XILINX
ALL PROGRAMMABLE

IoT를 위한 FPGA설계 기초

2023년 2학기

2023.12.04

전자공학부
김성용 교수

1

- PNM(Portable Anymap Format) 영상 포맷
  - 무압축 영상 포맷
  - PNM 파일 종류
    - PPM(Portable Pixmap Format) : 컬러 영상 표현
    - PGM(Portable Graymap Format) : 계조 영상(gray-level image) 표현
    - PBM(Portable Bitmap Format) : 이진 영상(binary image) 표현
  - PNM 파일 내용
    - 영상 해더부 : 영상의 포맷을 정의
    - 영상 데이터부 : 무압축 방식, 영상 해더부에 의해 영상 데이터 구조 결정

2

- PNM(Portable Anymap Format) 영상 포맷
  - PNM 파일 해더부 양식

해더부 내용	설명
Magic number	PNM 파일의 종류와 영상 데이터 저장 방식 설명
Image width	영상의 가로 폭 정보를 정수 형태로 표현
Image height	영상의 세로 높이 정보를 정수 형태로 표현
max	각 영상 채널 내에 존재하는 컬러나 흑백 표현값의 최대치
#	주석 정보 추가 시 사용

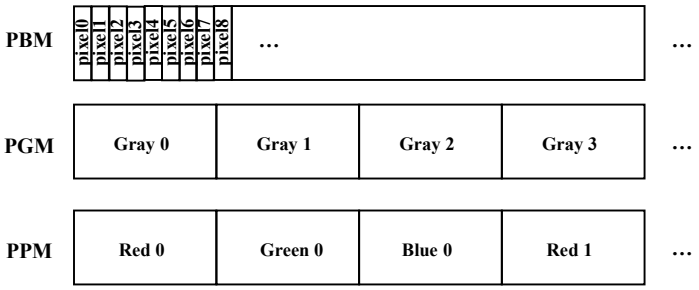
  - 해더부는 ASCII 코드 형태로 표시
  - 해더부 내 각각의 정보는 여백 문자로 분리
    - blanks, tabs, line feeds, carriage returns

- PNM(Portable Anymap Format) 영상 포맷
  - PNM 파일 해더부의 magic number
    - 2바이트를 이용하여 PNM 파일의 종류를 구분

Format	ASCII	Raw Data
PBM	P1	P4
PGM	P2	P5
PPM	P3	P6

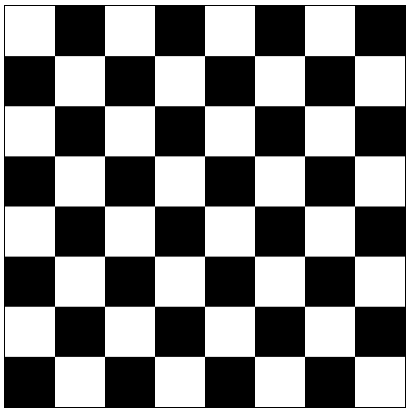
  - Magic Number
    - P1 ~ P3 : 영상 부분이 ASCII
    - P4 ~ P6 : 영상 부분이 raw data

- PNM 파일의 Raw Data 영상 저장 포맷



- chessboard.pbm 파일 내용 및 결과 영상

```
P1
# PBM Sample Image : Chessboard
8 8
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
0 1 0 1 0 1 0 1
1 0 1 0 1 0 1 0
```



- vhdl.pgm 파일 내용

```
P2
# PGM Sample Image : VHDL
25 9
255
0 0  0  0  0  0 0 0  0  0  0  0 0 0   0   0   0   0   0 0   0   0   0   0   0
0 32 0  0  0  32 0 64 0  0  0  64 0 128 128 128 128 0   0 255 0   0   0   0   0
0 32 0  0  0  32 0 64 0  0  0  64 0 0   128 0   0   128 0 255 0   0   0   0   0
0 32 0  0  0  32 0 64 0  0  0  64 0 0   128 0   0   128 0 255 0   0   0   0   0
0 32 0  0  0  32 0 64 64 64 64 64 0 0   128 0   0   128 0 255 0   0   0   0   0
0 32 0  0  0  32 0 64 0  0  0  64 0 0   128 0   0   128 0 255 0   0   0   0   0
0 0  32 0  32 0  0 64 0  0  0  64 0 0   128 0   0   128 0 255 0   0   0   0   0
0 0  0  32 0  0  0 64 0  0  0  64 0 128 128 128 128 0   0 255 255 255 255 255 0
0 0  0  0  0  0  0 0 0  0  0  0  0 0 0   0   0   0   0   0 0   0   0   0   0   0
```

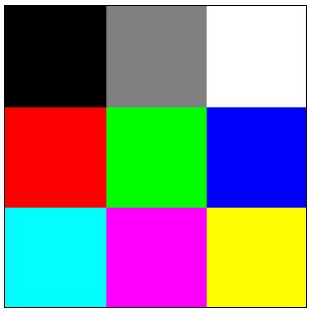

7

- vhdl.pgm 파일 결과 영상

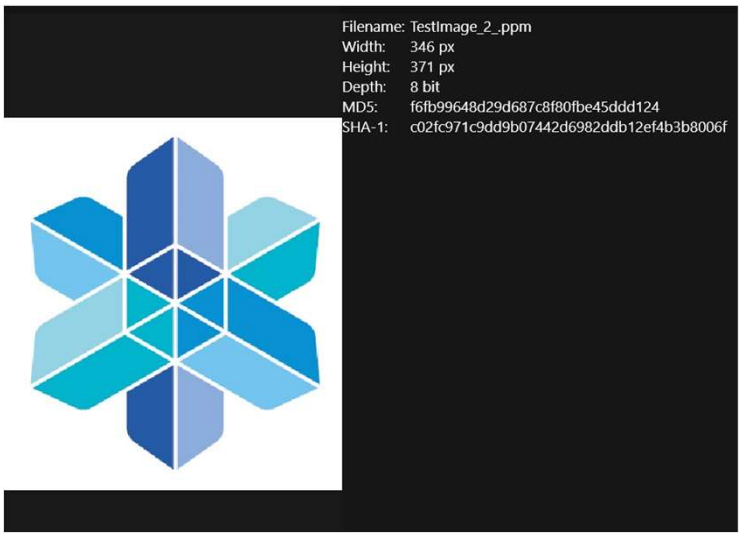
8

- color.ppm 파일 내용 및 결과 영상

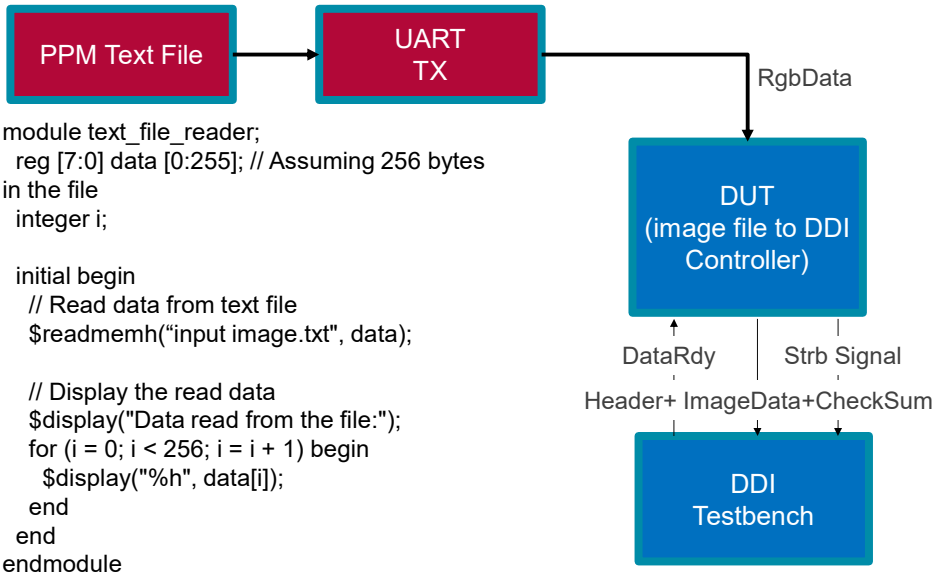
```
P3
3 3
255
0 0 0
128 128 128
255 255 255
255 0 0
0 255 0
0 0 255
0 255 255
255 0 255
255 255 0
```



9



10




```
module text_file_reader;
  reg [7:0] data [0:255]; // Assuming 256 bytes
in the file
  integer i;

  initial begin
    // Read data from text file
    $readmemh(“input image.txt", data);

    // Display the read data
    $display("Data read from the file:");
    for (i = 0; i < 256; i = i + 1) begin
      $display("%h", data[i]);
    end
  end
endmodule
```


11


```
module uart_tx(
  input wire clk,
  input wire rst,
  input wire [7:0] data,
  output reg tx
);

  // Parameters for UART configuration
  parameter BAUD_RATE = 9600; // Set your desired baud rate

  // Internal variables
  reg [3:0] count;
  reg [11:0] baud_counter;
  reg [10:0] bit_counter;
  reg start_bit;
```


12

```
// Initialize variables
always @(posedge clk or posedge rst) begin
  if (rst) begin
    count <= 0;
    baud_counter <= 0;
    bit_counter <= 0;
    start_bit <= 1;
    tx <= 1;
  end
  else begin
    // Baud rate generation
    if (baud_counter == BAUD_RATE / 2 - 1) begin
      baud_counter <= 0;
      count <= count + 1;
    end
    else begin
      baud_counter <= baud_counter + 1;
    end
```


13


```
    // Transmit data
    if (start_bit) begin
      tx <= 0; // Start bit
      if (count == 0) begin
        start_bit <= 0;
        bit_counter <= 0;
      end
    end
    else begin
      if (bit_counter < 8) begin
        tx <= data[bit_counter];
        bit_counter <= bit_counter + 1;
      end
      else if (bit_counter == 8) begin
        tx <= 1; // Stop bit
        bit_counter <= bit_counter + 1;
      end
      else begin
        start_bit <= 1;
        bit_counter <= 0;
      end
    end
  end
endmodule
```


14