7289 Ednor Crescent
Burnaby, BC
V5A 3J9
(604) 294-2651

March 16, 1999

Dr. Andrew Rawicz
School of Engineering Science
Simon Fraser University
Burnaby, BC
V5A 1S6

Re: ENSC 370 Design Specification, *Noverdose- Medication Delivery System*

Dr. Rawicz,

The improper use of medication is a common occurrence for elderly and confused persons, especially when multiple does are taken at different times of the day. Out of all the elderly persons that are hospitalized each year 30% of these cases are due to the over or under doses of medication.

The purpose of this document is to describe the design specifications of the Noverdose medication delivery system. The Noverdose medication dispenser is a device intended to replace the current, inadequate medication scheduling systems such as multi-boxes and bubble sheets. These current systems are often confusing and don't actually remind the patient to take their medication. Patients are alerted when their medication is ready, thereby greatly reducing the chance of an under dose. The Noverdose dispenses a patient's medication at the correct times and in the correct quantities, eliminating the possibility of an overdose.

Feel free to contact any member of Prescriptek at the phone number indicated above if you wish to discuss our proposal.

Sincerely yours,

The Prescriptek Team:
Bryce Pasechnik
Damian Nesbitt
Derek Young
Rob Boyes
Michael Boquist

ENSC 370
Design Specification

Noverdose Medication Delivery System

Team members:

Michael Boquist
Bryce Pasechnik
Damian Nesbitt
Derek Young
Rob Boyes

Contact:

Michael Boquist
294-2651
mboquist@sfu.ca

Date:

March 16, 1999

Abstract

Noverdose is an automated medication delivery system that dispenses the correct amount of medication at the correct times. A PIC16F84 is used to access scheduling information for a particular patient's prescriptions. At the proper dosing time, the Noverdose takes pills from storage bins and releases them into a collection tray. Sensors detect whether the correct number of pills have been dispensed. Sensors are also used to determine whether or not the patient has taken their medication. If a patient fails to take their medication, or if the machine is malfunctioning, the Noverdose will dial out to an emergency contact and play an error message.

Table of Contents

Abstract ... iii
Table of Contents ... iv
Table of Figures ... v
Table of Tables ... v
Introduction ... 1
1. Master Controller PIC ... 2
2. Mechanical Design ... 5
   2.1. Bulk Bin Removal area ... 7
      2.1.1. Bulk Bin ... 8
      2.1.2. Incline Chute ... 9
      2.1.3. Rotating Wheels ... 10
   2.2. Servo Control ... 11
   2.3. Linear Motion Area ... 13
   2.4. Accept / Reject Area ... 14
3. Serial RS-232 communication with the PC ... 16
4. Voice Playback Subsystem ... 18
   4.1. Voice Sample Storage and Compression ... 19
   4.2. Voice Streamer ... 22
5. The Real Time Clock ... 25
6. Phone Dialout ... 27
7. LED Driver and Control ... 30
8. Uninterruptable Power Supply (UPS) ... 32

Table of Figures

Figure 1. Flow Chart for Main Program That Will Run on the Master Controller PIC. ... 2
Figure 2. Flow Chart for Interrupt Driven Features controlled by master PIC................... 3
Figure 3. Wiring and Timing diagram for IPSP protocol 4
Figure 4. Conceptual overview of the dispensing mechanism 5
Figure 5. Flow chart of a pill moving through the dispensing mechanism......................... 7
Figure 6. Diagram of the bulk bin.. 8
Figure 7. Possible arrangement of eight bulk bins.. 9
Figure 8. Functionality of the gearing systems I... 10
Figure 9. Functionality of the gearing systems II 11
Figure 10. Forward driving pulse train 12
Figure 11. Reverse driving pulse train.. 12
Figure 12. Schematic of the photo diode 14
Figure 13. Reject / Accept platform.. 15
Figure 14. Block diagram showing RS-232 communication.. 16
Figure 15. RS-232 timing diagram 16
Figure 16. RS-232 schematic... 17
Figure 17. Block Diagram of the VPS.. 18
Figure 18. Memory Array. Circuit utilizes a 3to8 MUX to access 7 RAM chips 21
Figure 19. Sound Data Memory Organization... 22
Figure 20. Sound streaming circuit.. 23
Figure 21. Antialiasing Filter.. 24
Figure 22. Schematic of the RTC setup 26
Figure 23. Pinout of the M-8888.. 28
Figure 24. Pinout of the PIC16C84.. 27
Figure 25. Flow diagram of phoning progress.. 28
Figure 26. Schematic of phone dialer 29
Figure 27. LED Driver Setup.. 30
Figure 28. Chained LED shift registers 30
Figure 29. Schematic of the UPS.. 33

Table of Tables

Tables 1a. & 1b. Errors and Corresponding Voice Samples……………..………………..19
Table 2. Voice stitching samples………………………………………………………..19
Table 3. Chips used during backup mode…………………………………….…………32

Introduction

People who suffer from chronic diseases must take prescribed medications, some of which are required to be taken at different times during the day. A vast majority of these people are not taking their medications properly and are consequently put into care homes. Often these patients are visited by home care workers several times a day to have their medication administered. Unfortunately the care homes are full and care workers travelling door to door is a burden on Medicare.

The Noverdose medication dispenser is a device intended to replace the current, inadequate medication scheduling systems such as multi-boxes and bubble sheets. These current systems are often confusing and don't actually remind the patient to take their medication. Patients are alerted when their medication is ready, thereby greatly reducing the chance of an under dose. The Noverdose dispenses a patient's medication at the correct times and in the correct quantities, eliminating the possibility of an overdose.

The purpose of this document is to describe the design specifications of the Noverdose medication delivery system. The design specification for each unit of the system is discussed. The methods by which each unit interfaces with the other sections of the device is also described. The audience for this work is Dr. Andrew Rawicz, Mr. Steve Whitmore, and the design engineers of Prescriptek Health Systems.

1. Master Controller PIC

We are using a number of PIC16F84's and PIC16C84's to control various subsystems of the Noverdose. In order to control all of these sub systems, we are going to use a PIC16F84 as well. This 'Master Controller' PIC will be the main guts of the Noverdose. It will decide what happens and when according to the flow chart in figure 1.

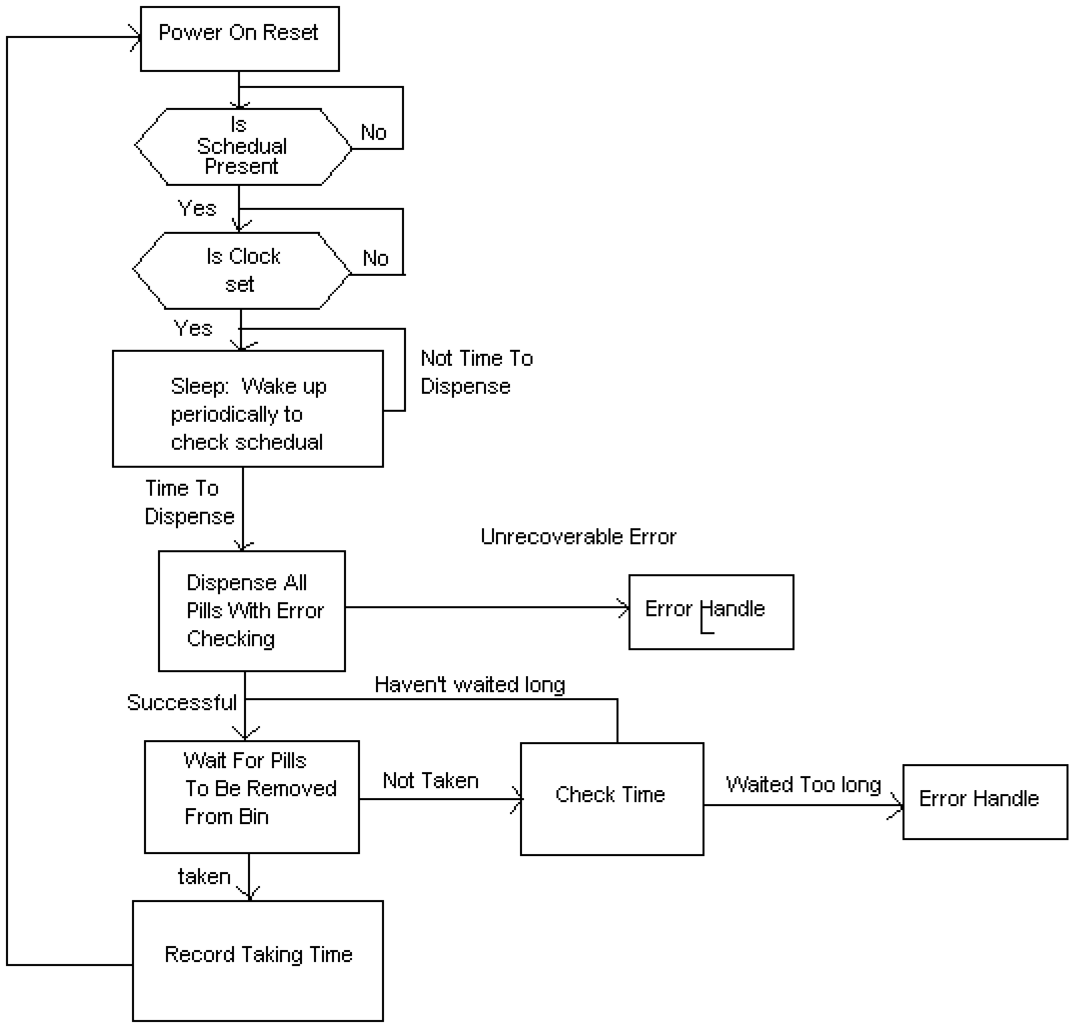


Figure 1. Flow Chart for Main Program That Will Run on the Master Controller PIC.

In addition to the functions depicted in figure 1, the master controller PIC will also deal with two interrupt driven functions described in the flow chart of figure 2.

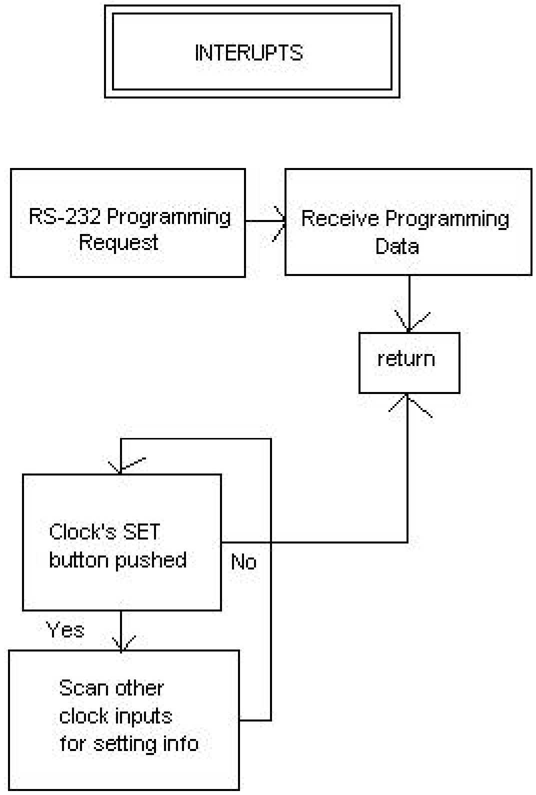


Figure 2. Flow Chart for Interrupt Driven Features controlled by master PIC

.

In order to control all of these different sub systems, the master controller will need a method of talking to the subsystems. If the subsystem is controlled by another PIC, we will use a protocol that we developed called Inter PIC Serial Protocal(IPSP). With this protocal, any two PIC's can be linked together with the use of only two of their Bi-directional pins. Figure 3 shows the wiring schematics for the IPSP protocol.

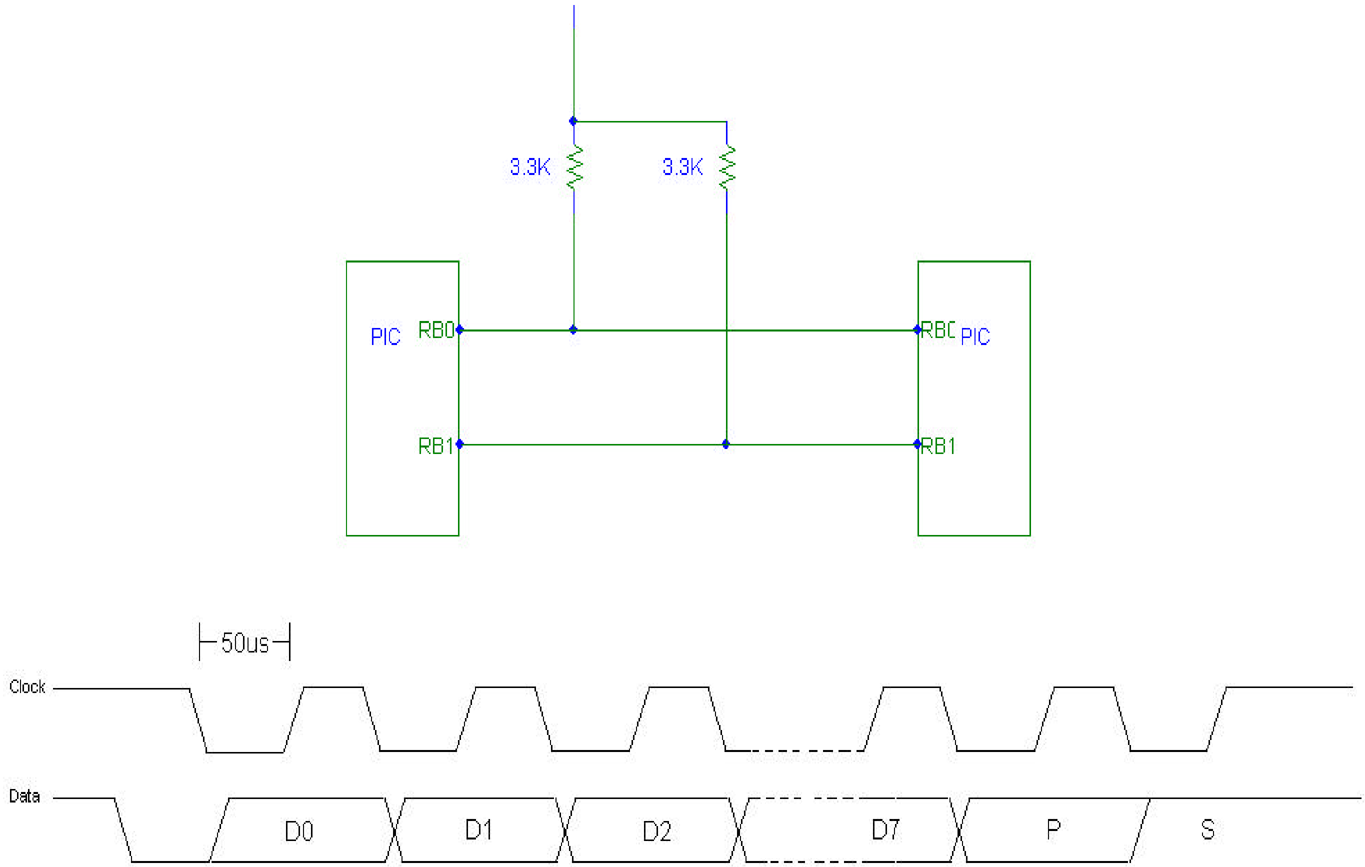


Figure 3. Wiring and Timing diagram for IPSP protocol

.

Figure 3 also shows the timing diagram for the IPSP protocol. The two lines are simulated open collector lines. When a PIC wants to drive one low, it outputs a zero on its pin, thus sinking current. When it wants to drive a line high, it sets its pin to high impedance input and the pullup resistors to +5V will pull the line high. At idle time, the two lines float at +5V. When a PIC wants to transmit, it brings the data line low for 50us and then brings the clock low. The transmitting PIC then changes the data line to the LSB of the byte being transferred and toggles the clock high. The protocol dictates that all data changes occur on the falling edge of the clock and that the data line is guaranteed to be valid when the clock is high. After the MSB has been clocked across the bus, the transmitter clocks a parity bit across the bus. IPSP uses ODD parity. As such, when there are an ODD number of 1's in the 8 data bits, the parity bit will be set. After the parity bit is clocked in, the transmitter produces one last clock pulse clocking in a stop bit (always 1).

Because we are using only two pins for communication, its important that each PIC know if they are transmitting or receiving. As such, we are going to take great care in ensuring that the state of the Noverdose is known and controlled at all times such that only one PIC will ever want to transmit to another PIC.

2. Mechanical Design

The purpose of our mechanical system is to remove solid medications from a bulk dispenser. Each bulk dispenser contains a single pill type so the number of different medications that the pill dispenser delivers depends on the number of bulk bins. The mechanics of the dispenser from the point of filling the bulk bin to the delivery of a pill can be broken into three sub sections. These sub-sections would include the bulk bin removal area, the linear motion area, and the accept/reject area. Figure 4 gives an abstract overview of the components, and how these components will fit together to dispense the medication. Each of the basic components will be described in greater detail in the following sections.

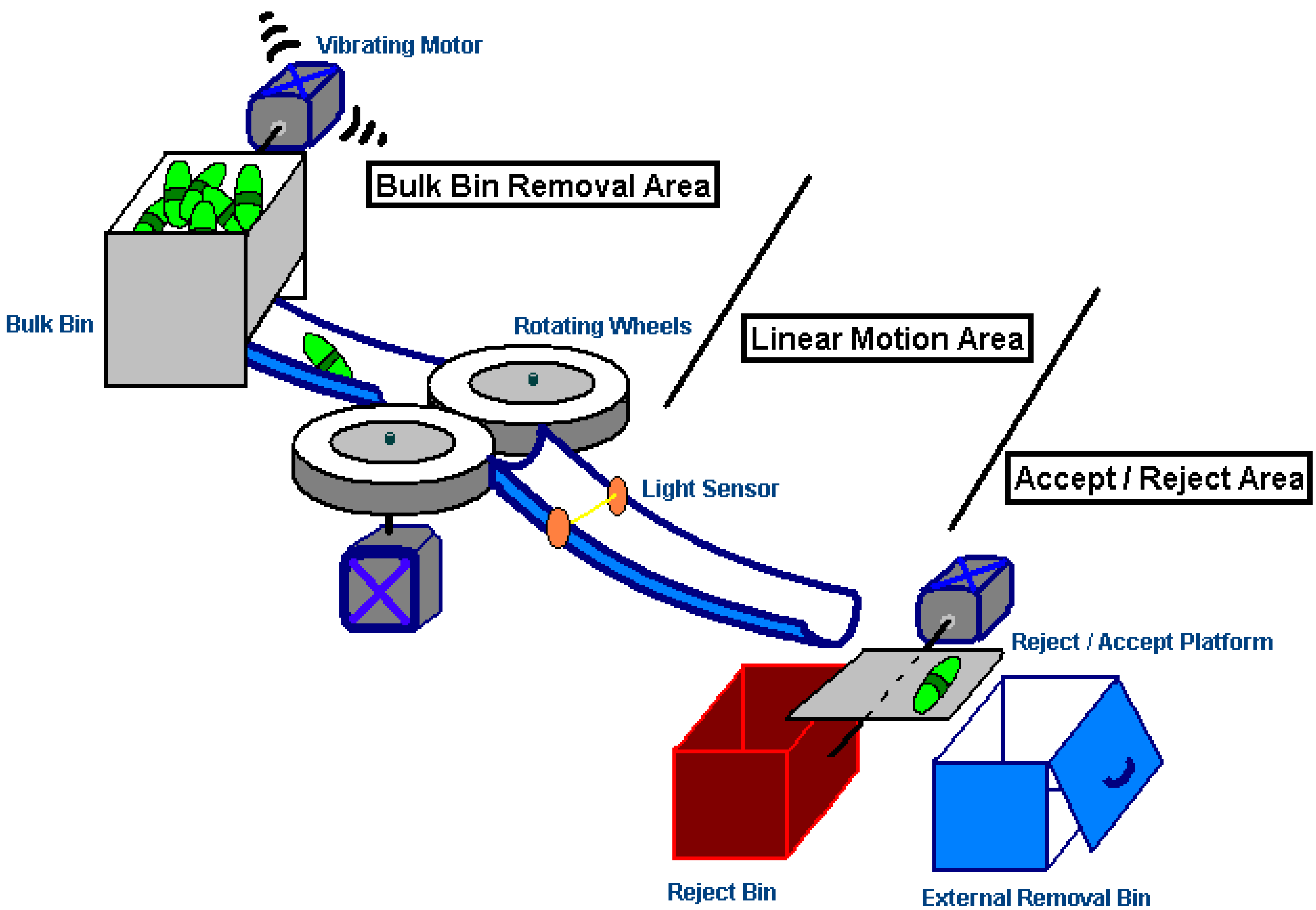


Figure 4. Conceptual overview of the dispensing mechanism

Figure 4 gives a basic overview of the mechanical system. Some of the components of this system may be modified in order to compensate for difficulties in construction and mechanical implications that we did not foresee in the design phase.

The path of a single pill through the dispensing system begins when the pills are first placed into the bulk bin. The pills will then move from the bulk bin into a V-shaped chute. This movement will be facilitated by the vibration of a counter-weighted motor as well as by gravity. At the end of the chute are the rotating wheels that function as a 'door' only allowing a single medication to pass through at a time. After the rotating wheel section is a second chute that passes the medications onwards to the accept/reject tray. This chute contains light sensors that will be used to control the motors, as well as to verify that only one pill has passed by. The pills will then be placed on the reject/accept platform. Depending on weather one or two pills are dispensed, the tray will reject the sample if it has an extra pill and it will accept the sample if there is a single pill. If the pill is accepted it will be placed into a removal tray and the patient will be alerted that their medication is ready. This mechanism of pill delivery will be carried out on each of the bulk bins until the particular dosage is complete. Figure 5 gives a flow diagram of this process.

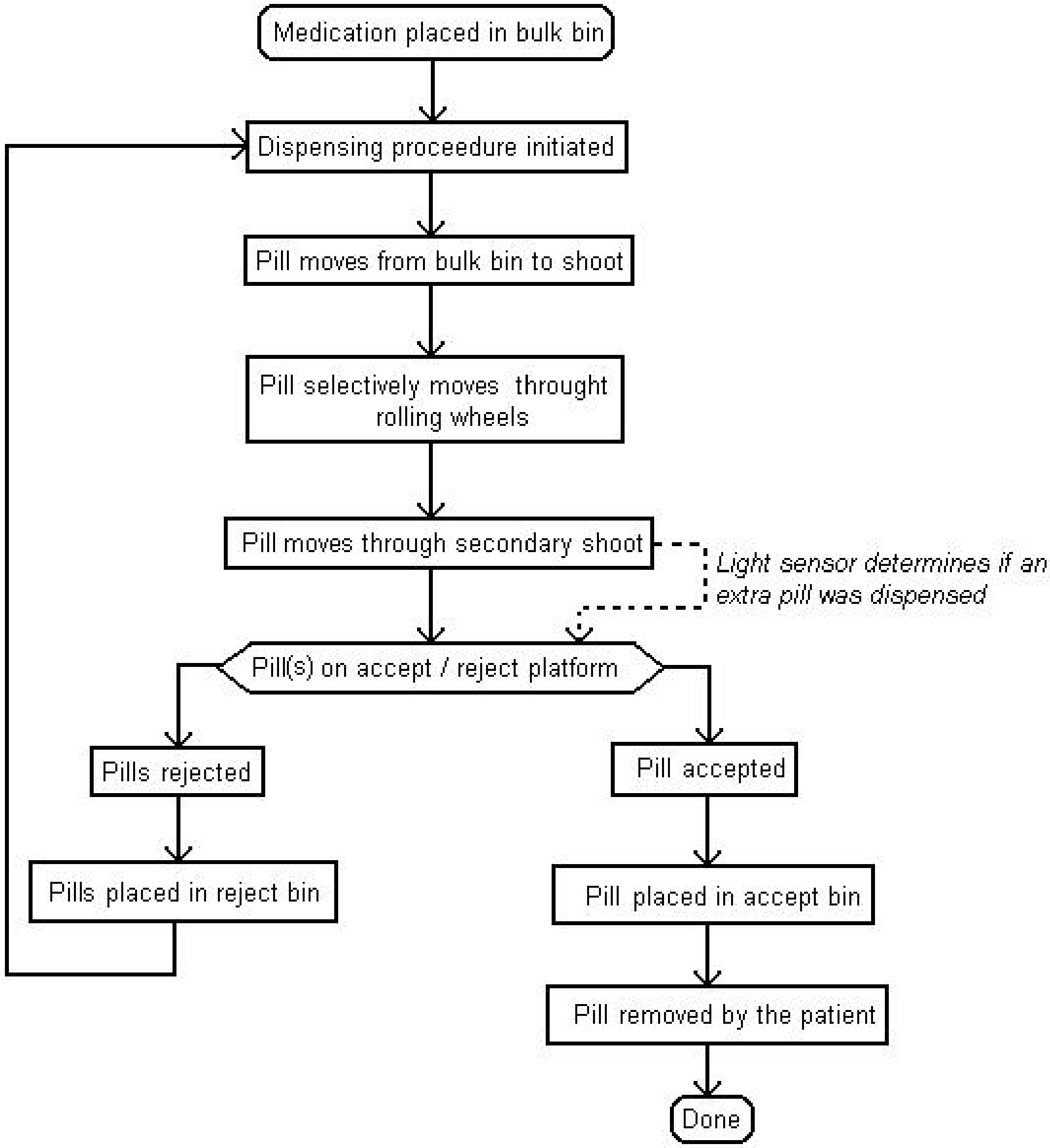


Figure 5. Flow chart of a pill moving through the dispensing mechanism

2.1. Bulk Bin Removal area

The bulk bin removal area contains the actual bulk pill bin, a vibrating mechanism and the rotating wheel 'door' device. The bin will have a chute that exits at a shallow angle and the chute will have the two wheels or 'doors' that will dispense the single pill at a time. A vibrating device such as a counter balanced motor will be used to ensure that the pills are flowing smoothly down the chute and the larger pills are not becoming jammed. In a final product that would is ready for production, any mechanical components that come in contact with the medication will be constructed of a bio-compatible plastic as deemed safe by the health and welfare Canada, and by the FDA (if we wish to market the device in the USA). For the prototype that we are making for our engineering class, we will use materials that are the least expensive and most available.

2.1.1. Bulk Bin

The bulk bin is where the pills will initially sit before they are separated and passed onto the other components in the system. The bulk bin will have a removable or hinged lid that can be removed to allow the health care worker to replace the used medication. The bin will not necessarily be very large, it will be in the 5 X 5 X 6 cm range. Internally, the bin will have two or more ramps that will allow the slowing down, and separating the medication so that it will be travelling in a row. Figure 6 gives a graphical representation of bulk bin.

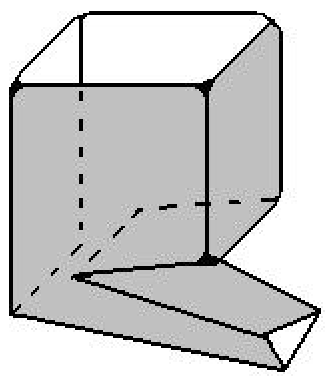


Figure 6. Diagram of the bulk bin

We have some flexibility in the design of the bulk bin. More platforms can be added so that the pills travel a greater distance before they are dispensed. This may help to separate the pills and ensure that they are not stacked upon one another. We also have some flexibility in the location of the bin. It may be possible to place it above the rotating wheels, in order to conserve space.

There are various possible spatial arrangements for the bulk bins. Figure 7 shows one of our main ideas.

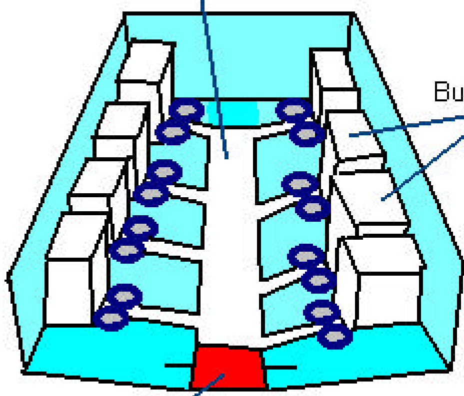


Figure 7. Possible arrangement of eight bulk bins

As seen in Figure 7, the bulk bins will be placed in two rows (Figure 7 shows eight bulk bins). Each of the bins will have a rolling wheel 'door' that feeds into a common slide. There is one reject / accept tray for the entire device.

The prototype that is being built for our engineering class will contain at least one working pill dispenser. We will build the chassis so that it can hold four of these dispensers. Our production product will have eight dispensers, as such the product would be able to distribute 8 different kinds of medication. If a person requires more than eight different types of medication, they would use two of our dispensers.

2.1.2. Incline Chute

The inline chute connects the bulk bin to the rotating wheels. The incline chute will simply be an extension of the bulk bin. Depending on the size constraints, the chute length will vary widely. However the chute itself will be in the form of a V, so that the pills will tend to form a row as they move towards the rotating wheels. The incline of the plane (5 - 30degrees) and the roof on the chute will stop the pills from stacking on one another. The roof on the chute will be high enough to allow the largest pill to pass through on its side. This may still allow the smaller pills to become stacked, so the incline of the chute must be small enough to avoid this. Another possibility is to have the chute roof height specific to different sizes of pills. Therefore, the dispensers would be made specifically to accommodate different pill sizes.

2.1.3. Rotating Wheels

The rotating wheels are to be made from a soft, medically certifiable foam like substance that will grip the pills and at the same time be gentle enough not to damage them. The wheels will move apart, thereby allowing the medication to pass by. The wheels will be turned very slowly to help control the flow of the pills and not allow an extra pill to pass by. A light sensor in the chute beyond the wheels monitors the exiting pills. Once the desired pill has been dispensed, the wheels are reversed to push any excess pills back up the chute.

The prototype will use servos to control the movement of the wheels. It is desirable to use only one servo for each set of wheels in order to reduce that cost of our product. We have proposed the following two methods to accomplish this: a simple gearing mechanism and large tooth gears. Figures 8 and 9 show these two gearing mechanisms.

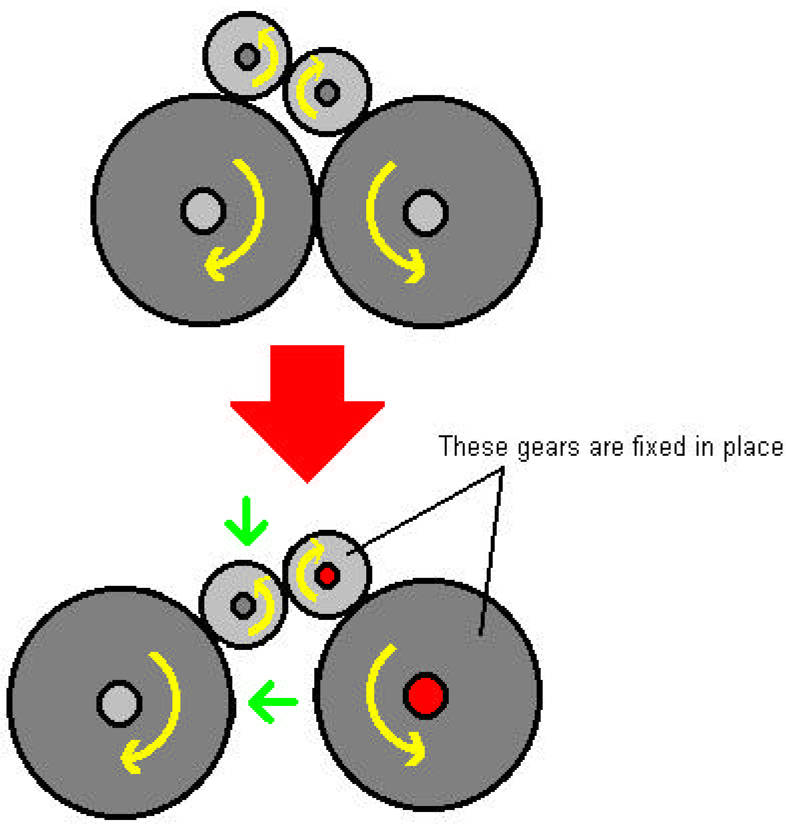


Figure 8. Functionality of the gearing systems I

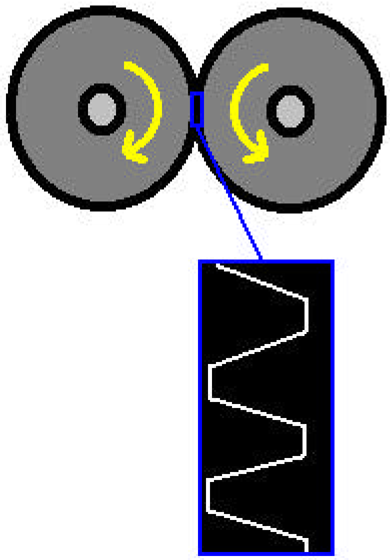


Figure 9. Functionality of the gearing systems II

The gearing system that requires four gears (Figure 8), is the most effective of the two designs, but the added gears makes it more difficult to build. Two of its gears are anchored in place allowing only rotational motion while the other two gears are moving along tracks and rotating at the same time. We are using two 87pt 42 pitch spur gears and two 30pt 42 pitch pinion gears to test this system.

The gearing system using large toothed gears (Figure 9), is the simplest to build, but it may be fallible. Pills that are too large may expand the wheels past the point where the gear teeth can grip. We are using two 30 pitch gears to test this functionality. The pill may be far enough between the wheels to pull it through, but it is possible that the pill would become jammed.

Both of these systems will be tested to determine which will be used in the medication dispenser.

2.2. Servo Control

The dispenser system of the Noverdose uses Futaba servo motors to drive the dispensers. These servos are controlled by a PIC chip which sends the driving information to the proper motor through a multiplexer. The PIC is ordered to dispense a pill from a certain motor by the master controller via the Inter-PIC Serial Protocol bus. The PIC will then begin an output routine designed to drive one of the motors. The routine is the same for all motors; the multiplexer routs the data to the motor chosen by the master controller.

Futaba servo motors use pulse width modulation to choose the direction and speed at which they rotate. To drive a servo forward, a pulse width of 1.25ms is required. To drive a servo in reverse, a pulse width of 1.75ms is required. Pulse widths between 1.25 and 1.75ms will cause slower speeds in their respective direction. A pulse width of 1.5ms will cause the servo to stay in a neutral position. Pulse widths of < 1.25ms or >1.75 ms will also result in no movement.

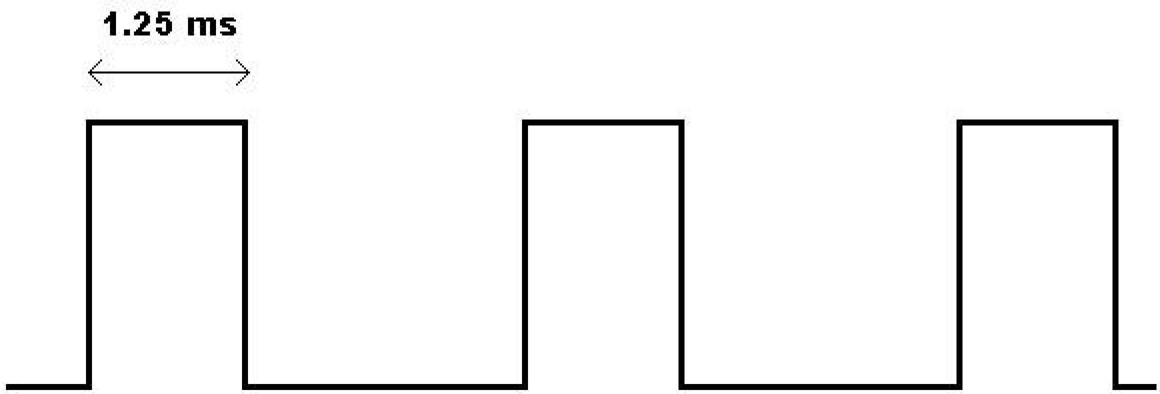


Figure 10. Forward driving pulse train

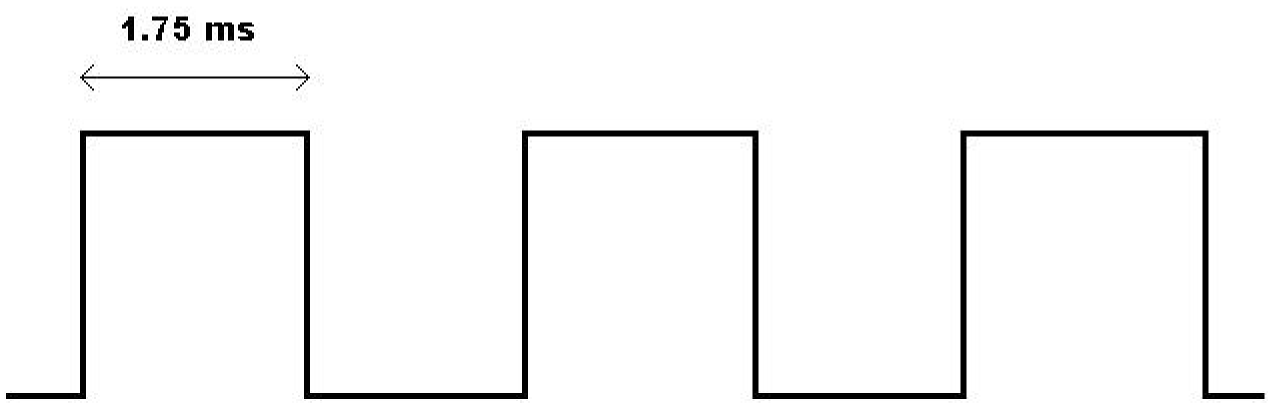


Figure 11. Reverse driving pulse train

The dispensing routine slowly drives a servo forward until a sensor beam is broken, indicating a pill has dropped. The motor is then driven full speed in reverse to drive any pills left in the dispenser back out of the dispensing rollers. If the sensor beam is broken again while the motor is reversing an error code is sent to the master controller because a second pill has dropped. If the sensor is not broken a second time, then an OK signal is sent to the master controller.

2.3. Linear Motion Area

The linear motion area is composed of a slide and the light sensor. The actual slide does not need to be in a V-shape like the slide before the roller wheels, it will be have a flat bottom so that the transition from the wheels to this slide will be as smooth as possible.

An infrared emitter and photodiode will be used to monitor the passage of medication. The emitter and photodiode will be located across from one another at the exit of the rolling wheels for each of the separate bulk bins. The sensors will be able to detect the passage of one or more pills and transmit the information to the PIC.

When the emitter is active the photodiode will show little resistance between its two terminals. When the something passes between the photodiode and the emitter the resistance becomes very large. This behavior can be utilized to switch the power between 0 and 5 volts. Figure 12 shows the schematic of photo diode.

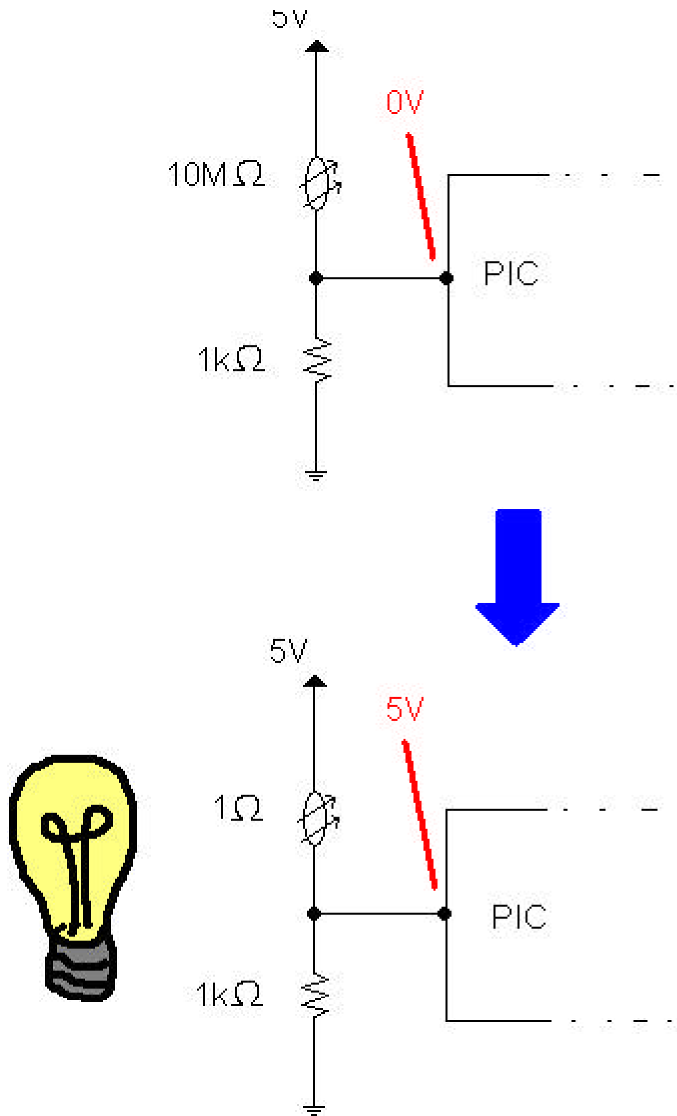


Figure 12. Schematic of the photo diode

2.4. Accept / Reject Area

The accept/reject area is composed of the accept/reject platform, the rejection bin and the removal tray. Any dispensing procedure that results in more than one pill ending up in this area will cause the pill to be rejected. If the correct amount of pills is delivered the medication will be placed into covered dish that the patient can remove the medication from.

The accept / reject platform has three possible states. These states are neutral, and the angled accept or reject state. The platform will rotate on a central axis and it will be driven by two solenoids into the different states. Figure 13 shows how the platform will be setup.

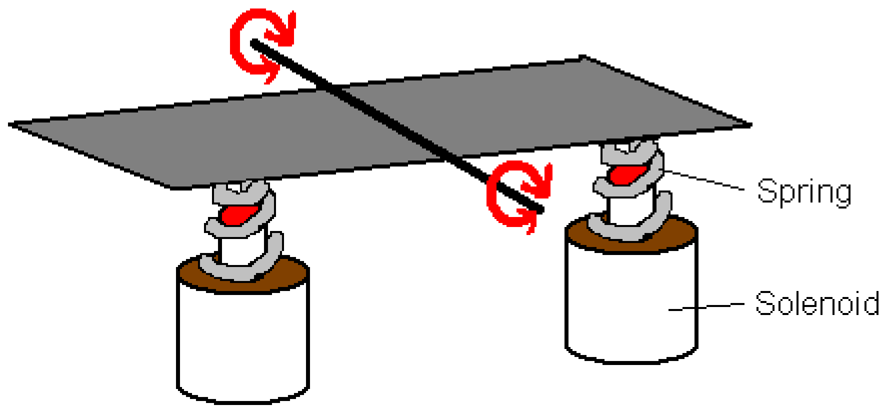


Figure 13. Reject / Accept platform

As can be seen in the above figure the solenoids are located on either side of the platform. The springs that are shown pull the solenoids back to the neutral position after one of the solenoids has been energized. The solenoids will be controlled by the PIC indirectly through a power MOSFET since the PIC cannot source enough current to drive a solenoid. The PIC will determine which of the solenoids to energize depending on feedback from the light sensor.

3. Serial RS-232 communication with the PC

A pharmacist using a PC will setup the medication schedule for the pill dispenser via a graphical user interface. After the schedule has been properly organized, it will be ready to be sent to the Noverdose unit. The data is sent to the PIC by communication lines and the schedule information is processed and stored in memory. The dispenser will then send the schedule back to the PC for comparison to confirm that the correct schedule has been sent. This process will require communication between the pharmacist's PC and the pill dispenser. We are using the asynchronous RS-232 standard communication to complete this task. RS-232 was chosen because it is an older and well-documented communication protocol, and it will only require two I / O pins to implement. Figure 14 is a block diagram showing the communication between the pharmacist's PC and the medication dispenser.

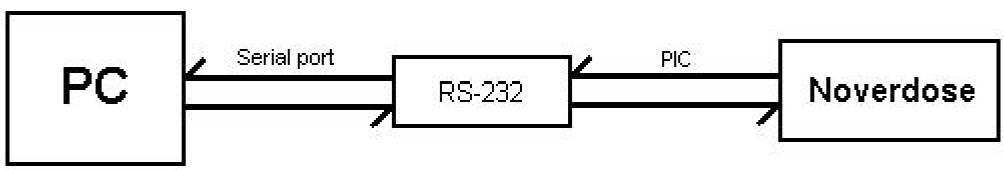


Figure 14. Block diagram showing RS-232 communication

Since RS-232 is asynchronous, both the computer and the PIC must be communicating at a constant frequency (in our case 9600bps). The actual data stream will consist of one start bit, one stop bit and no parity. Figure 15 shows the timing diagram for RS-232 communication.

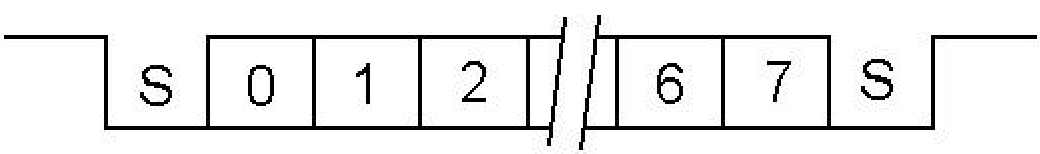


Figure 15. RS-232 timing diagram

A MAX232 chip is being used to change between the RS-232 voltages and the TTL levels of the PIC. We are currently using a null modem serial connection, as such the hand shaking lines are tied back, and only the receive data (RX) and transmit data (TX) are used. RX will be tied into an interrupt pin of the PIC (i.e. PORTB pin8), thereby allowing the checking for incoming data. Figure 16 gives the circuit diagram of our current RS-232 connection.

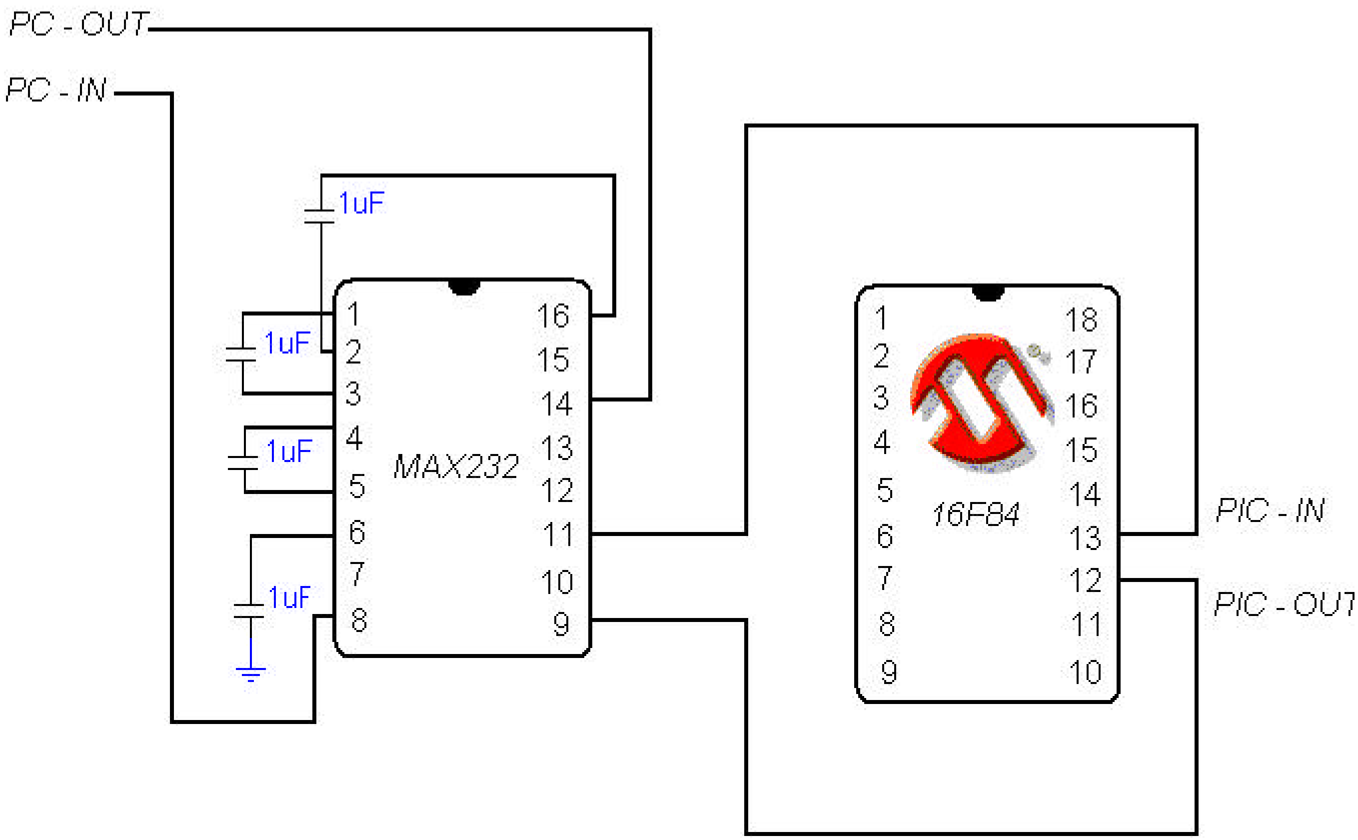


Figure 16. RS-232 schematic

We have written and tested code that can be used to send and receive data from a PIC 16F84. However this code may become unnecessary if we move to a larger microcontroller that has a serial module on board.

4. Voice Playback Subsystem (VPS)

When a Noverdose™ malfunction has been detected, the master controller PIC will enable the dial-out subsystem. Upon contact of the emergency contact person, the master controller PIC will tell the VPS to play a certain sequence of voice samples that corresponds to the malfunction detected. The block diagram in figure XX visualizes the operation of the VPS.

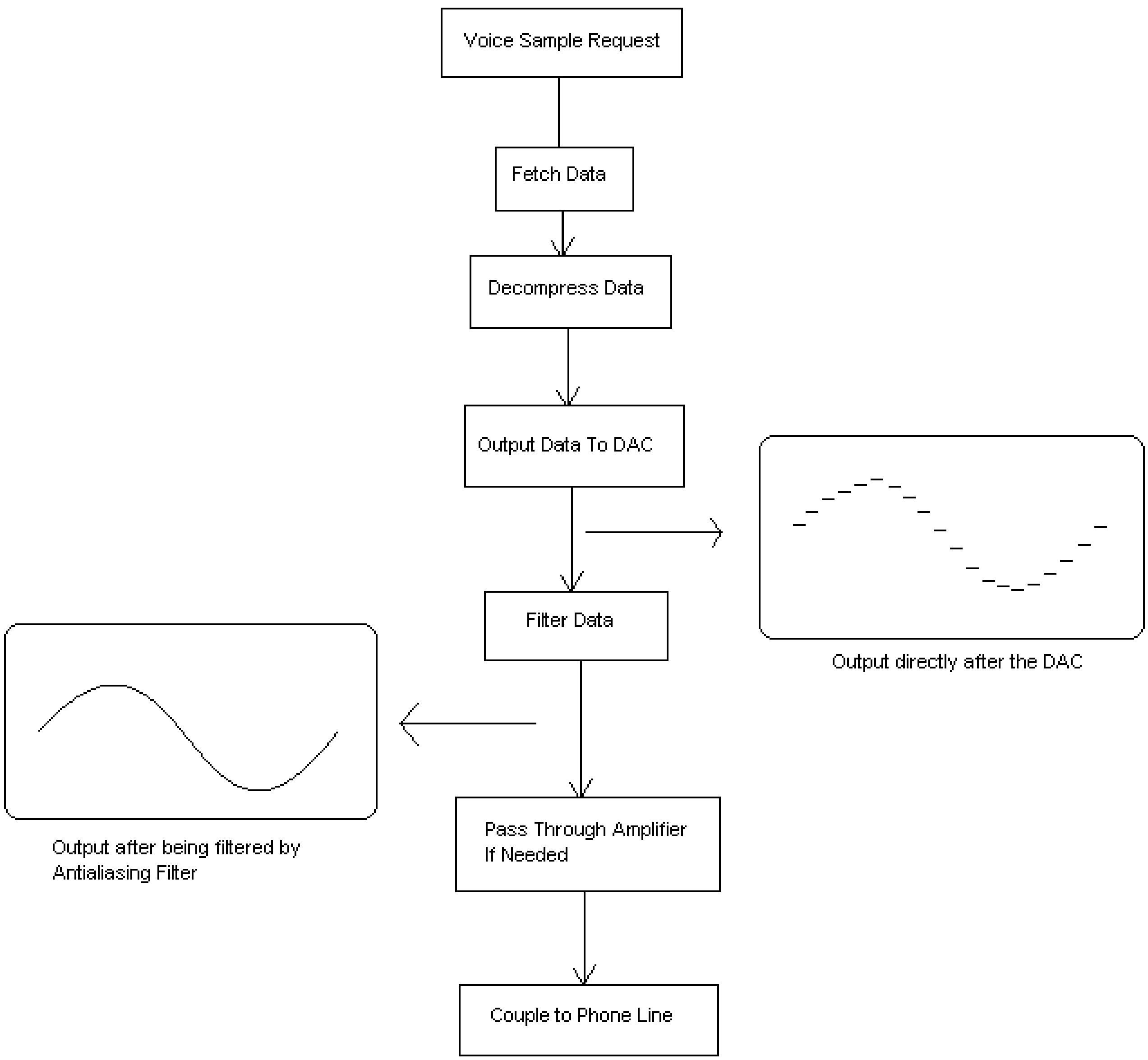


Figure 17. Block Diagram of the VPS

4.1. Voice Sample Storage and Compression

The Noverdose™ will be capable of detecting several errors. Table 1 shows the errors/malfunctions that will be detected along with their corresponding VPS voice samples.

Table 1a. Errors and Corresponding Voice Samples

Error Detected	Voice Sample
Dispenser Malfunctioned: i.e. it repeatedly failed to dispense a single pill (i.e. it continuously dispensed two or more pills).	Noverdose Malfunction Detected
Patient Non-Compliance: i.e. the Noverdose™ unit has detected that the user has not taken their medication within a specified time window.	Noverdose Patient Non-Compliance Detected
Noverdose™ unit was unable to dispense any pills of a certain type (device malfunction or pill bin is empty).	Noverdose is Empty

The Noverdose™ will also play several other voice samples described below in **Table 1b**.

Table 1b. Other Voice Samples that will be played

Event	Voice Sample
Greetings to emergency contact person	Greetings from Noverdose Unit number <XXXX>
Request confirmation from emergency contact person	Push the pound sign to confirm this warning.

Since memory isn't free, we are planning on conserving on memory space by stitching voice samples together. Table 2 contains all the voice samples that will be stored in memory along with their estimated length in seconds.

Table 2. Voice stitching samples

Voice Sample	Estimated Length
Numbers: 1,2,3,4,5,6,7,8,9,0	0.5s each ➔ Total = 5s
Noverdose	0.5s
Detected	0.5s
Patient Non-Compliance	1.0s
Is Empty	1.0s
Greetings From	1.0s
Unit Number	1.0s
Push the pound sign to confirm this warning	4.0s
TOTAL	14.0s

As can be seen, we will require 14.0s of audio data. At 8bit 8Khz, that would require 112Kb of memory. That is more than we want to use. As such, we are going to sample all of our data at 4Khz. We will then convert it to 4bit resolution. We are going to try several conversion methods including Linear and logarithmic mapping. We will also delta encode the audio data (a simple compression scheme where you start with a value and then store all succeeding values as differences of the first data value). By using these two methods, we will achieve approximately 8 to 1 compression on our audio data. As such, we'll require only 14KB of ram.

The VPS will store the vocal data in an array of non-volatile ram. We will be using I^2C serial memory. We are considering either EEPROM or FRAM (Ferroelectric RAM). While EEPROM is slightly cheaper than FRAM, FRAM is much faster than EEPROM's and for our voice playback, it may be an issue. Since the two memories abide by the I2C standard, we can design the memory subsystem with only one consideration. FRAM (a product by Ramtron) has a much faster write cycle than EEPROM's. In order to maintain compatibility between FRAM and EEPROM in our code, we must be sure to include the EEPROM write complete polling in our code even though it will not be used when using FRAM – FRAM will always return a done condition when polled for write completion. FRAM, since it is a new technology, offers a largest package size of 16Kbits for its I^2C compatible chips. Even with the audio compression we would require 7 of these 16Bbit chips. Since we were able to secure an excellent deal on these FRAM chips, we have decided to go through the minimal effort to link them into an array of memory as seen in Figure 18.

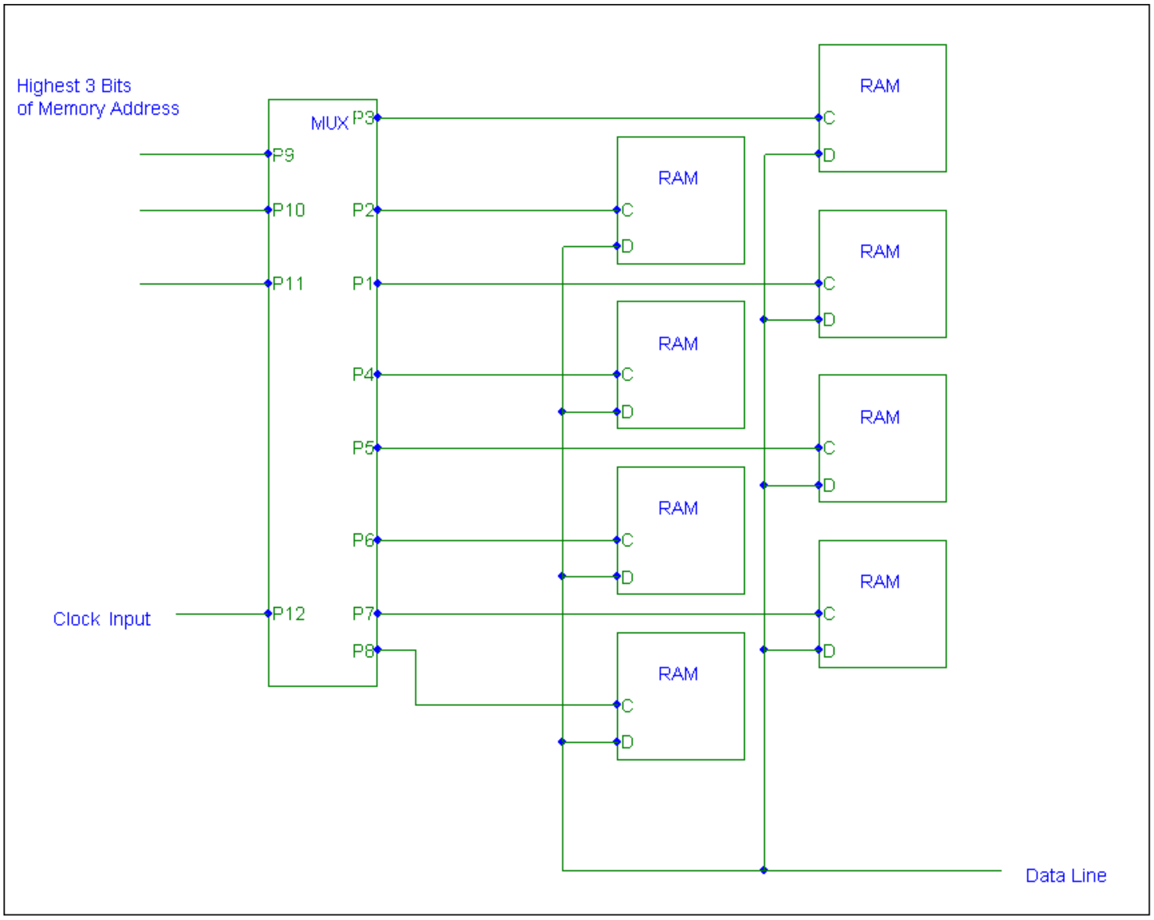


Figure 18. Memory Array. Circuit utilizes a 3to8 MUX to access 7 RAM chips
.

We may convert to a single chip solution later on by switching to a 128Kbit or 256Kbit 3 wire SPI serial EEPROM. Semiconductor companies that produce FRAM are planning on releasing 256Kbit SPI chips in the 4th quarter of 1999. In our final product, we will probably switch to a single chip solution due to cost considerations. However, since we sourced our FRAM as well as logic chips for next to nothing, we decided to use 16Kbit FRAM.

Data will be stored on the ram chips in a very simple and elegant way. Firstly, the main controlling software will keep track of the start location of the sound data (this will typically be zero). At that location the sound data will begin and will be organized according to Figure 19.

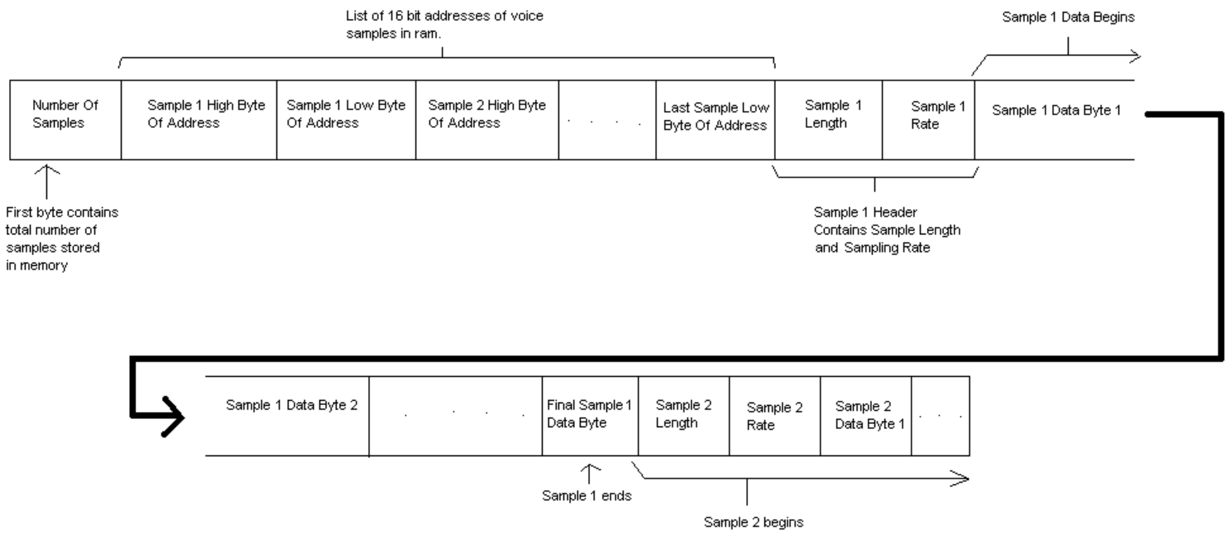


Figure 19. Sound Data Memory Organization

4.2. Voice Streamer

With the data stored in non-volatile ram in a digital format, it will be a simple matter to convert that data to an analog, audible signal. We are planning on using National Semiconductor's double buffered DAC0830 – an 8 bit parallel load DAC capable of running off of a single power supply 0-5V. We are considering, however, simply constructing our own 8 bit DAC out of a R-2R resistor network. If we end up using a linear mapping compression scheme from 8 bits to 4 bits of sample data, then a 4 bit DAC could be used, thus simplifying the circuit even further. Figure 20 shows the sound streaming circuit.

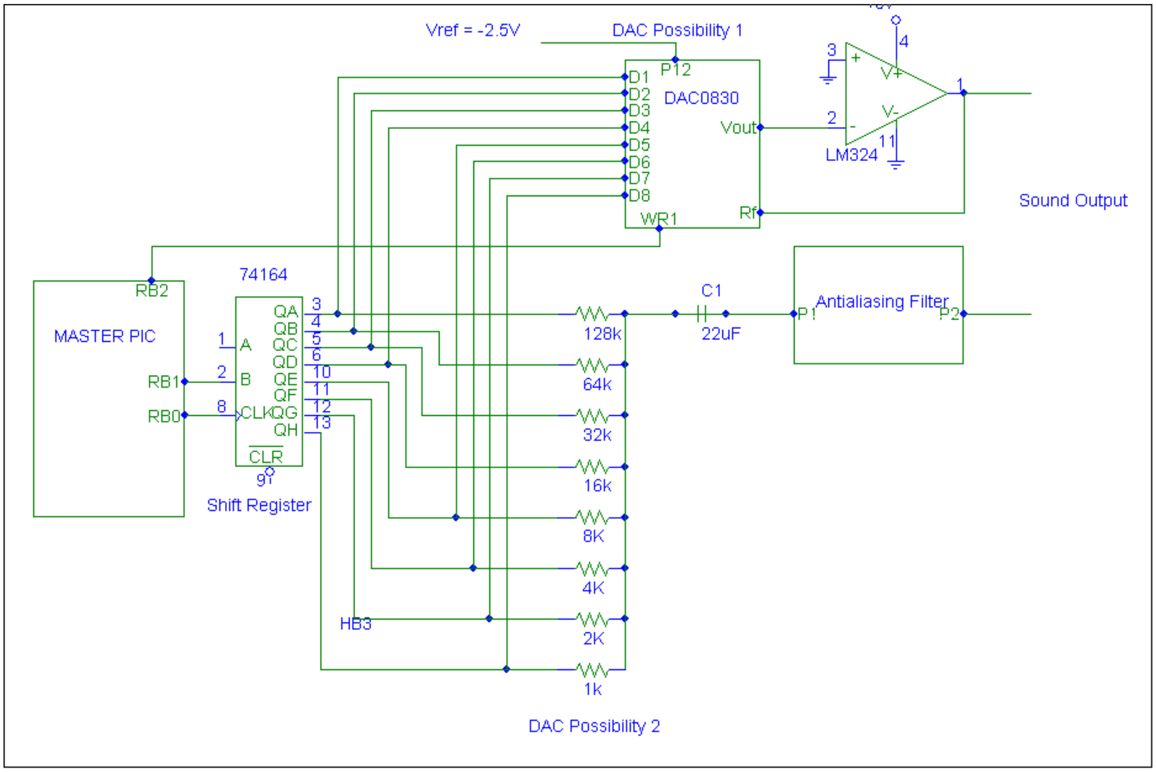


Figure 20. Sound streaming circuit

.

The master PIC streams audio through a serial shift register so as to conserve on output PIC pins. The output of the shift register goes into the DAC. Figure 20 shows two DAC's. One is a commercially available DAC IC. The other is an R-2R resistor network that is effectively a DAC. The digital signals on the left side of the resistors causes various currents to be generated across the resistors. There is less resistance on the resistors corresponding to the more significant bits of the R-2R DAC so as to generate more current when those lines are HIGH and consequently give them more weight in the final output. The antialiasing filter we will try using is shown in figure 21.

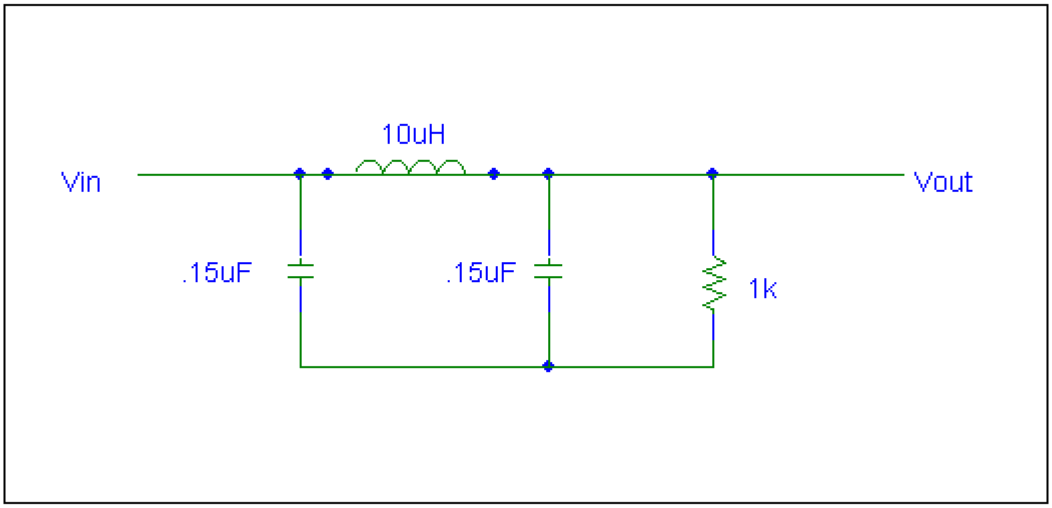


Figure 21. Antialiasing Filter

With the components in the antialiasing circuit of figure 21, it should filter out all frequencies above 5kHz. We will test this for sound quality and redesign if necessary.

5. The Real Time Clock

The Noverdose medication dispenser distributes medication based on a schedule that is stored in the RAM. In order to distribute the medications at the correct times, a real-time clock must be implemented. In most cases the clock will be programmed at the pharmacy via RS-232 port at the same time the schedule is uploaded. Another added option is the external switch system that allows a health care worker an easy way to set the clock without requiring the dispenser to be taken back to the pharmacy for reprogramming.

Using the hardware already existing on the PIC, a clock can be built which runs in the background of the main program. This requires the functionality of the TMR0 register as well as the prescaler bits. After the prescaler bits have been set to their max values, the TMR0 register can count to 65,535. At a 4Mhz clock speed this corresponds to 65.535us. To simplify the calculation of the time the TMR0 register will be reset every 50ms, this will allow for 50,000 instructions before the clock must be checked again to increment a counter. Twenty 50ms cycles is the equivalence of one second. If the main program executes more than 50,000 instructions before checking the clock, a second may be missed. It is possible that if a longer interval of instruction is required, the TMR0 overflow bit can be used as the 17^{th} bit, allowing another 50,000 instructions before the clock must be tested.

The prescaler is an eight-bit register that determines the ratio of the TMR0 register and the clock cycle. However this prescaler is not readable or writeable, as such the 50ms point cannot be determined directly. This difficulty can be accounted for by determining when the upper word is about to roll over to the 50ms point. When the bit rolls over it is easy to determine the actual time using a simple subtraction.

The clock will keep track of seconds, minutes and hours depending on the need of the schedule and the frequency of dosages. If it becomes necessary for a schedule the weeks and months will be counted as well.

As indicated above it is necessary to have the ability to set the clock externally while the device is inside the patient’s house. The clock set mechanism will require two buttons. The first of the buttons the S1 key selects which of the units are to be modified. The selected units are to be blanked for a second and flashed for a second. The S2 button will be used to increment the selected units. The selected unit will flash for about ten seconds then it will return to clock mode. Figure 22 gives shows the circuit setup of the clock reset.

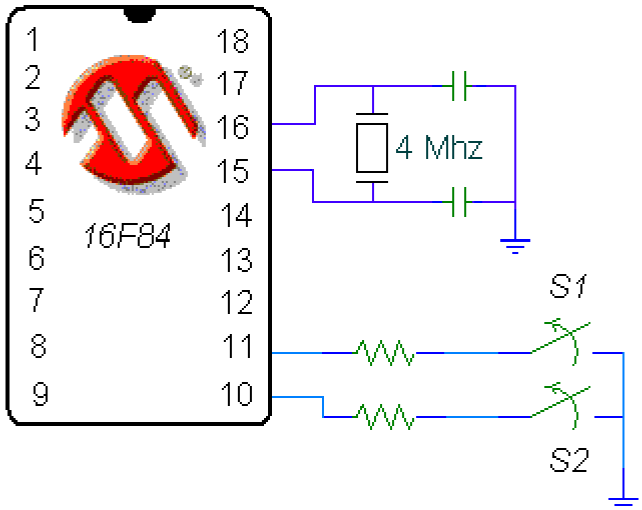
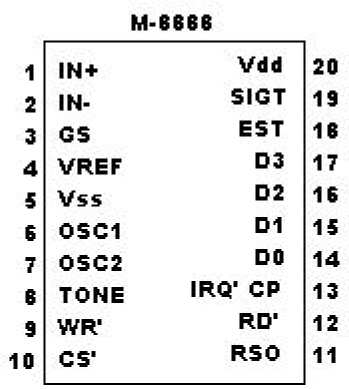


Figure 22. Schematic of the RTC setup

6. Phone Dialout

The telephony functions of the Noverdose will be implemented using a PIC16C84 micro-controller and a Teltone M-8888 DTMF chip. The M-8888 is a Dual Tone Multi-Frequency chip capable of generating and receiving call progress tones. These two chips perform all of the dialing and telephony related functions of the Noverdose system.



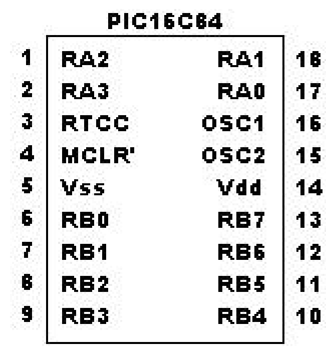


Figure 23. Pinout of the M-8888

Figure 24. Pinout of the PIC16C84

A phone request is first received by the PIC from the master micro-controller via the IPSP bus. The PIC responds by raising pin 17, RA0, high, signaling it is ready to receive a telephone number. The phone number is received one digit at a time on the four data pins numbered 6 to 9. After each digit is successfully received, the PIC pulses pin 17 low to inform the master controller that it is ready to receive another number. When the master controller is finished sending digits to the PIC it drops pin 3 to logic low to signal the end of the phone number.

The PIC now checks if the phone line is currently in use by measuring the voltage on the phone line. This is easily accomplished because the phone line stays at a nominal voltage of 48V when not in use; the line drops to 5V when is use. If the phone line is not being used, the PIC takes the phone off the hook by raising pin18, RA1, high – thereby loading the line and allowing dialing to commence. The PIC begins sending digits to the M-8888 on the four data pins numbered 6 to 9. The M-8888 uses a 3.579 MHz crystal to generate very precise dual frequency tones corresponding to each digit of the phone number. When all of the digits of the phone number have been sent to the M-8888, the PIC raises pin 1, RA2, high to place the M-8888 into call progress tracking mode.

If a busy signal is received, the PIC hangs up the phone by dropping pin18, RA1, low – thereby turning off the n-channel MOSFET, Q1, and removing the load on the line. The dialing process now begins again. Ten redial attempts will be made before the PIC gives up and sends an error code to the master PIC. If a ringback tone is detected on the line, the PIC waits for ten rings before giving up and redialing. Figure 25 displays a flow diagram of the calling process

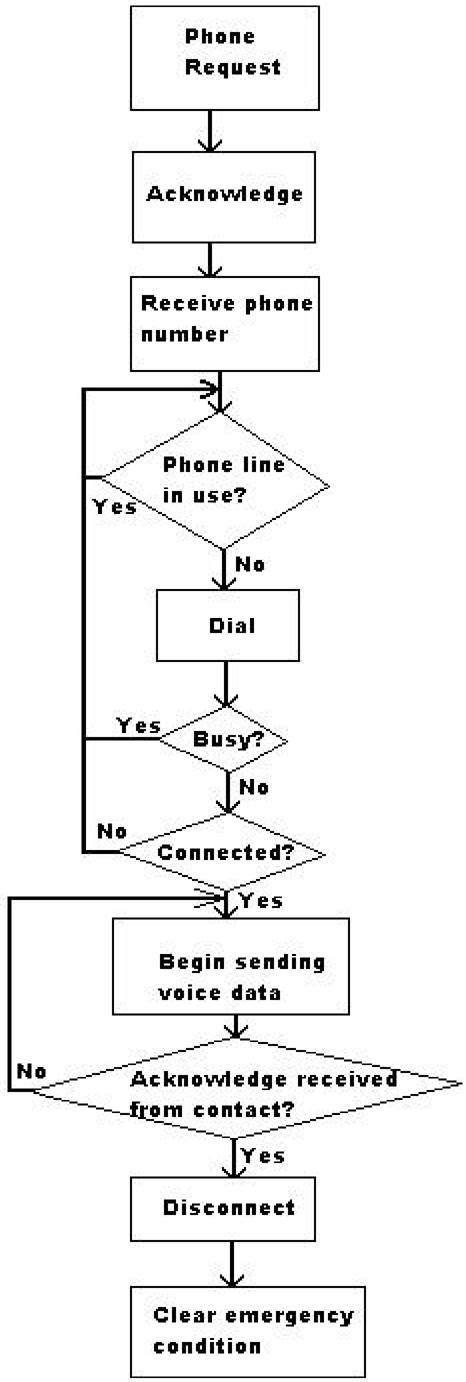


Figure 25. Flow diagram of phoning progress

If the recipient of the call picks up the phone, the M-8888 detects that a connection has been made. The PIC now signals the master controller to begin putting voice onto the phone line. The recipient of the call is played a short message identifying the patient, explaining the nature of the emergency, and asking the call recipient to press * on their touch-tone phone (e.g. "Patient number 354 has failed to take their medication. Press star to confirm you understand"). While the voice sample is being played on the line, the PIC polls the M-8888 to detect if a "*" tone has been received to confirm the call. If confirmation is received, the PIC disconnects and instructs the master controller to clear the emergency condition.

If no confirmation is received, the message is replayed once more. If confirmation is still not received, the PIC begins the redialing sequence.

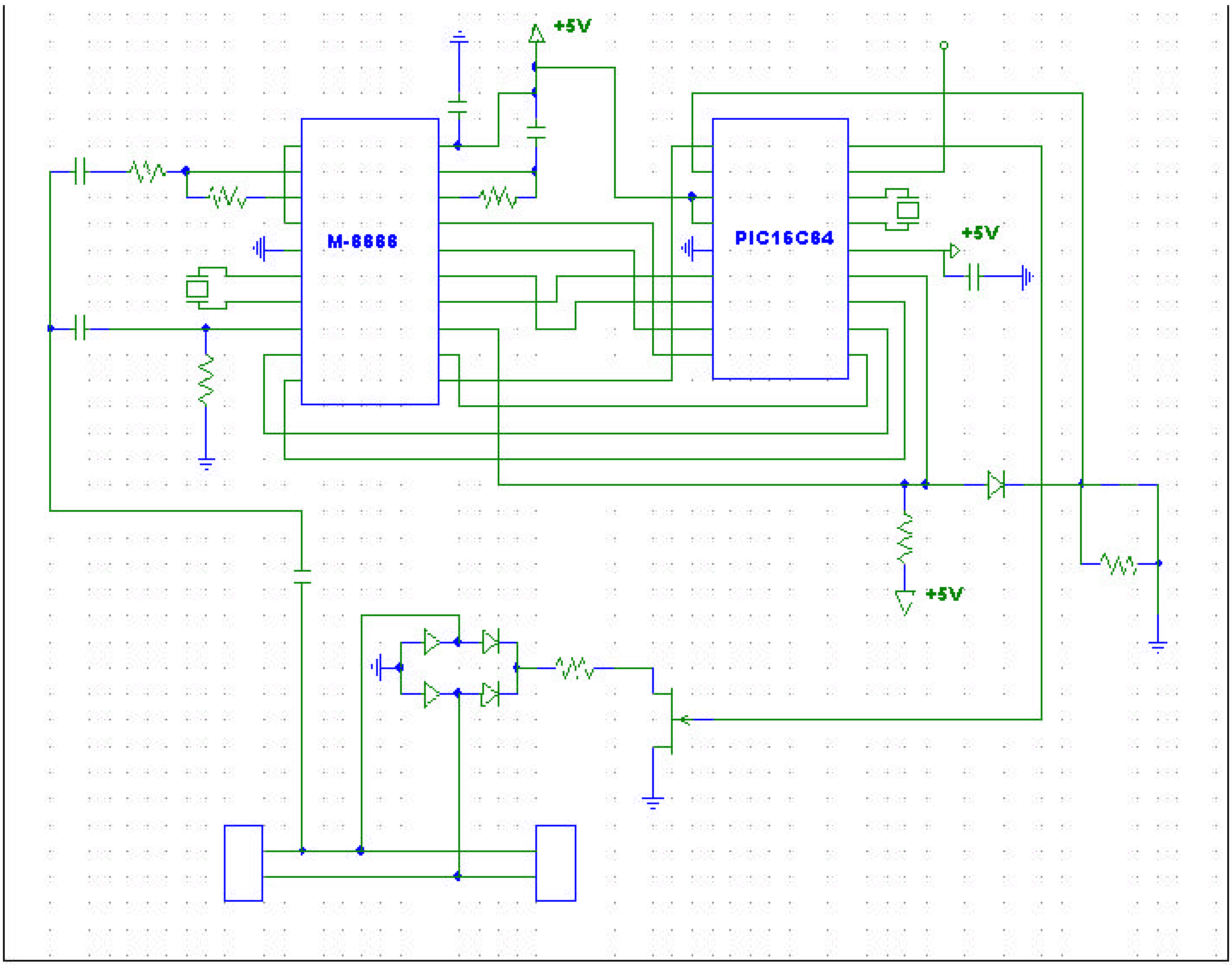


Figure 26. Schematic of phone dialer

7. LED Driver and Control

The main output to the patient will be done through a set of LED's mounted on the Noverdose unit. These LED's will display the current time and when medication is to be taken they will display the number of each type of pill that should have been dispensed. The LED's will be hooked up in series and will be latched and driven by 8-bit shift registers. This design was chosen because of the low number of output pins required from the master controller. Using a serial connection all of the LED's can be set from 2 output pins on our MCU. One pin will act as a clock for all the shift registers and the other pin will act as the input for the next bit to be shifted in. See Figure 27.

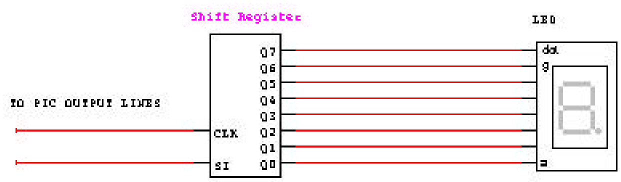


Figure 27. LED Driver Setup

The number of pills of each type is sent as a continuous bit-stream into the first shift register and is sequentially shifted through the chain of 8-bit shift registers until all of the data has been placed into the registers. The LED's are lit when pills have been dispensed. The LED's continue displaying the numbers of pills dispensed until ten minutes after the Noverdose detects the pills have been taken.

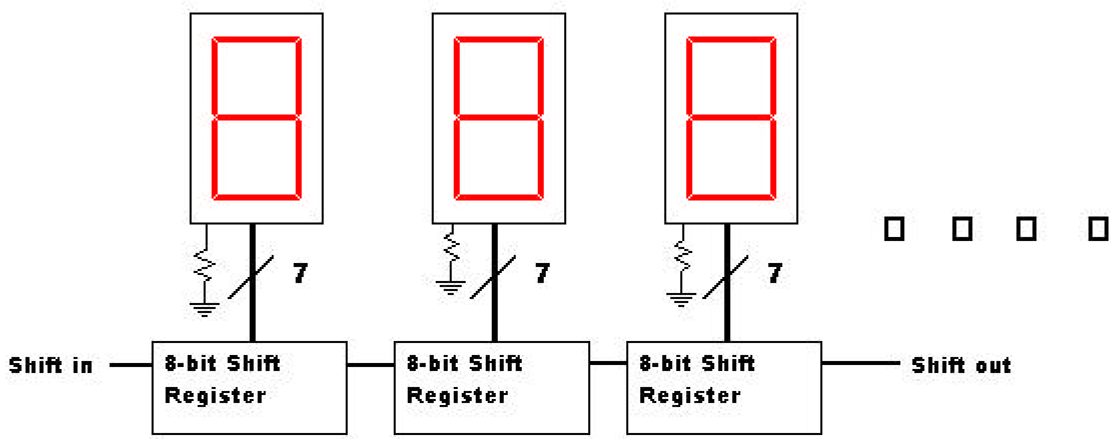


Figure 28. Chained LED shift registers

When not displaying pill information, the LED's will be used to display the time of day in hours and minutes. This time is obtained from the controlling PIC, which will have an algorithm for keeping time using the PIC's internal timer. The time display will be updated every minute.

8. Uninterruptable Power Supply (UPS)

The Noverdose will run on an uninterruptable power supply at all times. This UPS will eliminate prevent scheduling information from being lost in the event of a brownout or total power outage. The Noverdose will continue to function for two weeks with no external power supply. When running on backup power, the Noverdose will run in a minimal mode so as to use as little power as possible. The LED displays will be shut off entirely, as will the shift registers and PIC16C84 which control the LED's. Below is a table of the chips which will be used in backup mode and the operating voltages and currents they typically run at.

Table 3. Chips used during backup mode.

PART	QUANTITY	Typical Volts	Typical Current
PIC16C(F)84	3	5.0 V	1.8 mA
Teltone M-8888	1	5.0 V	10 mA
SN74164 Serial Shift Register	4	5.0 V	30 mA
DAC0830 8-bit DAC	1	5.0 V	1.2 mA
FM24C16 16Kb FRAM	7	5.0 V	115 uA
ADM223 RS-232 Driver Chip	1	5.0 V	2 mA

From the above table, a maximum power dissipation of approximately 700mW could be drawn if all of the above devices were operating at the same time. However, many of the above devices can operate in sleep mode when not in use, using much less power. At any given time in backup mode, the Noverdose will likely not draw more than 200mW. Two 9-volt batteries will be sufficient to power the device at these levels for two weeks. The dispensing motors will still function in backup mode, but only for five-minute periods, therefore not taxing the power stores too heavily.

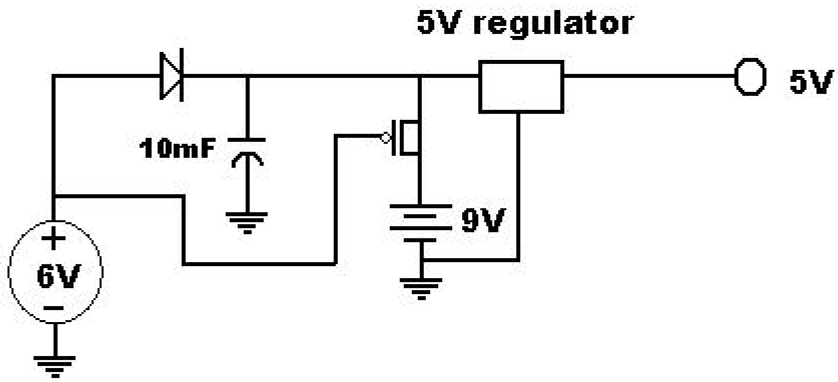


Figure 29. Schematic of the UPS