ECE 6140: Digital Systems Testing

Project Report by:

Ajinkya Munge,

GTID: 903321612,

amunge3@gatech.edu

Report

Algorithm:

1. Use 5 valued Logical Simulator as imply function.
2. Backtrace function acts as reverse path check from target pin to primary input.
3. Backtrace also gives Primary input(PI) value.
4. Objective function gives pin number to target and desired value at the targeted pin.
5. Run Imply everytime PI is assigned.
6. If the error at target pin is propagated to primary output then PODEM has succeeded.
7. Display the PODEM result and Input Vector calculated by PODEM.

Pseudo Code:

Initialize all lists required for PODEM

Open circuit file:

Set target\_pin and stuck at value

Logical\_simulator()

PODEM()

Update all the lists for PODEM output, and Input vectors.

PODEM():

Imply(pin\_no,pin\_val) #Logical simulator with PI pin\_no at pin\_val

d\_list = create\_d\_front() #Create D frontier

If ‘error at Primary output’:

return SUCCESS

req\_pin,pin\_val = objective(target\_pin,stuck\_at)  
pin\_no,pin\_val = backtrace(req\_pin,pin\_val)

if ‘no pins found by backtrace’=’no tests’

return FAILURE

Imply(pin\_no,pin\_val) #Logical simulator with PI pin\_no at pin\_val

d\_list = create\_d\_front() #Create D frontier

If PODEM() = SUCCESS

return SUCCESS

pin\_val = not(pin\_val) # reverse the value of PI

Imply(pin\_no,pin\_val) #Logical simulator with PI pin\_no at pin\_val

d\_list = create\_d\_front() #Create D frontier

If PODEM() = SUCCESS

return = SUCCESS

Imply(pin\_no,’x’) #backtrack

d\_list = create\_d\_front() #Create D frontier

return FAILURE #no tests possible

objective(target\_pin,stuck\_at):

if target\_pin is at x:

return target\_pin,not(pin\_val)

if dlist is not zero:

pick a gate from the dlist

select an input ‘j’ of gate with value ‘x’

c = controlling value of gate

return j,not (c)

return ‘no d list’,1

backtrace(target\_pin,pin\_val):

if target\_pin is ‘no d list’

return ‘no x path’, ‘no x path’

find gate whose output is target\_pin

if both inputs of gate are tried primary inputs:

delete this path and go back

if first input is a tried primary input:

I = inversion parity

pin\_val = xor(pin\_val,i)

backtrace(second input pin,pin\_val)

if only one input is ‘x’:

add input to x path

if both inputs are ‘x’:

add both inputs to x path

if input is not tried primary input and ‘x’:

I = inversion parity

pin\_val = xor(pin\_val,i)

return input,pin\_val

if input is ‘x’

return backtrace(input pin, pin\_val)

return ‘no x path‘

create\_d\_front(pin\_no,d\_list):

find gate for which pin\_no is the output.

If one the input is ‘D’ or ‘Dbar’ and output is x:

Update d\_list #Add gate to D frontier

return d\_list

return d\_list

Observation:

|  |  |  |  |
| --- | --- | --- | --- |
| Test Circuit | Faults | PODEM Result | Test Vectors |
| s27 | Net 16 s-a-0 | SUCCESS | 00xx011 |
| Net 10 s-a-1 | SUCCESS | 1000xx0 |
| Net 12 s–a-0 | SUCCESS | 1xxx1xx |
| Net 18 s-a-1 | SUCCESS | 10x1011 |
| s298f\_2 | Net 70 s-a-1 | SUCCESS | 01x1xxxxxxxxxx0xx |
| Net 73 s-a-0 | SUCCESS | 111xxxxxxxxxxx0xx |
| Net 26 s-a-1 | SUCCESS | xx110xxxxxxxxxxxx |
| Net 92 s-a-0 | SUCCESS | x10101xxxxxx0x0xx |
| s344f\_2 | Net 166 s-a-0 | SUCCESS | 11x0100xxx011010xxxxxxxx |
| Net 71 s-a-1 | SUCCESS | 10xxxxxxxxxxxxxxxxxxxxxx |
| Net 16 s-a-0 | SUCCESS | 1101110xxx1xxxx1xxxxxxxx |
| Net 91 s-a-1 | SUCCESS | 111xxxxxxxxxxxxxxxxxxxxx |
| s349f\_2 | Net 25 s-a-1 | SUCCESS | xxxxxxxxxxxxxxx1xxxxxxxx |
| Net 51 s-a-0 | SUCCESS | 010xxxxxxxxxxxx0xxxxxxxx |
| Net 105 s-a-1 | SUCCESS | 0011000xxx01xx10xxxxxxxx |
| Net 7 s-a-0 | SUCCESS | xxxxxx1xxxxxxxxxxxxxxxxx |

Flow Diagram:

PODEM

Open test circuit file in read mode

Imply(pin\_no,pin\_val)

Imply(pin\_no,pin\_val)

Backtrace

Objective

SUCCESS

YES

Error at PO

Target\_pin = ‘x’

YES

Create D Frontier

target\_pin,not(pin\_val)

Non zero d frontier

YES

Pick a gate from D frontier

No D frontier

Select ‘x’ value input j

c=controlling value of gate

Create D Frontier

j, not(c)

PODEM = SUCCESS

YES

Target\_pin = ‘no d\_list’

YES

SUCCESS

No x path

Imply(pin\_no,not(pin\_val))

Input,

pin\_value

YES

YES

Input is not tried PI and ‘x’

Add both pins to x path

Both pins are ‘x’

YES

Only one input is ‘x’

Add input to x path

Backtrace(second pin,pin\_value)

YES

first pin is tried PI

YES

Both pins are tried PI

Both pins are tried PI

Delete this path and go back

Delete this path and go back

YES

Find gate whose output is target\_pin

SUCCESS

YES

PODEM = SUCCESS

FAILURE

Imply(pin\_no,’x’)

Create D Frontier

Create D Frontier

Return d\_list

Return d\_list

Update d\_list

YES

Input is ‘D’ or ‘Dbar’ and output is ‘x’

Find gate whose pin no is the output

‘no x path’

Backtrace(input pin, pin\_val)

YES

Input is ‘x’