My interest in techie things started with a music friend. He taught me to build a simple volume pedal out of wood for my guitar. I bought some kits and books from radio shack and dabbled for a bit with a book on guitar effects. Not really getting anywhere.

Working as a doorman at the Orrington, DeVry had a seminar there one day. I talked to the recruiter and soon I was enrolled. I made it through the first four terms EET with a 4.0. Effectively getting an Associated degree. When I came in for my fifth term and to get that paper the presidential administration split up the loan money to give it to more students. My future loans were now cancelled as I had taken out too much money already. Even though this was my plan I was told I had to pay the overage of the loans back now. I was basically given a DeVry credit card and I tried to make it through the next few terms going to school from 7 am to 1 PM and driving a cab at night from 6 pm to 6 am. I learned to drive with one eye open and let the other eye rest. A girl told me whales do that. Strange but true, I guess. Why don't we let whales evolve onto the land? They can breathe air already, right?

Eventually I could not handle the load trying to pay for my own apartment down by the railroad tracks and drive the cab. I also had another gig going on which was both work and my DeVry project. I was so good in the lab my teacher would send kids to me when he was busy. I was teaching my teacher how to better use the Borland tools for C and ASM. I had an old laptop back then about the size of three shoe boxes I got from my contract. I hot wired and empty memory socket on it so I could burn the EEPROM for my DeVry project.

For my DeVry project I wire wrapped an 8088 microprocessor to EEPROM for programming and RAM for storage. The system had a phone keypad for commands, an LCD character display, 8251 serial and 8255 parallel ports. The purpose of my system was to identify Killer Bee swarms, count the number of bees in the hive and trigger a lock on the access tunnel and a fumigant to kill the hive at a certain count threshold.

The Killer Bee was identified by their wing beats at 273 hz. This is middle C. So I built a two stage amplifier with a gain of one million across each amp. That signal from a microphone was fed into a bandpass filter so only that frequency would pass. I used my guitar pitch pipe and a tuning fork to test. Once the analog signal was sent to a digital to analog converter the new value was passed into a filter. Here the wing beat was turned into a digital frequency value through a Schottky trigger. I passed that trigger into the 8255 parallel chip. Other signals included photo sensor diodes at the entrance and exit to count the bees in the hive. I could run a test program on my laptop that was connected to the trap through a serial port. I was dabbling in an FM transmitter and receiver. While I was doing this some doctors from a university basically did the same thing with a sound blaster.

I did not get that associates paper and I did not land a solid electronics job. I ended up driving and cooking. I did go into Best Buy and buy a book called "Tricks of the Game Programming Gurus". I had an old 286 computer, so I learned how to build a game like Wolfenstein. In the More Tricks there was more about a voxel engine. I remember it taking all night to compile. I would wake up and make some changes. Compile and go back to sleep. Living with my buddy in Chicago who spent his late night time on AOL was laudable when I got a 486 and my compile time when to twenty minutes. No more late night chats. That's how I learned to program a video game engine.

My first break was making a bar top quiz and card game. The unit did really well for sales. We used RAD Smacker and built a 2D engine with it. We had some problems with the lead and when he left, I was the

lead for a bit. The environment was encumbering so when a new chance was presented, I took it. This is not normally my way now.

I had the chance to be the lead on an arcade racing game. While we were working on the prototype, I built a modular track system and a drivable vehicle with a wind shield view and a third person view. We brought in an experienced racing developer from Japan. All we could share was If, while and other code terms. We had a translator to work with.

We needed a force feedback wheel for the racing cabinet. The new lead said it would take a SPARK workstation and about six months to design the circuit. In a couple of weekends, I had drawn up a circuit diagram and wire wrapped an amplifier we could connect to the 3DO M2 daughter card in the cabinet. The daughter card had parallel and serial ports already.

We went to test the board in the cabinet under power drain conditions connected to the wheel and the transistors popped and smoked. I was shocked and bummed. I went back to the drawing board. Verified the design was solid put in new transistors. We tried again and pop. Up in smoke in an instance. I was like no way. They gave me one more chance.

This time I wanted to see the leads code. He said: "You no see my code." I asked a couple of times. Same answer. I went to the white board, and I drew a pulse waveform. I asked through the interpreter "What is the duty cycle?" He said "Ah. Duty cycle on." In layman's terms the force feedback wheel was supposed to be pulsed as in the characteristic shaking. He had just flipped it full on. He corrected the code and on the third try we had a force feedback steering wheel sans the smoke. Voila!

My next role was converting an existing Play Station One game to the Nintendo 64. What a challenge. I had to port from fixed point math to floating point math system. This was when the floating-point standard itself was expanding and Watcom the compiler for the job was closing. At one point I was working so much and taking care of my dog at home the director let me work from home. I remember taking the Nintendo devkit home and hooking up the twelve-foot ribbon data cable. My buddy came by from few apartments down and asked if my TV reception was munged. It was. I had to put the ribbon cable in a foil blanket inside one of those metal lock boxes.

My next really big thing was getting a contract to build a share ware game shell. We wanted to build an arcade machine where the shareware game developers could show case their game. I was awarded a two-week contract to build a test unit. The prototype was able to start on a PC and load three shareware games and start them and end them. We ended up working on an API and worked with the developers for royalties and support.

I hired a trusted colleague from a previous job, and we built some amazing machines together. Over the three years plus developing the MegaRCade and LANZone we hired ten more people. It was the only job I was fired from three times. I am proud to say when I was not fired there was only one day when I was dreadfully sick that I was not there to open the shop at 7AM.

So, for a story: the first time I got fired was because the wiring harnesses for the Vegas show machines did not make it on Friday, so we had to cancel the shipment on Monday. It was not my fault the harnesses did not come in from the maker on time. I got fired that Friday. On Saturday morning I was in talking to the owner and by Monday I had created three wiring harnesses by hand and the machines were ready to go. The second time was when I had all these employees in the side shop and the owner

would not turn on the hot water heater for us to wash our hands in the bathroom. One of those things saying something to stick up for them and I was out the door. Once again, we talked it out. The third time was when he said he wasn't going to pay the health insurance for the twelve of us anymore. They went on to build a home unit for the MegaRCade.

In our system we did this thing called code injection. We basically took the direct x DLL and put some of our own code inside it. We did not mean any harm, we just wanted to run a 3D blimp across the shareware games so we could advertise. Microsoft called it an exploit and shut it down in future release. We also got to mess with XP embedded. When we went to the AMOAA show in 2002 we were the only arcade machine running XP. Some others shoe horned Windows 98 to run like a kiosk. We built our own XP kernel and took out things like outlook and internet explorer. I worked closely with big companies. One was Nvidia. We were able to get one of the first two prototype N-Force GPU embedded on a mother board. We also worked closely to work with Gateway when they were making devices smaller. We created an arcade machine for shareware developers based on the open arcade architecture just at the dusk of arcades.

We also built the LANZone. While other game centers had bench tables and chairs. We were importing really nice chairs and creating a pay for hour kiosk system where the player could play one of our games. We had games like Soldier of Fortune and others licensed to play. We were working on building the concept of E-Sports even before it was really a thing. Nice machines they could sit in a circle. Twenty-five-inch monitors with a keyboard and mouse shelf, drink holder and a bill collector. We deployed them in centers and malls. They never caught on because people could not figure them out or knew how to play and would not play for five bucks an hour even on nice rigs networked with a T1.

One other contract I almost failed they told me. In the end I was there seven years. In my two-week tenure for testing, I did not really catch the big picture they thought. It was a new Wii system for me and a huge code base going back to PSX One days as well. When my wo weeks was up they were letting me go for another week. In that time, they gave me a sound bug no one else could fix. The issue was in some disc handling code on the low end. I fixed it and I was good to go for my seven-year tenure.

I am not sure how my cube mate made it from contracting to full time switch. He would come in standup and repeat every day "I think I have a handle on this." And in the cube, he would ask me "What is an enum?". He had a bug assigned to him and he could not get it. It was the end of the game and sometimes the code would leave a green screen. It was because the code only cleared one buffer on quit. I showed him the one line of code to add. He was so excited he said' I got a line of code in the game, and you can never take that away from me". He was my drinking pal at the time, so I was glad to help. Later, he had a task to build the confidence system in a golf game. This was so that based on where the shot landed and some other factors the game could get easier. I laid out the basics of the system for him at that made it into another game as well. I also had my most difficult challenge in a profiling system for multiple consoles too.

I still remember the exact moment frozen in a picture. My lead asked me into his office one day and politely asked me to site down. I so thought I was fired for no reason.

He said…"I need you to build a user profiling and TRC system for the Wii, XBOX and PS2."

The PC doesn't even have a profile system. Monopoly Streets 75th on the Wii does.

I had created a FSM system in some down time between projects. It was used on the previous project me to create the AI for Nerf Strike. We used it all over the place on Monopoly.

My main task was the Profiling system for four players and the technical requirement checks lists. The fun stuff like "Saving… Please don't remove the memory card." In the final days I would go to lunch and three arm Alex as I affectionally called him would do just that while somehow managing to button mash a the right button at the right time.

I also noticed I had shut down bugs on the PS2 that every dev had as well. I talked to the scrum leader and wrangled them all as well. The PS2 shut down requirements are rigorous. It was one of those release issues. I had to have so many DVD's burned and delivered the tech and I were getting chummy.

I also added so much more code. Mr. Monopoly head tracking for the camera. The house and hotels growing as they go around. The GO tile itself. Animation controller for walking around the board. HQ logic for growth in the center and banking for the game. UX menus for the profiling and TRC dialogs. The FSM I created landed us in a great shape for the port.

In the end I am so proud to say the TRC and Profiling system made it through Sony, Nintendo and Microsoft certification on the first pass!

I worked on a lot of fun game play items too. I worked on the UI for PC version of Tiger Woods Golf. I also have an eight-foot amateur golfer in the game named "Bagz". In Nerf 1 I worked on the ballistics to float a nerf dart correctly. In Nerf 2 I worked on the AI systems.

For Sims Pets on the 3Ds I worked on the controls and dual cameras. We really worked on the design to alleviate eye strain going back and forth between the two screens. Other fun sims things included DJ Turntable, Cat Post and Litterbox, Werewolf and numerous little aspects of Sims life on five different PC expansions.

For me it's been a great technology journey of learning and adventure to here at which is an adventure in learning.