Copy of: IEEE 2002 Int. Conference on Microelectronic Test Structures, Vol. 15, April 2002

Logic Characterization Vehicle to Determine Process Variation Impact on Yield and Performance of Digital Circuits

Christopher Hess, Brian E. Stine, Larg H. Weiland,
Kazuhiro Sawada[1]

PDF Solutions Inc., San Jose, CA 95110, USA
Phone: +1-408-938-6436, FAX: +1-408-280-7915, Email: hess@pdf.com

[1]Toshiba Corporation, Kawasaki, Japan

Abstract – Manufacturing of integrated circuits relies on the sequence of many hundred process steps. Each of these steps will have more or less variation, which has to be within a certain limit to guarantee the chips functionality at a target speed. But, not every chip layout is susceptive to process variation the same way, which requires a link between process capabilities and product design. This paper will present a novel Logic Characterization Vehicle (LCV) to investigate the yield and performance impact of process variation on high volume product chips. The LCV combines and manipulates new or already documented circuits like memory cells and combinatorial logic circuits within a JIG interface that allows fast and easy testability. Beside the functionality of such circuits, also path delay as well as cross talk issues can be determined. A standard digital functional tester can be used, since all timing critical measurements will be performed within the JIG. The described method allows early implementation of existing circuits for future technology nodes (shrinks). A Design Of Experiments (DOE) based implementation of possible layout manipulations will determine their impact on yield and performance of a target design as well as its sensitivity to process variation. The described approach can be used at a much earlier stage of product and process development, which will significantly shorten yield ramp.

1 INTRODUCTION

Semiconductor manufacturing feature sizes have been reduced on a constant path over the last 30 years, which allows designers to put more and more devices and circuit components on a single chip. With every newly introduced technology node such designs will run faster as well. Given today's design and manufacturing complexity, it is crucial to maintain the bridge between chip design and manufacturing process steps to guarantee functionality, performance and yield [RoSt95]. Designing chips requires more accurate simulations not just of the single devices and parts of circuits, but also a timing and performance simulation of the entire system on a chip, that more often is a mixture of digital logic design, memory components as well as analog circuits. All these simulations rely on models that need a constantly increasing number of calibration variables that have to be extracted from a semiconductor manufacturing process. Chips containing a variety of test structures are the main vehicle to extract such variables as well as to calibrate and control each individual manufacturing process step [Bueh83]. But often these test structures do not well match the layout environment of regular product chips. Also test procedures are slow and not well adapted to a mass production yield ramp environment.

This paper will present a novel Logic Characterization Vehicle (LCV) to provide data for yield and performance improvement procedures. In contrast to typical process evaluation test chips, most of the LCV's chip area will be covered by layout pieces taken from real product chips. The LCV will further support:

- Efficient testing and circuit problem debugging
- Using a special JIG to control any embedded combinatorial circuit.
- Allow functionality test of such combinatorial circuits which are embedded in the JIG
- Ability to do a high speed performance test of such combinatorial circuit which are embedded in the JIG by connecting any number of outputs and any number of inputs of a combinatorial circuit as a ring oscillator
- Applying device and process neighborhood related changes to preexisting circuits to determine and model their impact on yield and performance
- Design Of Experiment (DOE) approach to methodical circuit manipulation
- Evaluation of yield and performance related FEOL (Front End Of Line) and BEOL (Back End Of Line) process issues
- Evaluation of yield and performance related device and gate layout/design issues
- Evaluation of yield and performance related interconnection/cross talk layout/design issues
- Evaluation of future generation of process design rules
- Evaluation of future generation of cell libraries
- Spatial or systematic failures within a die to be observed and modeled

2 Logic Characterization Vehicle (LCV) Design

The LCV must allow the functional test as well as a performance test of any circuitry. The detailed description is divided into the design flow and the test flow of the LCV. The design flow describe the different steps to design a LCV:

- Data requirements to design a LCV
- Description of a LCV module and the JIG controller
- Combinatorial circuit(s) within JIG
- FEOL/device/gate dominated combinatorial circuit (SC1)
- BEOL/interconnect dominated combinatorial circuit (SC2)
- Floorplan and DOE based design of variants

2.1 Design Data Requirements

One input to design a LCV is a set of process design rules that describe the layers and dimensions of elements drawn in these layers. So for instance connections of devices can be drawn as lines in so called interconnection layers. The design rules specify the layer name and the minimum line width of such a connection line as well as the minimum space between different connection lines. The principle design of the LCV is independent of these rules, which means that the LCV can be designed for any given set of design rules.

The second input is a library of cell elements. These cells are a layout description of logical functions. So, for instance a cell library contains such basic logical functions as NAND and NOR gates that are used to design a product chip. Cell libraries usually contain many hundred different cells to implement different logical functions. The cell libraries follow the set of design rules as described above and can be used in any combination of cells to perform the desired function of a circuit. Since the LCV just uses about a dozen very basic functions, the principle design of the LCV is independent of a cell library which means that the LCV can be designed for any given cell library. Furthermore only the layout description of the cells is required.

2.2 LCV Module Design Principle and JIG Controller

The basic module of a LCV consists of an arbitrary combinatorial circuit that will be placed in the center of a so called JIG as can be seen in Figure 1.

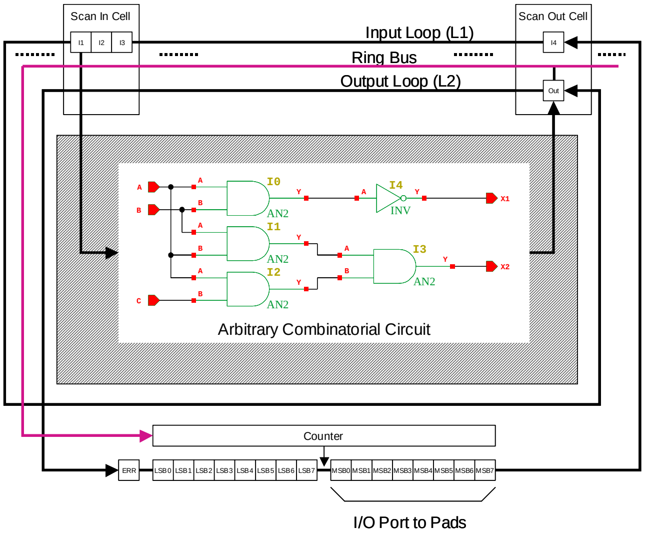


Figure 1: Basic concept of Logic CV

In general a combinatorial circuit just performs a function where its output signals solely depends on the values applied to its input signals. So, combinatorial logic circuits do not have any feedback loops. No feedback loops means the output of a gate in the circuit is not an input to the gate, direct or indirect. A combinatorial circuit does not have any memory elements and is not capable to store any data.

The JIG will control the test of the function and performance of its embedded combinatorial. Each input of the combinatorial circuit is connected with a SCAN IN cell of the JIG. Each output of the combinatorial circuit is connected with a SCAN OUT cell of the JIG. Any order of SCAN IN and SCAN OUT cells is possible in the JIG, so that they can follow any order given by the combinatorial circuit.

Through the I/O port data can be written into the JIG as well as read from the JIG. Once a word has been written into the JIG it gets shifted into the SCAN IN and SCAN OUT cells on the scan bus. This scan bus actually loops the JIG twice. The first loop, called INPUT LOOP, connects the three I-latches of each SCAN IN cell and the I-latch of the SCAN OUT cell that are dedicated to control signals that are written into the JIG. The second loop, called OUTPUT LOOP, connects the O-latch of the SCAN OUT cells and the counter cells, which are dedicated to the output information of the combinatorial circuit. Thus, one interface can handle the data transfer to and from the JIG. The two loops enable the minimal number of steps needed for the data transfer. Input data only need to be shifted through the first loop and output data only need to be shifted to the second loop. Due to the fact that both loops are connected, it is possible to write new input data while reading output data. This enables the fastest possible data access to the scan bus and reduces test time, which is very expensive.

According to the functionality of the latched as described in Table 1, the JIG can put any test pattern to the input signals of the combinatorial circuit using the latch I1 per SCAN IN cell and read the response of the output signals of the combinatorial circuit, which will be stored in the O latch of the SCAN OUT cells. So, a complete functional test can be done on the combinatorial circuit.

latch	loop type	cell type	purpose
I1	INPUT	SCAN IN	input test vector data
I2	INPUT	SCAN IN	select input of the scan cell to be connected to feedback ring bus
I3	INPUT	SCAN IN	invert signal of ring bus, if connected to the input of the scan cell
I4	INPUT	SCAN OUT	select output of the scan cell to be connected to feedback ring bus
Out	OUTPUT	SCAN OUT	output test vector data

Table 1: Functionality of the latches of the SCAN IN and SCAN OUT cells.

The SCAN IN and SCAN OUT cells are also connected to a Ring Bus, which surrounds the combinatorial circuit. Thus, the JIG can connect one or more output signals of the combinatorial circuit to one or more input signals of the combinatorial circuit like a ring oscillator to also measure the performance of the combinatorial circuit. To connect the RING BUS to an input signal of the combinatorial circuit, a “1” will be written into the I2-latch of the corresponding SCAN IN cell. To connect the RING BUS to an output signal of the combinatorial, a “1” will be written into the I4-latch of the corresponding SCAN OUT cell. If an additional inverter is needed to make the selected path ringing, a ”1” will be written in the I3-latch of the SCAN IN cell.

The JIG controller also contains the counter that is used to perform the performance measurement of the combinatorial circuit. There are two RING BUS signals, one carrying the signal of the selected outputs of the combinatorial circuit, the other one carrying the inverted signal of the selected outputs of combinatorial circuit. Each RING BUS will be hooked up to a so called primary counter. The primary counter hooked up to the non-inverted RING BUS signal is counting the rising edges occurring on this bus. The primary counter hooked up to the inverted RING BUS signal is counting the falling edges occurring on this bus. Since the frequency on the RING BUS is very high, these two primary counters are placed very close to the RING BUS to enable accurate counting. To prevent any glitches during start and stop of counting, the following approach is used:

- Connect the selected output signal (or signals) of the combinatorial circuit to the RING BUS.
- Connect the selected input signal (or signals) of the combinatorial circuit to the RING BUS. The RING BUS will now start ringing.
- After the ring frequency is stabilized, the RING BUS is connected to a pre-stage of the counter that will synchronize the start/stop signal to the RING BUS frequency. By doing so, a start/stop signal will only be passed to the counter if it does not interfere with a rising or falling edge on the RING BUS.
- Start the counter

- Count
- Stop the counter
- Deactivate the synchronizing pre-stage of the counter
- Disconnect the selected input signal (or signals) of the combinatorial circuit to the RING BUS. The RING BUS will now stop ringing.
- Counter data are ready to be read out from the tester.

Speed and area requirements of the fast primary counters lead to relatively small data. This results in a very short counting time, which may not be well controlled by the tester, since its measurement frequency is much slower than the frequency on the RING BUS. To widen up the counting window, an additional "second stage" counter is implemented. The input of this secondary counter is fed by the carry out signals of the two fast primary counters. Since this frequency is too slow to cause any timing related problems, the secondary counter can be placed anywhere in the JIG, why the interface block was chosen, to minimize the routing effort needed to include the counter results into the OUTPUT LOOP. Once, all counters hold a valid number of counted falling and rising edges, the added result can be transferred to the tester via the interface.

Each rising edge on the RING BUS will be followed by a falling edge on the RING BUS. Since the start/stop signal is synchronized to the edges it could be possible that there will be one more rising edge that falling edges or vice versa during the counting period. Thus, the difference between the counter values should not be larger than 1. However, if either rising edges or falling edges show glitches due to racing conditions of the combinatorial circuit, the difference of the two values in the 4 bit counters can become larger than 1. If this occurs, an error flag will be set and also transferred to the interface block.

The JIG also contains a chain of inverters to form a ring oscillator, once its input and output are hooked together. This simple ring oscillator can be connected to the RING BUS during a pre-test phase to allow a functionality test of the ring mode. The measurement results also provide a reference performance value of the RING BUS itself. Furthermore, the JIG can be switched into several debug modes to test the counters and other parts of the JIG to allow a "build in self test (BIST) of the JIG during the pre-test phase.

2.3 Combinatorial Circuit(s) within JIG

In general, any combinatorial circuit can be placed within a JIG to determine it's functionality and performance. So, for instance critical circuits of product chips can simple be copied and pasted into a JIG to study and solve observed problems.

But also, much more general combinatorial circuit can be chosen to allow any process and design related problem solving. Circuits consist of devices that are combined to combinatorial cells, also called gates, that are provided in a cell library. These elements are hooked up together to blocks that perform a given specification. So, there are basically two levels "devices/gates" as well as "interconnection" that need to be explored for yield and performance measurements.

During circuit manufacturing devices and interconnection are manufactured in different layout. The Front End Of Line (FEOL) is manufacturing the devices and the Back End Of Line (BEOL) is manufacturing the interconnections. So, it is recommended to use two JIGs one containing an FEOL/device/gate dominated logic circuit and one containing an BEOL/interconnect dominated logic circuit.

2.3.1 FEOL/device/gate dominated logic combinatorial circuit (SC1)

To get information about yield and performance of devices and gates, a FEOL/device/gate dominated logic circuit will be placed as combinatorial circuit within a JIG. So, the interconnection part of the circuit should be as little as possible. Also the functionality of the circuit should:

- Provide easy functionality for easy testing
- Contain as many gates as possible to fill a significant chip area with as little input and output signals as possible
- Allow back-tracing of failed test pattern to the device/gate that has caused the circuit to fail
- Contain different number of gates between input and output signals to allow conclusions on the path delay dependant on the JIG controlled performance measurements.

One possible circuit that fulfills these requirements is a combinatorial circuit to calculate the square root of an binary integer value:

OUTPUT = sqrt (INPUT)

So, if for instance the binary integer value 111111 (64) will be applied to the input signals of this circuit the output will provide the binary integer value 111 (8). It's obvious, that the functional description is easy (just one line) and due to that the generation of test vectors is easy as well.

If the circuit has n output signals, twice as many input signals are needed. The number of gates and therefore also the area for the circuit implementation quadraticly grows with the number of input and output signals. Back-tracing of faulty test vectors as well as and different path lengths depends on how the square root function is implemented. The implementation chosen here is based on the Newton-Raphson algorithm [Hwan79], [WaFl82]. A square root combinatorial circuit (SQRT) is implemented with 16 input and 8 output signals. Since it is known that gate density has a significant impact on circuit yield and performance, three SQRTs are placed and routed within one JIG to form the FEOL/device/gate dominated logic circuit (SC1) as can be seen in Figure 2. The schematics of the three combinatorial circuits (in this example 3 SQRT circuits) are identical, but they are placed and routed differently to achieve three levels of gate and device density as can be seen in Figure 3.

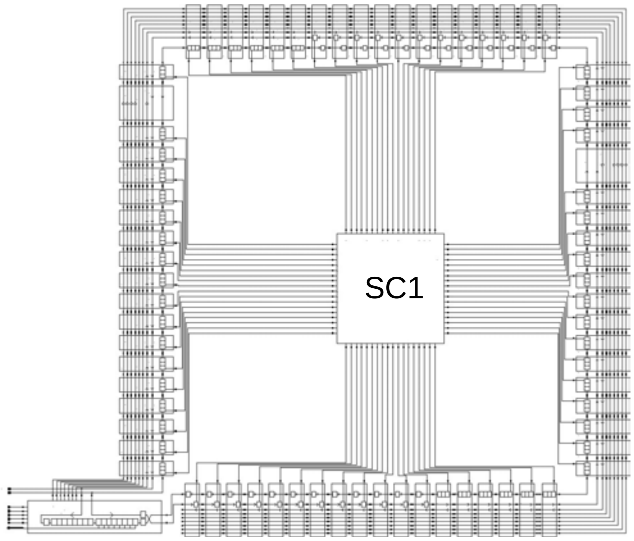


Figure 2: SC1 within a JIG.

Each SQRT has about 18,000 devices. SQRT A is 1.5 times larger than SQRT B, which is 1.5 times larger than SQRT C. Table 2 summarizes the layout properties of this design. The diffusion and poly density significantly changes across the three SQRT designs, while the metal density is mostly invariant. However, the overall length of metal does correlate with block size as expected.

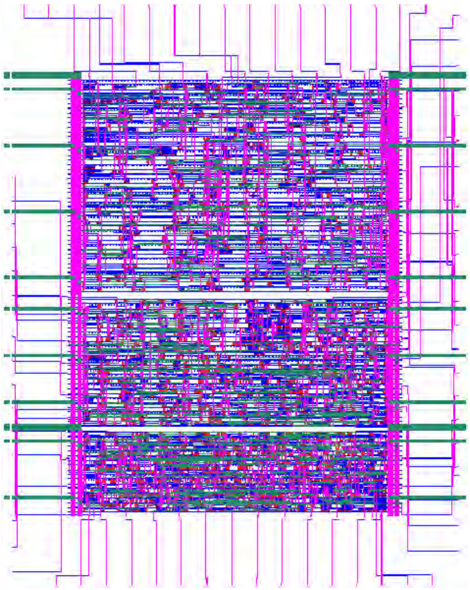


Figure 3: SC1 layout.

Sub-block	Density					Total Number			Overall length of Metal
	AA	Poly	M1	M2	M3	Cont	Via	Via2	[cm]
SQRT A	25%	5%	20%	11%	3%	34628	5824	1241	~ 9
SQRT B	31%	8%	22%	11%	4%	25604	4645	1261	~ 7
SQRT C	40%	12%	26%	12%	5%	20604	3961	1423	~ 5

Table 2: Layout properties of a SC1 design.

2.3.2 *BEOL/interconnect dominated logic combinatorial circuit (SC2)*

The purpose of the BEOL/interconnection dominated combinatorial circuit placed within a JIG is to get information about yield and performance of interconnection elements. So, the device/gate part of the circuit should be as little as possible. Also the functionality of the circuit should:

- Provide easy functionality for easy testing
- Individually connect as many interconnection line as possible to fill a significant chip area with as few input and output signals as possible
- Allow back-tracing of failed test pattern to the interconnection line and level that has caused the circuit to fail
- Contain different lengths of interconnection lines between input and output signals to allow conclusions on the path delay dependant on the JIG controlled performance measurements
- Allow cross talk related yield and performance loss

One possible circuit that fulfills these requirements is a ring oscillator that uses a long interconnection line between two gates. For the BEOL/interconnection dominated logic circuit (SC2), a couple of different lines will be actually placed between two gates of a ring oscillator. In general, any number of gates in the chain of the ring oscillator can be implemented, but the chain length should correspond to the number of gates in a typical path of a product chip. After an odd number of gates the signal will be connected to a group if interconnection lines. The lines are grouped in line style and line length. The line style can include but is not limited to:

- Bus type parallel running lines
- Randomly routed lines like on the left side of Figure 4
- Level/layer of interconnection line
- Single layer lines
- Via chains

The different line length values should at least cover one magnitude of values. Table 3 shows a possible “Design of Experiments” (DOE) for line styles and line length values.

Design	Variants	Parameter
line type	2	random wiring and bus wiring
line length	5	1mm, 2.5mm, 5mm, 10mm, 20mm
gate type	6	inverter, 2 input AND, 2 input NAND, tri-state buffer, 2:1 Mux (data path), 2:1 Mux (select path)
	60	total number of experiments

Table 3: Design of Experiment (DOE) for BEOL lines in SC2.

In general, the number of implemented lines is only limited by the number of SCAN IN cells and SCAN OUT cells of the JIG. To allow cross talk related performance measurements, each line can be set to a couple of possible signals. So, for instance each line can be:

- Set to constant “1”
- Set to constant “0”
- Connected within the ring oscillator chain (full frequency)
- Connected to the ring oscillator through a frequency divider (half frequency)
- Connected to a shifted ring oscillator frequency
- floating

For example, in the SC 2 described in Table 3 and Figure 4, a total of 10 interconnection lines have been implemented. In general, any number of lines can be implemented. To fill a given area within a JIG the BEOL/interconnection dominated combinatorial circuit (SC2) will actually contain not just one but a couple of ring oscillators with the same interconnection line configuration as described above. The difference between the ring oscillators is the kind of gate chosen. Table 3 shows a possible DOE for an SC2 having 6 different gates: NAND, AND, INVERTER, BUFFER, MUX data path, MUX selection path. In general, any number of gates can be chosen to fill up a given area within a JIG. The right side of Figure 4 shows the layout of this example SC2. The gates of the ring oscillators are in the center of the circuit. There are two interconnect blocks to the left and right of the center block.

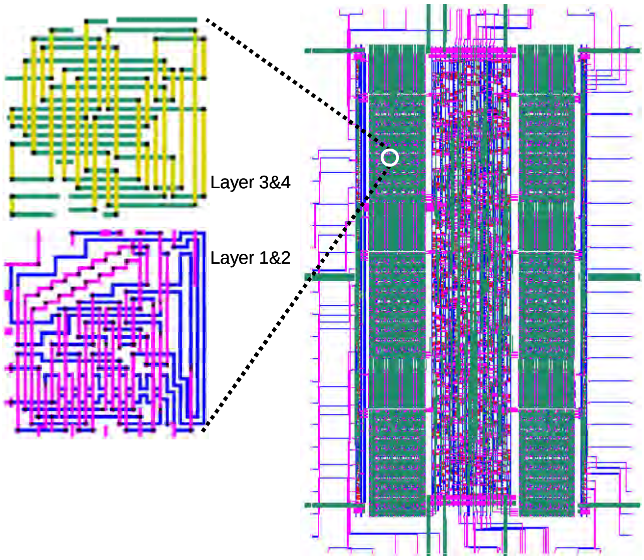


Figure 4: SC2 layout and example of random routing blocks.

2.4 Floorplan and DOE Based Design of Variants

Since memory blocks are widely used in product chips and known for easy functionality test they are also often used in test chips for yield and performance evaluation [MBRR01]. To allow parallel access and test of at least one memory block in conjunction with the JIGs, all signals of the memory block will be hooked up to the same data transfer bus that the JIGs are using. Figure 5 shows a LCV Module of one memory block and two Jigs. A memory block can include but is not limited to ROM, RAM, DRAM, SRAM, SDRAM, EDRAM, FlashRAM, EEPROM.

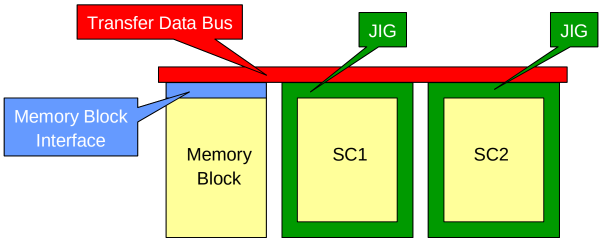
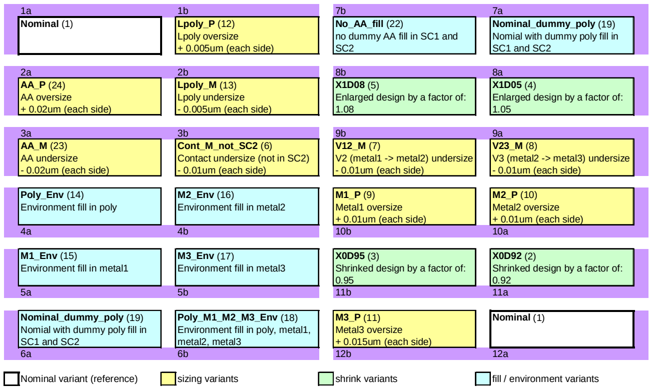


Figure 5: Basic design of a of LCV Module

Usually, the size of the memory block is taken to determine the minimum area required to efficiently design a LCV Module. One dimension (e.g. height) of the JIGs is chosen to match one dimension of the memory block to form a rectangle that contains the memory block, SC1 and SC2. Then the combinatorial circuits are designed to fit into the JIGs, which is of no problem, since the second dimension (e.g. width) for each JIG can be selected arbitrary.

The goal of the LCV is to provide yield and performance related information. Thus design and process related issues have to be included into the design. Device/gate related design issues are implemented in SC1. Interconnection related design issues are implemented inside SC2. Nevertheless during manufacturing a variety of changes can be applied to the circuits that are not part of the design. These changes are either due to the process steps applied during manufacturing and/or due to the design/layout neighborhood environment. So, a Design Of Experiment (DOE) approach on such changes is applied to selected parts and/or layer/level of the combinatorial circuits. Such DOE levels can include, but are not limited to parameters as can be seen in Figure 6, where beside the nominal reference variants three more DOE groups are implemented:

- Sizing group to determine sensitivity to selected resizing of layout properties like contact and via size or metal line width.
- Environment group to determine sensitivity to dummy fill in selected layers as can be seen in Figure 7.
- Shrink/Enlargement group to determine sensitivity to shrunk and enlarged circuits.



1a **Nominal** (1)	1b **Lpoly_P** (12) Lpoly oversize + 0.005um (each side)	7b **No_AA_fill** (22) no dummy AA fill in SC1 and SC2	7a **Nominal_dummy_poly** (19) Nomial with dummy poly fill in SC1 and SC2
2a **AA_P** (24) AA oversize + 0.02um (each side)	2b **Lpoly_M** (13) Lpoly undersize - 0.005um (each side)	8b **X1D08** (5) Enlarged design by a factor of: 1.08	8a **X1D05** (4) Enlarged design by a factor of: 1.05
3a **AA_M** (23) AA undersize - 0.02um (each side)	3b **Cont_M_not_SC2** (6) Contact undersize (not in SC2) - 0.01um (each side)	9b **V12_M** (7) V2 (metal1 -> metal2) undersize - 0.01um (each side)	9a **V23_M** (8) V3 (metal2 -> metal3) undersize - 0.01um (each side)
Poly_Env (14) Environment fill in poly 4a	**M2_Env** (16) Environment fill in metal2 4b	**M1_P** (9) Metal1 oversize + 0.01um (each side) 10b	**M2_P** (10) Metal2 oversize + 0.01um (each side) 10a
M1_Env (15) Environment fill in metal1 5a	**M3_Env** (17) Environment fill in metal3 5b	**X0D95** (3) Shrinked design by a factor of: 0.95 11b	**X0D92** (2) Shrinked design by a factor of: 0.92 11a
Nominal_dummy_poly (19) Nomial with dummy poly fill in SC1 and SC2 6a	**Poly_M1_M2_M3_Env** (18) Environment fill in poly, metal1, metal2, metal3 6b	**M3_P** (11) Metal3 oversize + 0.015um (each side) 12b	**Nominal** (1) 12a

□ Nominal variant (reference) 🟨 sizing variants 🟩 shrink variants ▨ fill / environment variants

Figure 6: Floorplan of the DOE variants.

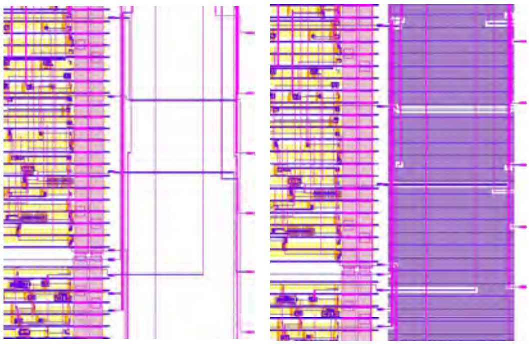


Figure 7: Comparison of cell without environment dummy fill (left) and cell with environment dummy fill (right).

The total number of DOE levels is chosen according to the maximal chip size. Figure 8 shows an entire LCV with 12 units each containing 2 LCV modules.

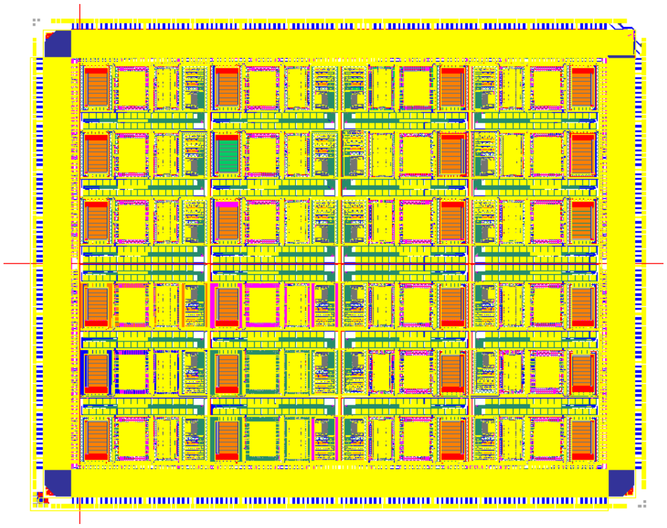


Figure 8: Layout of the entire Logic Characterization Vehicle.

3 TEST

Three different operating modes are available to test the combinatorial circuits embedded in a JIG:

- Functional test of combinatorial circuit
- Ring test of combinatorial circuit
- JIG test

3.1 Function Test of Combinatorial Circuit

There is a sequence of steps to access and control the JIG, which is summarized in Table 4.

ID	JIG Behavior			
	scan path	ring bus	ring -> counter	counter
1	store pad data in 8 counter latches	sleep	not connected	reset
2	shift	sleep	not connected	reset
3	tester reads data from 8 counter latches	sleep	not connected	reset
4	store SC1, SC2 and counter results in latches	only output of SC1 or SC2 connected	not connected	sleep
5	put data in input latches to SC1, SC2	only output of SC1 or SC2 connected	not connected	sleep
6	sleep	output and input(s) of SC1 or SC2 connected	not connected	sleep
7	sleep	osc.	synchronize signals	sleep
8	sleep	osc.	connected	counts edges on ring bus

Table 4: Control signals of the JIG.

To perform the functional test of the combinatorial circuit, which is embedded in the JIG, the steps described in this table have to be applied in the order described in Figure 9.

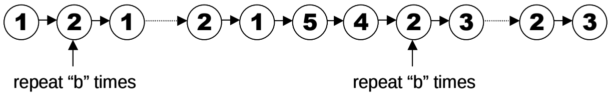


Figure 9: Functional Test Sequence.

First data will be written into the JIG via the TDB (ID 1). Then these data will be shifted into the SCAN IN cells, so that the data can be applied to the combinatorial circuit embedded in the JIG (ID 2 for each data shift). The sequence of steps 1 and 2 will be repeated until all latches of the SCAN IN and SCAN OUT cells are loaded, which are on the INPUT LOOP. Then in step ID 5 the data of the SCAN IN cells will be forwarded to the input signals of the combinatorial circuit. In the next step the response of the combinatorial circuit to its output signals will be loaded into the SCAN OUT cells (ID 4). After shifting the first b data bits of the latches that are on the OUTPUT LOOP into the interface block (ID 2), the tester can read the data from the JIG (ID 3). The sequence of steps 2 and 3 will be repeated until all output data have been read by the tester.

3.2 Ring Test of Combinatorial Circuit

To perform the ring test of the combinatorial circuit, which is embedded in the JIG, the steps described in Table 4 have to be applied in the order described in Figure 10.

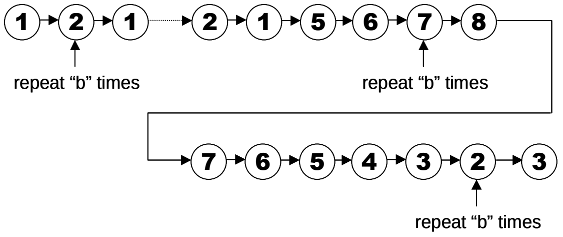


Figure 10: Performance Test Sequence.

First, data will be written into the JIG via the TDB (ID 1). Then these data will be shifted into the SCAN IN cells, so that the data can be applied to the combinatorial circuit embedded in the JIG (ID 2 for each data shift). The sequence of steps 1 and 2 will be repeated until all latches of the SCAN IN and SCAN OUT cells are loaded, which are on the INPUT LOOP. Then in step ID 5 the data of the SCAN IN cells will be forwarded to the input signals of the combinatorial circuit. Furthermore, selected output signals of the combinatorial circuit will be connected to the RING BUS. In step ID 6 selected input signals will be connected to the RING BUS, which will start the oscillation of the circuit. In step ID 7 the frequency on the RING BUS will be synchronized with the start/stop signal of the counters. Finally, in step ID 8 the counter will start counting the rising and falling edges on the RING BUS. Going back to step ID 7 will stop the counter, so that they now hold the total number of edges counted during the time step ID 8 was active. After disconnecting the synchronizing circuit (ID 6) and disconnecting the input signals of the combinatorial circuit from the RING BUS (ID 5), the RING BUS will stop oscillating. In step ID 4 the counter results will be forwarded to the interface block of the JIG. Now, the tester can read the data from the interface block of the JIG (ID 3). Then, the data in the latches that are on the OUTPUT LOOP as well as the remaining counter bits have to be shifted into the interface block (ID 2). The sequence of steps 3 and 2 will be repeated until all output data have been read by the tester.

3.3 JIG Test

Before testing the combinatorial circuits, which are embedded in a JIG, it’s necessary to test the functionality of the JIG itself. A first test will check the TDB and the ability to write data into the JIG and read them back. Once this has been successful, there will be applied a so called SCAN Test. Here data will be written into the JIG and then shifted through all latches on the two loops, until the same data appear in the I/O interface again. Then these data can be read from the tester and should be identical to the data that originally have been written into the JIG. The inverter chain implemented within the JIG will be used to check the functionality of the counters.

The “pre-test” JIG functionality results as well as the reference frequency measurements of the internal inverter chain of the JIG provide:

- Spatial (within die) yield trends, which are independent on DOE changes applied to the embedded combinatorial circuit
- Spatial (within die) performance trends, which are independent on DOE changes applied to the embedded combinatorial circuit
- Capability to deconvolve spatial (within die) trends from the DOE dependent changes for each individual combinatorial circuit inside the JIG of the LCV

4 Experimental Results

A Logic Characterization Vehicle has been manufactured at Toshiba Corporation, Japan. Followed are several experimental results. Figure 11 shows the performance difference between two implemented inverter chains in the JIG, one always staying at nominal design rules, while the other will follow the specified DOE changes. Shrunk designs as well as undersizing poly show the most significant improvements. Also, dummy fill has a positive impact on performance. Sizing of vias does not show an impact, which indicates that the BEOL delay is capacity and not resistance dominated. As expected, enlarging a circuit and oversizing poly will decrease the circuit performance.

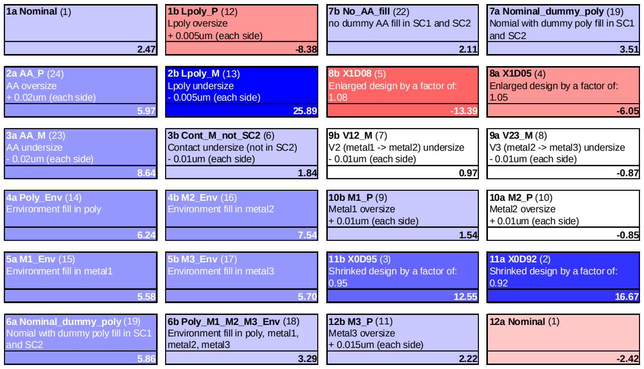


Figure 11: Performance evaluation dependent on the variant DOE.

Figure 12 shows more details on the dummy fill impact on circuit performance taken from the BEOL/Interconnect dominated SC2 circuit. Implementing dummy fill in metal layers will improve circuit performance. The impact of poly fill is smaller, since the SC2 circuit only contains a very limited number of devices. The dependency on the gate type used in the SC2 circuit path can also be seen. As expected an inverter is the fastest gate type while the multiplexer is the slowest gate type especially if connected through the select path.

Figure 13 summarizes how the interconnection routing style impacts the circuit performance. For the random type routing the speed continuously decreases as the total line length increases (R1-

R20). However, for bus type routing, only a 20 mm long line shows a significant performance loss.

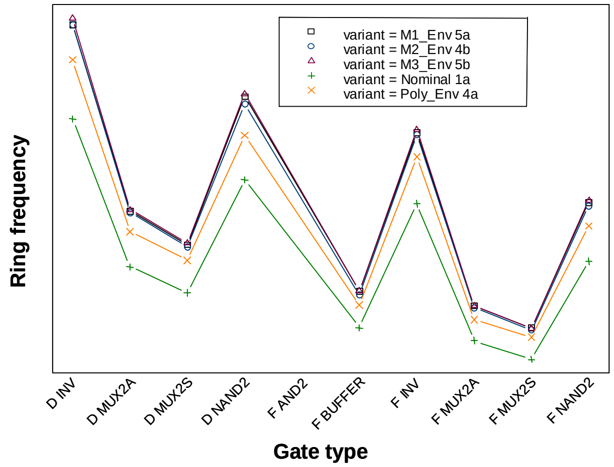


Figure 12: SC2: Performance evaluation dependent on the gate type and environment variant DOE.

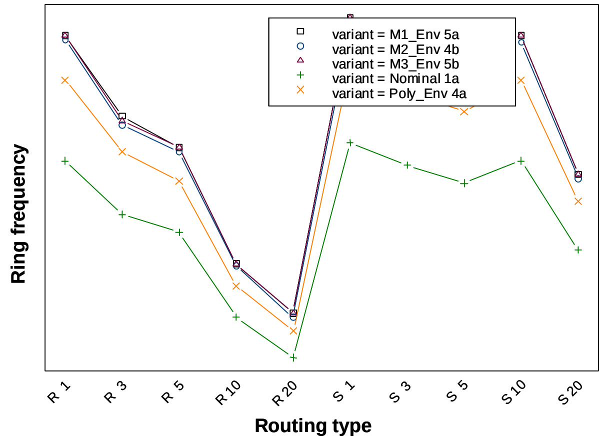


Figure 13: SC2: Performance evaluation dependent on routing line type (random(R) and bus (S)), line length, and environment variant DOE.

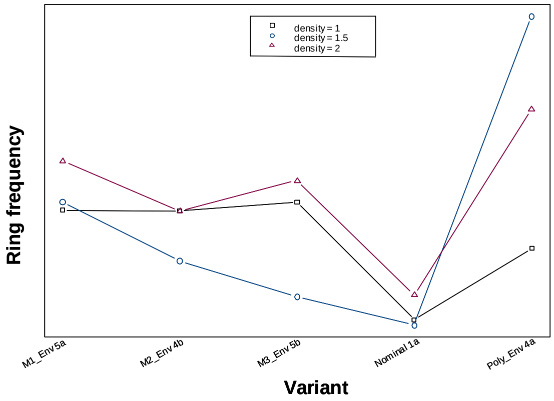


Figure 14: SC1: Performance evaluation dependent on circuit and environment variant DOE.

Figure 14 shows how the placement and routing (P&R) will effect the performance of the FEOL/gate/device dominated SC1 circuit. Any kind of dummy fill has a positive impact on the circuit performance. The low density P&R as well as the high density P&R show the best results.

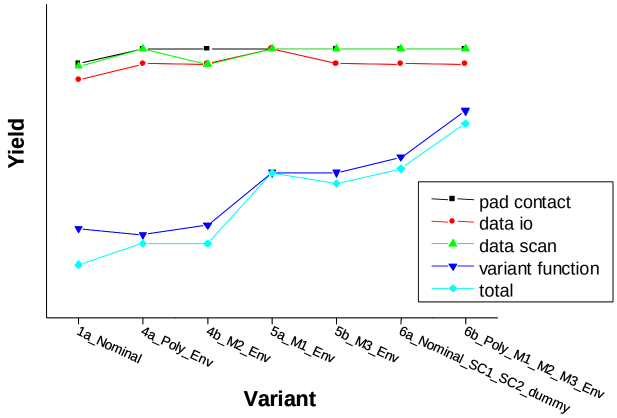


Figure 15: SC2: Yield evaluation of all dies on one wafer dependent on test sequence and environment variant DOE.

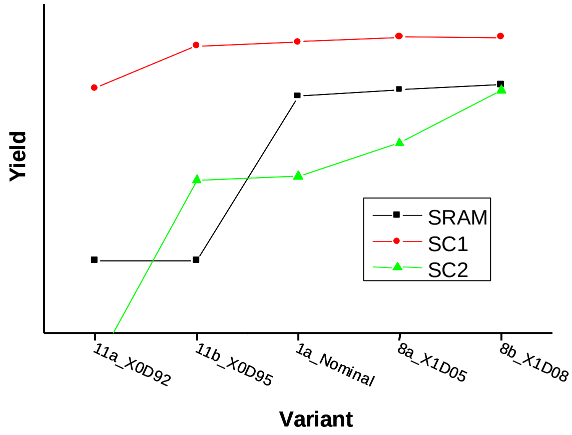


Figure 16: Yield evaluation dependent on circuit and shrink variant DOE.

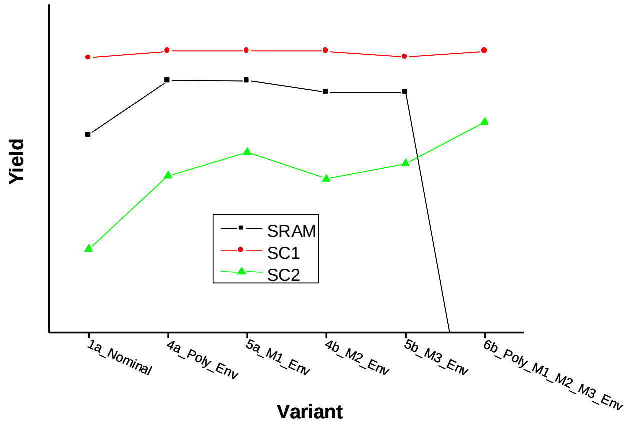


Figure 17: SC2: Performance evaluation dependent on routing line type (random(R) and bus (S)), line length, and environment variant DOE.

Figure 15 summarizes the yield of the SC2 circuit dependent on test mode and dummy fill. The pad connection test, and the JIG test (data io as well as scan test) show very high yield. However, there is a clear trend that the more dummy fill will be placed in metal layers the better the overall yield will be for the SC2 circuit.

Figure 16 shows the yield trend for the SC1 and SC2 circuit as well as an SRAM dependent on shrinking or enlarging the circuit. While SC1 only shows a significant yield loss for the 92% shrunk version, the SRAM and the SC2 circuit are already affected by shrinking them to 95%.

Looking at the dummy fill variants as shown in Figure 17 clearly indicate that the SRAM, the SC1 and the SC2 circuit show a different behavior, why it is insufficient to just use memory type test chips to do a yield assessment for circuits that are dominated by random logic.

Based on such LCV data as shown above, methods and procedures as for instance described at [CiLS00] can be used to implement specific design changes or selected process step changes to achieve yield and performance enhancements.

5 Conclusion

A novel approach has been described to explore the yield and performance dependency between chip layout and chip manufacturing. The Logic Characterization Vehicle (LCV) allows the implementation of a layout taken from a product chip within a JIG interface. Any combinatorial circuits as well as memory blocks can be part of this approach. The JIG guarantees fast and easy evaluation of the layouts functionality as well as its path delay using a standard digital tester. Different layout manipulations can be implemented to define the most robust layout regarding process variation. Careful selection of layouts to be implemented within a JIG can further allow the identification of possible layer specific process problems in FEOL and BEOL process steps. Also, possible systematic gate and device problems can be detected. Furthermore, the experimental results have shown that the pure usage of memory type test chips is insufficient to describe process related yield loss in such product chips that are dominated by random logic circuit blocks.

The LCV can be used in a very early stage of design and process development, since it can be implemented using the cell libraries of previous design libraries. Thus, yield and process related data will be available even before mass production starts. Based on these information final layout and process changes can be made to significantly reduce the time to ramp yield and performance specifically for high volume products.

Acknowledgement

The authors like to thank Richard Burch for many helpful data analysis discussions. We also like to thank Akira Yamaguchi and Koji Miyamoto for fruitful technical discussions. The authors also like to thank Seiji Sakurai and Akira Matsuoka for testing the digital parts of the LCV and Tomoko Ishikawa for global routing support.

References

[Bueh83] Buehler, M. G.
Microelectronic Test Chips for VLSI Electronics
VLSI Electronics Microstructure Science, pp. 529-576, Vol 9, Chap.9, Academic Press, 1983

[MBRR01] Daniel N. Maynard, Bette Bergman Reuter, Raymond J. Rosner
SwampFinder
Advanced Semiconductor Manufacturing Conference (ASMC), Munich, Germany, pp. 151-155, April 2001

[RoSt95] Staper, C. H., Rosner, R. J.
Integrated Circuit Yield Management and Yield Analysis: Development and Implementation
IEEE Transactions on Semiconductor Manufacturing, pp. 95-102, Vol. 8, No. 2, 1995

[CiLS00] Ciplickas, D. J., Li, X., Strojwas, A. J.
Predictive Yield Modeling of VLSI's
International Workshop on Statistical Metrology (IWSM), 2000

[Hwan79] Hwang, K.
Computer Arithmetic: Principles, Architecture and Design
John Wiley & Sons, USA, 1979

[WaFl82] Waser, S., Flynn, M. J.
Introduction to Arithmetic for Digital System Designers
CBS College Publishing, 1982