



Channel Coding (COMM604)

Practical Assignment Guidelines

Faculty of Information Engineering and Technology
Department of Communication Engineering





- You will have to read a video using matlab:

```
obj=VideoReader('highway.avi');  
a=read(obj);
```

- To get the number of frames in the video you can use the following:

```
frames=get(obj,'NumberOfFrames');
```

- To extract the frames of the video so you can work on them:

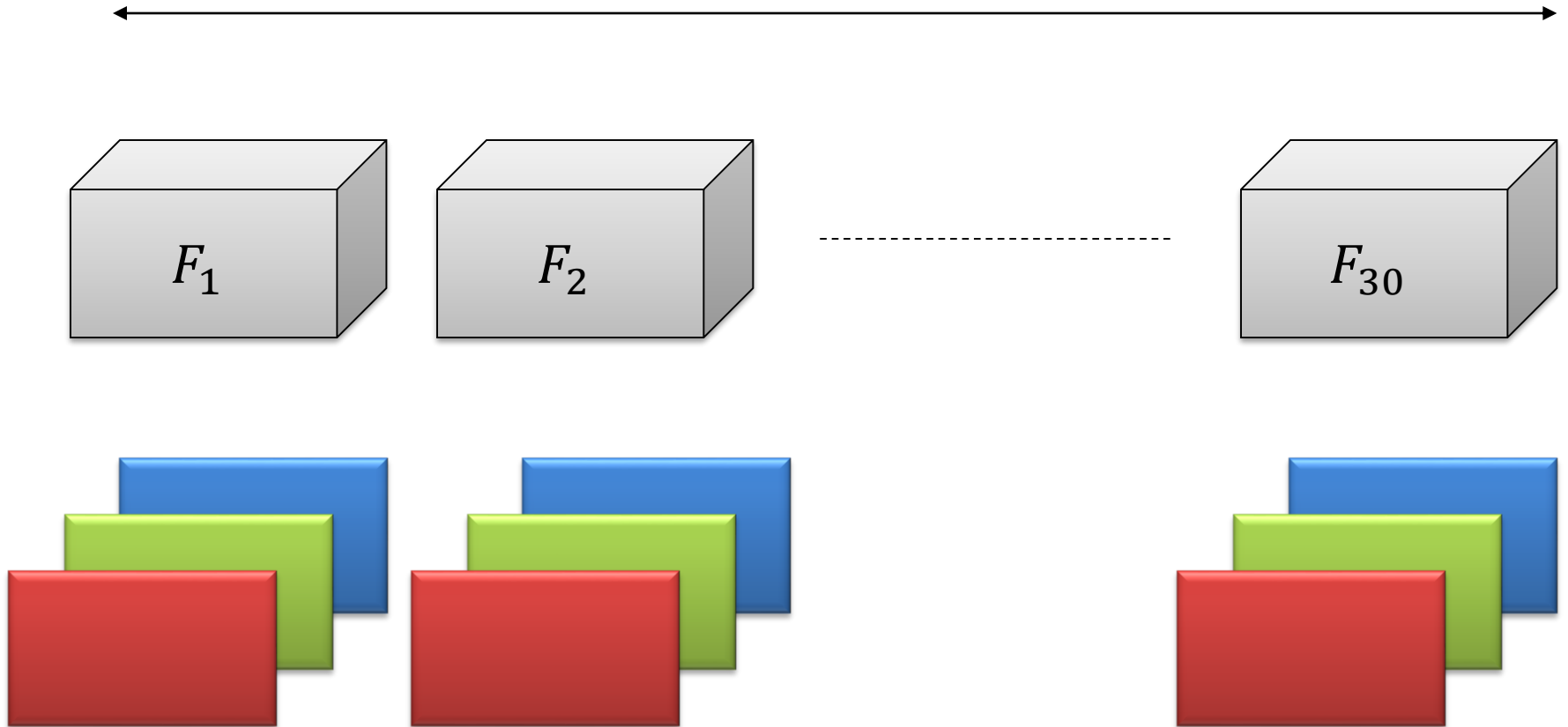
```
for i=1:frames  
    I(i).cdata=a(:,:,i);  
end
```

- In this code, you have to generate a new video with the same size as the original video so you can add these line :

```
s=size(I(1).cdata);  
mov(1:frames) =struct('cdata', zeros(s(1),s(2), 3, 'uint8'),'colormap', []);
```

Video

Frames = 30



1 Frame

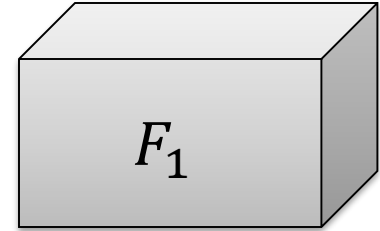
Red



Green



Blue





- You will have to extract the data of each colour in each frame:

%Red Components of the Frame

```
R=I(Frame).cdata(:,:,1);
```

%Green Components of the Frame

```
G=I(Frame).cdata(:,:,2);
```

%Blue Components of the Frame

```
B=I(Frame).cdata(:,:,3);
```



- You will have to convert the data from unsigned integers(the original format) to binary.
- There is no direct way so we convert unsigned integers to double and then from double to binary using the following:

Rdouble = double(R);

Gdouble = double(G);

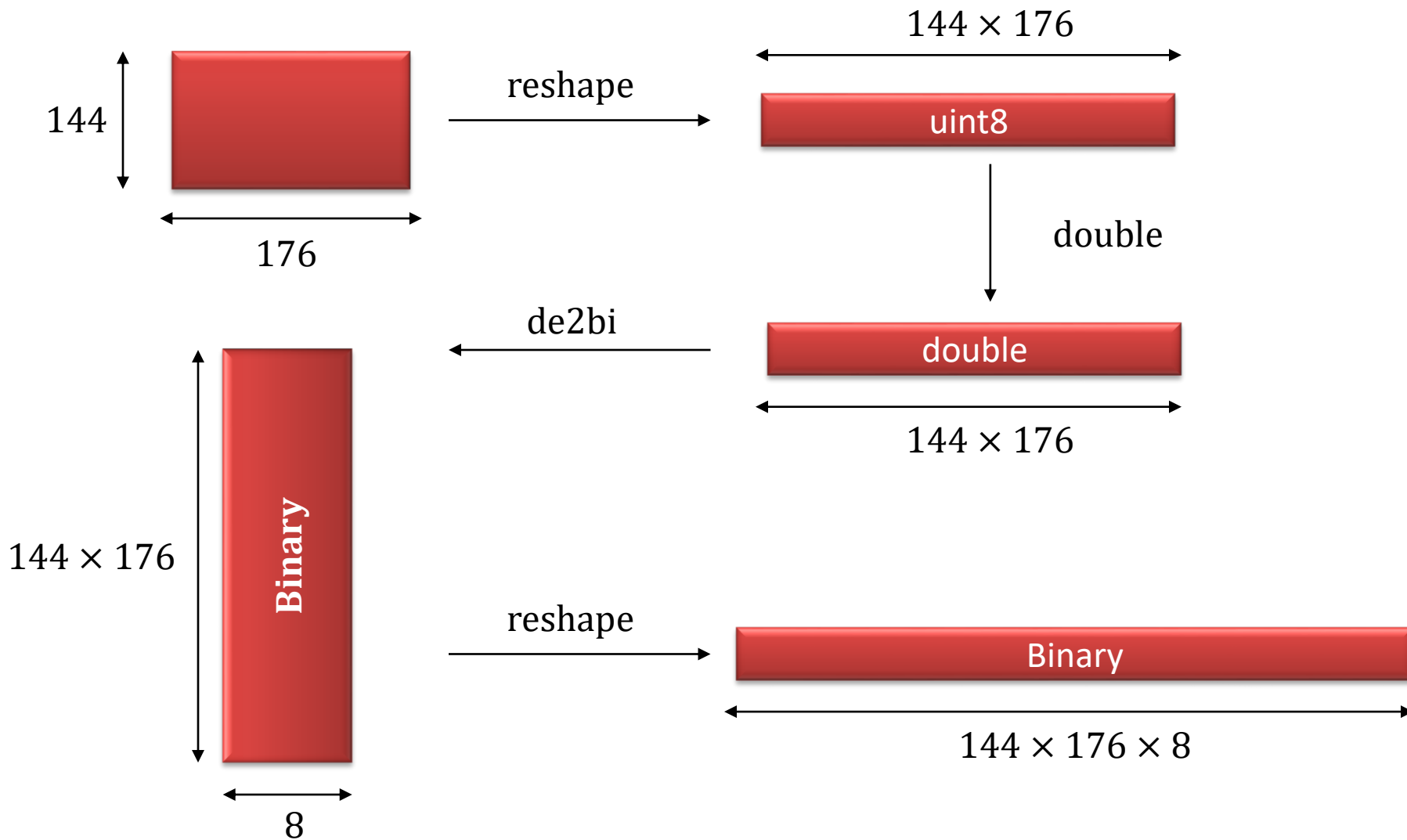
Bdouble = double(B);

Rbin = de2bi(Rdouble);

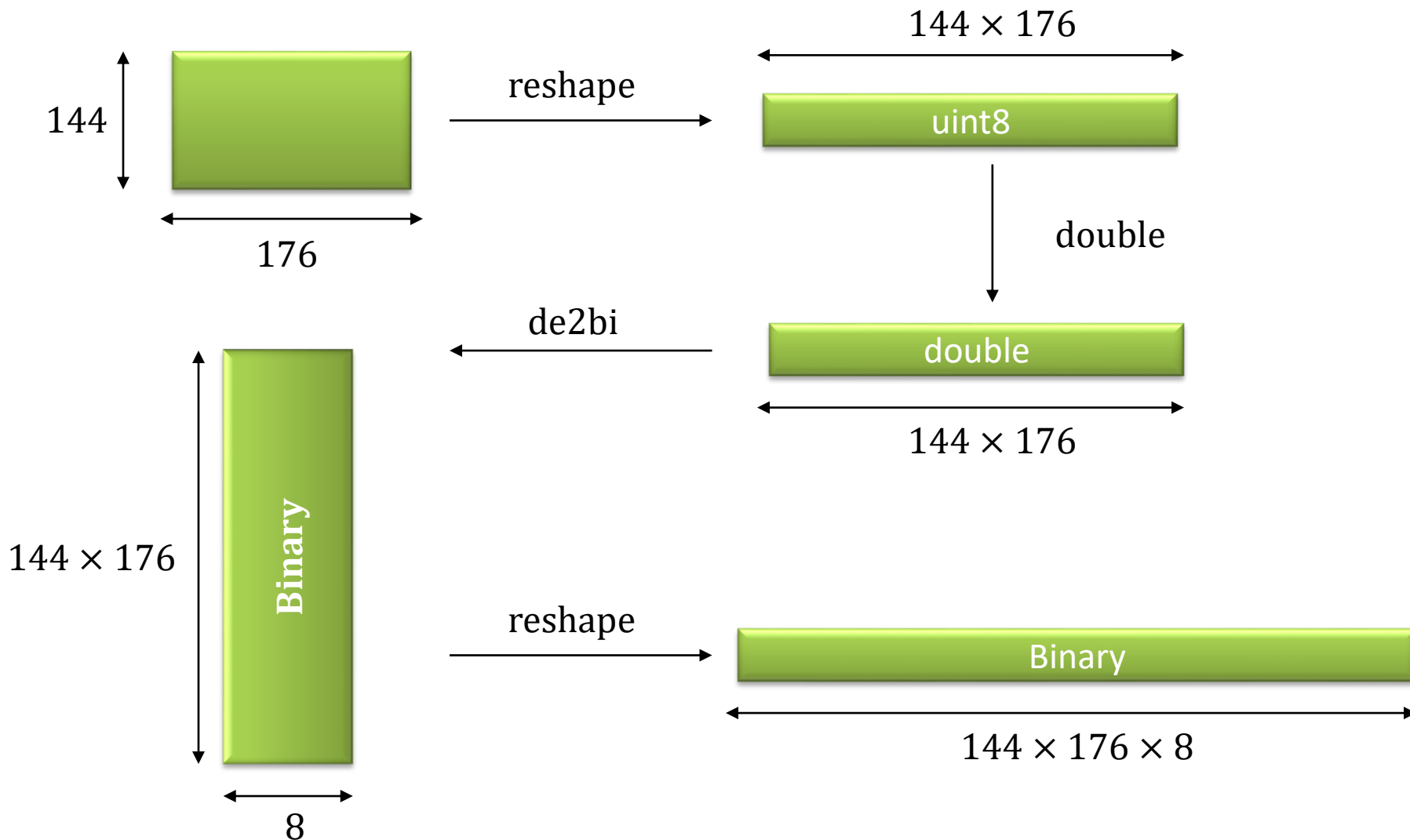
Gbin = de2bi(Gdouble);

Bbin = de2bi(Bdouble);

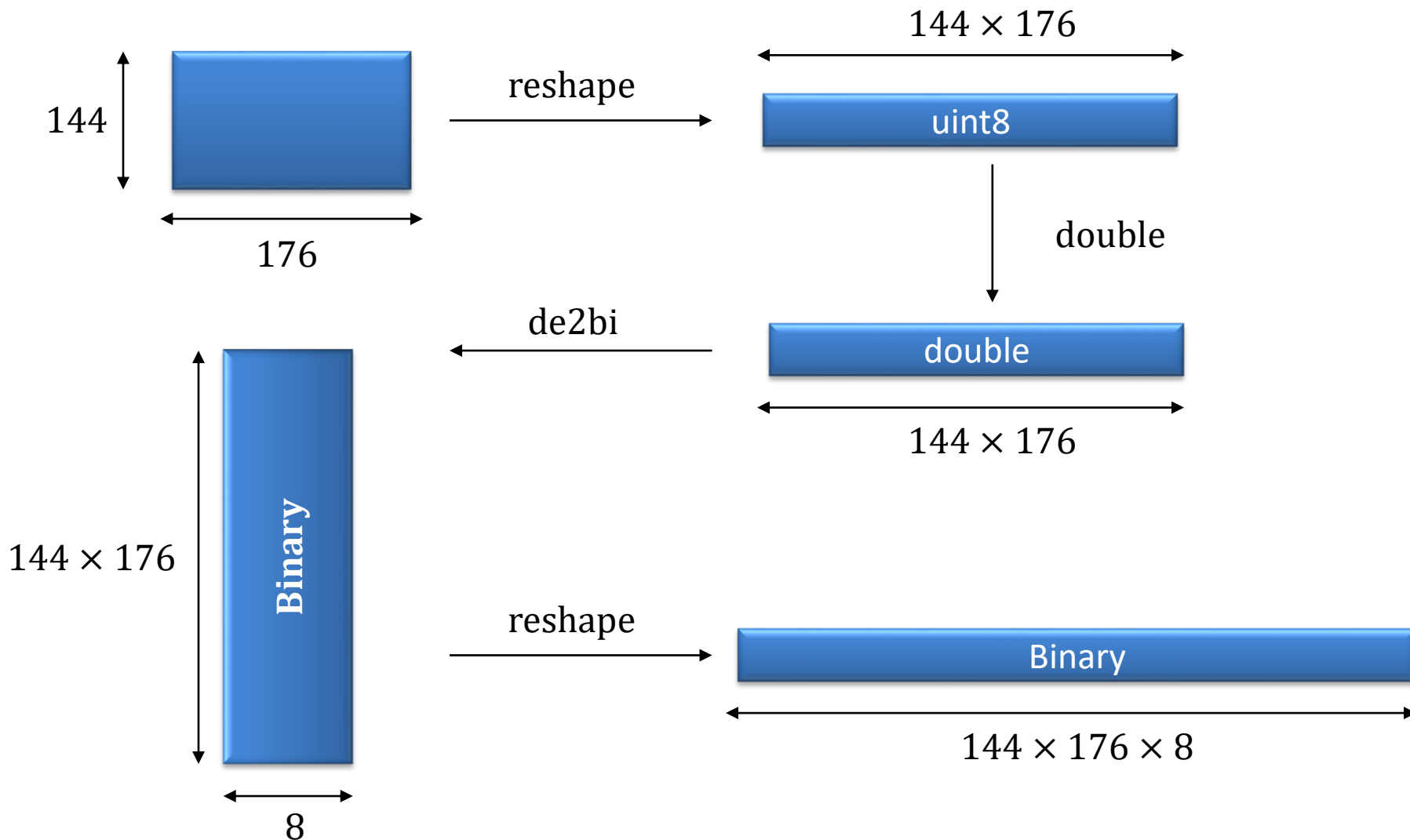
Red



Green



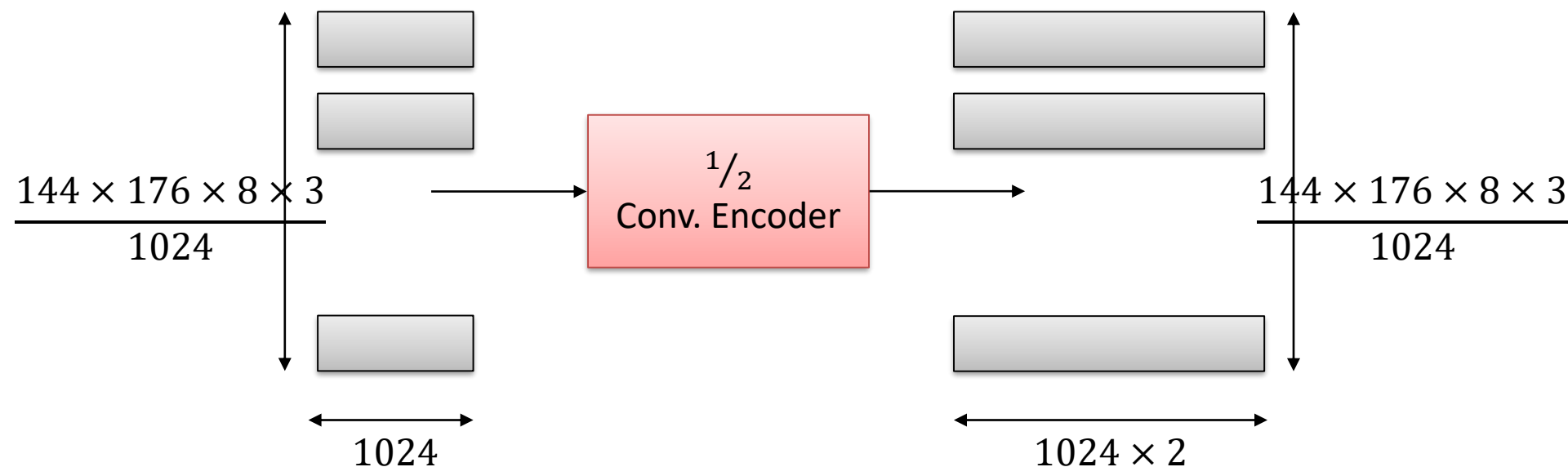
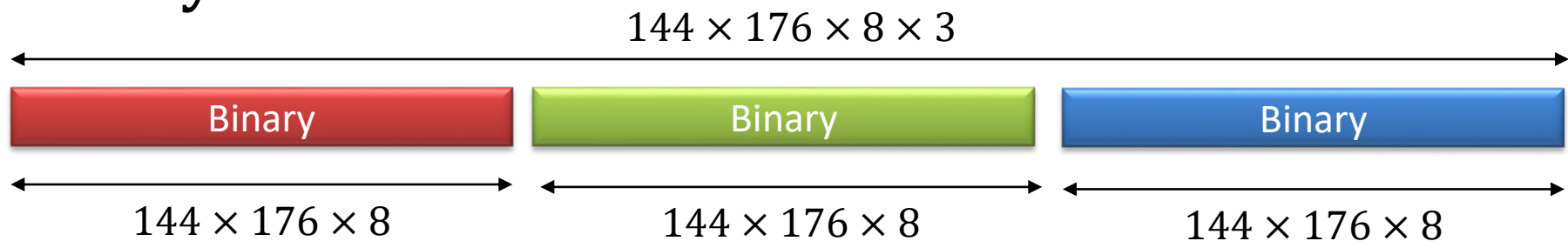
Blue





- For each colour now you have a stream of binary bits
- It is required to perform convolutional encoding on packets of bits, where each packet is 1024 bits, so you have to reshape the bits
- The following slide shows an illustration for an example using rate $1/2$

Binary Frame





- For matlab to build the trellis, you can use the following:

```
trellis = poly2trellis(7,[171 133]);
```

- To encode a certain packet:

```
encoded=convenc( packet, trellis,Puncturing Rule);
```

Note:

- In code rate 4/5 : You are given a puncturing rule given as follows:

X:1111 1111

Y:1000 1000

So the puncturing rule for the matlab will be given as follows:

Puncturing Rule =[1 1 1 0 1 0 1 1 1 0 1 0 1 0] %a digit from X followed by a digit from Y

- In case of rate $\frac{1}{2}$, you don't need to input a puncturing rule

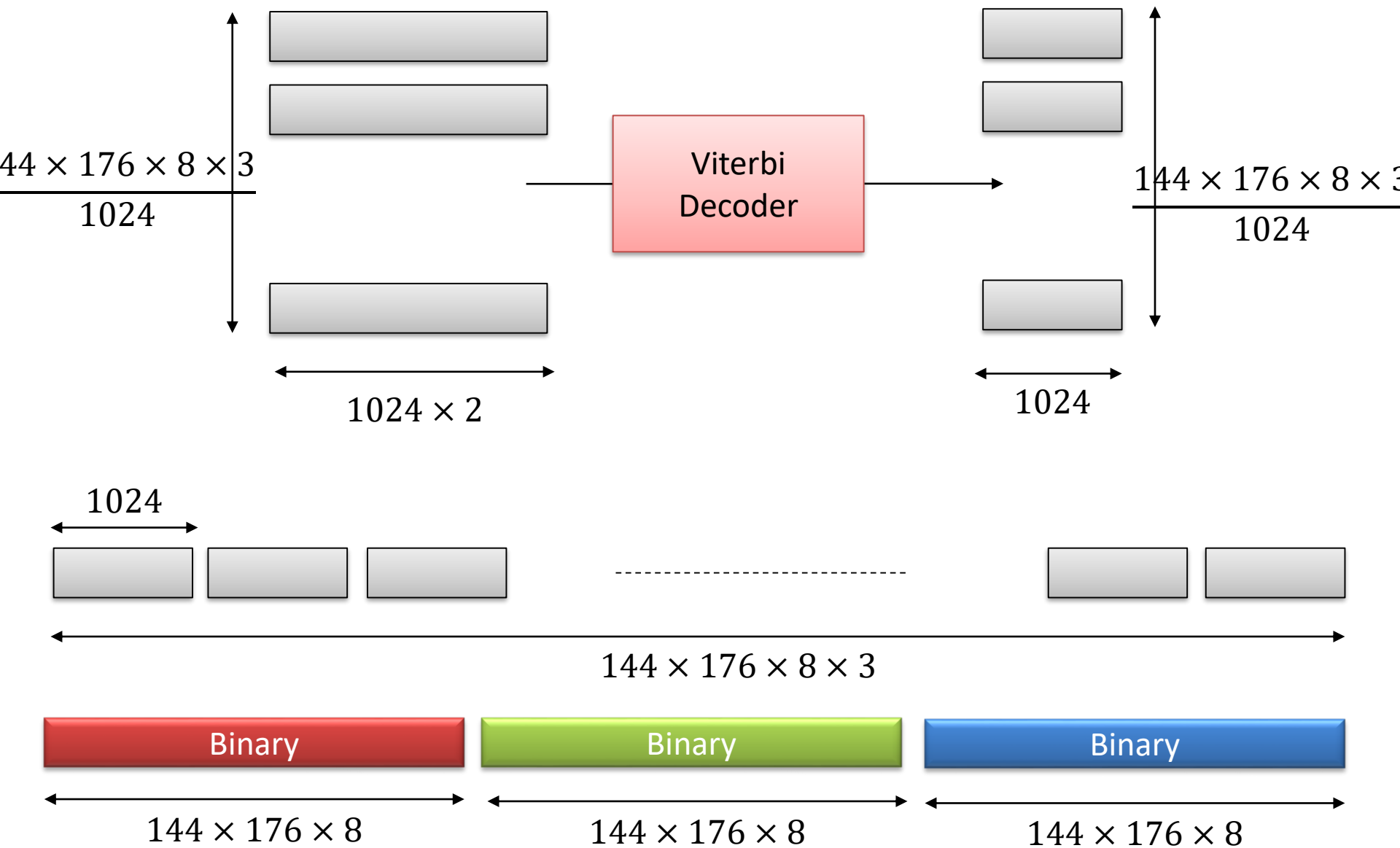
- To apply a certain probability of error p to an encoded packet, you can use :

```
Errored=bsc(encoded,p);
```

- To decode a received packet:

```
decoded=vitdec(Erored,trellis,35,'trunc','hard',punc1);
```

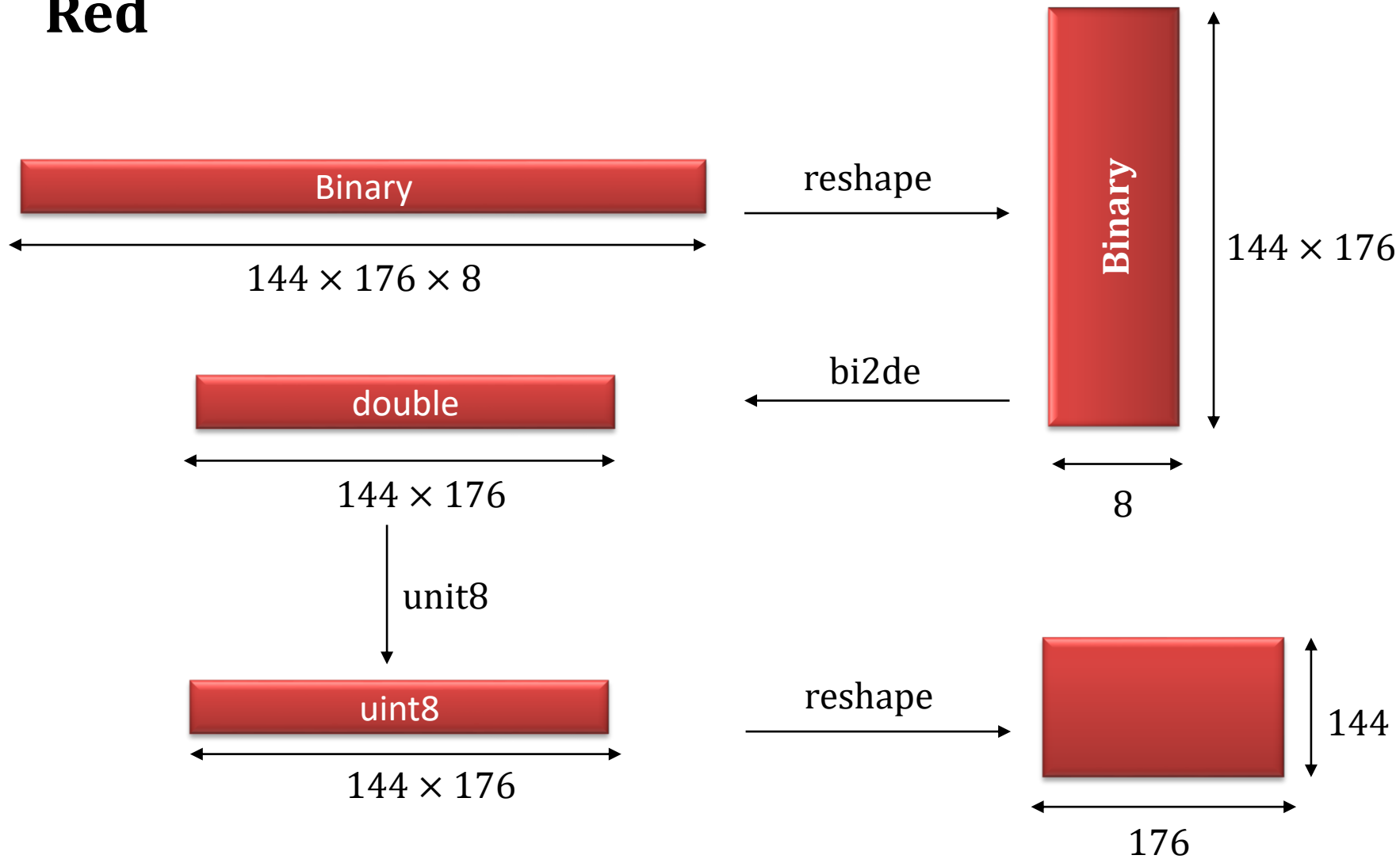
Receiver



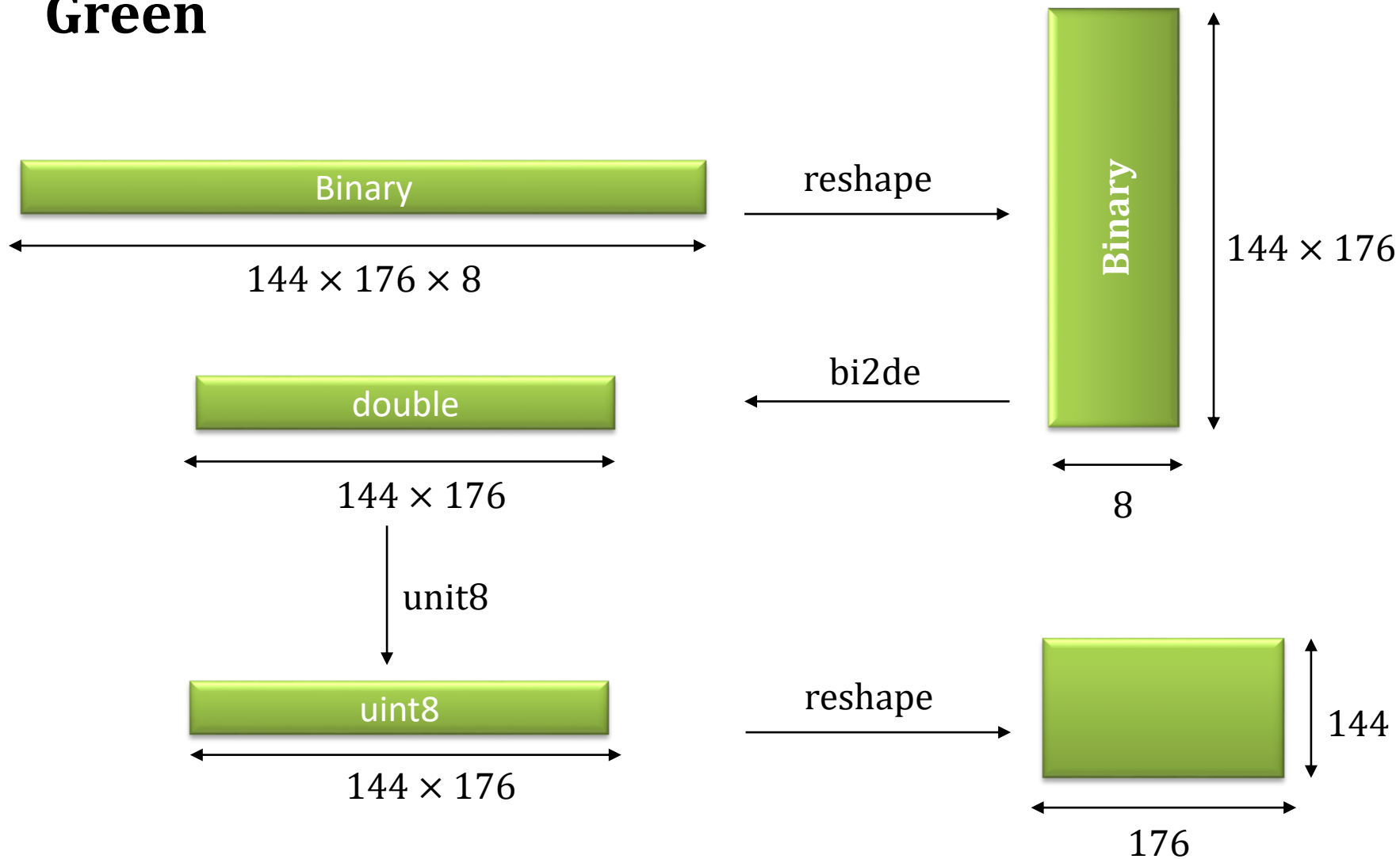


- You will have to convert the data back to the original format
- There is no direct way so we convert binary to double and then from double to unsigned integers

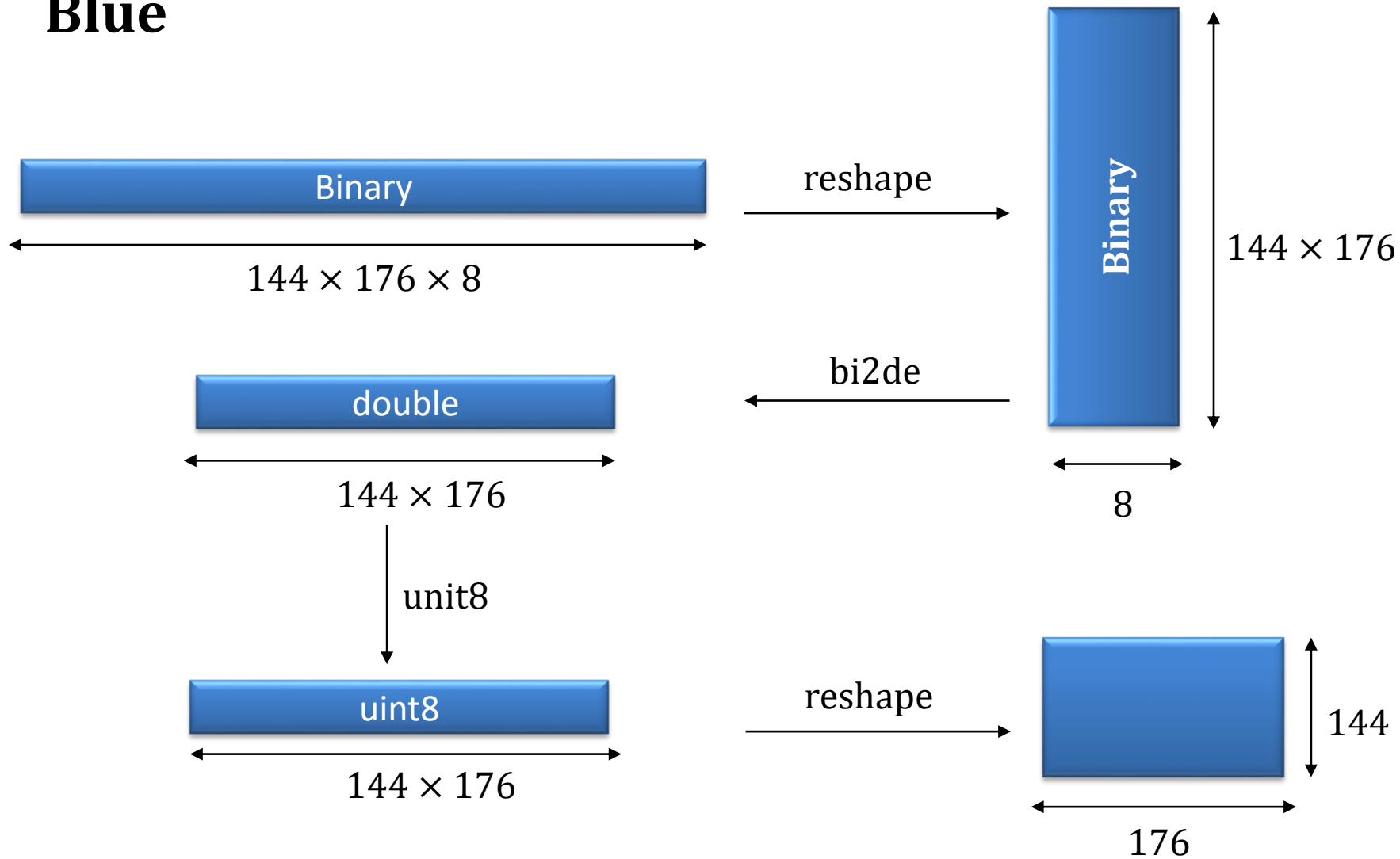
Red



Green



Blue



1 Frame

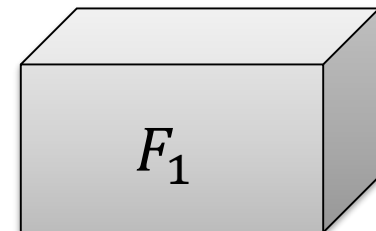
Red



Green

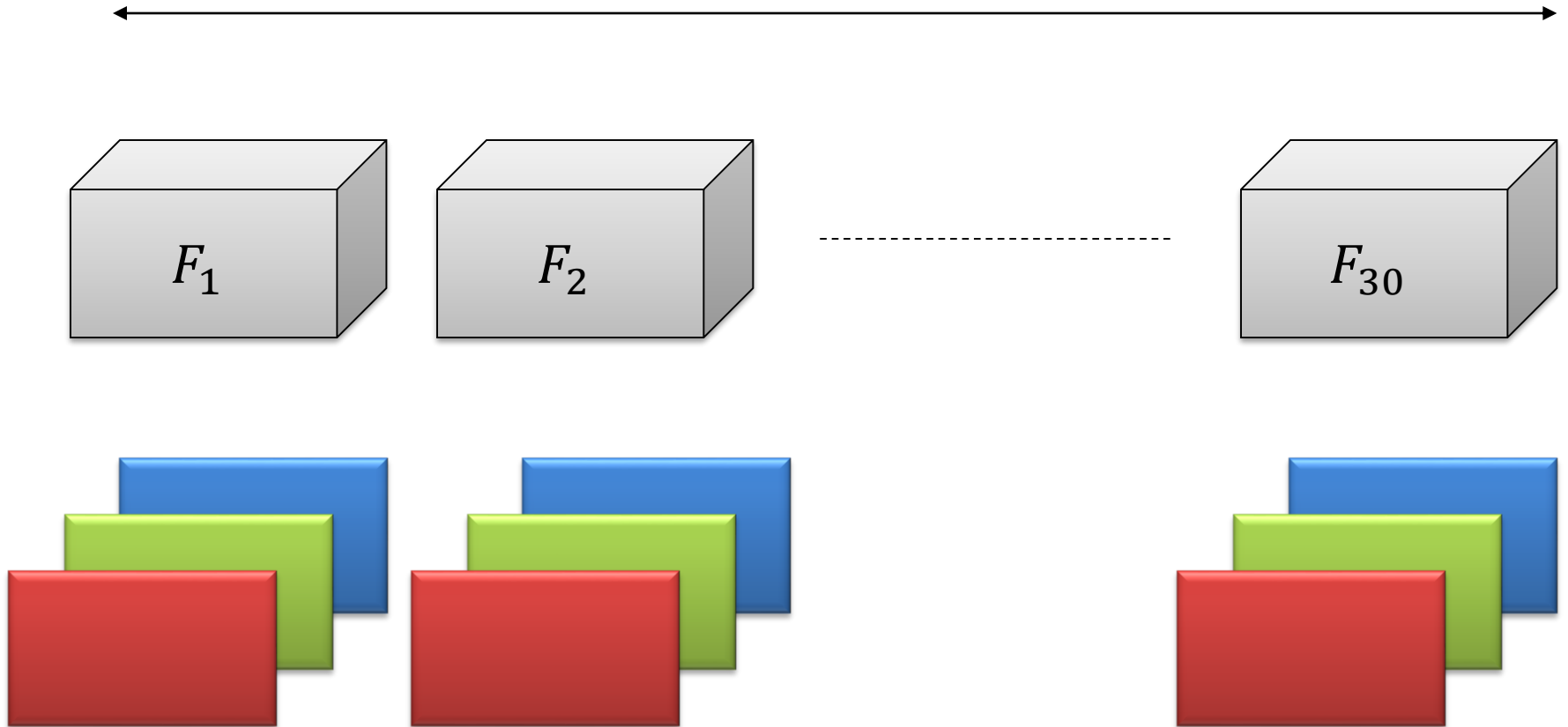


Blue



Video

Frames = 30





- To create a new video with the received frames, you have to insert frames by using:

```
mov(1,Frame).cdata(:, :, 1) = Rnewp;  
mov(1,Frame).cdata(:, :, 2) = Gnewp;  
mov(1,Frame).cdata(:, :, 3) = Bnewp;  
%where Frame is the index of the frame
```

- To save the video and play it

```
movie2avi(mov, 'D:\Channel Coding\Project\NewVideo.avi');
```

```
implay('NewVideo.avi')
```



Thank you

