



Programa CI Inovador
Disciplina: Introdução à Verificação
Profs.: Joseana Macêdo Fachine Régis de Araújo,
Marcos Moraes, Elmar Melcher - UASC/UAAE/UFCG

Atividade Prática 03
(Verificação Funcional - Utilização de *struct* e *union*)

Observação: Enviar as respostas para cada atividade em um arquivo (nome do arquivo: "IntroducaoVerificacao_AtividadePratica03_seunome").

OBJETIVO: Implementação/simulação de *struct* e *union* (ferramenta sugerida: [EDApplayground](https://www.edaplayground.com/)).

ATIVIDADE 1: considerar a implementação a seguir.

```
module struct_expr_operator();

    typedef struct {
        int x;
        int y;
    } myStruct;

    myStruct s1;
    int k = 1;

    initial begin
        #1 s1 = '{1, 2+k};
        #1 $display("Value of x = %g y = %g by position", s1.x, s1.y);
        #1 s1 = '{x:2, y:3+k};
        #1 $display("Value of s1 = %p", s1, " by name");
        #1 $finish;
    end

endmodule
```

a) Apresentar a saída ("print").

b) Justificar os resultados apresentados.

ATIVIDADE 2: considerar a implementação a seguir.

```
`timescale 1ns/10ps

typedef struct {
    byte a;
    reg b;
    shortint unsigned c;
} myStruct;

module struct_data ();

    struct {
        byte a;
        reg b;
        shortint unsigned c;
    } myLocalStruct = '{11,1,101};

    myStruct object = '{10,0,100};

    initial begin
        $display ("a = %b b = %b c = %h", object.a, object.b, object.c);
        $display ("a = %b b = %b c = %h", myLocalStruct.a,
            myLocalStruct.b, myLocalStruct.c);
        #1 $finish;
    end

endmodule
```

a) Apresentar a saída ("print").

b) Justificar os resultados apresentados.

ATIVIDADE 3: considerar a implementação a seguir.

```
module control;

    typedef union {
        bit[5:0] opcode;
        bit[2:0] funct;
    } control_unit;

    initial begin
        control_unit ctrl;
        ctrl.opcode = 'b100011;
        $display("Ctrl: %p", ctrl);
        ctrl.funct = 'b010;
        $display("Ctrl atualizado: %p", ctrl);
    end

endmodule
```

a) Apresentar a saída (“print”).

b) Justificar os resultados apresentados.

ATIVIDADE 4: considerar a implementação a seguir.

```
typedef union packed {  
    bit [31:0] wordView;  
    struct packed {  
        bit [23:0] fraction;  
        bit [7:0] exponent;  
    } floatView;  
} MyUnion;  
  
module Testbench;  
  
    MyUnion myData;  
  
    initial begin  
        myData.wordView = 200;  
        $display("Fraction: %h, Exponent: %h", myData.floatView.fraction,  
            myData.floatView.exponent);  
        myData.wordView = -200;  
        $display("Fraction: %h, Exponent: %h", myData.floatView.fraction,  
            myData.floatView.exponent);  
    end  
  
endmodule
```

a) Apresentar a saída ("print").

b) Justificar os resultados apresentados.