Lecture - 8 X Verilog Description Styles
Two different styles of description. Behavioral · procedural statements -> blocking -> non-blocking \* Data flow atyle · assign net type-variable = net-type-variable; assign nettype variable = register type variable;
The assignment is continuously active so it
is used to model combinational circuits.

I varilog code can contain any number of "assignificationals."

Esignal declaration  [assign statements]  [other statements]  [other statements]  [endmodule]  Typical structure of a Verilog code  * Example:  module generate MUX (data, celect, cut):  input [15:5] data;  input [3:0] select;  input [3:0] select;  input [3:0] select;  input [3:0] select;  endmodule  [DWaat will be the corresponding handware  supherized?  * Won-constant index in an empression on RHS.  generates a MUX.  . If a constant index is present [5: amon out-data[3])  just a write will be generated.  * Example:  module generate set of MUX(a, b, f, sel);  input sel;  output [0:3] f;  cutput [0:3] f;  cutput [0:3] f;  cutput lel;  cutput [0:3] f;  cutput lel;  cutput [0:3] f;  Conditional operator  assign f = sel ? a:b; [generates a MUX]  endmodule  Scanned with CamScanner	· module
Exign statements (will be discussed later)  endmodule  Typical absenctione of a Verilog code  *Example:  module generate MUX (data, select, out);  input 15:03 select;  output out;  assign out = data [select];  endmodule  DIWRAT will be the consesponding handware  sufficisized?  Input ontine index in an empression on RHS.  generates a MUX.  If a constant index is present (g: amyor out-data(1))  just a wise will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input 10:314;  content 10:314;  generates a MUX.  Conditional operator  assign q = sel ? a:b; (generates a MUX)  endmodule	
endmodule  Typical structure of a Verilog cocle  *Example:  module generate MUX (data, select, out);  input [15:6] data;  input [3:6] select;  assign out = data [select];  endmodule  Quihat will be the consequending handware  synthesized?  *Man-constant index in an expression on RHS.  generates a MUX.  . If a constant index is present (g: awy out-data(z);  just a wike will be generated.  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input [0:3] 4;  confinitional operator  assign q = sel ? a:b; generates a MUX  endwodule	
endmodule  Typical structure of a Verilog cocle  *Example:  module generate MUX (data, select, out);  input [15:6] data;  input [3:6] select;  assign out = data [select];  endmodule  Quihat will be the consequending handware  synthesized?  *Man-constant index in an expression on RHS.  generates a MUX.  . If a constant index is present (g: awy out-data(z);  just a wike will be generated.  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input [0:3] 4;  confinitional operator  assign q = sel ? a:b; generates a MUX  endwodule	Tarron statements
endmodule  Typical structure of a Verilog code  * Example:  module generate MUX (data, select, out);  input [150] data;  input [3:0] select;  output out;  assign out = data [select];  endmodule  DIWhat will be the corresponding handware sunhesized?  If a constant index in an expression on RHS.  generates a MUX.  If a constant index is present (Eg: assign out-data[3])  just a write will be generated.  * Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input [0:3] b;  Conditional operator  assign f = sel ? a:b;  Generates a MUX  endmodule	
endmodule  Typical structure of a Verilog code  * Example:  module generate MUX (data, select, out);  input [150] data;  input [3:0] select;  output out;  assign out = data [select];  endmodule  DIWhat will be the corresponding handware sunhesized?  If a constant index in an expression on RHS.  generates a MUX.  If a constant index is present (Eg: assign out-data[3])  just a write will be generated.  * Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input [0:3] b;  Conditional operator  assign f = sel ? a:b;  Generates a MUX  endmodule	Cother statements (will be discussed later)
* Example:  module generate MUX (data, select, out);  input [15:0] data;  input [3:0] select;  output out;  assign out = data [select];  endinodule  Q What will be the corresponding handware  mulhesized?  I won constant index in an empression on RHS.  generates a MUX.  If a constant index is present (Eg: amoun out-datale);  Just a wike will be generated.  * Example:  module generate set of MUX(a, b, b, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  continual operator  assign f = sel ? a:b; / Generates a MUX  endundule	
* Example:  module generate MUX (data, select, out);  input [15:0] data;  input [3:0] select;  output out;  assign out = data [select];  endinodule  Q What will be the corresponding handware  mulhesized?  I won constant index in an empression on RHS.  generates a MUX.  If a constant index is present (Eg: amoun out-datale);  Just a wike will be generated.  * Example:  module generate set of MUX(a, b, b, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  continual operator  assign f = sel ? a:b; / Generates a MUX  endundule	endmodule
* Example:  module generate MUX (data, select, out);  input [15:0] data;  input [3:0] select;  output out;  assign out = data [select];  endinodule  Q What will be the corresponding handware  mulhesized?  I won constant index in an empression on RHS.  generates a MUX.  If a constant index is present (Eg: amoun out-datale);  Just a wike will be generated.  * Example:  module generate set of MUX(a, b, b, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  continual operator  assign f = sel ? a:b; / Generates a MUX  endundule	Tunical structure à a Verilon code
module generate MUX (data, select, out);  input [3:0] data;  input [3:0] select;  output out;  assign out = data [select];  endmodule  Q What will be the corresponding hardware  suphesized?  I mone constant index in an expression on RHS.  generates a MUX.  If a constant index is present (Eq: amy out-datale);  just a write will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input [0:3] f;  output [0:3] f;  conditional operator  assign f = sel ? a:b; generates a MUX	The state of the s
module generate MUX (data, select, out);  input [3:0] data;  input [3:0] select;  output out;  assign out = data [select];  endmodule  Q What will be the corresponding hardware  suphesized?  I mone constant index in an expression on RHS.  generates a MUX.  If a constant index is present (Eq: amy out-datale);  just a write will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input [0:3] f;  output [0:3] f;  conditional operator  assign f = sel ? a:b; generates a MUX	
module generate MUX (data, select, out);  input [3:0] data;  input [3:0] select;  output out;  assign out = data [select];  endmodule  Q What will be the corresponding hardware  suphesized?  I mone constant index in an expression on RHS.  generates a MUX.  If a constant index is present (Eq: amy out-datale);  just a write will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input [0:3] f;  output [0:3] f;  conditional operator  assign f = sel ? a:b; generates a MUX	* Example:
input [3:0] select;  output out;  assign out = data [select];  endmodule  Q What will be the corresponding hardware  input exized?  Some constant index in an empression on RHS.  generates a MUX.  If a constant index is present (Eq. amon out-datale);  just a wive will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  contitional operator  arright = sel ? a:b;  Generates a MUX  endmodule	module acuerate MUX (data refert out).
input [3:0] select;  output out;  assign out = data [select];  endmodule  Q What will be the corresponding hardware  sufficised?  - Mon-constant index in an empression on RHS.  generates a MUX.  . If a constant index is present (g: amy out-datale);  just a wise will be generated.  * Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  conditional operator  arright = sel ? a:b; generates a MUX  endmodule	input [15:0] data:
assign out = data [select];  endmodule  Dishat will be the corresponding handware englicized?  Non-constant index in an empression on RHS.  generates a MUX.  The constant index is present (Eq: assign out-datale);  just a wise will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  constant [0:3] f;  assign f = sel l a:b;  Generates a MUX  endmodule	input [3:0] refert:
assign out = data [select];  endmodule  Diwhat will be the corresponding hardware  suphesized?  I wan-constant index in an expression on RHS.  generates a MUX.  If a constant index is present (Eq: assign out-datalet);  just a wise will be generated.  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  cassign f = sel ? a:b;  generates a MUX  endmodule	
endurable  Diwhat will be the corresponding hardware subserved?	assign out = data [select]:
Dwhat will be the corresponding hardware in Messized?  -> Mon-constant index in an empression on RHS.  generates a MUX.  . If a constant index is present (Eq: among out-datale);  just a wise will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input [0:3] f;  output [0:3] f;  assign f = sel ? a:b; generates a MUX  endmodule	en disodule
Jon-constant index in an empression on RHS.  generates a MUX.  . If a constant index is present (Eg: amgu out-data[2];  just a wire will be generated.  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  conditional operator  assign f = sel ? a:b; generates a MUX  enduadule	
Jon-constant index in an empression on RHS.  generates a MUX.  . If a constant index is present (Eg: amgu out-data[2];  just a wire will be generated.  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  conditional operator  assign f = sel ? a:b; generates a MUX  enduadule	D'What will be the corresponding for home
Jon-constant index in an expression on RHS.  generates a MUX.  . If a constant index is present (Eg: assign out-data[2])  just a wise will be generated.  *Example:  module generate set of MUX(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  conditional operator  assign f = sel ? a:b; generates a MUX  enducodule	sullesized?
generates a MUX.  If a constant index is present (Eg: awgu out-datal2);  just a wise will be generated.  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  conditional operator  assign f = sel ? a:b;  generates a MUX  enduadate	
Just a wise will be generated:  *Example:  module generate set of Mux(a, b, b, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  conditional operator  assign f = sel ? a:b;  generates a MUX  enduadule	
Just a write will be generated:  *Example:  module generate set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  assign f = sel ? a:b; generates a MUX  enduable	
*Example:  module generale set of Mux(a, b, b, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  assign f = sel { a:b; generales a MUX  endmodule	inst a visse seil he courset la amon out-datalis)
module generale set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  assign f = sel ? a:b; Generales a MUX  endmodule	Just with the destrated.
module generale set of Mux(a, b, f, sel);  input [0:3] a, b;  input sel;  output [0:3] f;  assign f = sel ? a:b; Generales a MUX  endmodule	* Example:
input [0:3] a, B;  input sel;  output [0:3] 4;  assign f = sel ? a:b; Generates a MUX  endmodule	
input sel; output [0:3]4; Conditional operator assign f = sel ? a:b; Generates a MUX endmodule	
output [0:3] 4; Conditional operator assign &= sel ? a:b; generates a MUX endudule	
assign f = sel ? a:b; generates a MUX endmodule	P 671
endmodule	Conditional operator
	assign &= set (a:b; generates a MUX
Scanned with CamScanner	englinodule
. M. Allinett Willi Calli M. Allinet	Scanned with Camscanner



