

Educational Tool for the Demonstration of DFT Principles Based on Scan Methodologies

Josef Strnadel and Zdeněk Kotásek
Faculty of Information Technology
Brno University of Technology



Presentation Outline

- **Scan Educational Tool (SET)**
- **Description of particular SET windows**
- **Scan-layout selection example**
- **Testability analysis example**
- **Loop-breaking example**
- **Test data propagation paths browser example**
- **Automated DFT process example**
- **SET requirements and limitations**
- **Conclusions**

Scan Educational Tool (SET)

Main window

- TA/DFT selection
- Circuit info
- View selection
- Design constraints settings
- GA settings
- History

Circuit schema

- Scan layout built in the circuit structure
- Diagnostic data path browser

Layout selection

- Selection of registers into scan chains
- Scan layout visualization

TA Results

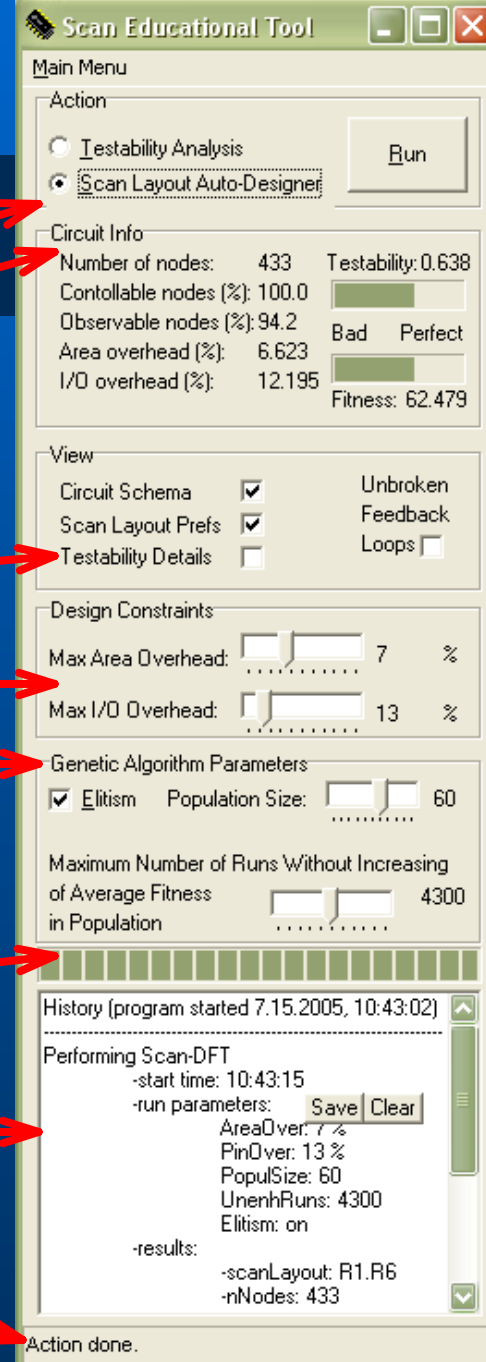
The screenshot displays the Scan Educational Tool (SET) interface with four main panels:

- Main window:** Contains a Main menu with 'Testability Analysis' and 'Scan Layout Auto-Designer' options. It also shows circuit statistics: 446 nodes, 93.7% controllable nodes, 93.7% observable nodes, 16.556% area overhead, 21.951% I/O overhead, 0.000 fitness, and 0.720 testability.
- Circuit Schema:** A logic diagram showing various components like multiplexers (MUX), registers (REG), and arithmetic units (ADD, SUB, COMP). Colored lines (green, blue, red) trace paths through the circuit, representing test data propagation paths.
- Scan Layout Selection:** A configuration panel for registers and scan chains. It lists registers REG1 through REG6 and their assigned scan chains (A, B, C, D) and in-chain orderings (1st, 2nd, 3rd).
- Testability Details:** A table showing testability metrics for various components.

| Port_name(width) | Con | Obs | Tst |
|------------------|----------|----------|---------|
| mux6.q(8) | 0.996094 | 0.622350 | 0.80922 |
| mux6.b(8) | 0.926946 | 0.619820 | 0.77338 |
| mux6.a(8) | 1.000000 | 0.619820 | 0.80991 |
| mux5.q(8) | 0.800037 | 0.800062 | 0.80005 |
| mux5.b(8) | 0.776507 | 0.796912 | 0.78670 |
| mux5.a(8) | 0.803199 | 0.796912 | 0.80005 |
| mux4.q(8) | 0.992188 | 0.587725 | 0.78995 |
| mux4.c(8) | 0.803199 | 0.582908 | 0.69305 |
| mux4.b(8) | 0.776507 | 0.582908 | 0.67970 |
| mux4.a(8) | 1.000000 | 0.582908 | 0.79145 |
| mux3.q(8) | 0.996094 | 0.604614 | 0.80035 |
| mux3.b(8) | 1.000000 | 0.602215 | 0.80110 |
| mux3.a(8) | 0.803199 | 0.602215 | 0.70270 |
| mux2.q(8) | 0.923163 | 0.774863 | 0.84901 |

SET: Main window

- Action selection
- Circuit information (valid after action is done)
- Selection of windows to view
- Design constraints (area, I/O) settings
- Genetic algorithm settings
 - Elitism
 - Population size
 - Max # of runs
- Progress bar
- History (setting+result logging) window
- Status bar (current action info)



Scan Educational Tool

Main Menu

Action

Testability Analysis **Scan Layout Auto-Designer**

Circuit Info

Number of nodes: 433 Testability: 0.638
 Controllable nodes (%): 100.0
 Observable nodes (%): 94.2 Bad Perfect
 Area overhead (%): 6.623
 I/O overhead (%): 12.195 Fitness: 62.479

View

Circuit Schema Unbroken
 Scan Layout Prefs Feedback
 Testability Details Loops

Design Constraints

Max Area Overhead: 7 %
 Max I/O Overhead: 13 %

Genetic Algorithm Parameters

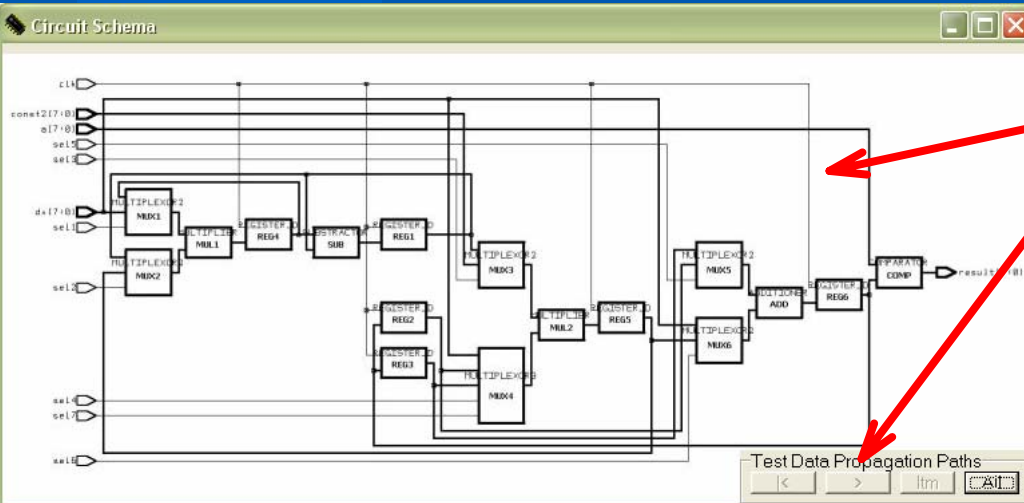
Elitism Population Size: 60
 Maximum Number of Runs Without Increasing of Average Fitness in Population: 4300

History (program started 7.15.2005, 10:43:02)

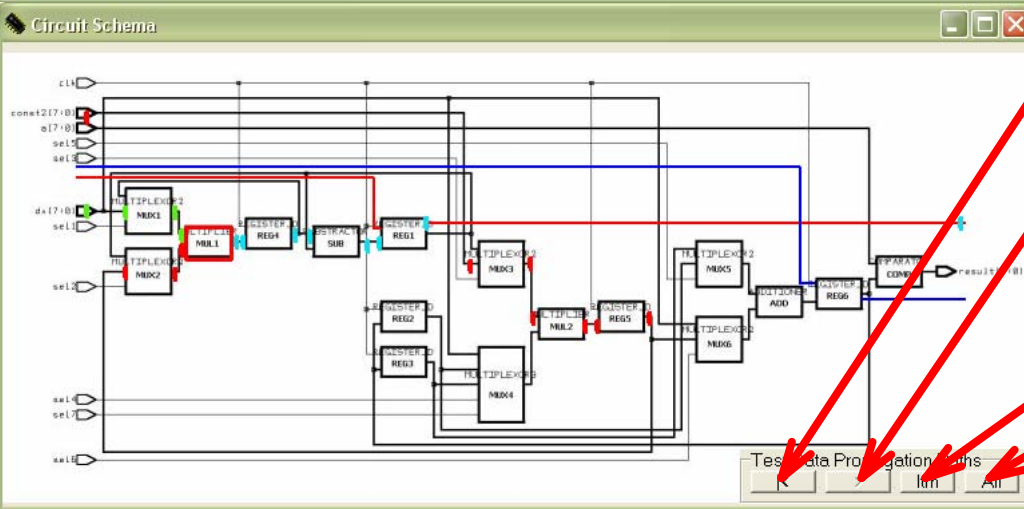
Performing Scan-DFT
 -start time: 10:43:15
 -run parameters:
 AreaOver: 7 %
 PinOver: 13 %
 PopulSize: 60
 UnenhRuns: 4300
 Elitism: on
 -results:
 -scanLayout: R1.R6
 -nNodes: 433

Action done.

SET: Circuit schema window



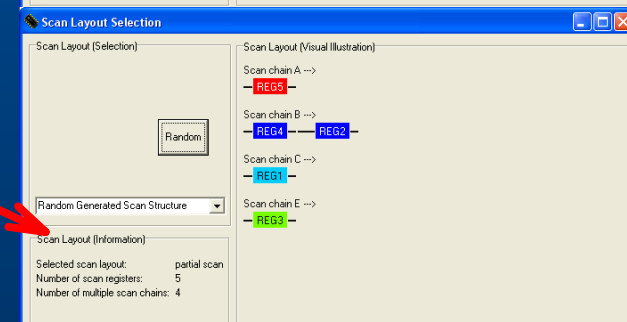
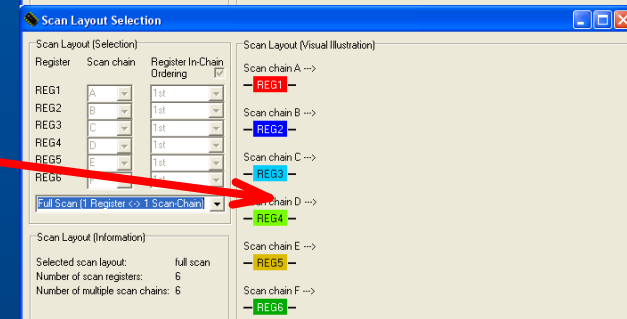
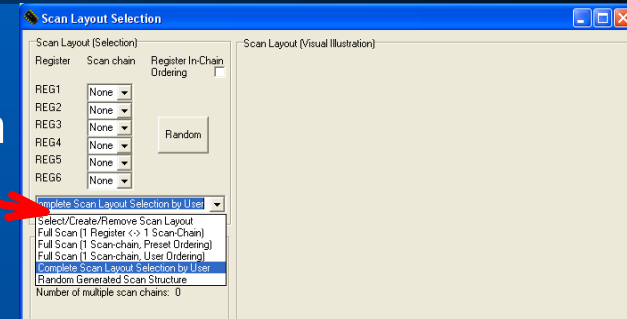
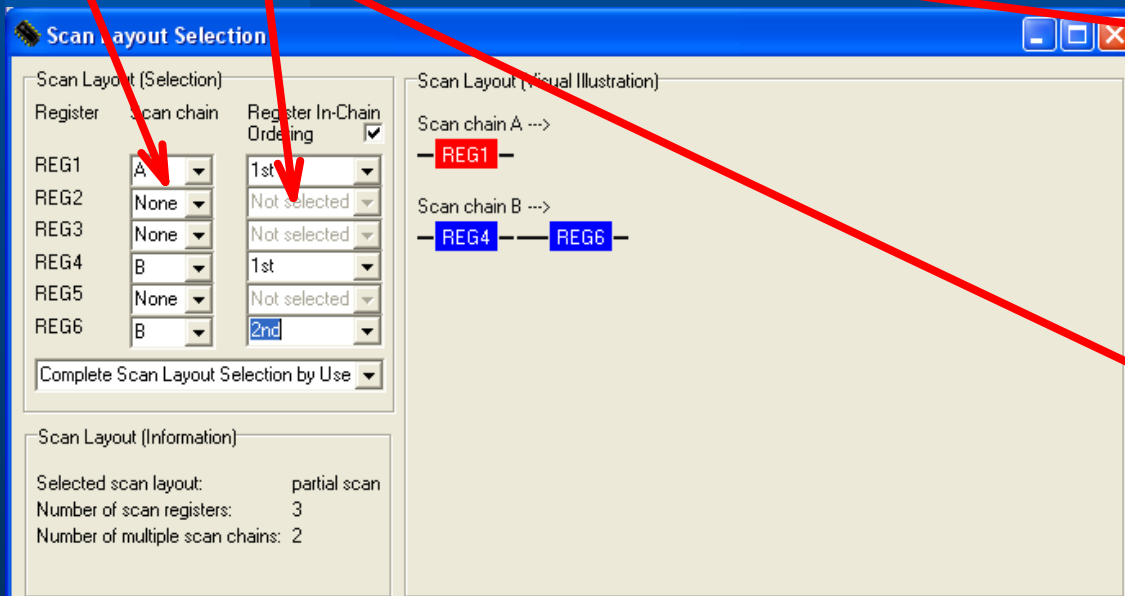
- Educational circuit schema
- Test data propagation paths browser for selected functional unit



- Go to the path-start (primary inputs)
- Perform 1 step towards path-end (primary outputs)
- Write information about test data propagation paths of selected functional unit into History window
- Selected functional unit
- All functional units

SET: Scan-layout window

- User/predefined/random (U/P/R) scan-layout selection
- U/P/R inclusion of registers into scan chains
- U/P/R ordering registers within scan chains
- Selected scan-layout
- Vizualization Information



SET: Testability results window

- Port information

- Name and bit-width
- Controllability
- Observability
- Testability

- Summary information

- “Nodes”

- #/ratio of controllable nodes
- #/ratio of observable nodes
- #/ratio of testable nodes

- “Circuit”

- Controllability
- Observability
- Testability

| Component | C | O | T |
|---------------------|--------------|--------------|-----------|
| mux1_b(8) | 1.000000 | 0.000000 | 0.000000 |
| mux1_a(8) | 0.667837 | 0.000000 | 0.000000 |
| reg5_q(8) | 0.000000 | 0.967480 | 0.000000 |
| reg6_d(8) | 0.000000 | 0.825784 | 0.000000 |
| reg5_q(8) | 0.815614 | 0.000000 | 0.000000 |
| reg7_a(8) | 0.955582 | 0.000000 | 0.000000 |
| reg4_q(8) | 0.667837 | 0.000000 | 0.000000 |
| reg4_d(8) | 0.782445 | 0.000000 | 0.000000 |
| reg3_q(8) | 0.000000 | 0.818957 | 0.000000 |
| reg3_d(8) | 0.000000 | 0.679572 | 0.000000 |
| reg2_q(8) | 0.000000 | 0.818957 | 0.000000 |
| reg2_d(8) | 0.000000 | 0.679572 | 0.000000 |
| reg1_q(8) | 0.000000 | 0.000000 | 0.000000 |
| reg1_d(8) | 0.000000 | 0.000000 | 0.000000 |
| comp_q(8) | 0.000000 | 1.000000 | 0.000000 |
| comp_b(8) | 0.000000 | 0.967480 | 0.000000 |
| comp_a(8) | 1.000000 | 0.000000 | 0.000000 |
| mul2_q(8) | 0.955582 | 0.000000 | 0.000000 |
| mul2_b(8) | 0.991870 | 0.000000 | 0.000000 |
| mul2_a(8) | 0.995935 | 0.000000 | 0.000000 |
| mul1_q(8) | 0.782445 | 0.000000 | 0.000000 |
| mul1_b(8) | 0.812158 | 0.000000 | 0.000000 |
| mul1_a(8) | 0.995935 | 0.000000 | 0.000000 |
| sub_q(8) | 0.000000 | 0.000000 | 0.000000 |
| sub_b(8) | 0.000000 | 0.000000 | 0.000000 |
| sub_a(8) | 0.667837 | 0.000000 | 0.000000 |
| add_q(8) | 0.000000 | 0.825784 | 0.000000 |
| add_b(8) | 0.995935 | 0.000000 | 0.000000 |
| add_a(8) | 0.000000 | 0.822427 | 0.000000 |
| diffeq_result(8) | 0.000000 | 1.000000 | 0.000000 |
| diffeq_const2(8) | 1.000000 | 0.000000 | 0.000000 |
| diffeq_dx(8) | 1.000000 | 0.000000 | 0.000000 |
| diffeq_a(8) | 1.000000 | 0.000000 | 0.000000 |
| Summary (Total=422) | C(246/58.3%) | O(112/26.5%) | T(0/0.0%) |
| GlobalMeasures | 0.440177 | 0.164195 | 0.072275 |

Scan-layout selection example

The image shows a sequence of operations in the Scan Educational Tool. The main window on the left contains a 'Main Menu' with 'Scan Layout Auto-Designer' selected. The 'Scan Layout Selection' dialog box is shown in multiple instances, illustrating the process of selecting a scan layout. The 'Scan Layout (Selection)' table in the dialog boxes shows the following configuration:

| Register | Scan chain | Register In-Chain Ordering |
|----------|------------|----------------------------|
| REG1 | A | 1st |
| REG2 | B | 1st |
| REG3 | C | 1st |
| REG4 | D | 1st |
| REG5 | E | 1st |
| REG6 | F | 1st |

The 'Scan Layout (Visual Illustration)' shows the scan chain A with registers REG1 through REG6 connected in sequence. The 'Scan Layout (Information)' section shows: Selected scan layout: full scan, Number of scan registers: 6, Number of multiple scan chains: 6.

The 'Info' dialog box contains the message: "Please, select ordering of registers in scan-chain, if required. Otherwise, random ordering of registers in scan-chain will be used." with an 'OK' button.

The 'Scan Layout Selection' dialog box also shows a list of options: Select/Create/Remove Scan Layout, Full Scan (1 Register <-> 1 Scan-Chain), Full Scan (1 Scan-chain, Preset Ordering), Full Scan (1 Scan-chain, User Ordering), Complete Scan Layout Selection by User, and Random Generated Scan Structure. The 'Number of multiple scan chains' is 0.

The 'Scan Layout Selection' dialog box also shows a 'Random' button and an 'Info' dialog box with the message: "To generate a scan structure, please press button." with an 'OK' button.

The 'Scan Layout Selection' dialog box also shows a 'Random' button and an 'Info' dialog box with the message: "Please, select registers to scan and their ordering in scan-chain(s), if required. Otherwise, random ordering of selected registers in scan-chain(s) will be used." with an 'OK' button.

The 'Scan Layout Selection' dialog box also shows a 'Random' button and an 'Info' dialog box with the message: "Please, select registers to scan and their ordering in scan-chain(s), if required. Otherwise, random ordering of selected registers in scan-chain(s) will be used." with an 'OK' button.

The 'Scan Layout Selection' dialog box also shows a 'Random' button and an 'Info' dialog box with the message: "Please, select registers to scan and their ordering in scan-chain(s), if required. Otherwise, random ordering of selected registers in scan-chain(s) will be used." with an 'OK' button.

Testability analysis example

Scan Educational Tool

Main Menu

Action

Testability Analysis
 Scan Layout Auto-Designer
 Run

Circuit Info

Number of nodes: 431 Testability: 0.520
 Controllable nodes (%): 100.0
 Observable nodes (%): 94.2 Bad Perfect
 Area overhead (%): 6.623
 I/O overhead (%): 7.317 Fitness: 0.000

View

Circuit Schema Unbroken
 Scan Layout Prefs Feedback
 Testability Details Loops

Design Constraints

Max Area Overhead: %
 Max I/O Overhead: %

Genetic Algorithm Parameters

Elitism Population Size:
 Maximum Number of Runs Without Increasing of Average Fitness in Population:

Testability Details

| | | | |
|---------------------|---------------|--------------|--------------|
| mux2.b(8) | 0.906971 | 0.109978 | 0.500475 |
| mux2.a(8) | 0.614095 | 0.109978 | 0.362037 |
| mux1.q(8) | 0.956000 | 0.103534 | 0.549767 |
| mux1.b(8) | 1.000000 | 0.103111 | 0.551525 |
| mux1.a(8) | 0.825776 | 0.103111 | 0.444444 |
| reg6.SCAN_OUT(1) | 0.234286 | 1.000000 | 0.617143 |
| reg6.SCAN_IN(1) | 0.614095 | 0.614095 | 0.614095 |
| reg6.q(8) | 0.234286 | 0.968000 | 0.601143 |
| reg6.d(8) | 0.179212 | 0.914262 | 0.546749 |
| reg5.q(8) | 0.906971 | 0.110120 | 0.508496 |
| reg5.d(8) | 0.956288 | 0.111132 | 0.528710 |
| reg4.q(8) | 0.825776 | 0.43874 | 0.484825 |
| reg4.d(8) | 0.870677 | 1.114631 | 0.492654 |
| reg3.q(8) | 0.186667 | 0.609173 | 0.397920 |
| reg3.d(8) | 0.234286 | 0.560037 | 0.397161 |
| reg2.q(8) | 0.186667 | 0.609173 | 0.397920 |
| reg2.d(8) | 0.234286 | 0.560037 | 0.397161 |
| reg1.SCAN_OUT(1) | 0.614095 | 0.614095 | 0.614095 |
| reg1.SCAN_IN(1) | 1.000000 | 0.234286 | 0.617143 |
| reg1.q(8) | 0.614095 | 0.187176 | 0.400636 |
| reg1.d(8) | 0.490202 | 0.234286 | 0.362244 |
| comp.q(8) | 0.226789 | 1.000000 | 0.613394 |
| comp.b(8) | 0.234286 | 0.968000 | 0.601143 |
| comp.a(8) | 1.000000 | 0.234286 | 0.617143 |
| mul2.q(8) | 0.956288 | 0.101132 | 0.528710 |
| mul2.b(8) | 0.992000 | 0.07342 | 0.544671 |
| mul2.a(8) | 0.996000 | 0.161323 | 0.548162 |
| mul1.q(8) | 0.870677 | 0.114631 | 0.492654 |
| mul1.b(8) | 0.903192 | 0.110445 | 0.506818 |
| mul1.a(8) | 0.996000 | 0.103534 | 0.549767 |
| sub.q(8) | 0.490202 | 0.234286 | 0.362244 |
| sub.b(8) | 0.614095 | 0.187176 | 0.400636 |
| sub.a(8) | 0.825776 | 0.143874 | 0.463925 |
| add.d(8) | 0.179212 | 0.614095 | 0.396654 |
| add.b(8) | 0.996000 | 0.110481 | 0.553240 |
| add.a(8) | 0.185905 | 0.611639 | 0.398772 |
| diffreq.SCAN_OUT(0) | 0.234286 | 1.000000 | 0.617143 |
| diffreq.SCAN_IN(0) | 1.000000 | 0.234286 | 0.617143 |
| diffreq.result(8) | 0.226789 | 1.000000 | 0.613394 |
| diffreq.const2(8) | 1.000000 | 0.099902 | 0.549951 |
| diffreq.dx(8) | 1.000000 | 0.110020 | 0.555010 |
| diffreq.a(8) | 1.000000 | 0.234286 | 0.617143 |
| #Nodes(Total=431) | C(431/100.0%) | O(406/94.2%) | T(406/94.2%) |
| GlobalMeasures | 0.830295 | 0.626357 | 0.520061 |

Scan Educational Tool

Main Menu

Action

Testability Analysis
 Scan Layout Auto-Designer
 Run

Circuit Info

Number of nodes: 431 Testability: 0.520
 Controllable nodes (%): 100.0
 Observable nodes (%): 94.2 Bad Perfect
 Area overhead (%): 6.623
 I/O overhead (%): 7.317 Fitness: 0.000

View

Circuit Schema Unbroken
 Scan Layout Prefs Feedback
 Testability Details Loops

Design Constraints

Max Area Overhead: %
 Max I/O Overhead: %

Genetic Algorithm Parameters

Elitism Population Size:
 Maximum Number of Runs Without Increasing of Average Fitness in Population:

Performing testability analysis

-start time: 13:37:52
 -results:
 -scanLayout: R1R6 Save | Clear
 -nNodes: 431
 -areaOver: 6.623 %
 -pinOver: 7.317 %
 -testability: 0.520000
 -obsRatio: 94.2 %
 -conRatio: 100.0 %
 -end time: 13:37:52

Action done.

Scan Layout Selection

Scan Layout (Selection)

Scan Layout Selection

Scan Layout (Selection)

Register Scan chain Register In-Chain Ordering

REG1 A

REG2 None

REG3 None

REG4 None

REG5 None

REG6 A

Scan Layout (Information)

Selected scan layout: partial scan
 Number of scan registers: 2
 Number of multiple scan chains: 1

Scan Layout (Visual Illustration)

Scan chain A -->

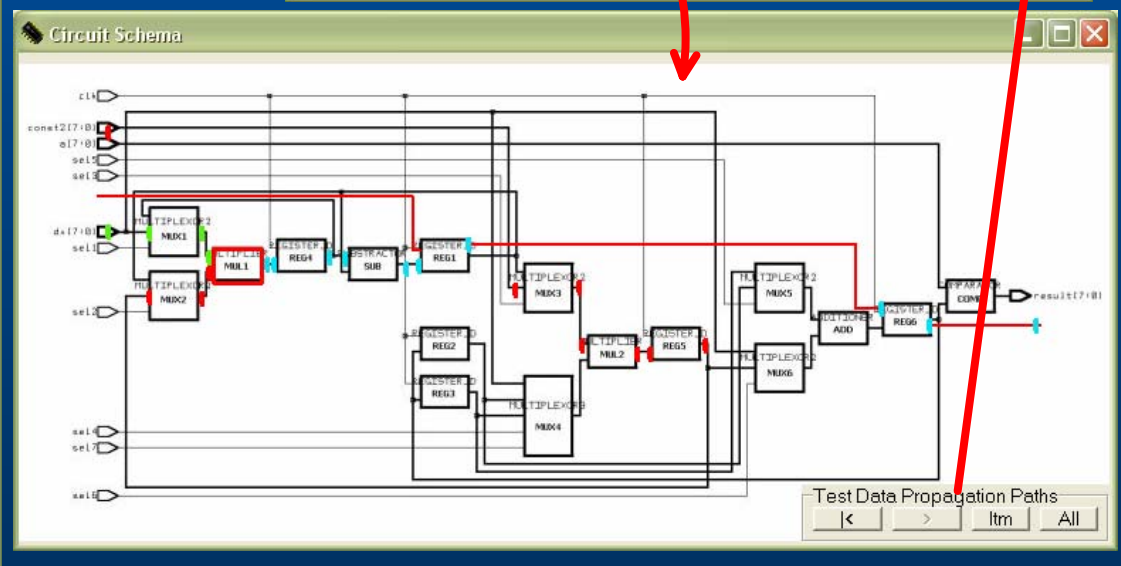
REG1 REG6

Genetic Algorithm Parameters

Elitism Population Size:
 Maximum Number of Runs Without Increasing of Average Fitness in Population:

Step 16] reg1.d
 Response observation
 Step 17] reg1.SCAN_OUT
 Response observation
 Step 18] reg6.SCAN_IN Save | Clear
 Response observation
 Step 19] reg6.SCAN_OUT
 Response observation
 Step 20] diffreq.SCAN_OUT
 Response observation
 Step 20] diffreq.SCAN_OUT
 End of data for 'mul1'.

Action done.



Loop-breaking example (1)

The screenshot shows the Scan Educational Tool interface. On the left, the 'Main Menu' has 'Testability Analysis' selected, with a red circle around it and an arrow pointing to the 'Run' button. Below this, 'Circuit Info' shows: Number of nodes: 422, Testability: 0.075, Controllable nodes (%): 58.3, Observable nodes (%): 26.5, Area overhead (%): 0.000, I/O overhead (%): 0.000, Fitness: 0.00. In the 'View' section, 'Unbroken', 'Feedback', and 'Loops' are checked, with a red circle around this section. The 'Design Constrains' section shows Max Area Overhead: 7% and Max I/O Overhead: 13%. The 'Genetic Algorithm Parameters' section shows Elitism checked, Population Size: 20, and Maximum Number of Runs Without Increasing of Average Fitness in Population: 500. A log window at the bottom shows testability analysis results: -scanLayout: none, -nNodes: Save | Clear, -areaOver: 0.000%, -pinOver: 0.000%, -testability: 0.075000, -obsRatio: 26.5%, -conRatio: 58.3%, -end time: 11:29:45. The main window, 'Circuit Schema', shows a circuit diagram with three loops highlighted in purple, pink, and blue, labeled 'loop_reg1', 'loop_reg2', and 'loop_reg3'. A red arrow points from the 'Run' button to the circuit diagram. The 'Scan Layout Selection' window at the bottom shows 'Scan Layout (Selection)' and 'Scan Layout (Visual Illustration)' both empty, and 'Scan Layout (Information)' showing Selected scan layout: none, Number of scan registers: 0, and Number of multiple scan chains: 0.

When no scan-layout is selected then all 3 loops remain unbroken

Loop-breaking example (2)

The screenshot shows the Scan Educational Tool interface. The 'Testability Analysis' window displays the following statistics:

| | | | |
|-------------------------|-------|--------------|---------|
| Number of nodes: | 428 | Testability: | 0.315 |
| Controllable nodes (%): | 70.1 | Bad: | Perfect |
| Observable nodes (%): | 83.2 | Fitness: | 0.0 |
| Area overhead (%): | 3.311 | | |
| I/O overhead (%): | 7.317 | | |

The 'Scan Layout Selection' window shows the following configuration:

| Register | Scan chain | Register In-Chain Ordering |
|----------|------------|----------------------------|
| REG1 | A | |
| REG2 | None | |
| REG3 | None | |
| REG4 | None | |
| REG5 | None | |
| REG6 | None | |

The 'Scan Layout (Visual Illustration)' window shows 'Scan chain A' containing 'REG1'.

When REG1 is included into scan chain A then

2 of 3 loops remain unbroken

Loop-breaking example (3)

The screenshot shows the Scan Educational Tool interface. On the left, the 'Main Menu' has 'Testability Analysis' and 'Run' circled in red. The 'Circuit Schema' window shows a circuit diagram with 'REG1' and 'REG6' circled in red. An 'Info' dialog box states: 'According to selected scan layout, there are no uncontrollable loops in the circuit.' The 'Scan Layout Selection' window shows a table for assigning registers to scan chains:

| Register | Scan chain | Register In-Chain Ordering |
|----------|------------|----------------------------|
| REG1 | A | |
| REG2 | None | |
| REG3 | None | |
| REG4 | None | |
| REG5 | None | |
| REG6 | B | |

Below the table, it indicates 'Selected scan layout: partial scan', 'Number of scan registers: 2', and 'Number of multiple scan chains: 2'. The 'Performing testability analysis' log at the bottom shows results: 'scanLayout: R1..R6', 'nNodes: 433', 'areaOver: 6.623%', 'pinOver: 12.195%', 'testability: 0.593000', 'obsRatio: 94.2%', and 'conRatio: 100.0%'.

When REG1, REG6 are included into scan chains

then

all 3 loops become broken

Test data propagation paths browser

Main Menu

- Action
 - Stability Analysis
 - Scan Layout Auto-Designer

Circuit Info

- Number of nodes: 453 Testability: 0.819
- Controllable nodes (%): 100.0
- Observable nodes (%): 93.6
- Area overhead (%): 19.868
- I/O overhead (%): 31.707
- Fitness: 0.000

View

- Circuit Schema Unbroken
- Scan Layout Prefs Feedback
- Testability Details Loops

Design Constraints

- Max Area Overhead: 7 %
- Max I/O Overhead: 10 %

Genetic Algorithm Parameters

- Elitism Population Size: 20
- Maximum Number of Runs Without Increasing of Average Fitness in Population: 500

Test Data Propagation Paths

Navigation: |< >| Itm All

Circuit Schema

REG1
REG2
REG3
REG4
REG5
REG6

Scan chain F --> REG6

Automated DFT process example (1)

The screenshot displays the Scan Educational Tool interface, which is used for automated Design for Testability (DFT) process. The interface is divided into several windows:

- Main Menu:** Contains a 'Run' button and a 'Scan Layout Auto-Designer' option, which is circled in red. A red arrow points from this option to the 'Scan Layout Selection' window. Below the 'Run' button, there are various parameters and options, including 'Circuit Info', 'View', 'Design Constraints', and 'Genetic Algorithm Parameters'. A red arrow points from the 'Run' button to the 'Scan Layout Selection' window.
- Scan Layout Selection:** Shows the selected scan layout (Selection) and a visual illustration of the scan chains. It displays 'Scan chain A' with 'REG1' and 'Scan chain B' with 'REG3'. A red arrow points from the 'Run' button to this window.
- Circuit Schema:** Displays a logic diagram of the circuit. The diagram includes various components such as Multiplexers (MUX1-MUX6), Registers (REG1-REG6), Adders, Multipliers, and Subtractors. A red line highlights the scan chain A path, and a blue line highlights the scan chain B path. A red arrow points from the 'Run' button to the 'Scan Layout Selection' window, and another red arrow points from the 'Scan Layout Selection' window to the 'Circuit Schema' window.

At the bottom of the interface, there is a 'Test Data Propagation Paths' section with navigation buttons: '<', '>', 'ftrn', and a button with a circuit symbol.

Automated DFT process example (2)

The screenshot displays a software interface for automated Design-for-Test (DFT) optimization. It is divided into several panels:

- Design Constraints:**
 - Max Area Overhead: 10 %
 - Max I/O Overhead: 8 %
- Genetic Algorithm Parameters:**
 - Elitism
 - Population Size: 20
 - Maximum Number of Runs Without Increasing of Average Fitness in Population: 500
- Results Panel:**
 - PinOver: 8 %
 - PopulSize: 20
 - UprnhRuns: 500
 - Elitism: on
 - results: Save Clear
 - scanLayout: RbRtF3
 - nNodes: 434
 - areaOver: 9.934 %
 - pinOver: 7.317 %
 - fitness: 59.653000
 - testability: 0.658000
 - obsRatio: 94.0 %
 - conRatio: 100.0 %
- Scan Layout (Visual Illustration):**
 - Scan chain A -->
 - REG6 — REG1 — REG3
- Circuit Diagram:**
 - A detailed logic diagram showing various components like MULTIPLEXOR, REGISTER, and ADD.
 - Registers REG1, REG2, REG3, REG4, REG5, and REG6 are highlighted with red boxes.
 - Red lines trace the scan chain paths through the circuit, connecting the registers in the order specified in the Scan Layout panel.
- Test Data Propagation Paths:**
 - Navigation buttons: < > ltm

Automated DFT process example (3)

Design Constraints

Max Area Overhead: %

Max I/O Overhead: %

Genetic Algorithm Parameters

Elitism Population Size:

Maximum Number of Runs Without Increasing of Average Fitness in Population:

UnenhRuns: 500
 Elitism: on

-results:

R6R3.R1R4R2

-scanLayout:

-nNodes: 442

-areaOver: 16.556 %

-pinOver: 12.195 %

-fitness: 71.483000

-testability: 0.732000

-obsRatio: 93.7 %

-conRatio: 100.0 %

-end time: 9:58:24

Action done.

Scan Layout (Visual Illustration)

Scan chain A --->

— REG6 — REG3 —

Scan chain B --->

— REG1 — REG4 — REG2 —

Circuit Schema

Test Data Propagation Paths

< > ltm CAID

Automated DFT process example (4)

The screenshot displays the final stage of an automated Design-for-Test (DFT) process. It is divided into several key sections:

- Design Constraints:** Shows a maximum area overhead of 20% and a maximum I/O overhead of 27%.
- Genetic Algorithm Parameters:** Includes 'Elitism' checked, a population size of 100, and a maximum of 500 runs without increasing average fitness.
- Results:** A red arrow points from the 'UnenhRuns: 500' field to the '-results:' section, which lists:
 - scanLayout: R3.R1R5.R6.R4.R2
 - nNodes: 451
 - areaOver: 19.868 %
 - pinOver: 26.829 %
 - fitness: 76.894000
 - testability: 0.806000
 - obsRatio: 93.6 %
 - conRatio: 100.0 %
 - end time: 10:03:03
- Scan Layout (Visual Illustration):** Lists six scan chains:
 - Scan chain A: REG3
 - Scan chain B: REG1, REG5
 - Scan chain C: REG6
 - Scan chain E: REG4
 - Scan chain F: REG2
- Circuit Schema:** A detailed logic diagram showing the circuit's internal structure. It includes multiplexers (MUX1-MUX6), registers (REG1-REG6), and logic blocks like 'MULTIPLY', 'SUBTRACT', 'ADD', and 'COMPARE'. Colored lines (red, green, blue, yellow) trace the scan paths through the circuit, corresponding to the scan chains listed in the adjacent window.

At the bottom left, the text 'Action done.' is visible.

SET requirements and limitations

Actual SET:

- **Requirements**

- WIN32 environment to run SET in graphical user interface mode
- Original/emulated MS-DOS environment to run SET in command line mode

- **Limitations**

- Non-commercial use only (see SET licencing conditions)
- Set of educational circuits is limited to one circuit in the set

Conclusions

Acknowledgements: The research and software development has been financially supported by the

- Grant Agency of the Czech Republic under contracts
 - No. 102/04/0737 “Modern methods of digital systems design”,
 - No. GA102/05/P193 “Optimizing Methods in Digital Systems Diagnosis”
- IST-2000-30193 project REASON.