

Electronic Circuits

Week 10: Inheritance



Fenerbahçe University



Professor & TAs

Prof: Dr. Vecdi Emre Levent

Office: 311

Email: emre.levent@fbu.edu.tr

TA: Arş. Gör. Uğur Özbalkan

Office: 311

Email: ugur.ozbalkan@fbu.edu.tr

Inheritance

System Verilog

Kalıtım, mevcut bir sınıfın değişken ve fonksiyonlarını, başka bir sınıfta kullanılmak istendiğinde, o değişken ve fonksiyonların kopyalarını oluşturmak yerine mevcut sınıftan o özellikleri kalıtım yapılmasıdır.

Inheritance

System Verilog

- Yanda frame ve tagframe isiminde iki sınıf örneği verilmektedir.

```
class frame;
  logic [4:0] addr;
  logic [7:0] payload;
  bit parity;
  ...
endclass

class tagframe extends frame;
  static int frmcount;
  int tag;
  ...
endclass

tagframe one = new();

initial begin
  one.addr = 0;
  ...
end
```

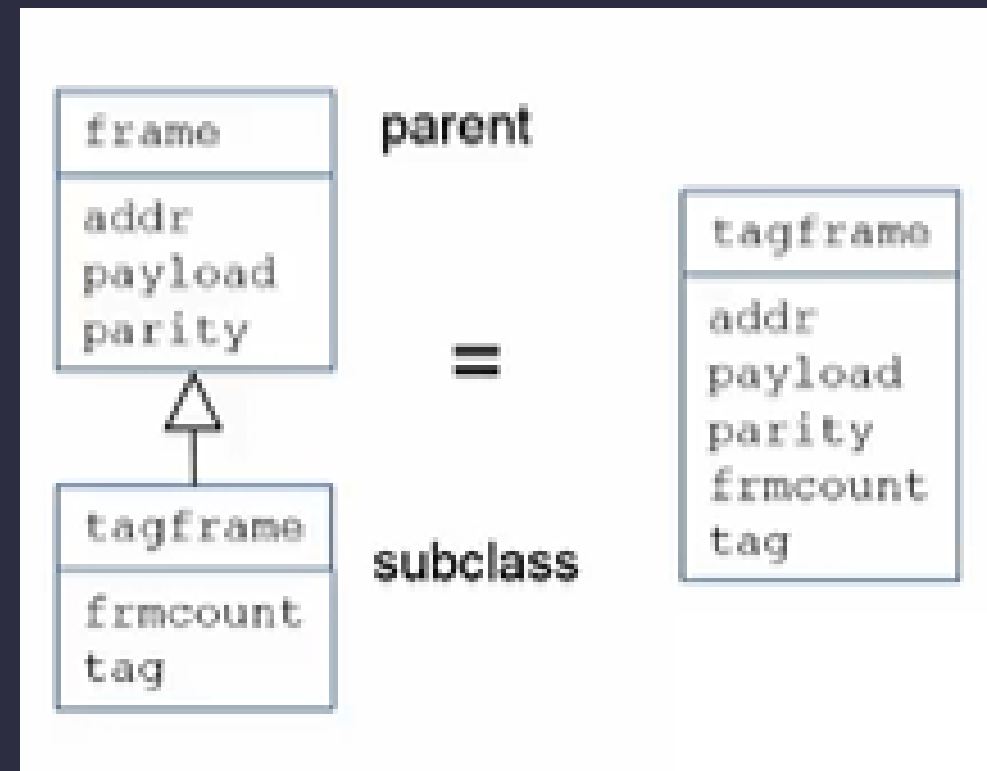
parent

subclass

Inheritance

System Verilog

- tagFrame sınıfı frame sınıfından türetilmiştir. Yani frame sınıfının tüm özellikleri tagFrame'de de bulunmaktadır.
- Yandaki şekilde sınıflar arasındaki ilişki gösterilmektedir.



Inheritance

System Verilog

- Yanda constructor barındıran bir base class örneği verilmiştir.

```
class frame;                                parent
    logic [4:0] addr;
    logic [7:0] payload;
    bit parity;

    function new(input int add, dat);
        addr = add;
        payload = dat;
        genpar();
    endfunction

    ...
endclass
```

Inheritance

System Verilog

- Eğer base class'ta constructor var ve argüman alıyorsa, `super.new` syntax'ı ile base class'ın constructor'una erişilmelidir.

Inheritance

System Verilog

- Bu yazılmazsa aşağıda sol tarafta verilen kod parçacığında olduğu gibi default olarak bir constructor çağırılması yapacak ancak argüman beslemeyeceği için compile hatası verecektir.

Inheritance

System Verilog

- Sağ tarafta ise base class'ın constructor'u doğru argümanlarla çağırıldığı için simülasyon yapılacaktır.

subclass

```
class badtagframe extends frame;
...
function new();
    super.new();
    frmcount++;
    tag = frmcount;
endfunction
...
```

subclass

```
class goodtagframe extends frame;
...
function new(input int add, dat);
    super.new(add, dat);
    frmcount++;
    tag = frmcount;
endfunction
...
```

Inheritance

System Verilog

- Kalıtım zincir şekilde de gerçekleştirilebilir. Yani bir sınıftan kalıtım yapılmış bir sınıftan başka bir sınıfta kalıtım yapabilir.

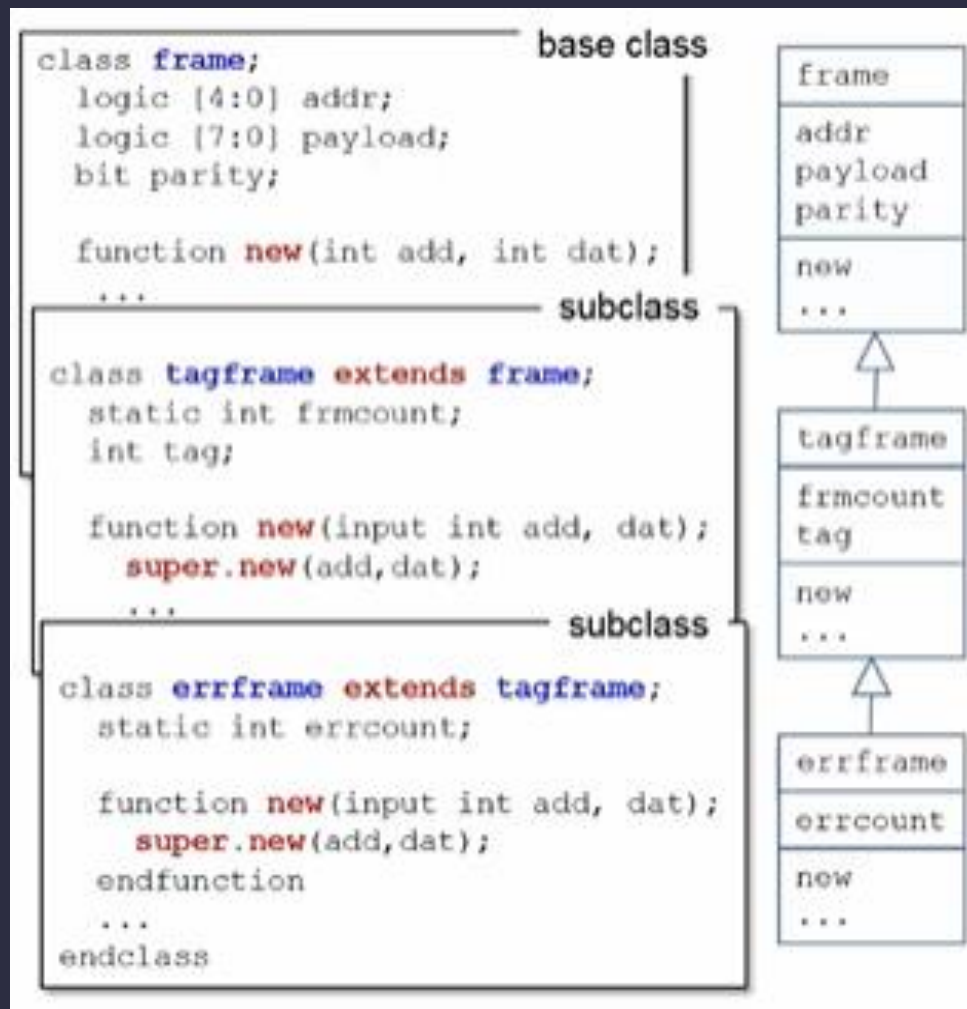
Inheritance

System Verilog

- Aşağıdaki kod parçacığında frame sınıfından tagframe sınıfı kalıtım yapmıştır. Tagframe sınıfından ise errFrame sınıfı kalıtım yapmıştır.
-
- Birbirlerinin constructorlarına zincirleme bir şekilde ulaşabilmektedirler. Ancak her bir sınıf bir üstteki sınıfın constructor'una ulaşabilir.
-
- Yani süper.super.new().. gibi bir ifade ile iki üst sınıftaki constructor'a ulaşmak mümkün değildir.

Inheritance

System Verilog



Inheritance

System Verilog

```
class parent_class;  
    bit [31:0] addr;  
endclass  
  
class child_class extends parent_class;  
    bit [31:0] data;  
endclass  
  
module inheritance;  
    initial begin  
        child_class c = new();  
        c.addr = 10;  
        c.data = 20;  
        $display("Value of addr = %0d data = %0d",c.addr,c.data);  
    end  
endmodule
```

Çıktı: Value of addr = 10 data = 20

Inheritance

System Verilog

Çıktı:

i is 1
j is 5

```
program class_inherit;
  class A ;
    integer j = 5;
    task print();
    begin
      $display("j is %0d",j);
    end
  endtask
endclass

  class B extends A;
    integer i = 1;
    // Override the parent class print
    task print();
    begin
      $display("i is %0d",i);
      $display("j is %0d",j);
    end
  endtask
endclass

  initial begin
    B b1;
    b1 = new;
    b1.print();
  end
endprogram
```