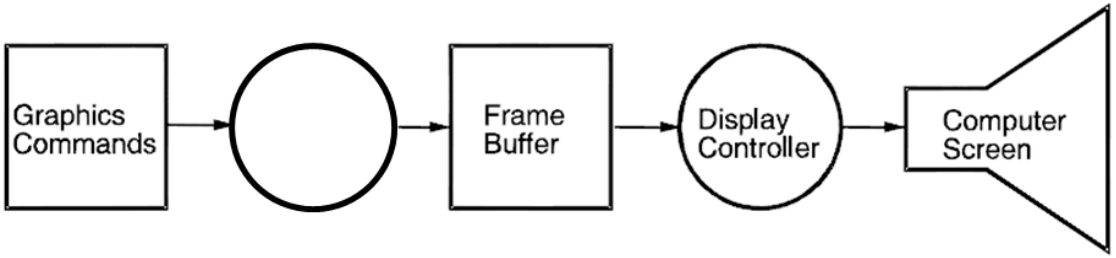
Tutorial 2
General Concepts about Multimedia

CS4185 Multimedia Technologies and Applications

1. Which component is left out in the following graphics block diagram?



2. What is the key function of each of the blocks in the above diagram?

3. Briefly explain the main differences between the missing block and a CPU.

4. Consider hosting an online single-player game. A server is dedicated to collect the player's inputs (mouse clicks), render the image frames and then send the frames to the player. The player's PC is connected to the server through the Internet. The image frames have a resolution of 1920×1080. Each pixel is represented in RGB format (3 Bytes).

   What is the size of one frame? The client's input is the position of a mouse click. How many bits are needed to describe one mouse click? If the display frame rate and the player's input rate are both 30 frames per second (*fps*), what is the player's network bandwidth needed? What is the problem here? How can it be addressed in practice?

1. Which component is left out in the following graphics block diagram?

   GPU.

2. What is the key function of each of the blocks in the above diagram?
   - **Graphics Commands**: these are the instructions generated by the CPU for the GPU to execute, e.g., draw polygon.
   - **GPU**: GPU reads the graphics commands and executes them one by one to generate output (e.g., pixel colors) to store in the frame buffer.
   - **Frame Buffer**: a 2D array of memory buffer to store the pixel colors corresponding to the display screen.
   - **Display Controller**: It reads the frame buffer and generates the control signals for the display screen to show the corresponding colors.

3. Briefly explain the main differences between the missing block and a CPU.

   A GPU normally has many more cores than the CPU, but each core is less powerful.

   GPU cores are specially designed for computations. They can perform floating point computations very efficiently. In contrast, CPU cores are designed for making decisions efficiently, e.g., executing branching instructions. They are not very good at computations.

4. Consider hosting an online single-player game. A server is dedicated to collect the player's inputs (mouse clicks), render the image frames and then send the frames to the player. The player's PC is connected to the server through the Internet. The image frames have a resolution of 1980×1080. Each pixel is represented in RGB format (3 Bytes).

   What is the size of one frame?

   $3 \times 1920 \times 1080 = 5.93\ \text{MB}$

   The client's input is the position of a mouse click. How many bits are needed to describe one mouse click?

   To address 1920 pixels, we need 11 bits.

   To address 1080 pixels, we also need 11 bits.

   Hence, we need to use a total of 22 bits to indicate the position.

If the display frame rate and the player's input rate are both 30 frames per second (*fps*), what is the player's network bandwidth needed?

Upstream: $22 \times 30\ bps = 660\ bps$

Downstream: $5.93 \times 30 \times 8\text{M}bps = 1.39\text{G}bps$

While the upstream rate is small, the downstream one is too high.

What is the problem here? How can it be addressed in practice?

This requires a very high network bandwidth to transmit the frame data. With the current technology, we can have two options. One is for the server to compress the images before sending, assuming that the player's PC can perform fast decompression. The other option is for the player's PC to perform the rendering locally. This assumes that the player's PC has a graphics card.