```
import cv2
import numpy as np

def int2bitarray(img):
    arr = []
    for i in range(img.shape[0]):
        for j in range(img.shape[1]):
            arr.append(np.binary_repr(img[i][j], width=8))
    return arr

def create_bit_plane_video(input_image_path, output_video_path):
    # Read the input image in grayscale
    img = cv2.imread(input_image_path, 0)

    # Convert the image to a bitstream array
    arr = np.array(int2bitarray(img))
    arr = arr.reshape(img.shape)

    # Create bit planes
    plane = np.zeros((8, img.shape[0], img.shape[1]), dtype=int)
    for k in range(8):
        for i in range(arr.shape[0]):
            for j in range(arr.shape[1]):
                plane[k, i, j] = int(arr[i, j][k])

    # Initialize the reconstructed image array
    reconstructed_image = np.zeros_like(img, dtype=np.uint8)

    # Define the video writer
    fourcc = cv2.VideoWriter_fourcc(*'XVID')
    height, width = img.shape
    video = cv2.VideoWriter(output_video_path, fourcc, 1.0, (width,
height), isColor=False)

    # Reconstruct the image and write frames to the video
    for k in range(8):
        reconstructed_image += (plane[k] * (1 << (7 -
k))).astype(np.uint8)

        # Convert to the proper format for video writing
```

```
        frame = reconstructed_image.astype(np.uint8)
        video.write(frame)

    # Release the video writer
    video.release()

# Example usage
create_bit_plane_video('/content/print 5.jpg', 'bit_plane_video.avi')
```