

```

// *****
// **
// ** LIN2MLAW.v - LINEAR 2'S COMPLEMENT TO MU-LAW CODE TRANSLATOR
// **
// *****
// **
// **          COPYRIGHT (c) 2001 YOUNG ENGINEERING
// **          ALL RIGHTS RESERVED
// **
// ** THIS PROGRAM IS CONFIDENTIAL AND A TRADE SECRET OF YOUNG ENGINEERING. THE RECEIPT OR
// ** POSSESSION OF THIS PROGRAM DOES NOT CONVEY ANY RIGHTS TO REPRODUCE OR DISCLOSE ITS
// ** CONTENTS, OR TO MANUFACTURE, USE, OR SELL ANYTHING THAT IT MAY DESCRIBE, IN WHOLE OR IN
// ** PART, WITHOUT THE SPECIFIC WRITTEN CONSENT OF YOUNG ENGINEERING.
// **
// *****
// ** Revision      : 1.0
// ** Modified Date : 11/01/2001
// ** Revision History:
// **
// ** 11/01/2001: Initial design
// **
// *****
// **          TABLE OF CONTENTS
// **
// -----
// **  DECLARATIONS
// -----
// **  FORMAT CONVERSION
// -----
// **  1.01: Input Bias + Sign Removal
// **  1.02: Linear to Mu-Law Table
// **  1.03: Mu-Law Output Inversion
// **
// *****

```

```
`timescale 1ns/10ps
```

```

module LIN2MLAW      ( DataI, DataO );

    input  [12:00]    DataI;          // data input - linear 2's complement
    output [07:00]    DataO;          // data output - Mu-law

// *****
// **  DECLARATIONS
// *****

    wire [11:00]      LinearData;     // linear data - unsigned
    wire [11:00]      BiasedData;     // linear data - biased

    reg  [06:00]      M_LawData;      // Mu-law encoded data

    wire [07:00]      DataO;          // data output - Mu-law

// *****
// **  FORMAT CONVERSION
// *****
// -----
// **  1.01: Input Bias + Sign Removal
// -----

    assign LinearData = DataI[12] ? (~DataI[11:00] + 1) : DataI[11:00];
    assign BiasedData = (LinearData > 4078) ? 4095 : LinearData + 16;

// -----
// **  1.02: Linear to Mu-Law Table
// -----

    always @(BiasedData) begin
        casex(BiasedData)
            12'b1xxxxxxxxxxx : M_LawData = {3'b111,BiasedData[10:07]}; // full scale
            12'b01xxxxxxxxxxx : M_LawData = {3'b110,BiasedData[09:06]};
            12'b001xxxxxxxxxxx : M_LawData = {3'b101,BiasedData[08:05]};
            12'b0001xxxxxxxxxx : M_LawData = {3'b100,BiasedData[07:04]};

```

```
12'b00001xxxxxxx : M_LawData = {3'b011,BiasedData[06:03]};
12'b000001xxxxxxx : M_LawData = {3'b010,BiasedData[05:02]};
12'b0000001xxxxxx : M_LawData = {3'b001,BiasedData[04:01]};
12'b00000001xxxxx : M_LawData = {3'b000,BiasedData[03:00]}; // zero point
    default : M_LawData = {3'b000,BiasedData[03:00]};
endcase
end

// -----
// 1.03: Mu-Law Output Inversion
// -----

assign DataO = {~DataI[12],~M_LawData[06:00]};

endmodule
```