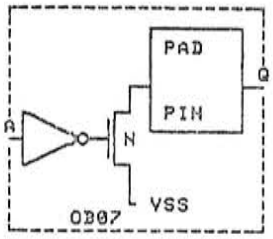
OBØ7

August 1984

Description: Open Drain Output Buffer

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Logic Symbol



Truth Table

A	Q
H	Z
L	L

Input Loading

Logic Input	*Equivalent Unit Load
A	5

*.21 Pf PER UNIT LOAD

Equivalent Gate Count: Contined Within Peripheral Cell, No Core Devices Used.

Bolt Syntax: Q OBØ7 A;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf
A 1K Resistor	Q	t_{PHL}	4.8	8.3	11.8
A 10K RESISTOR	Q	t_{PHL}	4.7	8.Ø	11.4

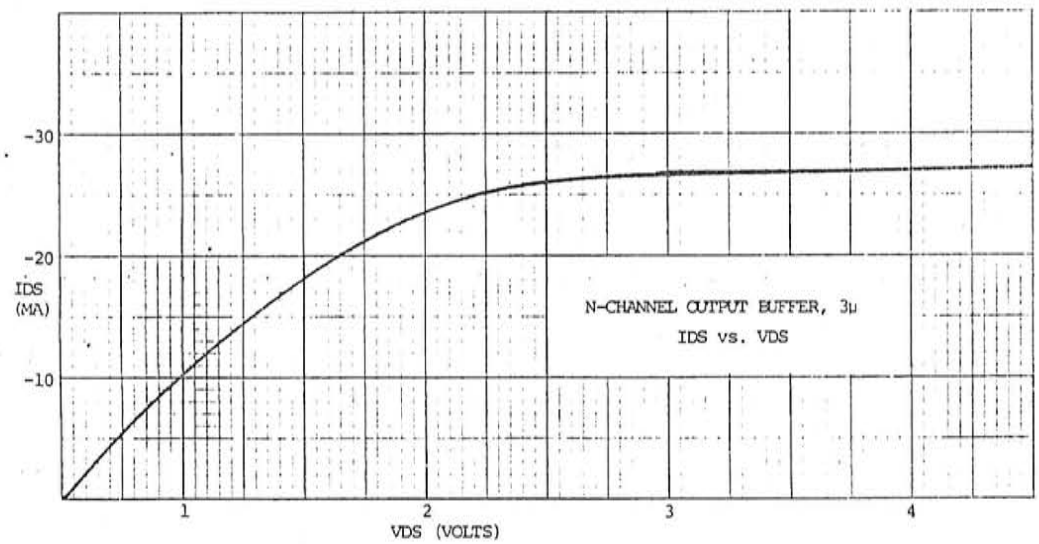
Delay Coefficients tdx	ktdx
1.3	Ø.1
1.3	Ø.1

Symbol	Parameters	Conditions	Min.	Typ.	Max.
IDSS	Output Leakage Current	VOUT=VDD			10 µA
VOL	Low Level Cutput Voltage	IOL=4.Ø mA			0.4V

Propagation Delay Equation: $tpx = tdx + ktdx * C_L$

August 1984

3μ DOUBLE - METAL HCMOS GATE ARRAYS



GOULD

AMI Semiconductors

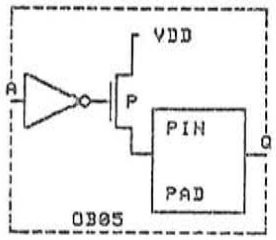
OBØ5

August 1984

Description: Open Drain Output Buffers

3U DOUBLE - METAL HCMOS
GATE ARRAYS

Logic Symbol



Truth Table

A	Q
H	H
L	Z

Input Loading

Logic Input	*Equivalent Unit Load
A	1

* .21 PF Per Unit Load

Equivalent Gate Count: Contained In The Periphery, No Core Devices Used

Bolt Syntax: Q .OBØ5 A;

Switching Characteristics: VDD = 5V, T_J = 25°C Typical Process

Max. Delay (ns) From	To	Parameter	Capacitive Load 50pf	100pf	150pf
A	Q	t_{PLH}	6.5	11.7	17.Ø

Delay Coefficients tdx	ktdx
1.3	.10

Symbol	Parameters	Conditions	Min.	Typ.	Max.
IDSS	Output Leakage Current		-10μA		
VOH	High Level Output Voltage	IOH=-1μA	4.95V		

Propagation Delay Equation: tpx = tdx + ktdx * C_L

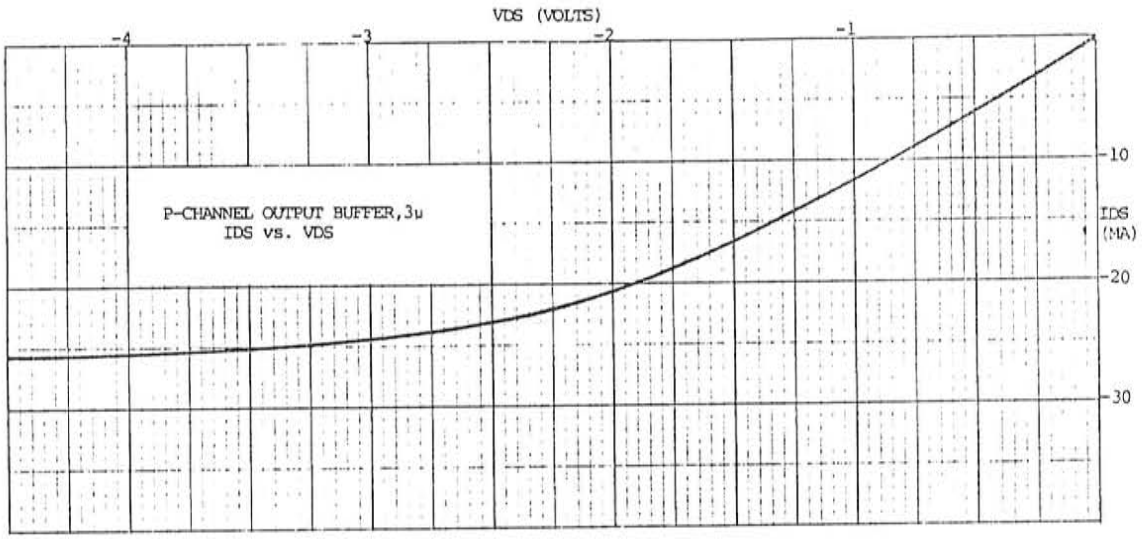
GOULD

AMI Semiconductors

0B05

August 1984

3μ DOUBLE - METAL HCMOS
GATE ARRAYS



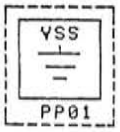
GOULD

AMI Semiconductors

PPØ1

August 1984

3µ DOUBLE - METAL HCMOS
GATE ARRAYS



.PP01

NOTE: ONE PPØ1 MUST BE USED PER GROUND (VSS) PIN.

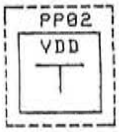
GOULD

AMI Semiconductors

PPØ2

August 1984

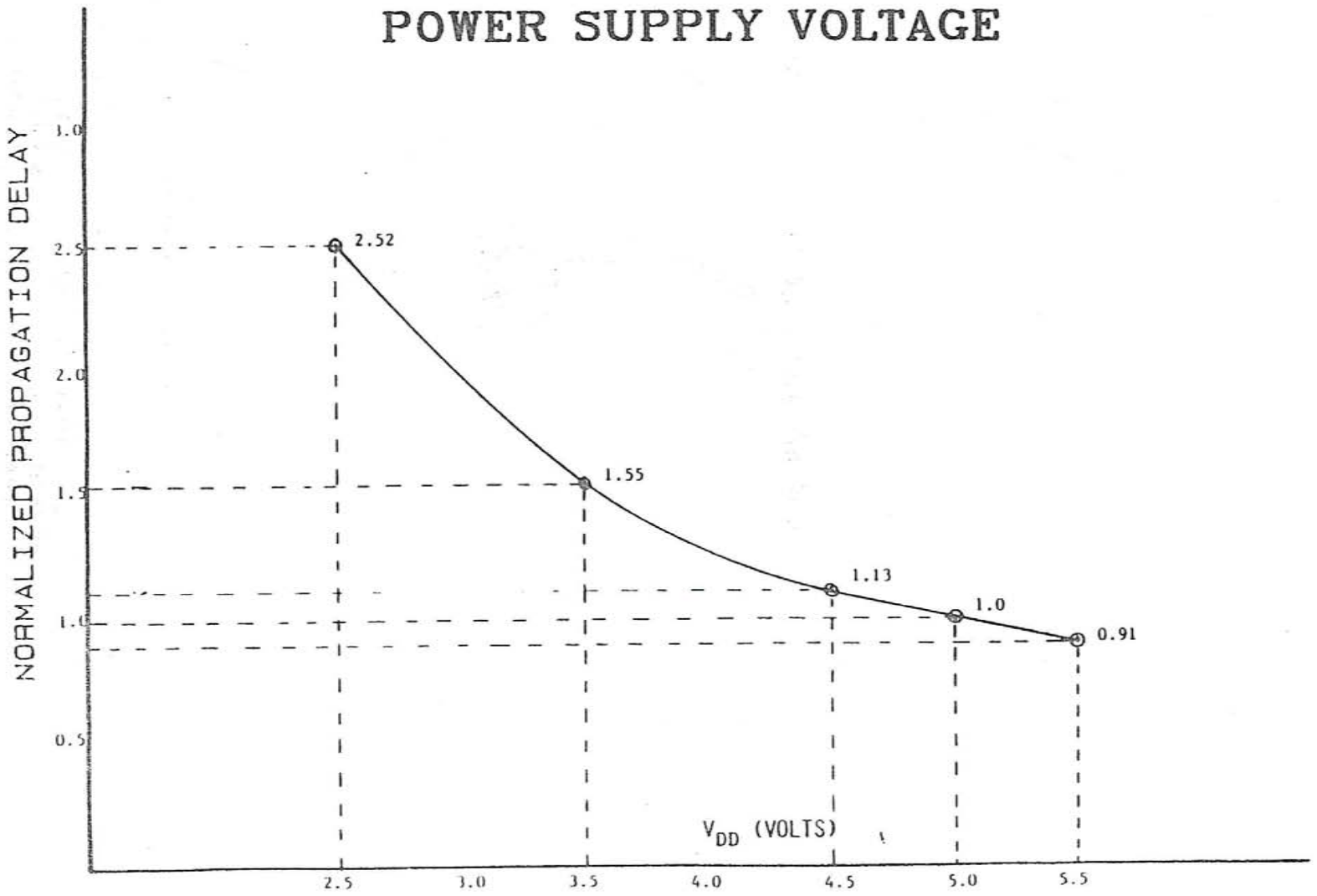
3U DOUBLE - METAL HCMOS
GATE ARRAYS



.PP02

NOTE: ONE PPØ2 MUST BE USED PER POWER (VDD) PIN.

NORMALIZED DELAYS VS. POWER SUPPLY VOLTAGE



NORMALIZED DELAYS VS. PROCESS MODEL AND TEMPERATURE

(DOUBLE METAL)