RAMAC RESTORATION PROJECT

by

Chris Hong, Hagop Kozanian and Neal Rambhia

SENIOR DESIGN PROJECT REPORT

Submitted in partial fulfillment of the requirements
for the degree of
Bachelor of Science in Electrical Engineering
School of Engineering
Santa Clara University

Santa Clara, California

June 9, 2004

Restoring the RAMAC

Department of Electrical Engineering
Santa Clara University
2004

ABSTRACT

The IBM RAMAC (Random Access Method for Accounting and Control) entered service in 1956 as the world's first magnetic disk drive. The drive consists of 50 2-foot diameter disks with a storage capacity of 5Mb, and it was originally controlled by a high power control unit using vacuum tube technology. Santa Clara University's MDHC (Magnetic Disk Heritage Center) is currently restoring a RAMAC as part of a heritage and education initiative.

As part of this effort, our project had a specific goal of implementing an automated control system for the RAMAC's access mechanism. This involved devising a method to move the carriage at controllable speeds, developing an accurate sensing interface and finally integrating the two by using a microcontroller as the central processing unit of our system. While the RAMAC mechanism itself was restored in accordance with its original design, our controller employed modern microcontroller technology and was furthermore interfaced to an internet-based remote access system. This provided for a unique blend of historic and modern technology.

In performing this project, our team contributed to the overall RAMAC restoration by characterizing key physical parameters of the RAMAC system and by implementing a simple controller for positioning the head at a desired disk and track address. Our systems maximum access time was on the order of 10 seconds. As the restoration project continues in the future, more sophisticated controllers and magnetic read/write functionality will be added in order to establish full functionality to the RAMAC drive.

Keywords: RAMAC, Restoration, IBM, Hard Drive, Automation.

Acknowledgements

This thesis would not have been accomplished without the help of our two advisors, Dr. Kitts and Dr. Hoagland. Their guidance and knowledge have shaped the outcome of this project immensely. The patience they had with us was invaluable to the learning process that is a senior design project.

We would like to thank IBM for giving us the chance to work on such an important historical artifact. The RAMAC would have not been conveniently located at Santa Clara University's Magnetic Disk Heritage Center were it not for their generous lease.

We would also like to thank:

- Dave Bennet, for his many donations and his personal involvement in our project.
- Pat Connolly, for rendering the mechanical components of the machine in impeccable condition prior to our project.
- Dave Bennet and Jack Grogan, for their instrumental roles as key mentors from IBM. Their experience with the machine and their advice was invaluable to our progress. Lou Taft of IBM, for advising us with a safe method to clean the disks. Thank you.
- Hitachi, for the model shop assistance they rendered

A portion of this work was supported by the National Science Foundation under Grant No. EIA0082041; any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

TABLE OF CONTENTS

page

LIST of FIGURES

Page

LIST of TABLES

Chapter 1: Introduction

The goal of our project was to restore the automated motion of the RAMAC's (Random Access Method for Accounting and Control) access mechanism. The RAMAC was the world's first magnetic disk drive, made by IBM in the 1950's. Its total capacity was a mere 5 megabytes, while the drive and its controller were gigantic, taking up more than 1080 cubic feet of space. Although it is extremely old, the RAMAC operated just as modern disk drives function. We hoped to demonstrate the functionality of the first magnetic disk drive, putting it in perspective of our modern era by combining technologies that are half a century apart. Our team enabled controllable motion, accurate sensing and full automation of the carriage and arm units of the RAMAC. The chapters that follow depict our thought process and detailed design implementation.

Throughout or project, we worked closely with faculty as well as industry engineers. We feel this interaction was greatly beneficial to us and to our society. Because of the increasing use of hard drives and their continual technological advancement, these amazing devices are often taken for granted. By witnessing the first magnetic disk drive function, people can better understand the basic operations needed to write and retrieve data.

1.1 Background

Disk drives are very significant in our lives everywhere today, and for this reason, it was a great pleasure to study, learn about, and work on restoring the first disk drive ever. Much of the productivity of the world is linked to the use of computers and the rise of information technology. Storage technology is a fundamental part of this, while magnetic disk-drive technology is a main element of storage. Over the past 50 years, this technology has drastically matured. Although some may think that the electronic storage that the RAMAC system provided was relatively minute to the amount of physical space it occupied, it is important to emphasize once again that the RAMAC is the first of all magnetic disk storage devices. The area density on the disks was merely 2000 bits/in^2, but it was the first system to store data in this manner. Other systems followed with improved densities such as the IBM 1301 and the IBM Winchester 3340. The IBM 1301 Disk Storage Unit was announced in June 1961 with an ability to monitor as many as 280 million characters of information in a single system by making use of comb-like arms flying on layers of air. Compared with the innovative IBM RAMAC, the 1301 provided a thirteen-fold increase in storage density and three times faster average access to information.[1] The Winchester 3340 was introduced in the 1970's. The IBM 3340 disk unit was an advanced technology, which more than doubled the information density on disk surfaces. It featured a smaller, lighter read/write head that was designed to ride on an air film only 18 millionths of an inch thick.[2] Winchester technology was adopted by the industry and used for the next two decades. Another way these drives have improved over the years is through their access times. The original IBM RAMAC had access times of 800 milliseconds (ms).[3] However, these days, a fast hard drive has access times of about 5 ms.

Half a century has passed since the RAMAC was first introduced. Technology has improved the speed and density of the system, but for the most part, it has not changed the way we store digital data. Currently, magnetic disk drives have an areal density of almost 100 gigabits per square inch. Compared to 2000 bits per square inch on the RAMAC, this represents an increase in storage density by a factor of 50 million over the last 50 years.[4] The latest disk drives, however, still work much like the first ones, with read-write heads darting over the surface of spinning disks. Because of this, the RAMAC, although extremely old, is still very interesting to study and explore. It has been declared an International Mechanical Engineering Landmark by the ASME and an IEEE Milestone by the IEEE.

This disk drive, which is the size of a refrigerator, is known as the RAMAC (for Random Access Method of Accounting and Control). It was designed by a small IBM laboratory in San Jose, California and was revealed by IBM in 1956. It was stored in a large cabinet and powered by a motor strong enough to run a small cement mixer.

The device is made up of a stack of 50 aluminum disks coated on both sides with a film of iron oxide. Each disk is two feet in diameter and the disk stack rotated at 1200 RPM. A pair of pneumatic read-write heads would quickly move up and down to reach a

---

[1] IBM Archives, http://www-1.ibm.com/ibm/history/reference/glossary_1.html
[2] IBM Archives, http://www-1.ibm.com/ibm/history/history/year_1973.html
[3] IBM Technical Report, pg. 10
[4] MDHC literature, http://www.mdhc.scu.edu

specific disk. Then, the disks would rotate in order for the head to access information at a designated location. There are 100 circular data tracks on each side of each disk, and each of these tracks could hold 500 characters.[5] So, the entire drive unit had a total capacity of only 5 megabytes. Below is a picture of the entire unit, just so one can see just how large it actually was.

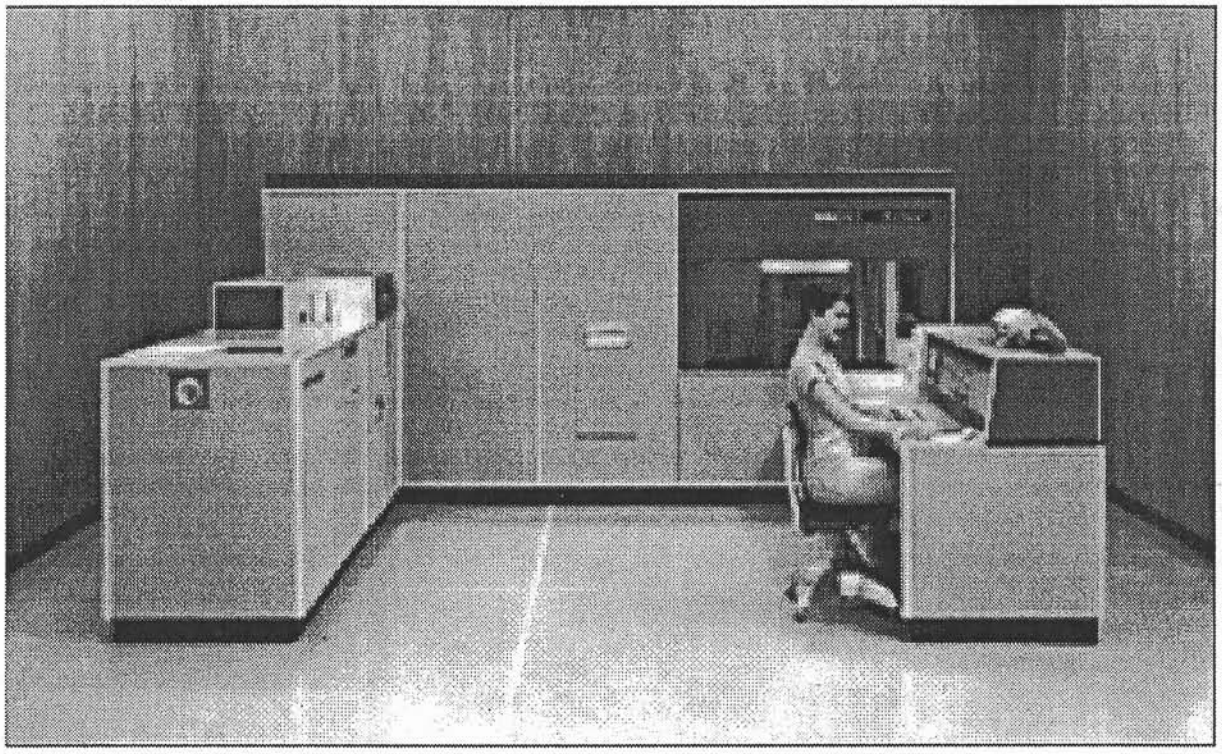


Figure 1.1-1: The entire unit of the IBM RAMAC 350

The disc drive consists of two mechanisms: The carriage, which carries the arm, and the disk array, which rotates via a 220V AC motor. The carriage was mechanically refurbished by Pat Connolly, a Mechanical Engineering student, during the summer prior to our project. The motor, which drives the disc array, however, was only recently tested – it proved to be in good shape. The original control system consisted of vacuum tubes, relays and analog processing techniques. Within the scope of our design, we planned to

---

[5] The RAMAC 350 File, pg. 63

implement a modern automated control system that directs the access arm to the selected disk and track.

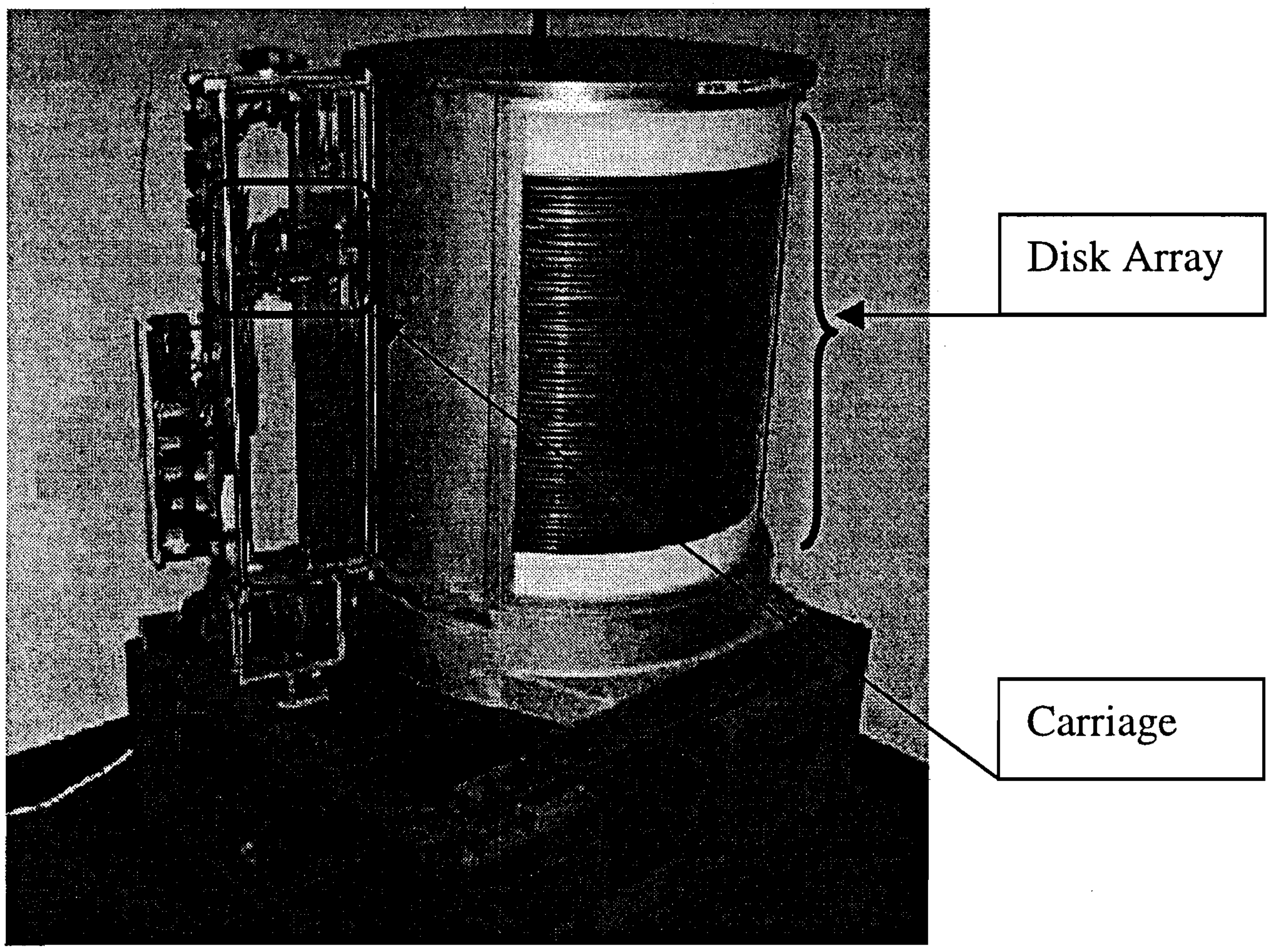


Figure 1.1-2: Image of RAMAC's carriage unit and disk array

1.2 Objectives

The following objectives define the goals of this project:

1. **Controllable manual motion**. Implement an actuator interface to allow carriage up/down and arm in/out motion at variable speeds.
2. **Accurate sensing**. Implement a sensor interface to provide carriage and arm position information.
3. **Automation**. Integrate motion and sensing into a user-friendly automated control system. Improve access accuracy and time through any means necessary.
4. **Internet-based interface**. Support the integration of an Internet-based interface that incorporated the RAMAC controller with the web so that it could be accessed from a remote system.

The controllable manual motion was the first completed element of our project. By using Pulse Width Modulation, we were able to control the speed of the carriage and arm effectively. Accurate sensing was next on the list. We were able to achieve accurate sensing quite easily via the resistor strip for the vertical motion and the potentiometer for the horizontal motion. The speed of the sensing interface however, proved to be the bottleneck in our access time. If the carriage or arm was moved too fast, the controller's sampling rate would lag behind that of the motion, rendering the sensing inaccurate. Finally, we integrated the motion and sensing into an automated system via a microcontroller. Once the system was working, we were able to optimize its access time and accuracy through various methods. An error checking and braking system were the two functions that helped us achieve access times under 10 seconds and an error rate of less than 1%. The Internet based interface was last on our list. With great help from the Robotic Systems lab at SCU, we were able to implement a web page that could be used as an interface to control our system.

The steps that were taken through the course of this year contributed greatly to the RAMAC restoration project. Although it can be improved, the robotic automation has been completed. The documentation we left behind will greatly help anyone who wishes to improve our system. Our failures, bottlenecks and achievements are highlighted throughout this thesis with the hopes that when the project is handed over to another team, so will our experience and knowledge.

1.3 System Overview

a. How does the RAMAC work?

The carriage, which is depicted in the Figure 1.3-1 below, can move up and down to access any of the disks. When the carriage is locked into place, the access arm can move in and out to its desired location.

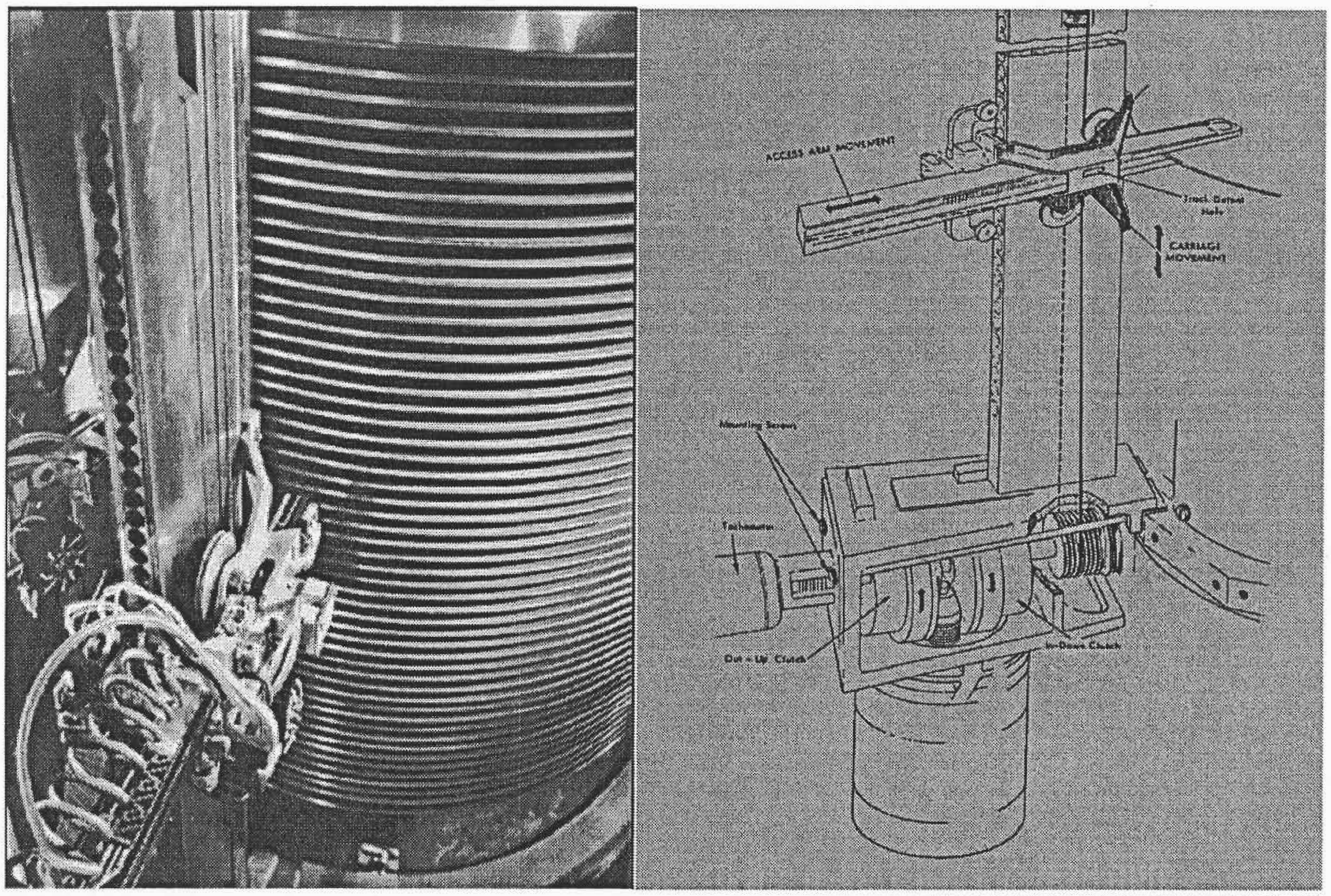


Figure 1.3-1: Image of the RAMAC's carriage unit

The AC motor at the bottom of the access mechanism drives the motion. A pinion mounted on the motor's shaft spins two magnetic clutches in opposite directions. When a DC current is applied to the clutches, the magnetic powder becomes more viscous and grips the horizontal shaft, which turns the pulley system and moves the carriage or arm.

b. Introduction to our System

Our design can be split into four main interfaces as shown in Figure 1.3-2. The sensor interface (1) obtains data on the position of the carriage and arm from the RAMAC. This information is sent to a controller interface (2) that can then control how to move the carriage or arm through the actuator interface (3), which translates the controller's commands into high power inputs to the RAMAC. The system can also be controlled from a remote site by using our Web-based control interface (4). The user interface is simply a PC equipped with the proper software.

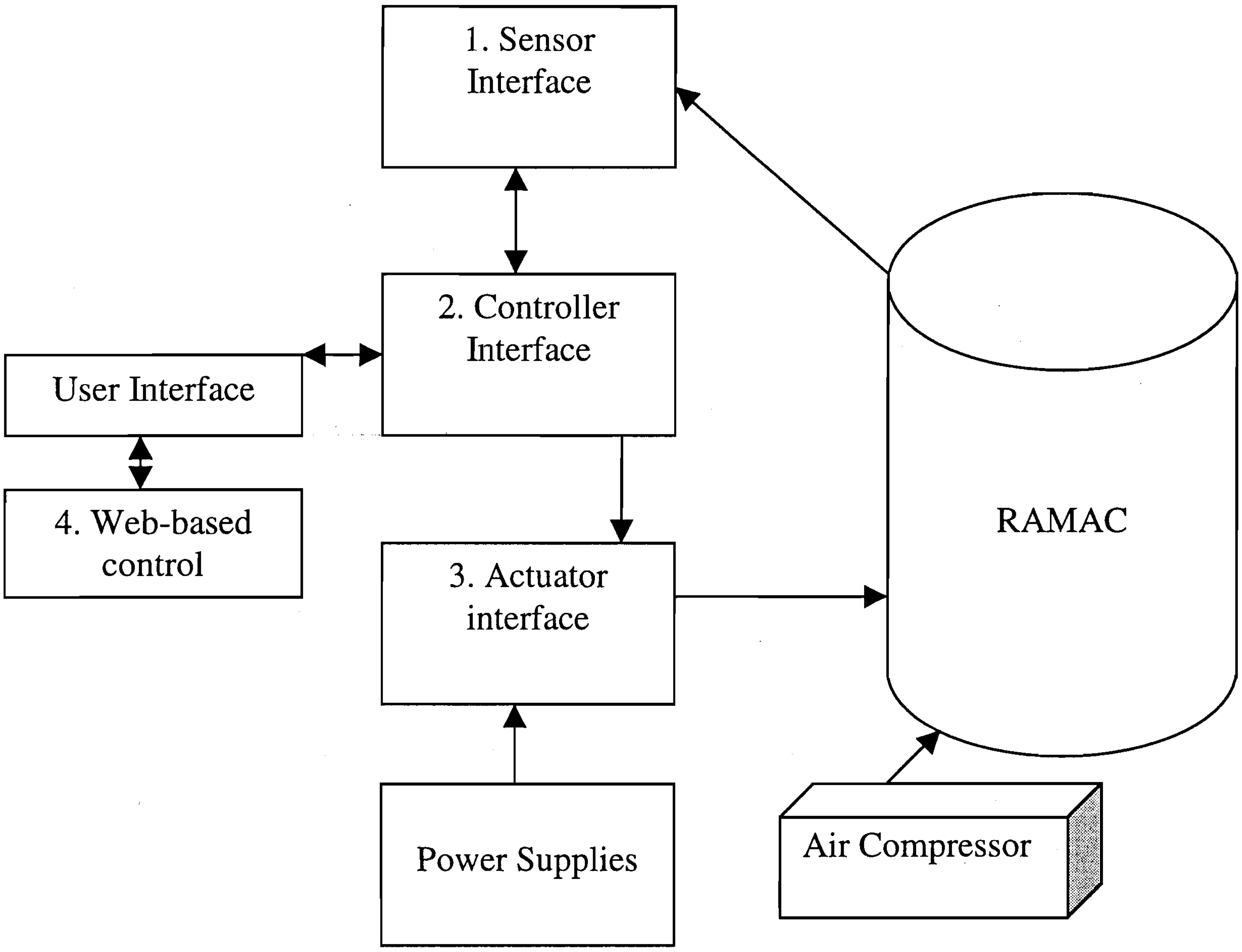


Figure 1.3-2: Diagram of RAMAC's 4 main interfaces

The physical components which comprise our system are: the cabinet, which carries all the controlling units, the hard drive unit, which is the historical device present prior to our project, and the air compressor, which is detached from the two but supplies air pressure to the solenoids. These components are shown in Figure 1.3-3.

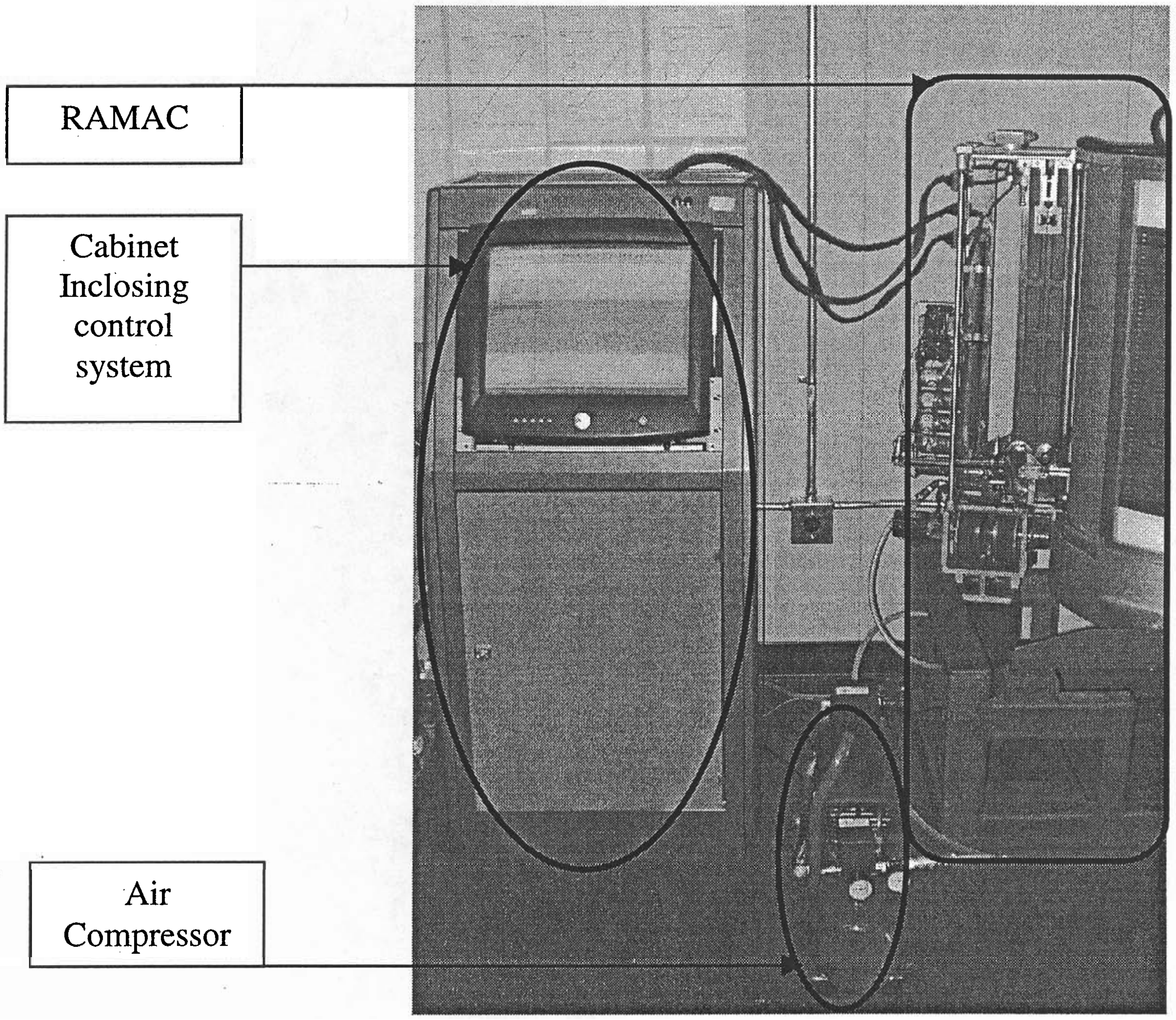


Figure 1.3-3: Picture of all the physical components that make up our system

Inside the cabinet (Figure 1.3-4) is: a PC, a microcontroller, a switching circuit and Voltage supplies. The PC serves to program the microcontroller, which then takes over the automation.

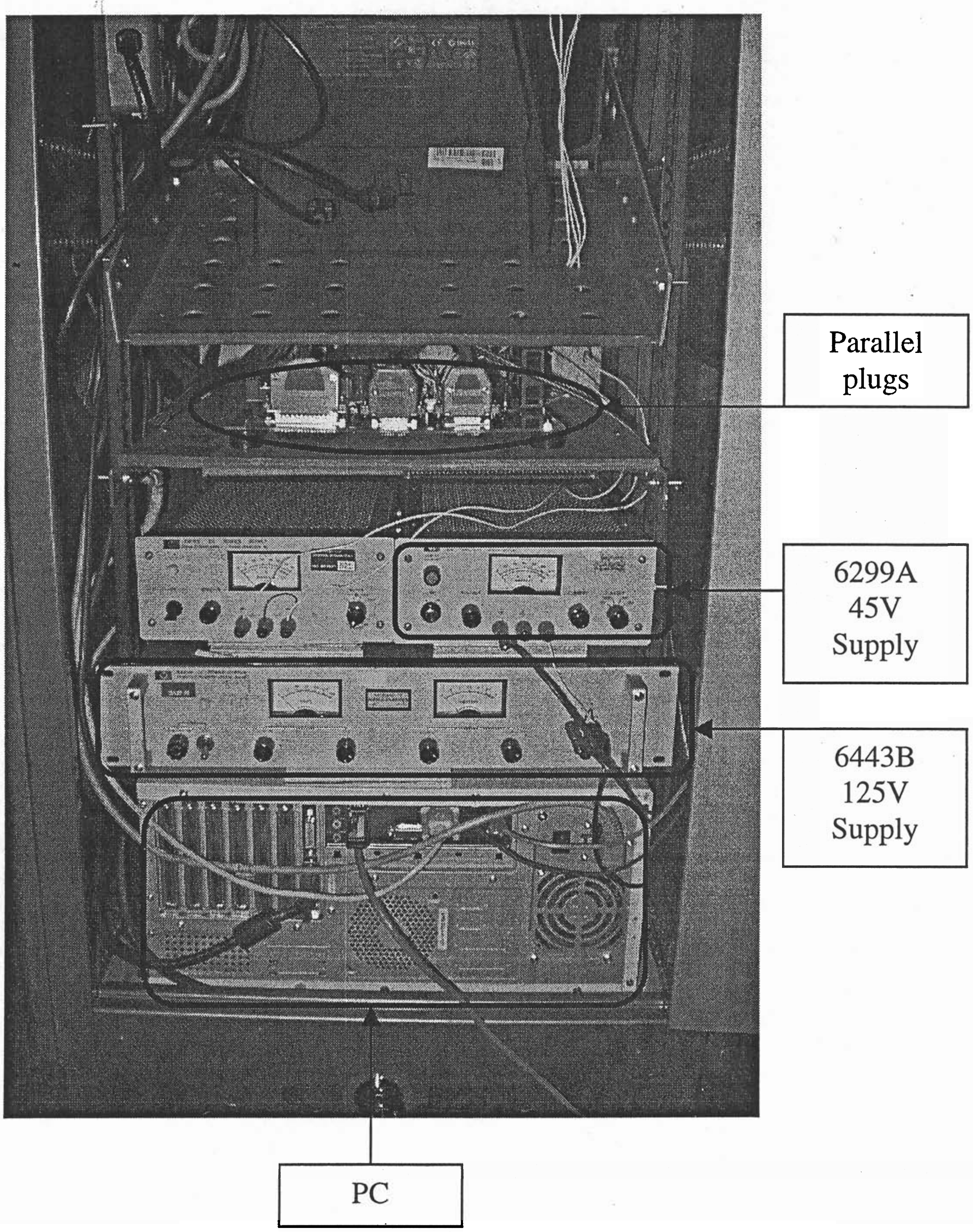
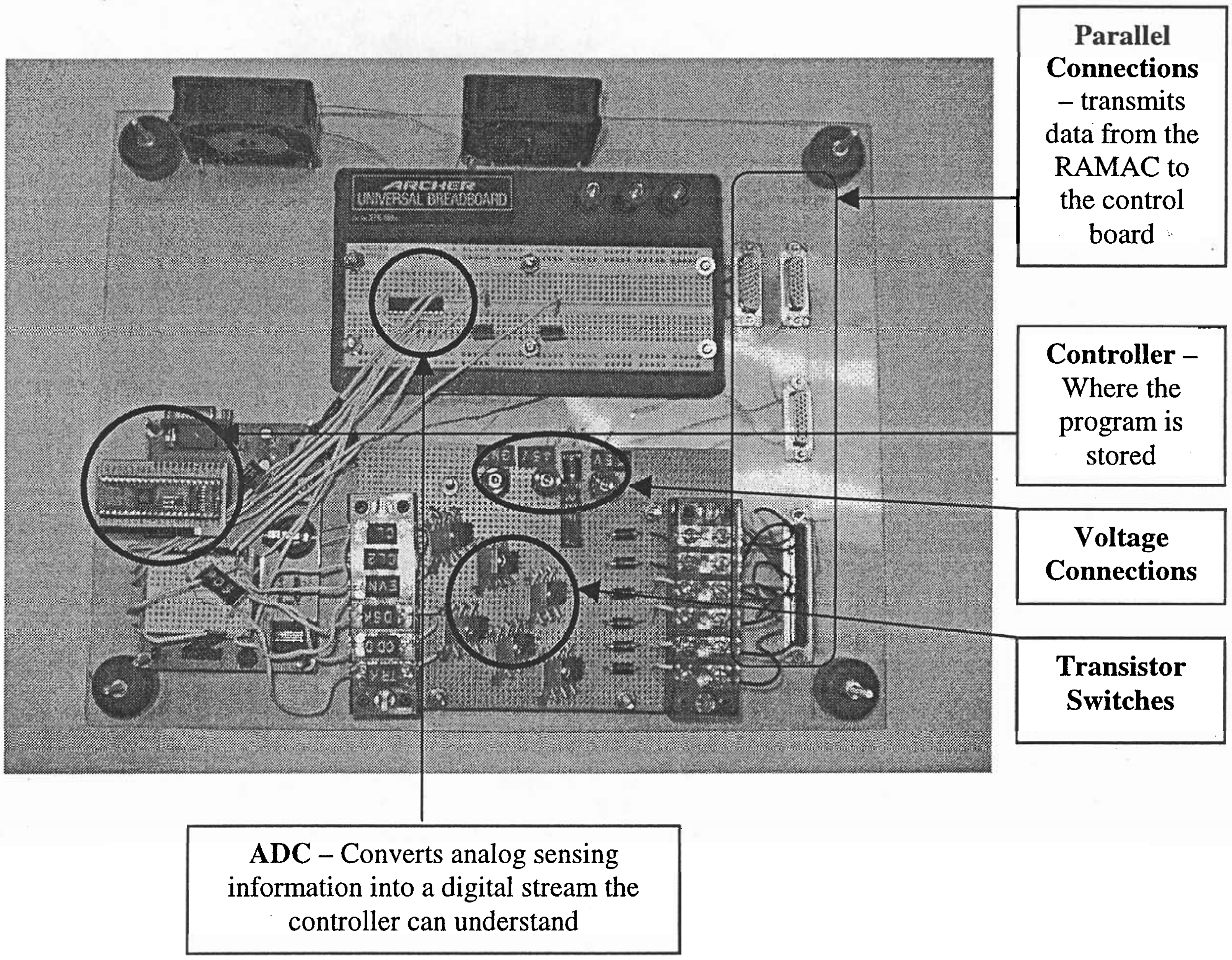


Figure 1.3-4: Image of the contents inside the cabinet

The control system we have designed can be divided into three main subsystems: The actuator interface, the sensing interface and the controller.

The actuating interface is comprised of transistor switches. Since the controller can only supply low voltages, the switching circuit is used as a medium between the commands generated from the controller and those supplied to the clutches/solenoids.

The sensing interface is comprised of three components: a resistor strip which lines the vertical way of the carriage provides analog voltage values; a potentiometer which is geared to the horizontal way provides similar values and an Analog to Digital converter which converts this analog sensing information into a digital stream the controller can understand.



Parallel Connections – transmits data from the RAMAC to the control board

Controller – Where the program is stored

Voltage Connections

Transistor Switches

ADC – Converts analog sensing information into a digital stream the controller can understand

Figure 1.3-5: Picture of the control board, which controls our entire system

The controller is where the program is stored and it is the decision maker of the system. The controller's memory is programmed using software provided by Parallax (The controller manufacturer), the language is PBASIC. Once the user enters a destination, the controller uses a sensing interface to find out where the carriage/arm is. Once the direction of travel is determined, the controller orders the high power switches to draw current through the clutches, which drives the motion. Once the destination is reached, the controller orders the appropriate solenoid switch to turn on, which stops the carriage or the arm in the right location.

1.4 Reader's Guide

The following 3 chapters detail the design of the three major subsystems: The actuating interface, the sensing system and the controller interface. Detailed drawings, graphs, pictures and diagrams are inserted throughout these chapters but can also be referred to through our table of contents. A user manual for our system is included in the appendix, which is at the end of this document. The user manual clearly and simply explains the physical layout of our system and gives directions to anyone who wishes to activate it. The source code in hard copy format is located in the appendix and is also included on the floppy disk that is in the envelope behind the front cover. Data sheets for all the hardware components used are also located in the appendix. Any drawing, diagram or picture that we believe are relevant but that did not fit in the body chapters are located in the appendix with short explanations. These diagrams and pictures can also be referred to through our table of contents.

Chapter 2: Actuation Subsystem

2.1 Why Controllable Motion

The first step was to obtain slow and controllable motion of the carriage. Slow motion was particularly important because we were not sure how fast our control system would be able to process information. Moving the system at slow speeds seemed like the logical place to start. Our first target speed was five inches per second. This speed was chosen because it was slow enough for us to fine-tune our control system before increasing the speed. This way, we knew our processing power would not the limiting factor.

Obtaining controllable motion depended on providing controllable torque levels from the magnetic powder clutches. The original system applied 120 volts directly to the either of the magnetic powder clutches, depending on the direction of motion. This magnitude of voltage caused the carriage to move at velocities of over 100 inches per second. At this speed, the carriage would move from one disk to the next in less than 4ms, resulting in a minimum sampling frequency of 500 Hz. After doing a little research on our chosen microcontroller (discussed later), we realized that this frequency was beyond our capabilities. Our options for controllable motion consisted of both analog and digital techniques.

2.2 Generating Controllable Motion

Our first method to generate low speeds was to decrease the rotations of the magnetic powder clutches by decreasing the power delivered from the AC motor. For example, instead of applying 110V AC, we would decreased the supply to 60V AC. The first drawback of this technique was that it required expensive equipment which we could not afford. The second was that although we would move the carriage more slowly, this method did not necessarily give us control of the speed, since the clutches rotation was permanently set at a 1:1 ratio with that of the pinion.

We then looked into applying a lower voltage to the clutches. We expected a linear dependency on torque and applied voltage. However, as one can see from the graph Figure 2.2-1, the voltage-torque relation was far from linear. We see from the graph that virtually no torque was produced when less than 25 V were applied to the clutches. However, for values of 25 V and higher, the clutches were essentially in full grasp, delivering high levels of torque to the output shaft. We found that we were unable to create a torque response in the controllable range through this analog technique.

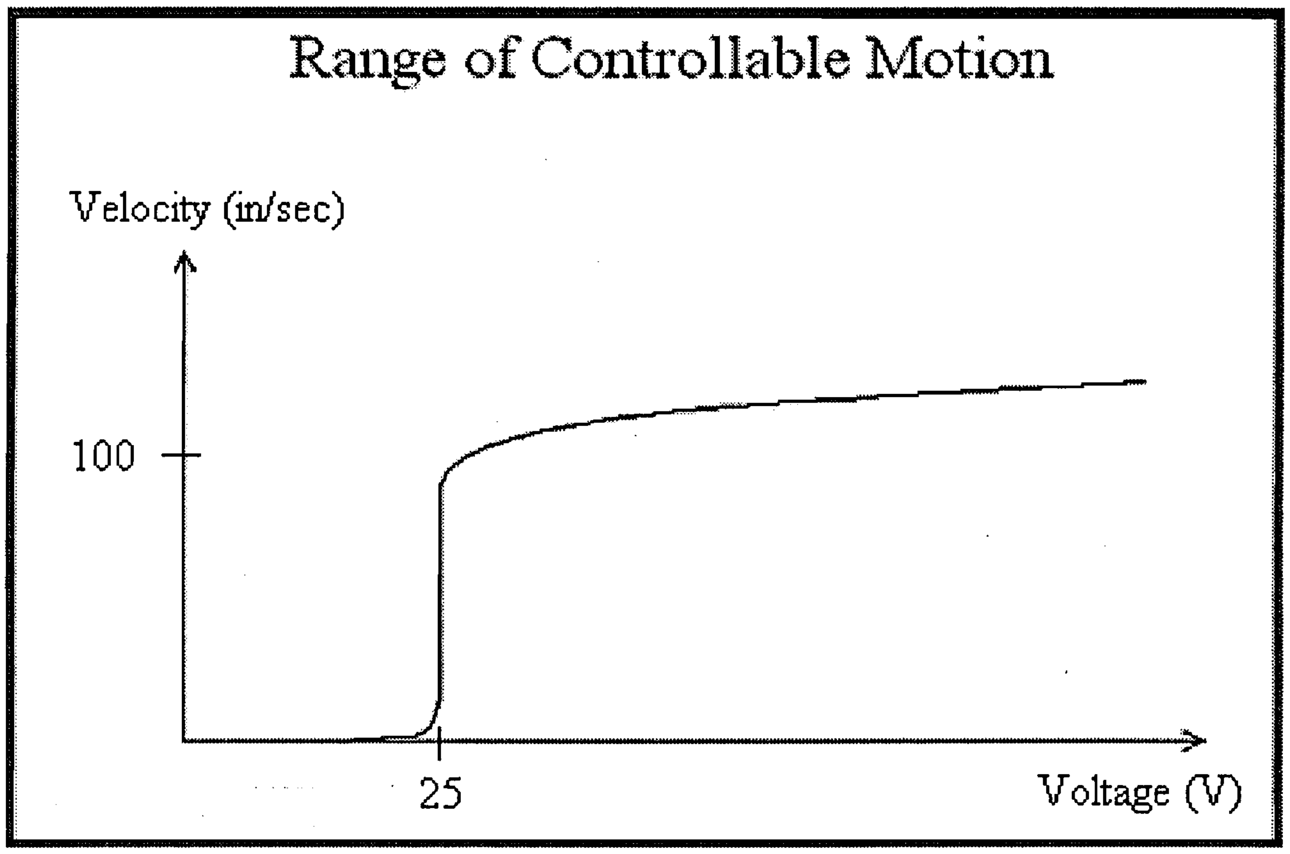


Figure 2.2-1 Uncontrollable velocity for minimum applied voltage.

We finally tried a digital method to produce a torque response in the controllable range. The method we employed was pulse width modulation (PWM) shown in Figure 2.2-2. With PWM, we are pulsing voltages on and off to the clutches. This method, commonly used in DC motors, provides a direct relation between the pulse frequency and the amount of current drawn through the clutches.

The PWM parameters we chose to obtain controllable motion were 125 V and duty cycles between 25% and 50%. These values were found through exhaustive iteration to be optimal values. Voltages far below 125 V required virtually full duty cycles for any clutch response, greatly reducing the versatility of the clutches. With 125 V, we were able to produce a wide range of velocities depending on the duty cycle we administered and remain in the controllable range as shown in Figure 2.2-3. This method was employed for both the vertical and horizontal motion.

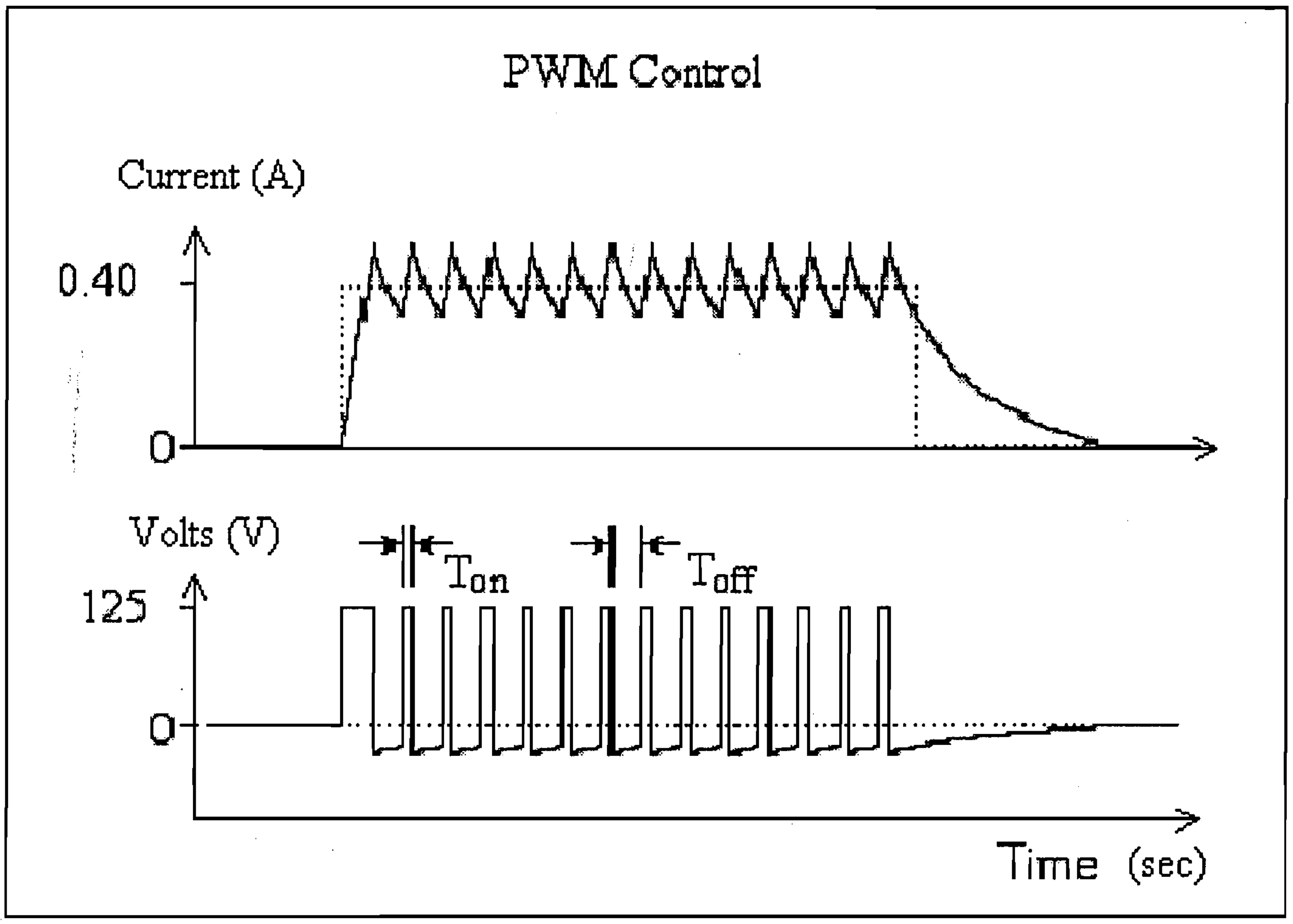


Figure 2.2-2 Pulsing Voltage to create a desired current.

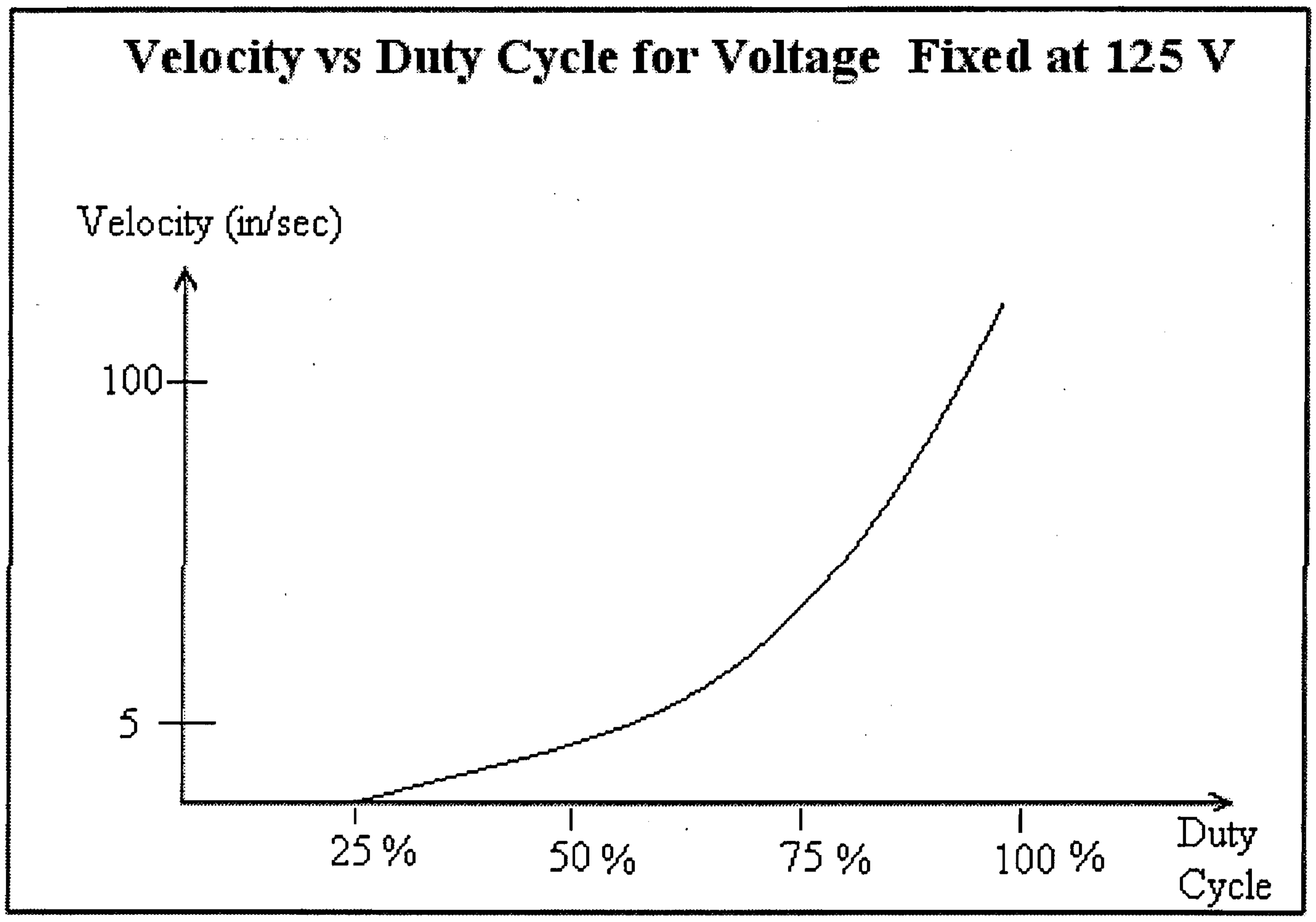


Figure 2.2-3 Controllable velocities for carriage motion.

2.3 Basic circuit

To enable PWM control, a switching circuit was required. A hardware interface was necessary between the microcontroller and the clutches since we could not switch these high voltages via the basic stamp alone. This hardware interface needed to use a 5V input from the controller to switch 125V to the load, drawing currents of up to 500mA.

The first Option we decided to explore was to use a prefabricated motor driver. We decided to use the LMD18200 from National Semiconductor because many student projects had used it in the passed and documentation on applications was readily available. This motor driver could withstand very high currents (3A continuous) but the maximum supply voltage it could handle was only 60V. As discussed earlier, our design required 125V to be pulsed at the clutches. The driver simply did not meet our requirements.

The second option was to purchase a pre-built motor driver board. Although these can be found to suit any application, the option was quickly ruled out because it was too expensive.

Our final decision was to build a simple transistor switching circuit as shown in Figure 2.3-1. Since our requirements were fairly simple, we felt that a "home made" motor driver would give us unmatched versatility and economic advantage. All that was left to do was to build a circuit which could switch 125V and draw a continuous 500mA safely.

The function of this circuit is very straightforward. The base of each switching transistor is connected to a pin on the microcontroller via a 700Ω resistor. When the pins are set high (approximately 5V), a small current is drawn through the base, this current drives a larger current through the load. The "Hfe" of the transistors is 25. The current through the load is thus 25 times that of the current through the base. The solenoid switches are identical to the ones used for the Clutches.

The solenoids are used to mechanically switch air pressure to the detents which lock the carriage or arm into place when the destination is reached. They will however never be pulsed, since we would like for them to be either fully engaged or disengaged. The transistors used were TIP47 NPN Bipolar junction transistors with a high sustaining voltage of 400V and a continuous collector current rating of 1Amp. We chose the TIP47 because it could sustain our high power requirements reliably and because they were readily available. Although the transistor's characteristics can hypothetically limit our switching frequency capability, the switching time of the TIP47 was so small that it did not create a bottleneck in our design. More details are available in the datasheet located in the appendix.

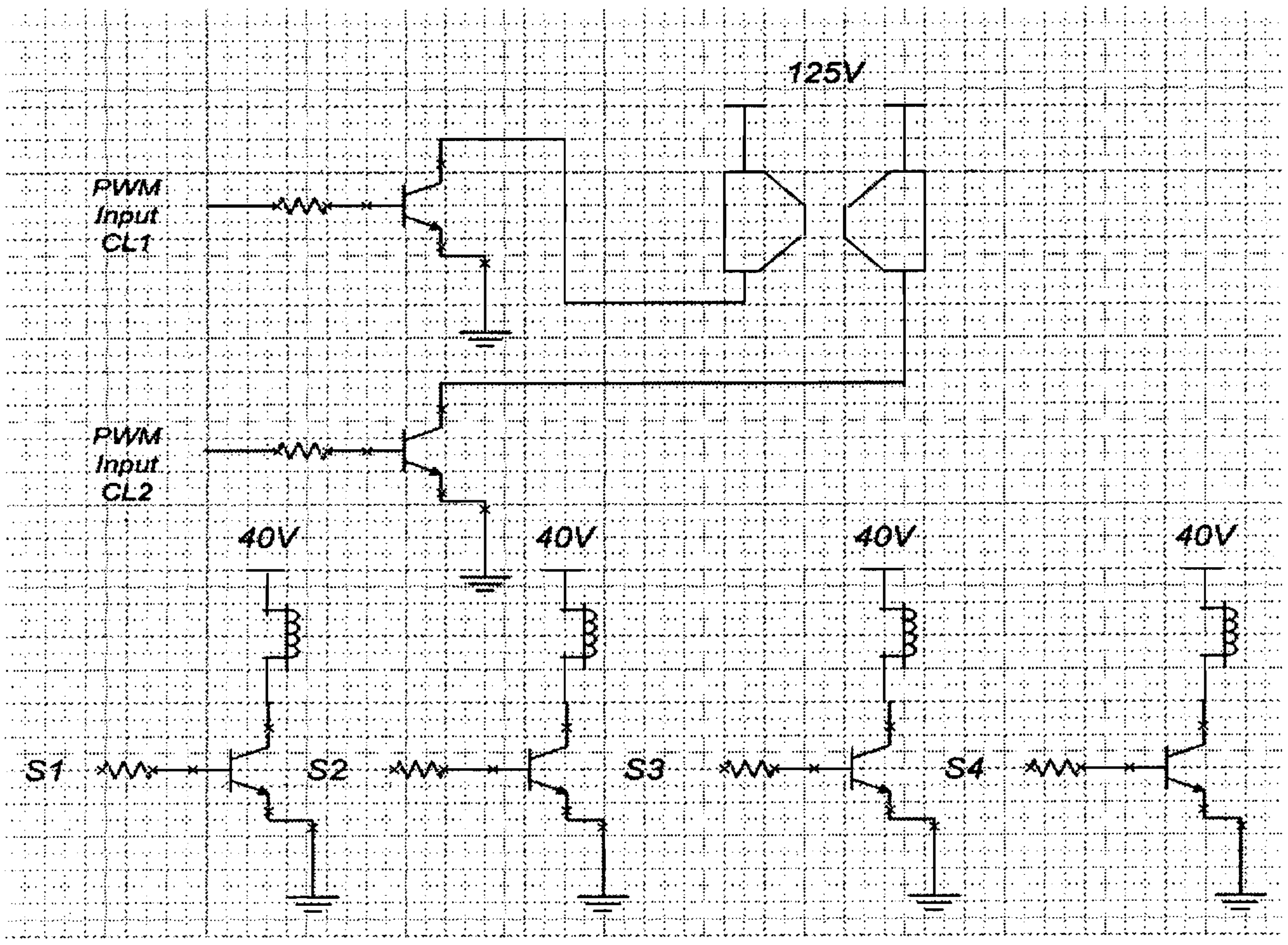


Figure 2.3-1 Simple transistor switching circuit.

2.4 Manually Actuated Test

The next step in our process was to test our circuit. Before using the microcontroller to pulse at the bases, we decided to run a manually actuated experiment. Physical switches were mounted before the base and a pulse train of an arbitrary duty cycle was applied before the switch depicted in Figure 2.4-1 below. When the switch was manually closed, current was drawn through the clutches, enabling us to test the hardware safely. If too much current were to be drawn and the carriage were to "slam" up or down, we quickly could open the circuit and prevent further damage. This experiment was very helpful; it helped us iterate a duty cycle that would drive the clutches to move the carriage slowly in both directions. We felt that slow motion was an optimal short-term solution, as it would make for slower sampling requirements during sensing.

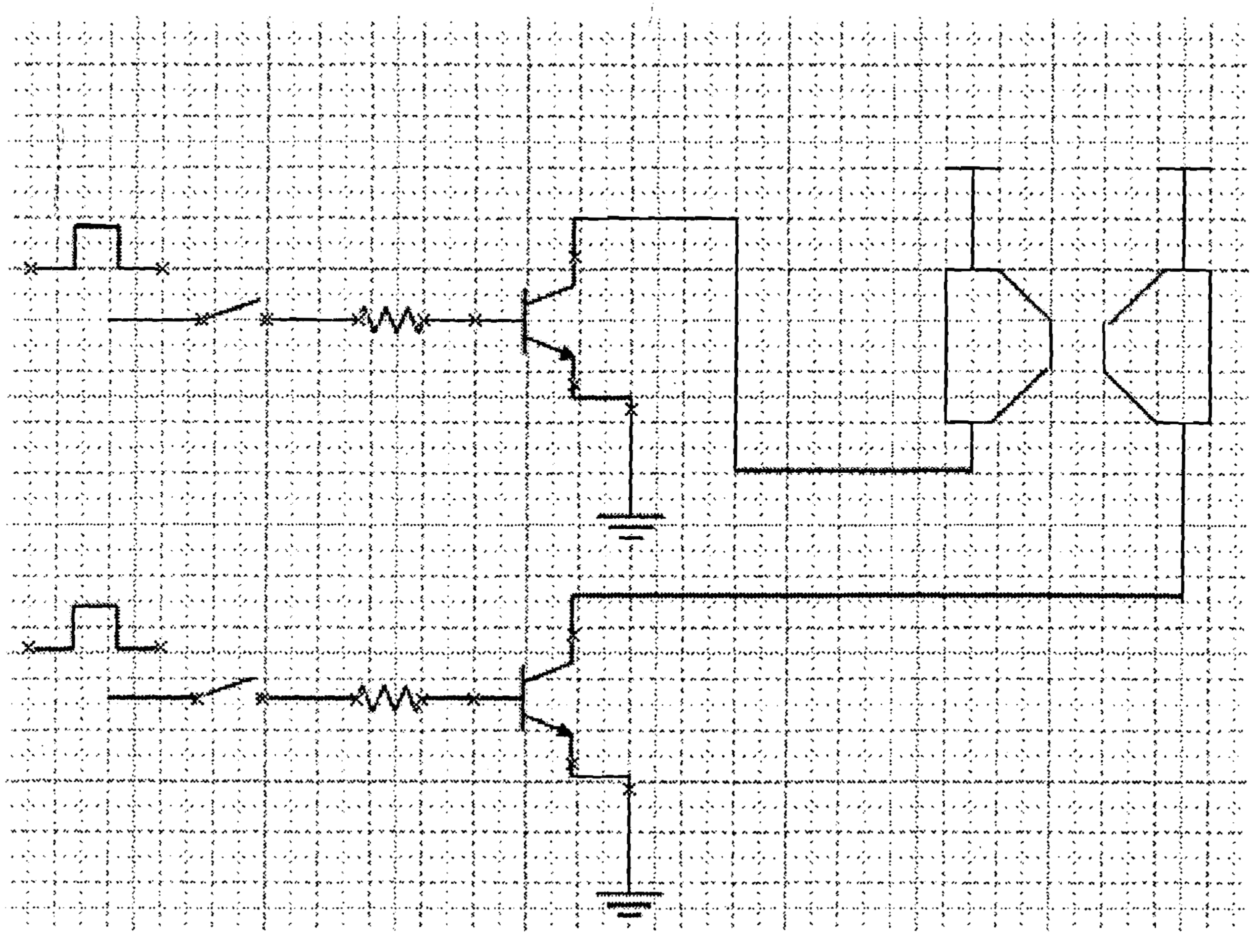


Figure 2.4-1 Circuit prior to improvements.

2.5 Improved Circuit

Although we knew we were headed in the right direction, our group had to take a few steps back in our circuit design. The back EMF that is caused when an inductive load is switched on and off can cause large voltages across the load. This build up of energy cannot be released without some sort of draining system, and can in the worst case cause arching and reverse currents into the pins. By installing these Kickback diodes, we protected our circuit from such a failure. The installation of kickback diodes is a standard technique in circuit protection; we felt it was a necessary addition to ensure the longevity of our design. These additions are shown in Figure 2.5-1.

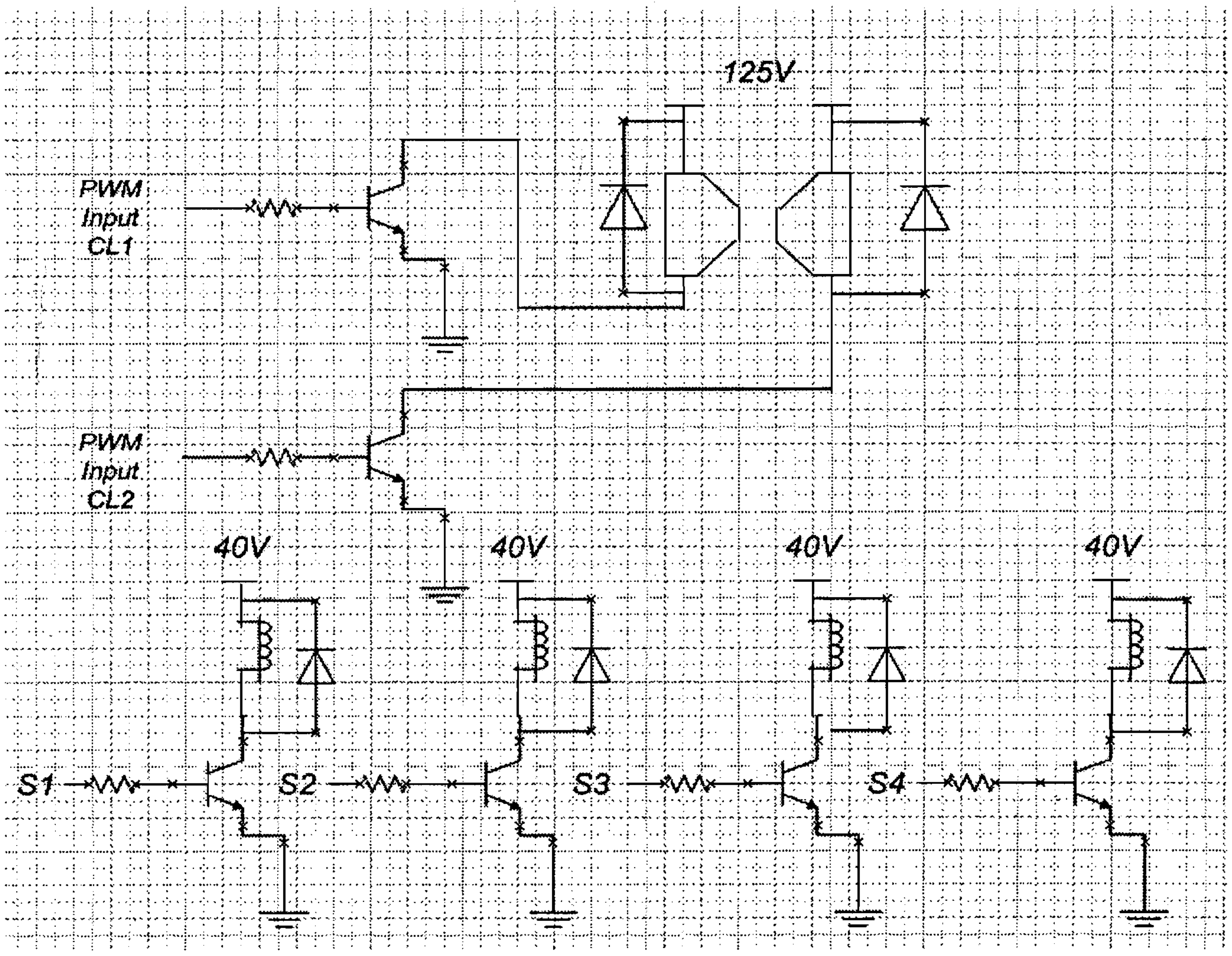


Figure 2.5-1 Circuit after improvements

Chapter 3: Sensing Subsystem

The sensing mechanism is comprised of two separate components. The vertical location is sensed via a resistor strip mounted on the vertical path and the horizontal location is sensed via a potentiometer geared to the arm. Voltages were applied across both of these devices and the values from the wipers represented disk and track addresses. These voltages were then converted into a digital stream through an Analog to Digital Converter in order for the controller to interpret it.

The requirements for our sensing system were to achieve accurate sensing. The system had to be accurate enough to sense 100 discrete locations between 0 and 5V. This was easily achieved since even an 8bit ADC, which was considered a low-resolution device, could output 255 discrete values.

3.1 Track Sensing

The RAMAC uses a 25 KΩ potentiometer to provide a way of indicating the location of the access arm. The potentiometer is mounted on the carriage but is geared to the access arm as shown in Figure 3.1-1.

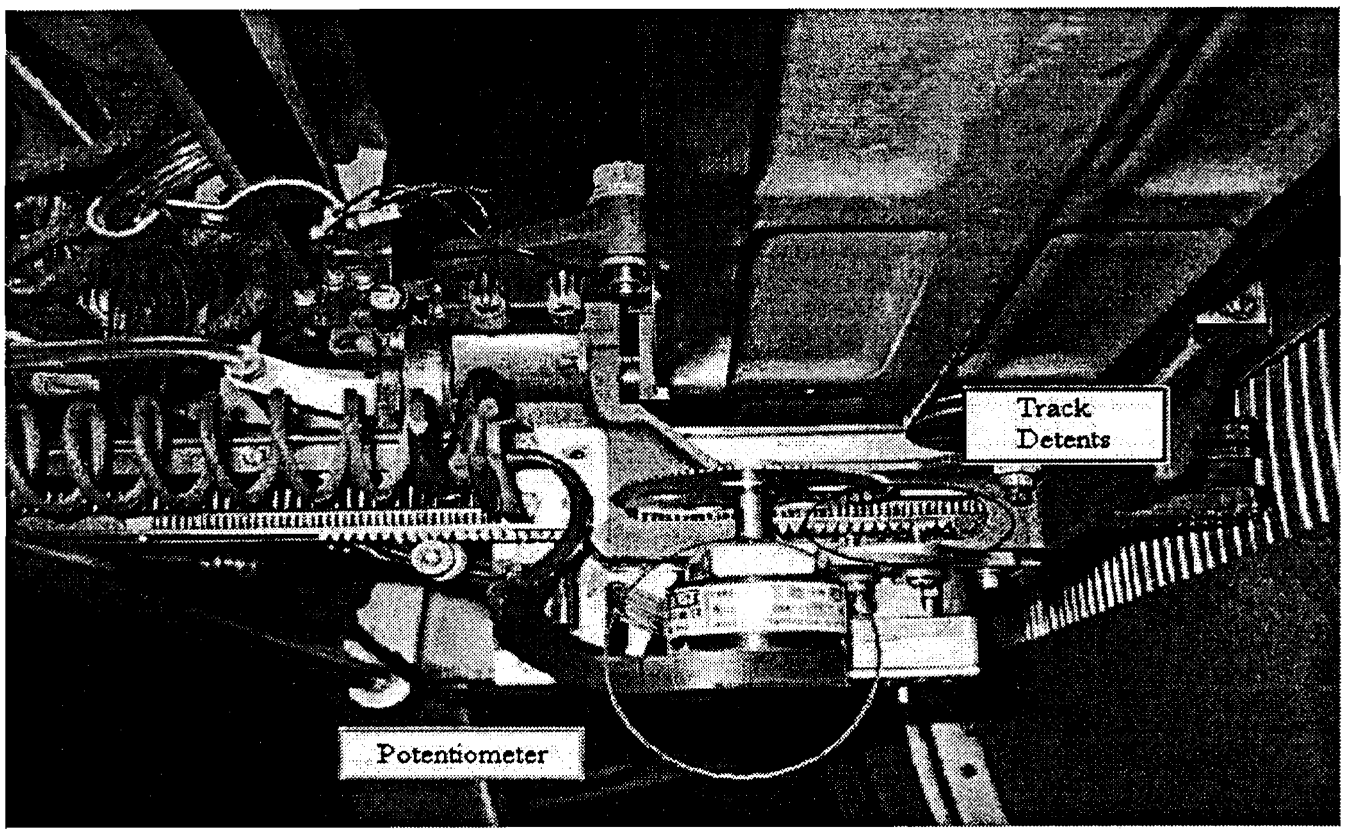


Figure 3.1-1: Track Sensing Potentiometer

As the arm moved in and out the gear rotated and the wiper moved to higher or lower resistance points in the potentiometer. The original design used a 150 V floating supply across the extremes of the potentiometer. This voltage was applied through taps which were installed at these two extremities. The desired track destination was grounded, the wiper would then move until it would not longer detect a voltage, which meant that it was at the proper location. The direction was determined by the sign of the voltage sensed. Because the potentiometer was not big enough to tap every track, a tap was installed at every 20 tracks. The system would differentiate between intermediate tracks by using an external voltage divider network connected between the each tap. An additional tap was installed in the wiper position which corresponded to a fully retracted arm. This was used as a "flag" in the system which indicated that the carriage could be released without damaging any of the disks[1].

Our preliminary task was to test the potentiometer was to find which pins corresponded to what tap on the potentiometer. Also, we wanted to test for the linearity of the resistor strip.

The first step was to record data on all 9 pins. We decided we would measure the resistances across the pins using an ohmmeter. While measuring the resistance across two particular pins, we moved the head in or out. If the resistance value changed, one of the two pins was the wiper. We used a brute force method of testing every possible combination of the pins. With a little bit of time and patience, we were able to generate the data in Table 3.1-1. Data for Pin 2 is omitted because it was the wiper.

After finding the wiper, we found the corresponding tap number for each pin. This was done by connecting the wiper pin and an additional pin to the ohmmeter. We then moved the head in and out the find the lowest resistance value possible (a few hundred Ohms). When the ohmmeter read this lowest value, we knew that the wiper and the other tapped pin were at the same location, we then could see what this location was. Our results are shown in Figure 3.1-1 and Figure 3.1-2 below.

---

[1] IBM RAMAC maintenance manual, pg. 68

	P4	P5	P6	P7	P10	P11
P4		16	7.8	4.1	4	8.4
P5	16		8.4	19.8	12.2	24.1
P6	7.8	8.4		11.6	4	16
P7	4.1	19.8	11.6		7.8	4.6
P10	4	12.2	4	7.8		12.2
P11	8.4	24.1	16	4.6	12.2	

Table 3.1-2: Resistance Values (KΩ) between contact pins

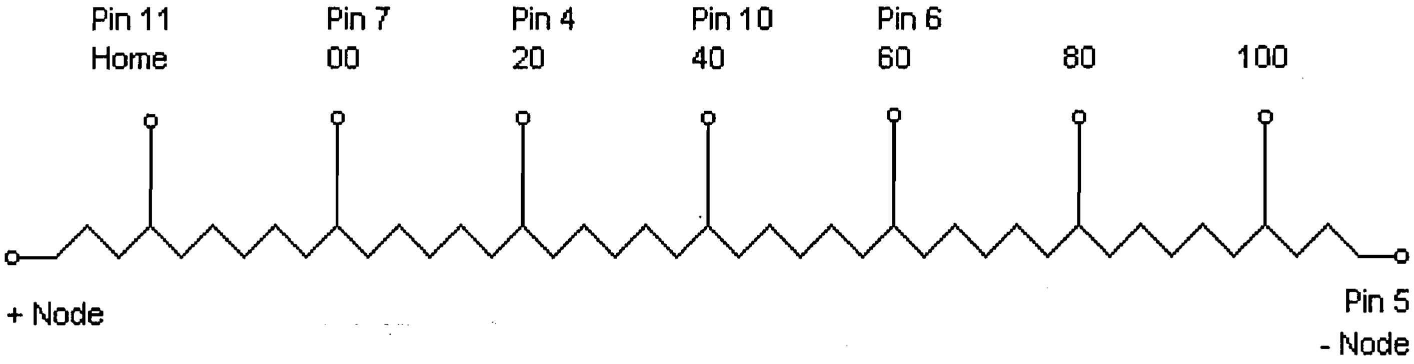


Figure 3.1-3: Diagram of Potentiometer

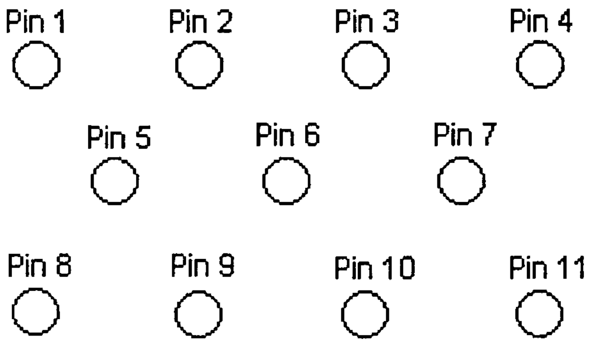
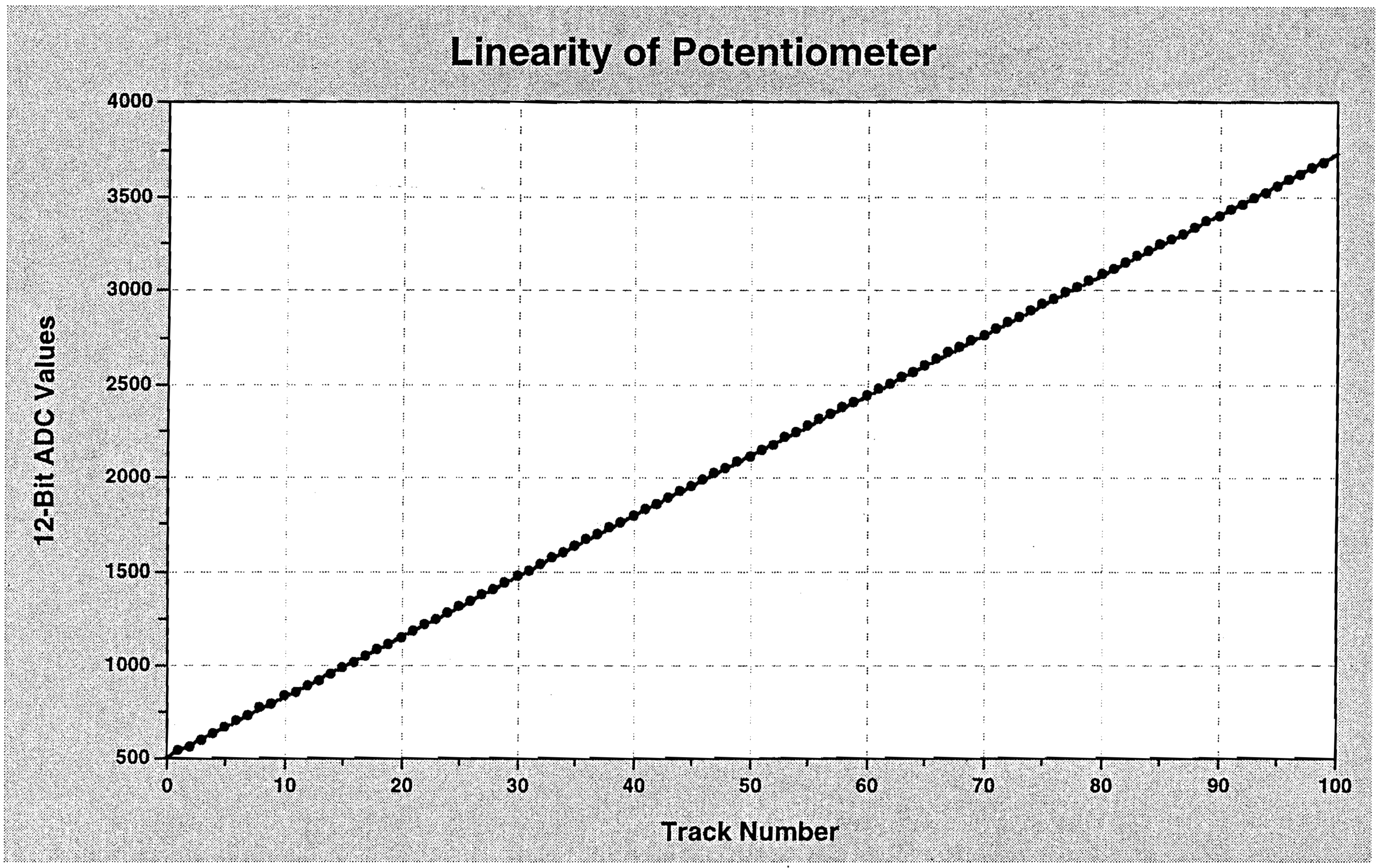


Figure 3.1-4: Orientation of Pins

Upon taking a closer look at the data, we were able to make several observations. First, we found that pins one, three, and eight were not conducting. They are the pins corresponding to the positive node, track 80, and track 100. In addition, we found that the gear that moved the wiper on the potentiometer has been offset by a several teeth. This offset a track pin by approximately 2 tracks. For example, pin 4 originally had its origin at track 20 but today, because of the offset, pin 4 resides at approximately track 22.

The constant resistance step between pin 5 and pin 11 shows that the potentiometer is fairly linear. Moving from pin 5 to pin 11, we found that we had a resistance step of 3.8 KΩ between each tap.



Linear Fit: y = 511.04606 + 32.176118x **Correlation Coefficient**: 0.9999911

Figure 3.1-5: ADC output Vs. Track Number

3.2 Disk Sensing

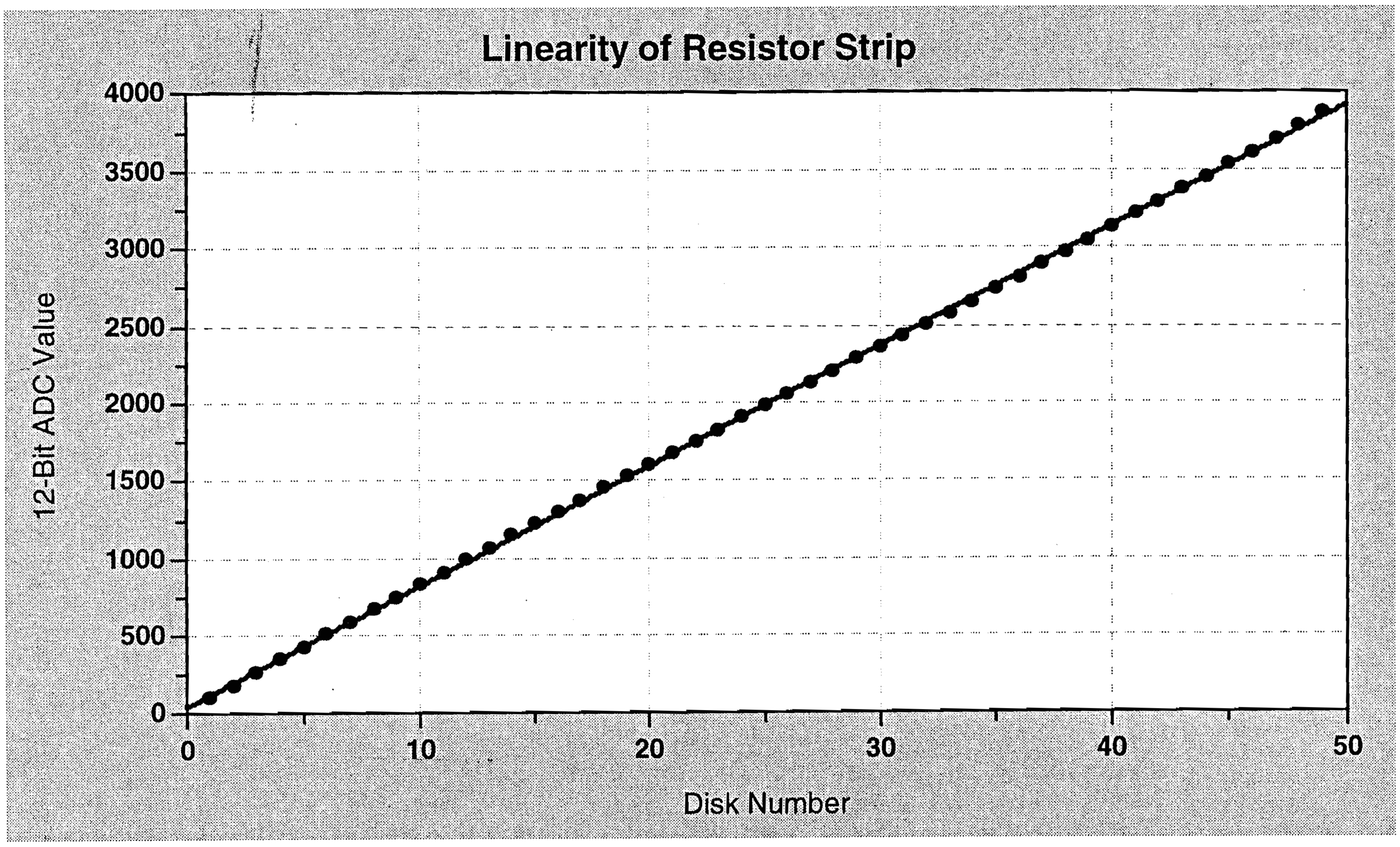
The disk resistor strip operated on the same principle as the track potentiometer. The only difference was that the disk array was physically big enough so that each one of the 50 disks was tapped. The desired destination disk was then grounded and the system would drive the clutches in the appropriate direction until the wiper, which was mounted on the carriage, would read a zero voltage value, which meant that it was at the proper location.

Testing the resistor strips linearity was fairly easy. We knew which pin array corresponded to that of the 50 disk taps because it had been documented during previous work on the RAMAC. We used an ohmmeter much in the same fashion as during the potentiometer test in order to determine the pin corresponding to the wiper. Once the wiper was found, it took some patience and perseverance to find every tap's corresponding disk number.

The RAMAC's original methods for sensing were based a purely analog system. The voltage read by the wiper was directly fed into a system of relays and switches. The amplitude and sign of that voltage would determine the direction and speed of the motion.

One option in our design was to recreate this analog circuit using modern analog and digital IC's in contrast with the original vacuum tubes and relays. This method, although feasible, was not adopted because it required quite sophisticated knowledge of analog design, which neither one of our three group members obtained. Furthermore, we wanted to add a "fresh" aspect to the restoration of the RAMAC. We felt that using a different method in sensing would make our design more exotic and interesting.

The method we decided to implement was actually simpler than the original one. We decided to apply a small voltage (5V) across the extremities of the potentiometer and resistor strips and read the analog value corresponding to the current location directly off of the wipers. This analog value was then converted into a digital stream by an Analog to Digital Converter (ADC) to make the information readable by the microcontroller. Because the potentiometer and resistor strip were so linear, a simple linear fit equation could be used in the code in order to assign a disk/track value to the binary output given by the ADC. The 12 Bit ADC values for all the track and disks are depicted by the following two graphs, Figure 3.1-5 and Figure 3.2-1.



Linear Fit: y = 46.403265 + 77.521441x **Correlation Coefficient**: 0.9999121

Figure 3.2-1: ADC output Vs. Disk Number

3.3 Issues in noise

As can be seen from the previous chapter, the linearity of the potentiometer and the resistor strip did not pose a threat to arriving at accurate disk and track locations. When stationary, the voltage levels measured from the potentiometer and the resistor strip were precise and repeatable. However, when any motion was introduced to the carriage or magnetic head, we faced considerable amounts of noise in this output voltage. Recorded values of noise ranged from 500 Hz up to well beyond 10 MHz.

When the magnetic head was in motion (track-to-track motion), we recorded noise spikes on the wiper voltage. These noise spikes lead to a signal to noise ratio (SNR) of approximately 20 dB. However, the track mechanics used two detents, as compared to the single detent used for the up and down motion. Because one detent was used solely for odd tracks, while the other was used solely for even tracks, a wider range of possible voltage values could be assigned to each track number. With the use of a 5 V reference across the potentiometer, there would be approximately 5 mV from one track to the next. For example, track 10 would have a value of 20 mV while track 11 and 12 would have a value of 25 mV and 30 mV respectively. With the use of even and odd detents, it would be impossible to engage on an even track with an odd detent, and vice versa. Therefore, it would be impossible to accidentally engage on track 11 when track 10 or track 12 were desired. With this arrangement, we were not concerned with the voltages between each track, but rather the voltages between every other track. This mechanical setup effectively doubles our SNR ratio. Even though the 20 dB SNR was not as high as we could have made it, we found it to be more than adequate to produce repeatable and consistent results.

When the carriage was in vertical motion, we recorded significantly larger spikes in noise. We found that we had an SNR of only 10.5 dB for the up and down motion. With a 10.5 dB SNR, we were unable to accurately move to a desired disk. And with only one disk detent, we were not able to increase our SNR as easily as we had for the track-to-track situation.

We found the source of this noise to be caused by parasitic induced voltages from the rotation of the magnetic powder clutches. Because these clutches were essential for any motion, we could not simply turn off this noise. Instead, we had to find a way around this noise issue.

In attempt to curb our problem with noise for the disk-to-disk motion and increase our SNR, our strategy was simple: decrease the noise strength or increase the signal strength. Our attempts to decrease the noise strength revolved around various filter designs. Our attempts to increase the signal strength included increasing the reference across the resistor strip, or splitting the resistor strip into smaller fragments, producing the same effect as increasing the reference voltage.

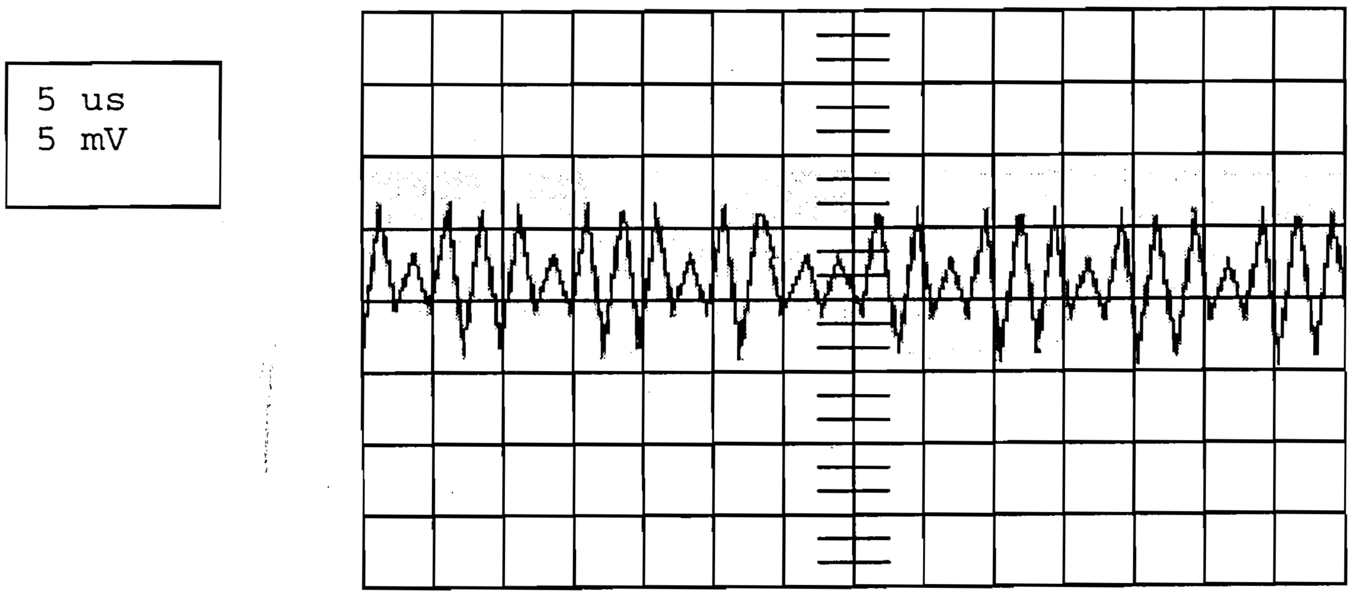


Figure 3.3-1: Noise seen with Oscilloscope

3.4 Noise solutions, options and tradeoffs

Our struggles with decreasing the noise strength involved building various passive and active filters. Because our noise started at 500 Hz, we wanted to use a low pass filter that would filter out very low frequency noise spurs. We were able to design and build an effective filter that highly attenuated all frequencies greater that 500 Hz. However, because the capacitance of a low pass filter needs to be greater as the cutoff frequency is lowered, we found ourselves having a time constant in the milliseconds range. The design we implemented is shown below.

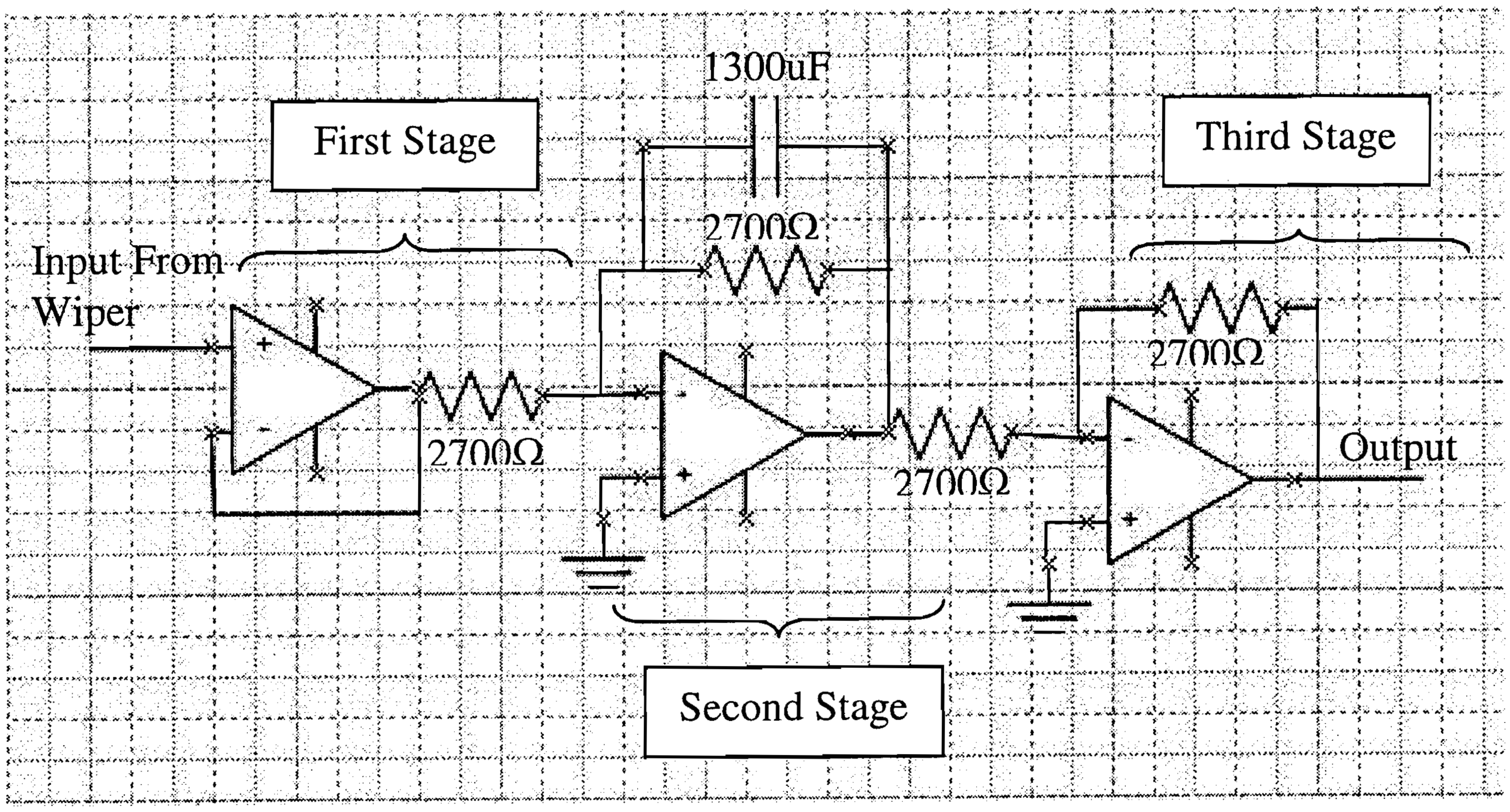


Figure 3.4-1: Filter schematic

The operational amplifiers used were National Semiconductor 741's. The first stage is a buffer, which prevents any current to be drawn away from the strip/potentiometer, the second stage is an inverting filter, where the large capacitance can be seen, and finally the third stage inverts the filtered signal in order for the output to be properly signed. With a time delay of the magnitude we faced, we found our disk-to-disk movement to be more inaccurate than without the use of a filter at all, since the sensing greatly lagged behind the motion of the carriage/arm. After some research and advising from Dr. Krishnan, who taught many analog courses at Santa Clara University, we gave up on this option. Designing a system to filter out low frequency noise without time delays proved to be more time consuming and complicated than our other option – increasing the signal strength.

The team now focused on increasing the signal strength rather than drowning out the noise parameter. Our first method was to simply increase the voltage reference across the resistor strip. Because we were currently using a 5 V reference across the resistor strip, we decided to instead use a 10 V reference and double our SNR. This did increase our signal strength, but we were unable to find an analog to digital converter that would receive voltages within the range. And because building a predictable attenuator proved to be an arduous task, we decided to increase signal strength through another means.

We found fragmenting the resistor strip into two smaller sections would give the same effect in signal strength as doubling the reference. What was once a 28 KΩ resistor strip, we fragmented into two smaller 14 KΩ resistor strips. This allowed for us to keep a reference voltage of 5 V without facing any difficulties of attenuating the signal.

The process of splitting the resistor strip required the middle tap of the 28 KΩ resistor strip to be able to take on values of 0 V or 5 V, while the top and bottom of the strip were constant at 5 V and 0 V respectively. When the carriage was in the top half of the disk stack, the midpoint tap would be set to 0 V to maintain a reference voltage of 5 V across 14 KΩ. When the carriage moved passed the midpoint onto the bottom half of the disk stack, the midpoint tap would then change to 5 V, again maintaining a reference voltage of 5 V across 14 KΩ. This method doubled our SNR with a minimal amount of hardware added to our system. We were then able to achieve accurate and repeatable results with the disk-to-disk movement.

3.5 Choosing a converter

Once a sensing method was established and the noise issue optimized, we started analyzing our options in analog to digital conversion. The voltages that corresponded to the different disk and track locations would mean nothing to the controller were they not translated into digital data. The first option we opted with was the TLC0831 by Texas Instruments shown in Figure 3.14-1. The 0831's are the most readily used ADC with the BASIC stamp and we were provided with ample documentation to get us started. It had 8 bits of resolution, a single input channel and a 32us conversion time. Unfortunately, we had to use two of these ADC's because of the two locations we hoped to receive sensing data from: the vertical strip and the potentiometer for the track to track motion. This was costly on our pin allocation and made us decide to explore more options. The second and final ADC we decided to utilize is the TLC2543 by Texas Instruments shown in Figure 3.14-2. The 2543 is a 12 bit ADC with a 10us conversion time and 11 internally multiplexed input channels, which proved to be very practical. One pin from the microcontroller could tell the ADC which input channel to convert at any given time. Using this chip, both the Disk and Track wipers could be connected to one ADC. Furthermore, the higher accuracy and speed of the converter would allow for more versatility in the sensing speed. It should be mentioned that both of these converters are successive approximation ADC's, which are typically slower than flash ADC's. This feature, however, was more than adequate for foreseeable sampling speeds in this system.

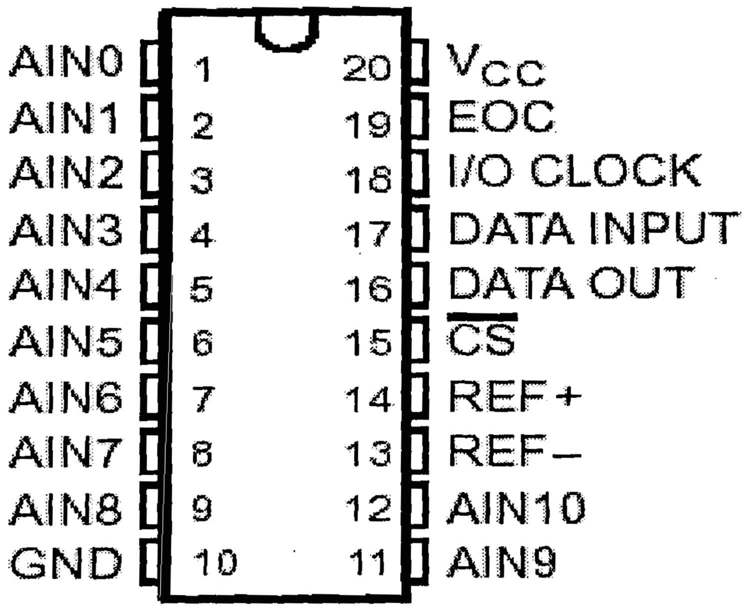


Figure 3.5-1: Layout of the *TLC0831*

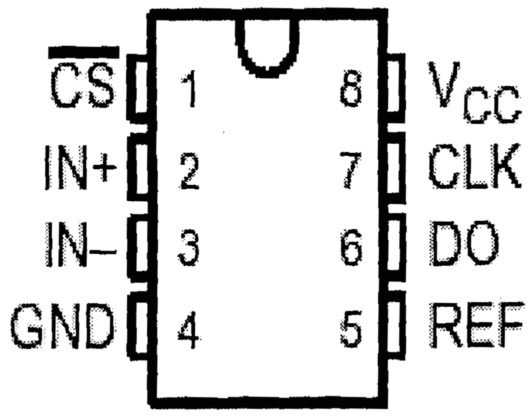


Figure 3.5-2: Layout of the *TLC2543*

Chapter 4: Control Subsystem

4.1 Choosing a controller: Parallax BASIC Stamp

After deciding to follow through with the digital method for controlling the clutches, we needed to select a microcontroller to create these pulses and later sense the location and possibly the speed of the carriage. However, because none of the teammates had any prior experience with microcontrollers, we were very cautious in selective a controller that best suited our needs and capabilities.

Potential controllers we were going to use were the ATMEL microcontoller, the Cypress Programmable System on a Chip (PSoC), and the Parallax BASIC Stamp. Although the ATMEL and the PSoC would have been more than adequate for our needs, they required steeper learning curves. We simply did not have the ability to fully submerge ourselves into all the capabilities of these controllers.

We eventually chose to go with the Parallax BS2P. We chose this controller because it met all of our processing needs. We felt that a 25 MHz processor would be adequate for the speeds we intended for the restored RAMAC to move. Also, the BS2P was readily available along with a plethora of good documentation and tutorials. Additionally, with the BS2P, we were able to purchase an accessory called the PWMPAL, which output up to four simultaneous PWM pulses in parallel with other processor computations.

However, the BS2P did carry various disadvantageous. Although we were comfortable with the 25 MHz processing speed, this controller had less computational power as compared to other available controllers. We would later find this to be our determining factor when we desired to increase speed. Additionally, the BS2P has limited memory and I/O pins, which eventually lead us to upgrade to the BS2P-40, a 40-pin version of the Parallax BS2P.

4.2 First integration

Once the sensing and the motion control were implemented, we were ready to give automation a first try. Our goals for automation were to integrate the controlled PWM motion with accurate sensing of the carriage and arm position information. The end product would be integrated motion and sensing through a user-friendly interface. Our access time goal was 10 Seconds, which as previously mentioned, is the time to traverse from the inner track of the top disk to the inner track of the bottom disk. We felt comfortable that with appropriate modifications, we could achieve this goal.

Figure 4.2-1 depicts the basic software protocol for our automation system. The program first prompts the user to input a disk and track address. Then the arm is retracted and the relative position of the carriage is sensed and the appropriate pulse is applied to the clutches. The sensing loop is then entered and when the destination is reached, the disk detent is engaged. At that point, the relative track position is sensed and the arm is moved in the proper direction. Again, once the proper destination is reached the track detent is engaged.

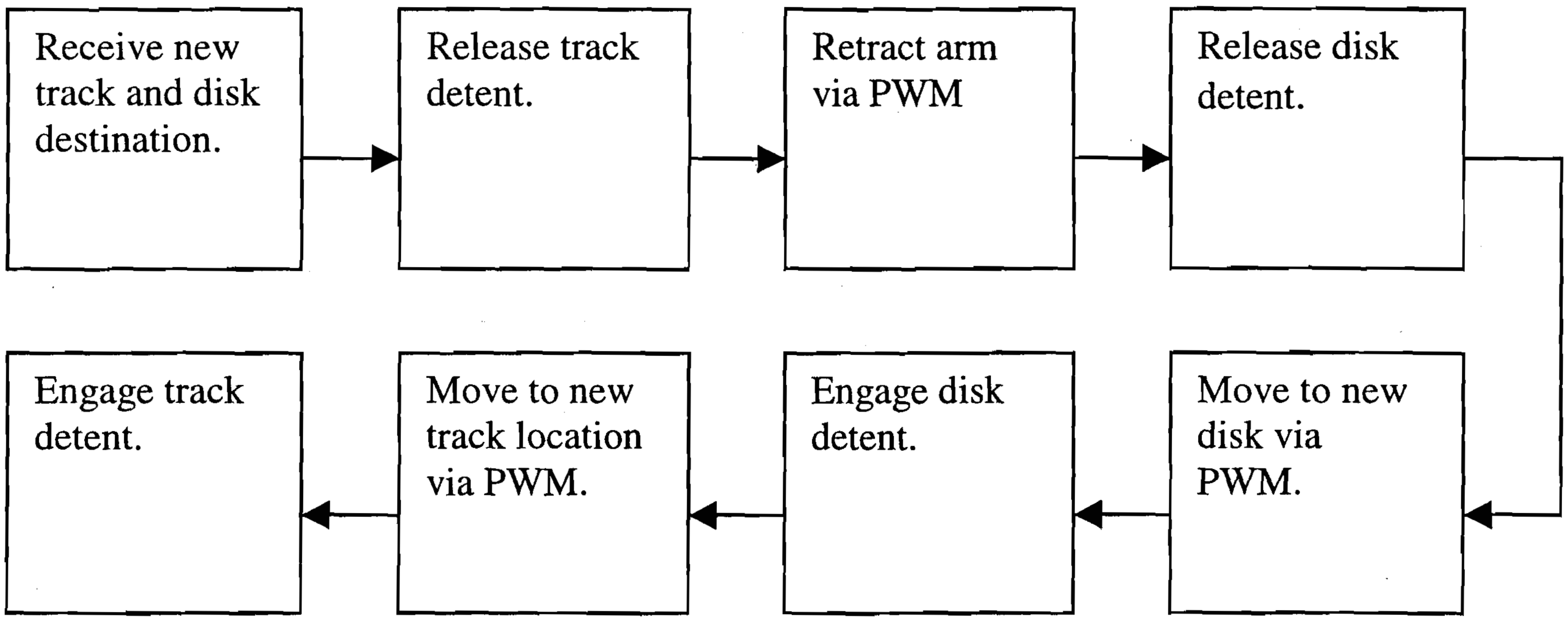


Figure 4.2-1 Basic software protocol

After a few weeks of testing and optimizing, the program seemed to work quite well. It would stop at the proper location quite often, missing the disk destination only about 10% of the time. This error was caused mainly because the carriage or arm would once in a while move faster than the system could sense. We did however realize that there was much room for improvement; the areas we decided to improve are discussed below. It should also be noted that when the carriage is being moved up or down, we are actually pulsing at the same clutch. The only difference being a lower duty cycle for the down motion, which counters the gravitational pull just enough to bring the carriage down slowly.

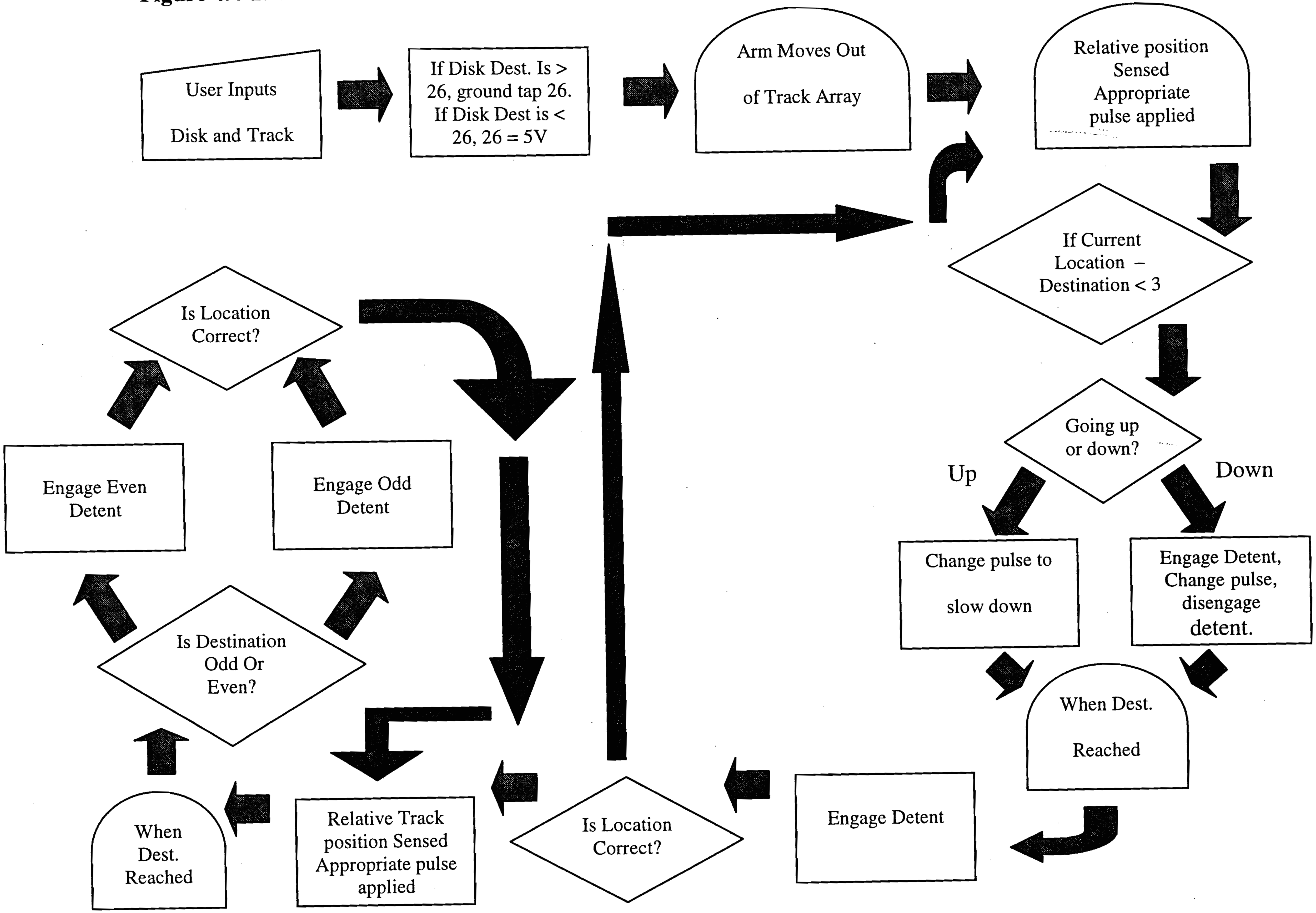
4.3 Improvements in automation- Error checking

Although the carriage would stop on the wrong disk or track once in a while, it would always eventually recognize its final position. For example, if the destination was disk 30 but the carriage stopped on disk 31, the debug screen would eventual read: “You are now on disk 31”. We decided that by implementing an error checking function, we could correct this error and make our automation error-proof. Although the carriage might take a while to arrive at its proper location, this greatly improved the accuracy of our system.

4.4 Improvements in automation- 2 stage PWM control

The program at this point was very accurate, but the motion was quite slow. The access times achieved were upwards of 25 seconds, which was more than twice our access time goal of 10 seconds. We decided that we could accelerate the carriage until it reached a relative distance from the destination; at this point, the pulses could be slowed down in order to sense more accurately the position of the carriage. After many trial and error tests, we came to the conclusion that the latest we could stop the carriage was when it was 3 disks away from its destination. The PWM-pal suffered some lag time in its reprogramming, which was no problem for the up motion; if the carriage dropped a little, it could quickly recover and come back up. For the down motion, however, we noticed that the delay in reprogramming caused the carriage to fall down too far and too fast. We decided to implement a different system for the down motion. When the carriage was three disks away from its destination, the detent would be quickly engaged, the PWM-pal reprogrammed, and the detent would then quickly be released. This proved to be a very effective method and saved us a lot of previously wasted time. Access times of around 10 seconds were achieved with this 2-stage PWM control, which met our original goals. The combination of the first program and these improvements resulted in the creation of our final program, which is depicted by the flowchart in Figure 4.4-1. Osciloscope graphs can also be seen in Figure 4.4-2 and 4.4-3.

Figure 4.4-1: Final automation flow chart



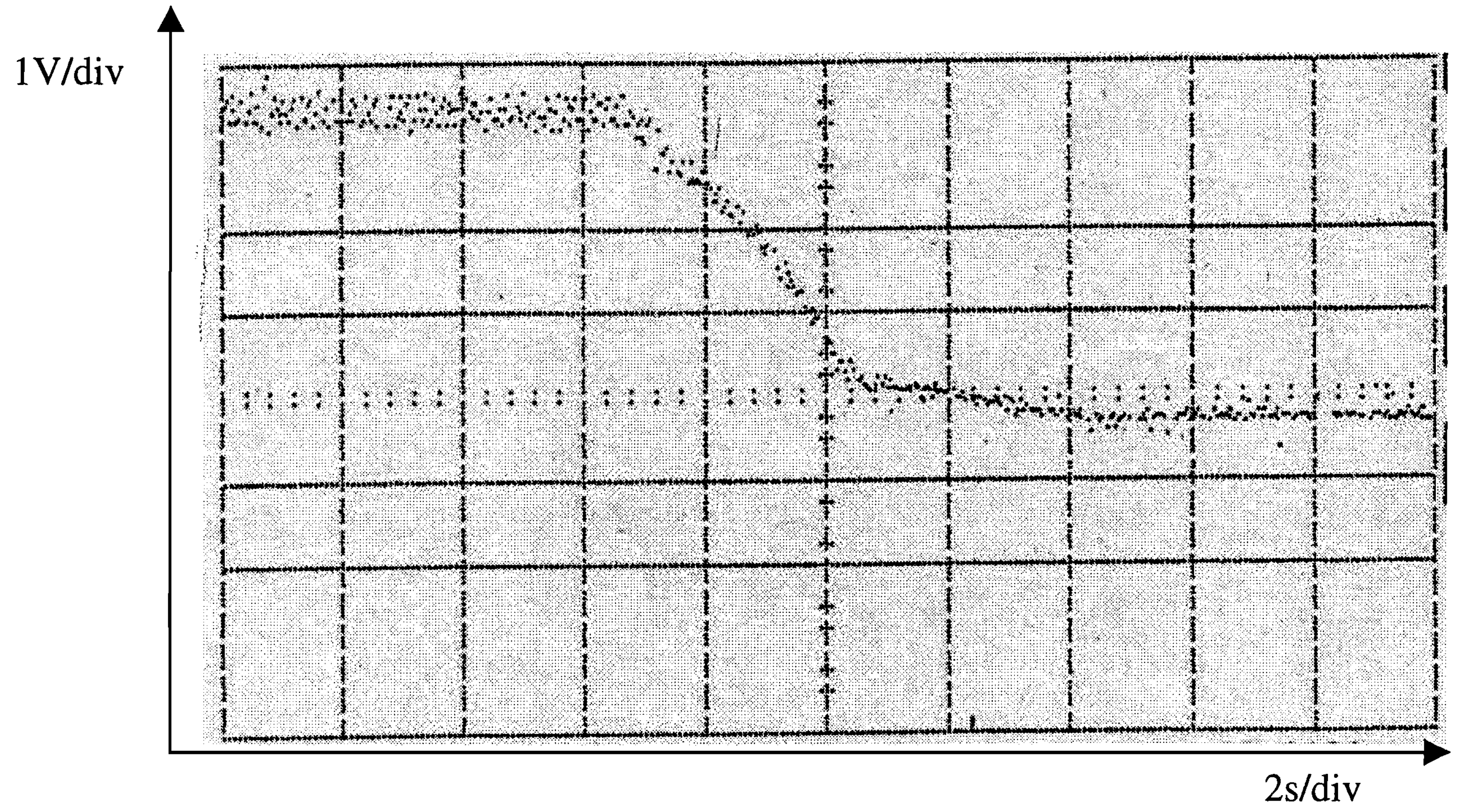


Figure 4.4-2: Voltage Vs. Time of disk wiper as the carriage moves from disk 2 to disk 46. Notice the change in slope, indicating that the carriage slows down as it senses it is approaching its destination.

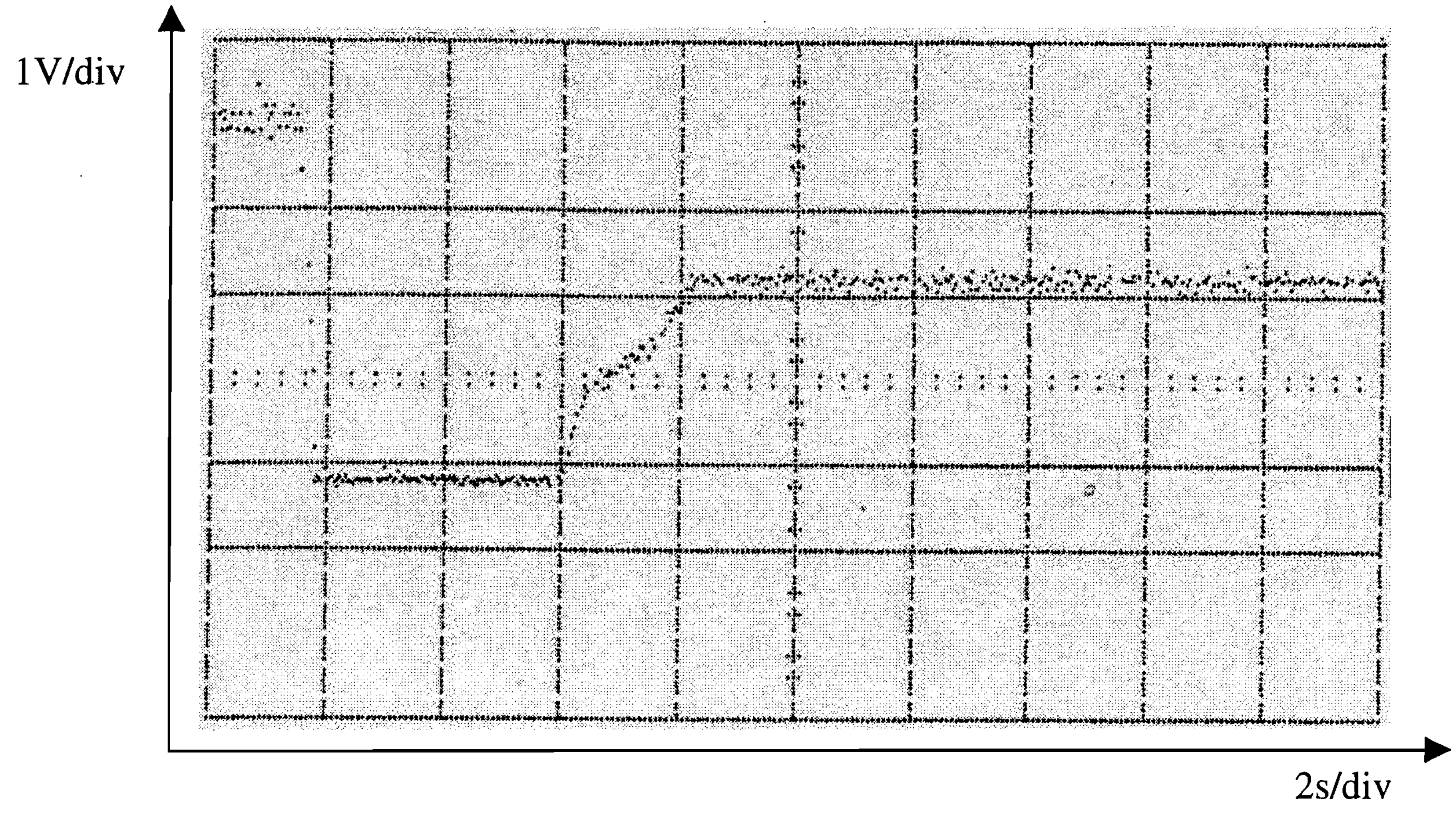


Figure 4.4-3: Voltage Vs. Time of track wiper as the arm moves from track 100 to track 50. Notice that the arm retracts all the way first, waits a little bit and then moves back into the array and reaches its destination, track 50.

Chapter 5: Societal Issues

1.Ethical

The restoration of the RAMAC is not an ethically controversial design project. Our group does however believe that our project, upon completion, will contribute to the greater good of our society. Few people believe that there are historical artifacts in the field of electronic technology, but as time has passed, milestones like the RAMAC need to be preserved as antiques. 100 years from now, when a student visits a museum to see the RAMAC at work, our contribution to society will be clearer. In conclusion, our project contributes to society a great deal while providing no harm what's so ever, this is why we feel comfortable in the ethical purpose of the restoration of the RAMAC.

2.Social

Social issues are of definite significance in the restoration process of the IBM RAMAC (Random Access Method of Accounting and Control). As described in the Engineering Handbook, the role of technologies in society, great technological achievements themselves, and people's needs are all social issues that must be looked upon. In working with the RAMAC, all of these issues are relevant.

The RAMAC is an amazing historical device that is the basic foundation of all the disk drives used today. Built in 1956, it was the first disk drive ever. For our project, we are in charge of restoring one of only four that are remaining today. The RAMAC has much historical worth, as it is to be an IEEE milestone, is a Mechanical Engineering landmark, and actually one of the first robotic devices ever. In restoring the RAMAC we will be working closely with faculty as well as industry engineers. We feel this interaction will be greatly beneficial to us and to our society. Because of the increasing use of hard drives and their continual technological advancement, these amazing devices are often taken for granted. By being able to watch and observe the first magnetic disk drive function, society can better understand the basic operations needed to read and write data. We will be able to put into perspective just how far we have come in technology by illustrating what we started with, and comparing the RAMAC with today's advanced technology. The original RAMAC was the size of a refrigerator, and had a total capacity of only 5 megabytes. Today, there are disk drives a fraction of that size that store a thousand times more. By showing this, we will have reviewed great technological developments and advancements of the past. Realizing this, society will have a better understanding of what to expect in the future in this line of technology. Most importantly though, society will hopefully gain an appreciation for this technology, and understand its role and significance in the world.

Another essential social aspect is that of human needs. Engineering is described as the business of meeting human needs through technological development. This definitely applies to our restoration of the RAMAC. Maslow's hierarchy of human needs consists of physiological needs, safety, love, esteem, and self-actualization. Of these, esteem and self-actualization seem most applicable to the issue at hand. Self-esteem comes from feeling good about what we have accomplished in life; a sense of satisfaction with the contributions that we make. The restoration of the RAMAC can provide this need of esteem in that once it is complete, society can see how far along we have come in terms of technology, as mentioned before. Although not every individual in society has directly been involved with the development and progression of this technology, as a whole, we have come a long way, and should be proud of just how far we have come. The human need of self-actualization, which is the ability to grow richer and more complete as life goes on, also applies to our project. In observing the original RAMAC with respect to the technology we have now, one can definitely point out how we have developed our strengths and overcome our weaknesses. Through this realization, we can all enjoy a sense of wholeness and completion. An important thing to keep in mind though, is that this self-actualization is a never-ending process, and so, as a society, we will continue to strive for better things, resulting in more advancements in technology.

Social aspects undoubtedly lie at the heart of restoring something as historical and important as the IBM RAMAC. In going through the process of restoration for our project, we will touch on the subjects of the role of technologies in society, great technological advancements, and people's basic needs.

3.Political

Political matters do not play a role in our project of restoring the IBM RAMAC. Our project has much significance to historical issues; however, it does not have anything to do with politics. The creation of the RAMAC took place in the 1950's, so as you can see, there are no important issues as to how the disk drive relates to our governmental or political issues in our society today. As mentioned before throughout the chapters of the handbook, our project is mainly focused on getting a historical engineering marvel to work again. In doing this, we are trying to show people just how far engineering has come. Our RAMAC, which is a large magnetic hard drive, is the size of a refrigerator; however, it only has a capacity of 5MB. By showing our project to people, they can see the world's first magnetic disk drive, and hopefully understand thoroughly how these drives work. In doing this, we hope they come to appreciate just how far technology has come, and where it can go in the future. Also, we are combining fairly recent technology with the old technology of the 1950's, and are going to see just how it comes together. However, nothing we can think of relates this project to political matters.

4.Economic

In the restoration process of the IBM RAMAC, the issue of economics, as discussed in the Engineering Handbook, does not have huge relevance; it is only marginally relevant. As discussed in the Engineering Handbook, the purpose of the economic analysis of a project is to determine whether the project should be carried out, from a financial perspective. So, this chapter is devoted strictly to the financial issues. What is important here is to look for a way to decide what it costs to produce a product, so that a team can determine whether they can sell the product at an acceptable price.

There are a few reasons as to why this issue does not have a great significance to our restoration of the IBM RAMAC. First of all, we are not restoring the RAMAC in order to see a profit of any kind. The RAMAC has been loaned to Santa Clara University for the purpose to restore it and get it in working order. There is much historical significance in doing a project like this, however, in no way are we attempting or planning to sell this product. Unlike other design projects, we are simply involved to see if we can ameliorate the condition of the existing RAMAC. Other teams may be working on something that in the long run, they can sell as a product for a profit. In these cases, they will probably look into mass production of their product. However, in our situation, we are not in this position, as IBM created the RAMAC, and our only interest is to be a part of this historical landmark. No mass production will be done, as none is desired with our project.

Although we have absolutely no intention to sell our project for financial reasons, throughout the course of our work on the RAMAC, we have had financial issues we have had to face. In the restoration of the RAMAC, we have had to actually purchase many items and equipment, and in doing so, we have always made sure to maintain economic efficiency. For example, in choosing certain parts, we only buy the ones that we will need, and not some fancy one that has extra, unwanted features. We keep in mind what exactly needs to be done according to our short and long term goals, and then obtain the part or equipment that is absolutely necessary to us, and nothing extra. A specific example of this is our microcontroller. We could have chosen another controller that had the functionality to do other things, however, we chose one that had just exactly what we needed. This way, we eliminate any unnecessary costs. We did this for all the additional things we needed to purchase. Although we are getting reimbursed by our advisor for the things needed for the project, we realize it is an important engineering practice to keep in mind what exactly is needed. This way, in the future, we will have had the practice in trying to minimize costs, and maximize profits.

5.Safety

Background

In the restoration process of the IBM RAMAC, the issue of safety is our primary concern. We are concerned with the potential dangers to the operators and the spectators of the RAMAC as well as the potential dangers to the RAMAC itself.

Human Safety

The RAMAC poses a potential danger to individuals in close proximity due to its massive components that move at high velocities. The motion of its mechanical arm and magnetic head accelerate and decelerate, changing direction frequently. With large mass and high changing velocities, the RAMAC has the ability to generate destructive magnitudes of force and momentum. Anyone in close proximity of the RAMAC is in potential danger of injuries.

Because it is the goal of the team to restore the RAMAC to a working state, rather than improve its safety specifications, we will not make direct changes on the RAMAC to accommodate higher safety regulations. For example, it is not in our interest to place a sensor on the arm or the head to sense if an operator is in its path and immediately stop movement. However, it is to our benefit that the causes of potential injuries are generally predictable.

Because the RAMAC is a technological innovation possessing large visible moving parts, people will naturally be drawn closer to see how it operates. As long as spectators are kept at a reasonable distance from the RAMAC in its operating state, there are virtually no health concerns to them.

Particular precautions will be taken in testing the RAMAC. Team members will communicate clearly to each other before any operation is made on the RAMAC. In addition, the motors will not be used until the final stages of testing have been reached. Until this state, the clutches will be rotated by hand to assure that the RAMAC is responding as intended.

Mechanical Safety

Because the RAMAC is one of the remaining four of its kind, we view the prevention of damage to the RAMAC equally important as to the safety of those operating and spectating. Although there are a few exceptions, the majority of precautions will take place in the coding of the controller. Before any movement takes place, the controller must follow strict protocol checking that every state is prepared for the desired motion. For example, there must be a sensor allowing the controller to know if the detent has been released prior to the arm moving up or down. Movement of the arm with the detent engaged could cause unnecessary wear. The microcontroller must know the location of the arm and the head at all times to prevent overshooting the destination and running to the end of the track. And code must be implemented to prevent current to both clutches for excessive amounts of time to prevent damage to the clutches. Failure to any of these precautions would lead to the termination of operation.

There are safety measures we must take in addition to software precautions. A cover guard must be placed over the disk stack to prevent dust from entering and damaging the disks and the head. This cover will also protect spectators and operators from touching the rotating disks. Also, some sort of padding, such as a block of foam, should be placed at the bottom of the mechanical arm track to catch the arm if it should fall to prevent damage to the arm and head.

Safety of the team, spectators, and the RAMAC are the primary concerns during the RAMAC restoration process. As the restoration process becomes more involved, additional safety measures will become more apparent. It is our goal to create a system with enough redundancy to account for virtually any error.

6.Manufacturability

In our design for the restoration of the IBM RAMAC, we called for specifications consistent with the Manufacturability guidelines in the Engineering Handbook. Because we are restoring the RAMAC, rather than mass-producing new disk drives, many of the guidelines did not apply to our project. However, few of the guidelines proved beneficial in our restoration process. Below is a description of the relevant guidelines.

Simplifying the design and reducing the number of parts is the first guideline mentioned in Chapter 1. In the restoration process, we designed a controller that was as simple as possible but still yielded our desired outcome. For example, we used a Parallax BASIC Stamp microcontroller, a controller that we felt was easier to learn to use than other available microcontrollers. Although we lost some computing power by selecting a simpler controller, we feel that the tradeoff was acceptable. Additionally, we will not use mechanisms from the original controller of the RAMAC that are unnecessary for our restoration process. For example, the RAMAC originally used a tachometer to measure the velocities of the magnetic head. For the restored control system, we will compute this velocity through our feedback system.

Manufacturability guidelines also call for acceptable costs. Whenever possible, we used donated items such as our Parallax microcontroller and PWMPAL. Also, we received many of our hardware components direct from the manufacturer by requesting for samples of the product.

The last relevant guideline was to use a mistake proof design. Because we want our designed controller to work for many years to come, we needed to add design specifications to prevent unnecessary wear to the RAMAC and the control system. For example, we added circuitry to prevent current from surging into the microcontroller from the switching circuit, a mistake that could destroy our microcontroller.

7.Sustainability

Because our senior design project revolves around restoring a single historical piece of hardware, we are not faced with the problem of over using natural resources. Rather than using up raw materials, we are essentially recycling a historical disk drive.

The only energy source we are using for the restoration of the IBM RAMAC is electricity. However, the original specifications used a very large 240 V control system that drew large amounts of current, while we are relying on 5 V microcontroller and a 120 V to drive the motor and magnetic clutches. We are effectively using much less energy and material than the original specifications of the RAMAC.

8.Environmental Impact

In the restoration of the IBM RAMAC, there is very little significance concerning the issue of the surrounding environment. We understand that if we were to mass-produce a product from a factory, of course we could be facing these issues. However, in working on the RAMAC, we are working with a one of a kind machine, and have no intentions on producing more. We are simply trying to get this machine in working order again. We do not have much waste at all, as we are using many of the original parts on the machine already. Also, we are not adding much to the structure of the RAMAC, in a physical sense. We are adding circuitry in order to get the RAMAC to do what we want, however, we have little to no waste. Also, when the project is complete, there are no intentions to recycle the RAMAC. It is a historical landmark, and should be kept intact, as is, for many years to come. Also, we work in a controlled environment, in an isolated room. So, producing pollutants that would harm the rest of the school is not really a possibility. This is because we do not really produce pollutants in running the RAMAC and seeing it operate. We can see how many other projects could have significant environmental impact issues; however, in restoring the RAMAC, we see none.

9.Usability

Usability has been a fairly important issue in restoring the RAMAC. The main purpose in this restoration is for future display of the machine in museums and galleries. As such, we need to make the controls of the machine 1: easy to use 2: durable. The ease of use is an aspect we have yet to refine. We are using bare code as of now to modify the motion of the head and carriage. It would be extremely impractical for a museum visitor to have to modify basic code every time they want to see the RAMAC move. We will surmount this usability obstacle by designing a user-friendly interface for anyone to use. To highlight Dr. Healy's five points, we hope to:

1: We want to make the user interface easy to learn and to remember how to use. We will do this by simply providing the user with two boxes to fill in and one "play" button to press. The user will assign a disk number and track number, which is really straightforward.

2: We hope to make the code and motion control of the RAMAC efficient. After obtaining a complete 'first draft' of the code, which we have not done yet, we will refine it in such a way as to make it more efficient, or essentially, faster.

3: Since the machine might be sitting in a museum for a fairly long time, it is important for us to make it relatively free of failure. We will test the code some more over the next few months and redesign the switching circuit in order to minimize failure. During our design review, we were informed of a flaw in our switching circuit, which we are currently fixing. This is just one of many steps we are going to take to minimize failure.

4: We hope that the control system of the RAMAC will be satisfying to use for any visitor of a museum or gallery. Experiencing the motion of such a huge milestone using a user friendly modern interface will hopefully spark some satisfaction in its user.

5: Sustainability is key in almost any engineering design. Keeping the system free of failure will make it sustainable. By making sure to take care of potential back-EMF and such issue, we believe the control mechanism can last a really long time without being altered.

10.Lifelong Learning

The RAMAC was a marvel of its time. Creating a novelty of this magnitude required for all the modern technology of the time to be incorporated in the design. If the saying holds true "...after six years half of what you learned in school is obsolete", our senior design group would be taking on a very trivial challenge. On the contrary, we have learned that that although technology has changed tremendously, the "Ways of the engineer" have remained timeless.

The structure of an engineering education

Our experience so far into the project proves this point, which contradicts the saying. To parallel Dr. Healy's "Ohm's law" example, the engineers who designed the RAMAC might have not has the technology that we do, but their problem solving and critical thinking methods were no different then than they are now.

Because our modern technology enables us to process data with such ease, three undergraduate students can implement a control system just as efficient than that of the IBM engineering team of the 50's. The head motion control system is a good example. In 1953, engineers had to design extremely complex analog computers to sense the position of the head, analyze its destination and control its motion. Our modern technology, a 3'' by 4'' micro controller board, enables us to sense the position of the head, send the position and destination signal to the processor who then seems to do much of the "engineering". Although somewhat dramatic, this view is not far from fact. Why then, one may ask, did we chose this senior design project as engineering majors? The answer lies in the fact that the processing units do not inherently know how to process the data given to them. Processors are silicon, they do no critical thinking, they do not learn how to analyze new situations, and they are not creative. These attributes have been and will remain within the grasp of only the human being, the engineer to be more specific. In our example, the processor does not know what to do with Destination: 4.586V, Location: 3.488V. The engineers, we, write the command in code for the processor to understand and to take appropriate action.

We can clearly see that the thought process is then in no way different now then it was 50 years ago. Technology has changed, it has made us more efficient engineers, it has facilitated and accelerated design, but it has not and will not replace the fundamental ability of the engineer, that of critical thinking and problem solving.

What's your style?

Within every design, all nine of Gardiner's intelligences are relevant. In order to better understand which of these qualities will be necessary and why, we thought it would be a good idea to assign to each an example pertaining to our senior design. Here they are:

Mathematical

The ability to process logical problems plays an important role in every design. Calculations will have to be made in order to write the proper code for the motion of the head. Momentum, gravity and acceleration are all factors which we will have to take into account in our calculations.

Verbal-Linguistic

The ability to learn and express ourselves through language has come into play already as we've had to read manuals and present material to our advisors.

Visual-Spatial:

The ability to think in images and pictures is always crucial in a design that involves such mechanical complexity.

Bodily-Kinesthetic:

The ability to skillfully accomplish a manual task is a quality we must possess, as we will directly be involved in the disassembly of the RAMAC.

Interpersonal:

The ability to understand other people and to learn by understanding another's viewpoint will be crucial in our project in our meetings with our advisors and industry engineers.

Interpersonal:

The ability to understand ourselves is vital to the progress of our design. We must each know our capabilities, our faults and our qualities alike in order to make our group work more efficient.

Naturalist:

The attention to detail enrobed in this type of intelligence is a quality anyone can make good use of. In our case, the choice we have to make between micro processing techniques, for example will require the great attention to detail a naturalist would have.

Musical:

The ability to recognize meaning within sound is of good use in any mechanical system. In our design, for example, we need to pay attention to the noise the array bearings make upon rotation. Some noise might be early signs of wear.

Finally, the ability to ponder fundamental questions about life was important in our choosing this senior design. The goal of our project is to restore the automated motion of the RAMAC's access mechanism. We hope to demonstrate the functionality of the first magnetic disk drive, putting it in perspective of our modern era by combining technologies that are half a century apart. Because of the increasing use of hard drives and their continual technological advancement, these amazing devices are often taken for granted. By displaying a functioning RAMAC, we can contribute to society by giving it a better understanding of the basic operations needed to read, write and retrieve data.

11.Compassion

The compassion chapter of the Engineering Handbook has little or no relevance for our project. Our project does not concern aiding the health of people, increasing productivity, or spreading world piece. Rather our project goals are to increase awareness of the historical significance of the IBM RAMAC and to build interest in electro-mechanical systems.

Chapter 6: Conclusion

6.1 Summary of Work

In our process of restoring the RAMAC, our objectives were the following:

5. **Controllable manual motion**. Implement an actuator interface to allow carriage up/down and arm in/out motion at variable speeds.
6. **Accurate sensing**. Implement a sensor interface to provide carriage and arm position information.
7. **Automation**. Integrate motion and sensing into a user-friendly automated control system. Improve access accuracy and time through any means necessary.
8. **Internet-based interface**. An Internet-based interface would assist in integrating the RAMAC controller with the web so that it could be accessed from a remote system.

We were able to obtain controllable manual motion. This was achieved with a robust switching system along with our micro controller to control the Pulse width Modulation. When different duty cycles were applied to the magnetic powder clutches through the switching system, we were able to control the velocity of the carriage and arm at variable speeds.

With the addition of our feedback mechanism, we were able to accurately provide sensing information. Although this sensing is not tolerant to high levels of noise induced at high velocities, it is able to maintain accurate sensing at low speeds.

With the success of controllable motion and accurate sensing, we were able to create automated motion for slow speeds. A user-friendly interface was implemented which allowed the user to enter a destination disk and track location.

The Internet based interface was implemented with the help of Santa Clara University's Robotic Systems Lab. It provided a user with the capability to control the system from any internet-equipped location.

6.2 Future Work

The final goal in the restoration of the RAMAC is to render the system fully operational. The full restoration of the RAMAC is a three step process:

1. Restoring the mechanical components of the hard drive. This was done by Pat Connolly during the summer prior to our project.
2. Designing an automated system to control the motion of the carriage and the arm. This has been successfully implemented by our design team.
3. Restoring the read/write capability of the hard drive. This will hopefully be accomplished by the next restoration team.

The third step in this process deserves some attention as it is actually the only "future" work on the list. Restoring the read/write capability of the RAMAC will require some preparation before any signal processing work is to be done. First on the list is the cleaning of the disks. The data can not be written or read from the disks in their current condition. Our advisor, Dr. Hoagland, with the help of Lou Taft of IBM developed a method to release the dust from the disks without damaging the magnetic data stored on them. Once the disks are clean, the arm should be carefully drawn in and out of the disk array while the disks are spinning. Our team did not try this because it was of no relevance to our design to have the disks spinning. It is of outmost importance however, to check this in order to prevent any possible damage to the disks. The next step in the process will be to verify that the head can load and unload from the disks without touching them. Once these tests have been run, the team can move on to the analysis of the data read. As mentioned above, this will require great knowledge and use of digital processing techniques.

Once full restoration is achieved, the RAMAC will be able to read and write data at near original specifications.

6.3 Lessons Learned

Although there are many lessons we learned all through this project, four main points were encountered over and over again throughout the year:

1. Data Always needs to be quantified. Although a problem may seem like it an be solved qualitatively, numbers often speak louder than words. Noise levels and frequencies, speed requirements and accuracy percentages are all examples of problems which were solved much more easily through quantitative comparison rather then through qualitative reasoning.
2. Standards in hardware protection need to always be followed. We learned this the hard way: when protecting diodes were not installed, we physically blew one of our transistors apart. This caused us to lose great time as we had to require and add components to our circuit. More importantly, though, hardware mishaps can be harmful to the users of a system. Furthermore, hardware protection prolongs the life of the circuit.
3. Methods should be strategically planned and thought about before being implemented. We lost a lot of time and patience in implementing methods that proved to be inefficient. Mathematical calculations and simulations are good tools to use prior to "getting your hands dirty".
4. Spare parts are never too abundant. No matter how many samples we ordered or wires we bought, we never had enough. Because many senior design teams are working on building hardware for the first time, they are also learning in the process, this lack of knowledge often leads to inefficient use of parts such as wires and mishandling of components such as chips. Furthermore, having extra parts at hands reach will prove useful for any team which wishes to take on an ongoing project. If a part breaks, there will be no delay in getting it replaced.

Although these four issues should not be ignored, the biggest lesson we learned is not in the bullet points above. As engineers who are soon to enter the workforce, we need to be prepared to work in teams and coordinate individual efforts. Throughout the course of the year this project has given us the opportunity to learn these attributes. By setting up weekly meetings and splitting up tasks, we stayed organized and locked in to our final goal. We also learned that although some work is inherently done independently when a project is split, it is always important to listen to your peer's advice. Two perspectives are better than one, and it can save a project a lot of time and frustration if the people working on it are open to suggestions from others.

Appendix A

Instruction Manual

Instruction Manual

a. How does the RAMAC work?

The carriage, which is depicted in Figures 1 and 2 below, can move up and down to access any of the disks. When the carriage is locked into place, the access arm can move in and out to its desired location.

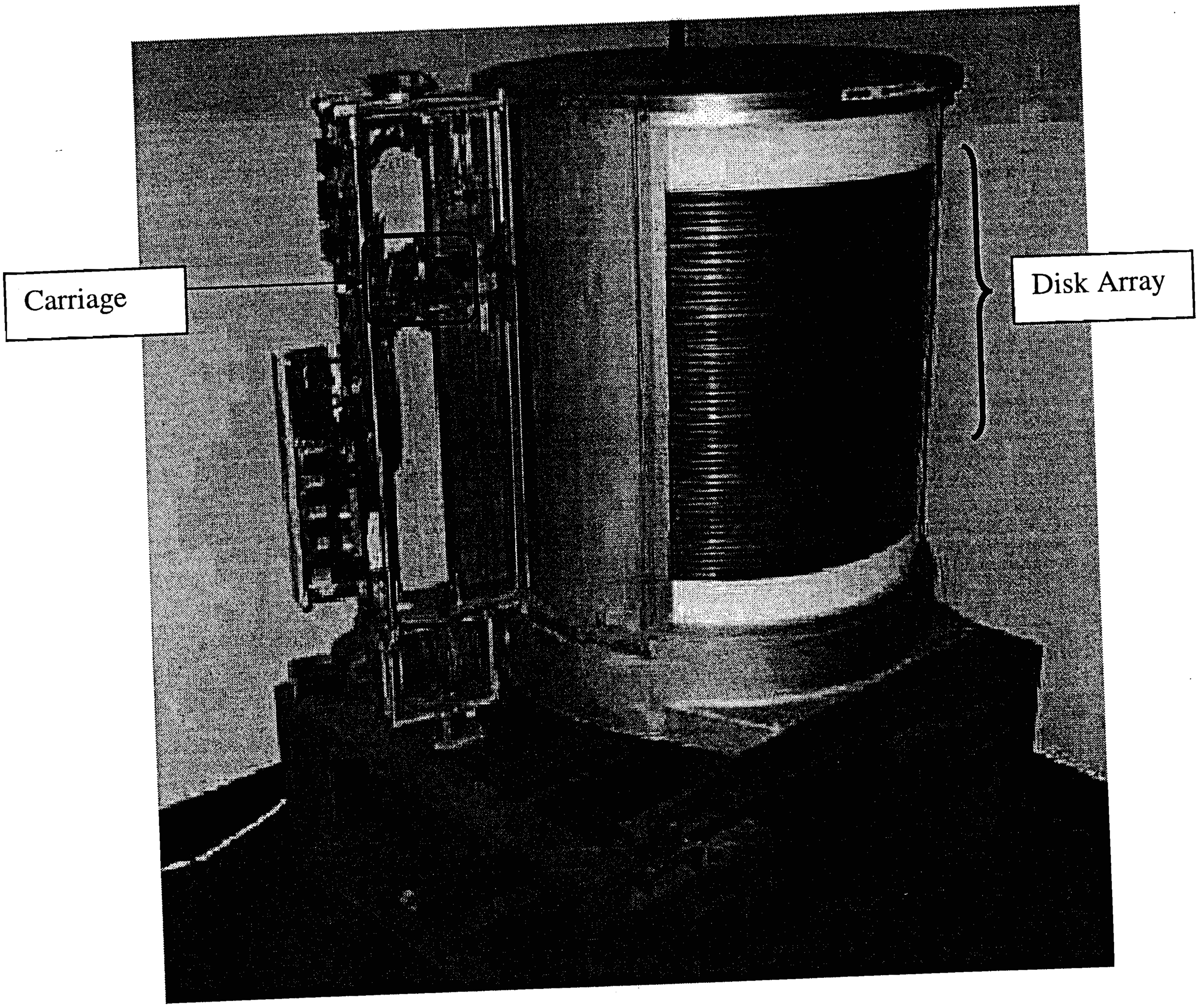


Figure 1: Hard Drive

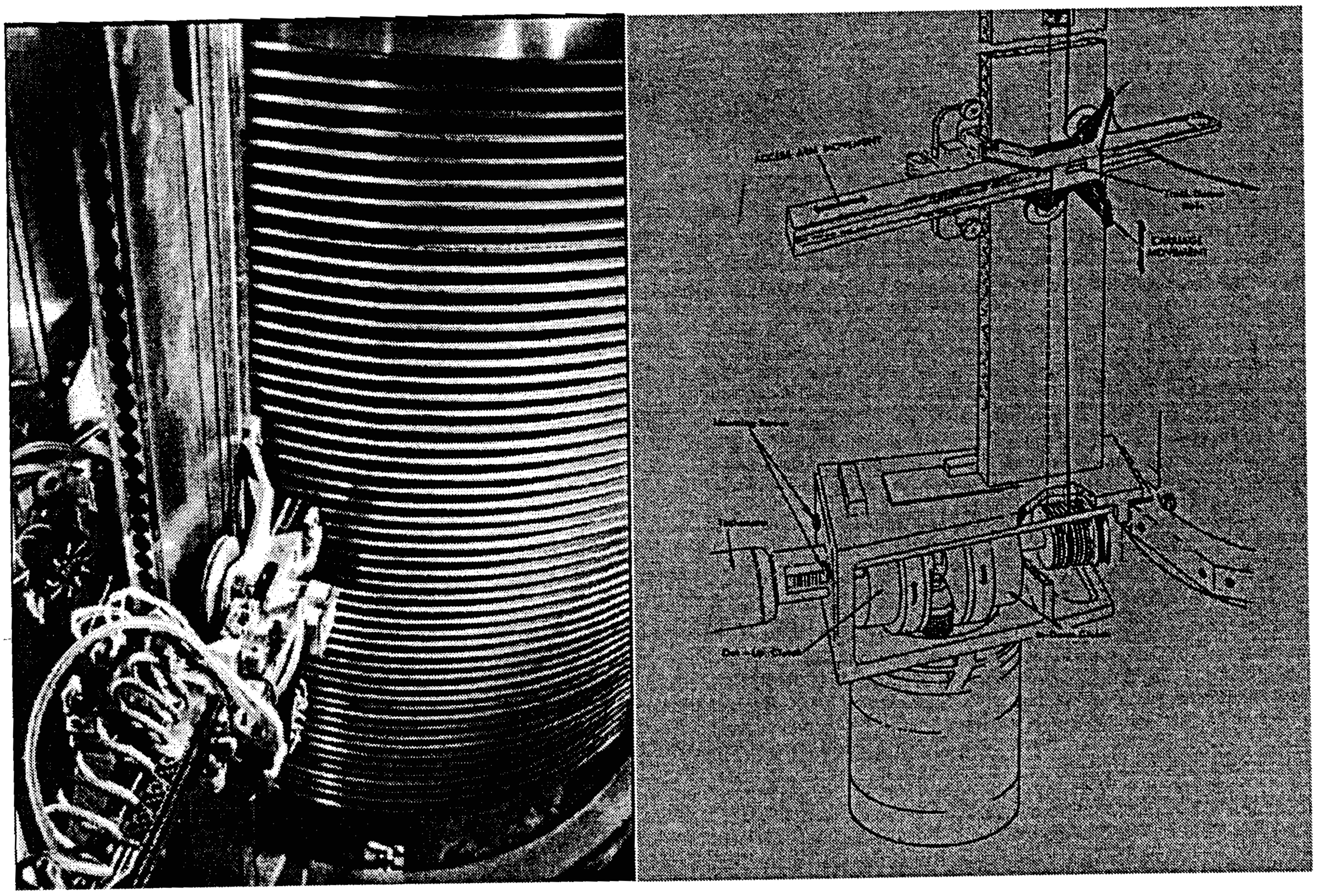


Figure 2: Access Mechanism

The AC motor at the bottom of the access mechanism drives the motion. A pinion mounted on the motor's shaft spins two magnetic clutches in opposite directions. When a DC current is applied to the clutches, the magnetic powder becomes more viscous and grips the horizontal shaft, which turns the pulley system and moves the carriage or arm.

b. Introduction to our System

Our design can be split into four main interfaces as shown in Figure 3. The sensor interface (1) obtains data on the position of the carriage and arm from the RAMAC. This information is sent to a controller interface (2) that can then control how to move the carriage or arm through the actuator interface (3), which translates the controller's commands into high power inputs to the RAMAC. The system can also be controlled from a remote site by using our Web-based control interface (4). The user interface is simply a PC equipped with the proper software.

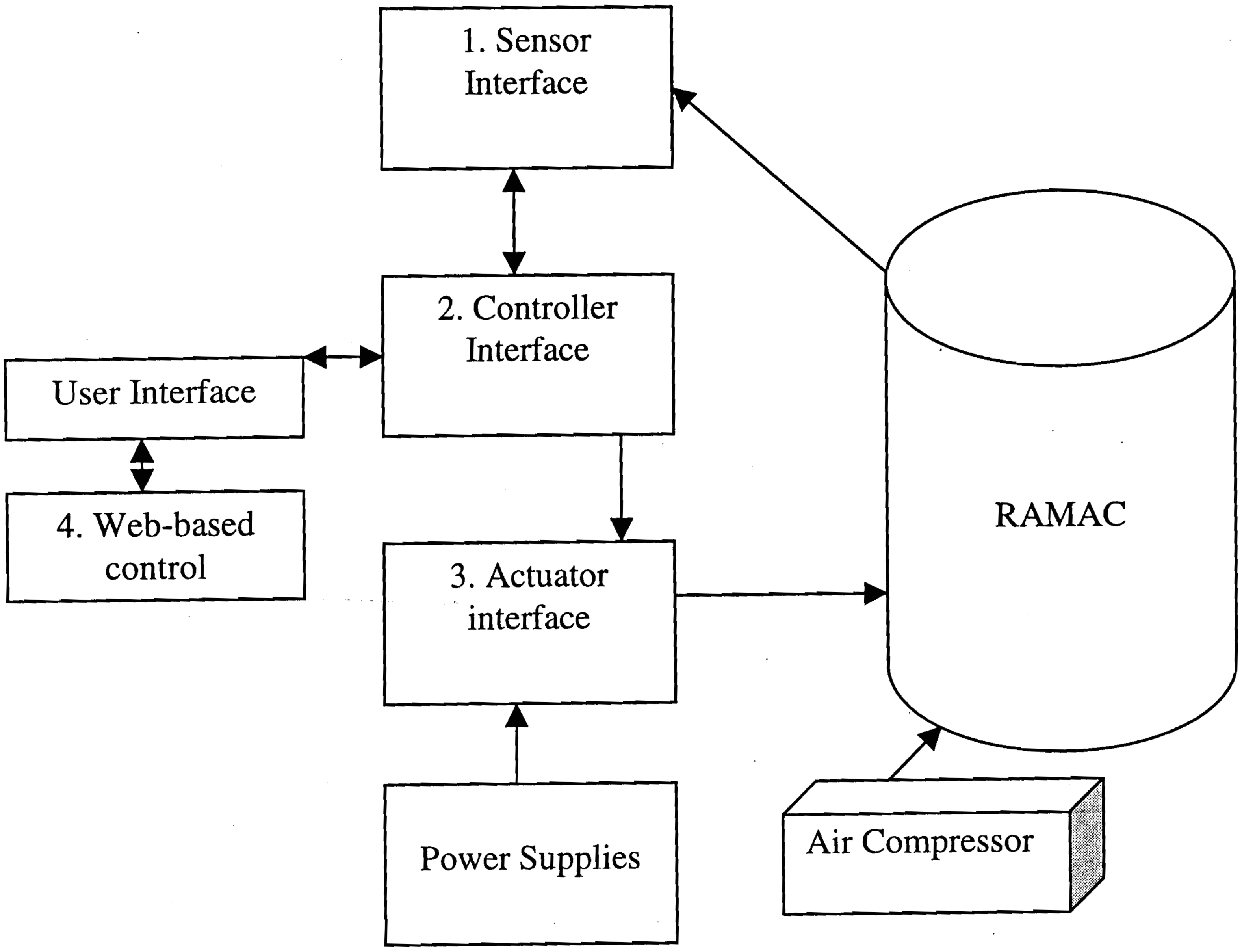


Figure 3: Interface Overview Diagram

The physical components which comprise our system are: the cabinet, which carries all the controlling units, the hard drive unit, which is the historical device present prior to our project, and the air compressor, which is detached from the two but supplies air pressure to the solenoids. These are shown in Figure 4.

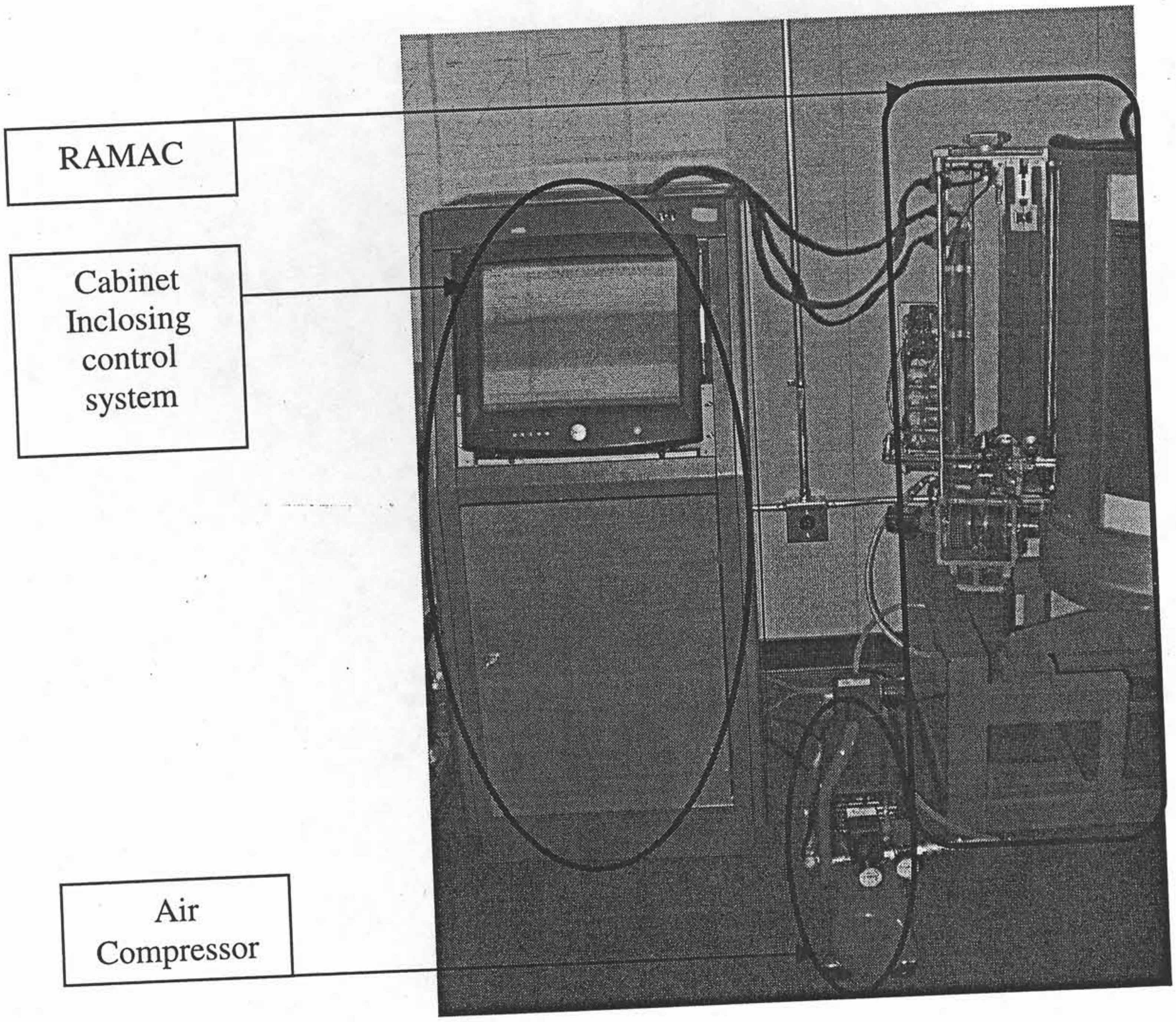


Figure 4

Inside the cabinet is: a PC, a microcontroller, a switching circuit and Voltage supplies. The PC serves to program the microcontroller, which then takes over the automation. These are shown in Figure 5.

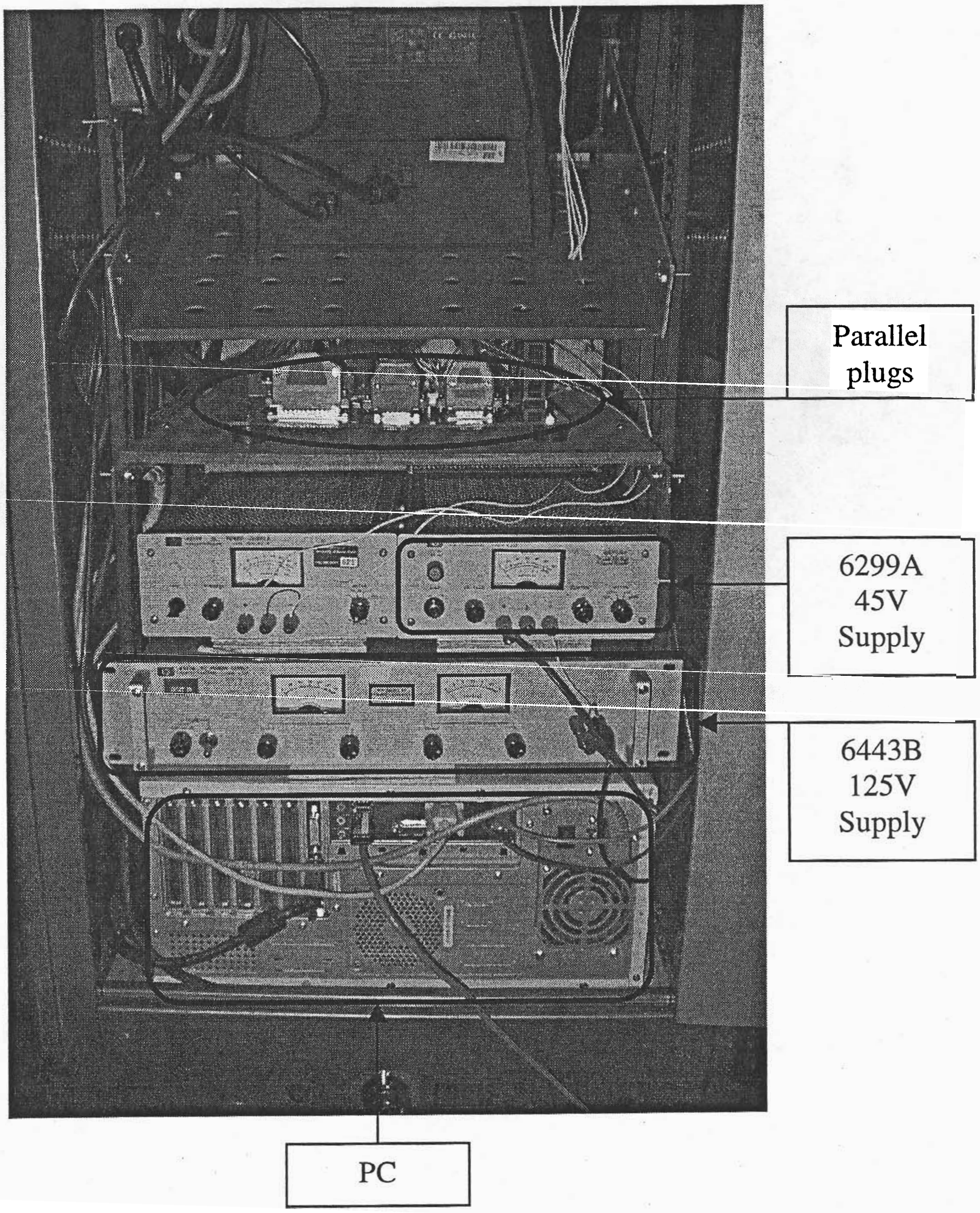
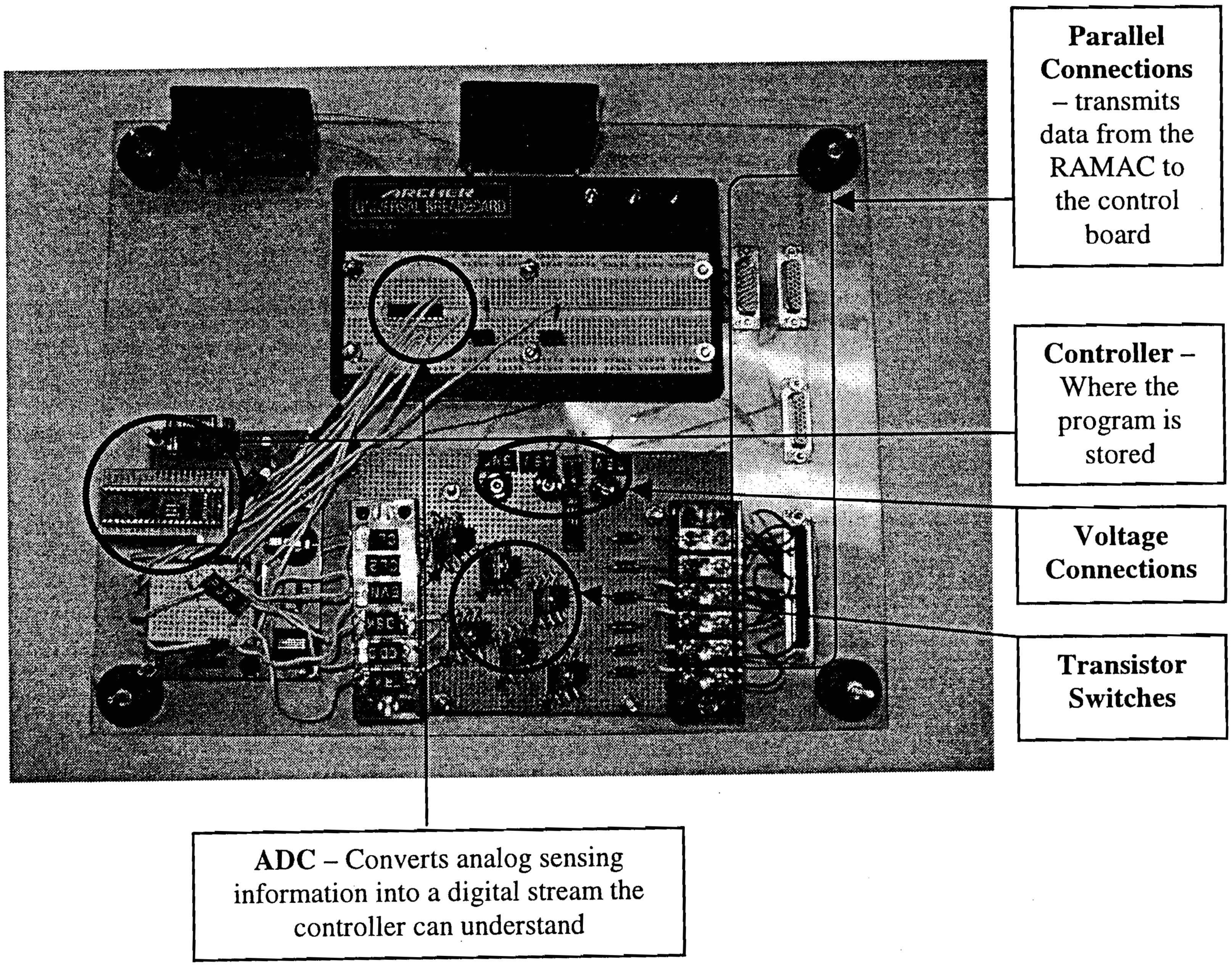


Figure 5: Cabinet Internals

The control system we have designed can be divided into three main subsystems: The actuator interface, the sensing interface and the controller.

The actuating interface is comprised of transistor switches. Since the controller can only supply low voltages, the switching circuit is used a medium between the commands generated from the controller and those supplied to the clutches/solenoids.

The sensing interface is comprised of three components: a resistor strip which lines the vertical way of the carriage provides analog voltage values; a potentiometer which is geared to the horizontal way provides similar values and an Analog to Digital converter which converts this analog sensing information into a digital stream the controller can understand.



Parallel Connections – transmits data from the RAMAC to the control board

Controller – Where the program is stored

Voltage Connections

Transistor Switches

ADC – Converts analog sensing information into a digital stream the controller can understand

Figure 6: Control Board

The controller is where the program is stored and it is the decision maker of the system. The controller's memory is programmed using software provided by Parallax (The controller manufacturer), the language is BASIC. Once the user enters a destination, the controller uses a sensing interface to find out where the carriage/arm is. Once the direction of travel is determined, the controller orders the high power switches to draw current through the clutches, which drives the motion. Once the destination is reached, the controller orders the appropriate solenoid switch to turn on, which stops the carriage or the arm in the right location.

3. Starting and using the system

a. Open the front cabinet door and turn the computer on. The computer is located at the bottom of the cabinet.

b. Open the rear cabinet door and turn on voltage supplies numbered 6443B and 6299A. Make sure the supplies are set at 125V and 45V respectively.

c. Make sure the parallel plugs located on the switching circuit board are connected properly.

d. Check that the compressor tube is installed on the RAMAC's air fitting properly. If it is not, make sure to reconnect it.

e. Pressure up the air compressor until the needle reads 50Psi. Please beware: the air compressor pump is fairly loud.

f. Plug the AC motor's power cord into a wall jack. Be careful not to touch the clutched or the pinion when you do this as they will start spinning really fast.

g. Open up the program on the Desktop labeled "Parallax Basic stamp Editor Version 2.1"

h. Once the program is open, go to file>open>Desktop>final presentation>Final Code. Open up the program

i. Run the program by hitting the "Play" button on the interface. A debug screen should eventually appear prompting you to enter the destination.

j. Enter a disk destination between 1 and 50 and hit enter. Once you hit enter, the program will prompt you to enter the track location, type a track location between 1 and 100 and hit enter. If all the supplies have been turned on and the connections are made properly, the carriage should begin moving to the desired location.

4.Turning the system off

a. Close the running program and the Parallax software
b. Unplug the ac motor's wall jack
c. Disconnect the air compressor fitting from the RAMAC
d. Open the rear cabinet door and turn off voltage supplies numbered 6443B and 6299A.
e. If you wish to turn the computer off, you may do so by going to Start>Shut down and then opening the front door and manually shutting off the PC.

The controller is where the program is stored and it is the decision maker of the system. The controller's memory is programmed using software provided by Parallax (The controller manufacturer), the language is BASIC. Once the user enters a destination, the controller uses a sensing interface to find out where the carriage/arm is. Once the direction of travel is determined, the controller orders the high power switches to draw current through the clutches, which drives the motion. Once the destination is reached, the controller orders the appropriate solenoid switch to turn on, which stops the carriage or the arm in the right location.

c. Starting and using the system

a. Open the front cabinet door and turn the computer on. The computer is located at the bottom of the cabinet.
b. Open the rear cabinet door and turn on voltage supplies numbered 6443B and 6299A. Make sure the supplies are set at 125V and 45V respectively.
c. Make sure the parallel plugs located on the switching circuit board are connected properly.
d. Check that the compressor tube is installed on the RAMAC's air fitting properly. If it is not, make sure to reconnect it.
e. Pressure up the air compressor until the needle reads 50Psi. Please beware: the air compressor pump is fairly loud.
f. Plug the AC motor's power cord into a wall jack. Be careful not to touch the clutched or the pinion when you do this as they will start spinning really fast.
g. Open up the program on the Desktop labeled "Parallax Basic stamp Editor Version 2.1"
h. Once the program is open, go to file>open>Desktop>final presentation>Final Code. Open up the program
i. Run the program by hitting the "Play" button on the interface. A debug screen should eventually appear prompting you to enter the destination.

j. Enter a disk destination between 1 and 50 and hit enter. Once you hit enter, the program will prompt you to enter the track location, type a track location between 1 and 100 and hit enter. If all the supplies have been turned on and the connections are made properly, the carriage should begin moving to the desired location.

d.Turning the system off

a. Close the running program and the Parallax software
b. Unplug the ac motor's wall jack
c. Disconnect the air compressor fitting from the RAMAC
d. Open the rear cabinet door and turn off voltage supplies numbered 6443B and 6299A.
e. If you wish to turn the computer off, you may do so by going to Start>Shut down and then opening the front door and manually shutting off the PC.

Appendix B

Source Code

```
'{$STAMP BS2p}
'{$PBASIC 2.5}

'----[Declarations]----
adcBits             VAR          Word
Disk_Current        VAR          Word
Disk_Dest           VAR          Word
Track_Current       VAR          Word
Track_Dest          VAR          Word
ADch                VAR          Word
Moving_UporOut      VAR          Word
Check_Disk          VAR          Word
Check_Track         VAR          Word
oddOrEven           VAR          Word


'----[Initialization]----
CS                  PIN          2
DataInput           PIN          3
DataOutput          PIN          4
CLK                 PIN          5


'******This Function asks the user for destination info******

User_Input_Disk:

DEBUG CLS,"What disk would you like me to go to?:  ",CR
DEBUGIN DEC2 Disk_Dest
Disk_Dest = Disk_Dest * 10 'this is to scale the input properly

IF ( Disk_Dest = 260 ) THEN
Disk_Dest = 255
ENDIF

IF (Disk_Dest =< 260) THEN
HIGH 1
ELSEIF (Disk_Dest > 260)  THEN
LOW 1
ENDIF


GOSUB User_Input_Track



'******This Function asks the user for destination info******

User_Input_Track:

DEBUG CLS,"What Track would you like me to go to?:  ",CR
DEBUGIN DEC3 Track_Dest
oddOrEven = Track_Dest // 2
```

```
Track_Dest = Track_Dest + 16
Track_Dest = Track_Dest * 10 'this is to scale the input properly

GOSUB Move_Out

'RETURN

Move_out:


    LOW 7
    LOW 6
    PAUSE 200
    SEROUT 0, 240, ["!PWMSS", %10100000]
    SEROUT 0, 240, ["!PWMM4", 65, 1, 100, 10]
    PAUSE 1500
    SEROUT 0, 240, ["!PWMSS", %10100000]
    SEROUT 0, 240, ["!PWMM4", 0, 0, 100, 10]


GOSUB Sense_And_Pulseout_Disk


'******This Function senses the position, calculates the relative position
'******and pulses out the proper PWM******

Sense_And_Pulseout_Disk:



Moving_UporOut = 0
ADch = 0

HIGH CS
LOW CS
LOW CLK
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]

IF (Disk_Dest =< 260) THEN
Disk_Current =( (adcbits*10 - 768 ) /155) 'This is the curve fit equation
ELSEIF (Disk_Dest > 260) THEN
Disk_Current =( ( (adcbits* 10) / 158 ) + (41422 /158) )
ENDIF

IF (Disk_Current > Disk_Dest) THEN

    SEROUT 0, 240, ["!PWMSS", %10000000]
    SEROUT 0, 240, ["!PWMM4", 150, 1, 180, 6]
    Moving_UporOut = 1
    PAUSE 500 ' so that gravity doesn't drag the carriage down
    HIGH 9   'These two are connected to the proper solenoids, they
    LOW 8    'will release the carriage and let it move up and down


ELSEIF (Disk_Current < Disk_Dest) THEN
```

```
 SEROUT 0, 240, ["!PWMSS", %10000000]
 SEROUT 0, 240, ["!PWMM4", 0, 1, 100, 6]
 Moving_UporOut = 0
 PAUSE 700

 HIGH 9
 LOW 8




ENDIF

GOSUB Proportional_Disk

'*****This function generates 2 stage proportional motion******

Proportional_Disk:

 DO

ADch = 0

HIGH CS
LOW CS
LOW CLK
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]

IF (Disk_Dest =< 260) THEN
Disk_Current =( (adcbits*10 - 768 ) /155) 'This is the curve fit equation
ELSEIF (Disk_Dest > 260) THEN
Disk_Current =( ( (adcbits* 10) / 158 ) + (41422 /158) )
ENDIF



IF ( Moving_UporOut = 1 AND ( (Disk_Current - Disk_Dest) < 20)) THEN

     SEROUT 0, 240, ["!PWMM4", 105, 1, 180, 6]

    GOSUB Sense_And_Lock_Disk

ELSEIF ( Moving_UporOut = 0 AND ( (Disk_Dest - Disk_Current) < 30) ) THEN


      HIGH 8
      LOW 9
      SEROUT 0, 240, ["!PWMM4", 30, 1, 100, 6]
      PAUSE 100
      LOW 8
```

```
      HIGH 9

    GOSUB Sense_And_Lock_Disk

ENDIF


LOOP



'******This function senses continuously and locks in when destination
'******has been reached******

Sense_And_Lock_Disk:


DO

ADch = 0

HIGH CS
LOW CS
LOW CLK
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]

IF (Disk_Dest =< 260) THEN
Disk_Current =( (adcbits*10 - 768 ) /155) 'This is the curve fit equation
ELSEIF (Disk_Dest > 260) THEN
Disk_Current =( ( (adcbits* 10) / 158 ) + (41422 /158) )
ENDIF

IF (Moving_UporOut = 1 AND Disk_Current =< Disk_Dest - 6  ) THEN
'This if statement provides the control system with a destination error margin
  HIGH 8
  LOW 9
  SEROUT 0, 240, ["!PWMSS", %10000000]
  SEROUT 0, 240, ["!PWMM4", 0, 0, 80, 6]
  SEROUT 0, 240, ["!PWMSS", %00100000]
  SEROUT 0, 240, ["!PWMM2", 0, 0, 80, 6]

 GOSUB Check_And_Correct_Disk

 ELSEIF (Moving_UporOut = 0 AND Disk_Current => Disk_Dest - 8) THEN

  HIGH 8
  LOW 9
  SEROUT 0, 240, ["!PWMSS", %10000000]
  SEROUT 0, 240, ["!PWMM4", 0, 0, 80, 6]
  SEROUT 0, 240, ["!PWMSS", %00100000]
  SEROUT 0, 240, ["!PWMM2", 0, 0, 80, 6]

 GOSUB Check_And_Correct_Disk
```

```
ENDIF


 DEBUG HOME
 DEBUG CR, CR, "Decimal Value:   ", DEC4 adcBits
 DEBUG CR, "You are on disk #  ", DEC3 (Disk_Current)

LOOP


'******This Function checks to make sure that the carriage is locked in
'******at the proper disk. If it is not, it calls the Sense_And_Pulseout
'******routine again, if it is, it moves on through the loop******

Check_And_Correct_Disk  :

FOR Check_Disk = 1 TO 3

ADch = 0

HIGH CS
LOW CS
LOW CLK
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]

IF (Disk_Dest =< 260) THEN
Disk_Current =( (adcbits*10 - 768 ) /155) 'This is the curve fit equation
ELSEIF (Disk_Dest > 260) THEN
Disk_Current =( ( (adcbits* 10) / 158 ) + (41422 /158) )
ENDIF

IF (Disk_Current < Disk_Dest - 6 OR Disk_Current > Disk_Dest + 6   ) THEN

     GOSUB Sense_And_Pulseout_Disk

ENDIF

 DEBUG HOME
 DEBUG CR, CR, "Decimal Value:   ", DEC4 adcBits
 DEBUG CR, "You are on disk #  ", DEC3 (Disk_Current)

NEXT


GOSUB Sense_And_Pulseout_Track


'******This Function senses the position, calculates the relative position
'******and pulses out the proper PWM******

Sense_And_Pulseout_Track:

ADch = 1

LOW CS
```

```
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch<<8\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]
HIGH CS

Track_Current = ( (adcbits*10)-55) /32

IF (Track_Current > Track_Dest) THEN
    Moving_UporOut = 1
    LOW 7
    LOW 6
    SEROUT 0, 240, ["!PWMSS", %10100000]    'need pwm for down motion
    SEROUT 0, 240, ["!PWMM4", 250, 0, 100, 10]
    'SEROUT 0, 240, ["!PWMM4", 160, 0, 100, 10]

ELSEIF (Track_Current < Track_Dest) THEN

    Moving_UporOut = 0
    LOW 7
    LOW 6
    SEROUT 0, 240, ["!PWMSS", %10100000]
    SEROUT 0, 240, ["!PWMM2", 200, 0, 100, 10]
    'SEROUT 0, 240, ["!PWMM2", 160, 0, 100, 10]

ENDIF

GOSUB Sense_And_Lock_Track


'******This function senses continuously and locks in when destination
'******has been reached******

Sense_And_Lock_Track:


DO

ADch = 1

LOW CS
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch<<8\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]
HIGH CS


Track_Current = ( (adcbits*10)-55) /32

IF (Moving_UporOut  = 1 AND (Track_Current < (Track_Dest + 2) )  ) THEN

'This if statement provides the control system with a destination error margin

     SEROUT 0, 240, ["!PWMSS", %10000000]    'need pwm for 0% duty cycle
     SEROUT 0, 240, ["!PWMM4", 0, 0, 100, 0]
     SEROUT 0, 240, ["!PWMSS", %00100000]    'need pwm for 0% duty cycle
     SEROUT 0, 240, ["!PWMM2", 0, 0, 100, 0]

   IF ( (oddOrEven) =1) THEN
```

```
    HIGH 7
    LOW 6
   ELSE
    HIGH 6
    LOW 7
   ENDIF

    GOSUB Check_And_Correct_Track

ELSEIF(Moving_UporOut  = 0 AND (Track_Current > (Track_Dest-2) ) ) THEN

    SEROUT 0, 240, ["!PWMSS", %10000000]    'need pwm for 0% duty cycle
    SEROUT 0, 240, ["!PWMM4", 0, 0, 100, 0]
    SEROUT 0, 240, ["!PWMSS", %00100000]    'need pwm for 0% duty cycle
    SEROUT 0, 240, ["!PWMM2", 0, 0, 100, 0]

   IF ( (oddOrEven) =1) THEN
    HIGH 7
    LOW 6
   ELSE
    HIGH 6
    LOW 7
   ENDIF

  GOSUB Check_And_Correct_Track

 ENDIF



  DEBUG HOME
  DEBUG CR, CR, "Decimal Value:   ", DEC4 adcBits
  DEBUG CR, "You are on Track #  ", DEC4 (Track_Current -160)

 LOOP

 'return








 '*******This Function checks to make sure that the carriage is locked in
 '*******at the proper disk. If it is not, it calls the Sense_And_Pulseout
 '*******routine again, if it is, it moves on through the loop******

 Check_And_Correct_Track  :

 FOR Check_Track = 0 TO 3

 ADch = 1
```

```
LOW CS
SHIFTOUT DataOutput, CLK, MSBFIRST, [ADch<<8\12]
SHIFTIN DataInput, CLK, MSBPRE,[adcBits\12]
HIGH CS

Track_Current = ( (adcbits*10)-55) /32

IF (Track_Current < Track_Dest - 8 OR Track_Current > Track_Dest + 8   ) THEN


      GOSUB Sense_And_Pulseout_Track

 ENDIF

  DEBUG HOME
  DEBUG CR, CR, "Decimal Value:   ", DEC4 adcBits
  DEBUG CR, "You are on disk #  ", DEC4 (Track_Current - 160)

 NEXT

 GOSUB User_Input_Disk


 'RETURN
```

Appendix C

Data Sheets