Table of Contents

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
%   EE314 – Term Project    : SaleTerminal
    %
%   Description             : The Helper script to generate Memory
    %
%                            Initiation File (.mif) storing RGB
 values    %
%                            of product images.
    %
%                            (# of bits required = 100x100x3x8x12)
   %
%   Owner                   : Ataberk ÖKLÜ – 2305142
    %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%
```

Clean Start

```
clc;
clear;
close all;
```

Some Constants

```
PRDC_IMG_WIDTH  = 100;
PRDC_IMG_HEIGHT = 100;
PRDC_NUM        = 12;
COLOR_CHNL_NUM  = 3;
FPGA_CLR_WIDTH  = 8;
```

Directory Definitions

```
WORKING_DIR = "../";
IMAGES_DIR  = append(WORKING_DIR, "Images/");
ROM_DIR     = append(WORKING_DIR, "ROM_Init/");
```

Extracting Image Files

```
Image_Files_List = dir(IMAGES_DIR);
Image_Files_List = Image_Files_List(3:end);
number_of_images = length(Image_Files_List);
assert(PRDC_NUM == number_of_images, "# of Products does not match the
 number of images");
```

Creating MIF File

```
fileID = fopen (append(ROM_DIR, 'ROM_IMAGES.mif'), 'w');
% Properly Format the File
fprintf(fileID, 'DEPTH = %d;\n', 100*100*number_of_images);
fprintf(fileID, 'WIDTH = %d;\n', 8*3);
fprintf(fileID, 'ADDRESS_RADIX = UNS;\n');
fprintf(fileID, 'DATA_RADIX = HEX;\n');
fprintf(fileID, 'CONTENT\nBEGIN\n');
```

Extracting Images and Adding to MIF file

```
for j = 1:number_of_images
    target_file = append(Image_Files_List(j).folder, '\', ...
                            Image_Files_List(j).name);
    [target_path, target_name, target_ext] = fileparts(target_file);
    I = imread(target_file);

    %Extract RED, GREEN and BLUE components from the image
    R = I(:,:,1);
    G = I(:,:,2);
    B = I(:,:,3);

    %Save values to .mif file in HEX format
    for i = 1:size(R(:), 1)
        fprintf (fileID, '%d : %02x%02x%02x;\n', ...
            i+PRDC_IMG_WIDTH*PRDC_IMG_HEIGHT*(j-1)-1, R(i), G(i),
 B(i));
    end

end
```

Close and Save File

```
fprintf (fileID, '%s', "END;"); % COLOR (dec) -> print to file (hex)
fclose (fileID);
```

Published with MATLAB® R2020b