THE DESIGN OF AN EMULATOR

Alpay DORUK
Namık Kemal University

H. Nusret BULUŞ
Namık Kemal University

Abstract
The purpose of this article is not about only informing about emulation, visualization technology or programming emulators but also discussing about the fundamental mentality behind emulators and emulation technology

The story of emulation had begun in the early 1960's in IBM. During these times, IBM had wide range of systems; each generation had important differences from previous ones. This fact about these systems had caused several difficulties to customers about keeping up them with the new changes and requirement of the each new system. Also because of the multiple kernel technology had not been founded in these times, the computers could not be able to do multiple tasks at the same time, that's why Batch processing were used. The process could not be executed without executing the previous processes. To be able to execute multiple processes at the same time, more than one computer were required, which means unnecessary cost due of increase of used hardware. With this purpose, IBM had started to work on S/360 mainframe system as a broad replacement of for many of their other systems. This idea had led to the born of the emulation system. [1]

Today, emulation technology is widely used in Computer Science and even in our daily routines. Emulation Technology is not used for saving from the hardware cost only but it is also used for testing new facts about different environments such as Operating Systems. Nowadays, it is possible to use Virtual Machines, game emulators, products of the network and hardware virtualization with the help of Emulation Technology.

Keywords: emulator, design, software, computers.

INTRODUCTION

The "Emulation" term comes from "Emulate" term, which actually refers to imitating, mimicking. While the Emulator term used for describing the software, Emulation is the technology itself. The main idea behind an Emulator is imitating components such as hardware to be able to run the applications on different environments that is compatible to current environment. Emulation term is mostly confused with Simulation. However, simulation is imitating functions of a device itself instead of hardware.

Emulation term is mostly confused with Simulation. However, simulation is imitating functions of a device itself instead of hardware.

Like in everything else, technology can be used for both good and bad purposes and it can not be blamed to technology itself. Unfortunately, also emulation technology had been used for piracy mostly in Entertainment Industry.

When it comes to law, it is very important to define critical lines about usage of products. The most essential point in this issue is, harming a person, company or industry itself. Emulators or console copiers are not illegal itself, but what is being done with them determines their legality. Using emulators, making copies of the cartridge of the owned copyrighted game by using console copiers is not against the laws but making any profit from the copies of the copyrighted games, downloading even the ROM file of owned copyrighted game is illegal since it affects the sales and hurts companies and Industry. [3]

It can be said history of Emulator Development includes numerous commercial competitions and those competitions can not be ignored about contribution of development process. Many companies had been found in

Entertainment Industry like Taito, Konami, Nintendo, Commodore, Atari and their great work couraged other developers. Even though some developers like Aaron Kaluszka claim [4] the first idea of Emulation comes from Colossus, it would certainly right to define 1962 as the beginning year of the development of the emulation mentality

THE USAGE OF EMULATION TECHNOLOGY

Like mentioned above, emulation technology can be used for mimicking both software and hardware and even for hacking, a great example of this is the BIOS code of a computer could be attained with BIOS dump feature of A-Max. Emulators can be categorized with numerous ways but the best way would be the categorization from usage to understand the purpose of the existence of emulators.

1. Videogame Console Emulation

The emulator that gives the ability of playing a console game in a different system such as PC or Smartphone to users is called Videogame Console Emulator. It works as same like other emulators; the only exception is the ROM files of the games which is actually necessary to run the games. The extensions of the ROM files can differ depends on the console system. For example, NES ROMs have "nes" while Sega Genesis ROMs have "smd" extension, ROMs of disc operated consoles such as PlayStation usually have iso, bin, and cue formats. [5]

2. Virtual Machines

Emulators developed for testing operating systems and processes with operation of computer architecture and functions, in a computer are called Virtual Machine. The fundamental idea behind a virtual machine is creating a virtual environment which disallows any leak between the host (actual) and guest (virtual) machine and this is the main reason why Virtual Machines can be used for concretizing the security of the system. The architecture of a Virtual Machine is more complicated if compared to Videogame Console Emulators but similar mentality between ROMs and disc or floppy image files exists. The most known Virtual Machine softwares are VMWare Workstation and WMWare Player from WMWare and VirtualBox from Oracle. After the image file booted successfully, Virtual Machine can be used. Virtual Machines can be separated into two major classes dependent on their usage [6]:

System Virtual Machine is the virtual machine that provides the complete system platform and supports the execution of the processes of the operating system. For example: VirtualBox, VMWare…

Process Virtual Machine, (also known as Application Virtual Machine) is developed for handling only one process. It is used for running the process that is compatible with different environment on current system. It is created, when the process starts and killed, when the process finishes. For example: Wine in Linux, etc.

3. In Circuit Emulation

Emulation technology is also used in embedded systems. In embedded systems, an ideal tool is the one that provides the visibility into any internal operation of any hardware or component to developer. In Circuit Emulator (shortly as ICE) is the tool that provides the mentioned visibility and behaves as same with any targeted component. ICE is mostly used for debugging and testing the code in emulated microprocessor.

In microprocessor development, emulators are classified into two groups as In Circuit Emulators and ROM Emulators. In Circuit Emulators are used for emulating the microprocessor while the task of the ROM Emulator is emulating the ROM as it can be understood easily.

In Circuit Emulators consist of a small dual port pods; the first pod is for the communication with the microprocessor socket while the second pod is used for creating an interface with workstation. [7]

4. Hardware Emulation

Hardware Emulation is a product of Firmware Approach from John Haanstra. The fundamental difference between Hardware Emulation and In-Circuit Emulation is In Circuit Emulation is focused to imitating a

microprocessor, ROM or any electronic component while Hardware Emulation targets the emulation of whole complete system.

System of Hardware Emulation based on four fundamental technological entities as the hardware, the compiler, the run time environment and supporting verification intellectual properties known as VIPs. The success at four subject mentioned is a must for ensuring the accomplished implementation which makes the information about listed critic areas obligation at Hardware Emulation [8]:

- Hardware Development
- Software Development: RTL Compilation
- Software Development: Runtime Control and Debug
- Emulation VIP Creation

5. Network Emulation

Network Emulation is a technology which aims to perform experiments on the real network system and used for performance analysis, testing and validation of the current network. In the beginning, Network Emulators developed for providence of various testbeds at Network but nowadays are very common in Network Protocol, Network Application and Mobile Application Developments. Mostly, Network Emulators are developed with characteristics of the Network Simulators. The fundamental difference between a Network Emulator and Network Simulator is Network Emulators are capable to emulate the end systems of the network and behave as they are connected to a real network while Network Simulators aren't. Another important fact is Network Simulators are weak against Real World Complexity while Network Emulators are compatible with its own virtual network.

Network Emulation is separated into three categories as,

- Hardware Based Network Emulation
  Used for shaping two or more network ports. Most notable feature is the provision of the maximum performance and precision from the dedicated hardware while the cost may be higher than expected due of the required hardware. (Ex: Linktropy 7500 Pro)
- Software Based Network Emulation
  Used for traffic analyzing over the network, controlling and manipulating the incoming and outgoing packets on current communication. It provides the maximum flexibility due of the dynamic configuration and many various implementations can be applied for different cases. Dependence of the hardware limits the performance and accurary. (Ex: Linux NISTNet, Linux NetEm)
- Network Emulation Environments
  Mostly used for developing testbeds for developers and universities for scientific researches. (Ex: PlanetLab, Orbit, EmuLab)

THE WORKING PRINCIPLE OF EMULATION

Before understanding the fundamentality of emulation running discipline, it is pretty reasonable to touch on the emulation approaches mentioned before. Like noted, emulation mentality consists of three essential approaches as La Grande Approach (Software), Firmware Approach (Hardware) and Combo Approach (Combination), those approaches can't be ignored in Emulator Development.

1. La Grande Approach

La Grande Approach, known also as True Approach is the software oriented emulation approach and developed after the simulation languages are released. An emulator that contained entirely software is called Software Emulator. The main advantage of the software based emulation is the flexibility and compatibility, reconfigurations are mostly done easily and availability in different platforms provides various opportunities to users. Software Emulators also don't need the actual hardware to be able to run since current hardware is programmed to behave as the target hardware. Thus makes the Software Emulation hardware dependent emulation system and this fact is the main reason why Software Emulation is limited with the capabilities of the hardware. Due to the dependence of hardware, Software Emulation is accepted as the slowest form of the

emulation. Video game console emulators such as PCDitto, VirtuaNES, Fusion can be given as an example to Software Emulators.

2. Firmware Approach

Firmware Approach, known also as Hardware Approach is the hardware oriented emulation approach and developed by John Haanstra in IBM System/360 Team. An emulator that contained entirely hardware is called Hardware Emulator. Usage of Firmware Approach is very limited, mostly it's used for providing back compatibility characteristic against the older systems. It is accepted as the fastest emulation approach due of the independence from hardware but includes lack of the flexibility since reconfigurations are hard to be done. Spartan from Mimic Systems can be given as an example.

3. Combo Approach

Combo Approach, known also as Combination Approach contains of both hardware and software characteristics. Suggested by IBM Engineer Larry Moss, the Father of Emulators and creates the most essential part of the Emulation. Implementation is applied by programming at software layer and using various hardwares such as microchips for mimicking the hardware at hardware layer. Combo Approach is suggested for the System/360 for back-compatibility of the old systems firstly which created the 7070 Emulator, later used as the prevalent approach in Industry. Z80 Softcard by Microsoft can be given as an example for Combo Approach.

Since the subject is about Video Game Console Emulation, it would be useful to mention about the principles of Console Emulator Softwares.

Like all emulators, console emulators are work with behaving the target system but there are several essential facts that conclude the difference from the other emulators.

The main problem is the unavailability of playing console games without the dedicated console system while the solution about it in Console Emulation is the creating the availability of playing different console games from different console systems. Defined desired goal is the Console Emulator itself and the primary benefit as the result is the independence from the console system.

In general, since different dedicated hardware components are not a requirement for running the Console Emulators successfully and the implementations consist of imitating the components only with programming, it can certainly be said Console Emulation is the result of the La Grande Approach.

Console Emulators are software and consists of programming codes that orders to hardware components how to behave. The usual way of working principle of a console emulator is dumping the ROM file, processing the inputs and serving the outputs like defined in programming codes.

The ROM files contain of the data in the Game Cartridges. Cartridges are actually a ROM (Read Only Memory), the data is only available for reading but not for writing. The ROM files in the computer system can be deleted, manipulated or even the game can be changed with ROM Hacking Techniques since it is stored in Hard Disk instead of ROM.

Several implementations which are going to be explained later are being applied while the emulator runs. Mentioned implementations consist of:

- CPU Emulation
- Memory Emulation
- Input and Output Emulations
- Graphic Emulation
- Sound Emulation
- Interrupts and Timing

THE PROGRAMMING OF AN EMULATOR

Before explaining how to program an Emulator, it should be noted that it is not an easy task since experience in programming and exhaustive information about hardware systems are fundamental requirements in Console Emulation Development. Developing a very simple game for the decided platform before developing the console emulator is a good idea to understand how the system works. To be able to write a Console Emulator, various certain steps should be followed. Mentioned steps depend on the determined result. The steps should include:

- Determination of the problem
- Determination of the platform for the emulation (Atari2600, NES, GameBoy etc.)
- Research about the legal facts
- Determination of the development environment (Programming languages, additional related programs)
- Informed about the components of the hardware
- Research about each components tasks and processes in system
- Emulation In Programming

EMULATION IN PROGRAMMING

Like mentioned before, several implementation of the hardware should be applied. Even though it depends which console is going to be emulated, in general the most essential implementation is the CPU Implementation. Except the CPU Implementation, emulation of several periodic tasks exists in Emulation Development such as output devices like joystick, vertical retrace interrupts related with video display and timer interrupts.

The second way of emulation is memory emulation. The emulation of the memory and the IO with the devices includes the emulation of the bus of the computer since the communication is provided via buses. Emulating the real behavior of the buses (arbitration, limited access) is unnecessary but sometimes it must be taken into account to for accuracy of the emulator.

Interrupt and timing emulation is another important case in Emulation. Both the external and internal timing is an important fact in Emulation. The emulator should exactly work like the usual system; it shouldn't be too slow or reverse, too fast. The games or programs must run at the same speed. Interrupt handling is one of the primary issues in mentioned fact.

Communication between the user and computer is obtained by the graphics. For computers the graphic hardware is very important and they are very strong for today's computers. The graphical hardware helps CPU in performing calculations for animation and reduces the overhead over the CPU. So the graphic emulation is also another important issue in Emulation technology.

In computers, sound and graphic hardware emulation is related to each other. The algorithms used in the emulation are different, but both graphic and sound hardware are very important in the emulated computers. Both have different types of hardware, which try to reduce the calculations in the CPU of the computer and reduce the communication between the CPU and the hardware.

CONCLUSION

Emulators are important tools used in many areas in computer domain. Emulators have different features and internal compositions. They are widely used in virtualization of hardware and software, especially Operating systems. Also they are used in testing of developed software systems.

The emulation technology plays an important role in software and hardware development.

REFERENCE

[1] History of Virtualization, Web Address: http://www.everythingvm.com/content/history-virtualization

[2] Rizatti L., "The Future of Hardware Emulation", Web Address: http://www.eetimes.com/ author.asp?section_id=36&doc_id=1321628

[3] Legality of Emulators, Web Address: http://patpend.net/ articles/ legality.shtml

[4] History of Emulation, Web Address: http://kaluszka.com/vt/emulation/history.html

[5] An introduction to game emulators, Web Address: http://www.myconsole.io/an-introduction-to-game-emulators/

[6] Difference between "process virtual machine" with "system virtual machine", Web Address: https://stackoverflow.com/questions/22477128/ difference-between-process-virtual-machine-with-system-virtual-machine

[7] Kneen J., "In Cicuit Emulators", Web Address: http://educationalstuff1.tripod.com/ice.pdf

[8] Rizzatti L., "Four Technologies Converge In Hardware Emulation", Web Address: http://electronicdesign.com/eda/four-technologies-converge-hardware-emulation