**Advanced Digital Design 1: Project Proposal  
IBM PC/AT Bus IO Controller Emulator**

The IBM PC/AT, released in 1984, was the second version of the original IBM PC, featuring a 2MHz Intel 80286 CPU and a whopping maximum of 16MiB of RAM. It also expanded the amount and width of the IO channels of the original PC/XT, sporting a full 16-bit extension bus (officially known as the AT Bus, colloquially as the ISA Bus), as well as 15 IRQ lines, and 7 DMA channels. This allowed the PC/AT to be a highly modifiable, and many manufacturers capitalized on this by creating add on boards for the AT Bus. Coupled with Windows 95, it was possible to have “Plug ‘n Play”, a method that would allow components to automatically be configured by the Operating System, i.e not having an IRQ Dip Switch on the board.

One such company was Creative Labs. Creative ruled the PC sound market in the late 80s and early 90s with products such as the SoundBlaster, SoundBlaster16, Vibra16 etc. The SoundBlaster16 was capable of full 16-bit digital Audio, as well as having an add in slot that allowed for the use of General MIDI to play music files from the PC. It was also possible to connect a Joystick to the Gameport on the read of the card, as well as a CD-ROM drive internally via IDE.

This project will focus mainly on emulating the IO components of the AT system to allow a *Creative Labs Vibra16* to play sound and possibly MIDI. The reasoning for this is that the DE10 doesn’t have enough I/O to interface with every possible ISA card (though there are ways around this, for simplicity this will not be attempted). This would involve creating an emulated “Super I/O” (though a super I/O chip would not contain a DMA controller) chip that contains:

* A control register set to control I/O functionality, enabling data lines, address lines, general chip control etcetera.
* A way of handling interrupts, from the interrupt control lines of the Vibra16 into the DE10-Nano, which tells the board it *must* service an I/O request.
* A way of providing data to the card via DMA, or direct write. Some kind of DMA controller that handles DACK/DRQ would be required.
* An interface to the ARM Processor core via Linux, so that data can be DMA’d directly in. The ARM core (running Linux) would act in place of the original 80286 in the PC/AT.

A custom riser board has been fabricated to allow the card to be interfaced with the DE10, sitting either on top of the board itself, or being connected via a 40 pin ribbon cable. The core components of the I/O chip can then be written in Verilog and flashed to the board, which should then allow access to the card.