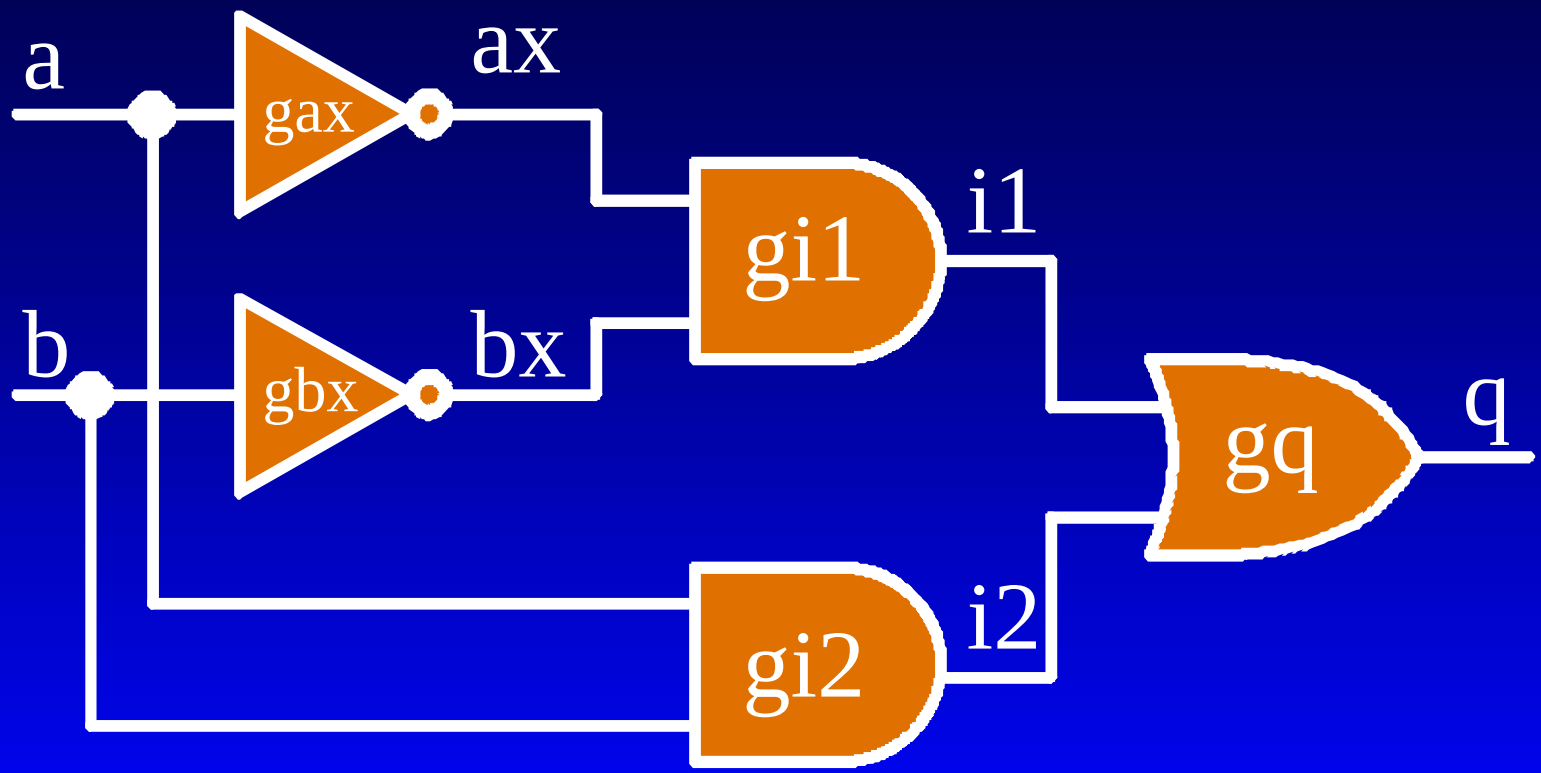

Introduction to FHDL

A Sample Circuit



Equivalent FHDL

```
Test1:  circuit
        inputs  a,b
        outputs q
gq:     or      (i1,i2),q
gi2     and     (a,b),i2
gi1:    and     (ax,bx),i1
gax:    not     a,ax
gbx:    not     b,bx
        endcircuit
```

Coding Rules

- Statements have the form:
  <NAME> : <OP-CODE> <OPERANDS>
  - <OP-CODE> must be surrounded by spaces
  - <NAME>: is sometimes optional
- Circuits must begin with:
  <NAME>: circuit
- Circuits must end with:
  endcircuit

Coding Rules II

- Primary Inputs must be declared using the statement:

  inputs <NAME 1>,<NAME 2>, ...

- Primary Outputs must be declared using the statement:

  outputs <NAME 1>,<NAME 2>, ...

Coding Rules III

- The Remainder of the Circuit is Gate Specifications of the form:
  <Name>: <GateType> (<Inputs>),(<Outputs>)
- If <Inputs> or <Outputs> is a single name, Omit the Parenthesis
- Nets are declared implicitly
- The “wire” statement can alter net properties

Circuit Must Be Parsed

- Internal Tables are Constructed from Specifications
- Circuit-Processing programs use Parsed form of Circuit
- Many Different Structuring Techniques can be Used

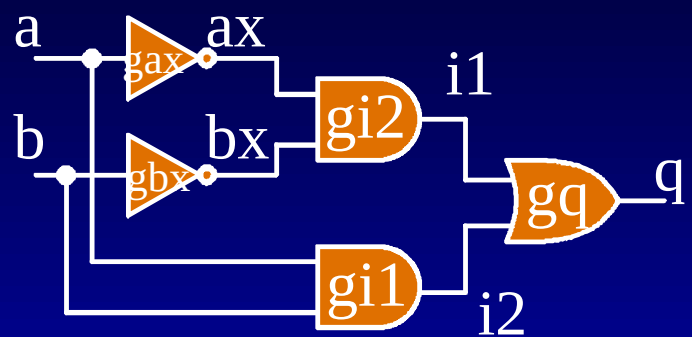
Possible Result of Parsing Gates

Gate Table				
Index	*Name*	*Type*	*Inputs*	*Outputs*
0	gq	or	3,4	2
1	gi2	and	0,1	4
2	gi1	and	5,6	3
3	gaprime	not	0	5
4	gbprime	not	1	6

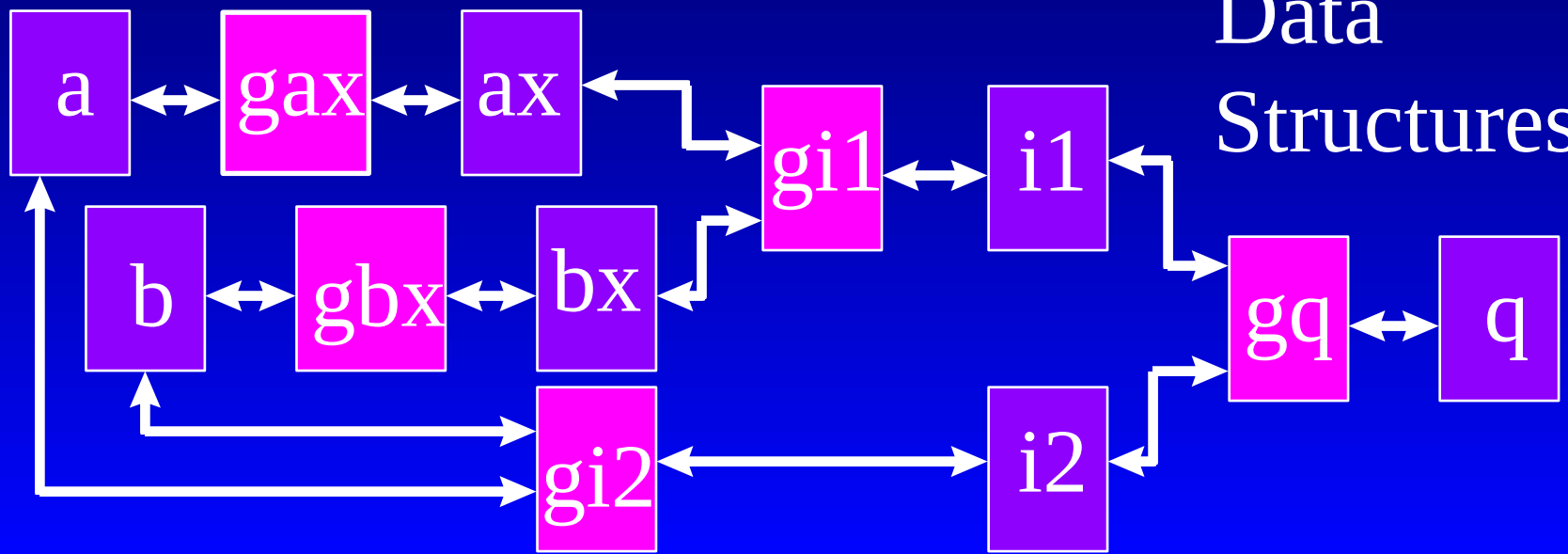
Possible Result of Parsing Nets

Net Table				
Index	*Name*	*Type*	*Fanout*	*Value*
0	a	PI	1,3	0
1	b	PI	1,4	0
2	q	PO		0
3	i1		0	0
4	i2		0	0
5	aprime		2	0
6	bprime		2	0

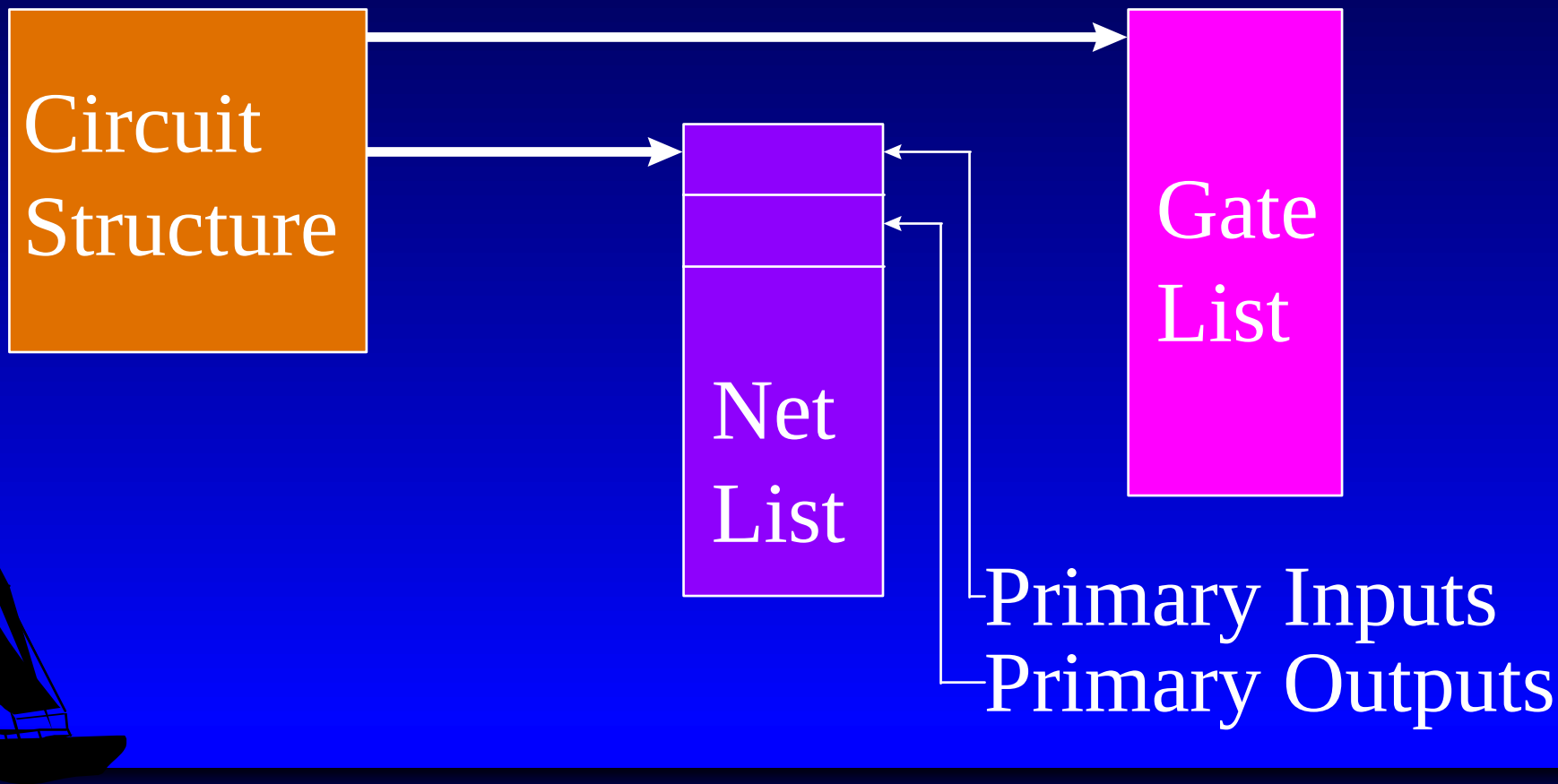
WINFHDL Style



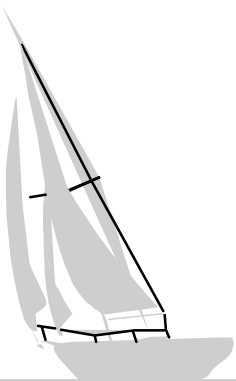
Linked Data Structures



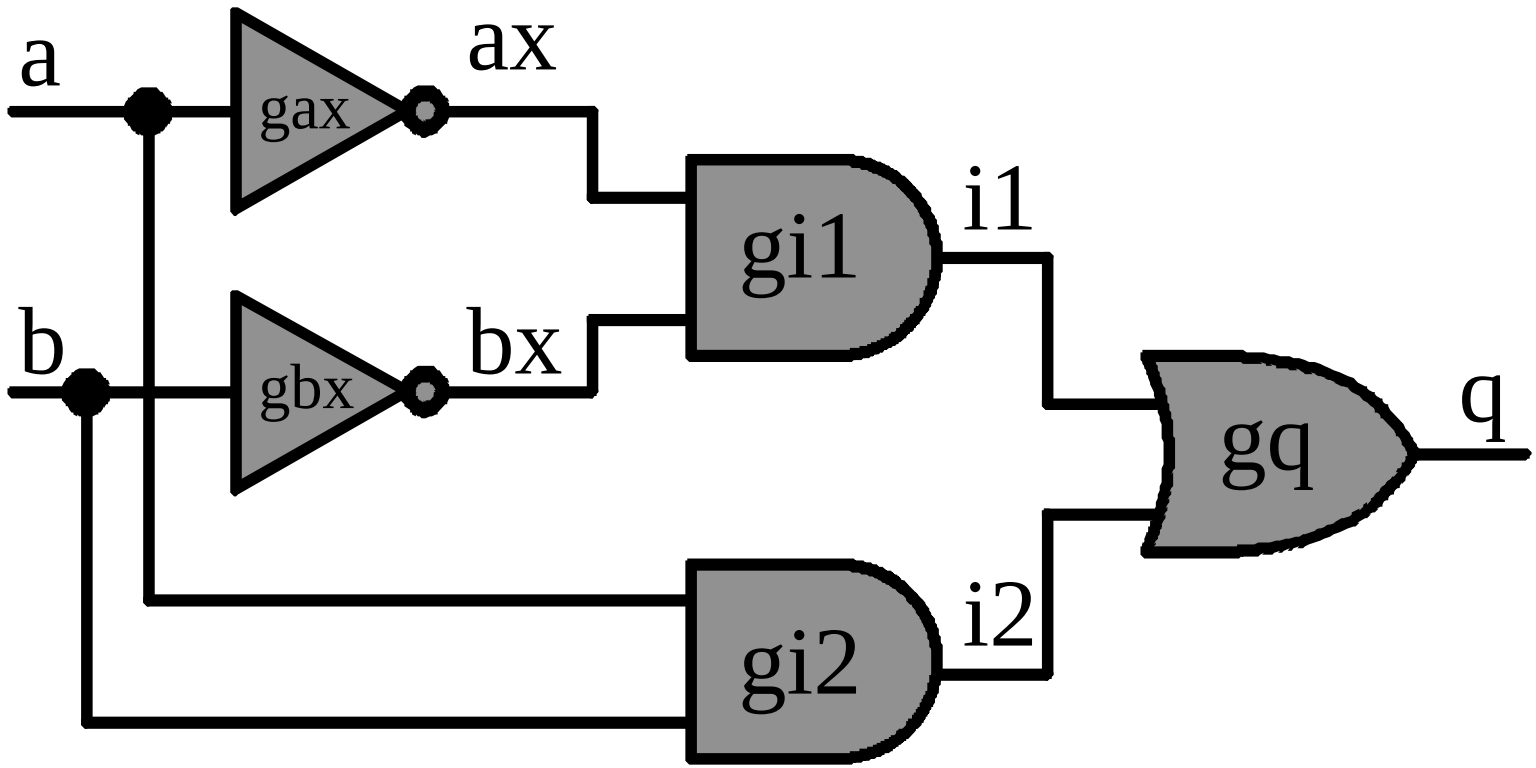
Access to Structures



Introduction to FHDL

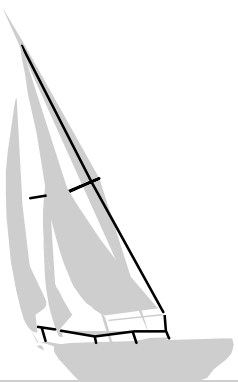


A Sample Circuit



Equivalent FHDL

```
Test1: circuit
       inputs a,b
       outputs q
gq:    or     (i1,i2),q
gi2    and    (a,b),i2
gi1:   and    (ax,bx),i1
gax:   not    a,ax
gbx:   not    b,bx
       endcircuit
```



Coding Rules

- Statements have the form:
  <NAME> : <OP-CODE> <OPERANDS>
  - <OP-CODE> must be surrounded by spaces
  - <NAME>: is sometimes optional
- Circuits must begin with:
  <NAME>: circuit
- Circuits must end with:
  endcircuit

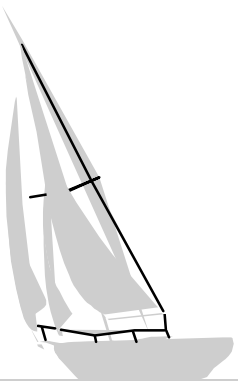
Coding Rules II

- Primary Inputs must be declared using the statement:

  inputs <NAME 1>,<NAME 2>, ...

- Primary Outputs must be declared using the statement:

  outputs <NAME 1>,<NAME 2>, ...

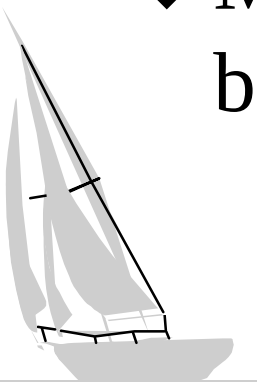


Coding Rules III

- The Remainder of the Circuit is Gate Specifications of the form:
  <Name>: <GateType> (<Inputs>),(<Outputs>)
- If <Inputs> or <Outputs> is a single name, Omit the Parenthesis
- Nets are declared implicitly
- The “wire” statement can alter net properties

Circuit Must Be Parsed

- Internal Tables are Constructed from Specifications
- Circuit-Processing programs use Parsed form of Circuit
- Many Different Structuring Techniques can be Used



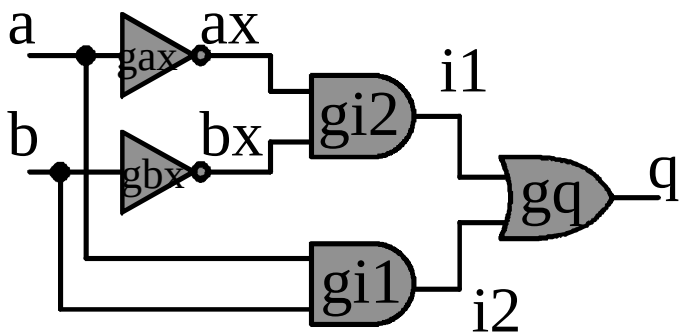
Possible Result of Parsing Gates

Gate Table				
Index	***Name***	***Type***	***Inputs***	***Outputs***
0	**gq**	**or**	**3,4**	**2**
1	**gi2**	**and**	**0,1**	**4**
2	**gi1**	**and**	**5,6**	**3**
3	**gaprime**	**not**	**0**	**5**
4	**gbprime**	**not**	**1**	**6**

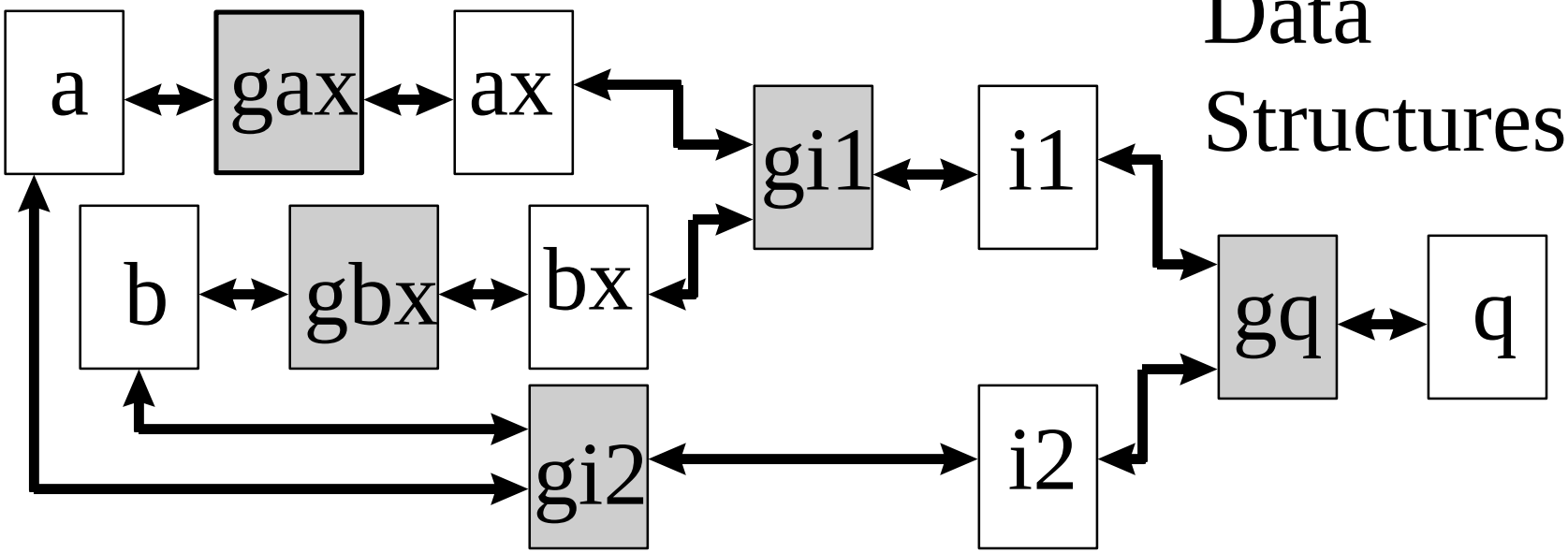
Possible Result of Parsing Nets

Net Table				
Index	*Name*	*Type*	*Fanout*	*Value*
0	a	PI	1,3	0
1	b	PI	1,4	0
2	q	PO		0
3	i1		0	0
4	i2		0	0
5	aprime		2	0
6	bprime		2	0

WINFHDL Style



Linked Data Structures



Access to Structures