```
function handles = getFileNameFormat( handles )
% This function describes the expected file name format.

%% Specify File Format
% Written to anticipate additional versions of file formats.
% Currently hard coded for just Version 1. Can add additional cases in the future if additional formats are needed.

% Version 1 Format
% Example: VRTC_20170329_0507_Ford_Fusion_2010_ICE_30PassBy_002.wav
% VRTC: TestLocation
% 20170329: yyyymmdd
% 0507: 4-char daily sequence number
% Ford: Make
% Fusion: Model
% 2010: Model Year
% ICE: Vehicle type (ICE/EV/HEV)
% 30PassBy: Op type (30PassBy, 20PassBy, 10PassBy, Stationary, Reverse)
% 02: 3- digit trial number

fileFormat = 'Version 1' ;

%% Variable assignments based on file format

%Get all files names from the directory
batchFiles = handles.batchFileNames;
numberBatchFiles = length( batchFiles );

switch fileFormat
    case 'Version 1'

        % Variable Preallocation
        batchTestLocation = cell ( 1, numberBatchFiles ) ;
        batchDate = cell ( 1, numberBatchFiles ) ;
        batchSeqNumber = cell ( 1, numberBatchFiles ) ;
        batchMake = cell ( 1, numberBatchFiles ) ;
        batchModel = cell ( 1, numberBatchFiles ) ;
        batchModelYear = cell ( 1, numberBatchFiles ) ;
        batchVehicleType = cell ( 1, numberBatchFiles ) ;
        batchOpType = cell ( 1, numberBatchFiles ) ;
        batchTrialNumber = cell ( 1, numberBatchFiles ) ;

        for ink = 1 : numberBatchFiles
            tempFileName = batchFiles( ink ).name ;
            underscoreLocations = strfind( tempFileName, '_') ;

            % File Name Based Parameters
            batchTestLocation{ ink } = tempFileName( 1  : underscoreLocations( 1 ) - 1 ); % (VRTC in Version 1 example above)
            try
                batchDate{ ink } = tempFileName( underscoreLocations( 1 ) + 1  : underscoreLocations( 2 ) - 1 ); %
```

```
            (20170329 in Version 1 example above)
        catch
            disp(ink)
            disp(tempFileName)
            disp(underscoreLocations)
        end
        batchSeqNumber{ ink } = tempFileName( underscoreLocations( 2 ) + 1  : underscoreLocations( 3 ) - 1 ); % (0507 in Version 1 example above)
        batchMake{ ink } = tempFileName( underscoreLocations( 3 ) + 1  : underscoreLocations( 4 ) - 1 ); % (Ford in Version 1 example above)
        batchModel{ ink } = tempFileName( underscoreLocations( 4 ) + 1  : underscoreLocations( 5 ) - 1 ); % (Fusion in Version 1 example above)
        batchModelYear{ ink } = tempFileName( underscoreLocations( 5 ) + 1  : underscoreLocations( 6 ) - 1 ); % (2010 in Version 1 example above)
        batchVehicleType{ ink } = tempFileName( underscoreLocations( 6 ) + 1  : underscoreLocations( 7 ) - 1 ); % (ICE in Version 1 example above)
        batchOpType{ ink } = tempFileName( underscoreLocations( 7 ) + 1  : underscoreLocations( 8 ) - 1 ); % (30PassBy in Version 1 example above)
        batchTrialNumber{ ink } = tempFileName( underscoreLocations( 8 ) + 1 : end - 4 ); % (002 in Version 1 example above)

    end

    % Number of signal files only
    idxAmbients = strfind( batchOpType, 'Ambient' ) ;
    removeAmbients = cellfun( 'isempty', idxAmbients) ;
    batchSignalNamesOnly = batchOpType( removeAmbients ) ;
    numberBatchSignalsOnly = length( batchSignalNamesOnly ) ;
    batchSignalFileNamesOnly = { batchFiles( removeAmbients ).name };
    handles.batchSignalTrialNumber = batchTrialNumber( removeAmbients );

    % Ambient File Names
    idxPreAmbient10 = strfind( batchOpType, '10PreAmbient' ) ;
    retainOnlyPreAmbient10 = find( ~cellfun( @isempty, idxPreAmbient10 )) ;
    PreAmbient10 = batchFiles( retainOnlyPreAmbient10 ) ;

    idxPostAmbient10 = strfind( batchOpType, '10PostAmbient' ) ;
    retainOnlyPostAmbient10 = find( ~cellfun( @isempty, idxPostAmbient10 )) ;
    PostAmbient10 = batchFiles( retainOnlyPostAmbient10 ) ;

    idxPreAmbient20 = strfind( batchOpType, '20PreAmbient' ) ;
    retainOnlyPreAmbient20 = find( ~cellfun( @isempty, idxPreAmbient20 )) ;
    PreAmbient20 = batchFiles( retainOnlyPreAmbient20 ) ;

    idxPostAmbient20 = strfind( batchOpType, '20PostAmbient' ) ;
    retainOnlyPostAmbient20 = find( ~cellfun( @isempty, idxPostAmbient20 )) ;
    PostAmbient20 = batchFiles( retainOnlyPostAmbient20 ) ;

    idxPreAmbient30 = strfind( batchOpType, '30PreAmbient' ) ;
    retainOnlyPreAmbient30 = find( ~cellfun( @isempty, idxPreAmbient30 )) ;
```

```
        PreAmbient30 = batchFiles( retainOnlyPreAmbient30 ) ;

        idxPostAmbient30 = strfind( batchOpType, '30PostAmbient' ) ;
        retainOnlyPostAmbient30 = find( ~cellfun( @isempty, idxPostAmbient30 )) ;
        PostAmbient30 = batchFiles( retainOnlyPostAmbient30 ) ;

        idxPreAmbientStat = strfind( batchOpType, 'StatPreAmbient' ) ;
        retainOnlyPreAmbientStat = find( ~cellfun( @isempty, idxPreAmbientStat )) ;
        PreAmbientStat = batchFiles( retainOnlyPreAmbientStat ) ;

        idxPostAmbientStat = strfind( batchOpType, 'StatPostAmbient' ) ;
        retainOnlyPostAmbientStat = find( ~cellfun( @isempty, idxPostAmbientStat )) ;
        PostAmbientStat = batchFiles( retainOnlyPostAmbientStat ) ;

        idxPreAmbientRev = strfind( batchOpType, 'RevPreAmbient' ) ;
        retainOnlyPreAmbientRev = find( ~cellfun( @isempty, idxPreAmbientRev )) ;
        PreAmbientRev = batchFiles( retainOnlyPreAmbientRev ) ;

        idxPostAmbientRev = strfind( batchOpType, 'RevPostAmbient' ) ;
        retainOnlyPostAmbientRev = find( ~cellfun( @isempty, idxPostAmbientRev )) ;
        PostAmbientRev = batchFiles( retainOnlyPostAmbientRev ) ;

        % Update remaining handles
        handles.batchTestLocation = batchTestLocation ;
        handles.batchDate = batchDate ;
        handles.batchSeqNumber = batchSeqNumber ;
        handles.batchMake = batchMake ;
        handles.batchModel = batchModel ;
        handles.batchModelYear = batchModelYear ;
        handles.batchVehicleType = batchVehicleType ;
        handles.batchOpType = batchOpType ;
        handles.batchTrialNumber = batchTrialNumber ;
        handles.batchSignalNamesOnly = batchSignalNamesOnly ;
        handles.numberBatchSignalsOnly = numberBatchSignalsOnly ;
        handles.batchSignalFileNamesOnly = batchSignalFileNamesOnly;
        handles.fileFormat = fileFormat;

        handles.fileNamePreAmbient10 = PreAmbient10.name ;
        handles.fileNamePostAmbient10 = PostAmbient10.name ;
        handles.fileNamePreAmbient20 = PreAmbient20.name ;
        handles.fileNamePostAmbient20 = PostAmbient20.name ;
        handles.fileNamePreAmbient30 = PreAmbient30.name ;
        handles.fileNamePostAmbient30 = PostAmbient30.name ;
        handles.fileNamePreAmbientStat = PreAmbientStat.name ;
        handles.fileNamePostAmbientStat = PostAmbientStat.name ;
        handles.fileNamePreAmbientRev = PreAmbientRev.name ;
        handles.fileNamePostAmbientRev = PostAmbientRev.name ;

end
```