A MICROPROCESSOR BASED SPEED AND CURRENT LEVEL CONTROLLER FOR A VARIABLE MUTUAL RELUCTANCE MACHINE

by

WILLIAM ROBERT GANDLER

B.S., Massachusetts Institute of Technology

(1974)

SUBMITTED IN PARTIAL FULFILLMENT OF THE

REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

July, 1982

Signature of Author: .....
Department of Electrical Engineering and
Computer Science, July 8, 1982

Certified by: .....
Richard D. Thornton
Thesis Supervisor

Accepted by: .....
Arthur Smith
Chairman, Department Committee on Graduate Students

A MICROPROCESSOR BASED SPEED AND CURRENT LEVEL CONTROLLER FOR A VARIABLE MUTUAL RELUCTANCE MACHINE

by

William Robert Gandler

Submitted to the Department of Electrical Engineering and Computer Science on July 8, 1982, in partial fulfillment of the requirements for the Degree of Master of Science.

ABSTRACT

The purpose of this thesis is to further William Wong's work in controlling the same experimental variable mutual reluctance machine. The complexity of the microprocessor based system and the accompanying software is increased beyond that used in the Wong implementation. Sense coil waveform detection of a new phase arrival generates an external interrupt to the microprocessor. Software programs using only position feedback and both position and velocity feedback that allowed independent speed and current level control were implemented. Current level control allows torque adjustment for different loads. Current level control is implemented by having a microprocessor informed DAC - op amp combination supply a reference voltage to one comparator input terminal while the voltage across a sensing resistor is supplied to the other comparator terminal. The current level at which the field transistor is turned off and the minimum and maximum levels that the phase currents are chopped between are controlled in this manner. Speed control is achieved by variation of a time delay between new phase detection and new phase switching. An excellent linear correlation was found to exist between the time delay and 1/speed. Speed control via continuous phase current adjustment was also implemented but worked only over a very narrow speed range. Finally, a program outputting phase duration counts onto LEDs allowed acceleration profiles to be obtained.

Thesis Supervisor: Richard D. Thornton

Title: Professor of Electrical Engineering

ACKNOWLEDGEMENTS

I would like to thank my advisor, Professor Thornton. I am also grateful for the constant help and support of Davida Pekarsky, Poh Ser Hsu, and David Otten. Above all, I am grateful to my parents for their continuous support.

TABLE OF CONTENTS

CHAPTER 1 INTRODUCTION AND LITERATURE SURVEY OF STEP MOTOR SPEED CONTROL

1. Introduction

A 15 pole, 4 phase, variable mutual-reluctance machine was designed by Professor Richard Thornton, and an experimental model was built by Pipat Eamsherangkoon.[1,2] This stepper like motor had a field winding and four phase windings -- A, B, C, and D. The phase windings A and C were wound in the same position but in opposite directions and likewise with phases B and D. A unipolar drive scheme was used because of its simplicity. Four sense windings were wound in the same manner as the phase windings so that the voltages across the sense windings were proportional to those across the phase windings.

William Wong, after making some modifications to the motor, used an INTEL 8748 microprocessor based motor drive that in a closed loop system determined position by voltage sensing across the sense windings, controlled choppers for current regulation in the motor phases, and performed switching of the phases. A three stage control program was used to bring the motor up to high speed with two phases on control, but the software program did not provide for variable speed control, and the chopper control was rather crude in that after the initial switch on time for a phase the duty cycle was kept constant.[3]

The purpose of this thesis is to further William Wong's work by implementation of current level control and speed control based upon the variation of a time delay between new phase detection and new phase switching. In addition, LEDs on which

acceleration profile information can be outputted are provided. The complexity of the microprocessor based system and the accompanying software is increased beyond that used in the Wong implementation. A photograph of the constructed circuit is shown in Figure 1.1.

2. Literature Survey of Step Motor Speed Control

2.1 Open Loop Control

If only limited performance is desired, stepping motors can be run open loop with phase switching pulses being given to the drive circuitry at carefully predetermined intervals. The step motor runs in synchronism with the pulse train provided that the motor can supply the needed torque and resonance problems are avoided. These resonance problems are related to the resonant frequency of the rotor. Periodic excitation of the rotor at its resonant frequency or some submultiple of it will reinforce the resonance and result in a loss of position synchronism, or a complete absence of motion, if insufficient damping is present. To prevent loss of synchronism during acceleration due to resonance points, it may be necessary to increase load inertia, increase load friction, or use a mechanical or viscous-inertia damper.[4,5]

The maximum rate at which a step motor may be started or stopped depends upon both the frictional and inertial loads. This information is usually presented in the form of a start-stop family of curves, where each curve represents a different value of load inertia, and torque is plotted against the stepping rate.

In many cases the stopping rate of the motor is slightly greater than the starting rate. This is to be expected since friction hinders acceleration but aids in deceleration. The final steady state speed that can be achieved is given by a curve of torque versus speed known as a slew curve. Since the inertial load of the rotor will only affect the amount of time required for the motor to reach the final steady state speed and not the value of the final speed, the final speed that can be achieved will be a function of the frictional load but will be independent of the inertial load.

In general, the shape of the acceleration or deceleration ramp given to a motor in open loop control is optimally determined by the slew curve for a given motor and driver combination. At a given speed, if the frictional load torque is subtracted from the torque on the slew curve, then the torque available for accelerating the load is determined, and the maximum acceleration possible is given by the available torque divided by the sum of the rotor and load inertias. Since available torque falls with speed, the rate of acceleration must decrease. Likewise, the pace of deceleration should start slow and then continue at a quicker pace. The ideal deceleration ramp will be a mirror image of the ideal acceleration ramp. If the torque were constant with speed, then the optimum acceleration curve would be a linear ramp. If torque were to decrease rapidly with speed, then an exponential(or a curve close to an exponential) ramp might be optimal. In practice, linear ramps should be used if torque drops off only very slowly with speed, but exponential ramps should be used if torque drops off rapidly

with speed.[4]

2.1.1 An Example of a Microprocessor Based Open Loop Control Scheme

An Intel 8080A microprocessor was used by Lafreniere to control pulses to a step motor motor driving a constant load in a computer output recorder. Time periods between motor pulses were controlled using time interval values stored in an acceleration/deceleration table. The table was optimized by running various profiles and saving a copy of the table that produced the optimum profile. Acceleration was accomplished using one linear segment and deceleration was accomplished using one to three piecewise linear segments as shown in Figure 1.2.

The necessary parameters used to generate the acceleration/deceleration tables were:

1. Start frequency(in steps/sec)
2. Maximum frequency
3. Acceleration slope(steps/sec^2)
4. Deceleration slope 1
5. Deceleration slope 2
6. Deceleration slope 3
7. Stopping frequency
8. Deceleration frequency 2
9. Deceleration frequency 3

The stopping frequency had to be carefully chosen to obtain good settling characteristics.

The lowest stepping rate required was 200 steps/second and

for stepping at 200 steps/second each pulse had to be 5 milliseconds apart. In order that one eight-bit byte would represent 5 milliseconds, each count had to be equal to 5 milliseconds divided by 255 or approximately 20 microseconds. Thus, the number of counts stored in a byte represented the number of 20 microsecond delays required. Acceleration values were calculated from the equation:

$$D = 1/(((A \times S) + Fo)(I)) \qquad (1)$$

where

D = delay count

S = step number(S = 0 to N)

A = acceleration slope in steps/sec^2

Fo = start frequency in steps/sec

I = delay increment(20 microseconds)

The required table length was determined by the equation:

$$S = (Fm - Fo)/A \qquad (2)$$

with

Fm = maximum frequency

S = steps required to reach maximum frequency

Fo = start frequency

A = acceleration slope in steps/sec^2

Deceleration tables were calculated in a similar way by going back from the stopping frequency toward the highest step rate.

In running the program instruction execution times are taken into account in setting the time delays. If the number of steps to be moved is sufficiently large, the acceleration table values will be used until maximum speed is achieved. If deceleration is

performed from maximum speed, then the microprocessor simply runs through the deceleration table. However, if maximum speed is not achieved by the point at which the number of steps remaining is equal to the number of steps in the deceleration table, then acceleration will continue only as long as the new acceleration frequency is less than the initial deceleration frequency that would be used for the remaining number of steps. When the initial deceleration frequency that would be used for the remaining steps is less than the new acceleration frequency, then the deceleration table is used for the rest of the steps.[6]

Miyamoto and Goeldel in similar work noted that the use of a general acceleration/deceleration table became less optimal as the number of steps decreased, and so used different tables for different small step increments. They calculated the best acceleration profile with a computer simulation that compared the motor response over a small time increment with the present phase remaining on versus the response obtained if the next phase was switched on. The simulation chose the sequence yielding the higher velocity and used it to establish initial conditions for the next time interval.[7]

2.2 Closed Loop Control

Step motors realize only limited performance in the open loop mode since there is no way to tell if the motor has missed a pulse or if the speed response is too oscillatory. If the input pulses arrive at too large a frequency, the motor may fail to follow. Great improvement of step motor performance can be

realized by using positional feedback and/or velocity feedback to determine the appropriate phase switching in relation to the rotor position. Closed loop control permits such improvements as more accurate position control, much higher speed control, and more constant speed control.

Closed loop control schemes traditionally use mechanical to electrical position transducers to provide feedback information of position and possibly velocity. Slotted disc tachometers with photoelectric sensors and permanent magnets mounted on the rotor with permanent magnet pick-up sensors on the stator are two commonly used schemes. Other schemes involve dc and ac tachometers, ac synchros, and potentiometers. In traditional closed loop control the motor is started initially with one pulse from the input command, and the following pulses are generated from the encoder assembly.

Recently closed loop motor systems have been implemented that use waveform detection; that is, feedback pulses are generated by a waveform detector. A waveform detector has obvious advantages over traditional encoders. The waveform detector can be completely an electronic device without any moving parts. Thus, it need not be mechanically linked to the motor. The motor could, therefore, be located out in a harsh environment with a detector, drive circuitry, and power supplies stowed away in a more favorable location.

The literature read showed only lead angle variation being used as a method of speed control in closed loop systems that employed waveform detection schemes. However, four other methods of closed loop speed control have certainly been used on other

occasions:

1) Voltage regulated control increases speed by increasing voltage. In addition to feedback which does the phase to phase switching, feedback control of velocity is used for regulating the voltage. Proportional velocity feedback is used to reduce the system's time constant and integral velocity feedback can be used to reduce the steady state error in velocity. Such a scheme is less efficient than other speed control schemes.

2) A chopped voltage control of speed can be performed by varying the switching ratio, the fraction of the total time the voltage is on. As with voltage regulated control, proportional and integral feedback may be employed. A nonlinear element should be put in the feedback loop so that the system does not try to drive the switching ratio higher than unity.

3) Bang-bang control whereby zero voltage is applied if the speed is too high and full voltage is applied if the speed is too low can also be used for speed control. However, such a scheme demands a continuous and accurate velocity feedback.

4) Finally, phase lock loop techniques may be used for speed regulation.

2.2.1 Waveform Detection

The waveform detection schemes implemented to date have all involved the detection of peaks -- either maxima or minima -- or zero crossings. To a fair extent the work has been experimental rather than analytical -- motor waveforms have been observed and only afterwards related back to rotor position and justified

analytically. Waveform detection schemes must be specifically tailored to the particular motor, driving circuitry, driving scheme, and operating conditions under consideration. In some work on variable reluctance step motors Kuo and Cassat found, for example, peaks that were present at low and high speeds but disappeared at medium speeds.[8] Kuo, Lin, and Goerke found a scheme in a permanent magnet step motor that worked by detecting phase peaks in one-phase-on operation, but this scheme could not be implemented in two-phase-on operation.[9] Often one waveform must be used for starting the motor and low speed operation and another waveform must be used for high speed operation. The detection schemes were complicated by the fact that switching of motor phases led to transients or noise that would lead to false detections if they were not ignored. This has been done by ignoring any detections for a predetermined interval around switching. In a closed loop control scheme used by Kuo, Lin, and Goerke a second noise rejecting circuit was also used to blank out false pulses caused by voltage peaks that were generated when the rotor was oscillating about the detent position when the motor was running at a low speed or coming to a stop.[9]

Peak detection of waveforms can be implemented either by a sample-and-hold circuit or by a differentiation circuit. A sample-and-hold circuit samples the waveform at a high rate, and when the present sample magnitude is less than the previous sample magnitude, the circuit interprets this as a peak detection. A differentiation circuit consists of an op-amp connected as a differentiator followed by a zero crossing detector. Kuo, Lin, and Goerke found the peak detector to be the

most critical element in the controller. Inappropriate pole placement in the differentiator design would cause improper operation of the controller.[9] Likewise, Unger noted that a better peak detector would have improved the operation of his system.[10]

Either voltage or current sensing can be used to provide detections. In sensing a current, the current must be converted into a voltage. This is performed by putting a small resistor of known value in the current path and measuring the voltage developed across it. Since the resistor will increase power dissipation and hence reduce the motor efficiency, its value should be kept as small as possible. While either current peak sensing or voltage peak sensing schemes may be used, voltage peak sensing schemes have the advantage that dead zones do not occur in the voltage waveforms if suppression diodes are used. Because of the positive forward bias voltage of the suppression diode, a dead zone may be present in the current waveform under low stepping rates. This dead zone in the current waveform could cause false detection errors.[9]

2.2.2 Speed Control via Lead Angle Variation

The waveform detection closed loop schemes that controlled speed did so by variation in the lead angle, the angle in advance of a particular equilibrium position at which the corresponding phase is turned on. Much of the literature uses the complementary concept of switching angle or feedback angle rather than lead angle. The switching angle is the angle the rotor

moves from an equilibrium position before receiving its first feedback pulse. If the switching angle is α degrees, after the initial starting pulse to the motor, the second pulse and all following pulses are sent after α degrees of motion from the equilibrium position. If the step angle is R degrees, then a switching angle of α degrees corresponds to a lead angle of $2R-\alpha$ degrees.

If currents were instantaneously established and decayed instantaneously, the same lead angle would produce maximum torque at all speeds, but obviously currents take time to buildup and decay. While the time required for current to buildup represents only a small distance at low speeds, this time represents a larger distance at higher speeds. Thus, the lead angle must be increased to increase speed. As speed increases, the maximum torque that can be produced must decrease due to increases in impedance and back emf.[11] Thus, the maximum torque decreases at increased speeds, and the lead angle that produces the maximum torque increases. At low speeds small lead angles will produce more torque than large lead angles, but at high speeds large lead angles will produce more torque than small lead angles. Usually, a step motor operates at lead angles between 1 and 2.5 steps, with larger lead angles resulting in higher speed but lower torque. In a four phase step motor, the maximum speed is achieved at a lead angle of about 2.5 steps.[12] Usually, as the lead angle nears three steps, the speed will fall and eventually the motor will stall. It must be remembered that the assumption of constant speed operation is only valid if the average torque, which is a function of lead angle, is counterbalanced by the drag

torque, including the coulomb frictional and viscous frictional torque, which is exerted on the rotor. The lead angle is adjusted in order to achieve this.

Average torque as a function of switching angle for constant speed operation has been calculated analytically. Tal calculated the average torque = $(1/2\pi)\int_0^{2\pi} T_g(\theta)d\theta$ (3) where θ = electrical angle and $T_g(\theta)$ = generated torque for constant speed operation as a function of switching angle for two-phase bifilar wound permanent magnet stepping motors with two-phase-on constant voltage excitation and for three-phase permanent magnet stepping motors with one-phase-on constant voltage excitation. By taking a derivative, he then found the switching angle that maximized the average torque at a given speed and the resulting value of the maximum average torque.[13]

Kuo used the above method and the principle of time continuity of flux linkages to calculate torque-speed curves for various switching angles that could be obtained in a two-phase bifilar wound permanent magnet step motor under various drive and control schemes. In addition to calculating the maximum average torque, he calculated the minimum average torque that could be used for stopping the motor. Specifically, chopping, bilevel, and two-phase-on control schemes were considered. A chopping drive was approximated by assuming that the chopper kept currents at a constant level. As expected, the magnitude of the maximum average torque for two-phase-on excitation was $2^{0.5}$ times greater than for one-phase-on excitation, but oddly enough his calculations also showed that a bilevel drive resulted in only a

slight improvement in the torque produced in the one-phase-on scheme at low and medium speeds and actually decreased torque produced at high speeds.[14] Kuo, Lin, and Yen calculated torque-speed characteristics in three phase variable reluctance stepping motors using one-phase-on drive with constant voltage for different switching angles.[15]

The waveform figure used for detection must be sufficiently in advance of the equilibrium position of the switched on phase to provide a large enough lead angle to maintain the maximum desired speed. Smaller lead angles and hence lower speeds are achieved by adding in an electronic time delay between the time that waveform feature detection occurs and the time that phase switching occurs. Thus, by increasing the electronic time delay, the steady state speed is lowered.

Lead angles that incorporate time delays have distinct advantages over fixed reference lead angles without any time delays. The electronic time delay results in better acceleration characteristics. Since the time delay corresponds to a smaller distance at low speeds, the effective lead angle is larger until the final speed is achieved, and at speeds above some low level that is quickly reached, the larger lead angles produce a higher torque that accelerates the motor more rapidly. Likewise, electronic time delay also provides the motor with better deceleration characteristics. In addition, electronic time delay makes the motor speed less dependent upon load variations. For example, if the load is increased, speed tends to decrease, so the rotor moves a shorter distance during the time delay, so the lead angle increases, tending to increase the speed, thereby

partially offsetting the effect of the load increase.[12,16]

In an optical detection system the angle of detection, the lead angle with no time delay, is always constant. This need not be the case in a waveform detection scheme. In one waveform detection scheme the angle of detection increased from 2.1 steps to 2.4 steps as the motor speed increased. The motor will operate properly as long as the curve of the detection angle versus steady state speed has a positive or zero slope. With such a curve, when the motor is loaded, the speed will be reduced and hence the lead angle is reduced. This is a welcome outcome since the smaller lead angle will produce a greater torque to apply to the load. A curve of detection angle versus steady state speed with a negative slope would not work, since applying a heavier load would slow the motor down and thus increase the lead angle so that less torque would be available, and eventually the motor would stall.[17]

In control schemes with encoder feedback control, such as an optical scheme, the injection of extra pulses may be used to achieve a lead angle greater than two steps. In control schemes with waveform detection this would be analagous to the use of a small angle of detection using one waveform feature at low speeds and a large angle of detection using another waveform feature at high speeds. In both cases a larger lead angle causes a higher speed.[8]

2.2.3 Speed Control with Traditional Logic Circuitry

Using traditional logic circuitry, the lead angle can be

varied via the use of a "fixed-unit time delay speed controller." The speed controller is composed of three main parts: a speed comparator, a time delay selector, and a time delay. The speed comparator compares the time taken to perform a motor step with the desired step period and decides whether or not the average speed over the last finished step is too fast or too slow. If the average step speed is too slow, the time delay selector adjusts the time delay for the next feedback pulse to be one fixed time unit less than the preceding time delay. If the average step speed is too fast, the time delay selector adjusts the time delay for the next feedback pulse to be one time unit greater than the previous time delay. Because the time delay can only be adjusted by one time unit after each feedback pulse, a tradeoff exists between the accuracy or fine tuning of the steady state speed and the time taken to achieve the steady state speed. A "variable-unit time delay controller" that uses a time delay increment which is proportional to the error in speed can be used to avoid this tradeoff but only at the expense of more complicated electronic circuitry if traditional logic circuitry is used.[12]

Yackel has given the complete circuit diagram for one particular implementation of a fixed-unit time-delay speed controller. A network of gated one-shots can produce a delay of up to 70 fixed units with each unit being equal to 25 microseconds.[18]

2.2.4 Microprocessor Control of Acceleration, Deceleration, and Constant Speed

Complex step motor control schemes require specially designed logic circuitry that is usually expensive and time consuming to construct. By using microprocessors complicated control algorithms may be implemented without complex logic circuitry at a lower cost.

B.H. Wells used an Intel 8080 microprocessor with a two microsecond cycle time to implement acceleration and deceleration algorithms on a three phase step motor with a feedback system using two photoelectric sensors and a slotted disc in an encoder arrangement. Speed was determined by the time between encoder pulses. For either acceleration, deceleration, or constant speed operation the microprocessor would perform four basic tasks for each motor step:

1. Delay a specified amount of time past an encoder interrupt of the microprocessor. The delay was calculated during the last step.
2. After the delay is over, send a pulse to the motor driver card.
3. Calculate the delay for the next step.
4. Wait until the next encoder interrupt which will restart the sequence.

To provide maximum acceleration, an algorithm is used to provide the motor with maximum torque at all times. With instantaneous buildup and decay of current, a switching angle of 0.75 times the step angle(or a lead angle of 1.25 steps) would produce maximum forward torque in a three phase step motor. However, the current has finite buildup and decay times, and the

finite buildup time must be included in the acceleration algorithm. The acceleration algorithm is:

1. Measure the time for the present step.
2. Predict time for the next step(TN).
3. Delay = 0.75(TN)-buildup time

To predict the time for the next step, the following equation is used:

$$T_{k+1} = T_k + \alpha(T_k - T_{k-1}) \tag{4}$$

Changing the value of the constant α will change the degree of curvature. However, in order to implement an acceleration algorithm, a modification of equation (4) is necessary. If the time for the next step is predicted, then actually it is the time for the present step that is currently underway that is being predicted. To solve this problem an algorithm that predicts two steps in advance is used:

$$T_{k+2} = T_k + \alpha(\alpha + 2)(T_k - T_{k-1}) \tag{5}$$

Experimental data showed $\alpha(\alpha + 2) = 0.5$.

For low speed operation utilizing large switching angles, the execution order of the four basic steps could be rearranged as follows:

1. Calculate the delay for the next step once an encoder interrupt is received.
2. Delay the specified amount of time just calculated.
3. After the delay is over, send a pulse to the motor driver card.
4. Wait until the next encoder interrupt which will restart the sequence.

While this rearrangement of the four basic steps does not

allow a small enough switching angle for maximum speed operation, it has the advantage of requiring that step times be predicted only one step in advance rather than the two step in advance prediction that must be used with the first ordering given for the four basic steps. This, of course, improves the accuracy of the prediction.

For the first three steps, the two step in advance algorithm cannot be used. Two alternative measures can be used. First, encoders can be used to generate a suitable switching angle. Secondly, if the load is not subject to much variation, delays can be preselected. For the third step delay, the time value for the first step could be included in a calculation.

Because deceleration took a much smaller portion of the total time than acceleration, Wells used a simple deceleration scheme. By skipping one pulse and setting the delay to a constant, the lead angle was changed from about two steps(depending on the speed) to less than one step. When the motor had been decelerated, the delay was set to zero in order to provide a one step lead angle for slow constant speed.

Wells proposed but did not implement an algorithm for constant speed control. One 8 bit byte is used to specify the speed in steps/second divided by 10. A speed from 10 to 2550 steps/second is specified. A nominal delay is calculated for the specified speed. The unloaded speed versus switching angle curve is represented by a linear approximation. For the motor used the linear approximation is:

$$\alpha = 9.4 - (0.00571)(1/T) \qquad (6)$$

with α = the switching angle and T = sec/step. Also:

$$\text{delay} = (T\alpha)/15 \tag{7}$$

with 15 the number of degrees/step.

By algebra this yields:

$$\text{delay} = 0.627T - 3.81 \times 10^{-3} \tag{8}$$

If one computer unit = 0.8×10^{-6} second, then:

$$\text{delay}_{\mu} = 0.627T_{\mu} - 476 \tag{9}$$

or

$$\text{delay}_{\mu} = (7.84 \times 10^{4}/(\text{speed}/10)) - 476 \tag{10}$$

A processor can perform this calculation before the motor is set in motion. After the nominal delay is calculated, the motor is accelerated until the acceleration algorithm calculates a delay less than the nominal value, and from this point on a special error routine is used instead of the acceleration routine. The error routine provides an adjustment in the delay proportional to the difference between the measured and desired speeds so as to minimize the error. The following control law is suggested:

$$\text{delay}_{k+2} - \text{delay}_{k+1} = \alpha(T_n - T_k) \tag{11}$$

where $0<\alpha<0.617$ in order to insure stability and T_n is the desired time for each step.[19]

2.2.5 Phase Superposition

Phase superposition refers to the overlap of phases or the extent to which phases are turned on simultaneously. Wetter, Jufer, and Imhof defined the rate of phase superposition as

$$k_s = (t_s/T)(100) \tag{12}$$

with t_s = time of superposition, the time phases are turned on

simultaneously, and T = the interval between two phases switching on or T = 1/f where f = steps/second.

If P phases are present, k_s = -100% if all phases are off and k_s = (P-1)(100%) if all phases are on. In a four phase motor k_s would be:

-100% for all phases off

0% for one phase on

100% for two phases on

200% for three phases on

300% for four phases on

By varying the phase overlap any particular percentage between -100% and 300% could be achieved.

The rate of phase superposition yielding maximum torque was found to vary from one frequency to another. For a definite fixed load, the superposition rate that minimized speed oscillations or maximized dynamic stability was not necessarily the same as that resulting in maximum torque.[20]

2.2.6 Specific Schemes Implemented or Simulated

Singh and Kuo did a computer simulation of a single stack four phase variable reluctance stepping motor using dual voltage drive that drove the printhead in a high-speed impact printer system. After an excursion of an even number of steps ranging from 2 to 100, the printhead had to come to rest in a fully damped manner with a tolerance of about seven percent of a motor step within 30 milliseconds. The simulation was performed by integrating six nonlinear equations for the motor using the

fourth-order variable-step size runge-kutta method. The simulation was accomplished via seven steps:

1. The best switching angle for acceleration was determined.
2. The number of steps after which deceleration started was determined.
3. The best switching angle for deceleration was determined.
4. Either (a) the number of steps after which the switching angle was changed for low velocity operation was determined or a probably better method (b) the proper motor speed at which to change the switching angle for low velocity operation was determined.
5. The low velocity switching angle was determined.
6. The pulse to the last step was inhibited with delayed last step damping.
7. At low velocity indicating near peak overshoot the last pulse was given.

The motor came to a uniform low speed mode before the command to stop so that a single damping scheme was possible for all step increments.[21]

Frus and Kuo ran a three phase single stack variable reluctance step motor in the one-phase-on scheme using detection from a waveform in the on-phase mode.[17]

J.D. Unger used parameters in the current waveforms of a three phase variable reluctance step motor in the one-phase-on scheme tó determine the damping delays needed for optimum damping using an electronic backphasing scheme. A "position" peak that occurred in the current waveform of one phase exactly when the rotor crossed the detent position of another phase provided the

initial information needed for damping. It told the system to begin the damping process. The time difference between the "position" peak and a peak in another phase current following shortly thereafter was found to vary with load, so the time difference between these two peaks provided information that told how much time should pass after the "position" peak until a backphasing pulse was applied. The duration of the backphasing pulse was kept constant to avoid circuit complexity, but modifying it would have yielded some damping improvement.[10]

Kuo and Cassat, working with a three phase variable reluctance step motor, developed a control scheme for one-phase-on operation by detecting current peaks in the phases. First-off mode detection was defined as detection of a waveform feature in a phase that was on during the last cycle and second-off mode detection was defined as detection of a waveform feature that was on two cycles ago. It was found that peak detection in the on-phase current was difficult to predict and not advisable, for even if peaks were found at low and high speeds, they might vanish completely at medium speeds. The first-off mode was usually reliable at all speeds and recommended for closed loop control. The second-off mode could only be used after the motor was brought up to speed, but had the advantage of sometimes allowing higher speeds. Switching the waveform detection from the first-off mode to the second-off mode is analagous to the injection of an extra pulse in a closed loop scheme with an encoder and optical detection.[8]

Pittet and Jufer used as feedback detection of a zero in a

current difference to achieve closed loop control of a one phase stepping motor. They have referred to such closed loop control as self-synchronization.[22]

Mckee used current feedback from the on-phase of a three phase variable reluctance step motor to achieve load adaptive damping of single steps using electronic backphasing. The height of the current waveform was found to be inversely proportional to the square root of the inertia. As the height decreased, both the optimal time after the peak to begin the backphasing pulse and the optimal duration of the backphasing pulse increased exponentially. The correct timer intervals for optimal damping are achieved by application of a scaled version of the amplitude of the peak to the timer's RC network. This is possible because the pulse width of the timer increases exponentially as the voltage decreases; that is, the relation is fortuitously similar to those between the amplitude height of the local peak and the two timer parameters.[23]

Lin, Kuo, and Goerke found three waveform detection schemes that coud be employed in a bifilar-wound four phase permanent magnet step motor. The voltage waveforms across the phase windings exhibited detectable peaks, a positive peak in the first-off mode and a negative peak in the third-off mode, at practically all speeds. A closed loop scheme was developed that was based on detecting the first positive peak in the first-off mode. Current waveforms had detectable peaks at low speeds, but, unfortunately, at high speeds peaks did not not always appear in one specific phase. However, detectable peaks could always be found in the difference between currents in opposite phases, such

as $i_a - i_c$ or $i_b - i_d$, if the two phases were in the first-off and third-off modes. This control scheme based on current differences was successfully implemented in the one-phase-on scheme, but because of the existence of some unknown parameter could not be implemented in the two-phase-on scheme. Finally, detent positions of a permanent magnet step motor were detected by sensing the zero crossings of the back emf generated by the permanent magnet flux linkage while the rotor was turning. The voltage across phase a was written:

$$V_a = R_a i_a + d\lambda_a/dt \tag{13}$$

with λ_a being the flux linkage of phase a which was expressed as

$$\lambda_a = L_a i_a - L_c i_c + K_1 \cos\theta \tag{14}$$

with L_a and L_c the average inductances of phases a and c, K_1 the maximum flux linkage due to the permanent magnet, and θ the rotor position in electrical radians. Taking the derivative of equation (14) :

$$d\lambda_a/dt = L[(di_a/dt) - (di_c/dt)] - K_1 \omega \sin\theta \tag{15}$$

where $L = L_a = L_c$ and $\omega = d\theta/dt$.

Then using simple algebra on equations (13) and (15) :

$$K_1 \omega \sin\theta = L[(di_a/dt) - (di_c/dt)] - V_a + R_a i_a \tag{16}$$

$K_1 \omega \sin\theta$ was defined as the back emf generated by the permanent magnet and was obtained by using the appropriate op-amp network on measurements of $L[d(i_a - i_c)/dt]$, V_a, and $R_a i_a$.

By similar manipulations a back emf waveform of the type $-K_1 \omega \cos\theta$ can be obtained with phase B and phase D. In theory a pair of measurements of $K_1 \omega \sin\theta$ and $-K_1 \omega \cos\theta$ could have been used to uniquely determine the rotor position, but in practice the

amount of switching noise did not allow this to be done, so zero crossings were detected to show rotor detent positions. For operation over a wide speed range, zero crossings of the first-off and second-off modes were used. However, when the motor was started, because the speed was very low, there were no detectable zero crossings in the off-mode waveforms, so the peak of the on-phase back emf was used for starting the motor. This peak in $\omega\sin\theta$ has the advantage of showing load-adaptation characteristics. The back emf detection scheme has the advantage over other waveform detection schemes of being invariant under different drive schemes. Both one-phase-on and two-phase-on drives yield the same back emf waveform. This scheme has the disadvantages of requiring a complex controller and placing an upper limit on the steady state speed due to noise occupying an increasing portion of the back emf waveform at higher speeds due to an increased frequency of switching between the phases in this constant voltage drive scheme.[9,24]

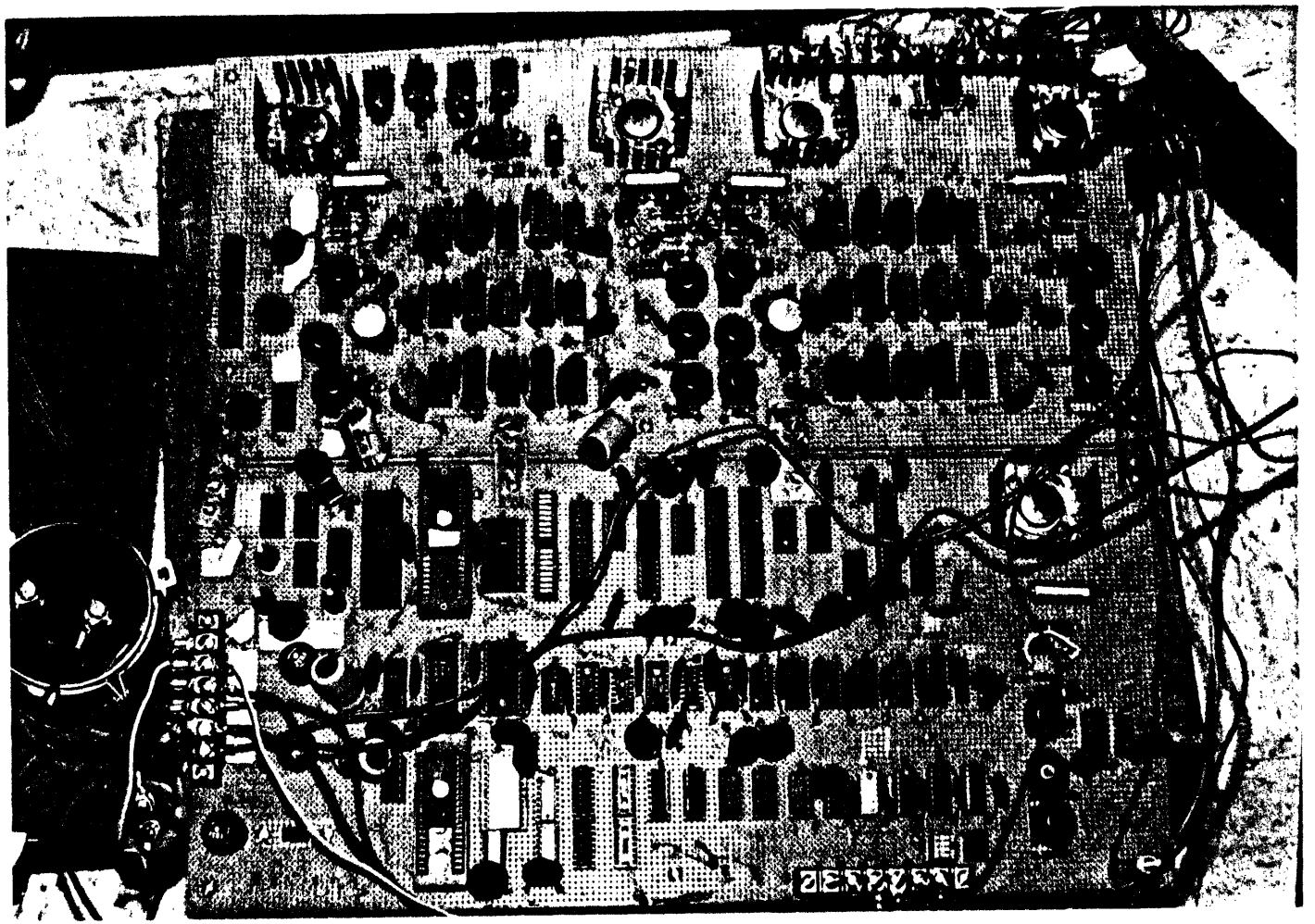


Figure 1.1 Photograph of Constructed Circuit

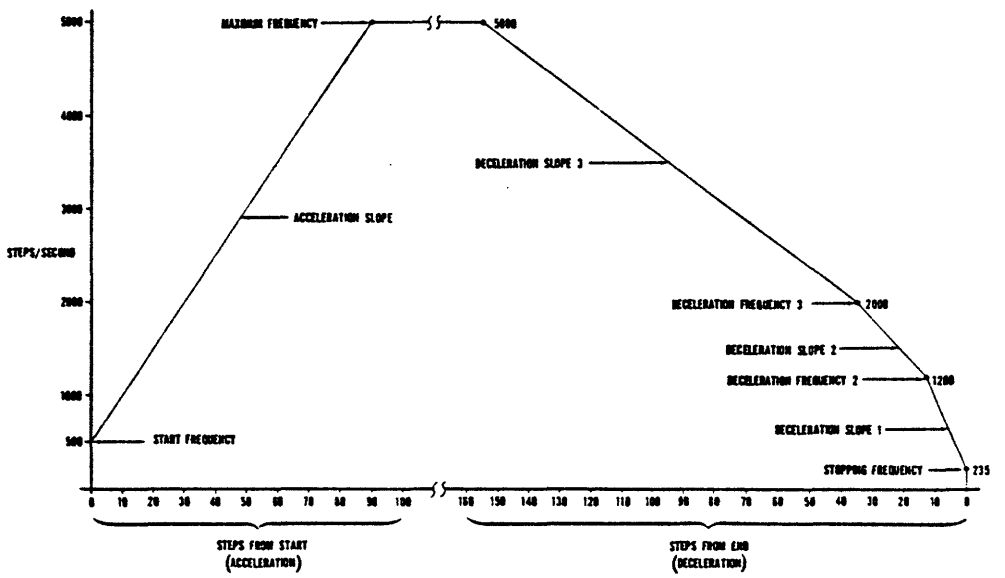


Figure 1.2 Acceleration/Deceleration Profile Used by Lafreniere

CHAPTER 2 OVERALL CIRCUIT DESIGN ISSUES

1. The Three Phase States Encountered During Chopping

Suppose phase D is being chopped on and off. The resulting phase B and D current waveforms are shown in Figure 2.1. The three states of voltage and current situations existing in the field and phase coils are shown in Figures 2.2a - 2.2c.

When phase D is chopped off, an opposite current of nearly equal magnitude appears in the reverse coupled phase B. When phase D is off, the phase B current and the field current are being returned to the voltage supply through a diode in antiparallel with the field coil.

When phase D is first switched on, by reverse coupling it almost immediately assumes a value equal in magnitude to that of the phase B current at the end of τ_3, the interval of reverse current flow. During τ_1 the phase D current is less than the field current. As the phase D current increases, less of the field current flows through the diode antiparallel to the field coil and more flows through the phase D coil. τ_1 ends and τ_2 begins when the value of the phase D current becomes equal to the value of the field current. During τ_2 the phase D current is equal to the field current and the diode antiparallel to the field coil is off. Since approximately the entire supply voltage is across the phase D coil during τ_1, while during τ_2 the supply voltage is spread across both the field and phase coils, the phase D current rises more quickly during τ_1 than during τ_2. With a duty cycle of 0.5, i_B will decay all the way to zero producing an interval of zero torque, so the duty cycle should be

kept greater than 0.5.

2. Position Detection by Sense Coil Voltages

$$V_D = L_{DD}d(i_D)/dt + L_{DF}(\theta)d(i_F)/dt + i_F\omega dL_{DF}(\theta)/d\theta$$

represents the voltage across phase coil D in τ_1 and τ_2 when phase current flows only through the D coil. Likewise,

$$V_F = L_{FF}d(i_F)/dt + L_{FD}(\theta)d(i_D)/dt + i_D\omega dL_{FD}(\theta)/d\theta$$

represents the voltage in the field coil in τ_1 and τ_2. When phase current flows only through the D coil during τ_1 the additional constraint $V_F = -V_{DON}$ is present. At startup the terms containing ω are very small and may be ignored. So at startup during τ_1

$$L_{FF}d(i_F)/dt + L_{FD}(\theta)d(i_D)/dt = -V_{DON}$$

Noting that $L_{FD}(\theta)$, a sinusoidal function of position, can be either positive or negative, L_{FF} is always positive, and V_{DON} is close to zero, it is seen that during τ_1 $d(i_F)/dt$ at startup can be either positive or negative.

$$V_B = L_{BB}d(i_B)/dt + L_{BF}(\theta)d(i_F)/dt + i_F\omega dL_{BF}(\theta)/d\theta$$

represents the voltage across the phase B coil during τ_3 when phase current flows only through the B coil. Likewise,

$$V_F = L_{FF}d(i_F)/dt + L_{FB}(\theta)d(i_B)/dt + i_B\omega dL_{FB}(\theta)/d\theta$$

represents the field coil voltage during τ_3 when phase current flows only through the B coil. $V_F = -V_{DON}$ during τ_3, so at startup during τ_3

$$L_{FF}d(i_F)/dt + L_{FB}(\theta)d(i_B)/dt = -V_{DON}$$

Noting that $L_{FB}(\theta)$, a sinusoidal function of position, can be either positive or negative, L_{FF} is always positive, and V_{DON} is

close to zero it is seen that during τ_3 $d(i_F)/dt$ at startup can be either positive or negative.

Because the diode antiparallel to the field coil is off during τ_2, $d(i_F)/dt$ must always be positive during τ_2. In conclusion, at startup $d(i_F)/dt$ is always positive during τ_2 but can assume either polarity during τ_1 or τ_3. This is verified experimentally by repetitively switching a single phase while the motor shaft is slowly turned manually.

Rather than sensing directly from the phase coils, sensing is actually performed on separate sense windings, which are wound in the same manner as the phase windings so that the voltages across the sense windings are proportional to the voltages across the phase windings. This is done so that the voltage swing on the sense windings will be within the -5 to +5 volt range of the comparator. If sensing had been performed directly from the phase windings, positive and negative comparator input voltage limits larger than the 20V used to supply the field and phase coils would have been needed, and positive and negative supply voltages greater than 20V would have been needed to power the comparators.

When chopping is being performed on a given phase, waveform detection is performed on the following phases's sense coil voltage. Hence, when A-B-C-D-A activation is employed, when phase D is being chopped sensing will be done on sense coil A to detect the arrival of phase A. Crossings of the sense voltages from negative to positive polarity cause detections, so phase D is turned off and phase A is turned on when the polarity of the A sense voltage becomes positive.

The voltage across the phase A coil is:

$V_A = L_{AA}d(i_A)/dt + L_{AB}d(i_B)/dt + L_{AC}d(i_C)/dt + L_{AD}d(i_D)/dt + L_{AF}(\theta)d(i_F)/dt + i_F\omega dL_{AF}(\theta)/d\theta$

L_{AB} and L_{AD} are small and since sensing is only done in phase A when phase A and phase C have been turned off long enough for the A and C phase currents to have decayed to zero, then $di_A/dt = 0$ and $di_C/dt = 0$ during a phase A sensing. Then when sensing phase A:

$$V_A = L_{AF}(\theta)d(i_F)/dt + i_F\omega dL_{AF}(\theta)/d\theta$$

At startup when the field transistor is turned on and the entire supply voltage falls across the field coil giving a large $d(i_F)/dt$, the polarities of the $L_{AF}(\theta)d(i_F)/dt$, $L_{BF}(\theta)d(i_F)/dt$, and $L_{DF}(\theta)d(i_F)/dt$ voltages are sensed to indicate the initial position. After the initial sensing at turnon, before the motor picks up speed and the $i_F\omega dL_{AF}(\theta)/d\theta$ term predominates over the $L_{AF}(\theta)d(i_F)/dt$ term, sensing must only be performed on phase A just before phase D is chopped off, that is, during τ_2 when the field current is equal to the phase D current. As previously discussed, $d(i_F)/dt$ is always positive during τ_2 but can be of either polarity during τ_1 or τ_3. Thus, the $L_{AF}(\theta)d(i_F)/dt$ term only conveys positional information during the τ_2 interval.

Torque for phase A, $T_A = i_Ai_FdL_{AF}(\theta)/d\theta$, becomes positive 180 degrees before the equilibrium position for phase A. Thus, the $i_F\omega dL_{AF}(\theta)/d\theta$ term becomes positive at the same time the torque becomes positive, 180 degrees before the equilibrium position. The $L_{AF}(\theta)d(i_F)/dt$ term becomes positive only at the start of the second half of the positive torque interval for

phase A, 90 degrees before the equilibrium for phase A. At turnon the initial sensing of position based purely on $L_{AF}(\theta)d(i_F)/dt$, $L_{BF}(\theta)d(i_F)/dt$, and $L_{DF}(\theta)d(i_F)/dt$ terms activates the phase that is located between 180 and 90 degrees from the equilibrium position.

Startup failures may occur when ω is very low and $L_{AF}(\theta)d(i_F)/dt$ predominates over $i_F\omega dL_{AF}(\theta)/d\theta$. Then, phase A is turned on 90 degrees before its equilibrium position, and if ω remains very low, switching to phase B will not occur until phase A is at its equilibrium position. Suppose that phase A arrives at its equilibrium position and phase B has just missed being detected. Then, the motor will stay indefinitely at the phase A equilibrium position and phase B will never be detected. Increasing the initial acceleration by increasing the field current will solve this problem.

When phase D is being chopped, phase A is being used for waveform detection, and ω is appreciable, then the $i_F\omega dL_{AF}(\theta)/d\theta$ term predominates over the $L_{AF}(\theta)d(i_F)/dt$ term. In a one phase on scheme, phase A is activated 180 degrees before its equilibrium position and phase A is turned off and phase B is turned on when phase A is 90 degrees before its equilibrium position. In a two phase on scheme, phase A would be turned on 180 degrees before its equilibrium position and turned off at its equilibrium position, that is, phase A would be on during the entire interval of positive torque except during windows occurring in the second half of its positive torque interval when it would be turned off so that i_A would decay to zero to allow sensing in phase C. The use of such windows would have the

disadvantage of reducing available torque. In this thesis only a one phase on scheme is employed.

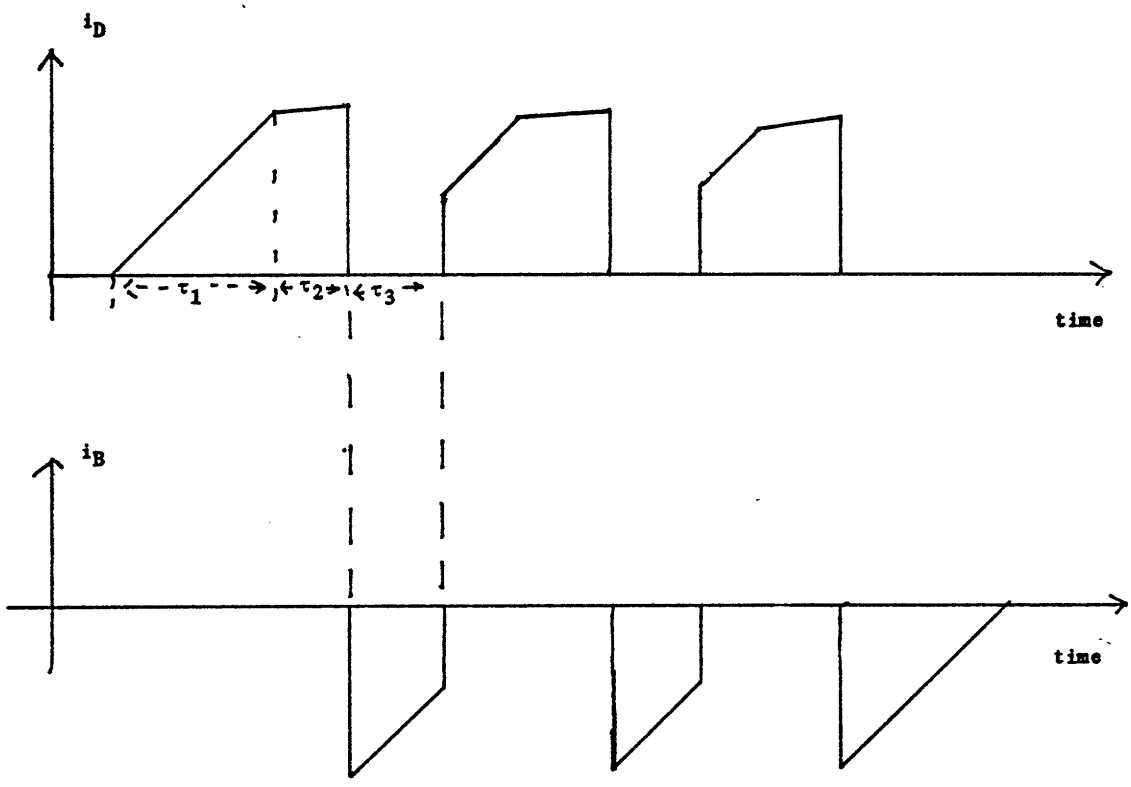


Figure 2.1
Phase D is chopped on and off 3 times.

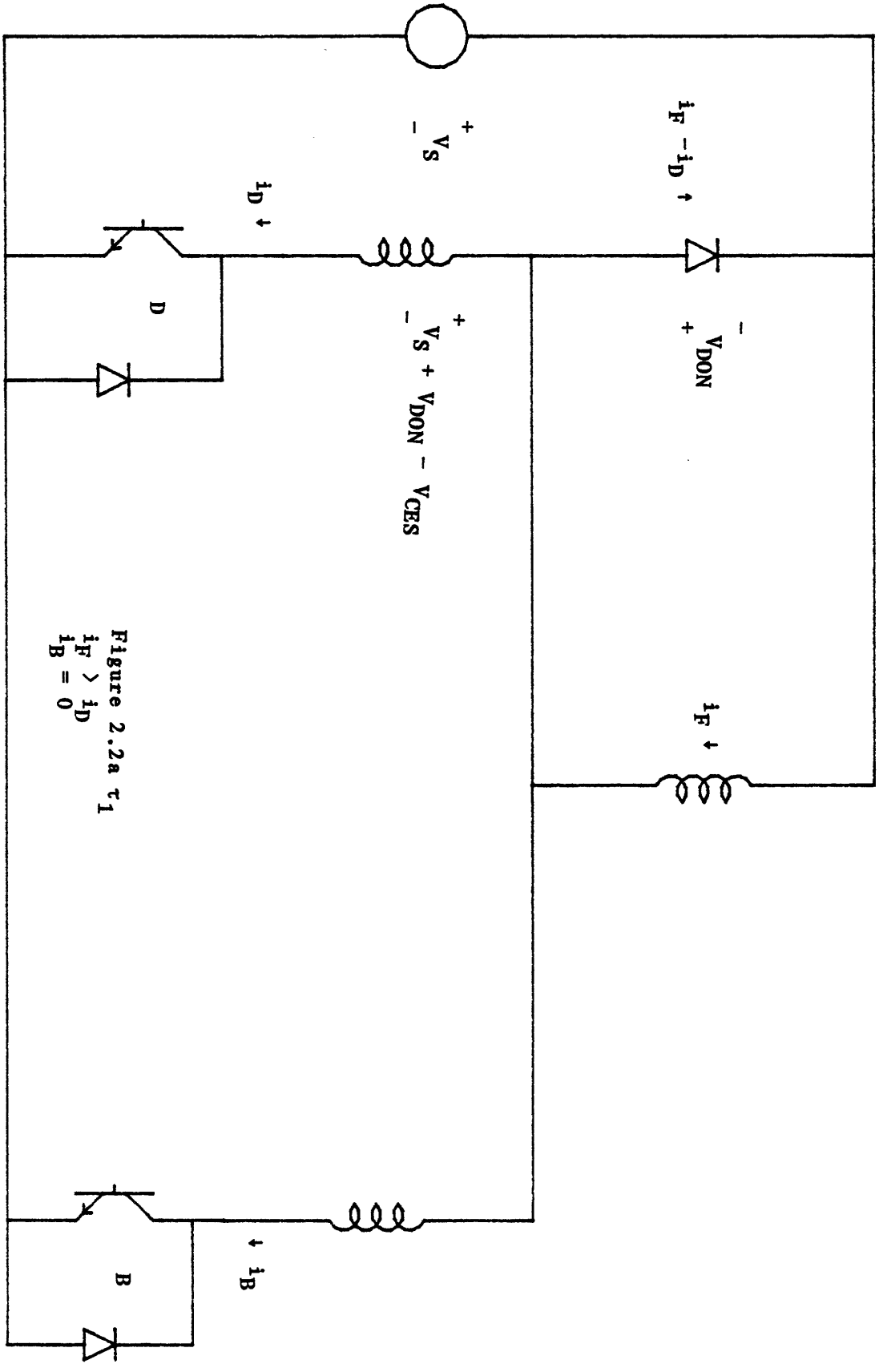


Figure 2.2a τ_1
$i_F > i_D$
$i_B = 0$

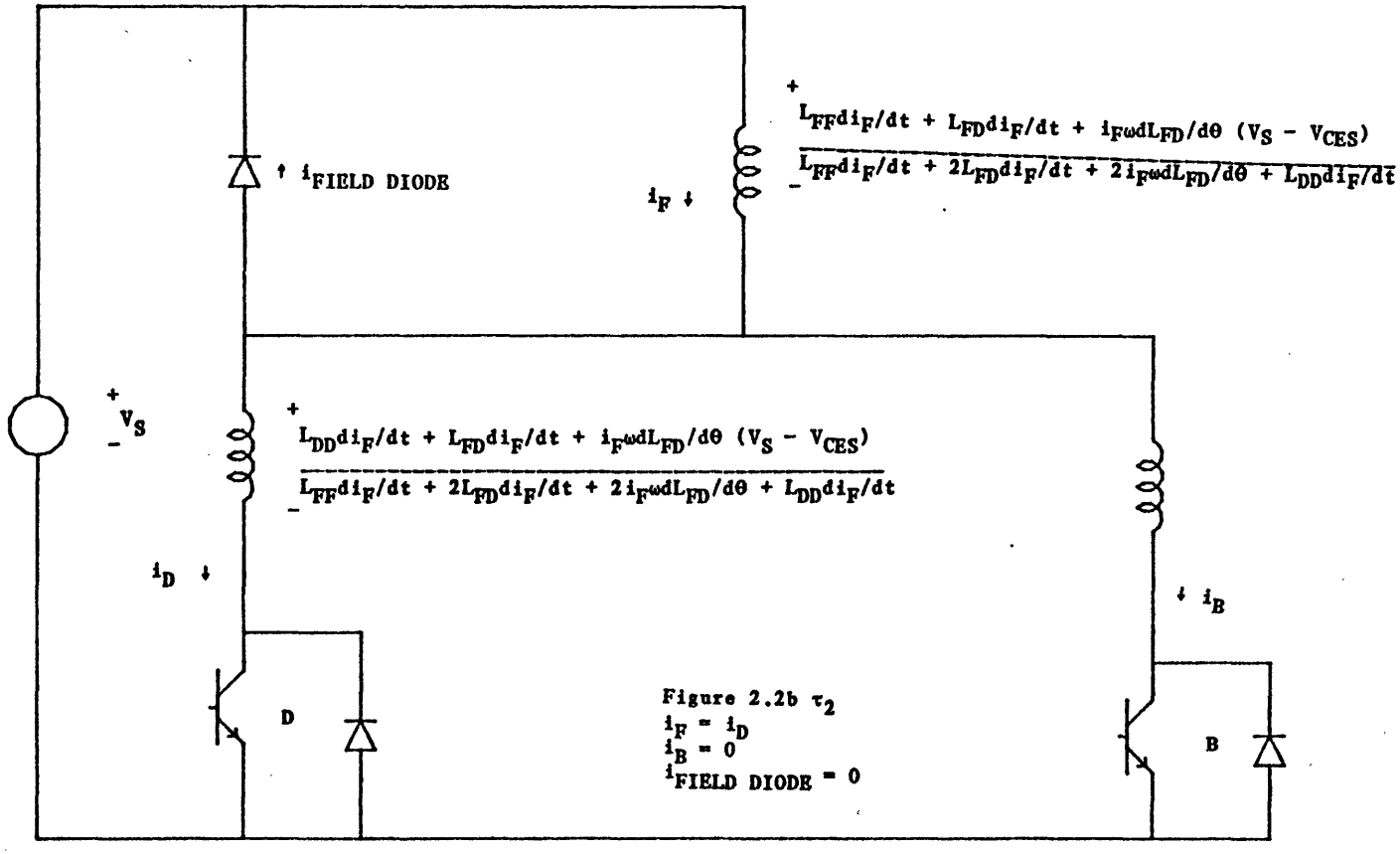
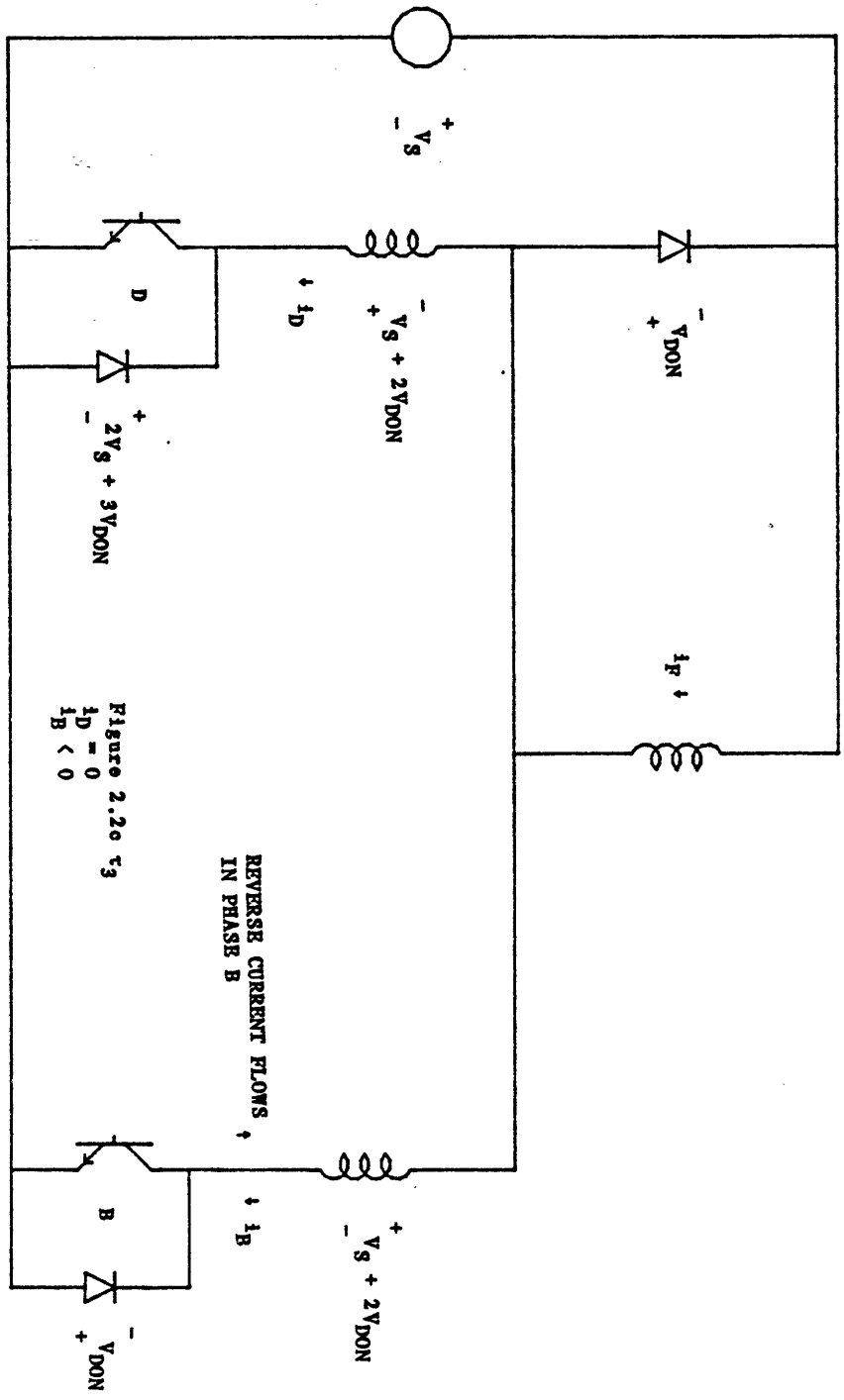


Figure 2.2b τ_2
$i_F = i_D$
$i_B = 0$
$i_{FIELD\ DIODE} = 0$


+
V_S
-
D
i_D
-
$V_S + 2V_{DON}$
+
+
$2V_S + 3V_{DON}$
-
-
V_{DON}
+
i_F
Figure 2.2o τ_3
$i_D = 0$
$i_B < 0$
REVERSE CURRENT FLOWS
IN PHASE B
B
i_B
+
$V_S + 2V_{DON}$
-
-
V_{DON}
+

CHAPTER 3 HARDWARE DESIGN ISSUES

1. Overall Block Diagram

An overall block diagram of the hardware scheme employed is shown in Figure 3.1.

2. The Case For Using Fast Recovery Diodes

Current and voltage waveforms showing how switching compares for real and ideal diodes are sketched in Figure 3.2. Fast turn-on diodes will have more ideal turn-on characteristics than slow diodes and fast turn-off diodes will have more ideal turn-off characteristics than slow diodes. Technological considerations usually result in a fast turn-off diode being fast turn-on. Generally, the turn-on surge voltages and power losses are of secondary importance compared to turn-off surge voltages and power losses.

When phase D was chopped on and off in my predecessor's circuit, a circuit without snubber networks but with diodes connected antiparallel to the switching transistors to allow reverse phase current flow, then a careful examination of phases B and D revealed positive current spikes occurring in both phases during the instant of phase D turn-on. A highly schematic nonscaled diagram of these current spikes is shown in Figure 3.3. These current spikes were caused by a reverse current flowing through the diode in phase B immediately after that diode was reverse biased. Since coils B and D are reverse coupled, a current spike in phase D had to occur in the same direction so that an immediate change in flux did not take place. To minimize

these spikes in my scheme a MR852 fast recovery diode was used instead of the 1N4720 general purpose rectifier used by my predecessor.

The recovery charge Q_R, equal to the integral of reverse current through the diode, is often 200 to 500 times greater in a standard diode than in a fast diode. Thus, the use of fast diodes allows near total or total suppression of these current surges to be achieved. The energy produced each turn-off is equal to $Q_R E_C$, where E_C is the reverse voltage across the diode immediately after switching. By taking the worst case, where all of the energy is dissipated in the diode, an upper bound on the diode power dissipation due to switching is obtained. Thus, $P=Q_R E_C f$, where f is the switching frequency, gives the worst case diode switching power dissipation. Hence, the use of fast diodes drastically reduces the diode switching power dissipation. High losses during switching would prohibit the use of standard diodes at high frequencies.

Eliminating the diode reverse surge current also eliminates extra turn-on losses of the power transistor. Often, the turn-on losses of a transistor are concentrated in its structure(hot spots), and result in a fast fatigue of the transistor.

RC networks are often put in parallel with standard diodes to protect them against high surge voltages at turn-off. Generally, to eliminate surge voltages replacing standard diodes with fast diodes is better than using standard diodes with RC protection networks; the RC networks consume a large amount of energy at high frequencies.

One final advantage of the use of fast diodes is the reduction of radioelectric interference. During diode switching, an abrupt variation of current and hence of magnetic field takes place. The amplitude of the interference is proportional to the recovered charge Q_R.

Fast diodes have some drawbacks; increased leakage currents, greater forward voltage drops, and lower maximum reverse voltage ratings are often the penalties of making a diode fast.[1]

3. Transistor Switching Network

The transistor switching network used for the 4 phase coils is shown in Figure 3.4. The transistor switching network used for the field coil is shown in Figure 3.5.[2] The field network is the same as that used for a phase coil except that a 7405 is used instead of a 7401, the flywheel diode, a MR821, is placed antiparallel to the field coil, and the sensing resistance measures out to .079 ohms rather than .080 ohms.

In my predecessor's scheme a .075 ohm wire wound resistor was used as a sensing resistance, but a wire wound resistor acts as an inductance. Noninductive sensing resistances were constructed by placing 16 - 1/2 watt carbon resistors averaging 1.28 ohms in parallel. Placing the resistors between 2 cut pieces of a PC board, each with 16 holes, allowed a neat compact construction. Since the standard deviation of different samples around a mean is inversely proportional to the square root of the sample size, paralleling resistors had the advantage of producing sensing resistances whose standard deviation around a mean value was 1/4 the standard deviation of the individual 1.28 ohm

resistors around a mean value. The value of the sensing resistances was kept as small as possible to minimize power losses. Finally, note that tantalum capacitors of a few microfarads were used to keep the collectors of T_1 and T_3 and the emitters of T_2 from deviating from their steady state voltages.

3.1. Discussion of the Switching Network

A and C are reverse coupled phases and B and D are reverse coupled phases. Suppose phase D is on and no other phase is carrying current. When the phase D power transistor is turned off, the phase D coil voltage drops and changes polarity so as to maintain the phase D coil current. If perfect reverse coupling were present, the phase D coil voltage would drop to $-(V_S + 2V_{DON})$, V_{CE} of transistor D would rise from $V_{CE(SAT)}$ to $2V_S + 3V_{DON}$, and the voltage across the reverse coupled phase B coil would rise to $V_S + 2V_{DON}$. When the phase B coil voltage reached $V_S + 2V_{DON}$, then the current in the phase D coil would stop and a current of equal magnitude would flow through phase B in the reverse direction so as to maintain continuity of flux. However, due to imperfect coupling or leakage inductances, this transfer of energy to the reverse coupled phase is not perfect. Because of the leakage inductance, at turn-off of phase D with no snubber network present, V_{CE} of transistor D exhibits a brief duration large positive voltage spike before settling to a steady state value of $2V_S + 3V_{DON}$ and the phase D coil voltage exhibits a brief duration large negative voltage spike before settling to a value of $-(V_S + 2V_{DON})$.

A turn-off snubber is used to attenuate the brief duration positive voltage spike in collector to emitter voltage due to the leakage inductance. If a snubber is not used, the power transistor is quickly destroyed by the occurrence of collector to emitter voltages much greater than the V_{CEO} rating of the transistor. The turn-off snubber network used is shown in Figure 3.6.

As the value of the capacitance increases, the turn-off switching time increases since dV_{CE}/dt decreases, but the turn-off switching losses in the transistor decrease. For small values of capacitance, the total turn-off losses of the snubber and transistor are smaller than those of the unaided transistor.

Let the phase current be I_p and assume that during the transistor fall time t_f the current in the inductive load remains constant. During the fall time t_f the current decreases linearly in the transistor and increases correspondingly in the capacitor. Then, if no leakage inductance is present, at the completion of transistor turnoff, the voltage V_0 across the transistor is given by:

$$CV_0 = \int_0^{t_f} I_{capacitor} dt$$

Therefore, $V_0 = I_p t_f/2C$, the V_{CE} across the transistor at the end of turnoff in the absence of leakage inductance. Thus, in the absence of leakage inductance, the turn-off switching power dissipation of the transistor is decreased by making $I_{pt_f}/2C < 2V_S + 3V_{DON}$, the value to which V_{CE} rises.

Now consider the effect of the leakage inductance L_E. If the transistor has been completely turned off and all of the current is flowing into the snubber, $L_E I_p^2/2$, the energy in the

leakage inductance, must be transferred to the snubber. This energy will increase the value of V_{CE} above $2V_S + 3V_{DON}$ by a ΔV such that

$$L_E I_P^2/2 = C[(2V_S + 3V_{DON} + \Delta V)^2 - (2V_S + 3V_{DON})^2]/2$$

$$= C[(\Delta V)^2 + 2(\Delta V)(2V_S + 3V_{DON})]/2$$

Thus, to obtain a particular ΔV for a given leakage inductance L_E set

$$C = L_E I_P^2/[(\Delta V)^2 + 2\Delta V(2V_S + 3V_{DON})]$$

Also, the capacitor and resistor values must be chosen so that the capacitor is completely discharged during the on interval of the transistor, that is, $RC << t_{ON}$.

In addition, R must be large enough so that the turn-on discharge current of the capacitor

$$\Delta i = [2V_S + 3V_{DON} + \Delta V - V_{CE(SAT)}]/R$$

that will flow through the transistor at turn-on does not become too excessive.

Finally, the power rating of the resistor must be larger than $[C(2V_S + 3V_{DON} + \Delta V)^2 f]/2$, where f is the switching frequency.

The question arises as to whether to use a turn-on switching aid network to reduce the power transistor turn-on switching losses and reduce the turn-on surge currents. Generally, turn-on switching transients are less of a danger to transistors than turn-off switching transients so only a turn-off snubber is used.[3,4]

3.2. Calculations for the Transistor Switching Network

$V_{EC(SAT)}$ of T_2 = 0.5V

+ V_{BE} of T_1 = 0.6V

+ V_{BE} of T_P = 0.8V

+ Voltage across sensing resistor = 0.25V

+ Voltage across R_1

= 5.0V

Thus, the voltage across R_1 = 2.85V.

Let the base current into T_P be limited to 200 mamp.

$$R_1 = 2.85V/0.2 \text{ amps} = 15 \text{ ohms}$$

The transistor specifications indicate that this level of base current should drive the transistor well into saturation for collector currents up to 5 amps.

Let the current through R_3 = 2 ma.

$$R_3 = 0.6V/\ 2 \text{ ma} = 330 \text{ ohms}$$

Let 4 ma come from the base of T_2.

$$R_4 = (4.4V - 0.2V)/(4 \text{ ma} + 2 \text{ ma}) = 680 \text{ ohms}$$

Voltage across R_2 = 9.5V. Let 20 ma flow through R2.

$$R_2 = 9.5V/20 \text{ ma} = 470 \text{ ohms}$$

No current limiting resistor is put in series with the 1N4933 diode since the peak reverse base current pulled from the 2N6339 is experimentally determined to be 0.6 amps.

t_f for the 2N6339 is about .13 μsec at 2.0 amps. Ignoring the leakage inductance, at the end of t_f the transistor collector to emitter voltage $V_0 = Ipt_f/2C$. Let $V_0 = V_{CEO}/2$ = 60V. Then $C = Ipt_f/2V_0 = (2 \text{ amps})(1.3 \times 10^{-7}\text{sec})/2(60V) = 2.2 \times 10^{-9}$ farad

Now consider the effects of leakage inductance.

$$L_{EA} = L_{EC} = 20\ \mu h.^5$$

Assume $2V_S + 3V_{DON}$ is kept to 80V. To be safe keep $2V_S + 3V_{DON} + \Delta V$ to 100V. Then $\Delta V = 20V$.

$$C = [L_E I_p^2]/[(\Delta V)^2 + 2\Delta V(2V_S + 3V_{DON})]$$

$$= [(20 \times 10^{-6}\ H)(2\ amps)^2]/[(20)^2 + 2(20)(80)]$$

$$= 2.2 \times 10^{-8}\ farad$$

Clearly the leakage inductance determines the size of the capacitor used in the turn-off snubber.

At turnoff,

$$\Delta V_{CE}/\Delta t = I_p/C = 2\ amps\ /2.2 \times 10^{-8}\ farad = 10^8\ volts/sec$$

If $\Delta V_{CE} = 100V$, then the turn-off switching time Δt is only 1 μsec.

The transistor can handle 25 amps continuous current. So let the initial current discharge of the turn-off snubber

$$\Delta i = (2V_S + 3V_{DON} + \Delta V)/R < 100V/R = 10\ amps$$

$$R = 100V/\ 10\ amps = 10\ ohms$$

$RC = (10\ Ohms)(2.2 \times 10^{-8}\ farad) = .22$ μsec, which will be far less than the on interval of the transistor.

4. Pushbutton Inputs to the 8039 Microprocessor

The three pushbutton inputs to the 8039 are shown in Figure 3.7. Traditional debounce latches with nand gates follow each of the pushbuttons. Since a microprocessor reset sets all the output ports to one thereby turning on all the transistors, a long $\overline{\text{RESET}}$ input to the microprocessor would burn out a fuse or one or more transistors. Thus, a 74121 monostable is used to keep the $\overline{\text{RESET}}$ input short in duration. The RESET pin must be held at ground(.5V) for at least 10 milliseconds if a reset is performed just as the power supply comes within tolerance.

However, only 5 machine cycles are required if the power is already on and the oscillator has stabilized. Since a reset is always performed many seconds after the circuit has been powered up, the pulse width need only be at least 5 machine cycles = 5(1.36μsec) = 6.8μsec. The monostable RC network yields a pulse width = 0.7RC = $0.7(1.5 \times 10^4\Omega)(10^{-9}\text{farad}) = 10.5\mu\text{sec}$.

5. Intel Microcomputer Components

The Intel microcomputer parts used are shown in Figures 3.8a-c. One 8039 microprocessor, one 8212 address latch, two 2716-1 2K x 8 EPROMs, one 8185-2 1K x 8 bit static RAM, and three 8243 I/O expanders are used.

New programs are installed by erasing the EPROMs under UV light for 25 minutes and then programming with the universal prom programmer of an Intel microcomputer development system. With programs of 2K or less of memory space only the #1 2716-1 EPROM is required. Because the EPROMs are continuously removed from their sockets for reprogramming, zero insertion pressure (ZIP) sockets are used for the EPROMs. Use of regular sockets would rapidly lead to pin destruction through mechanical wear during DIP insertion and extraction.

Note that the 2716-1 is used instead of the 2716. The 2716-1 is the same 2K x 8 EPROM as the 2716. It has just been selected out because of a faster speed. According to the 1979 Intel Component Data Catalog the 2716 has a maximum access time of or an address to output delay time of 450 nsec. For the 2716-1 the maximum access time is 350 nsec. The 8039

microcomputer has a maximum address setup to data in time of 400 nsec. The 8212 address latch has a maximum write enable to output delay time of 40 nsec. Thus, the EPROM must have a maximum acess time no greater than 400 - 40 = 360nsec. Therefore, the 2716-1 is suitable while the 2716 is not. Using one 4K x 8 2732 EPROM would be simpler than using two 2K x 8 EPROMs, but unfortunately EPSEL lab does not have the equipment needed to program a 4K x 8 EPROM.

The 8039 has an 8-bit CPU, an 128 x 8 RAM data memory, and an 8 bit timer/event counter. The 8039 operates with an 11 MHz crystal whose output is divided to form 1.36 μsec machine cycles consisting of 5 machine states. Each instruction is executed in one or two machine cycles.

Two 8 pin SPST switches provide input data to the 16 ports on the #2 8243 I/O expander. Such information as the phase to be tested, the maximum and minimum phase current levels, the initial switching delay time, and the frequency with which the delay time is updated can be inputted on these switches. Input data should be read in with the high order bit corresponding to P73, P63, P53, or P43 and data should be outputted with the high order bit corresponding to P73, P63, P53, or P43 because if this order is reversed then the ordering of bits in the accumulator becomes opposite to the ordering of bits on the I/O expanders and arithmetic cannot be performed on accumulator numbers moving between the accumulator and an I/O expander.

The #3 8243 I/O expander is used for two purposes. Before or while the motor is running, it outputs current level information to the DAC latches. After the motor has stopped

running, it can output phase duration counts to 7407 buffers driving LEDs thereby allowing acceleration profiles to be obtained.

The 8185-2 RAM is used solely to store the phase duration counts used in acceleration profiles.

6. Current Level Regulation

Four types of current level information are provided. When the motor is started, the field transistor is turned on. As soon as the field current reaches a desired field current level determined by the field DAC input to the field comparator, the field transistor is turned off and the desired phase transistor is turned on. After a phase transistor has been turned on, the phase current rises until it reaches a maximum current level determined by the maximum DAC input to that particular phase's maximum comparator. After the maximum current level has been reached, the phase transistor is chopped off and a reverse current flows in the complementary phase. When the complementary reverse phase current has fallen in magnitude to a minimum reverse current level determined by the minimum DAC input to that particular complementary phase's minimum comparator, then the original phase transistor is chopped on again. The bottom DAC and comparator are present solely to check for proper functioning of reverse phase coupling. The bottom comparators prevent a phase turn on signal unless the reverse current detected is above a certain magnitude. If very little or no reverse phase coupling occurs, and the reverse phase current is always less than the

bottom current value, then the phase current will not be chopped on again, a new phase will not be detected, and the motor will stop.

To illustrate this better, suppose the field current level is set at 2.9 amps, the maximum current level is set at 3 amps, the minimum current level is set at 1.5 amps, and the bottom current level is set at 0.5 amps. Then, the field transistor is turned off forever(unless a restart occurs) when the field current reaches 2.9 amps. Phase currents are chopped off when they reach 3 amps and are chopped back on again as soon as a reverse complementary phase current between 1.5 and 0.5 amps is detected. If reverse phase coupling is functioning properly, a detection of a reverse phase current between 1.5 and 0.5 amps will occur as soon as the magnitude of the reverse phase current has decreased to 1.5 amps. Thus, chopping will keep the phase currents between 1.5 amps and 3 amps.

Figure 3.9 shows the eight 74LS75 latches used to output current level information from the #3 8243 I/O expander to the four DACs. Since each latch has 4 bits and each DAC has 8 bits, two latches are used for each DAC. An 8243 can sink 5 ma at .45V on each of its 16 I/O lines simultaneously. On each I/O line is a 7407(driving a LED) requiring a -1.6 ma low level input current. Each enable or G input on a 74LS75 requires a -1.6 ma low level input current. Then, the 4 enable pins required to address the 2 latches driving 1 DAC require -6.4 ma low level input current. Hence, the 4 enables for the 2 latches driving 1 DAC must be driven by 2 separate I/O lines.

Figure 3.10 shows the DAC - op amp networks used to generate

reference voltage levels for the current level comparators. The field and maximum DAC - op amp networks generate outputs between 0 and +0.5 volts. The minimum and bottom DAC - op amp networks generate outputs botween 0 and -0.5 volts. The op amps input to pots with wipers positioned to input 4/5 of the output voltage as a reference level to a comparator input terminal. With pots compensations can be made for differences in phase comparators and sensing resistances. A pot was used for the field comparator only so the scale factor for the field current level would be the same as that for the other current levels. Reference voltages with magnitudes up to 400 mv can be placed at the comparator input terminals with unit increments to the DAC corresponding to gradations of 1.56 mv at the comparator terminals. Gradations of 1.56 mv across a .080Ω sensing resistance correspond to current gradations of 1.56 mv/.080Ω = 20 ma. When the power transistor is turned on, 0.2 amp of base current flows through the power transistor so for any given maximum current level I_{MAX} not exceeding an upper bound of 4.8 amps set the comparator reference voltage at 16 mv + I_{MAX}(80 mv/amp).

Of course, the current-carrying capacity of the wire used for winding the motor must also be considered. Varnish coated number 19 copper wire was used for winding the motor.[6] According to one reference #18 wire has a current-carrying capacity of 11 amps and #20 wire has a current-carrying capacity of 7 amps,[7] so by interpolation the motor wiring should have a current-carrying capacity of 9 amps.

Careful attention must be given to grounding issues to

ensure the proper functioning of the comparators. All the sensing resistances employed in this circuit - the 4 resistances sensing the phase currents, the resistance sensing the current through the field transistor, and the resistance sensing whether or not the diode antiparallel to the field coil is on - have all been located physically close together so that the distance and hence the inductance of the wires joining the ground sides of these sensing resistances is minimized so the voltage variation between the grounds of the sensing resistances is as small as possible. This will prevent problems caused by the fact that comparators have a poor common mode rejection ratio. The ground terminal of the sensing resistances is tied directly to the ground terminal of the circuit board and does not connect with the logic chip grounds until the ground terminal. This separation of logic and power grounds is necessary to prevent the occurrence of noise in the logic grounds from the switching of the power transistor currents into the inductance presented by the logic grounds. Note that the pots determining reference levels for the comparators are connected to the ground side of the corresponding sensing resistance rather than to the logic ground whereas the ground pins for the comparators are connected to the logic grounds(Figures 3.11,3.12a-b, and 3.14). This is necessary because use of a logic ground for the pot would have the effect of putting a long wire length or an inductance between the input terminals of the comparator - the distance being the wire distance from the ground side of the sensing resistance to the ground terminal along the power ground wire plus the length along the logic ground wire from the ground terminal to the logic

ground point connected to the reference pot.

To prevent comparator oscillation compensation networks were applied to some of the comparators used in this circuit(Figures 3.11, 3.12a-b, and 3.14). Oscillations are particularly a problem with the maximum level comparators because of the slow rate of rise of the phase current toward its final value during the τ_2 state. Positive feedback or hysteresis applied to the balance pin 5 removes these oscillations. The .002μf capacitor between the 2 balance pins serves as a high frequency filter. It provides a low impedance shunt to any high frequency noise. The resistor network for comparator compensation was determined experimentally. Networks were constructed with the 2.2KΩ and 33KΩ resistors and the value of the third resistor was increased until a square wave input would not cause oscillations in any one of four 311 comparators tested.

The network for sensing the field current level is shown in Figure 3.11.

The network used for current regulation of phases A and C is shown in Figures 3.12a-e. An identical network is used for current regulation of phases B and D. The two monostables shown in Figure 3.12e that provide the clocking for the JK flip flops of phases A and C will also provide the clocking for the JK flip flops of phases B and D.

The monostables shown in Figure 3.12c that produce 5.7 and 1.89 μsec positive pulses cause a delay of 5.7 μsec to occur before the J input of the flip flop shown in Figure 3.12d can be presented with the information that the phase current has reached

its upper limit. The current must exceed its maximum limit for 5.7 μsec to send a signal to the J input. This serves two purposes. First, it ensures that a transient maximum current detection occurring right at the instant of transistor turnon does not immediately turn the transistor off again. Second, it allows sensing to be done in the 5.7 μsec just before the phase is turned off. Since not all phase chops include a τ_2 state, this does not ensure that all the detections will occur during the τ_2 state, but it does ensure that the current will reach the specified maximum level before the next phase activation occurs.

The monostables shown in Figure 3.12c that produce 12.6 μsec negative pulses ensure that a phase is not turned on immediately after being turned off - that is, it makes sure that a detection does not occur during the current transfer between the two reverse coupled phases.

7. External Interrupt Generation

An external interrupt to the 8039 microcomputer is generated whenever a next phase detection is made. The circuitry used to accomplish this task is shown in Figures 3.13 to 3.16.

Figure 3.13 shows the comparators used to detect the sense coil voltage polarities. As discussed in chapter two, at startup the outputs from these comparators provide the initial position information to the #1 8243 I/O expander. Note that the ground of the sense coils is tied directly to the ground terminal of the circuit board. The LM111 specifications indicate that neither input terminal should be allowed to become more negative than the negative supply voltage. With the 1N4148 diodes connecting -5V

to the noninverting terminals, negative voltage spikes occurring in the sense waveforms cannot become more negative than one diode drop below -5V.

Figure 3.14 shows a comparator network used to detect the presence of state τ_2. During τ_2 the diode antiparallel to the field coil is off and a positive voltage exists across the field coil. At least 3V will appear across the 5.6MΩ resistor during state τ_2. Then a base current of 3V/5.6MΩ = .54 μa will flow. With this PNP transistor an i_B = .54 μa yields an $i_C > 110$ μa. Hence, a collector current of 110 μa is set equal to a reference voltage of 0.1V. Thus, a sensing resistor of 0.1V/110 μa = 1KΩ is used. No more than 50V ever appears across the field coil. Then, i_B = 50V/5.6MΩ = 8.9 μa. For this PNP transistor an i_B = 8.9 μa yields an $i_C < 3.7$ ma. Hence, (3.7 ma)(1KΩ) = 3.7V is the largest voltage that should appear at the positive input terminal of the 311 comparator. The 1.2KΩ resistor is put in merely as an added protection for the comparator. The positive input voltage limit is 30V above the negative supply or 25V. Thus, if the PNP transistor shorts, the maximum voltage appearing at the positive input terminal will be equal to (1KΩ/(1KΩ + 1.2KΩ))(50V) = 23V, and the comparator will be protected.

Figure 3.15 shows the network that determines if a τ_2 state is an external interrupt requirement. When OVERRIDE τ_2 RESTRICTION = 0, an interrupt can only occur during state τ_2. When OVERRIDE τ_2 RESTRICTION = 1, then an interrupt can occur during τ_1 or τ_2.

When sensing phase A, $V_A = L_{AF}(\theta)d(i_F)/dt + i_F\omega dL_{AF}(\theta)/d\theta$.

At startup the $L_{AF}(\theta)d(i_F)/dt$ term predominates. Thus, at startup all detection of the next phase must be done during τ_2 when the polarity of $d(i_F)/dt$ is known to be positive rather than during τ_1 or τ_3 when the polarity of $d(i_F)/dt$ can be either positive or negative. As the motor speeds up, the $i_F\omega dL_{AF}(\theta)/d\theta$ term predominates and detection of the next phase need not be restricted to τ_2. In fact, with high speed operation the number of chops is maintained at a smaller and more uniform number if detection occurs during both τ_1 and τ_2. When next phase detection is restricted to τ_2, the average number of chops per phase becomes larger and less uniform. A string of phases containing mostly one or two chops will also show an occasional phase containing three, four, or five chops. In summary, at startup OVERRIDE τ_2 RESTRICTION should be 0 but should become 1 as speed increases.

The question arises as how to determine the proper point for switching OVERRIDE τ_2 RESTRICTION from 0 to 1. The switching from 0 to 1 should occur when the $i_F\omega dL_{AF}(\theta)/d\theta$ term predominates over the $L_{AF}(\theta)d(i_F)/dt$ term. Since $d(i_F)/dt$ is proportional to the supply voltage and for a phase containing a fixed number of chops ω is proportional to the supply voltage, then for a phase containing a fixed number of chops ω is proportional to $d(i_F)/dt$. Hence, since i_F, $L_{AF}(\theta)$, and $dL_{AF}(\theta)/d\theta$ are independent of supply voltage, the switching of OVERRIDE τ_2 RESTRICTION from 0 to 1 should occur when the number of chops per phase falls below a certain level.

In the network shown whenever the number of chops per phase is less than a value determined by the input data switches for 2

consecutive phases, then OVERRIDE τ_2 RESTRICTION is equal to 1. Otherwise, OVERRIDE τ_2 RESTRICTION is equal to 0. If 1111 or 1110 is the initial value put on the input data switch, then OVERRIDE τ_2 RESTRICTION is equal to 0. If 1101 is the initial value, then whenever only one chop per phase occurs for two consecutive phases, OVERRIDE τ_2 RESTRICTION is equal to 1. If 0000 is the initial value on the input data switch, then whenever the number of chops per phase is less than fifteen for 2 consecutive phases, OVERRIDE τ_2 RESTRICTION is equal to 1.

A chain of three 74LS04 inverters is used to ensure that the load input of the 74LS161 is low for a sufficient time before the clock goes high.

Figure 3.16 shows the network generating the external interrupt to the 8039 microcomputer. The interrupt is sampled every machine cycle during ALE, so the pulse of the monostable serving as a nand gate input must be at least one machine cycle in length or 1.36 μsec long. To prevent a false reading caused by a glitch or transient, a sense coil positive voltage during a phase maximum current signal must be confirmed 2.31 μsec later to indicate a next phase detection. This is particularly important in preventing a false detection during a short CURMAX glitch that could occur at transistor turnon if OVERRIDE τ_2 RESTRICTION = 1.

8. Miscellaneous Details

According to T.I. gold plating on wire wrap posts is not necessary. A T.I. technical report concludes that unplated wrap is stable after exposure to harsh environments.[8] Therefore, the

use of unplated wire wrap sockets is not expected to cause any problem.

Adequate use is made of capacitors. Before using the circuit three huge electrolytic capacitors are attached to the board's barrier strip for voltage supplies. They are placed between the power supply voltage and ground, +5V and ground, and -5V and ground. 0.1μf ceramic capacitors are used between +5V and ground and where applicable between -5V and ground for every DIP on the board. In addition, six electrolytic 1000μf capacitors are scattered over the board. Also, tantalum capacitors of a few microfarads are used to keep the collectors of T_1 and T_3 and the emitters of T_2 from deviating from their steady state voltages. Finally, .05μf ceramic and 6.8μf tantalum capacitors are used to keep comparator reference voltges constant.

Three 3 amp fuses are present in the circuit. They are located between the power supply voltage and ground, +5V and ground, and -5V and ground.

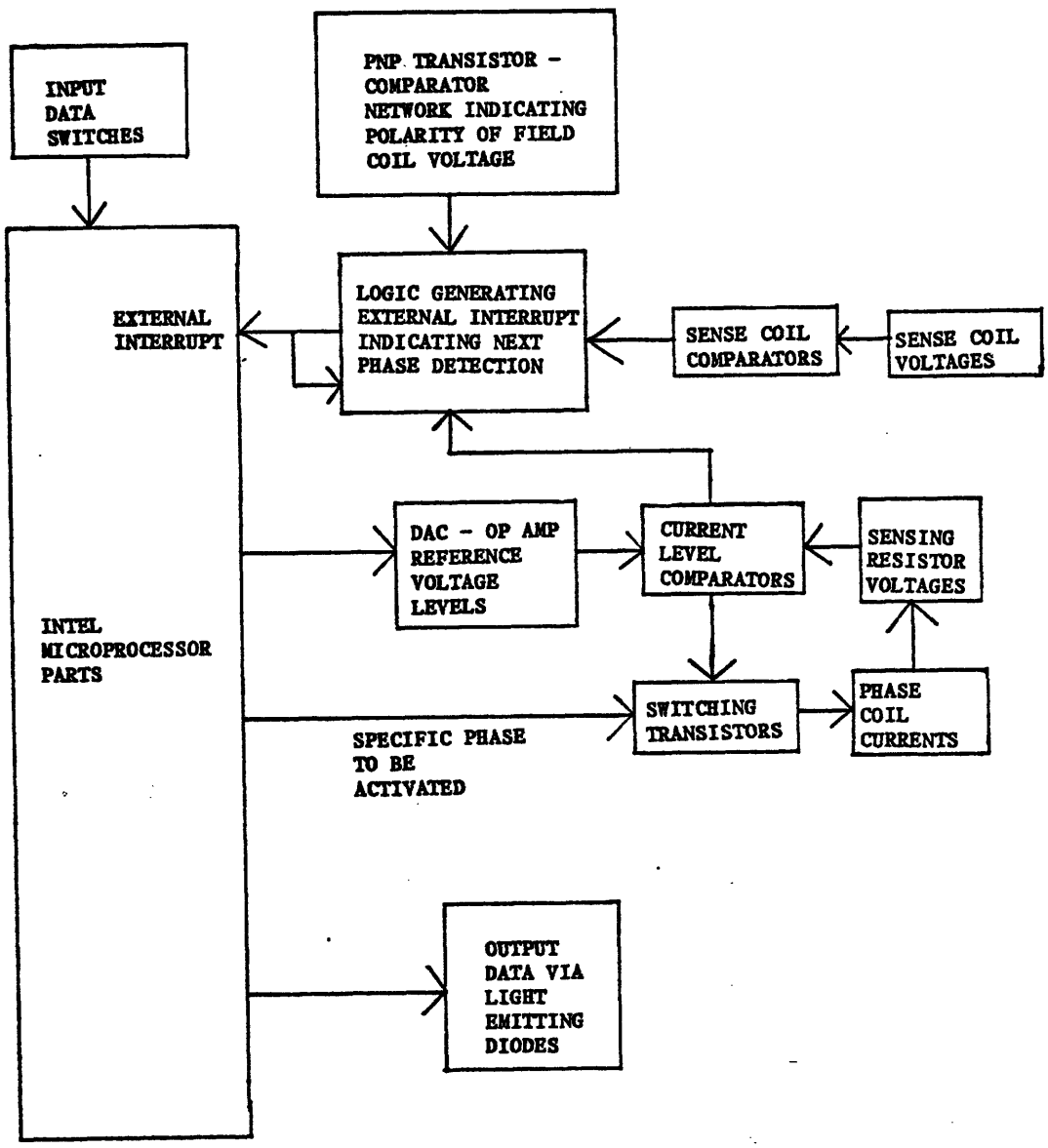


Figure 3.1 An overall block diagram of the hardware scheme used

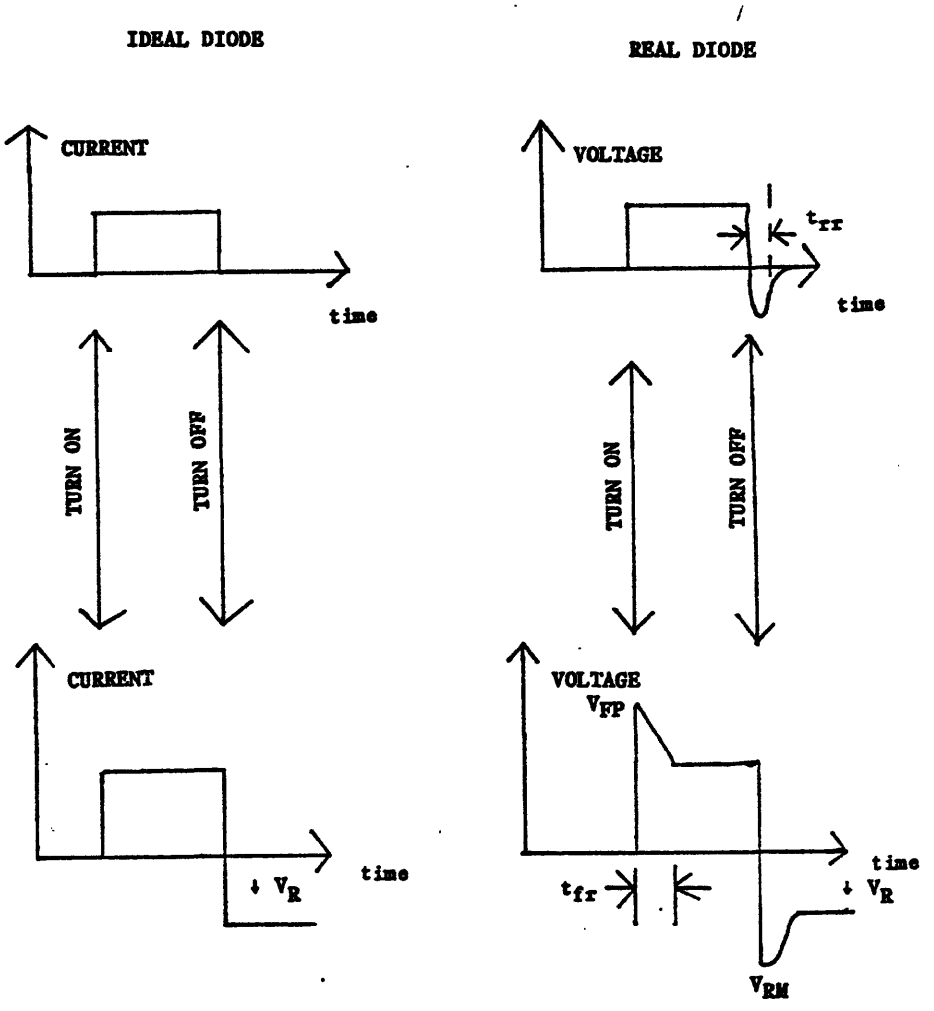


Figure 3.2
Current and voltage waveforms for real and ideal diodes

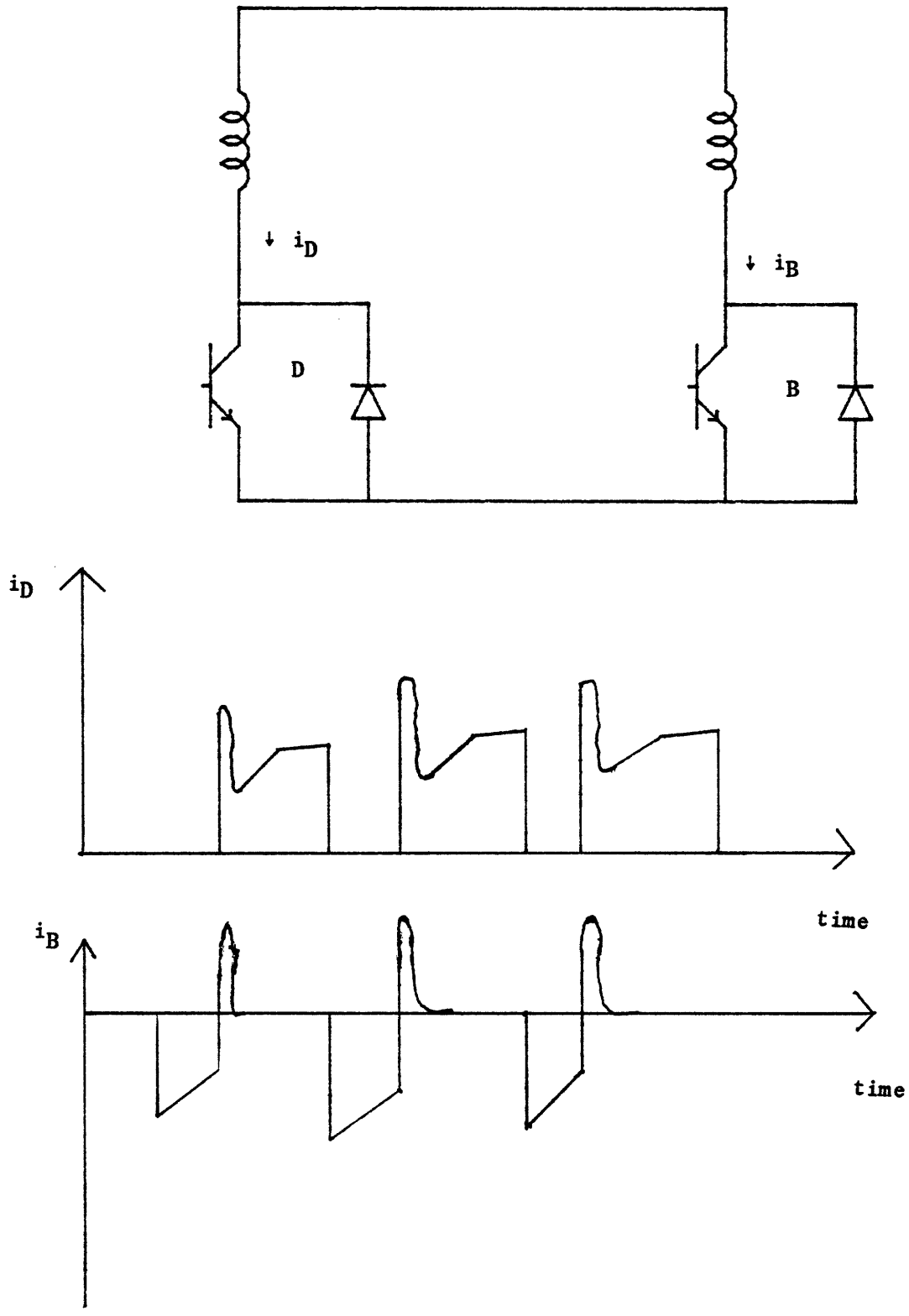


Figure 3.3 A highly schematic drawing of current spikes due to the use of slow diodes

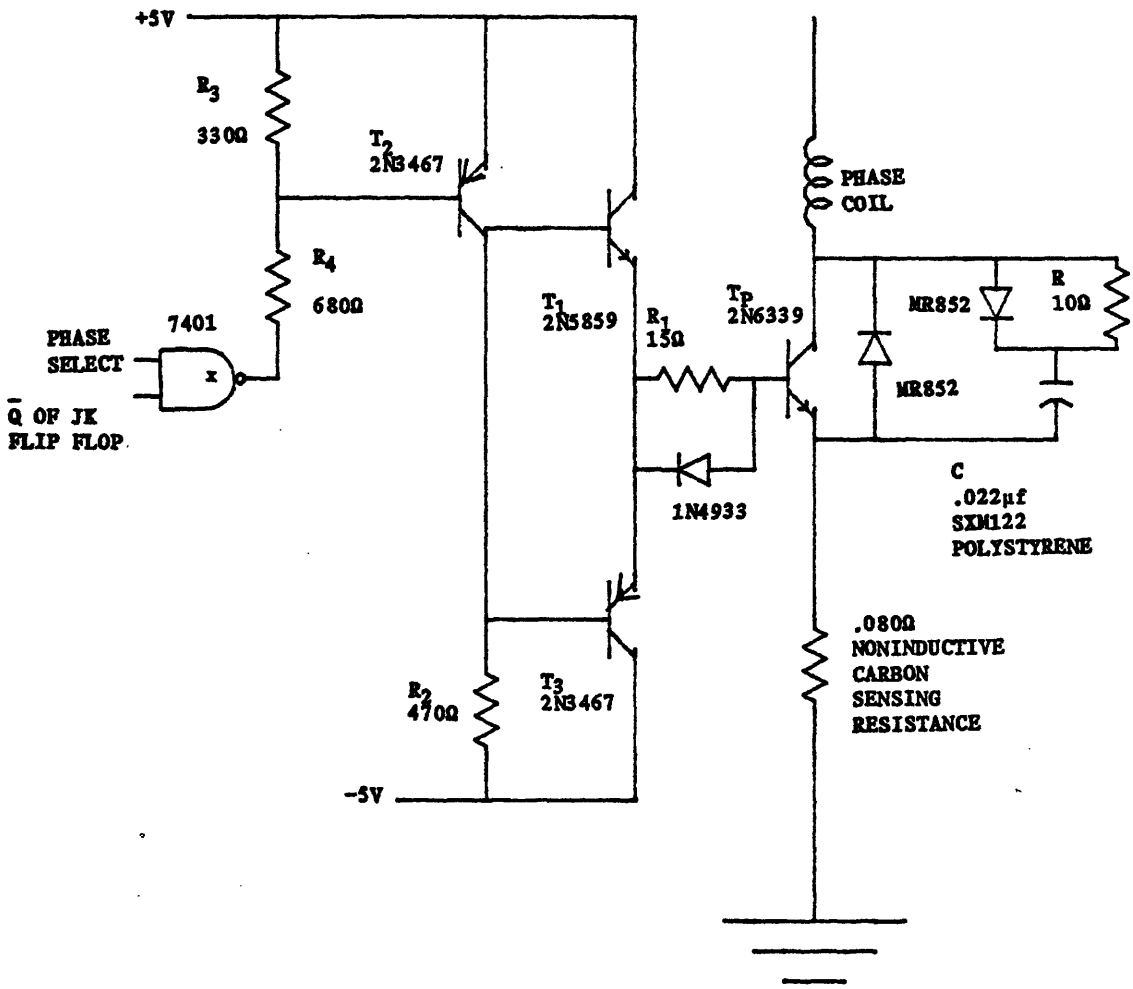


Figure 3.4 Switching network used for the 4 phase coils

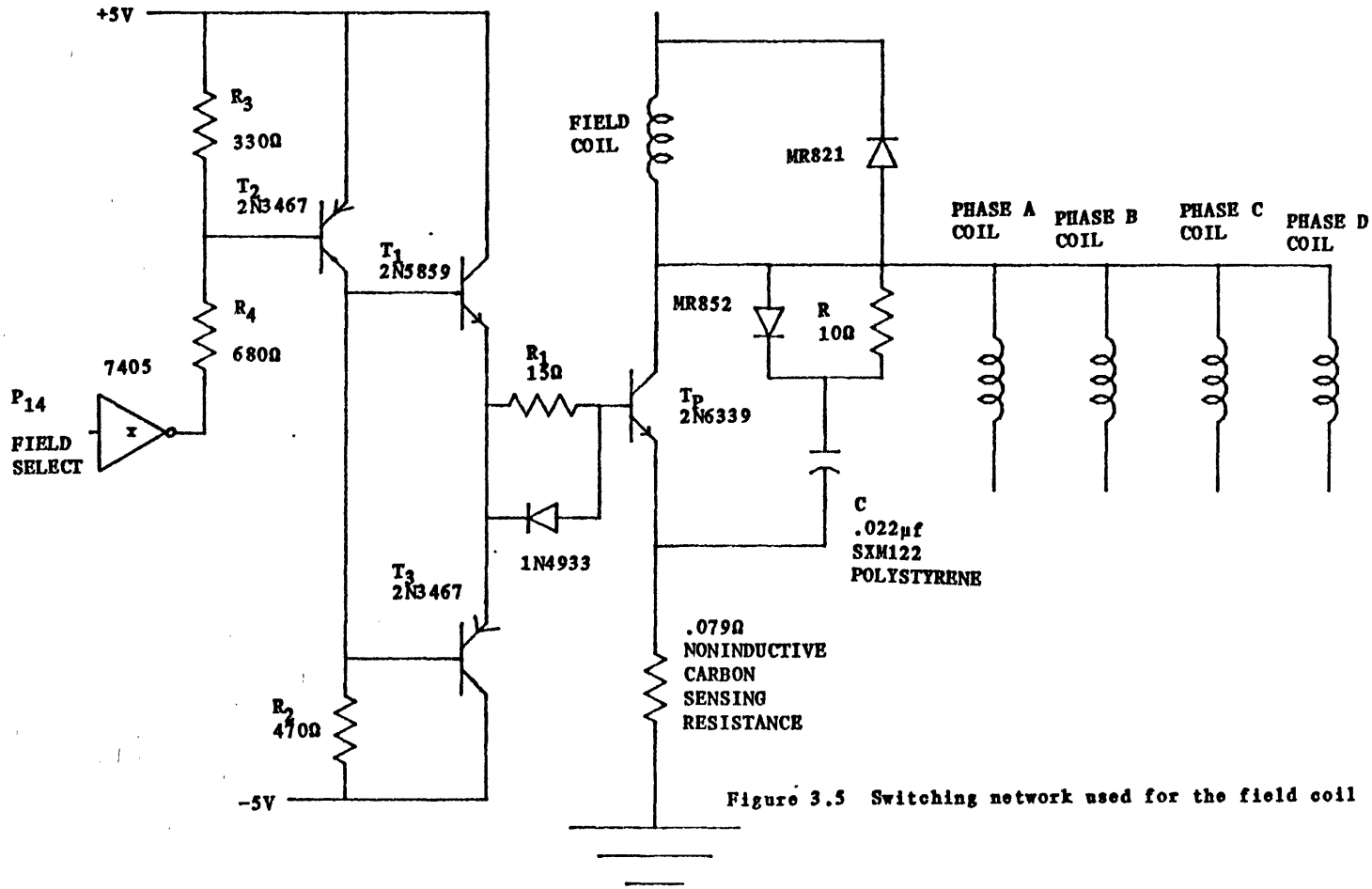


Figure 3.5 Switching network used for the field coil

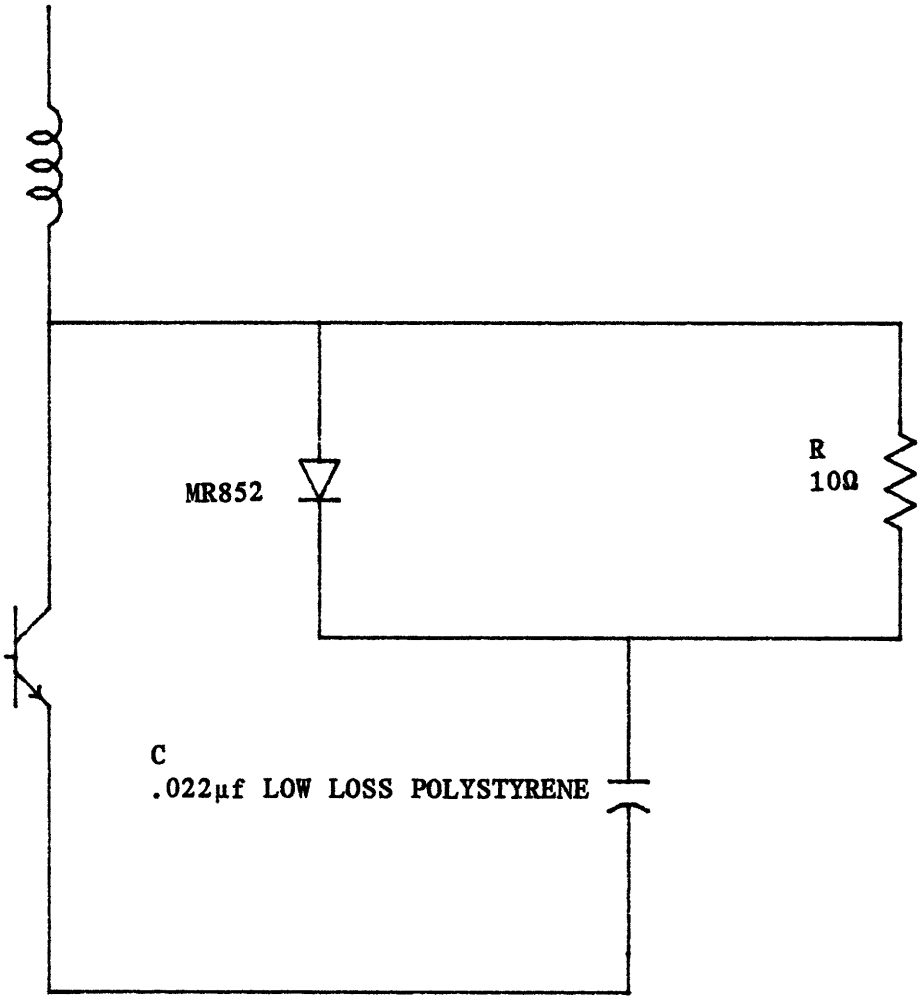


Figure 3.6 Snubber

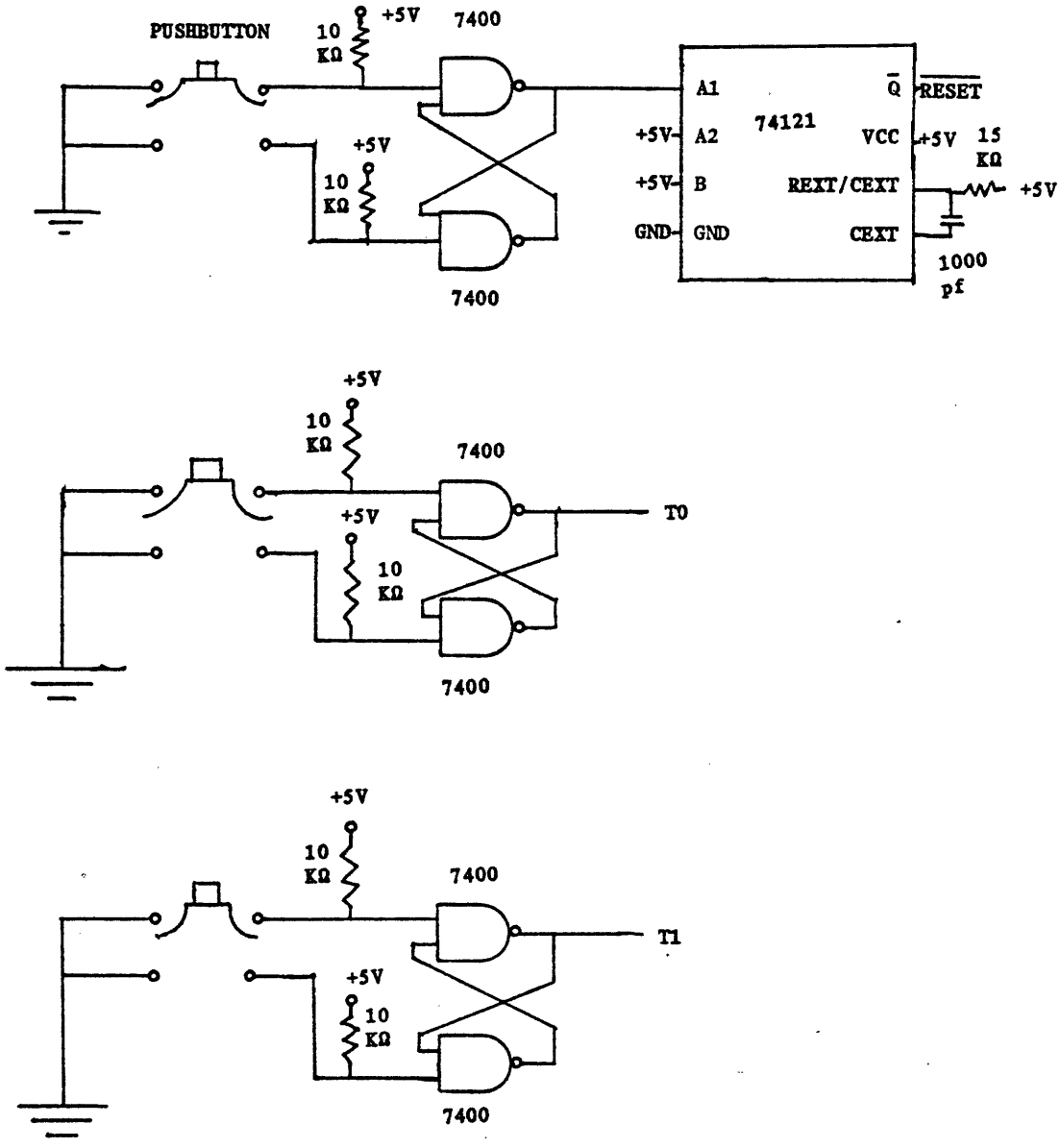


Figure 3.7 Pushbutton inputs to the 8039 microcomputer

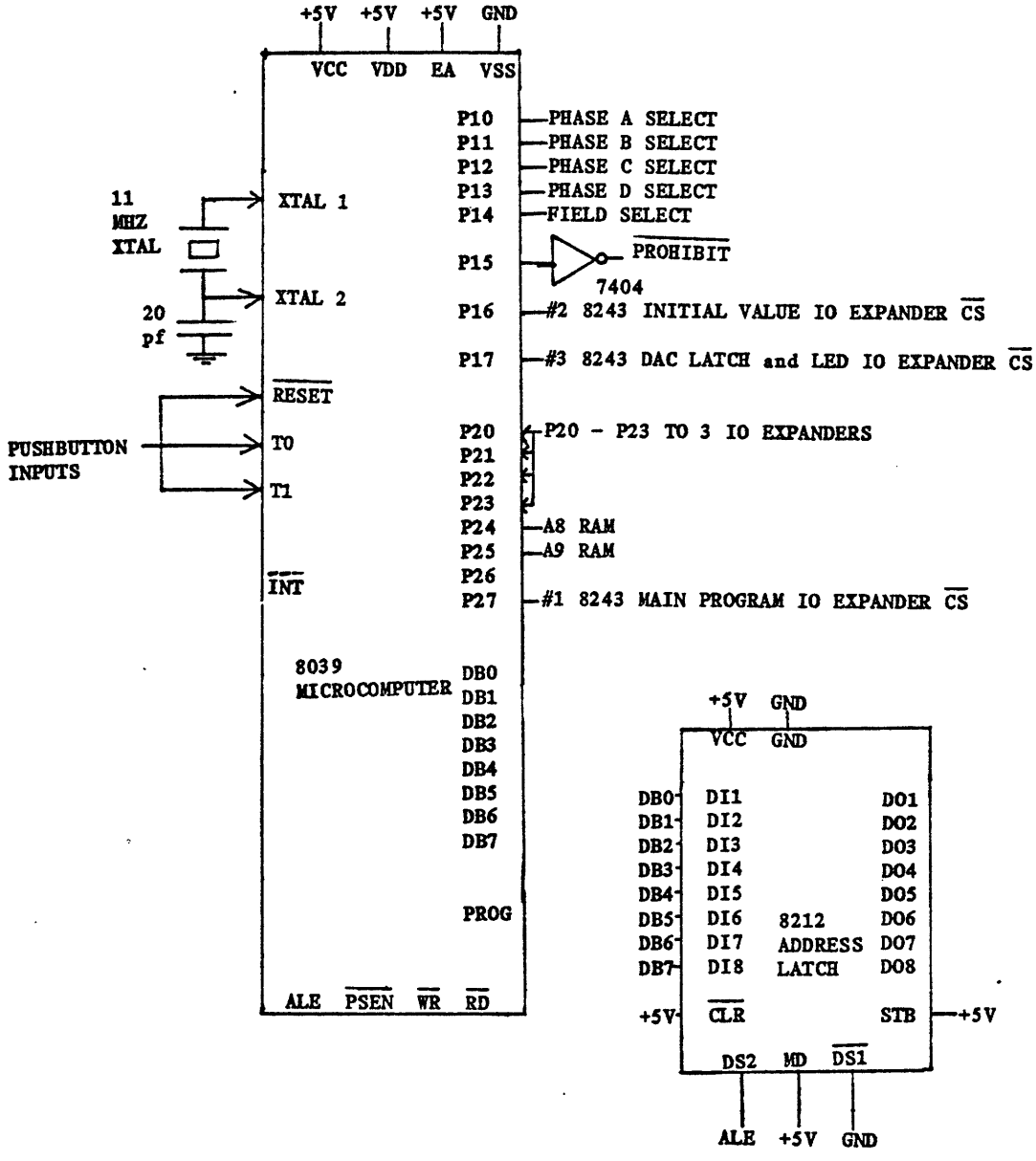


Figure 3.8a Intel microprocessor components

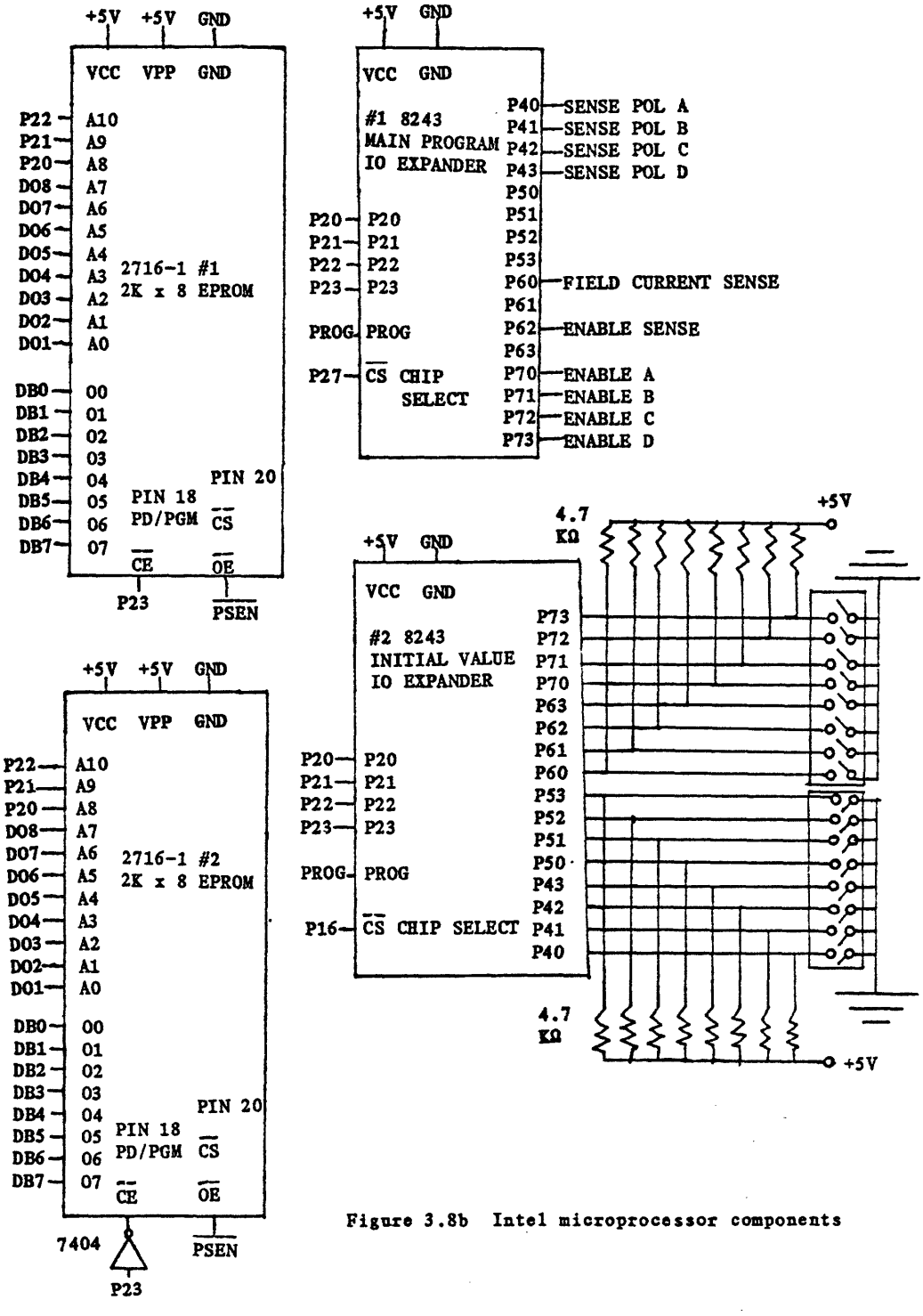


Figure 3.8b Intel microprocessor components

P73 to P40 also output to the DAC latches. See Figure 3.9.

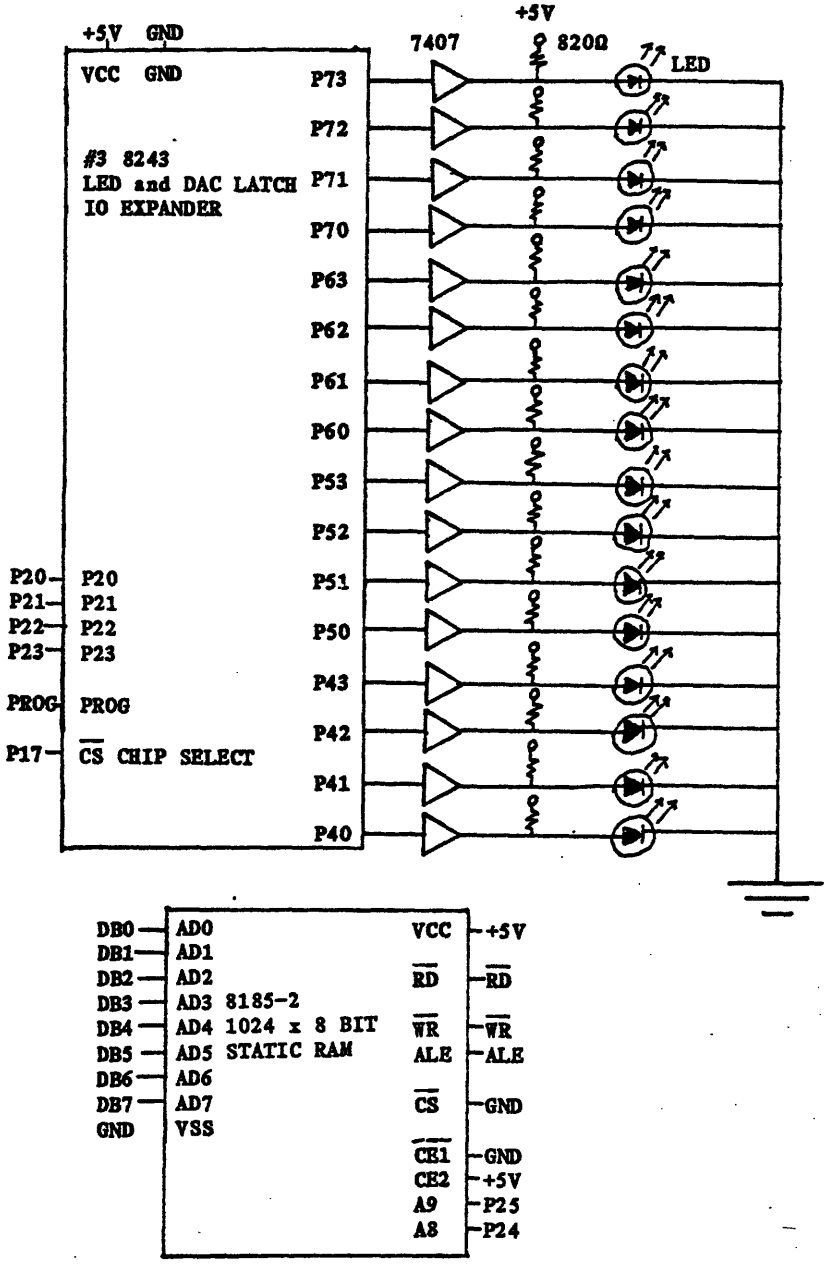


Figure 3.8c Intel microprocessor components

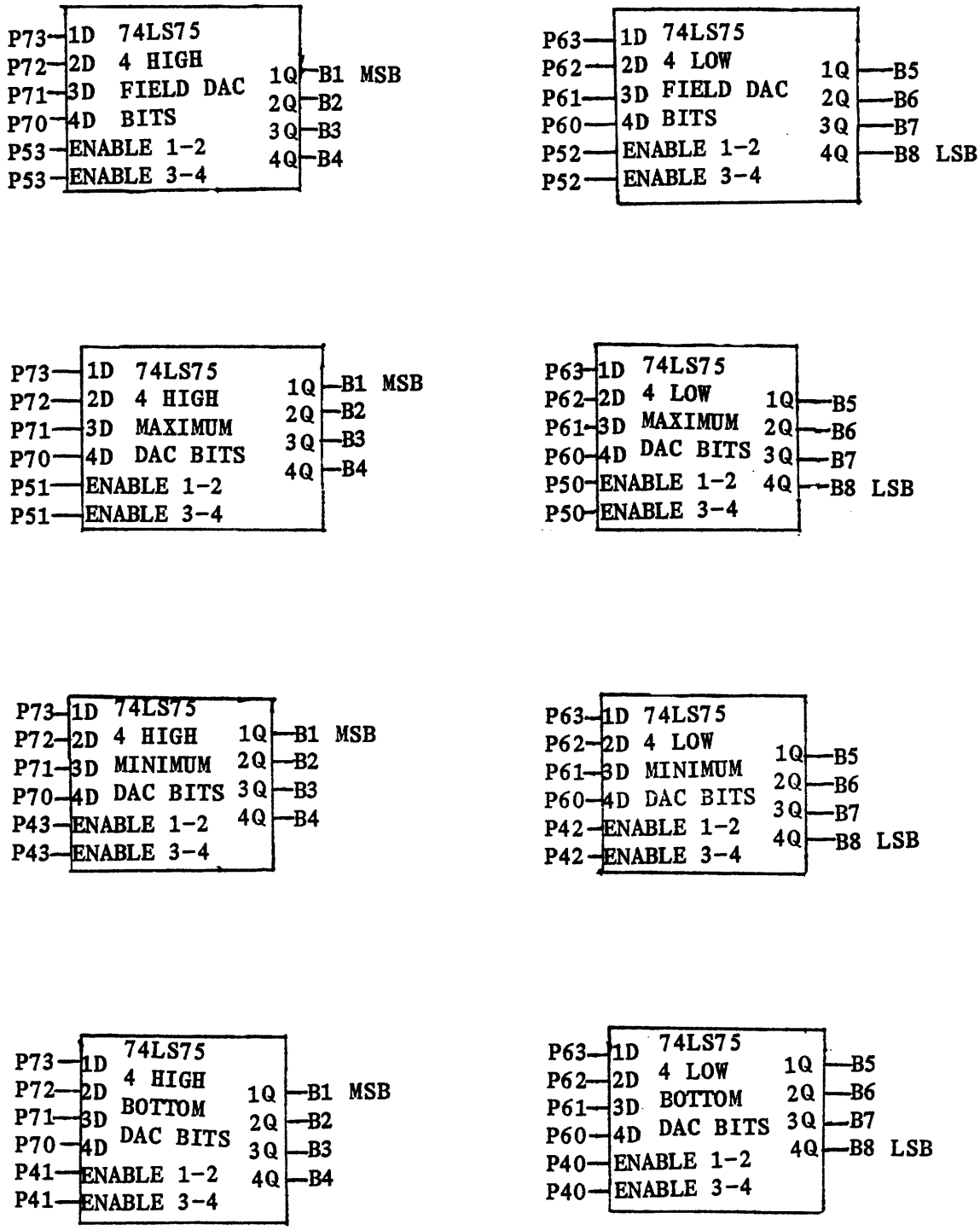
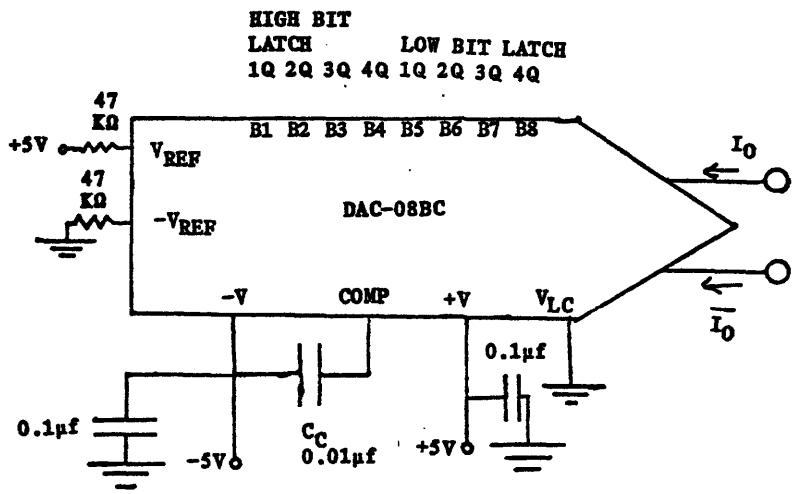
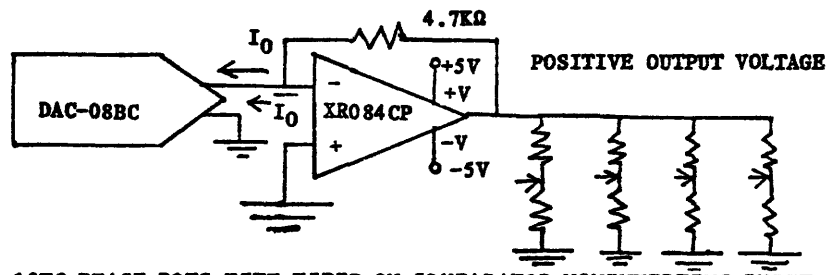


Figure 3.9 DAC Latches. The inputs to the DAC latches are from the #3 8243 IO expander and the outputs are to the 4 DACS.

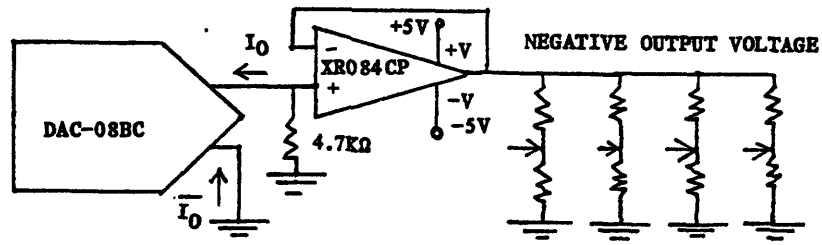


For field and maximum phase current levels:



4 10KΩ PHASE POTS WITH WIPER ON COMPARATOR NONINVERTING INPUT TERMINAL FOR MAXIMUM CURRENT LEVEL
1 5KΩ FIELD POT WITH WIPER ON COMPATATOR NONINVERTING INPUT TERMINAL FOR FIELD CURRENT LEVEL

For minimum and bottom phase current levels:



4 10KΩ PHASE POTS WITH WIPER ON COMPARATOR NONINVERTING INPUT TERMINAL FOR MINIMUM CURRENT LEVEL
4 10KΩ PHASE POTS WITH WIPER ON COMPATATOR INVERTING INPUT TERMINAL FOR BOTTOM CURRENT LEVEL

Figure 3.10 DAC op amp networks used to generate reference voltage levels

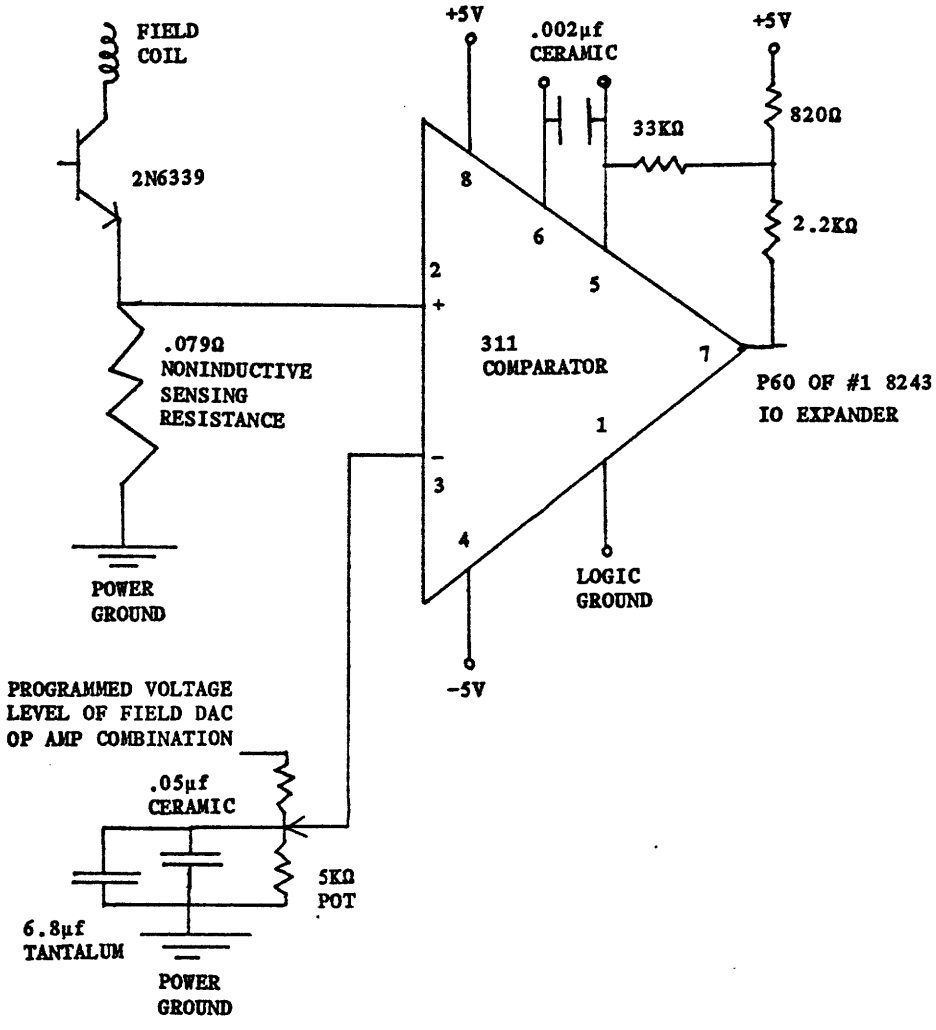


Figure 3.11 Network for sensing the field current level

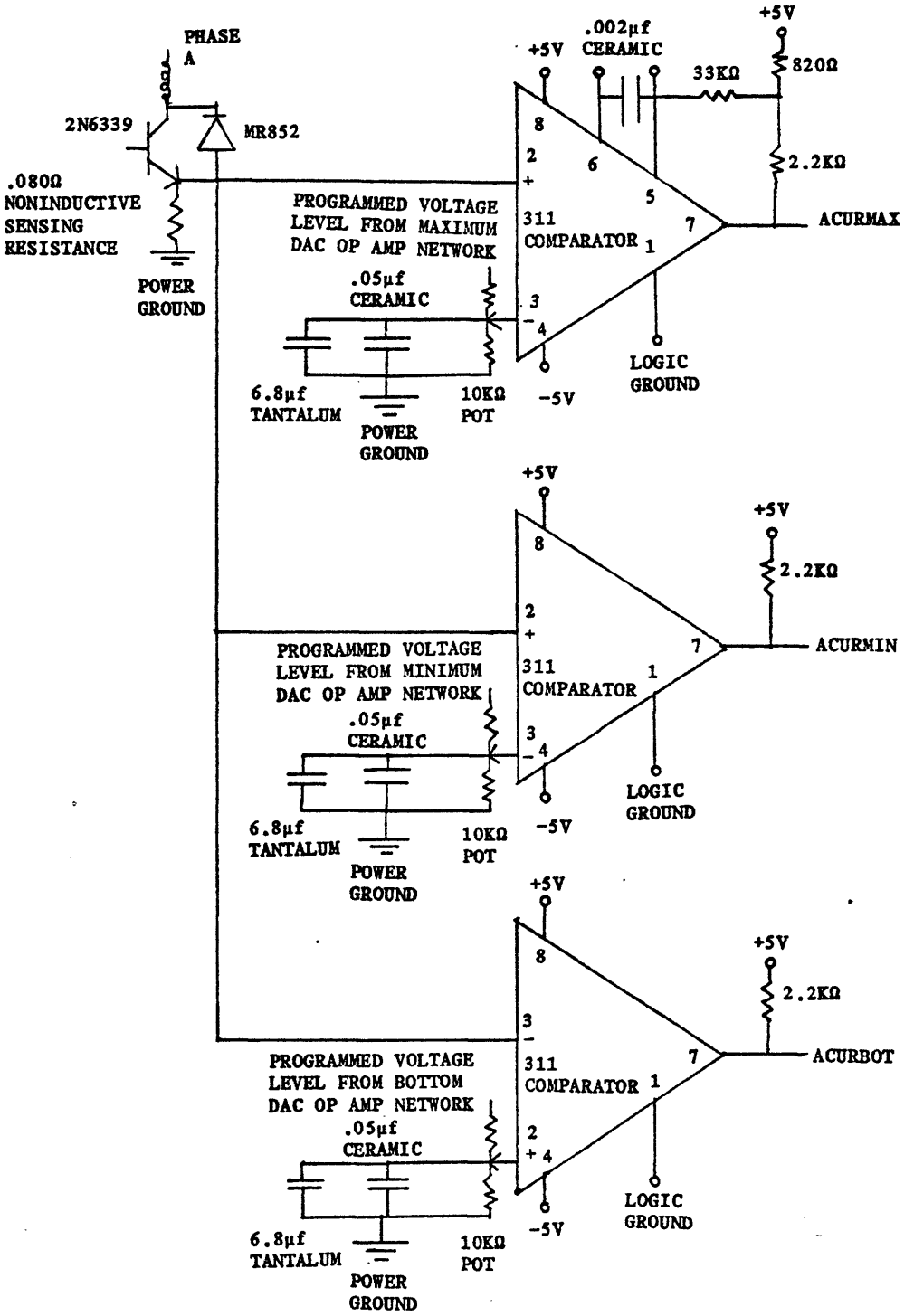


Figure 3.12a Network used for current regulation of phases A and C

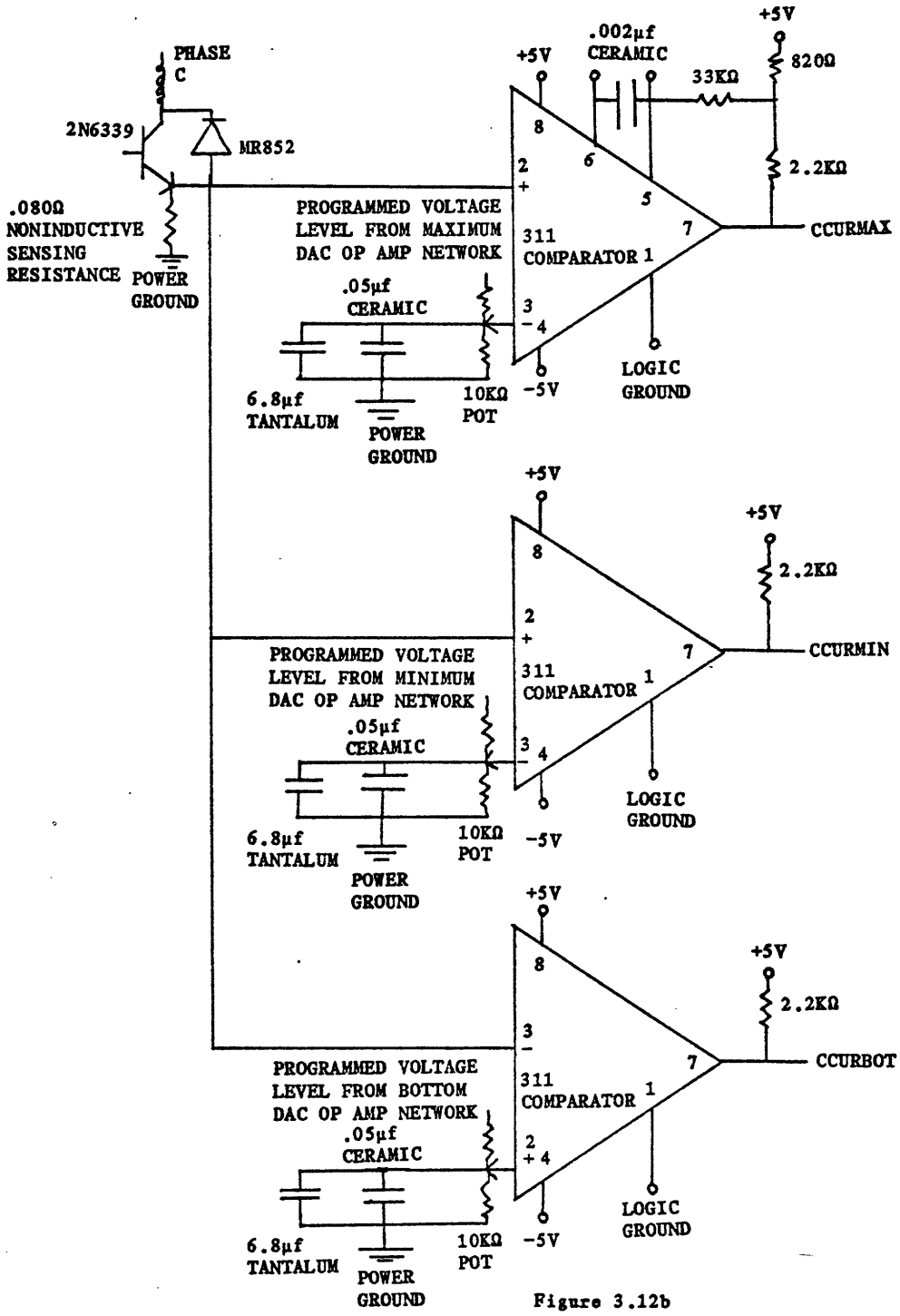

PHASE C
2N6339
MR852
.080Ω NONINDUCTIVE SENSING RESISTANCE
POWER GROUND
+5V
.002µf CERAMIC
33KΩ
820Ω
2.2KΩ
CCURMAX
PROGRAMMED VOLTAGE LEVEL FROM MAXIMUM DAC OP AMP NETWORK
.05µf CERAMIC
6.8µf TANTALUM
10KΩ POT
311 COMPARATOR
LOGIC GROUND
-5V
CCURMIN
PROGRAMMED VOLTAGE LEVEL FROM MINIMUM DAC OP AMP NETWORK
CCURBOT
PROGRAMMED VOLTAGE LEVEL FROM BOTTOM DAC OP AMP NETWORK

Figure 3.12b

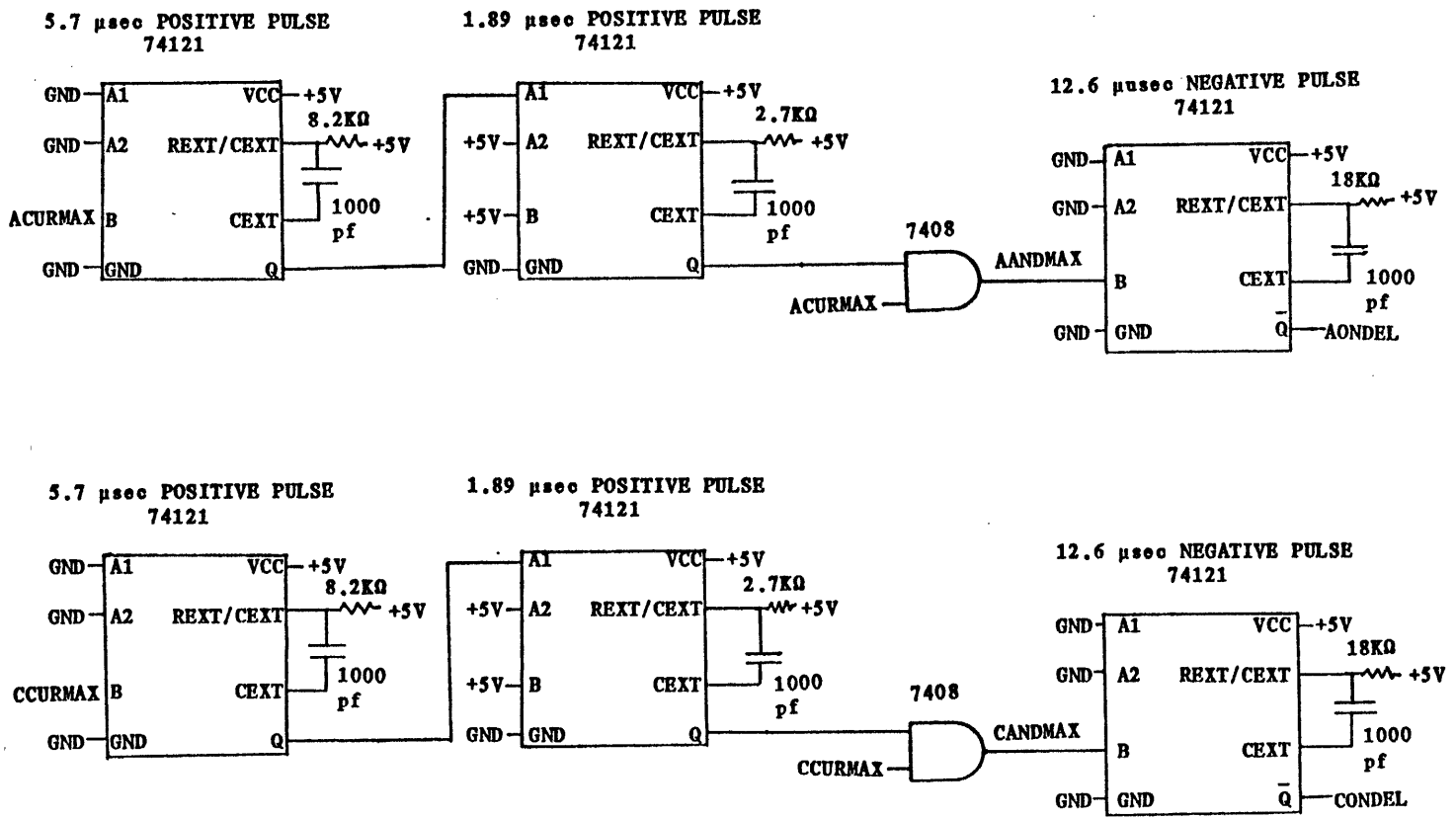

5.7 µsec POSITIVE PULSE
74121
1.89 µsec POSITIVE PULSE
74121
12.6 µsec NEGATIVE PULSE
74121
GND
A1
VCC
+5V
8.2KΩ
GND
A2
REXT/CEXT
+5V
ACURMAX
B
CEXT
1000
pf
GND
GND
Q
+5V
A2
REXT/CEXT
2.7KΩ
+5V
+5V
B
CEXT
1000
pf
GND
GND
Q
7408
ACURMAX
AANDMAX
GND
A1
VCC
+5V
18KΩ
GND
A2
REXT/CEXT
+5V
B
CEXT
1000
pf
GND
GND
Q
AONDEL
5.7 µsec POSITIVE PULSE
74121
1.89 µsec POSITIVE PULSE
74121
12.6 µsec NEGATIVE PULSE
74121
GND
A1
VCC
+5V
8.2KΩ
GND
A2
REXT/CEXT
+5V
CCURMAX
B
CEXT
1000
pf
GND
GND
Q
+5V
A2
REXT/CEXT
2.7KΩ
+5V
+5V
B
CEXT
1000
pf
GND
GND
Q
7408
CCURMAX
CANDMAX
GND
A1
VCC
+5V
18KΩ
GND
A2
REXT/CEXT
+5V
B
CEXT
1000
pf
GND
GND
Q
CONDEL

Figure 3.12c

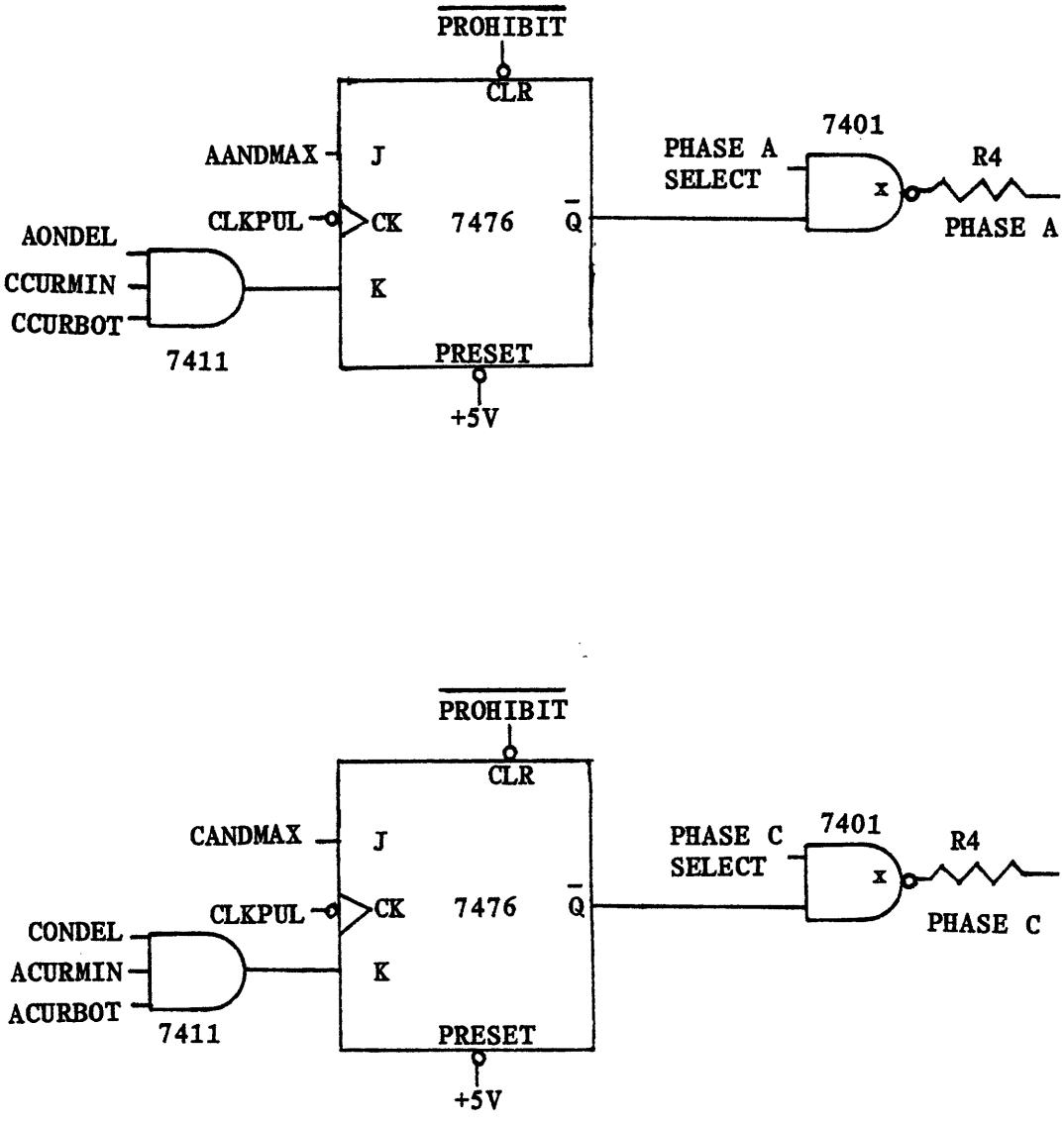

PROHIBIT
CLR
AANDMAX
J
CLKPUL
CK
7476
Q
AONDEL
CCURMIN
CCURBOT
K
7411
PRESET
+5V
7401
PHASE A
SELECT
x
R4
PHASE A
PROHIBIT
CLR
CANDMAX
J
CLKPUL
CK
7476
Q
CONDEL
ACURMIN
ACURBOT
K
7411
PRESET
+5V
7401
PHASE C
SELECT
x
R4
PHASE C

Figure 3.12d

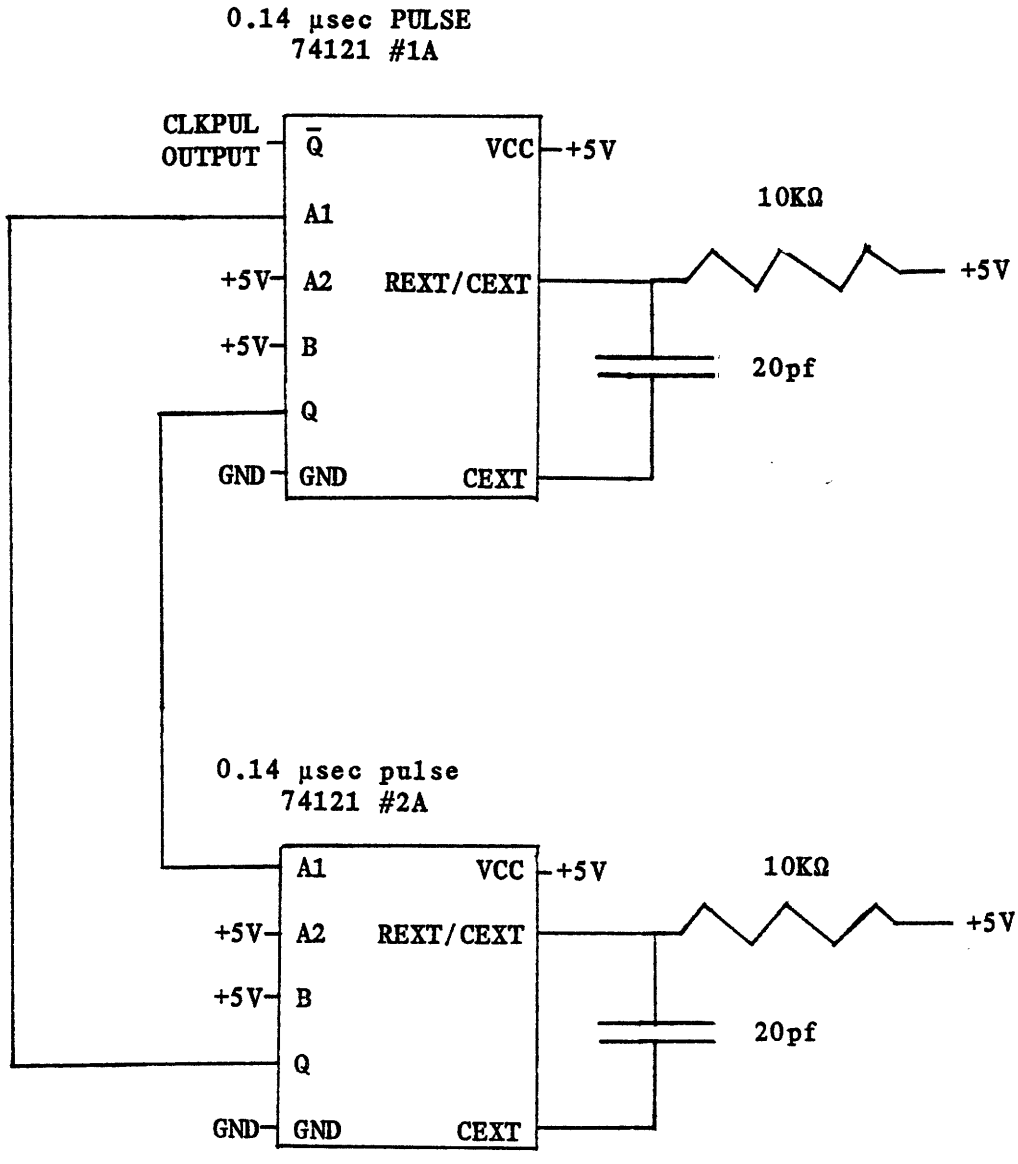

0.14 µsec PULSE
74121 #1A
CLKPUL
OUTPUT
Q
VCC
+5V
A1
10KΩ
+5V
A2
REXT/CEXT
+5V
+5V
B
20pf
Q
GND
GND
GND
CEXT
0.14 µsec pulse
74121 #2A
A1
VCC
+5V
10KΩ
+5V
A2
REXT/CEXT
+5V
+5V
B
20pf
Q
GND
GND
CEXT

Figure 3.12e

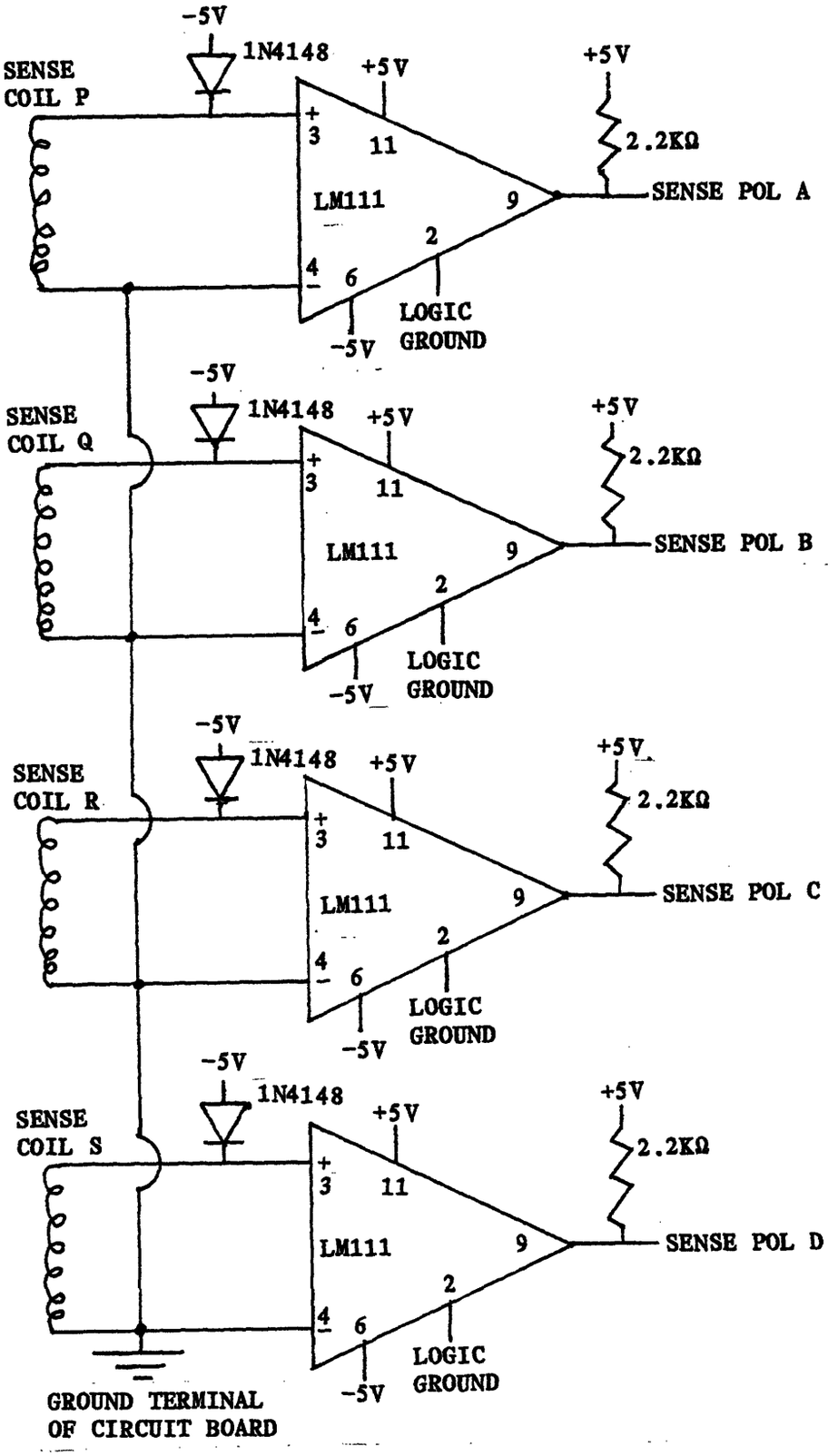


Figure 3.13 Comparators determine the sense coil voltage polarities.

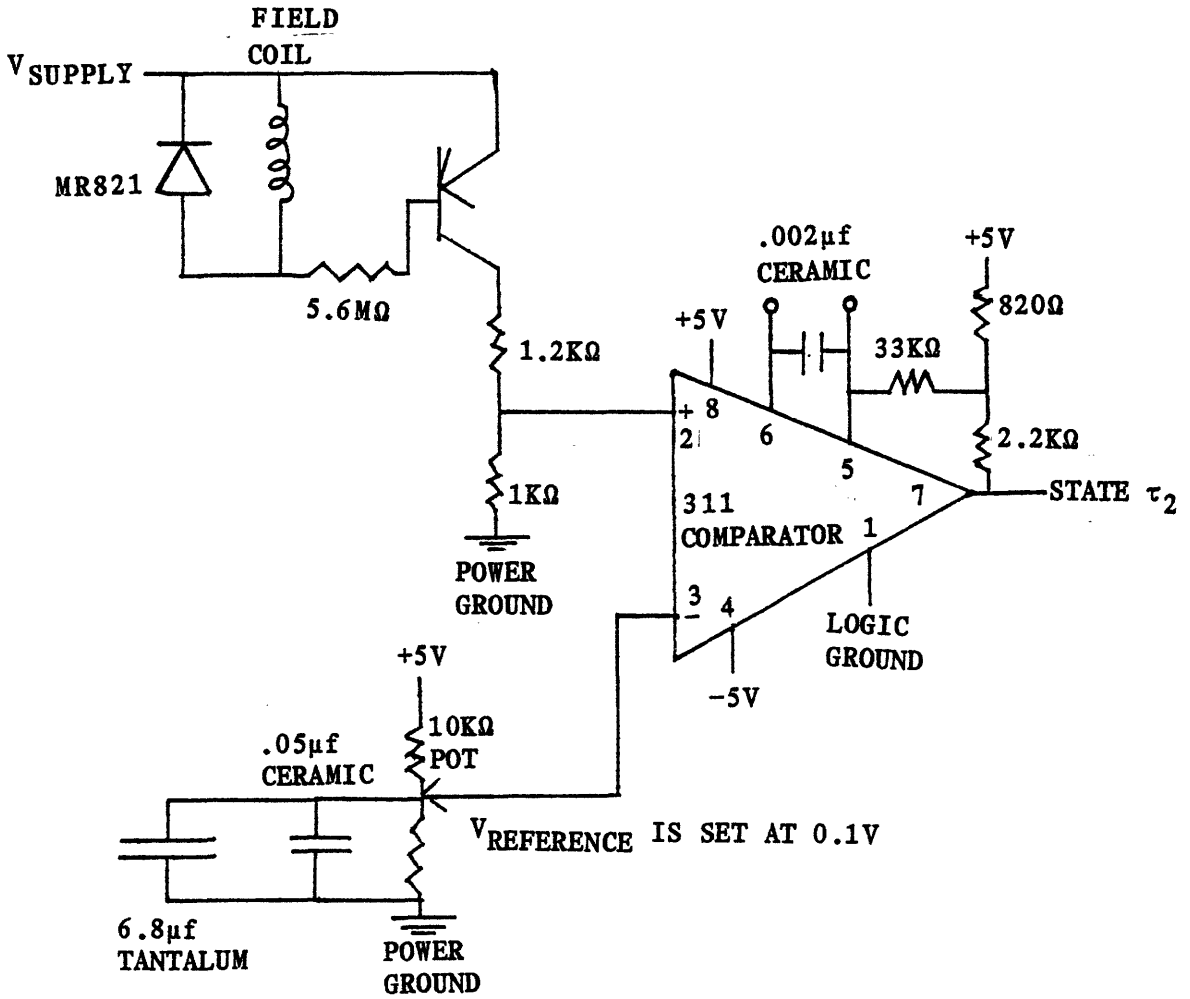


Figure 3.14 The comparator detects the presence of state τ_2

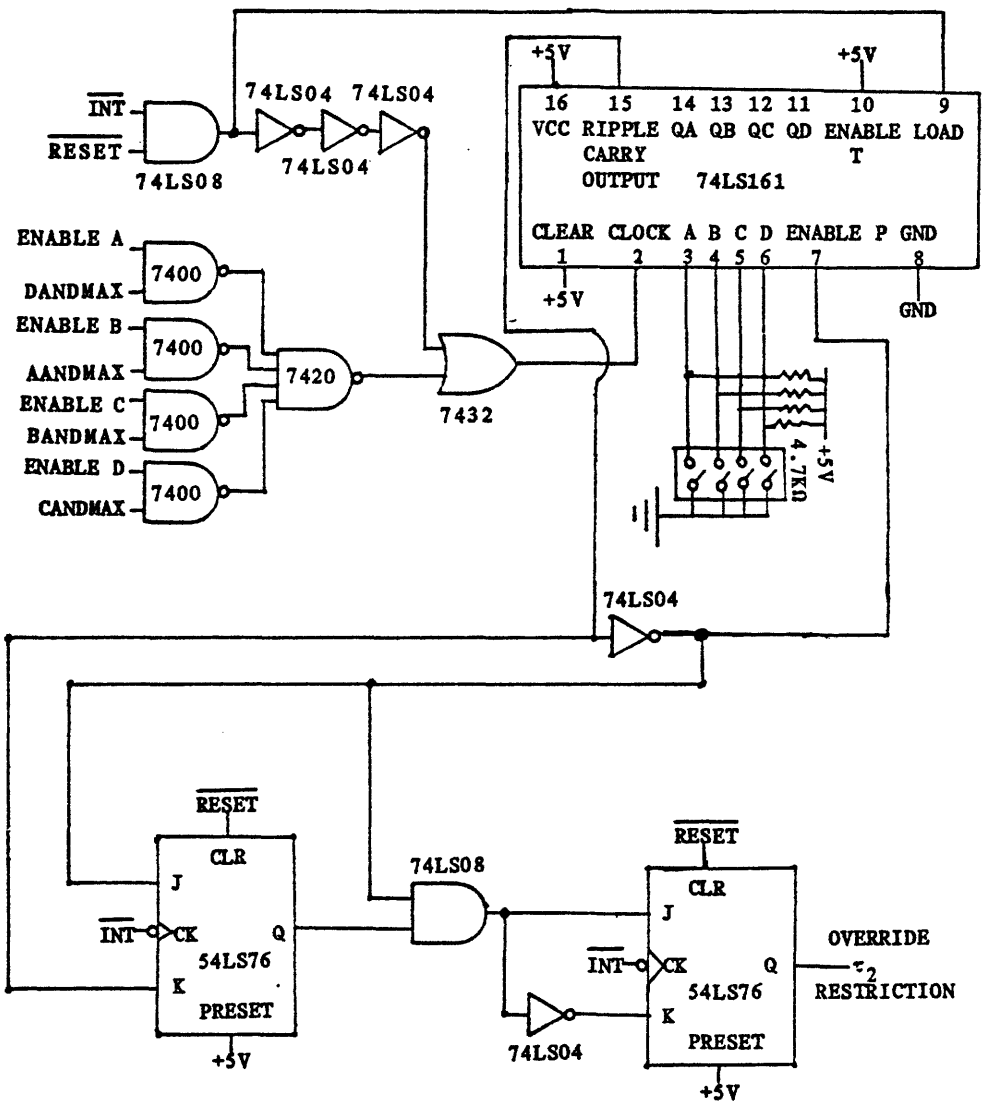


Figure 3.15 Network determines if a τ_2 state is an external interrupt requirement.

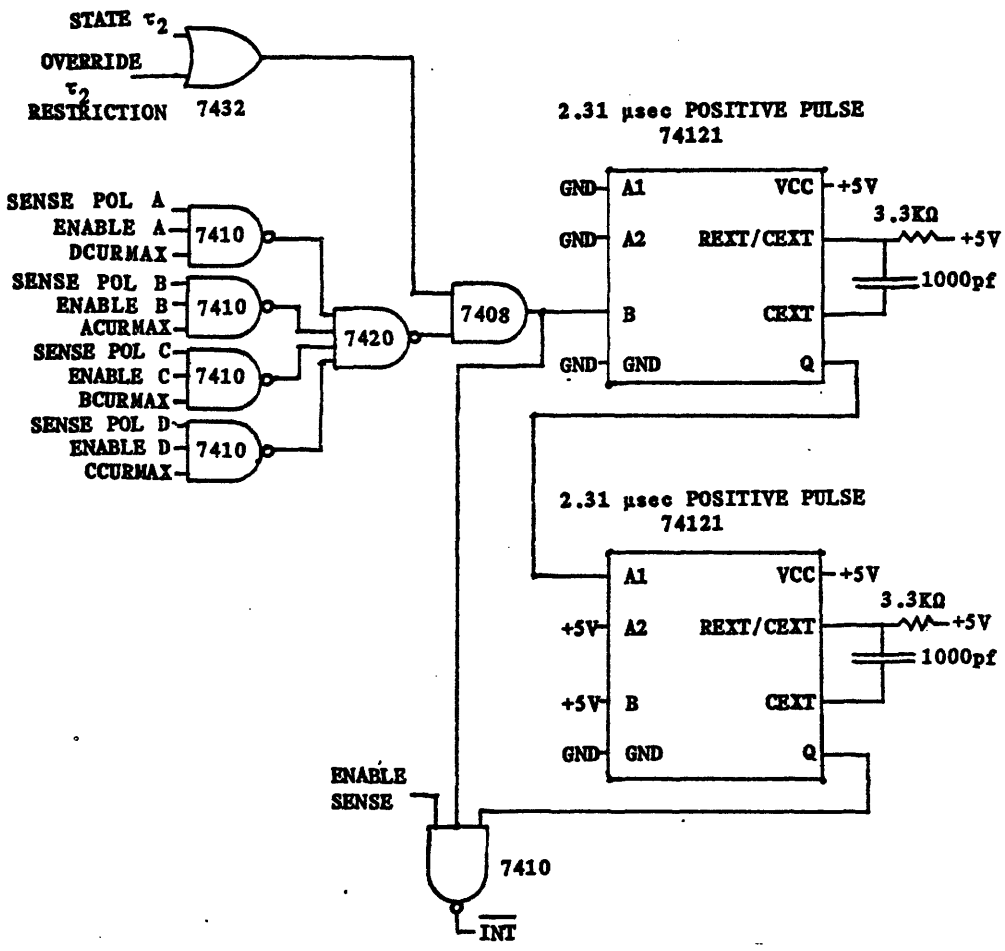


Figure 3.16 Network generating the external interrupt to the microcomputer

CHAPTER 4 SOFTWARE ISSUES AND RESULTS

1. Simple Test Programs

An important part of a good design procedure for a system is the inclusion of simple test programs used for checking or debugging the circuitry. Two such programs are included here.

The first one is titled D TO A SECTION TEST. This program sends reference voltage levels to the appropriate comparator input terminals. This allows an information route going from the input data switches to the input switch I/O expander to the 8039 microcomputer to the LED/DAC I/O expander to the 74LS75 DAC latches to the DACs to the op amps to the pots to the comparator input terminals to be easily debugged.

The second test program is titled TRANSISTOR SWITCHING TEST. With this program any of the 5 transistor switching banks can be repetitively switched. A phase bank will be repetitively switched between a specified minimum and maximum current level. If the field transistor bank is tested, the field current is brought up to the specified field current level and then the field transistor is turned off for 44.6 milliseconds, a time duration which allows the field current sufficient time to decay to zero. Then, the field transistor bank is switched on again. This program allows easy debugging of the transistor switching banks.

2. The RUN THE MOTOR Program

In the program titled RUN THE MOTOR the current level values

used at startup are replaced with a second set of values when a button is pressed. Thus, since the motor's greatest current requirement occurs at startup, the phase current levels can be decreased once steady state motion is achieved. This program will now be examined in some detail.

The initial reset state sets all the output ports to one thereby turning on all the transistors. The transistors are turned off, the input switch I/O expander is enabled, and the LED/DAC I/O and main program I/O expanders are disabled. When T0 is pressed, the field and first maximum values are inputted to RAM locations on the 8039 via the input switch I/O expander. When T1 is pressed, the first minimum and bottom values are inputted. Then, T0 is pressed to input the second maximum value. Finally, T1 is pressed to input the second minimum and bottom values.

T0 is pressed to start the motor running. The LED/DAC I/O expander is enabled and the input switch I/O expander is disabled. The first set of current level values stored in the 8039 RAM are outputted from the 8039 RAM to the DAC latches via the LED/DAC I/O expander. Then, the LED/DAC I/O expander is disabled and the main program I/O expander is enabled. After an 18 unit timer wait, the field transistor is turned on until the field current level rises to the desired value. A delayed check of the field current level is made for extra certainty. Then, an initial position sensing is made by inputting the sense coil voltage polarities to the main program I/O expander. The initial sensing information causes the phase located between 180 and 90 degrees from its equilibrium position to be activated as the

field transistor is turned off. The location containing the first interrupt address is stored in R3, the timer is started from zero, and the external interrupt is enabled. Before the current levels are changed, when the program is waiting for the first interrupt or is between interrupts, it will circle in a simple loop testing for six timer overflows since the last phase turnon and for the pressing of the T1 button.

The generation of an external interrupt will cause the execution of the first of the program's two external interrupt routines. The first interrupt routine turns off the old phase, turns on the new phase, restarts the timer from zero, updates the phase registers containing the present and next phase information, and resets R4 to allow for another 10 restart attempts.

Before the current levels are changed, after a new phase has been activated, if more than 6 timer overflows occur before an external interrupt indicating a next phase detection occurs, then a restart occurs. In a restart the phase transistor is turned off and the program is reentered earlier at the MREST location where the 18 unit timer wait occurs just before the field transistor is turned on. Ten consecutive restart attempts are allowed. After ten consecutive unsuccessful restart attempts the transistors are turned off until the T0 button is pressed again. Pressing the T0 button resets R4 to allow another 10 restart attempts and causes the program to be reentered at the MREST location.

When the T1 button is pressed, the location containing the

second interrupt address is stored in R3 and the next interrupt generated will cause the execution of the second rather than the first interrupt routine. The instructions at the beginning of the second interrupt routine are identical to those in the first routine. Then, it disables the main program I/O expander and enables the LED/DAC I/O expander. The second set of current level values are then moved out from the 8039 RAM to the DAC latches replacing the first set of values. The LED/DAC I/O expander is disabled and the main program I/O expander is enabled. Finally, the address of the first interrupt routine is stored in R3 so that all further interrupts will result in execution of the first interrupt routine. Between interrupts the program circles in a simple loop testing for six timer overflows since the new phase activation. If six timer overflows occur, then the program is reentered earlier at the START2 location where R4 is reset to allow 10 consecutive restart attempts just before the first set of current level values is outputted to the DAC latches.

This program will accelerate the motor up to a maximum speed of about one chop per phase relatively quickly. Although most of the phase intervals contain only one chop, an occasional phase has more than one so the speeds obtained are slightly less than would be obtained by purely one chop per phase. Because the current rate of change across an inductance is proportional to voltage, then at higher voltages the amount of time per chop is less and hence the speed is greater. Also, since higher current levels take longer to achieve, once a current level that can maintain a speed of one chop per phase is reached, further

increases in the current level increase the time per chop and hence decrease the speed. At 20V with the current levels kept between 1.75 and 3.15 amps, a speed of 1625 R.P.M. is obtained.

3. Speed Control Programs

The program titled CONSTANT SWITCHING DELAY allows constant load speed control. A time delay is put at the start of every interrupt routine so that a time delay exists exists between the detection of the new phase and the switching on of the new phase. The 8 low bits of time delay are stored in R1 and the 8 high bits of time delay are stored in R2. The first interrupt routine takes advantage of these stored timed delays by starting as follows:

```
SERVE1:  DJNZ R1,SERVE1 ; DECREMENT THE 8 LOW BITS OF TIME DELAY
         DJNZ R2,SERVE1 ; DECREMENT THE 8 HIGH BITS OF TIME DELAY
```

The execution of a DJNZ R_r(decrement register and jump if the contents are not zero) instruction requires 2 instruction cycles or 2.72 microseconds. Thus, time delays can be changed in increments of 2.72 microseconds. Unfortunately, the use of 2 DJNZ instructions adds 5.44 microseconds to the minimum possible delay since both DJNZ instructions must be passed through at least one time during the execution of an interrupt routine.

Steady state speed decreases as time delay increases. Measurements of speed resulting from the inputted constant time delays were obtained for the unloaded motor running with a 20V supply voltage and the phase currents kept between 1.75 and 3.15 amps over a speed range going from 17.6 to 1625 R.P.M. At speeds

equal to or greater than 108 R.P.M. a stroboscope was used for measurements. For speeds less than 108 R.P.M. visual counting was performed with the aid of a stopwatch or a pushbutton electronic counter. The results are shown in Table 4.1 with speed in R.P.M. and 1/(speed in R.P.M.) resulting from the inputted constant time delay in units of 2.72 microseconds(exceeding the minimum possible 5.44 microsecond delay caused by one pass through the 2 DJNZ instructions).

Plots of time delay in units of 2.72 microseconds versus 1/(speed in R.P.M.) are shown in Figures 4.1a-c. These plots of delay units versus 1/speed show a very good linear fit. Performing a least squares fit on the 131 data points with a HP33C calculator yields the linear equation:

delay in units of 2.72 microseconds = 3.190×10^5(1/(speed in R.P.M.)) - 130.3

with an excellent correlation coefficient of .99957. As mentioned in Chapter 1 B.H. Wells expected an equation relating delay and speed of the above type to result for a step motor whose unloaded speed versus switching angle curve was nearly linear.[1]

The program titled VARIABLE SWITCHING DELAY runs the motor at a fixed speed by continuously varying the time delay between new phase detection and new phase switching. The 8039 timer is used to measure the time interval for 4 consecutive phases so as to eliminate measurement problems caused by differences in the individual phases. A calculation of the actual time interval for 4 consecutive phases minus the desired time interval for 4 consecutive phases is performed using 2 register arithmetic. If

the result is positive, then the motor speed is too slow and the delay time is decreased. If the result is negative, then the motor speed is too fast and the delay time is increased. If the result is zero, then the speed is correct and the delay time is left unchanged. The number of phases that are to elapse between every set of 4 consecutive phases that is used for speed correction is fed in on the input data switches. At low speeds the maximum correction rate is compatible with good speed regulation, but at high speeds the maximum correction rate causes marked fluctuations in speed to occur. Decreasing the correction rate alleviates this problem. This program could run the unloaded motor with a 20V supply voltage and phase currents kept between 1.75 and 3.15 amps over a speed range going from 100 R.P.M. to 1640 R.P.M. Because feedback is used, this program can be employed in varying load situations.

The program titled CONSTANT SPEED VIA VARYING CURRENT runs the motor at a fixed speed by continuously varying the phase current levels. The minimum phase current level is always set equal to half the maximum level. The 8039 timer is used to measure the time interval for 4 consecutive phases. A calculation of the actual time interval for 4 consecutive phases minus the desired time interval for 4 consecutive phases is performed using 2 register arithmetic. If the result is positive, then the motor speed is too slow and the current level is increased. If the result is negative, then the motor speed is too fast and the current level is decreased. If the result is zero, then the speed is correct and the current level is left

unchanged.

The number of phases that are to elapse between every set of 4 consecutive phases that is used for speed correction is fed in on the input data switches. Table 4.2 shows the speed fluctuations around an average speed of 1565 R.P.M. resulting in this varying current scheme from changing the number of phases elapsing between every set of 4 consecutive phases that is used in speed correction. The best speed correction occurs with 20 to 24 interspersed phases. A narrow range of updating rates yields maximum speed stability, and updating either more or less frequently increases the speed fluctuations.

This program can only run the motor with average speeds ranging from 1400 to 1710 R.P.M. The upper speed bound occurs because the time per phase chop increases as current level increases so once a speed near one chop per phase is reached further current level increases decrease the speed. The lower speed bound occurs because the current starts to fall to zero very precipitously.

4. Acceleration Profile Program

The program titled SUMMARY ACCELERATION PROFILE runs the motor from rest to full speed while storing in the 8185-2 RAM timer duration counts for 512 sets of the designated number of consecutive phases. In this program the timer interrupt is used to increment the 8 upper time bits when a time counter overflow occurs. After 512 sets have been stored in the RAM, all transistors are turned off so the motor stops. Then, T0 and T1 are alternately pressed to read out the timer duration counts in

the LEDs.

Three trials were performed with timer counts obtained for sets of 60 consecutive phases or 1 revolution.(The program titled TRANSISTOR SWITCHING TEST can be used to prove that 60 phases occur per revolution by single stepping the motor in an A-B-C-D-A activation sequence.) In these trials measurements were made for the unloaded motor running at 20V supply voltage and phase currents kept between 1.75 and 3.15 amps. Table 4.3 shows the number of timer counts in each of the first 100 revolutions for each trial. The final average speed is taken as that speed given by averaging the timer counts of the last 20 revolutions. Half final average speed is obtained at some point from 14 to 16 revolutions and 9/10 final average speed is obtained at some point from 39 to 43 revolutions.

Table 4.1 Speed in R.P.M. and 1/(speed in R.P.M.) resulting from the inputted constant time delays in units of 2.72 microseconds.

DELAY IN UNITS OF 2.72 MICROSECONDS	SPEED IN R.P.M.	1/(SPEED IN R.P.M.)
0	1625	6.154×10^{-4}
4	1610	6.211×10^{-4}
6	1595	6.270×10^{-4}
8	1580	6.329×10^{-4}
10	1565	6.390×10^{-4}
12	1550	6.452×10^{-4}
14	1540	6.494×10^{-4}
16	1530	6.536×10^{-4}
20	1515	6.601×10^{-4}
24	1490	6.711×10^{-4}
28	1470	6.803×10^{-4}
32	1455	6.873×10^{-4}
36	1435	6.969×10^{-4}
38	1420	7.042×10^{-4}
40	1410	7.092×10^{-4}
44	1390	7.194×10^{-4}
48	1370	7.299×10^{-4}
50	1360	7.353×10^{-4}
52	1345	7.435×10^{-4}
56	1325	7.547×10^{-4}
60	1305	7.663×10^{-4}
64	1285	7.782×10^{-4}
68	1265	7.905×10^{-4}
72	1250	8.000×10^{-4}
76	1235	8.097×10^{-4}
80	1220	8.197×10^{-4}
84	1210	8.264×10^{-4}
88	1200	8.333×10^{-4}
96	1180	8.475×10^{-4}
100	1165	8.584×10^{-4}
104	1150	8.696×10^{-4}
108	1140	8.772×10^{-4}
112	1125	8.889×10^{-4}
116	1115	8.969×10^{-4}
120	1110	9.091×10^{-4}
124	1090	9.174×10^{-4}
128	1080	9.259×10^{-4}
136	1060	9.434×10^{-4}
140	1050	9.524×10^{-4}
144	1035	9.662×10^{-4}
152	1015	9.852×10^{-4}
160	995	1.005×10^{-3}
168	975	1.026×10^{-3}
176	955	1.047×10^{-3}
184	935	1.070×10^{-3}
192	915	1.093×10^{-3}
200	900	1.111×10^{-3}
208	880	1.136×10^{-3}
216	860	1.163×10^{-3}

DELAY IN UNITS OF 2.72 MICROSECONDS	SPEED IN R.P.M.	1/(SPEED IN R.P.M.)
224	840	1.190×10^{-3}
232	830	1.205×10^{-3}
240	815	1.227×10^{-3}
248	800	1.250×10^{-3}
256	785	1.274×10^{-3}
264	770	1.299×10^{-3}
272	760	1.316×10^{-3}
280	750	1.333×10^{-3}
288	740	1.351×10^{-3}
304	720	1.389×10^{-3}
312	705	1.418×10^{-3}
320	668	1.497×10^{-3}
328	655	1.527×10^{-3}
336	646	1.548×10^{-3}
344	646	1.548×10^{-3}
360	640	1.563×10^{-3}
364	630	1.587×10^{-3}
368	619	1.616×10^{-3}
372	617	1.621×10^{-3}
376	610	1.639×10^{-3}
384	600	1.667×10^{-3}
400	583	1.715×10^{-3}
416	575	1.739×10^{-3}
432	554	1.805×10^{-3}
448	552	1.812×10^{-3}
480	526	1.901×10^{-3}
512	514	1.946×10^{-3}
544	483	2.070×10^{-3}
576	470	2.128×10^{-3}
608	458	2.183×10^{-3}
640	432	2.315×10^{-3}
704	405	2.469×10^{-3}
768	375	2.667×10^{-3}
800	360	2.778×10^{-3}
832	346	2.890×10^{-3}
864	335	2.985×10^{-3}
896	326	3.067×10^{-3}
960	305	3.279×10^{-3}
1024	288	3.472×10^{-3}
1088	270	3.704×10^{-3}
1152	256	3.906×10^{-3}
1216	244	4.098×10^{-3}
1280	236	4.237×10^{-3}
1408	217	4.608×10^{-3}
1536	203	4.926×10^{-3}
1664	190	5.263×10^{-3}
1792	178	5.618×10^{-3}
1920	167	5.988×10^{-3}
2048	158	6.329×10^{-3}
2304	142	7.042×10^{-3}
2432	135	7.407×10^{-3}

DELAY IN UNITS OF 2.72 MICROSECONDS	SPEED IN R.P.M.	1/(SPEED IN R.P.M.)
2560	130	7.692×10^{-3}
2816	119	8.403×10^{-3}
3072	108	9.259×10^{-3}
3328	100	1.000×10^{-2}
3584	91.6	1.092×10^{-2}
3840	84.3	1.186×10^{-2}
4096	77.8	1.285×10^{-2}
4352	72.8	1.374×10^{-2}
4608	67.8	1.475×10^{-2}
4864	64.1	1.560×10^{-2}
5120	60.9	1.642×10^{-2}
5376	57.8	1.730×10^{-2}
5632	55.4	1.805×10^{-2}
5888	52.1	1.919×10^{-2}
6144	49.7	2.012×10^{-2}
6400	48.0	2.083×10^{-2}
6656	46.2	2.165×10^{-2}
7168	42.8	2.336×10^{-2}
7680	40.6	2.463×10^{-2}
8192	38.2	2.618×10^{-2}
8704	35.8	2.793×10^{-2}
9216	34.2	2.924×10^{-2}
9728	32.1	3.115×10^{-2}
10240	30.9	3.236×10^{-2}
11264	28.2	3.546×10^{-2}
12288	25.8	3.876×10^{-2}
13312	23.7	4.219×10^{-2}
14336	23.1	4.329×10^{-2}
15360	20.5	4.878×10^{-2}
16384	19.3	5.181×10^{-2}
17408	17.6	5.682×10^{-2}

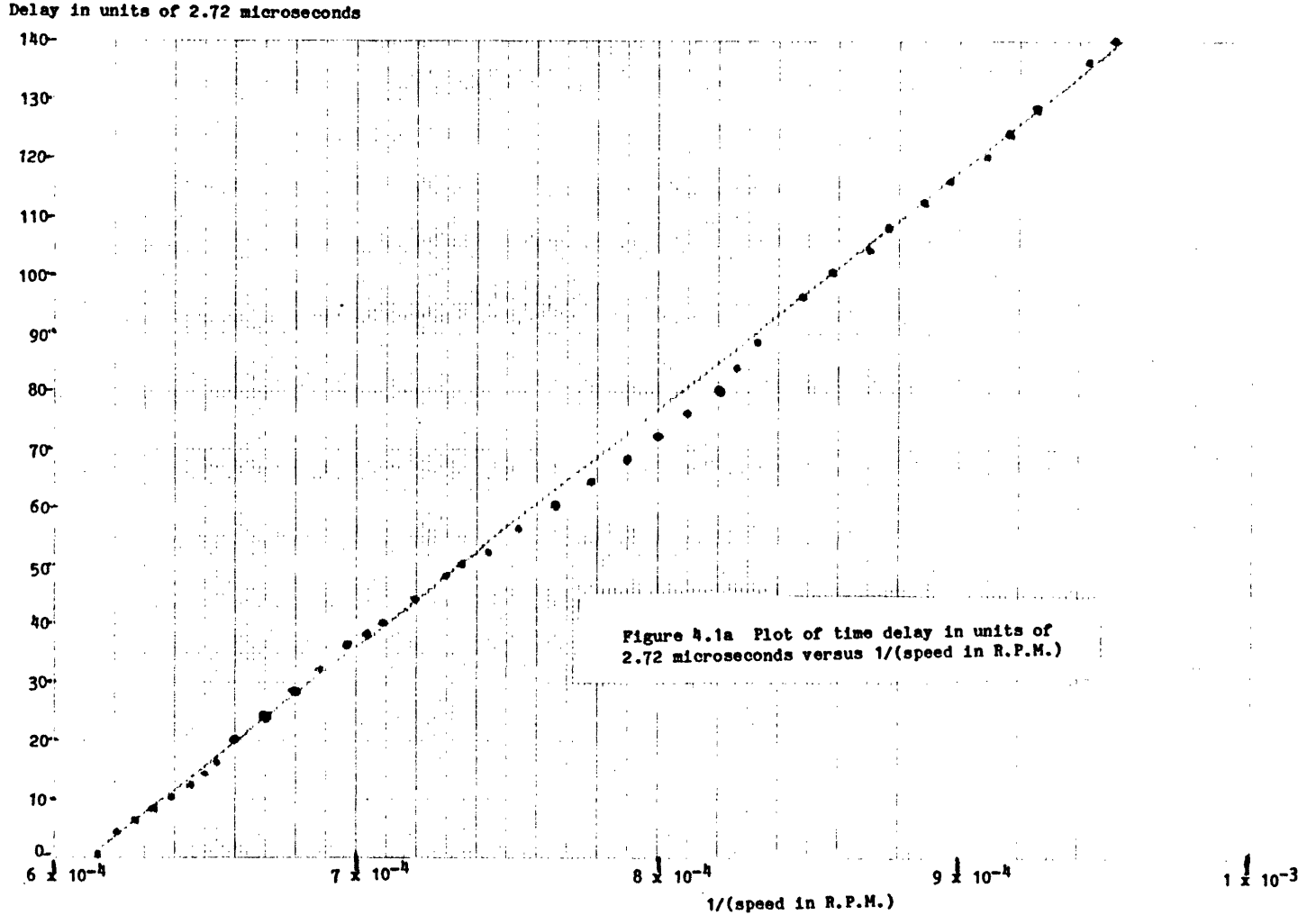


Figure 4.1a Plot of time delay in units of 2.72 microseconds versus 1/(speed in R.P.M.)

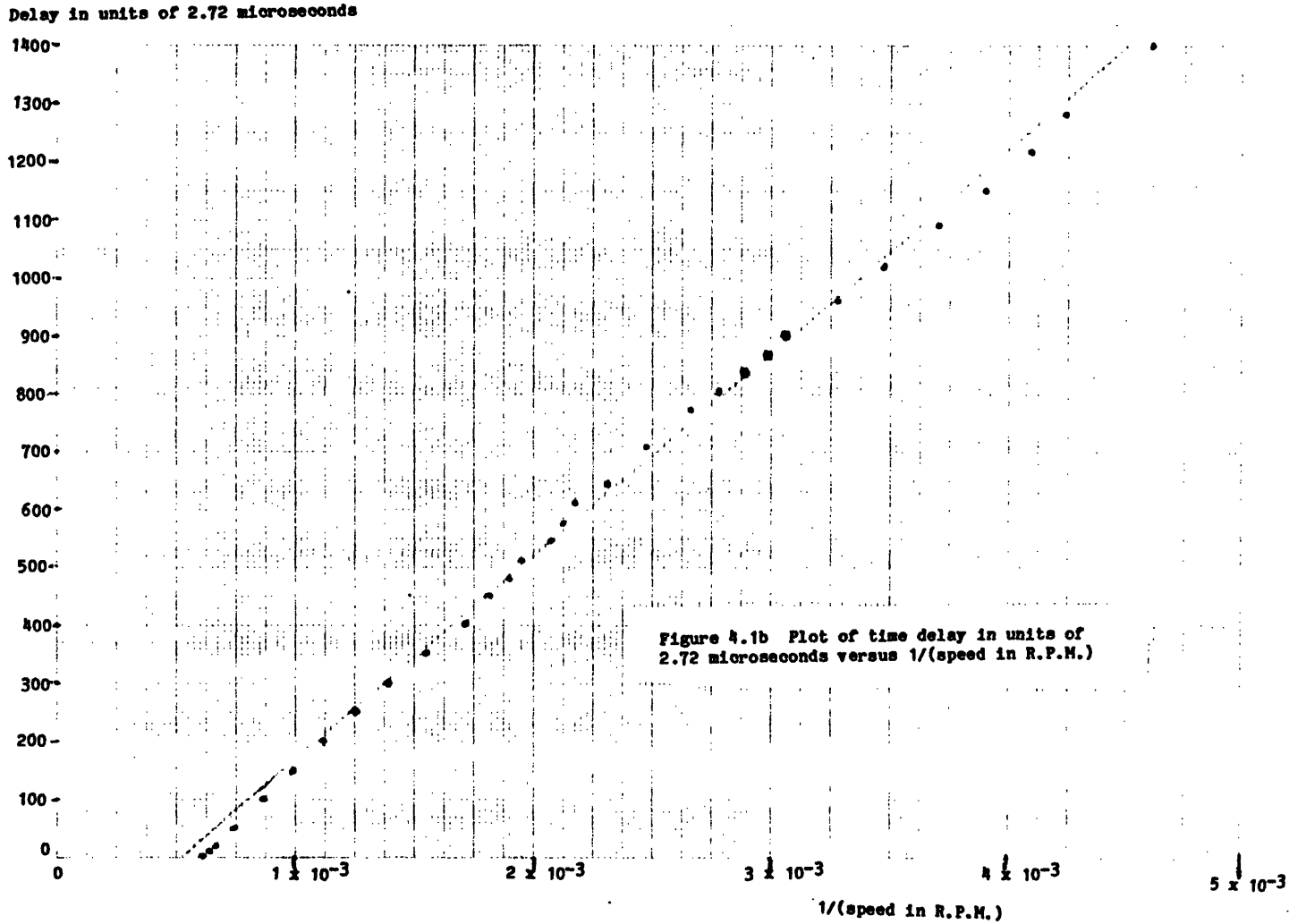


Figure 4.1b Plot of time delay in units of 2.72 microseconds versus 1/(speed in R.P.M.)

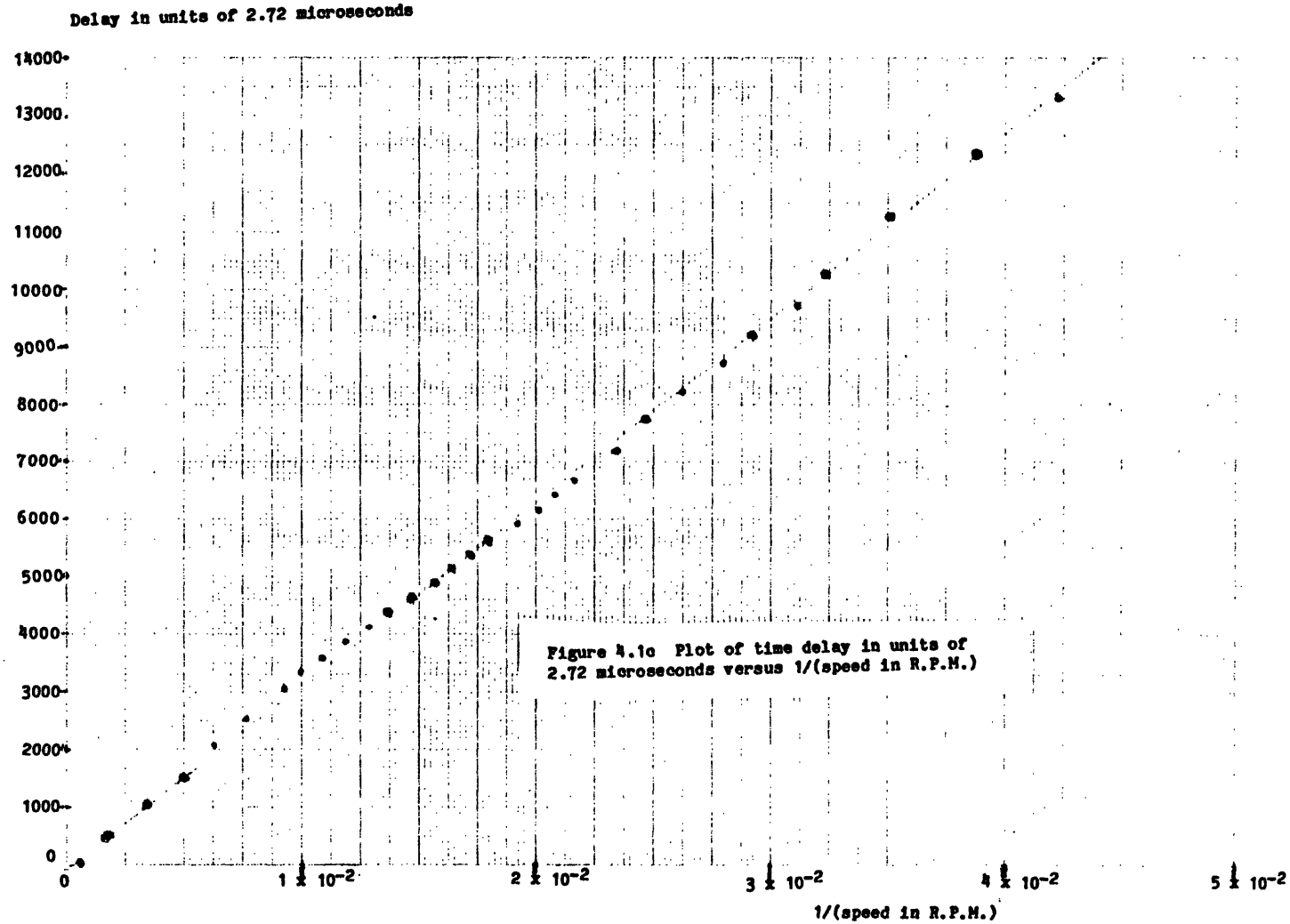


Figure 4.1c Plot of time delay in units of 2.72 microseconds versus 1/(speed in R.P.M.)

Table 4.2 The speed fluctuations around an average speed of 1565 R.P.M. observed for differing numbers of phases elapsing between every set of 4 consecutive phases used in speed correction in the program titled CONSTANT SPEED VIA VARYING CURRENT

ELAPSED PHASES	MINIMUM SPEED	MAXIMUM SPEED
1	1310	1910
2	1280	1910
4	1300	1880
8	1300	1840
16	1520	1610
20	1540	1590
24	1540	1590
28	1480	1550
32	1500	1630
64	1460	1640
128	1360	1720
256	1280	1720

Table 4.3 Three acceleration profiles showing the number of timer counts for each of the first 100 revolutions as the motor is accelerated from rest to full speed

REVOLUTION #	TRIAL #1	TRIAL #2	TRIAL #3
1	3943	3545	2936
2	1745	2040	4067
3	3031	1511	4011
4	4098	2057	3406
5	2969	3679	3529
6	3121	2849	2894
7	2800	3167	2359
8	2611	3026	2286
9	2315	2437	2093
10	1982	2168	1915
11	1970	2145	1793
12	1837	1962	1713
13	1733	2107	1711
14	1657	1857	1764
15	1562	1767	1510
16	1503	1582	1413
17	1450	1511	1384
18	1399	1448	1340
19	1351	1401	1387
20	1317	1362	1267
21	1264	1315	1233
22	1249	1279	1199
23	1214	1245	1196
24	1174	1219	1131
25	1170	1193	1127
26	1124	1173	1102
27	1114	1135	1087
28	1085	1109	1066
29	1068	1100	1065
30	1069	1071	1015
31	1035	1063	1016
32	1018	1045	1010
33	999	1089	982
34	1000	1012	979
35	980	1010	966
36	962	984	954
37	957	963	936
38	946	966	943
39	936	948	926
40	927	936	916
41	934	936	910
42	901	939	913
43	930	909	895
44	889	910	888
45	895	909	934
46	893	895	938
47	881	876	867
48	876	896	872
49	868	865	853

REVOLUTION #	TRIAL #1	TRIAL #2	TRIAL #3
50	859	876	857
51	871	881	852
52	862	869	857
53	859	857	830
54	858	853	856
55	844	868	831
56	846	857	845
57	849	854	836
58	952	856	833
59	824	858	833
60	831	836	877
61	830	857	891
62	830	833	825
63	832	858	877
64	870	855	822
65	873	900	820
66	823	847	819
67	815	900	818
68	819	831	818
69	819	910	819
70	819	878	811
71	821	840	820
72	816	822	868
73	817	829	814
74	819	831	814
75	814	830	808
76	815	821	822
77	817	821	812
78	867	822	868
79	814	819	815
80	815	878	815
81	813	823	813
82	813	824	813
83	819	877	813
84	862	824	813
85	818	825	815
86	812	823	922
87	870	879	817
88	815	827	812
89	872	879	816
90	815	828	925
91	819	871	815
92	814	826	830
93	814	827	823
94	816	822	814
95	815	820	822
96	869	820	830
97	872	820	882
98	815	818	817
99	815	819	876
100	821	987	879
AVERAGE LAST 20 REVOLUTIONS	829.0	842.0	837.4

	TRIAL #1	TRIAL #2	TRIAL #3
REV. TO ACHIEVE 1/2 FINAL SPEED	14	16	15
REV. TO ACHIEVE 9/10 FINAL SPEED	42	43	39

```
ASM48 :F1:ST2DAC.MAC DEBUG XREF TITLE('D TO A SECTION TEST')


ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    1
D TO A SECTION TEST

  LOC  OBJ          LINE        SOURCE STATEMENT

                      1         ; TESTDAC
                      2         ; THIS PROGRAM TESTS THE REFERENCE VOLTAGE LEVELS SENT TO THE
                      3         ; INVERTING INPUT TERMINALS OF THE 1 FIELD, 4 MAXIMUM(TOP), AND
                      4         ; 4 MINIMUM(MIDDLE) COMPARATORS AND TO THE NONINVERTING TERMINALS
                      5         ; OF THE 4 BOTTOM COMPARATORS BY THE 4 CORRESPONDING DIGITAL TO
                      6         ; ANALOG CONVERTERS.  WITH ALL POTS SET TO APPLY 4/5 OF THE OPAMP
                      7         ; OUTPUT VOLTAGE TO THE COMPARATOR INPUT TERMINALS, AN INCREASE OF
                      8         ; ONE IN THE 8 DIGIT NUMBER APPLIED TO THE DAC SHOULD RESULT IN A
                      9         ; 1.56 MV INCREMENT AT THE COMPARATOR INPUT TERMINAL.  IGNORING
                     10         ; OPAMP OFFSETS AND OTHER NONIDEALITIES, 00000000 WILL CORRESPOND
                     11         ; TO 0 MV AND 11111111 WILL CORRESPOND TO 400 MV AT THE COMPARATOR
                     12         ; INPUT TERMINAL OR ABOUT 5 AMPS OF CURRENT.
                     13         ; THE TOP SET OF DATA INPUT SWITCHES GOING TO P73 TO P40 WILL
                     14         ; INDICATE THE VALUE TO BE APPLIED TO THE DAC WITH P73 AS THE MOST
                     15         ; SIGNIFICANT BIT AND P40 AS THE LEAST SIGNIFICANT BIT.
                     16         ; P53 = 1 ENABLES THE LATCH ADDRESSING THE 4 HIGH BITS OF THE
                     17         ; FIELD DAC,
                     18         ; P52 = 1 ENABLES THE LATCH ADDRESSING THE 4 LOW BITS OF THE
                     19         ; FIELD DAC,
                     20         ; P51 = 1 ENABLES THE LATCH ADDRESSING THE 4 HIGH BITS OF THE
                     21         ; MAXIMUM(TOP) DAC,
                     22         ; P50 = 1 ENABLES THE LATCH ADDRESSING THE 4 LOW BITS OF THE
                     23         ; MAXIMUM(TOP) DAC,
                     24         ; P43 = 1 ENABLES THE LATCH ADDRESSING THE 4 HIGH BITS OF THE
                     25         ; MINIMUM(MIDDLE) DAC,
                     26         ; P42 = 1 ENABLES THE LATCH ADDRESSING THE 4 LOW BITS OF THE
                     27         ; MINIMUM(MIDDLE) DAC,
                     28         ; P41 = 1 ENABLES THE LATCH ADDRESSING THE 4 HIGH BITS OF THE
                     29         ; BOTTOM DAC,
                     30         ; P40 = 1 ENABLES THE LATCH ADDRESSING THE 4 LOW BITS OF THE
                     31         ; BOTTOM DAC,
                     32
0000                 33         ORG 0
                     34 SYSRST: ; SYSTEM RESET
0000 0409            35         JMP RESET
                     36
0003                 37         ORG 3
                     38 EXTINT: ; EXTERNAL INTERRUPT
0003 15              39         DIS I
0004 93              40         RETR
                     41
0007                 42         ORG 7
                     43 TIMINT: ; TIMER INTERRUPT
0007 35              44         DIS TCNTI
0008 93              45         RETR
                     46
0009 2380            47 RESET:  MOV A,#80H
0008 3A              48         OUTL P2,A ; DISABLE THE MAIN PROGRAM IO EXPANDER
000C 39              49 HERE:   OUTL P1,A ; TURN OFF THE TRANSISTORS, DISABLE THE LED/DAC IO
                     50                   ; EXPANDER, AND ENABLE THE INPUT SWITCH IO EXPANDER
000D 0F              51         MOVD A,P7
000E AC              52         MOV R4,A
000F 0E              53         MOVD A,P6
0010 AD              54         MOV R5,A
```

```
ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    2
D TO A SECTION TEST

  LOC  OBJ          LINE        SOURCE STATEMENT

  0011 0D             55         MOVD A,P5
  0012 AE             56         MOV R6,A
  0013 0C             57         MOVD A,P4
  0014 AF             58         MOV R7,A
  0015 2340           59         MOV A,#40H ; DISABLE THE INPUT SWITCH IO EXPANDER
  0017 39             60         OUTL P1,A  ; AND ENABLE THE LED/DAC IO EXPANDER
  0018 FC             61         MOV A,R4
  0019 3F             62         MOVD P7,A
  001A FD             63         MOV A,R5
  001B 3E             64         MOVD P6,A
  001C FE             65         MOV A,R6
  001D 3D             66         MOVD P5,A
  001E FF             67         MOV A,R7
  001F 3C             68         MOVD P4,A
  0020 2380           69         MOV A,#80H ; DISABLE THE LED/DAC IO EXPANDER
  0022 040C           70         JMP HERE   ; AND ENABLE THE INPUT SWITCH IO EXPANDER
                      71         END

USER SYMBOLS



EXTINT 0003    HERE   000C    RESET  0009    SYSRST 0000    TIMINT 0007

ASSEMBLY COMPLETE,   NO ERRORS
```

```
ASM48 :F1:ST210N.MAC DEBUG XREF TITLE('TRANSISTOR SWITCHING TEST')


ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    1
TRANSISTOR SWITCHING TEST

 LOC  OBJ          LINE        SOURCE STATEMENT

                      1         ; TRANSISTOR SWITCHING TEST
                      2         ; THIS PROGRAM TESTS TRANSISTOR SWITCHING WITH CURRENT REGULATION
                      3         ; FROM THE COMPARATORS.  THAT IS, THE PROHIBIT OUTPUT IS ZERO.
                      4         ; FIRST THE APPROPRIATE VALUES ARE PLACED ON THE LATCHES ADDRESSING
                      5         ; THE DACS.
                      6         ; 1.) ON P73 TO P60, THE TOP SET OF DATA INPUT SWITCHES, INDICATE
                      7         ; THE VALUE TO BE APPLIED TO THE FIELD DAC WITH P73 AS THE MSB
                      8         ; AND P60 AS THE LSB.  LIKEWISE, ON P53 TO P40, THE BOTTOM SET
                      9         ; OF DATA INPUT SWITCHES, INDICATE THE VALUE TO BE APPLIED TO
                     10         ; THE DAC ADDRESSING THE MAXIMUM VALUE(TOP) COMPARATORS.
                     11         ; PRESS T0.
                     12         ; 2.) ON P73 TO P60 INDICATE THE VALUE TO BE APPLIED TO THE DAC
                     13         ; ADDRESSING THE MINIMUM VALUE(MIDDLE) COMPARATORS, ON P53 TO
                     14         ; P40 INDICATE THE VALUE TO BE APPLIED TO THE DAC ADDRESSING THE
                     15         ; BOTTOM COMPARATORS.
                     16         ; PRESS T1.
                     17         ; 3.) THEN, SELECT THE TRANSISTOR BANK TO BE TESTED.
                     18         ; P73 = 1 TESTS PHASE A.
                     19         ; P72 = 1 TESTS PHASE B.
                     20         ; P71 = 1 TESTS PHASE C.
                     21         ; P70 = 1 TESTS PHASE D.
                     22         ; PHASE CURRENTS ARE SWITCHED BETWEEN THE SELECTED MINIMUM AND
                     23         ; MAXIMUM VALUES.
                     24         ; IF NONE OF THE ABOVE ARE CHOSEN, THE FIELD IS TESTED.  AFTER
                     25         ; REACHING THE SELECTED FIELD VALUE, THE FIELD TRANSISTOR IS TURNED
                     26         ; OFF FOR ABOUT 44.6 MILLISECONDS.
                     27         ; PRESS T0
                     28         ; 4.) TO SELECT A NEW TRANSISTOR BANK MERELY CHANGE THE

                     29         ; VALUE OF THE P7 SWITCHES.
                     30
0000                 31         ORG 0
                     32 SYSRST: ; SYSTEM RESET
0000 0409            33         JMP RESET
                     34
0003                 35         ORG 3
                     36 EXTINT: ; EXTERNAL INTERRUPT
0003 15              37         DIS I
0004 93              38         RETR
                     39
0007                 40         ORG 7
                     41 TIMINT: ; TIMER INTERRUPT
0007 35              42         DIS TCNTI
0008 93              43         RETR
                     44
0009 2380            45 RESET:  MOV A,#80H
000B 3A              46         OUTL P2,A ; DISABLE THE MAIN IO EXPANDER
000C 39              47         OUTL P1,A ; TURN OFF THE TRANSISTORS, DISABLE THE LED/DAC IO
                     48                   ; EXPANDER, AND ENABLE THE INPUT SWITCH IO EXPANDER.
000D 260D            49 MAIN:   JNT0 MAIN
000F 360F            50 HERE1:  JT0 HERE1 ; WAIT FOR T0 TO BE PRESSED
0011 0F              51         MOVD A,P7 ; MOVE IN THE 4 HIGH BITS
0012 AC              52         MOV R4,A  ; OF THE FIELD DAC VALUE
0013 0E              53         MOVD A,P6 ; MOVE IN THE 4 LOW BITS
0014 AD              54         MOV R5,A  ; OF THE FIELD DAC VALUE
```

```
LOC  OBJ        LINE        SOURCE STATEMENT

0015 0D           55          MOVD A,P5 ; MOVE IN THE 4 HIGH BITS
0016 AE           56          MOV R6,A  ; OF THE MAXIMUM DAC VALUE
0017 0C           57          MOVD A,P4 ; MOVE IN THE 4 LOW BITS
0018 AF           58          MOV R7,A  ; OF THE MAXIMUM DAC VALUE
0019 4619         59 HERE2:   JNT1 HERE2
001B 561B         60 HERE3:   JT1 HERE3 ; WAIT FOR T1 TO BE PRESSED
001D 0F           61          MOVD A,P7 ; MOVE IN THE 4 HIGH BITS
001E A8           62          MOV R0,A  ; OF THE MINIMUM DAC VALUE
001F 0E           63          MOVD A,P6 ; MOVE IN THE 4 LOW BITS
0020 A9           64          MOV R1,A  ; OF THE MINIMUM DAC VALUE
0021 0D           65          MOVD A,P5 ; MOVE IN THE 4 HIGH BITS
0022 AA           66          MOV R2,A  ; OF THE BOTTOM DAC VALUE
0023 0C           67          MOVD A,P4 ; MOVE IN THE 4 LOW BITS
0024 AB           68          MOV R3,A  ; OF THE BOTTOM DAC VALUE
0025 2340         69          MOV A,#40H ; DISABLE THE INPUT SWITCH IO EXPANDER
0027 39           70          OUTL P1,A  ; AND ENABLE THE LED/DAC IO EXPANDER
0028 230C         71          MOV A,#0CH ; DISABLE THE 2 MAXIMUM DAC LATCHES
002A 3D           72          MOVD P5,A  ; AND ENABLE THE 2 FIELD DAC LATCHES
002B 2300         73          MOV A,#00H ; DISABLE THE 2 MINIMUM AND 2 BOTTOM LATCHES
002D 3C           74          MOVD P4,A
002E FC           75          MOV A,R4 ; MOVE THE 4 HIGH BITS
002F 3F           76          MOVD P7,A ; OUT TO THE FIELD DAC
0030 FD           77          MOV A,R5 ; MOVE THE 4 LOW BITS
0031 3E           78          MOVD P6,A ; OUT TO THE FIELD DAC
0032 2303         79          MOV A,#03H ; DISABLE THE 2 FIELD DAC LATCHES
0034 3D           80          MOVD P5,A  ; AND ENABLE THE 2 MAXIMUM DAC LATCHES
0035 FE           81          MOV A,R6 ; MOVE THE 4 HIGH BITS
0036 3F           82          MOVD P7,A ; OUT TO THE MAXIMUM VALUE DAC
0037 FF           83          MOV A,R7  ; MOVE THE 4 LOW BITS
0038 3E           84          MOVD P6,A ; OUT TO THE MAXIMUM VALUE DAC
0039 2300         85          MOV A,#00H ; DISABLE THE 2 MAXIMUM VALUE DAC LATCHES
003B 3D           86          MOVD P5,A
003C 230C         87          MOV A,#0CH ; ENABLE THE 2 MINIMUM VALUE DAC LATCHES
003E 3C           88          MOVD P4,A
003F F8           89          MOV A,R0  ; MOVE THE 4 HIGH BITS
0040 3F           90          MOVD P7,A ; OUT TO THE MINIMUM VALUE DAC
0041 F9           91          MOV A,R1  ; MOVE THE 4 LOW BITS
0042 3E           92          MOVD P6,A ; OUT TO THE MINIMUM VALUE DAC
0043 2303         93          MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES



0045 3C           94          MOVD P4,A  ; AND ENABLE THE 2 BOTTOM DAC LATCHES
0046 FA           95          MOV A,R2 ; MOVE THE 4 HIGH BITS
0047 3F           96          MOVD P7,A ; OUT TO THE BOTTOM DAC
0048 FB           97          MOV A,R3  ; MOVE THE 4 LOW BITS
0049 3E           98          MOVD P6,A ; OUT TO THE BOTTOM DAC
004A 2300         99          MOV A,#00H ; DISABLE THE 2 BOTTOM DAC LATCHES
004C 3C          100          MOVD P4,A
004D 2640        101 HERE4:   JNT0 HERE4
004F 364F        102 HERE5:   JT0 HERE5 ; WAIT FOR T0 TO BE PRESSED
0051 2380        103 MOV A,#80H ; DISABLE THE LED/DAC IO EXPANDER
0053 39          104 OUTL P1,A ; AND ENABLE THE INPUT SWITCH IO EXPANDER
0054 0F          105 SELECT: MOVD A,P7
0055 727D        106          JB3 AONLY
0057 5282        107          JB2 BONLY
0059 3287        108          JB1 CONLY
005B 128C        109          JB0 DONLY
```

```
 LOC  OBJ        LINE       SOURCE STATEMENT

                  110
 005D 2300        111 FONLY:  MOV A,#00H : ENABLE THE MAIN PROGRAM TO EXPANDER
 005F 3A          112         OUTL P2,A
 0060 2380        113         MOV A,#080H : TURN ON THE FIELD TRANSISTOR
 0062 39          114         OUTL P1,A ; AND DISABLE THE INITIAL VALUE TO EXPANDER
 0063 0E          115 FSENSE: MOVD A,P6 : CHECK FIELD CURRENT LEVEL
 0064 A9          116         MOV R1,A
 0065 0E          117         MOVD A,P6 : CONFIRM WITH A DELAYED CHECK
 0066 59          118         ANL A,R1
 0067 1248        119         JB0 FOFF
 0069 0463        120         JMP FSENSE
 006B 2380        121 FOFF:   MOV A,#80H : TURN OFF THE FIELD TRANSISTOR
 006D 39          122         OUTL P1,A ; AND ENABLE THE INITIAL VALUE TO EXPANDER
 006E 3A          123         OUTL P2,A ; DISABLE THE MAIN PROGRAM TO EXPANDER
 006F 27          124         CLR A : WAIT FOR ABOUT 44,6 MILLISECONDS
 0070 42          125         MOV T,A
 0071 55          126         STRT T
 0072 B904        127         MOV R1,#4 : NUMBER OF TIMER OVERFLOWS BEFORE THE FIELD IS TURNED
                  128                   : ON AGAIN
 0074 1678        129 WAIT:   JTF MORE
 0074 0474        130         JMP WAIT
 0078 E974        131 MORE:   DJNZ R1,WAIT
 007A 65          132         STOP TCNT
 007B 0454        133         JMP SELECT
                  134
 007D 2381        135 AONLY:  MOV A,#81H
 007F 39          136         OUTL P1,A ; TURN ON PHASE A
 0080 0454        137         JMP SELECT
                  138
 0082 2382        139 BONLY:  MOV A,#82H
 0084 39          140         OUTL P1,A ; TURN ON PHASE B
 0085 0454        141         JMP SELECT
                  142
 0087 2384        143 CONLY:  MOV A,#84H
 0089 39          144         OUTL P1,A ; TURN ON PHASE C
 008A 0454        145         JMP SELECT
                  146
 008C 2388        147 DONLY:  MOV A,#88H
 008E 39          148         OUTL P1,A ; TURN ON PHASE D
 008F 0454        149         JMP SELECT
                  150
                  151         END

USER SYMBOLS
AONLY  007D   BONLY  0082   CONLY  0087   DONLY  008C   EXTINT 0003   FOFF   006B   FONLY  005D   FSENSE 0063
HERE1  000F   HERE2  0019   HERE3  0018   HERE4  004D   HERE5  004F   MAIN   000D   MORE   0078   RESET  0009
SELECT 0054   SYSRST 0000   TIMINT 0007   WAIT   0074

ASSEMBLY COMPLETE,  NO ERRORS
```

```
ASM48 :F1:ST2RUN.MAC DEBUG XREF TITLE('RUN THE MOTOR')


ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    1
RUN THE MOTOR

  LOC  OBJ          LINE        SOURCE STATEMENT

                       1        ; PROGRAM TO RUN THE MOTOR
                       2        ; INITIALLY THE FIELD TRANSISTOR IS TURNED ON AND THE FIELD CURRENT
                       3        ; IS ALLOWED TO RISE UNTIL IT REACHES A LEVEL DETERMINED BY THE
                       4        ; FIELD DAC VALUE FED IN ON THE INPUT DATA SWITCHES.  A RESTART
                       5        ; WILL ALSO SUBJECT FIELD CURRENT TO THE SAME REGULATION.  PHASE
                       6        ; CURRENT WILL BE KEPT BETWEEN MAXIMUM AND MINIMUM LEVELS DETERMINED
                       7        ; BY THE MAXIMUM AND MINIMUM DAC VALUES FED IN ON THE INPUT DATA
                       8        ; SWITCHES.  PRESSING THE T0 BUTTON WILL START THE MOTOR AND PRESSING
                       9        ; THE T1 BUTTON WILL REPLACE THE ORIGINAL MAXIMUM AND MINIMUM PHASE
                      10        ; LEVELS WITH A SECOND SET OF MAXIMUM AND MINIMUM PHASE LEVELS.
                      11        ; BECAUSE THE MAXIMUM CURRENT REQUIREMENT OF THE MOTOR OCCURS AT
                      12        ; STARTUP, IT IS USUALLY DESIRABLE TO DECREASE PHASE CURRENT ONCE
                      13        ; STEADY STATE MOTION HAS BEEN ACHIEVED.  EXTERNAL INTERRUPTS TO
                      14        ; INDICATE DETECTION OF THE NEXT PHASE WILL BE TRIGGERED BY ZERO
                      15        ; CROSSINGS OF THE SENSE WAVEFORMS GOING FROM NEGATIVE TO POSITIVE
                      16        ; POLARITY.
                      17        ; TO RUN THE MOTOR:
                      18        ; 1.) ON P73 TO P60, THE TOP SET OF DATA INPUT SWITCHES, INDICATE
                      19        ; THE FIELD DAC VALUE WITH P73 AS THE MSB AND P60 AS THE LSB.  ON P53
                      20        ; TO P40, THE BOTTOM SET OF DATA INPUT SWITCHES, INDICATE THE FIRST
                      21        ; VALUE TO BE APPLIED TO THE DAC ADDRESSING THE MAXIMUM VALUE(TOP)
                      22        ; COMPARATORS WITH P53 AS THE MSB AND P40 AS THE LSB.
                      23        ; PRESS T0.
                      24        ; 2.) ON P73 TO P60 INDICATE THE FIRST VALUE TO BE APPLIED TO THE
                      25        ; DAC ADDRESSING THE MINIMUM VALUE(MIDDLE) COMPARATORS.  ON P53 TO
                      26        ; P40 INDICATE THE FIRST VALUE TO BE APPLIED TO THE DAC ADDRESSING
                      27        ; THE BOTTOM COMPARATORS.
                      28        ; PRESS T1.
                      29        ; 3.) ON P53 TO P40 INDICATE THE SECOND MAXIMUM VALUE.
                      30        ; PRESS T0.
                      31        ; 4.) ON P73 TO P60 INDICATE THE SECOND MINIMUM VALUE.  ON P53 TO
                      32        ; P40 INDICATE THE SECOND BOTTOM VALUE.
                      33        ; PRESS T1.
                      34        ; 5.) PRESS T0 TO START THE MOTOR WITH THE FIRST SET OF VALUES.
                      35        ; 6.) PRESS T1 TO REPLACE THE FIRST SET OF VALUES WITH THE SECOND
                      36        ; SET.
                      37
  0000                38                ORG 0
                      39 SYSRST:        ; SYSTEM RESET
  0000 040B           40                JMP RESET
                      41
  0003                42                ORG 3
                      43 EXTINT:        ; EXTERNAL INTERRUPT
  0003 FB             44                MOV A,R3 ; JUMP TO THE EXTERNAL INTERRUPT ROUTINE WHOSE LOCATION
  0004 B3             45                JMPP @A  ; IS STORED AT THE ADDRESS IN R3
                      46
  0007                47                ORG 7
                      48 TIMINT:        ; TIMER INTERRUPT
  0007 35             49                DIS TCNTI
  0008 93             50                RETR
                      51
  0009 8F             52 DATA1:         DB LOW SERVE1
  000A A6             53 DATA2:         DB LOW SERVE2
                      54
```

RUN THE MOTOR

```
LOC  OBJ       LINE     SOURCE STATEMENT


000B 2380        55 RESET:  MOV A,#80H ; TURN OFF THE TRANSISTORS, ENABLE THE INPUT SWITCH
000D 39          56         OUTL P1,A  ; SWITCH IO EXPANDER, AND DISABLE THE LED/DAC TO EXPANDER
000E 3A          57         OUTL P2,A  ; DISABLE THE MAIN PROGRAM TO EXPANDER
000F 260F        58 HERE1:  JNT0 HERE1
0011 3611        59 HERE2:  JT0 HERE2 ; WAIT FOR T0 TO BE PRESSED
0013 B820        60         MOV R0,#20H
0015 0F          61         MOVD A,P7  ; MOVE THE 4 HIGH BITS OF THE
0016 A0          62         MOV @R0,A  ; FIELD VALUE TO MEMORY LOCATION 20H
0017 18          63         INC R0
0018 0E          64         MOVD A,P6 ; MOVE THE 4 LOW BITS OF THE
0019 A0          65         MOV @R0,A ; FIELD VALUE VALUE TO DATA MEMORY LOCATION 21H
001A 18          66         INC R0
001B 0D          67         MOVD A,P5 ; MOVE THE 4 HIGH BITS OF THE FIRST
001C A0          68         MOV @R0,A ; MAXIMUM VALUE TO DATA MEMORY LOCATION 22H
001D 18          69         INC R0
001E 0C          70         MOVD A,P4 : MOVE THE 4 LOW BITS OF THE FIRST
001F A0          71         MOV @R0,A ; MAXIMUM VALUE TO DATA MEMORY LOCATION 23H
0020 4A20        72 HERE3:  JNT1 HERE3
0022 5622        73 HERE4:  JT1 HERE4 ; WAIT FOR T1 TO BE PRESSED
0024 18          74         INC R0
0025 0F          75         MOVD A,P7 ; MOVE THE 4 HIGH BITS OF THE FIRST
0026 A0          76         MOV @R0,A ; MINIMUM VALUE TO DATA MEMORY LOCATION 24H
0027 18          77         INC R0
0028 0E          78         MOVD A,P6 ; MOVE THE 4 LOW BITS OF THE FIRST
0029 A0          79         MOV @R0,A ; MINIMUM VALUE TO DATA MEMORY LOCATION 25H
002A 18          80         INC R0
002B 0D          81         MOVD A,P5 ; MOVE THE 4 HIGH BITS OF THE FIRST
002C A0          82         MOV @R0,A ; BOTTOM VALUE TO DATA MEMORY LOCATION 26H
002D 18          83         INC R0
002E 0C          84         MOVD A,P4 ; MOVE THE 4 LOW BITS OF THE FIRST
002F A0          85         MOV @R0,A ; BOTTOM VALUE TO DATA MEMORY LOCATION 27H
0030 2630        86 HERE5:  JNT0 HERE5
0032 3632        87 HERE6:  JT0 HERE6 ; WAIT FOR T0 TO BE PRESSED
0034 18          88         INC R0
0035 0D          89         MOVD A,P5 ; MOVE THE 4 HIGH BITS OF THE SECOND
0036 A0          90         MOV @R0,A ; MAXIMUM VALUE TO DATA MEMORY LOCATION 28H
0037 18          91         INC R0
0038 0C          92         MOVD A,P4 ; MOVE THE 4 LOW BITS OF THE SECOND
0039 A0          93         MOV @R0,A ; MAXIMUM VALUE TO DATA MEMORY LOCATION 29H
003A 463A        94 HERE7:  JNT1 HERE7
003C 563C        95 HERE8:  JT1 HERE8 ; WAIT FOR T1 TO BE PRESSED
003E 18          96         INC R0
003F 0F          97         MOVD A,P7 ; MOVE THE 4 HIGH BITS OF THE SECOND
0040 A0          98         MOV @R0,A ; MINIMUM VALUE TO DATA MEMORY LOCATION 2AH
0041 18          99         INC R0
0042 0E         100         MOVD A,P6 ; MOVE THE 4 LOW BITS OF THE SECOND
0043 A0         101         MOV @R0,A ; MINIMUM VALUE TO DATA MEMORY LOCATION 2BH
0044 18         102         INC R0
0045 0D         103         MOVD A,P5 ; MOVE THE 4 HIGH BITS OF THE SECOND
0046 A0         104         MOV @R0,A ; BOTTOM VALUE TO DATA MEMORY LOCATION 2CH
0047 18         105         INC R0
0048 0C         106         MOVD A,P4 ; MOVE THE 4 LOW BITS OF THE SECOND
0049 A0         107         MOV @R0,A ; BOTTOM VALUE TO DATA MEMORY LOCATION 2DH
004A 264A       108 HERE9:  JNT0 HERE9
004C 364C       109 HERE10: JT0 HERE10 ; WAIT FOR TO BEFORE THE MOTOR STARTS
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

004E BCF6        110 START2: MOV R4,#-10 ; ALLOW ONLY 10 ATTEMPTS TO START THE MOTOR
0050 2340        111         MOV A,#40H ; DISABLE THE INPUT SWITCH IO EXPANDER
0052 39          112         OUTL P1,A ; AND ENABLE THE LED/DAC IO EXPANDER
0053 230C        113         MOV A,#0CH ; DISABLE THE 2 MAXIMUM DAC LATCHES
0055 3D          114         MOVD P5,A ; AND ENABLE THE 2 FIELD DAC LATCHES
0056 27          115         CLR A ; DISABLE THE 2 MINIMUM
0057 3C          116         MOVD P4,A ; AND 2 BOTTOM DAC LATCHES
0058 B820        117         MOV R0,#20H
005A F0          118         MOV A,@R0 ; MOVE THE 4 HIGH BITS

005B 3F          119         MOVD P7,A ; OUT TO THE FIELD DAC
005C 18          120         INC R0
005D F0          121         MOV A,@R0 ; MOVE THE 4 LOW BITS
005E 3E          122         MOVD P6,A ; OUT TO THE FIELD DAC
005F 2303        123         MOV A,#03H ; DISABLE THE 2 FIELD DAC LATCHES
0061 3D          124         MOVD P5,A  ; AND ENABLE THE 2 MAXIMUM DAC LATCHES
0062 18          125         INC R0
0063 F0          126         MOV A,@R0 ; MOVE THE FIRST 4 HIGH BITS
0064 3F          127         MOVD P7,A ; OUT TO THE MAXIMUM VALUE DAC
0065 18          128         INC R0
0066 F0          129         MOV A,@R0 ; MOVE THE FIRST 4 LOW BITS
0067 3E          130         MOVD P6,A ; OUT TO THE MAXIMUM VALUE DAC
0068 27          131         CLR A ; DISABLE THE 2 MAXIMUM DAC LATCHES
0069 3D          132         MOVD P5,A
006A 230C        133         MOV A,#0CH ; ENABLE THE 2 MINIMUM DAC LATCHES
006C 3C          134         MOVD P4,A
006D 18          135         INC R0
006E F0          136         MOV A,@R0 ; MOVE THE FIRST 4 HIGH BITS
006F 3F          137         MOVD P7,A ; OUT TO THE MINIMUM VALUE DAC
0070 18          138         INC R0
0071 F0          139         MOV A,@R0 ; MOVE THE FIRST 4 LOW BITS
0072 3F          140         MOVD P6,A ; OUT TO THE MINIMUM VALUE DAC
0073 2303        141         MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES
0075 3C          142         MOVD P4,A  ; AND ENABLE THE 2 BOTTOM DAC LATCHES
0076 18          143         INC R0
0077 F0          144         MOV A,@R0 ; MOVE THE FIRST 4 HIGH BITS
0078 3F          145         MOVD P7,A ; OUT TO THE BOTTOM VALUE DAC
0079 18          146         INC R0
007A F0          147         MOV A,@R0 ; MOVE THE FIRST 4 LOW BITS
007B 3E          148         MOVD P6,A ; OUT TO THE BOTTOM VALUE DAC
007C 27          149         CLR A ; DISABLE THE 2 BOTTOM DAC LATCHES
007D 3C          150         MOVD P4,A
007E 23C0        151         MOV A,#0C0H ; DISABLE THE LED/DAC IO EXPANDER
0080 39          152         OUTL P1,A
0081 27          153         CLR A ; ENABLE THE MAIN PROGRAM IO EXPANDER
0082 3A          154         OUTL P2,A
0083 04E7        155         JMP MREST
                 156
                 157         ; SUBROUTINES
0085 23FF        158 WAIT1T: MOV A,#-1 ; WAIT1T WAITS FOR 1 TIMER UNIT
0087 62          159 WAITTM: MOV T,A ; WAITTM WAITS FOR # OF UNITS OF TIME
0088 55          160         STRT T  ; EQUAL TO THAT IN A
0089 1680        161 WAITTF: JTF WTMR ; WAITTF WAITS FOR TIMER FLAG
008B 0489        162         JMP WAITTF
008D 65          163 WTMR:   STOP TCNT
008E 93          164         RETR
```

```
LOC  OBJ        LINE        SOURCE STATEMENT

                  165
                  166       ; EXTERNAL INTERRUPT ROUTINES
008F FE           167 SERVE1: MOV A,R6
0090 43E0         168       ORL A,#0E0H
0092 39           169       OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
0093 990F         170       ANL P1,#00FH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                  171
0095 27           172       CLR A
0096 62           173       MOV T,A
0097 55           174       STRT T ; START THE TIMER FROM 0
0098 BD06         175       MOV R5,#6 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                  176
009A FE           177       MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
009B AF           178       MOV R7,A
009C 47           179       SWAP A
009D 4E           180       ORL A,R6
009E E7           181       RL A
009F 530F         182       ANL A,#0FH
00A1 AE           183       MOV R6,A



                  184
00A2 3F           185       MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                  186
00A3 BCFA         187       MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                  188
00A5 93           189       RETR
                  190
00A6 FE           191 SERVE2: MOV A,R6
00A7 43E0         192       ORL A,#0E0H
00A9 39           193       OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
00AA 990F         194       ANL P1,#00FH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                  195
00AC 27           196       CLR A
00AD 62           197       MOV T,A
00AE 55           198       STRT T ; START THE TIMER FROM 0
00AF BD06         199       MOV R5,#6 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                  200
00B1 FE           201       MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
00B2 AF           202       MOV R7,A
00B3 47           203       SWAP A
00B4 4E           204       ORL A,R6
00B5 E7           205       RL A
00B6 530F         206       ANL A,#0FH
00B8 AE           207       MOV R6,A
                  208
00B9 3F           209       MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                  210
00BA 8A80         211       ORL P2,#80H ; DISABLE THE MAIN PROGRAM TO EXPANDER
00BC 997F         212       ANL P1,#7FH ; ENABLE THE LED/DAC TO EXPANDER
00BE 2303         213       MOV A,#03H ; DISABLE THE FIELD DAC LATCHES
00C0 3D           214       MOVD P5,A  ; AND ENABLE THE MAXIMUM DAC LATCHES
00C1 27           215       CLR A ; DISABLE THE MINIMUM AND BOTTOM DAC LATCHES
00C2 3C           216       MOVD P4,A
00C3 B828         217       MOV R0,#28H
00C5 F0           218       MOV A,@R0 ; MOVE THE SECOND 4 HIGH BITS
00C6 3F           219       MOVD P7,A ; OUT TO THE MAXIMUM VALUE DAC
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

00C7 18          220        INC R0
00C8 F0          221        MOV A,@R0 ; MOVE THE SECOND 4 LOW BITS
00C9 3E          222        MOVD P6,A ; OUT TO THE MAXIMUM VALUE DAC
00CA 27          223        CLR A ; DISABLE THE MAXIMUM DAC LATCHES
00CB 3D          224        MOVD P5,A
00CC 230C        225        MOV A,#0CH ; ENABLE THE MINIMUM DAC LATCHES
00CE 3C          226        MOVD P4,A
00CF 18          227        INC R0
00D0 F0          228        MOV A,@R0 ; MOVE THE SECOND 4 HIGH BITS
00D1 3F          229        MOVD P7,A ; OUT TO THE MINIMUM VALUE DAC
00D2 18          230        INC R0
00D3 F0          231        MOV A,@R0 ; MOVE THE SECOND 4 LOW BITS
00D4 3E          232        MOVD P6,A ; OUT TO THE MINIMUM VALUE DAC
00D5 2303        233        MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES
00D7 3C          234        MOVD P4,A ; AND ENABLE THE 2 BOTTOM DAC LATCHES
00D8 18          235        INC R0
00D9 F0          236        MOV A,@R0 ; MOVE THE SECOND 4 HIGH BITS
00DA 3F          237        MOVD P7,A ; OUT TO THE BOTTOM DAC
00DB 18          238        INC R0
00DC F0          239        MOV A,@R0 ; MOVE THE SECOND 4 LOW BITS
00DD 3E          240        MOVD P6,A ; OUT TO THE BOTTOM DAC
00DE 27          241        CLR A ; DISABLE THE 2 BOTTOM DAC LATCHES
00DF 3C          242        MOVD P4,A
00E0 8980        243        ORL P1,#80H ; DISABLE THE LED/DAC IO EXPANDER
00E2 9A7F        244        ANL P2,#7FH ; ENABLE THE MAIN PROGRAM IO EXPANDER
                 245
00E4 BB09        246        MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE ADDRESS
                 247                          ; OF THE FIRST INTERRUPT ROUTINE IN R3
00E6 93          248        RETR




                 249
00E7 23EE        250 MREST: MOV A,#-18 ; START DELAYED FOR 18 UNITS
00E9 1437        251        CALL WAITTM
                 252
00EB 8910        253        ORL P1,#10H ; SWITCH ON THE FIELD TRANSISTOR UNTIL THE FIELD
                 254                    ; CURRENT REACHES THE VALUE DETERMINED BY THE FIELD DAC
00ED 0F          255 FON:   MOVD A,P6 ; CHECK FIELD CURRENT LEVEL
00EE 49          256        MOV R1,A
00EF 0E          257        MOVD A,P6 ; CONFIRM WITH A DELAYED CHECK
00F0 59          258        ANL A,R1
00F1 12F5        259        JB0 MSEN
00F3 04ED        260        JMP FON
                 261
00F5 0C          262 MSEN:  MOVD A,P4 ; INITIAL POSITION SENSE. SIGNAL IS DUE TO D(IF)/D(T)
00F6 2400        263        JMP NPAGE
0100             264        ORG 100H
0100 1206        265 NPAGE: JB0 SOBC
0102 720A        266 SOAD:  JB3 SA0 ; SENSED CD MEANING POSITION A
0104 241C        267        JMP SD0 ; SENSED BC MEANING POSITION D
0106 3216        268 SOBC:  JB1 SC0 ; SENSED AB MEANING POSITION C
0108 2410        269        JMP SB0 ; SENSED DA MEANING POSITION B
                 270
                 271        ; DEFINE R7 AS CURRENT PHASE SWITCH, R6 AS NEXT PHASE SWITCH
010A BF01        272 SA0:   MOV R7,#01H
010C BE02        273        MOV R6,#02H
010E 2420        274        JMP S1
```

RUN THE MOTOR

```
 LOC  OBJ        LINE        SOURCE STATEMENT

0110 BF02        275 SR0:    MOV R7,#02H
0112 BF04        276         MOV R6,#04H
0114 2420        277         JMP S1
0116 BF04        278 SC0:    MOV R7,#04H
0118 BF08        279         MOV R6,#08H
011A 2420        280         JMP S1
011C BF08        281 SD0:    MOV R7,#08H
011E BF01        282         MOV R6,#01H
                 283
                 284         ; STAGE 1
                 285
0120 2304        286 S1:     MOV A,#04H ; SELECT SENSE INTERRUPT
0122 3E          287         MOVD P6,A
0123 BB09        288         MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
                 289                           ; ADDRESS IN R3
0125 FE          290         MOV A,R6
0126 3F          291         MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
0127 FF          292         MOV A,R7
0128 43E0        293         ORL A,#0E0H
012A 39          294         OUTL P1,A ; TURN OFF THE FIELD AND TURN ON THE SELECTED PHASE
012B 990F        295         ANL P1,#00FH ; TURN OFF THE PROHIBITION OF DAC CURRENT REGULATION
                 296
012D 27          297         CLR A
012E 62          298         MOV T,A
012F 55          299         STRT T ; START THE TIMER FROM 0
0130 BB04        300         MOV R5,#4 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 301
0132 05          302         EN I ; ENABLE THE EXTERNAL INTERRUPT
                 303
0133 4A3F        304 WORKL1: JNT1 WORKL2
0135 1639        305         JTF MORET1
0137 2433        306         JMP WORKL1
0139 FD33        307 MORET1: DJNZ R5,WORKL1
013B 15          308         DIS I
013C 65          309         STOP TCNT
013D 2449        310         JMP RESTRT
                 311
013F 5A5A        312 WORKL2: JT1 S2
0141 1645        313         JTF MORET2




0143 243F        314         JMP WORKL2
0145 FD3F        315 MORET2: DJNZ R5,WORKL2
0147 15          316         DIS I
0148 65          317         STOP TCNT
0149 23D0        318 RESTRT: MOV A,#0D0H ; TURN OFF THE TRANSISTOR
014B 39          319         OUTL P1,A
014C 1C          320         INC R4 ; IF 10 TRIALS ARE UP THEN STOP TILL TO IS PRESSED AGAIN
014D FC          321         MOV A,R4
014E D452        322         J7 HERE11
0150 04E7        323         JMP MREST
0152 2652        324 HERE11: JNT0 HERE11
0154 3654        325 HERE12: JT0 HERE12
0156 BCF6        326         MOV R4,#-10 ; ALLOW 10 RESTART ATTEMPTS
0158 04E7        327         JMP MREST
                 328
                 329         ; STAGE 2
```

```
  LOC  OBJ          LINE        SOURCE STATEMENT

  015A BB0A          330 S2:     MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS OF
                     331                           ; THE INTERRUPT ROUTINE THAT CHANGES THE DAC
                     332                           ; LEVELS IN R3
  015C 1660          333 WORKL3: JTF MORET3
  015E 245C          334         JMP WORKL3
  0160 EDFC          335 MORET3: DJNZ R5,WORKL3
  0162 15            336         DIS I
  0163 65            337         STOP TCNT
  0164 2389          338         MOV A,#89H ; DISABLE THE MAIN PROGRAM IO EXPANDER



  0166 3A            339         OUTL P2,A
  0167 044E          340         JMP START2
                     341
                     342         END
```

```
USER SYMBOLS
DATA1  0009   DATA2  000A   EXTINT 0003   FON    00ED   HERE1  000F   HERE10 004C   HERE11 0152   HERE12 015A
HERE2  0011   HERE3  0020   HERE4  0022   HERE5  0030   HERE6  0032   HERE7  003A   HERE8  003C   HERE9  004A
MORET1 0139   MORET2 0145   MORET3 0160   MREST  00E7   MSEN   00F5   NPAGE  0100   RESET  0008   RESTRT 0149
SOAD   0102   SOBC   0106   S1     0120   S2     015A   SA0    010A   SB0    0110   SC0    0116   SD0    011C
SERVE1 008F   SERVE2 00A6   START2 004E   SYSRST 0000   TIMINT 0007   WAIT1T 0085   WAITTF 0089   WAITTM 0087
WORKL1 0133   WORKL2 013F   WORKL3 015C   WTMR   008D

ASSEMBLY COMPLETE,  NO ERRORS
```

```
ASM48 :F1:TSTDEL.MAC DEBUG XREF TITLE('CONSTANT SWITCHING DELAY')


ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    1
CONSTANT SWITCHING DELAY

 LOC  OBJ          LINE        SOURCE STATEMENT

                     1          ; PROGRAM TO RUN THE MOTOR WITH A FIXED TIME DELAY AFTER NEXT
                     2          ; PHASE DETECTION BEFORE PHASE SWITCHING
                     3          ; INITIALLY THE FIELD TRANSISTOR IS TURNED ON AND THE FIELD CURRENT
                     4          ; IS ALLOWED TO RISE UNTIL IT REACHES A LEVEL DETERMINED BY THE
                     5          ; FIELD DAC VALUE FED IN ON THE INPUT DATA SWITCHES.  A RESTART
                     6          ; WILL ALSO SUBJECT FIELD CURRENT TO THE SAME REGULATION.  PHASE
                     7          ; CURRENT WILL BE KEPT BETWEEN MAXIMUM AND MINIMUM LEVELS DETERMINED
                     8          ; BY THE MAXIMUM AND MINIMUM DAC VALUES FED IN ON THE INPUT DATA
                     9          ; SWITCHES.  EXTERNAL INTERRUPTS TO INDICATE DETECTION OF THE
                    10          ; NEXT PHASE WILL BE TRIGGERED BY ZERO CROSSINGS OF THE SENSE
                    11          ; WAVEFORMS GOING FROM NEGATIVE TO POSITIVE POLARITY.  PRESSING
                    12          ; THE T1 BUTTON WILL START THE MOTOR WITH THE TIME DELAY ORIGINALLY
                    13          ; SPECIFIED.  TO CHANGE TO NEW TIME DELAY VALUES SET THE INPUT
                    14          ; DATA SWITCHES APPROPRIATELY AND ALTERNATELY PRESS TO AND T1.
                    15          ; TO RUN THE MOTOR:
                    16          ; 1.) ON P73 TO P60, THE TOP SET OF INPUT DATA SWITCHES, INDICATE
                    17          ; THE FIELD DAC VALUE WITH P73 AS THE MSB AND P60 AS THE LSB.
                    18          ; ON P53 TO P40, THE BOTTOM SET OF INPUT DATA SWITCHES, INDICATE
                    19          ; THE VALUE TO BE APPLIED TO THE DAC ADDRESSING THE MAXIMUM VALUE
                    20          ; (TOP) COMPARATORS WITH P53 AS THE MSB AND P40 AS THE LSB.
                    21          ; PRESS TO.
                    22          ; 2.) ON P73 TO P60 INDICATE THE VALUE TO BE APPLIED TO THE DAC
                    23          ; ADDRESSING THE MINIMUM VALUE(MIDDLE) COMPARATORS.  ON P53 TO
                    24          ; P40 INDICATE THE VALUE TO BE APPLIED TO THE DAC ADDRESSING THE
                    25          ; BOTTOM COMPARATORS.
                    26          ; PRESS T1.
                    27          ; 3.) ON P73 TO P40 INDICATE THE DELAY TIME AFTER NEXT PHASE DETECTION
                    28          ; BEFORE PHASE SWITCHING IS TO OCCUR.
                    29          ; PRESS TO.
                    30          ; 4.) PRESS T1 TO RUN THE MOTOR.
                    31          ; 5.) TO CHANGE TO A NEW TIME DELAY INDICATE THE DESIRED VALUE AND
                    32          ; ALTERNATELY PRESS TO AND T1.
                    33
0000                34          ORG 0
                    35 SYSRST:  ; SYSTEM RESET
0000 040R           36          JMP RESET
                    37
0003                38          ORG 3
                    39 EXTINT:  ; EXTERNAL INTERRUPT
0003 F8             40          MOV A,R3 ; JUMP TO THE EXTERNAL INTERRUPT ROUTINE WHOSE LOCATION
0004 B3             41          JMPP @A  ; IS STORED AT THE ADDRESS IN R3
                    42
0007                43          ORG 7
                    44 TIMINT:  ; TIMER INTERRUPT
0007 35             45          DIS TCNTI
0008 93             46          RETR
                    47
0009 92             48 DATA1:   DB LOW SERVE1
000A 84             49 DATA2:   DB LOW SERVE2
                    50
000B 23B0           51 RESET:   MOV A,#80H ; TURN OFF THE TRANSISTORS, ENABLE THE INPUT SWITCH
000D 39             52          OUTL P1,A  ; IO EXPANDER, AND DISABLE THE LED/DAC IO EXPANDER
000E 3A             53          OUTL P2,A ; DISABLE THE MAIN PROGRAM IO EXPANDER
000F 240F           54 HERE1:   JMP0 HERE1
```

CONSTANT SWITCHING DELAY

```
LOC  OBJ        LINE        SOURCE STATEMENT

0011 3611         55 HERE2:  JTO HERE2 ; WAIT FOR T0 TO BE PRESSED
0013 B820         56         MOV R0,#20H
0015 0F           57         MOVD A,P7
0016 47           58         SWAP A
0017 AA           59         MOV R2,A
0018 0F           60         MOVD A,P6
0019 4A           61         ORL A,R2
001A A0           62         MOV @R0,A ; MOVE THE FIELD DAC VALUE INTO LOCATION 20H
001B 18           63         INC R0
001C 0D           64         MOVD A,P5
001D 47           65         SWAP A
001E AA           66         MOV R2,A
001F 0C           67         MOVD A,P4
0020 4A           68         ORL A,R2
0021 A0           69         MOV @R0,A ; MOVE THE MAXIMUM DAC VALUE INTO LOCATION 21H



0022 4622         70 HERE3:  JNT1 HERE3
0024 5624         71 HERE4:  JT1 HERE4 ; WAIT FOR T1 TO BE PRESSED
0026 18           72         INC R0
0027 0F           73         MOVD A,P7
0028 47           74         SWAP A
0029 AA           75         MOV R2,A
002A 0F           76         MOVD A,P6
002B 4A           77         ORL A,R2
002C A0           78         MOV @R0,A ; MOVE THE MINIMUM DAC VALUE INTO LOCATION 22H
002D 18           79         INC R0
002E 0D           80         MOVD A,P5
002F 47           81         SWAP A
0030 AA           82         MOV R2,A
0031 0C           83         MOVD A,P4
0032 4A           84         ORL A,R2
0033 A0           85         MOV @R0,A ; MOVE THE BOTTOM DAC VALUE INTO LOCATION 23H
0034 2634         86 HERE5:  JNT0 HERE5
0036 3636         87 HERE6:  JT0 HERE6 ; WAIT FOR T0 TO BE PRESSED
0038 0F           88         MOVD A,P7
0039 47           89         SWAP A
003A AA           90         MOV R2,A
003B 0F           91         MOVD A,P6
003C 4A           92         ORL A,R2
003D A9           93         MOV R1,A ; PUT THE 8 HIGH BITS IN R1
003E 0D           94         MOVD A,P5
003F 47           95         SWAP A
0040 AA           96         MOV R2,A
0041 0C           97         MOVD A,P4
0042 4A           98         ORL A,R2 ; PUT THE 8 LOW BITS IN A
0043 17           99         INC A ; ADD 1 TO THE 8 LOW INPUT BITS SO THAT AN INPUT 0 YIELDS
                 100               ; THE SMALLEST TIME DELAY
0044 19          101         INC R1 ; ADD 1 TO THE 8 HIGH INPUT BITS SO THAT AN INPUT 0 YIELDS
                 102               ; THE SMALLEST TIME DELAY
0045 18          103         INC R0 ; PUT THE 8 LOW DELAY BITS FOR THE DJNZ DELAY INTO LOCATION
0046 A0          104         MOV @R0,A ; 24H
0047 F9          105         MOV A,R1
0048 18          106         INC R0 ; PUT THE 8 HIGH DELAY BITS FOR THE DJNZ DELAY INTO LOCATION
0049 A0          107         MOV @R0,A ; 25H
                 108
004A 2340        109         MOV A,#40H ; DISABLE THE INPUT SWITCH TO EXPANDER
```

```
LOC  OBJ       LINE       SOURCE STATEMENT

004C 39         110        OUTL P1,A  ; AND ENABLE THE LED/DAC TO EXPANDER
004D 230C       111        MOV A,#0CH ; DISABLE THE 2 MAXIMUM DAC LATCHES
004F 3B         112        MOVD P5,A  ; AND ENABLE THE 2 FIELD DAC LATCHES
0050 27         113        CLR A      ; DISABLE THE 2 MINIMUM
0051 3C         114        MOVD P4,A  ; AND 2 BOTTOM DAC LATCHES
0052 B820       115        MOV R0,#20H
0054 F0         116        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 20H OUT TO
0055 3E         117        MOVD P6,A ; THE FIELD DAC
0056 47         118        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 20H OUT TO
0057 3F         119        MOVD P7,A ; THE FIELD DAC
0058 2303       120        MOV A,#03H ; DISABLE THE 2 FIELD DAC LATCHES
005A 3B         121        MOVD P5,A  ; AND ENABLE THE 2 MAXIMUM DAC LATCHES
005B 18         122        INC R0
005C F0         123        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 21H OUT TO
005D 3E         124        MOVD P6,A ; THE MAXIMUM VALUE DAC
005E 47         125        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 21H OUT TO
005F 3F         126        MOVD P7,A ; THE MAXIMUM VALUE DAC
0060 27         127        CLR A
0061 3B         128        MOVD P5,A ; DISABLE THE 2 MAXIMUM DAC LATCHES
0062 230C       129        MOV A,#0CH
0064 3C         130        MOVD P4,A ; ENABLE THE 2 MINIMUM DAC LATCHES
0065 18         131        INC R0
0066 F0         132        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 22H OUT TO
0067 3E         133        MOVD P6,A ; THE MINIMUM VALUE DAC
0068 47         134        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 22H OUT TO
0069 3F         135        MOVD P7,A ; THE MINIMUM VALUE DAC
006A 2303       136        MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES
006C 3C         137        MOVD P4,A  ; AND ENABLE THE 2 BOTTOM DAC LATCHES
006D 18         138        INC R0
006F F0         139        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 23H OUT TO
006F 3E         140        MOVD P6,A ; THE BOTTOM DAC
0070 47         141        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 23H OUT TO
0071 3F         142        MOVD P7,A ; THE BOTTOM DAC
0072 27         143        CLR A
0073 3C         144        MOVD P4,A ; DISABLE THE 2 BOTTOM DAC LATCHES
0074 23C0       145        MOV A,#0C0H
0076 39         146        OUTL P1,A ; DISABLE THE LED/DAC TO EXPANDER
0077 27         147        CLR A
0078 3A         148        OUTL P2,A ; ENABLE THE MAIN PROGRAM TO EXPANDER
0079 4679       149 HERE7: JNT1 HERE7
007B 567B       150 HERE8: JT1 HERE8 ; WAIT FOR T1 TO BE PRESSED
007D BCF6       151        MOV R4,#-10 ; ALLOW ONLY 10 ATTEMPTS TO START THE MOTOR
007F BB09       152 START2: MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST
                153                           ; INTERRUPT ADDRESS IN R3
0081 2304       154        MOV A,#04H
0083 3E         155        MOVD P6,A ; SELECT SENSE INTERRUPT
0084 BD14       156        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
0086 04EE       157        JMP MREST
                158
                159
                160        ; SUBROUTINES
0088 23FF       161 WAIT1T: MOV A,#-1 ; WAIT1T WAITS FOR 1 TIMER UNIT
008A 62         162 WAITTM: MOV T,A ; WAITTM WAITS FOR # OF UNITS OF TIME
008B 55         163        STRT T  ; EQUAL TO THAT IN A
008C 1690       164 WAITTF: JTF WTMR ; WAITTF WAITS FOR TIMER FLAG
```

```
LOC  OBJ        LINE     SOURCE STATEMENT


008F 048C        165        JMP WAITTF
0090 65          166 WTMR:  STOP TCNT
0091 93          167        RETR
                 168
                 169        ; EXTERNAL INTERRUPT ROUTINES
0092 E992        170 SERVE1: DJNZ R1,SERVE1 ; DECREMENT 8 LOW BITS OF TIME DELAY
0094 EA92        171        DJNZ R2,SERVE1 ; DECREMENT 8 HIGH BITS OF TIME DELAY
                 172
0096 FE          173        MOV A,R6
0097 43E0        174        ORL A,#0E0H
0099 39          175        OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
009A 99DF        176        ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                 177
009C 27          178        CLR A
009D 62          179        MOV T,A
009E 55          180        STRT T ; START THE TIMER FROM 0
009F BD14        181        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 182
00A1 FE          183        MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
00A2 AF          184        MOV R7,A
00A3 47          185        SWAP A
00A4 4E          186        ORL A,R6
00A5 F7          187        RL A
00A6 530F        188        ANL A,#0FH
00A8 AE          189        MOV R6,A
                 190
00A9 3F          191        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 192
00AA BCF6        193        MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                 194
00AC B824        195        MOV R0,#24H
00AE F0          196        MOV A,@R0 ; PUT THE 8 LOW BITS OF TIME DELAY IN LOCATION 24H
00AF A9          197        MOV R1,A  ; INTO R1
00B0 18          198        INC R0
00B1 F0          199        MOV A,@R0 ; PUT THE 8 HIGH BITS OF TIME DELAY IN LOCATION 25H
00B2 AA          200        MOV R2,A  ; INTO R2
00B3 93          201        RETR
                 202
00B4 E9B4        203 SERVE2: DJNZ R1,SERVE2 ; DECREMENT THE 8 LOW BITS OF TIME DELAY
00B6 EAB4        204        DJNZ R2,SERVE2 ; DECREMENT THE 8 HIGH BITS OF TIME DELAY
                 205
00B8 FE          206        MOV A,R6
00B9 43E0        207        ORL A,#0E0H
00BB 39          208        OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
00BC 99DF        209        ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                 210
00BE 27          211        CLR A
00BF 62          212        MOV T,A
00C0 55          213        STRT T ; START THE TIMER FROM 0
00C1 BD14        214        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 215
00C3 FE          216        MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
00C4 AF          217        MOV R7,A
00C5 47          218        SWAP A
00C6 4E          219        ORL A,R6
```

```
LOC  OBJ        LINE        SOURCE STATEMENT

00C7 F7          220         RL A
00C8 530F        221         ANL A,#0FH
00CA AF          222         MOV R6,A
                 223
00CB 3F          224         MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 225
00CC BCF6        226         MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                 227
00CE 8A80        228         ORL P2,#80H ; DISABLE THE MAIN PROGRAM TO EXPANDER



00D0 998F        229         ANL P1,#08FH ; ENABLE THE INPUT SWITCH TO EXPANDER
00D2 0D          230         MOVD A,P5
00D3 47          231         SWAP A
00D4 AA          232         MOV R2,A
00D5 0C          233         MOVD A,P4
00D6 4A          234         ORL A,R2
00D7 A9          235         MOV R1,A ; PUT THE 8 LOW BITS IN R1
00D8 0F          236         MOVD A,P7
00D9 47          237         SWAP A
00DA AA          238         MOV R2,A
00DB 0E          239         MOVD A,P6
00DC 4A          240         ORL A,R2
00DD AA          241         MOV R2,A ; PUT THE 8 HIGH BITS IN R2
00DF 19          242         INC R1 ; ADD 1 TO THE 8 LOW BITS SO THAT AN INPUT 0 YIELDS THE
                 243                ; SMALLEST TIME DELAY
00DF 1A          244         INC R2 ; ADD 1 TO THE 8 HIGH BITS SO THAT AN INPUT 0 YIELDS THE
                 245                ; SMALLEST TIME DELAY
00F0 F9          246         MOV A,R1
00E1 B824        247         MOV R0,#24H ; PUT THE 8 LOW BITS OF TIME DELAY
00E3 A0          248         MOV @R0,A   ; INTO LOCATION 24H
00E4 FA          249         MOV A,R2
00E5 18          250         INC R0      ; PUT THE 8 HIGH BITS OF TIME DELAY
00E6 A0          251         MOV @R0,A ; INTO LOCATION 25H
00E7 8940        252         ORL P1,#40H ; DISABLE THE INPUT SWITCH TO EXPANDER
00E9 9A7F        253         ANL P2,#7FH ; ENABLE THE MAIN PROGRAM TO EXPANDER
00EB BB09        254         MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
                 255                           ; ADDRESS IN R3
00ED 93          256         RETR
                 257
00EF 23EE        258 MREST:  MOV A,#-18 ; START DELAYED FOR 18 UNITS
00F0 148A        259         CALL WAITTM
                 260
00F2 8910        261         ORL P1,#10H ; SWITCH ON THE FIELD TRANSISTOR UNTIL THE FIELD CURRENT
                 262                     ; REACHES THE VALUE DETERMINED BY THE FIELD DAC
00F4 0F          263 FON:    MOVD A,P6 ; CHECK FIELD CURRENT LEVEL
00F5 A9          264         MOV R1,A
00F6 0E          265         MOVD A,P6 ; CONFIRM WITH A DELAYED CHECK
00F7 59          266         ANL A,R1
00F8 12FC        267         JB0 MSEN
00FA 04F4        268         JMP FON
                 269
00FC 0C          270 MSEN:   MOVD A,P4 ; INITIAL POSITION SENSE, SIGNAL IS DUE TO D(IF)/D(T)
00FD 2400        271         JMP NPAGE
0100             272         ORG 100H
0100 1206        273 NP40E:  JB0 S00C
0102 720A        274 S0A0:   JB3 SA0 ; SENSED CO MEANING POSITION A
```

```
LOC  OBJ       LINE       SOURCE STATEMENT

0104 241C       275        JMP SD0 ; SENSED BC MEANING POSITION D
0106 3216       276 SORC:  JB1 SC0 ; SENSED AB MEANING POSITION C
0108 2410       277        JMP SR0 ; SENSED DA MEANING POSITION B
                278
                279        ; DEFINE R7 AS CURRENT PHASE SWITCH, R6 AS NEXT PHASE SWITCH
010A BF01       280 SA0:   MOV R7,#01H
010C BF02       281        MOV R6,#02H
010E 2420       282        JMP S1
0110 BF02       283 SB0:   MOV R7,#02H
0112 BE04       284        MOV R6,#04H
0114 2420       285        JMP S1
0116 BF04       286 SC0:   MOV R7,#04H
0118 BE08       287        MOV R6,#08H
011A 2420       288        JMP S1
011C BF08       289 SD0:   MOV R7,#08H
011E BE01       290        MOV R6,#01H
                291
                292        ; STAGE 1
                293



0120 B824       294 S1:    MOV R0,#24H
0122 F0         295        MOV A,@R0 ; PUT THE 8 LOW BITS OF TIME DELAY IN LOCATION 24H
0123 A9         296        MOV R1,A  ; INTO R1
0124 18         297        INC R0
0125 F0         298        MOV A,@R0 ; PUT THE 8 HIGH BITS OF TIME DELAY IN LOCATION 25H
0126 AA         299        MOV R2,A  ; INTO R2
                300
0127 FE         301        MOV A,R6
0128 3F         302        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
0129 FF         303        MOV A,R7
012A 43E0       304        ORL A,#0E0H
012C 39         305        OUTL P1,A ; TURN OFF THE FIELD AND TURN ON THE SELECTED PHASE
012D 99DF       306        ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC CURRENT REGULATION
                307
012F 27         308        CLR A
0130 62         309        MOV T,A
0131 55         310        STRT T ; START THE TIMER FROM 0
                311
0132 05         312        EN I ; ENABLE THE EXTERNAL INTERRUPT
                313
0133 263F       314 WORKL1: JNT0 WORKL2
0135 1639       315        JTF MORET1
0137 2433       316        JMP WORKL1
0139 ED33       317 MORET1: DJNZ R5,WORKL1
013B 15         318        DIS I
013C 65         319        STOP TCNT
013D 2463       320        JMP RESTRT
                321
013F 3648       322 WORKL2: JT0 CHANG1
0141 1645       323        JTF MORET2
0143 243F       324        JMP WORKL2
0145 ED3F       325 MORET2: DJNZ R5, WORKL2
0147 15         326        DIS I
0148 65         327        STOP TCNT
0149 2463       328        JMP RESTRT
                329
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

014R BR0A        330 CHANG1: MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS OF
                 331                           ; THE INTERRUPT ROUTINE THAT CHANGES THE DELAY
                 332                           ; TIME IN R3
                 333
014D 4A59        334 WORKL3: JNT1 WORKL4
014F 1653        335         JTF MORET3
0151 244D        336         JMP WORKL3
0153 ED4D        337 MORET3: DJNZ R5,WORKL3
0155 15          338         DIS I
0156 65          339         STOP TCNT
0157 2463        340         JMP RESTRT
                 341
0159 566E        342 WORKL4: JT1 CHANG2
015B 165F        343         JTF MORET4
015D 2459        344         JMP WORKL4
015F ED59        345 MORET4: DJNZ R5,WORKL4
0161 15          346         DIS I
0162 65          347         STOP TCNT
0163 23C0        348 RESTRT: MOV A,#0C0H
0165 39          349         OUTL P1,A ; TURN OFF THE TRANSISTOR
0166 1C          350         INC R4 ; IF 10 TRIALS ARE UP THEN STOP TILL T1 IS PRESSED AGAIN
0167 FC          351         MOV A,R4
0168 C66C        352         JZ STEP
016A 047F        353         JMP START2
016C 0479        354 STEP:   JMP HERE7
                 355
016E BR0A        356 CHANG2: MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS OF
                 357                           ; THE INTERRUPT ROUTINE THAT CHANGES THE DELAY
                 358                           ; TIME IN R3




0170 2433        359         JMP WORKL1
                 360
                 361         END
```

USER SYMBOLS

CHANG1	014R	CHANG2	016F	DATA1	0009	DATA2	000A	EXTINT	0003	F0N	00F4	HERE1	000F	HERE2	0011
HERE3	0022	HERE4	0024	HERE5	0034	HERE6	0036	HERE7	0079	HERE8	007R	MORET1	0139	MORET2	0145
MORET3	0153	MORET4	015F	MRFST	00EE	MSEN	00FC	NPAGE	0100	RESET	000B	RESTRT	0163	SOAD	0107
SORC	0106	S1	0120	SA0	010A	S30	0110	SC0	0116	SD0	011C	SERVE1	0092	SERVE2	0084
START2	007F	STEP	016C	SYSRST	0000	TIMINT	0007	WAIT1T	0088	WAITTF	008C	WAITTM	008A	WORKL1	0133
WORKL2	013F	WORKL3	014D	WORKL4	0159	WTMR	0090								

ASSEMBLY COMPLETE, NO ERRORS

```
ASM48 :F1:VARDEL.MAC DEBUG XREF TITLE('VARIABLE SWITCHING DELAY')


ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    1
VARIABLE SWITCHING DELAY

 LOC  OBJ          LINE        SOURCE STATEMENT

                      1        ; PROGRAM TO RUN THE MOTOR AT A FIXED SPEED BY VARYING THE TIME
                      2        ; DELAY AFTER NEXT PHASE DETECTION BEFORE PHASE SWITCHING
                      3        ; INITIALLY THE FIELD TRANSISTOR IS TURNED ON AND THE FIELD CURRENT
                      4        ; IS ALLOWED TO RISE UNTIL IT REACHES A LEVEL DETERMINED BY THE
                      5        ; FIELD DAC VALUE FED IN ON THE INPUT DATA SWITCHES.  A RESTART
                      6        ; WILL ALSO SUBJECT FIELD CURRENT TO THE SAME REGULATION.  PHASE
                      7        ; CURRENT WILL BE KEPT BETWEEN MAXIMUM AND MINIMUM LEVELS DETERMINED
                      8        ; BY THE MAXIMUM AND MINIMUM VALUES FED IN ON THE INPUT DATA SWITCHES.
                      9        ; THE INITIAL TIME DELAY AND THE TIME INTERVAL FOR A CONSECUTIVE
                     10        ; PHASES AT THE DESIRED FIXED SPEED ARE FED IN ON THE INPUT DATA
                     11        ; SWITCHES.  THE CALCULATION OF ACTUAL TIME INTERVAL - DESIRED
                     12        ; TIME INTERVAL IS PERFORMED FOR 4 CONSECUTIVE PHASES USING 2
                     13        ; REGISTER ARITHMETIC.  IF THE RESULT IS POSITIVE, THEN THE MOTOR
                     14        ; SPEED IS TOO SLOW AND THE DELAY TIME IS DECREASED.  IF THE RESULT
                     15        ; IS NEGATIVE, THEN THE MOTOR SPEED IS TOO FAST AND THE DELAY
                     16        ; TIME IS INCREASED.  IF THE RESULT IS ZERO, THE SPEED IS CORRECT
                     17        ; AND THE DELAY TIME IS LEFT UNCHANGED.  A RESTART WILL NOT RESTORE
                     18        ; THE INITIAL TIME DELAY.  ALSO FED IN ON THE INPUT DATA SWITCHES
                     19        ; IS THE NUMBER OF PHASES(FROM 0 TO 255) THAT ARE TO ELAPSE BETWEEN
                     20        ; EVERY SET OF 4 CONSECUTIVE PHASES THAT IS USED FOR SPEED CORRECTION.
                     21        ; EXTERNAL INTERRUPTS TO INDICATE DETECTION OF THE NEXT PHASE
                     22        ; WILL BE TRIGGERED BY ZERO CROSSINGS OF THE SENSE WAVEFORMS GOING
                     23        ; FROM NEGATIVE TO POSITIVE POLARITY.
                     24        ; TO RUN THE MOTOR:
                     25        ; 1.) ON P73 TO P60, THE TOP SET OF DATA INPUT SWITCHES, INDICATE
                     26        ; THE FIELD DAC VALUE WITH P73 AS THE MSB AND P60 AS THE LSB.
                     27        ; ON P53 TO P40, THE BOTTOM SET OF INPUT DATA SWITCHES, INDICATE
                     28        ; THE VALUE TO BE APPLIED TO THE DAC ADDRESSING THE MAXIMUM VALUE
                     29        ; (TOP) COMPARATORS WITH P53 AS THE MSB AND P40 AS THE LSB.
                     30        ; PRESS T0.
                     31        ; 2.) ON P73 TO P60 INDICATE THE VALUE TO BE APPLIED TO THE DAC
                     32        ; ADDRESSING THE MINIMUM VALUE(MIDDLE) COMPARATORS.  ON P53 TO
                     33        ; P40 INDICATE THE VALUE TO BE APPLIED TO THE DAC ADDRESSING THE
                     34        ; BOTTOM COMPARATORS.
                     35        ; PRESS T1.
                     36        ; 3.) ON P73 TO P40 INDICATE THE INITIAL DELAY TIME.
                     37        ; PRESS T0.
                     38        ; 4.) ON P72 TO P40 INDICATE THE DESIRED TIME INTERVAL FOR 4
                     39        ; CONSECUTIVE PHASES.  LEAVE THE P73 IN THE 0 POSITION SINCE
                     40        ; THIS BIT MUST SERVE AS A + OR - SIGN IN THE SUBTRACTION OPERATION
                     41        ; OF ACTUAL TIME INTERVAL - DESIRED TIME INTERVAL.
                     42        ; PRESS T1.
                     43        ; 5.) ON P73 TO P40 INDICATE THE NUMBER OF PHASES THAT ARE TO
                     44        ; ELAPSE BETWEEN EVERY SET OF 4 CONSECUTIVE PHASES THAT IS USED
                     45        ; FOR SPEED CORRECTION.
                     46        ; PRESS T0.
                     47        ; 6.) PRESS T1 TO RUN OR RESTART THE MOTOR.
                     48        ; 7.) TO CHANGE TO A NEW TIME INTERVAL FOR 4 CONSECUTIVE PHASES
                     49        ; INDICATE THE DESIRED VALUE ON P73 TO P40 AND ALTERNATELY PRESS
                     50        ; T0 AND T1.
                     51
0000                 52        ORG 0
                     53 SYSRST: ; SYSTEM RESET
0000 243C            54        JMP RESET
```

```
LOC  OBJ          LINE       SOURCE STATEMENT


                    55
0003                56         ORG 3
                    57 EXTINT: ; EXTERNAL INTERRUPT
0003 F8             58         MOV A,R3 ; JUMP TO THE EXTERNAL INTERRUPT ROUTINE WHOSE LOCATION
0004 B3             59         JMPP @A  ; IS STORED AT THE ADDRESS IN R3
                    60
                    61
0007                62         ORG 7
                    63 TIMINT: ; TIMER INTERRUPT
0007 35             64         DIS TCNTI
0008 93             65         RETR
                    66
0009 16             67 DATA1:  DB LOW SERVE1
000A 4B             68 DATA2:  DB LOW SERVE2
000B FA             69 DATA3:  DB LOW SERVE3
                    70
                    71         ; SUBROUTINES
000C 23FF           72 WAIT1T: MOV A,#-1 ; WAIT1T WAITS FOR 1 TIMER UNIT
000E 62             73 WAITTM: MOV T,A ; WAITTM WAITS FOR # OF UNITS OF TIME
000F 55             74         STRT T  ; EQUAL TO THAT IN A
0010 1614           75 WAITTF: JTF WTMR ; WAITTF WAITS FOR TIMER FLAG
0012 0410           76         JMP WAITTF
0014 65             77 WTMR:   STOP TCNT
0015 93             78         RETR
                    79
                    80         ; EXTERNAL INTERRUPT ROUTINES
001A F916           81 SERVE1: DJNZ R1,SERVE1 ; DECREMENT THE 8 LOW BITS OF TIME DELAY
0018 FA16           82         DJNZ R2,SERVE1 ; DECREMENT THE 8 HIGH BITS OF TIME DELAY
                    83
001A FE             84         MOV A,R6
001B 43E0           85         ORL A,#0E0H
001D 39             86         OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
001E 99DF           87         ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                    88
0020 27             89         CLR A
0021 62             90         MOV T,A
0022 55             91         STRT T ; START THE TIMER FROM 0
0023 B835           92         MOV R0,#35H ; CLEAR THE 8 HIGH BITS OF ACTUAL TIME IN LOCATION
0025 A0             93         MOV @R0,A ; 35H FOR THE NEXT 4 PHASE CYCLE
0026 BD14           94         MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                    95
0028 FE             96         MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
0029 AF             97         MOV R7,A
002A 47             98         SWAP A
002B 4E             99         ORL A,R6
002C F7            100         RL A
002D 530F          101         ANL A,#0FH
002F AE            102         MOV R6,A
                   103
0030 3F            104         MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                   105
0031 BCF6          106         MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                   107
0033 B824          108         MOV R0,#24H
0035 F0            109         MOV A,@R0 ; PUT THE 8 LOW BITS OF TIME DELAY IN LOCATION 24H
```

```
LOC  OBJ       LINE     SOURCE STATEMENT

0036 A9          110        MOV R1,A  ; INTO R1
0037 18          111        INC R0
0038 F0          112        MOV A,@R0 ; PUT THE 8 HIGH BITS OF TIME DELAY IN LOCATION 25H
0039 AA          113        MOV R2,A  ; INTO R2
                 114
003A B838        115        MOV R0,#38H ; DECREMENT THE WORKING REGISTER LOCATION 38H USED
003C F0          116        MOV A,@R0 ; TO KEEP TRACK OF THE NUMBER OF PHASES REMAINING
003D 07          117        DEC A ; BEFORE A SPEED COUNT IS MADE
003E C642        118        JZ BACKS2



0040 A0          119        MOV @R0,A
0041 93          120        RETR
0042 B828        121 BACKS2: MOV R0,#28H
0044 F0          122        MOV A,@R0 ; INITIALIZE THE WORKING REGISTER LOCATION 38H USED
0045 B838        123        MOV R0,#38H ; TO KEEP TRACK OF THE NUMBER OF PHASES REMAINING
0047 A0          124        MOV @R0,A ; BEFORE A SPEED COUNT IS MADE
0048 BB0A        125        MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS
                 126                          ; OF THE INTERRUPT THAT COUNTS FOR 4 PHASES
                 127                          ; IN R3
004A 93          128        RETR
                 129
004B 65          130 SERVE2: STOP TCNT ; STOP THE TIMER COUNT
                 131
004C E94B        132        DJNZ R1,SERVE2 ; DECREMENT THE 8 LOW BITS OF TIME DELAY
004E EA4B        133        DJNZ R2,SERVE2 ; DECREMENT THE 8 HIGH BITS OF TIME DELAY
                 134
0050 FE          135        MOV A,R6
0051 43E0        136        ORL A,#0E0H
0053 39          137        OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
0054 99DF        138        ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                 139
0056 42          140        MOV A,T
0057 AA          141        MOV R2,A ; PUT THE TIMER COUNT IN R2
0058 27          142        CLR A
0059 62          143        MOV T,A
005A 55          144        STRT T ; START THE TIMER FROM 0
005B BD14        145        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 146
005D FE          147        MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
005E AF          148        MOV R7,A
005F 47          149        SWAP A
0060 4E          150        ORL A,R6
0061 E7          151        RL A
0062 530F        152        ANL A,#0FH
0064 AE          153        MOV R6,A
                 154
0065 3F          155        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 156
0066 BCFA        157        MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                 158
0068 B829        159        MOV R0,#29H
006A F0          160        MOV A,@R0
006B E7          161        RL A
006C A0          162        MOV @R0,A ; UPDATE POSITION INDEX IN 4 PHASE CYCLE
006D 129B        163        JB0 DELCHK ; HAVE COMPLETED A 4 PHASE CYCLE
006F 327C        164        JB1 STRTCT ; JUST STARTING A CYCLE
```

```
LOC  OBJ       LINE       SOURCE STATEMENT

0071 B834       165        MOV R0,#34H ; 34H CONTAINS THE 8 LOW BITS OF ACTUAL TIME
                166                    ; 35H CONTAINS THE 8 HIGH BITS OF ACTUAL TIME
0073 FA         167        MOV A,R2
0074 60         168        ADD A,@R0 ; UPDATE THE 8 LOW BITS OF ACTUAL TIME IN 34H
0075 A0         169        MOV @R0,A ; WITH TIME ELAPSED OUTSIDE THE INTERRUPT
0076 E6F2       170        JNC FINIS
0078 18         171        INC R0 ; UPDATE THE 8 HIGH BITS OF ACTUAL TIME IN 35H WITH TIME
0079 10         172        INC @R0 ; ELAPSED OUTSIDE THE INTERRUPT
007A 04F2       173        JMP FINIS
007C B834       174 STRTCT: MOV R0,#34H
007E FA         175        MOV A,R2 ; PUT THE 8 LOW BITS OF TIME ELAPSED OUTSIDE THE INTERRUPT
007F A0         176        MOV @R0,A ; IN 34H
0080 B824       177        MOV R0,#24H
0082 F0         178        MOV A,@R0
0083 07         179        DEC A ; DECREMENT SINCE 1 REPRESENTS THE SHORTEST DELAY TIME
                180           ; AND 0 REPRESENTS THE LONGEST DELAY TIME IN LOCATION 24H
0084 AA         181        MOV R2,A ; PUT THE DECREMENTED 8 LOW BITS OF DELAY TIME IN R2
0085 18         182        INC R0
0086 F0         183        MOV A,@R0 ; PUT THE 8 HIGH BITS OF DELAY TIME IN THE ACCUMULATOR



0087 07         184        DEC A ; DECREMENT SINCE 1 REPRESENTS THE SHORTEST DELAY TIME
                185           ; AND 0 REPRESENTS THE LONGEST DELAY TIME IN LOCATION 25H
0088 97         186        CLR C
0089 67         187        RRC A ; ROTATE THE 8 HIGH BITS TO THE RIGHT
008A 2A         188        XCH A,R2
008B 67         189        RRC A ; ROTATE THE 8 LOW BITS TO THE RIGHT
008C 2A         190        XCH A,R2
008D 97         191        CLR C
008E 67         192        RRC A ; ROTATE THE 8 HIGH BITS TO THE RIGHT
008F 2A         193        XCH A,R2
0090 67         194        RRC A ; ROTATE THE 8 LOW BITS TO THE RIGHT
                195        ; ADD THE DELAY TIME DURING 4 INTERRUPTS TO THE 8 LOW BITS OF
                196        ; ACTUAL TIME OUTSIDE THE INTERRUPT IN LOCATION 34H.  NOTE THAT
                197        ; A DJNZ INSTRUCTION IS 2 CYCLES WHILE A TIMER INCREMENT OCCURS
                198        ; EVERY 32 CYCLES.  THUS, OVER 4 INTERRUPTS A DJNZ INSTRUCTION
                199        ; IS 8 CYCLES, SO 2 RIGHTWARD SHIFTS ON THE DJNZ DELAY TIMES THEN
                200        ; GIVE UNITS OF 32 CYCLES OVER A PERIOD OF 4 INTERRUPTS.
0091 B834       201        MOV R0,#34H
0093 60         202        ADD A,@R0
0094 A0         203        MOV @R0,A
0095 18         204        INC R0 ; THEN ADD THE DELAY TIME DURING 4 INTERRUPTS TO THE 8
0096 FA         205        MOV A,R2 ; HIGH BITS OF ACTUAL TIME IN LOCATION 35H
0097 70         206        ADDC A,@R0
0098 A0         207        MOV @R0,A
0099 04F2       208        JMP FINIS
009B B828       209 DELCHK: MOV R0,#28H ; PUT THE NUMBER OF PHASES TO ELAPSE BETWEEN EVERY
009D F0         210        MOV A,@R0 ; SET OF 4 CONSECUTIVE PHASES USED FOR SPEED CORRECTION
                211                  ; IN THE ACCUMULATOR
009E 96A2       212        JNZ WATSP1
00A0 04A4       213        JMP TOHERE ; NO OTHER PHASES ELAPSE BETWEEN EVERY SET OF 4
00A2 BB09       214 WATSP1: MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
                215                           ; ADDRESS IN R3
00A4 B834       216 TOHERE: MOV R0,#34H
00A6 FA         217        MOV A,R2 ; PLACE THE UPDATED 8 LOW BITS OF ACTUAL TIME IN THE
00A7 60         218        ADD A,@R0 ; ACCUMULATOR
00A8 E6AC       219        JNC SKIPIN
```

```
LOC  OBJ        LINE        SOURCE STATEMENT

00AA 18          220         INC R0 ; UPDATE THE 8 HIGH BITS OF ACTUAL TIME IN 35H
00AB 10          221         INC @R0
00AC B826        222 SKIPIN: MOV R0,#26H
00AE 37          223         CPL A
00AF 60          224         ADD A,@R0
00B0 37          225         CPL A ; PUT THE 8 LOW BITS OF ACTUAL TIME - THE 8 LOW BITS OF
00B1 AA          226         MOV R2,A ; DESIRED TIME IN R2
00B2 B835        227         MOV R0,#35H ; PUT THE 8 HIGH BITS OF ACTUAL TIME IN THE
00B4 F0          228         MOV A,@R0 ; ACCUMULATOR
00B5 B000        229         MOV @R0,#0 ; CLEAR THE 8 HIGH BITS OF ACTUAL TIME IN LOCATION
                 230                    ; 35H FOR THE NEXT CYCLE
00B7 B827        231         MOV R0,#27H
00B9 37          232         CPL A
00BA 70          233         ADDC A,@R0 ; HIGH 8 BITS OF ACTUAL TIME - HIGH 8 BITS OF DESIRED
00BB 37          234         CPL A ; TIME
00BC F2DF        235         JB7 DELINC ; ACTUAL TIME IS LESS THAN DESIRED TIME
                 236                    ; ACTUAL SPEED IS GREATER THAN DESIRED SPEED
                 237                    ; INCREASE THE TIME DELAY
00BE 96C5        238         JNZ DELDCR ; FOR ANY OTHER NONZERO NUMBER DECREASE THE TIME
00C0 FA          239         MOV A,R2 ; DELAY
00C1 96C5        240         JNZ DELDCR
00C3 04F2        241         JMP FINIS
00C5 B824        242 DELDCR: MOV R0,#24H
00C7 F0          243         MOV A,@R0
00C8 07          244         DEC A ; DECREMENT FROM THE 8 LOW BITS AND PUT THE NEW VALUE IN
00C9 A0          245         MOV @R0,A ; LOCATION 24H
00CA CACE        246         JZ SKIPDE ; IF A 0 RESULTS DECREMENT FROM THE 8 HIGH BITS SINCE
                 247                   ; 1 REPRESENTS THE LOWEST TIME DELAY AND 0 REPRESENTS
                 248                   ; THE HIGHEST TIME DELAY


00CC 04F2        249         JMP FINIS
00CE 18          250 SKIPDE: INC R0
00CF F0          251         MOV A,@R0 ; DECREMENT FROM THE 8 HIGH BITS AND PUT THE NEW VALUE
00D0 07          252         DEC A ; IN LOCATION 25H
00D1 C6D6        253         JZ LOWEST ; UNLESS A ZERO RESULTS IN THE 8 HIGH BITS
00D3 A0          254         MOV @R0,A
00D4 04F2        255         JMP FINIS
00D6 B824        256 LOWEST: MOV R0,#24H ; IN WHICH CASE THE DELAY IS ALREADY AS LOW AS POSSIBLE
00D8 B001        257         MOV @R0,#1 ; THEN KEEP IT THIS WAY BY PUTTING A ONE VALUE IN BOTH
00DA 18          258         INC R0 ; DELAY REGISTERS
00DB B001        259         MOV @R0,#1
00DD 04F2        260         JMP FINIS
00DF B824        261 DELINC: MOV R0,#24H
00E1 F0          262         MOV A,@R0 ; INCREMENT THE 8 LOW BITS IN LOCATION 24H
00E2 10          263         INC @R0
00E3 C6E7        264         JZ MOREIN ; IF A ZERO, THE HIGHEST POSSIBLE DELAY, WAS INCREMENTED
                 265                   ; TO 1, THE LOWEST POSSIBLE DELAY, THEN INCREMENT THE
                 266                   ; 8 HIGH BITS
00E5 04F2        267         JMP FINIS
00E7 18          268 MOREIN: INC R0 ; IF THE 8 HIGH BITS ARE ALREADY 0, THEN THE DELAY IS
00E8 F0          269         MOV A,@R0 ; AS LARGE AS POSSIBLE
00E9 CAEE        270         JZ HIGHES ; IF THE 8 HIGH BITS ARE NOT 0, THEN INCREMENT THE HIGH
00EB 10          271         INC @R0 ; BIT DELAY REGISTER
00EC 04F2        272         JMP FINIS
00EE B824        273 HIGHES: MOV R0,#24H ; IF THE DELAY IS AS LARGE AS POSSIBLE, THEN KEEP
00F0 B000        274         MOV @R0,#0 ; IT THIS WAY BY PUTTING A ZERO IN THE LOW BIT DELAY
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

                 275                  ; REGISTER
00F2 B824        276 FINIS:  MOV R0,#24H
00F4 F0          277         MOV A,@R0 ; PUT THE 8 LOW BITS OF TIME DELAY IN LOCATION 24H
00F5 A9          278         MOV R1,A ; INTO R1
00F6 18          279         INC R0
00F7 F0          280         MOV A,@R0 ; PUT THE 8 HIGH BITS OF TIME DELAY IN LOCATION 25H
00F8 AA          281         MOV R2,A ; INTO R2
00F9 93          282         RETR
                 283
00FA E9FA        284 SERVE3: DJNZ R1,SERVE3 ; DECREMENT THE 8 LOW BITS OF TIME DELAY
00FC EAFA        285         DJNZ R2,SERVE3 ; DECREMENT THE 8 HIGH BITS OF TIME DELAY
                 286
00FE FE          287         MOV A,R6
00FF 43E0        288         ORL A,#0E0H
0101 39          289         OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
0102 990F        290         ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                 291
0104 27          292         CLR A
0105 62          293         MOV T,A
0106 55          294         STRT T ; START THE TIMER FROM 0
0107 B835        295         MOV R0,#35H ; CLEAR THE 8 HIGH BITS OF ACTUAL TIME IN LOCATION 35H
0109 A0          296         MOV @R0,A   ; FOR THE NEXT 4 PHASE CYCLE
010A BD14        297         MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 298
010C FE          299         MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
010D AF          300         MOV R7,A
010E 47          301         SWAP A
010F 4E          302         ORL A,R6
0110 F7          303         RL A
0111 530F        304         ANL A,#0FH
0113 AE          305         MOV R6,A
                 306
0114 3F          307         MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 308
0115 BCF6        309         MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                 310
0117 8A80        311         ORL P2,#80H ; DISABLE THE MAIN PROGRAM IO EXPANDER
0119 99BF        312         ANL P1,#0BFH ; ENABLE THE INPUT SWITCH IO EXPANDER
011B 0D          313         MOVD A,P5




011C 47          314         SWAP A
011D AA          315         MOV R2,A
011E 0C          316         MOVD A,P4
011F 4A          317         ORL A,R2
0120 B826        318         MOV R0,#26H ; PUT THE 8 LOW BITS OF THE DESIRED TIME INTERVAL
0122 A0          319         MOV @R0,A ; FOR 4 CONSECUTIVE PHASES IN LOCATION 26H
0123 0F          320         MOVD A,P7
0124 47          321         SWAP A
0125 AA          322         MOV R2,A
0126 0E          323         MOVD A,P6
0127 4A          324         ORL A,R2
0128 18          325         INC R0 ; PUT THE 8 HIGH BITS OF THE DESIRED TIME INTERVAL FOR
0129 A0          326         MOV @R0,A ; 4 CONSECUTIVE PHASES IN LOCATION 27H
012A 8940        327         ORL P1,#40H ; DISABLE THE INPUT SWITCH IO EXPANDER
012C 947F        328         ANL P2,#7FH ; ENABLE THE MAIN PROGRAM IO EXPANDER
                 329
```

```
LOC  OBJ       LINE       SOURCE STATEMENT

012E B824        330        MOV R0,#24H
0130 F0          331        MOV A,@R0 ; PUT THE 8 LOW BITS OF TIME DELAY IN LOCATION 24H
0131 A9          332        MOV R1,A  ; INTO R1
0132 18          333        INC R0
0133 F0          334        MOV A,@R0 ; PUT THE 8 HIGH BITS OF TIME DELAY IN LOCATION 25H
0134 AA          335        MOV R2,A  ; INTO R2
                 336
0135 B829        337        MOV R0,#29H
0137 B011        338        MOV @R0,#11H ; INITIALIZE TO START OF 4 PHASE CYCLE
                 339
0139 B80A        340        MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS
                 341                          ; OF THE INTERRUPT THAT COUNTS FOR 4 PHASES
                 342                          ; IN R3
                 343
013B 93          344        RETR
                 345
013C 2380        346 RESET: MOV A,#80H ; TURN OFF THE TRANSISTORS, ENABLE THE INPUT SWITCH
013E 39          347        OUTL P1,A ; IO EXPANDER, AND DISABLE THE LED/DAC IO EXPANDER
013F 3A          348        OUTL P2,A ; DISABLE THE MAIN PROGRAM IO EXPANDER
0140 2640        349 HERE1: JNT0 HERE1
0142 2642        350 HERE2: JT0 HERE2 ; WAIT FOR T0 TO BE PRESSED
0144 B820        351        MOV R0,#20H
0146 0F          352        MOVD A,P7
0147 47          353        SWAP A
0148 AA          354        MOV R2,A
0149 0E          355        MOVD A,P6
014A 4A          356        ORL A,R2
014B A0          357        MOV @R0,A ; PUT THE FIELD DAC VALUE IN LOCATION 20H
014C 18          358        INC R0
014D 0D          359        MOVD A,P5
014E 47          360        SWAP A
014F AA          361        MOV R2,A
0150 0C          362        MOVD A,P4
0151 4A          363        ORL A,R2
0152 A0          364        MOV @R0,A ; PUT THE MAXIMUM DAC VALUE IN LOCATION 21H
0153 4453        365 HERE3: JNT1 HERE3
0155 5655        366 HERE4: JT1 HERE4 ; WAIT FOR T1 TO BE PRESSED
0157 18          367        INC R0
0158 0F          368        MOVD A,P7
0159 47          369        SWAP A
015A AA          370        MOV R2,A
015B 0E          371        MOVD A,P6
015C 4A          372        ORL A,R2
015D A0          373        MOV @R0,A ; PUT THE MINIMUM DAC VALUE IN LOCATION 22H
015E 18          374        INC R0
015F 0D          375        MOVD A,P5
0160 47          376        SWAP A
0161 AA          377        MOV R2,A
0162 0C          378        MOVD A,P4

0163 4A          379        ORL A,R2
0164 A0          380        MOV @R0,A ; PUT THE BOTTOM DAC VALUE IN LOCATION 23H
0165 2665        381 HERE5: JNT0 HERE5
0167 3667        382 HERE6: JT0 HERE6 ; WAIT FOR T0 TO BE PRESSED
0169 18          383        INC R0
016A 0D          384        MOVD A,P5
```

```
LOC  OBJ        LINE        SOURCE STATEMENT

016B 47          385         SWAP A
016C AA          386         MOV R2,A
016D 0C          387         MOVD A,P4
016E 4A          388         ORL A,R2
016F 17          389         INC A ; ADD 1 TO THE 8 LOW INPUT BITS SO THAT AN INPUT 0 YIELDS
                 390             ; THE SMALLEST TIME DELAY
0170 A0          391         MOV @R0,A ; PUT THE 8 LOW BITS OF INITIAL TIME DELAY IN LOCATION
                 392                   ; 24H
0171 18          393         INC R0
0172 0F          394         MOVD A,P7
0173 47          395         SWAP A
0174 AA          396         MOV R2,A
0175 0E          397         MOVD A,P6
0176 4A          398         ORL A,R2
0177 17          399         INC A ; ADD 1 TO THE 8 HIGH INPUT BITS SO THAT AN INPUT 0 YIELDS
                 400             ; THE SMALLEST TIME DELAY
0178 A0          401         MOV @R0,A ; PUT THE 8 HIGH BITS OF INITIAL TIME DELAY IN LOCATION
                 402                   ; 25H
0179 4679        403 HERE7:  JNT1 HERE7
017B 567B        404 HERE8:  JT1 HERE8 ; WAIT FOR T1 TO BE PRESSED
017D 18          405         INC R0
017E 0D          406         MOVD A,P5
017F 47          407         SWAP A
0180 AA          408         MOV R2,A
0181 0C          409         MOVD A,P4
0182 4A          410         ORL A,R2 ; PUT THE 8 LOW BITS OF THE DESIRED TIME INTERVAL FOR
0183 A0          411         MOV @R0,A ; 4 CONSECUTIVE PHASES IN LOCATION 26H
0184 18          412         INC R0
0185 0F          413         MOVD A,P7
0186 47          414         SWAP A
0187 AA          415         MOV R2,A
0188 0E          416         MOVD A,P6
0189 4A          417         ORL A,R2 ; PUT THE 8 HIGH BITS OF THE DESIRED TIME INTERVAL FOR
018A A0          418         MOV @R0,A ; 4 CONSECUTIVE PHASES IN 27H
018B 268B        419 HERE9:  JNT0 HERE9
018D 368D        420 HERE10: JT0 HERE10 ; WAIT FOR T0 TO BE PRESSED
018F 18          421         INC R0
0190 0F          422         MOVD A,P7
0191 47          423         SWAP A
0192 AA          424         MOV R2,A
0193 0E          425         MOVD A,P6 ; PUT THE NUMBER OF PHASES THAT ARE TO ELAPSE BETWEEN
0194 4A          426         ORL A,R2 ; EVERY SET OF 4 CONSECUTIVE PHASES THAT IS USED FOR
0195 A0          427         MOV @R0,A ; SPEED CORRECTION IN LOCATION 28H
                 428
0196 2340        429         MOV A,#40H ; DISABLE THE INPUT SWITCH IO EXPANDER
0198 39          430         OUTL P1,A ; AND ENABLE THE LED/DAC IO EXPANDER
0199 230C        431         MOV A,#0CH ; DISABLE THE 2 MAXIMUM DAC LATCHES
019B 3D          432         MOVD P5,A ; AND ENABLE THE 2 FIELD DAC LATCHES
019C 27          433         CLR A ; DISABLE THE 2 MINIMUM
019D 3C          434         MOVD P4,A ; AND 2 BOTTOM DAC LATCHES
019F B820        435         MOV R0,#20H
01A0 F0          436         MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 20H
01A1 3E          437         MOVD P6,A ; OUT TO THE FIELD DAC
01A2 47          438         SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 20H
01A3 3F          439         MOVD P7,A ; OUT TO THE FIELD DAC
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

01A4 2303        440        MOV A,#03H ; DISABLE THE 2 FIELD DAC LATCHES
01A6 3D          441        MOVD P5,A ; AND ENABLE THE 2 MAXIMUM DAC LATCHES
01A7 18          442        INC R0
01A8 F0          443        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 21H
01A9 3E          444        MOVD P6,A ; OUT TO THE MAXIMUM VALUE DAC
01AA 47          445        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 21H
01AB 3F          446        MOVD P7,A ; OUT TO THE MAXIMUM VALUE DAC
01AC 27          447        CLR A
01AD 3D          448        MOVD P5,A ; DISABLE THE 2 MAXIMUM DAC LATCHES
01AF 230C        449        MOV A,#0CH
01B0 3C          450        MOVD P4,A ; ENABLE THE 2 MINIMUM DAC LATCHES
01B1 18          451        INC R0
01B2 F0          452        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 22H
01B3 3E          453        MOVD P6,A ; OUT TO THE MINIMUM VALUE DAC
01B4 47          454        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 22H
01B5 3F          455        MOVD P7,A ; OUT TO THE MINIMUM VALUE DAC
01B6 2303        456        MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES
01B8 3C          457        MOVD P4,A  ; AND ENABLE THE 2 BOTTOM DAC LATCHES
01B9 18          458        INC R0
01BA F0          459        MOV A,@R0 ; MOVE THE 4 LOW BITS IN LOCATION 23H
01BB 3E          460        MOVD P6,A ; OUT TO THE BOTTOM DAC
01BC 47          461        SWAP A    ; MOVE THE 4 HIGH BITS IN LOCATION 23H
01BD 3F          462        MOVD P7,A ; OUT TO THE BOTTOM DAC
01BE 27          463        CLR A
01BF 3C          464        MOVD P4,A ; DISABLE THE 2 BOTTOM DAC LATCHES
                 465
01C0 23C0        466        MOV A,#0C0H
01C2 99          467        OUTL P1,A ; DISABLE THE LED/DAC IO EXPANDER
01C3 27          468        CLR A
01C4 9A          469        OUTL P2,A ; ENABLE THE MAIN PROGRAM IO EXPANDER
01C5 44C5        470 HERE11: JNT1 HERE11
01C7 54C7        471 HERE12: JT1 HERE12 ; WAIT FOR T1 TO BE PRESSED BEFORE STARTING THE MOTOR
01C9 BCF5        472        MOV R4,#-10 ; ALLOW ONLY 10 ATTEMPTS TO START THE MOTOR
01CB B829        473 RESTRT: MOV R0,#29H
01CD B011        474        MOV @R0,#11H ; INITIALIZE TO START OF 4 PHASE CYCLE
01CF B828        475        MOV R0,#28H ; PUT THE NUMBER OF PHASES TO ELAPSE BETWEEN EVERY
01D1 F0          476        MOV A,@R0 ; SET OF 4 CONSECUTIVE PHASES USED FOR SPEED CORRECTION
                 477                  ; IN THE ACCUMULATOR
01D2 96D8        478        JNZ WAITSP
01D4 BB0A        479        MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS
                 480                          ; OF THE INTERRUPT THAT COUNTS FOR 4 PHASES
                 481                          ; IN R3
01D6 24DD        482        JMP ONWARD
01D8 BB09        483 WAITSP: MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST
                 484                          ; INTERRUPT ADDRESS IN R3
01DA B838        485        MOV R0,#38H ; USE LOCATION 38H AS THE WORKING REGISTER TO KEEP
01DC A0          486        MOV @R0,A ; TRACK OF THE NUMBER OF PHASES REMAINING BEFORE A SPEED
                 487                  ; COUNT IS MADE
01DD 2304        488 ONWARD: MOV A,#04H
01DF 3F          489        MOVD P4,A ; SELECT SENSE INTERRUPT
01E0 BD14        490        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 491
01E2 23FF        492 MREST:  MOV A,#-18 ; START DELAYED FOR 18 UNITS
01E4 140E        493        CALL WAITTM
                 494
```

```
LOC  OBJ        LINE        SOURCE STATEMENT

01FA 8910        495         ORL P1,#10H ; SWITCH ON THE FIELD TRANSISTOR UNTIL THE FIELD
                 496                     ; CURRENT REACHES THE VALUE DETERMINED BY THE FIELD DAC
01FB 4400        497         JMP FON
0200             498         ORG 200H
0200 0F          499 FON:    MOVD A,P6 ; CHECK FIELD CURRENT LEVEL
0201 A9          500         MOV R1,A
0202 0F          501         MOVD A,P6 ; CONFIRM WITH A DELAYED CHECK
0203 59          502         ANL A,R1
0204 1208        503         JB0 MSEN
0206 4400        504         JMP FON
                 505
0208 0C          506 MSEN:   MOVD A,P4 ; INITIAL POSITION SENSE, SIGNAL IS DUE TO D(IF)/D(T)
0209 120F        507         JB0 SOBC
020B 7213        508 SOAD:   JB3 SA0 ; SENSED CD MEANING POSITION A
020D 4425        509         JMP SD0 ; SENSED BC MEANING POSITION D
020F 321F        510 SOBC:   JB1 SC0 ; SENSED AB MEANING POSITION C
0211 4419        511         JMP SB0 ; SENSED DA MEANING POSITION B
                 512
                 513         ; DEFINE R7 AS CURRENT PHASE SWITCH, R6 AS NEXT PHASE SWITCH
0213 BF01        514 SA0:    MOV R7,#01H
0215 BE02        515         MOV R6,#02H
0217 4429        516         JMP S1
0219 BF02        517 SB0:    MOV R7,#02H
021B BE04        518         MOV R6,#04H
021D 4429        519         JMP S1
021F BF04        520 SC0:    MOV R7,#04H
0221 BE08        521         MOV R6,#08H
0223 4429        522         JMP S1
0225 BF08        523 SD0:    MOV R7,#08H
0227 BE01        524         MOV R6,#01H
                 525
                 526         ; STAGE 1
                 527
0229 B824        528 S1:     MOV R0,#24H
022B F0          529         MOV A,@R0 ; PUT THE 8 LOW BITS OF TIME DELAY IN LOCATION 24H
022C A9          530         MOV R1,A ; INTO R1
022D 18          531         INC R0
022E F0          532         MOV A,@R0 ; PUT THE 8 HIGH BITS OF TIME DELAY IN LOCATION 25H
022F AA          533         MOV R2,A ; INTO R2
                 534
0230 FF          535         MOV A,R6
0231 3F          536         MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
0232 FF          537         MOV A,R7
0233 43E0        538         ORL A,#0E0H
0235 39          539         OUTL P1,A ; TURN OFF THE FIELD AND TURN ON THE SELECTED PHASE
0236 990F        540         ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC CURRENT REGULATION
                 541
0238 27          542         CLR A
0239 62          543         MOV T,A
023A 55          544         STRT T ; START THE TIMER FROM 0
023B B835        545         MOV R0,#35H ; CLEAR THE 8 HIGH BITS OF ACTUAL TIME IN 35H FOR
023D A0          546         MOV @R0,A ; THE FIRST CYCLE
                 547
023E 05          548         EN I ; ENABLE THE EXTERNAL INTERRUPT
                 549
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

023F 264C        550 WORKL1: JNT0 WORKL2
0241 1645        551         JTF MORET1
0243 443F        552         JMP WORKL1
0245 B835        553 MORET1: MOV R0,#35H ; INCREMENT THE 8 HIGH BITS OF ACTUAL TIME IN
0247 10          554         INC @R0     ; LOCATION 35H



0248 FD3F        555         DJNZ R5,WORKL1
024A 4473        556         JMP NSTRT
                 557
024C 3659        558 WORKL2: JT0 CHANG1
024E 1652        559         JTF MORET2
0250 444C        560         JMP WORKL2
0252 B835        561 MORET2: MOV R0,#35H ; INCREMENT THE 8 HIGH BITS OF ACTUAL TIME IN
0254 10          562         INC @R0     ; LOCATION 35H
0255 FD4C        563         DJNZ R5,WORKL2
0257 4473        564         JMP NSTRT
                 565
0259 BB0B        566 CHANG1: MOV R3,#LOW DATA3 ; STORE THE LOCATION CONTAINING THE ADDRESS
                 567                           ; OF THE INTERRUPT ROUTINE THAT CHANGES THE
                 568                           ; DESIRED SPEED IN R3
                 569
025B 4A68        570 WORKL3: JNT1 WORKL4
025D 1A61        571         JTF MORET3
025F 445B        572         JMP WORKL3
0261 B835        573 MORET3: MOV R0,#35H ; INCREMENT THE 8 HIGH BITS OF ACTUAL TIME IN
0263 10          574         INC @R0     ; LOCATION 35H
0264 FD5B        575         DJNZ R5,WORKL3
0266 4473        576         JMP NSTRT
                 577
0268 5680        578 WORKL4: JT1 CHANG2
026A 1A6E        579         JTF MORET4
026C 4468        580         JMP WORKL4
026F B835        581 MORET4: MOV R0,#35H ; INCREMENT THE 8 HIGH BITS OF ACTUAL TIME IN
0270 10          582         INC @R0     ; LOCATION 35H
0271 FD68        583         DJNZ R5,WORKL4
0273 15          584 NSTRT:  DIS I
0274 65          585         STOP TCNT
0275 23C0        586         MOV A,#0C0H
0277 39          587         OUTL P1,A ; TURN OFF THE TRANSISTOR
0278 1C          588         INC R4 ; IF 10 TRIALS ARE UP THEN STOP TILL T1 IS PRESSED AGAIN
0279 FC          589         MOV A,R4
027A C67E        590         JZ OVERT0
027C 24CB        591         JMP RESTRT
027E 2405        592 OVERT0: JMP HERE11
                 593
0280 BB0B        594 CHANG2: MOV R3,#LOW DATA3 ; STORE THE LOCATION CONTAINING THE ADDRESS
                 595                           ; OF THE INTERRUPT ROUTINE THAT CHANGES THE
                 596                           ; DESIRED SPEED IN R3
0282 443F        597         JMP WORKL1
                 598
                 599         END

USER SYMBOLS
BACKS2 0042   CHANG1 0259   CHANG2 0280   DATA1  0009   DATA2  000A   DATA3  000B   DELCHK 0038   DELDCR 00C5
DELINC 00BF   EXTINT 0003   FINIS  00F2   FON    0200   HERE1  0140   HERE10 0180   HERE11 01C5   HERE12 01C7
HERE2  0142   HERE3  0153   HERE4  0155   HERE5  0165   HERE6  0167   HERE7  0179   HERE8  017B   HERE9  0188
```

VARIABLE SWITCHING DELAY

```
HIGHES 00FF    LOWEST 0006    MOREIN 00E7    MORET1 0245    MORET2 0252    MORET3 02A1    MORET4 026E    MREST  01E2
MSEN   0208    NSTRT  0273    ONWARD 01DD    OVERTO 027E    RESET  013C    RESTRT 01CB    S0A0   020A    S0BC   020F
S1     0229    SA0    0213    SB0    0219    SC0    021F    SD0    0225    SERVE1 0016    SERVE2 004A    SERVE3 00FA
SKIPDE 000E    SKIPIN 00AC    STRTCT 007C    SYSRST 0000    TIMINT 0007    TOHERE 00A4    WAIT1T 000C    WAITSP 01D3
WAITTF 0010    WAITTM 000F    WATSP1 00A2    WORKL1 023F    WORKL2 024C    WORKL3 025A    WORKL4 0268    WTMR   0014
```

ASSEMBLY COMPLETE, NO ERRORS

```
LOC  OBJ        LINE     SOURCE STATEMENT

                    1     ; PROGRAM TO RUN THE MOTOR AT A FIXED SPEED
                    2     ; INITIALLY THE FIELD TRANSISTOR IS TURNED ON AND THE FIELD
                    3     ; CURRENT IS ALLOWED TO RISE UNTIL IT REACHES A MAXIMUM LEVEL
                    4     ; DETERMINED BY THE MAXIMUM VALUE FED IN ON THE DATA INPUT SWITCHES.
                    5     ; A RESTART WILL ALSO SUBJECT FIELD CURRENT TO THE SAME REGULATION.
                    6     ; PHASE CURRENT WILL BE KEPT BETWEEN THE MAXIMUM AND MINIMUM
                    7     ; LEVELS DETERMINED BY THE MAXIMUM VALUE FED IN ON THE DATA INPUT
                    8     ; SWITCHES.  THE MINIMUM LEVEL IS SET AT HALF THE MAXIMUM LEVEL.
                    9     ; THE TIME INTERVAL FOR 4 CONSECUTIVE PHASES AT THE DESIRED FIXED
                   10     ; SPEED IS FED IN ON THE INPUT DATA SWITCHES.  THE CALCULATION OF
                   11     ; ACTUAL TIME INTERVAL - DESIRED TIME INTERVAL IS PERFORMED
                   12     ; FOR 4 PHASES USING 2 REGISTER ARITHMETIC.  IF THE RESULT IS
                   13     ; POSITIVE, THEN THE MOTOR SPEED IS TOO SLOW AND THE CURRENT LEVEL
                   14     ; IS INCREASED.  IF THE RESULT IS NEGATIVE, THEN THE MOTOR SPEED
                   15     ; IS TOO FAST AND THE CURRENT LEVEL IS DECREASED.  ALSO FED IN
                   16     ; ON THE INPUT DATA SWITCHES IS THE NUMBER OF PHASES THAT ARE TO
                   17     ; ELAPSE BETWEEN EVERY SET OF 4 CONSECUTIVE PHASES THAT IS USED
                   18     ; FOR SPEED CORRECTION.  EXTERNAL INTERRUPTS TO INDICATE DETECTION
                   19     ; OF THE NEXT PHASE WILL BE TRIGGERED BY ZERO CROSSINGS OF THE
                   20     ; SENSE WAVEFORMS GOING FROM NEGATIVE TO POSITIVE POLARITY.
                   21     ; TO RUN THE MOTOR:
                   22     ; 1.) ON P73 TO P60 INDICATE THE MAXIMUM CURRENT LEVEL WITH P73
                   23     ; AS THE MSB AND P60 AS THE LSB.  ON P53 TO P40 INDICATE THE
                   24     ; NUMBER OF PHASES THAT ARE TO ELAPSE BETWEEN EVERY SET OF 4
                   25     ; CONSECUTIVE PHASES THAT IS USED FOR SPEED CORRECTION.
                   26     ; PRESS T0.
                   27     ; 2.) ON P73 TO P40 INDICATE THE DESIRED TIME INTERVAL WITH P73
                   28     ; AS THE MSB AND P40 AS THE LSB.
                   29     ; PRESS T1.
                   30     ; 3.) PRESS T0 TO RUN THE MOTOR.
                   31
0000               32           ORG 0
                   33 SYSRST: ; SYSTEM RESET
0000 040B          34           JMP RESET
                   35
0003               36           ORG 3
                   37 EXTINT: ; EXTERNAL INTERRUPT
0003 FB            38           MOV A,R3 ; JUMP TO THE EXTERNAL INTERRUPT ROUTINE WHOSE LOCATION
0004 B3            39           JMPP @A  ; IS STORED AT THE ADDRESS IN R3
                   40
0007               41           ORG 7
                   42 TIMINT: ; TIMER INTERRUPT
0007 35            43           DIS TCNTI
0008 93            44           RETR
                   45
0009 80            46 DATA1:  DB LOW SERVE1
000A A0            47 DATA2:  DB LOW SERVE2
                   48
                   49
000B 2380          50 RESET:  MOV A,#80H ; TURN OFF THE TRANSISTORS, ENABLE THE INPUT SWITCH TO
000D 39            51           OUTL P1,A  ; EXPANDER, AND DISABLE THE LED/DAC TO EXPANDER
000E 3A            52           OUTL P2,A ; DISABLE THE MAIN PROGRAM IO EXPANDER
000F 240F          53 HERE1:  JNTO HERE1
0011 3611          54 HERE2:  JTO HERE2 ; WAIT FOR T0 TO BE PRESSED
```

```
LOC  OBJ       LINE       SOURCE STATEMENT

0013 B820        55         MOV R0,#20H
0015 B930        56         MOV R1,#30H
0017 0F          57         MOVD A,P7
0018 47          58         SWAP A
0019 AA          59         MOV R2,A
001A 0E          60         MOVD A,P6
001B 4A          61         ORL A,R2
001C A0          62         MOV @R0,A ; KEEP THE ORIGINAL MAXIMUM VALUE IN LOCATION 20H
001D A1          63         MOV @R1,A ; KEEP THE UPDATED MAXIMUM VALUE IN LOCATION 30H
001E 97          64         CLR C ; DIVIDE THE MAXIMUM VALUE BY 2
001F 67          65         RRC A ; TO OBTAIN THE MINIMUM VALUE
0020 18          66         INC R0
0021 19          67         INC R1
0022 A0          68         MOV @R0,A ; KEEP THE ORIGINAL MINIMUM VALUE IN LOCATION 21H
0023 A1          69         MOV @R1,A ; KEEP THE UPDATED MINIMUM VALUE IN LOCATION 31H
0024 0D          70         MOVD A,P5
0025 47          71         SWAP A
0026 AA          72         MOV R2,A
0027 0E          73         MOVD A,P6
0028 4A          74         ORL A,R2
0029 18          75         INC R0 ; PUT THE NUMBER OF ELAPSED PHASES IN LOCATION 22H
002A A0          76         MOV @R0,A
002B 462B        77 HERE3:  JNT1 HERE3
002D 5420        78 HERE4:  JT1 HERE4 ; WAIT FOR T1 TO BE PRESSED
002F 0D          79         MOVD A,P5
0030 47          80         SWAP A
0031 AA          81         MOV R2,A
0032 0C          82         MOVD A,P4
0033 4A          83         ORL A,R2
0034 19          84         INC R1 ; MOVE THE 8 LOW BITS OF THE DESIRED TIME TO MEMORY
0035 A1          85         MOV @R1,A ; LOCATION 32H
0036 0F          86         MOVD A,P7
0037 47          87         SWAP A
0038 AA          88         MOV R2,A
0039 0E          89         MOVD A,P6
003A 4A          90         ORL A,R2
003B 19          91         INC R1 ; MOVE THE 8 HIGH BITS OF DESIRED TIME TO MEMORY LOCATION
003C A1          92         MOV @R1,A ; 33H
003D 263D        93 HERE5:  JNT0 HERE5
003F 363F        94 HERE6:  JT0 HERE6 ; WAIT FOR T0 BEFORE RUNNING THE MOTOR
0041 BCF6        95         MOV R4,#-10 ; ALLOW ONLY 10 ATTEMPTS TO START THE MOTOR
0043 B824        96 RESTRT: MOV R0,#24H ; INITIALIZE TO START OF 4 PHASE CYCLE
0045 B011        97         MOV @R0,#11H
0047 BB09        98         MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
                 99                           ; ADDRESS IN R3
0049 B822       100         MOV R0,#22H ; MOVE THE NUMBER OF PHASES TO ELAPSE BEFORE THE FIRST
004B F0         101         MOV A,@R0   ; 4 PHASE COUNTS INTO R2
004C AA         102         MOV R2,A
004D 2340       103         MOV A,#40H ; DISABLE THE INPUT SWITCH TO EXPANDER
004F 39         104         OUTL P1,A  ; AND ENABLE THE LED/DAC TO EXPANDER
0050 230F       105         MOV A,#0FH ; ENABLE THE 2 FIELD DAC LATCHES
0052 3D         106         MOVD P5,A  ; AND THE 2 MAXIMUM DAC LATCHES
0053 27         107         CLR A      ; DISABLE THE 2 MINIMUM
0054 3C         108         MOVD P4,A ; AND 2 BOTTOM DAC LATCHES
0055 B820       109         MOV R0,#20H
```

```
LOC  OBJ          LINE       SOURCE STATEMENT

0057 F0           110         MOV A,@R0 ; MOVE THE INITIAL 4 LOW BITS OUT
0058 3E           111         MOVD P6,A ; TO THE FIELD AND MAXIMUM DACS
0059 47           112         SWAP A    ; MOVE THE INITIAL 4 HIGH BITS OUT
005A 3F           113         MOVD P7,A ; TO THE FIELD AND MAXIMUM DACS
005B 27           114         CLR A     ; DISABLE THE 2 FIELD
005C 3D           115         MOVD P5,A ; AND 2 MAXIMUM DAC LATCHES
005D 230C         116         MOV A,#0CH ; ENABLE THE 2 MINIMUM DAC LATCHES
005F 3C           117         MOVD P4,A
0060 18           118         INC R0
0061 F0           119         MOV A,@R0 ; MOVE THE INITIAL 4 LOW BITS OUT
0062 3E           120         MOVD P6,A ; TO THE MINIMUM DAC
0063 47           121         SWAP A    ; MOVE THE INITIAL 4 HIGH BITS OUT
0064 3F           122         MOVD P7,A ; TO THE MINIMUM DAC
0065 2303         123         MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES
0067 3C           124         MOVD P4,A  ; AND ENABLE THE 2 BOTTOM DAC LATCHES
0068 27           125         CLR A     ; MOVE THE 4 HIGH BITS OUT
0069 3F           126         MOVD P7,A ; TO THE BOTTOM VALUE DAC
006A 2308         127         MOV A,#8  ; MOVE THE 4 LOW BITS OUT
006C 3E           128         MOVD P6,A ; TO THE BOTTOM VALUE DAC
006D 27           129         CLR A ; DISABLE THE 2 BOTTOM DAC LATCHES
006E 3C           130         MOVD P4,A
006F 23C0         131         MOV A,#0C0H ; DISABLE THE LED/DAC IO EXPANDER
0071 39           132         OUTL P1,A
0072 27           133         CLR A ; ENABLE THE MAIN PROGRAM IO EXPANDER
0073 3A           134         OUTL P2,A
0074 246C         135         JMP MREST
                  136
                  137         ; SUBROUTINES
0076 23FF         138 WAIT1T: MOV A,#-1 ; WAIT1T WAITS FOR 1 TIMER UNIT
0078 62           139 WAITTM: MOV T,A ; WAITTM WAITS FOR # OF UNITS OF TIME
0079 55           140         STRT T  ; EQUAL TO THAT IN A
007A 167E         141 WAITTF: JTF WTMR ; WAITTF WAITS FOR TIMER FLAG
007C 047A         142         JMP WAITTF
007E 65           143 WTMR:   STOP TCNT
007F 93           144         RETR
                  145
                  146         ; EXTERNAL INTERRUPT ROUTINES
                  147
0080 FE           148 SERVE1: MOV A,R6
0081 43E0         149         ORL A,#0E0H
0083 39           150         OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
0084 99DF         151         ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                  152
0086 27           153         CLR A
0087 62           154         MOV T,A
0088 55           155         STRT T ; START THE TIMER FROM 0
0089 B835         156         MOV R0,#35H ; CLEAR THE 8 HIGH BITS
008B A0           157         MOV @R0,A ; OF ACTUAL TIME IN 35H
008C BD14         158         MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                  159
008E FE           160         MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
008F AF           161         MOV R7,A
0090 47           162         SWAP A
0091 4F           163         ORL A,R6
0092 F7           164         RL A
```

```
LOC  OBJ        LINE     SOURCE STATEMENT

0093 530F        165        ANL A,#0FH
0095 AE          166        MOV R6,A
                 167
0096 3F          168        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 169
0097 BCF6        170        MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                 171




0099 CA          172        DEC R2
009A FA          173        MOV A,R2
009B 969F        174        JNZ FINIS
009D B80A        175        MOV R3,#LOW DATA2 ; STORE THE LOCATION CONTAINING THE ADDRESS OF
                 176                          ; THE INTERRUPT THAT COUNTS FOR 4 PHASES IN R3
009F 93          177 FINIS: RETR
                 178
00A0 65          179 SERVE2: STOP TCNT ; STOP THE TIMER COUNT
                 180
00A1 FE          181        MOV A,R6
00A2 43E0        182        ORL A,#0E0H
00A4 39          183        OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
00A5 990F        184        ANL P1,#00FH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                 185
00A7 42          186        MOV A,T
00A8 AA          187        MOV R2,A ; PUT THE TIMER COUNT IN R2
00A9 27          188        CLR A
00AA 62          189        MOV T,A
00AB 55          190        STRT T ; START THE TIMER FROM 0
00AC BD14        191        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                 192
00AE FE          193        MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
00AF AF          194        MOV R7,A
00B0 47          195        SWAP A
00B1 4F          196        ORL A,R6
00B2 E7          197        RL A
00B3 530F        198        ANL A,#0FH
00B5 AF          199        MOV R6,A
                 200
00B6 3F          201        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 202
00B7 BCFF        203        MOV R4,#-10 ; IF AN UNDERSPEED OCCURS ALLOW 10 RESTART ATTEMPTS
                 204
00B9 B824        205        MOV R0,#24H
00BB F0          206        MOV A,@R0
00BC E7          207        RL A
00BD A0          208        MOV @R0,A ; UPDATE POSITION INDEX IN 4 PHASE CYCLE
00BE 12D1        209        JB0 CURCHK ; HAVE COMPLETED A 4 PHASE CYCLE
00C0 32CC        210        JB1 STRTCT ; JUST STARTING A CYCLE
00C2 B834        211        MOV R0,#34H ; 34H CONTAINS THE 8 LOW BITS OF ACTUAL TIME
                 212                    ; 35H CONTAINS THE 8 HIGH BITS OF ACTUAL TIME
00C4 FA          213        MOV A,R2
00C5 60          214        ADD A,@R0 ; UPDATE THE 8 LOW BITS OF ACTUAL TIME IN 34H
00C6 A0          215        MOV @R0,A
00C7 E6CB        216        JNC LATER1
00C9 18          217        INC R0
00CA 10          218        INC @R0 ; UPDATE THE 8 HIGH BITS OF ACTUAL TIME IN 35H
00CB 93          219 LATER1: RETR
```

```
LOC  OBJ       LINE       SOURCE STATEMENT

00CC B834        220 STRTCT: MOV R0,#34H
00CE FA          221         MOV A,R2 ; PUT THE LOW 8 BITS OF
00CF A0          222         MOV @R0,A ; ACTUAL TIME IN 34H
00D0 93          223         RETR
00D1 B834        224 CURCHK: MOV R0,#34H
00D3 FA          225         MOV A,R2 ; PLACE THE UPDATED 8 LOW BITS OF
00D4 60          226         ADD A,@R0 ; ACTUAL TIME IN THE ACCUMULATOR
00D5 E609        227         JNC SKIPIN
00D7 18          228         INC R0 ; UPDATE THE 8 HIGH BITS OF ACTUAL TIME IN 35H
00D8 10          229         INC @R0
00D9 B832        230 SKIPIN: MOV R0,#32H
00DB 37          231         CPL A
00DC A0          232         ADD A,@R0
00DD 37          233         CPL A
00DE AA          234         MOV R2,A ; PUT THE LOW 8 BITS OF ACTUAL TIME - THE LOW 8 BITS OF
                 235                  ; DESIRED TIME IN R2
00DF B835        236         MOV R0,#35H



00E1 F0          237         MOV A,@R0 ; PUT THE 8 HIGH BITS OF ACTUAL TIME IN THE ACCUMULATOR
00E2 B000        238         MOV @R0,#0 ; CLEAR THE 8 HIGH BITS OF ACTUAL TIME IN LOCATION
                 239                    ; 35H FOR THE NEXT CYCLE
00E4 B833        240         MOV R0,#33H
00E6 37          241         CPL A
00E7 70          242         ADDC A,@R0
00E8 37          243         CPL A ; HIGH 8 BITS OF ACTUAL TIME - HIGH 8 BITS OF DESIRED TIME
00E9 2400        244         JMP NPAGE
0100             245         ORG 100H
0100 F23F        246 NPAGE:  JB7 CURDCR ; ACTUAL TIME IS LESS THAN DESIRED TIME
                 247                    ; ACTUAL SPEED IS GREATER THAN DESIRED SPEED
                 248                    ; DECREASE THE CURRENT
0102 960E        249         JNZ CURINC ; FOR ANY OTHER NONZERO NUMBER INCREASE THE CURRENT
0104 FA          250         MOV A,R2
0105 960E        251         JNZ CURINC
0107 B822        252         MOV R0,#22H ; MOVE THE NUMBER OF PHASES TO ELAPSE BEFORE THE
0109 F0          253         MOV A,@R0   ; NEXT 4 PHASE COUNT INTO R2
010A AA          254         MOV R2,A
010B BB09        255         MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
                 256                           ; ADDRESS IN R3
010D 93          257         RETR ; IF ACTUAL TIME = DESIRED TIME, THEN LEAVE THE CURRENT UNCHANGED
010E 8A80        258 CURINC: ORL P2,#80H ; DISABLE THE MAIN PROGRAM IO EXPANDER
0110 997F        259         ANL P1,#7FH ; ENABLE THE LED/DAC IO EXPANDER
0112 27          260         CLR A ; DISABLE THE FIELD AND MAXIMUM DAC LATCHES
0113 3D          261         MOVD P5,A
0114 230C        262         MOV A,#0CH ; DISABLE THE BOTTOM DAC LATCHES
0116 3C          263         MOVD P4,A  ; AND ENABLE THE MINIMUM DAC LATCHES
0117 B830        264         MOV R0,#30H
0119 10          265         INC @R0 ; INCREASE THE MAXIMUM CURRENT VALUE BY 1 DAC LEVEL
011A F0          266         MOV A,@R0
011B 9620        267         JNZ NOTHI
011D B0FF        268         MOV @R0,#0FFH ; UNLESS THE MAXIMUM VALUE ALREADY IS AS
011F F0          269         MOV A,@R0     ; LARGE AS POSSIBLE
0120 97          270 NOTHI:  CLR C ; DIVIDE THE MAXIMUM VALUE BY 2
0121 67          271         RRC A ; TO OBTAIN THE MINIMUM VALUE
0122 18          272         INC R0
0123 A0          273         MOV @R0,A ; PUT THE UPDATED MINIMUM VALUE IN LOCATION 31H
0124 3E          274         MOVD P6,A ; MOVE THE 4 UPDATED LOW BITS OUT TO THE MINIMUM DAC
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

0125 47          275        SWAP A
0126 3F          276        MOVD P7,A ; MOVE THE 4 UPDATED HIGH BITS OUT TO THE MINIMUM DAC
0127 27          277        CLR A ; DISABLE THE 2 MINIMUM DAC LATCHES
0128 3C          278        MOVD P4,A
0129 2303        279        MOV A,#03H ; ENABLE THE 2 MAXIMUM DAC LATCHES
012B 3D          280        MOVD P5,A
012C B830        281        MOV R0,#30H
012E F0          282        MOV A,@R0 ; MOVE THE UPDATED 4 LOW BITS OUT
012F 3E          283        MOVD P6,A ; TO THE MAXIMUM DAC
0130 47          284        SWAP A    ; MOVE THE UPDATED 4 HIGH BITS OUT
0131 3F          285        MOVD P7,A ; TO THE MAXIMUM DAC
0132 27          286        CLR A ; DISABLE THE 2 MAXIMUM DAC LATCHES
0133 3D          287        MOVD P5,A
0134 8980        288        ORL P1,#80H ; DISABLE THE LED/DAC IO EXPANDER
0136 9A7F        289        ANL P2,#7FH ; ENABLE THE MAIN PROGRAM IO EXPANDER
0138 B822        290        MOV R0,#22H ; MOVE THE NUMBER OF PHASES TO ELAPSE BEFORE THE
013A F0          291        MOV A,@R0   ; NEXT 4 PHASE COUNT INTO R2
013B AA          292        MOV R2,A
013C BB09        293        MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
                 294                          ; ADDRESS IN R3
013E 93          295        RETR
013F 8A80        296 CURDCR: ORL P2,#80H ; DISABLE THE MAIN PROGRAM IO EXPANDER
0141 997F        297        ANL P1,#7FH ; ENABLE THE LED/DAC IO EXPANDER
0143 27          298        CLR A ; DISABLE THE FIELD AND MAXIMUM DAC LATCHES
0144 3D          299        MOVD P5,A
0145 230C        300        MOV A,#0CH ; DISABLE THE BOTTOM DAC LATCHES
0147 3C          301        MOVD P4,A  ; AND ENABLE THE MINIMUM DAC LATCHES



0148 B830        302        MOV R0,#30H
014A F0          303        MOV A,@R0
014B 07          304        DEC A
014C A0          305        MOV @R0,A ; DECREASE THE MAXIMUM CURRENT VALUE BY 1 DAC LEVEL
014D 97          306        CLR C ; DIVIDE THE UPDATED MAXIMUM VALUE BY 2
014E 67          307        RRC A ; TO OBTAIN THE UPDATED MINIMUM VALUE
014F 18          308        INC R0
0150 A0          309        MOV @R0,A ; PUT THE UPDATED MINIMUM VALUE IN LOCATION 31H
0151 3E          310        MOVD P6,A ; MOVE THE UPDATED 4 LOW BITS OUT TO THE MINIMUM DAC
0152 47          311        SWAP A    ; MOVE THE UPDATED 4 HIGH BITS
0153 3F          312        MOVD P7,A ; OUT TO THE MINIMUM DAC
0154 27          313        CLR A ; DISABLE THE 2 MINIMUM DAC LATCHES
0155 3C          314        MOVD P4,A
0156 2303        315        MOV A,#03H ; ENABLE THE 2 MAXIMUM DAC LATCHES
0158 3D          316        MOVD P5,A
0159 B830        317        MOV R0,#30H
015B F0          318        MOV A,@R0 ; MOVE THE UPDATED 4 LOW BITS
015C 3E          319        MOVD P6,A ; OUT TO THE MAXIMUM DAC
015D 47          320        SWAP A    ; MOVE THE UPDATED 4 HIGH BITS
015E 3F          321        MOVD P7,A ; OUT TO THE MAXIMUM DAC
015F 27          322        CLR A ; DISABLE THE 2 MAXIMUM DAC LATCHES
0160 3D          323        MOVD P5,A
0161 8980        324        ORL P1,#80H ; DISABLE THE LED/DAC IO EXPANDER
0163 9A7F        325        ANL P2,#7FH ; ENABLE THE MAIN PROGRAM IO EXPANDER
0165 B822        326        MOV R0,#22H ; MOVE THE NUMBER OF PHASES TO ELAPSE BEFORE THE
0167 F0          327        MOV A,@R0   ; NEXT 4 PHASE COUNT INTO R2
0168 AA          328        MOV R2,A
0169 BB09        329        MOV R3,#LOW DATA1 ; STORE THE LOCATION CONTAINING THE FIRST INTERRUPT
```

```
LOC  OBJ        LINE       SOURCE STATEMENT

                  330                          ; ADDRESS IN R3
016B 93           331        RETR
                  332
016C 23EE         333 MREST: MOV A,#-18 ; START DELAYED FOR 18 UNITS
016E 1478         334        CALL WAITTM
                  335
0170 8910         336        ORL P1,#10H ; SWITCH ON THE FIELD TRANSISTOR UNTIL THE FIELD
                  337                    ; CURRENT REACHES THE VALUE DETERMINED BY THE FIELD DAC
0172 0E           338 FON:   MOVD A,P6 ; CHECK FIELD CURRENT LEVEL
0173 A9           339        MOV R1,A
0174 0E           340        MOVD A,P6 ; CONFIRM WITH A DELAYED CHECK
0175 59           341        ANL A,R1
0176 127A         342        JB0 MSEN
0178 2472         343        JMP FON
                  344
017A 0C           345 MSFN:  MOVD A,P4 ; INITIAL POSITION SENSE, SIGNAL IS DUE TO D(IF)/D(T)
017B 1281         346        JB0 SOBC
017D 7285         347 SOAD:  JB3 SA0 ; SENSED CD MEANING POSITION A
017F 2497         348        JMP SD0 ; SENSED BC MEANING POSITION D
0181 3291         349 SOBC:  JB1 SC0 ; SENSED AB MEANING POSITION C
0183 248B         350        JMP SB0 ; SENSED DA MEANING POSITION B
                  351
                  352        ; DEFINE R7 AS CURRENT PHASE SWITCH, R6 AS NEXT PHASE SWITCH
0185 BF01         353 SA0:   MOV R7,#01H
0187 BE02         354        MOV R6,#02H
0189 249B         355        JMP S1
018B BF02         356 SB0:   MOV R7,#02H
018D BE04         357        MOV R6,#04H
018F 249B         358        JMP S1
0191 BF04         359 SC0:   MOV R7,#04H
0193 BE08         360        MOV R6,#08H
0195 249B         361        JMP S1
0197 BF08         362 SD0:   MOV R7,#08H
0199 BE01         363        MOV R6,#01H
                  364
                  365        ; STAGE 1
                  366
019B 2304         367 S1:    MOV A,#04H ; SELECT SENSE INTERRUPT
019D 3E           368        MOVD P6,A
019E FF           369        MOV A,R6
019F 3F           370        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
01A0 FF           371        MOV A,R7
01A1 43E0         372        ORL A,#0E0H
01A3 39           373        OUTL P1,A ; TURN OFF THE FIELD AND TURN ON THE SELECTED PHASE
01A4 990F         374        ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC CURRENT REGULATION
                  375
01A6 27           376        CLR A
01A7 62           377        MOV T,A
01A8 B835         378        MOV R0,#35H ; CLEAR THE 8 HIGH BITS OF ACTUAL TIME IN 35H
01AA A0           379        MOV @R0,A   ; FOR THE FIRST CYCLE
01AB 55           380        STRT T ; START THE TIMER FROM 0
01AC BD14         381        MOV R5,#20 ; NUMBER OF TIMER OVERFLOWS BEFORE A RESTART IS ENTERED
                  382
01AE 05           383        EN I ; ENABLE THE EXTERNAL INTERRUPT
                  384
```

```
  LOC  OBJ        LINE       SOURCE STATEMENT

  01AF 16B3        385 WORKL1: JTF MORET1
  01B1 24AF        386         JMP WORKL1
  01B3 B835        387 MORET1: MOV R0,#35H
  01B5 10          388         INC @R0 ; INCREMENT THE 8 HIGH BITS OF ACTUAL TIME IN LOCATION
                   389                 ; 35H
  01B6 FDAF        390         DJNZ R5,WORKL1
  01B8 15          391         DIS I
  01B9 65          392         STOP TCNT
  01BA 2340        393         MOV A,#40H ; TURN OFF THE TRANSISTOR
  01BC 39          394         OUTL P1,A  ; AND ENABLE THE LED/DAC IO EXPANDER
  01BD 8A80        395         ORL P2,#80H ; DISABLE THE MAIN PROGRAM IO EXPANDER
  01BF B820        396         MOV R0,#20H ; PUT THE ORIGINAL MAXIMUM VALUE CONTAINED IN LOCATION
  01C1 F0          397         MOV A,@R0   ; 20H
  01C2 B930        398         MOV R1,#30H ; INTO LOCATION 30H CONTAINING THE UPDATED MAXIMUM
  01C4 A1          399         MOV @R1,A   ; VALUE
  01C5 18          400         INC R0      ; PUT THE ORIGINAL MINIMUM VALUE CONTAINED IN LOCATION
  01C6 F0          401         MOV A,@R0   ; 21H
  01C7 19          402         INC R1      ; INTO LOCATION 31H CONTAINING THE UPDATED MINIMUM
  01C8 A1          403         MOV @R1,A   ; VALUE
  01C9 1C          404         INC R4 ; IF 10 TRIALS ARE UP THEN STOP TILL TO IS PRESSED AGAIN
  01CA FC          405         MOV A,R4
  01CB CACF        406         JZ AGAIN
  01CD 0443        407         JMP RESTRT
  01CF 043D        408 AGAIN:  JMP HERE5
                   409
                   410         END
```

USER SYMBOLS

AGAIN	01CF	CURCHK	00D1	CURDCR	013F	CURINC	010E	DATA1	0009	DATA2	000A	EXTINT	0003	FINIS	009F
FON	0172	HERE1	000F	HERE2	0011	HERE3	0028	HERE4	0020	HERE5	003D	HERE6	003F	LATER1	00CB
MORET1	01B3	MRFST	01AC	MSEN	017A	NOTHI	0120	NPAGE	0100	RESET	000B	RESTRT	0043	S0AD	017D
S0BC	0181	S1	019B	SA0	0185	SB0	018B	SC0	0191	SD0	0197	SERVE1	0080	SERVE2	00A0
SKIPIN	00D9	STRTCT	00CC	SYSRST	0000	TIMINT	0007	WAIT1T	0076	WAITTF	007A	WAITTM	0078	WORKL1	01AF
WTMR	007E														

ASSEMBLY COMPLETE, NO ERRORS

```
ASM48 SUMACC.MAC DEBUG XREF TITLE('SUMMARY ACCELERATION PROFILE')


ISIS-II MCS-48/UPI-41 MACRO ASSEMBLER, V3.0            PAGE    1
SUMMARY ACCELERATION PROFILE

  LOC  OBJ          LINE        SOURCE STATEMENT

                      1         ; PROGRAM FOR ORTAINING A SUMMARY ACCELERATION PROFILE
                      2         ; INITIALLY THE FIELD TRANSISTOR IS TURNED ON AND THE FIELD CURRENT
                      3         ; IS ALLOWED TO RISE UNTIL IT REACHES A LEVEL DETERMINED BY THE
                      4         ; FIELD DAC VALUE FED IN ON THE INPUT DATA SWITCHES.  PHASE
                      5         ; CURRENT WILL BE KEPT BETWEEN MAXIMUM AND MINIMUM LEVELS DETERMINED
                      6         ; BY THE MAXIMUM AND MINIMUM DAC VALUES FED IN ON THE INPUT DATA
                      7         ; SWITCHES. PRESSING THE T1 BUTTON WILL START THE MOTOR. EXTERNAL
                      8         ; INTERRUPTS TO INDICATE DETECTION OF THE NEXT PHASE WILL BE TRIGGERED
                      9         ; BY ZERO CROSSINGS OF THE SENSE WAVEFORMS GOING FROM NEGATIVE TO
                     10         ; POSITIVE POLARITY.
                     11         ; TO RUN THE MOTOR:
                     12         ; 1.) ON P73 TO P60. THE TOP SET OF DATA INPUT SWITCHES. INDICATE
                     13         ; THE FIELD DAC VALUE WITH P73 AS THE MSB AND P60 AS THE LSB.  ON P53
                     14         ; TO P40. THE BOTTOM SET OF DATA INPUT SWITCHES. INDICATE THE FIRST
                     15         ; VALUE TO BE APPLIED TO THE DAC ADDRESSING THE MAXIMUM VALUE(TOP)
                     16         ; COMPARATORS WITH P53 AS THE MSB AND P40 AS THE LSB.
                     17         ; PRESS T0.
                     18         ; 2.) ON P73 TO P60 INDICATE THE FIRST VALUE TO BE APPLIED TO THE
                     19         ; DAC ADDRESSING THE MINIMUM VALUE(MIDDLE) COMPARATORS.  ON P53 TO
                     20         ; P40 INDICATE THE FIRST VALUE TO BE APPLIED TO THE DAC ADDRESSING
                     21         ; THE BOTTOM COMPARATORS.
                     22         ; PRESS T1.
                     23         ; 3.) ON P73 TO P60 INDICATE THE NUMBER OF CONSECUTIVE PHASES
                     24         ; FOR WHICH A TIME DURATION IS TAKEN.
                     25         ; PRESS T0.
                     26         ; 4.) PRESS T1 TO START THE MOTOR.  FOR 512 SETS OF THE DESIGNATED
                     27         ; NUMBER OF CONSECUTIVE PHASES, TIME COUNTS WILL BE STORED IN
                     28         ; THE 8185-2 RAM.  THEN ALL THE TRANSISTORS WILL BE TURNED OFF.
                     29         ; 5.) ALTERNATELY PRESS T0 AND T1 TO READ OUT THE TIME DURATIONS
                     30         ; IN THE LEDS.
                     31
  0000               32                 ORG 0
                     33 SYSRST:  ; SYSTEM RESET
  0000 0408          34                 JMP RESET
                     35
  0003               36                 ORG 3
                     37 EXTINT:  ; EXTERNAL INTERRUPT
  0003 047D          38                 JMP SERVIC ; JUMP TO THE EXTERNAL INTERRUPT ROUTINE
                     39
  0007               40                 ORG 7
                     41 TIMINT:  ; TIMER INTERRUPT
  0007 1D            42                 INC R5 ; WHEN THE LOWER 3 TIME BITS OVERFLOW THEN INCREMENT
                     43                        ; THE 8 UPPER TIME BITS
  0008 160A          44                 JTF CLRTF
  000A 93            45 CLRTF:          RETR
                     46
  000B 2380          47 RESET:          MOV A,#80H ; TURN OFF THE TRANSISTORS, ENABLE THE INPUT SWITCH
  000D 39            48                 OUTL P1,A  ; SWITCH IO EXPANDER, AND DISABLE THE LED/DAC IO EXPANDER
  000E 3A            49                 OUTL P2,A  ; DISABLE THE MAIN PROGRAM IO EXPANDER
  000F 2A0F          50 HERE1:          JNT0 HERE1
  0011 3611          51 HERE2:          JT0 HERE2 ; WAIT FOR T0 TO BE PRESSED
  0013 B820          52                 MOV R0,#20H




  0015 0F            53                 MOVD A,P7  ; MOVE THE 4 HIGH BITS OF THE
  0016 A0            54                 MOV @R0,A  ; FIELD VALUE TO MEMORY LOCATION 20H

                                                                                    142
```

SUMMARY ACCELERATION PROFILE

```
LOC  OBJ          LINE        SOURCE STATEMENT

0017 18             55          INC R0
0018 0E             56          MOVD A,P6 ; MOVE THE 4 LOW BITS OF THE
0019 A0             57          MOV @R0,A ; FIELD VALUE VALUE TO DATA MEMORY LOCATION 21H
001A 18             58          INC R0
001B 0D             59          MOVD A,P5 ; MOVE THE 4 HIGH BITS OF THE FIRST
001C A0             60          MOV @R0,A ; MAXIMUM VALUE TO DATA MEMORY LOCATION 22H
001D 18             61          INC R0
001E 0C             62          MOVD A,P4 ; MOVE THE 4 LOW BITS OF THE FIRST
001F A0             63          MOV @R0,A ; MAXIMUM VALUE TO DATA MEMORY LOCATION 23H
0020 4620           64 HERE3:   JNT1 HERE3
0022 5A22           65 HERE4:   JT1 HERE4 ; WAIT FOR T1 TO BE PRESSED
0024 18             66          INC R0
0025 0F             67          MOVD A,P7 ; MOVE THE 4 HIGH BITS OF THE FIRST
0026 A0             68          MOV @R0,A ; MINIMUM VALUE TO DATA MEMORY LOCATION 24H
0027 18             69          INC R0
0028 0E             70          MOVD A,P6 ; MOVE THE 4 LOW BITS OF THE FIRST
0029 A0             71          MOV @R0,A ; MINIMUM VALUE TO DATA MEMORY LOCATION 25H
002A 18             72          INC R0
002B 0D             73          MOVD A,P5 ; MOVE THE 4 HIGH BITS OF THE FIRST
002C A0             74          MOV @R0,A ; BOTTOM VALUE TO DATA MEMORY LOCATION 26H
002D 18             75          INC R0
002E 0C             76          MOVD A,P4 ; MOVE THE 4 LOW BITS OF THE FIRST
002F A0             77          MOV @R0,A ; BOTTOM VALUE TO DATA MEMORY LOCATION 27H
0030 2630           78 HERE5:   JNT0 HERE5
0032 3632           79 HERE6:   JT0 HERE6 ; WAIT FOR T0 TO BE PRESSED
0034 0F             80          MOVD A,P7
0035 47             81          SWAP A
0036 AA             82          MOV R2,A
0037 0E             83          MOVD A,P6
0038 4A             84          ORL A,R2
0039 AB             85          MOV R3,A ; PUT THE NUMBER OF CONSECUTIVE PHASES FOR WHICH A
                    86                   ; TIME DURATION IS TAKEN IN R3
003A 4A3A           87 HERE9:   JNT1 HERE9
003C 563C           88 HERE10:  JT1 HERE10 ; WAIT FOR T1 BEFORE THE MOTOR STARTS
003E 2340           89          MOV A,#40H ; DISABLE THE INPUT SWITCH TO EXPANDER
0040 39             90          OUTL P1,A ; AND ENABLE THE LED/DAC TO EXPANDER
0041 230C           91          MOV A,#0CH ; DISABLE THE 2 MAXIMUM DAC LATCHES
0043 3D             92          MOVD P5,A ; AND ENABLE THE 2 FIELD DAC LATCHES
0044 27             93          CLR A ; DISABLE THE 2 MINIMUM
0045 3C             94          MOVD P4,A ; AND 2 BOTTOM DAC LATCHES
0046 B820           95          MOV R0,#20H
0048 F0             96          MOV A,@R0 ; MOVE THE 4 HIGH BITS
0049 3F             97          MOVD P7,A ; OUT TO THE FIELD DAC
004A 18             98          INC R0
004B F0             99          MOV A,@R0 ; MOVE THE 4 LOW BITS
004C 3E            100          MOVD P6,A ; OUT TO THE FIELD DAC
004D 2303          101          MOV A,#03H ; DISABLE THE 2 FIELD DAC LATCHES
004F 3D            102          MOVD P5,A  ; AND ENABLE THE 2 MAXIMUM DAC LATCHES
0050 18            103          INC R0
0051 F0            104          MOV A,@R0 ; MOVE THE FIRST 4 HIGH BITS
0052 3F            105          MOVD P7,A ; OUT TO THE MAXIMUM VALUE DAC
0053 18            106          INC R0
0054 F0            107          MOV A,@R0 ; MOVE THE FIRST 4 LOW BITS
0055 3E            108          MOVD P6,A ; OUT TO THE MAXIMUM VALUE DAC
0056 27            109          CLR A ; DISABLE THE 2 MAXIMUM DAC LATCHES
```

```
LOC  OBJ          LINE       SOURCE STATEMENT

0057 3D           110        MOVD P5,A
0058 230C         111        MOV A,#0CH ; ENABLE THE 2 MINIMUM DAC LATCHES
005A 3C           112        MOVD P4,A
005B 18           113        INC R0
005C F0           114        MOV A,@R0 ; MOVE THE FIRST 4 HIGH BITS
005D 3F           115        MOVD P7,A ; OUT TO THE MINIMUM VALUE DAC
005E 18           116        INC R0
005F F0           117        MOV A,@R0 ; MOVE THE FIRST 4 LOW BITS
0060 3E           118        MOVD P6,A ; OUT TO THE MINIMUM VALUE DAC
0061 2303         119        MOV A,#03H ; DISABLE THE 2 MINIMUM DAC LATCHES
0063 3C           120        MOVD P4,A  ; AND ENABLE THE 2 BOTTOM DAC LATCHES
0064 18           121        INC R0
0065 F0           122        MOV A,@R0 ; MOVE THE FIRST 4 HIGH BITS
0066 3F           123        MOVD P7,A ; OUT TO THE BOTTOM VALUE DAC
0067 18           124        INC R0
0068 F0           125        MOV A,@R0 ; MOVE THE FIRST 4 LOW BITS
0069 3E           126        MOVD P6,A ; OUT TO THE BOTTOM VALUE DAC
006A 27           127        CLR A ; DISABLE THE 2 BOTTOM DAC LATCHES
006B 3C           128        MOVD P4,A
006C 23C0         129        MOV A,#0C0H ; DISABLE THE LED/DAC TO EXPANDER
006E 39           130        OUTL P1,A
006F 27           131        CLR A ; ENABLE THE MAIN PROGRAM TO EXPANDER
0070 3A           132        OUTL P2,A
0071 040F         133        JMP MREST
                  134
                  135        ; SUBROUTINES
0073 23FF         136 WAITIT: MOV A,#-1 ; WAITIT WAITS FOR 1 TIMER INIT
0075 62           137 WAITTM: MOV T,A ; WAITTM WAITS FOR # OF INITS OF TIME
0076 55           138        STRT T  ; EQUAL TO THAT IN A
0077 167B         139 WAITTF: JTF WTMR ; WAITTF WAITS FOR TIMER FLAG
0079 0477         140        JMP WAITTF
007B A5           141 WTMR:   STOP TCNT
007C 93           142        RETR
                  143
                  144        ; EXTERNAL INTERRUPT ROUTINES
007D FE           145 SERVIC: MOV A,R6
007F 43E0         146        ORL A,#0E0H
0080 39           147        OUTL P1,A ; TURN OFF THE OLD PHASE AND TURN ON THE NEW PHASE
0081 990F         148        ANL P1,#0DFH ; TURN OFF THE PROHIBITION OF DAC REGULATION
                  149
0083 FCA2         150        DJNZ R4,UPDATE ; CONTINUE COUNTING IF THE REQUIRED NUMBER OF
                  151                       ; CONSECUTIVE PHASES HAVE NOT ELAPSED
0085 65           152        STOP TCNT
0086 168A         153        JTF NEXT1 ; IF A TIMER OVERFLOW HAS OCCURRED, THEN INCREMENT THE
                  154                  ; UPPER 8 BIT COUNT
0088 0480         155        JMP NEXT2
008A 1D           156 NEXT1:  INC R5
008B 35           157        DIS TCNTI ; REMOVE ANY PENDING TIMER INTERRUPT
008C 25           158        EN TCNTI ; THEN RESTORE THE TIMER INTERRUPT
008D 42           159 NEXT2:  MOV A,T ; STORE THE TIMER COUNT IN AN EVEN EXTERNAL RAM LOCATION
008E 90           160        MOVX @R0,A
008F 27           161        CLR A
0090 62           162        MOV T,A
0091 55           163        STRT T ; START THE TIMER FROM 0
0092 18           164        INC R0
```

```
LOC  OBJ       LINE     SOURCE STATEMENT


0093 FD          165        MOV A,R5 ; STORE THE UPPER 8 BIT COUNT IN AN ODD EXTERNAL RAM
0094 90          166        MOVX @R0,A ; LOCATION
0095 BD00        167        MOV R5,#0 ; RESTORE THE UPPER 8 TIME COUNT BITS TO 0
0097 FB          168        MOV A,R3 ; MOVE THE NUMBER OF CONSECUTIVE PHASES FOR WHICH A
0098 AC          169        MOV R4,A ; TIME DURATION IS TAKEN IN R3 INTO R4
0099 18          170        INC R0
009A F8          171        MOV A,R0 ; CHANGE THE A9A8 ADDRESS BITS TO THE EXTERNAL RAM FROM
009B 96A2        172        JNZ UPDATE ; 00 TO 01 TO 10 TO 11 WHEN THE 8 LOWER ADDRESS BITS
                 173                   ; BECOME 0
009D 19          174        INC R1
009E F1          175        MOV A,@R1
009F CAAC        176        JZ LASTPH ; WHEN A9A8 IS READY TO RETURN TO 00 AGAIN THE EXTERNAL
                 177                  ; RAM LOCATIONS ARE ALL FULL AND THE TRANSISTORS WILL
                 178                  ; BE TURNED OFF
00A1 3A          179        OUTL P2,A
                 180
00A2 FE          181 UPDATE: MOV A,R6 ; UPDATE PHASE REGISTERS R7,R6
00A3 AF          182        MOV R7,A
00A4 47          183        SWAP A
00A5 4F          184        ORL A,R6
00A6 F7          185        RL A
00A7 530F        186        ANL A,#0FH
00A9 AE          187        MOV R6,A
                 188
00AA 3F          189        MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
                 190
00AB 93          191        RETR
                 192
00AC 2380        193 LASTPH: MOV A,#80H ; DISABLE THE MAIN PROGRAM TO EXPANDER AND INITIALIZE
00AE 3A          194        OUTL P2,A ; THE EXTERNAL RAM A9A8 ADDRESS BITS TO 00
00AF 2340        195        MOV A,#40H ; TURN OFF THE PHASE TRANSISTOR
00B1 39          196        OUTL P1,A ; AND ENABLE THE LED/DAC TO EXPANDER
00B2 26B2        197 HERE15: JNTO HERE15 ; THEN ALTERNATELY PRESS TO AND T1 TO OUTPUT THE
00B4 36B4        198 HERE16: JTO HERE16 ; TIMER COUNT TO THE LOWER LEDS AND THE R5 COUNT TO
00B6 80          199        MOVX A,@R0 ; THE UPPER LEDS
00B7 3C          200        MOVD P4,A
00B8 47          201        SWAP A
00B9 3D          202        MOVD P5,A
00BA 18          203        INC R0
00BB 80          204        MOVX A,@R0
00BC 3E          205        MOVD P6,A
00BD 47          206        SWAP A
00BE 3F          207        MOVD P7,A
00BF 18          208        INC R0
00C0 4AC0        209 HERE17: JNT1 HERE17
00C2 5AC2        210 HERE18: JT1 HERE18
00C4 80          211        MOVX A,@R0
00C5 3C          212        MOVD P4,A
00C6 47          213        SWAP A
00C7 3D          214        MOVD P5,A
00C8 18          215        INC R0
00C9 80          216        MOVX A,@R0
00CA 3E          217        MOVD P6,A
00CB 47          218        SWAP A
00CC 3F          219        MOVD P7,A
```

SUMMARY ACCELERATION PROFILE

```
LOC  OBJ        LINE       SOURCE STATEMENT

00CD 18          220        INC R0
00CE F8          221        MOV A,R0 ; SEQUENCE A9A8 FROM 00 TO 01 TO 10 TO 11 TO 00
00CF 96B2        222        JNZ HERE15
00D1 19          223        INC R1
00D2 F1          224        MOV A,@R1
00D3 C6D8        225        JZ LASTCT
00D5 3A          226        OUTL P2,A
00D6 04B2        227        JMP HERE15
00D8 2380        228 LASTCT: MOV A,#80H



00DA 3A          229        OUTL P2,A
00DB B944        230        MOV R1,#44H
00DD 04B2        231        JMP HERE15
                 232
00DF 23EE        233 MREST:  MOV A,#-18 ; START DELAYED FOR 18 UNITS
00E1 1475        234        CALL WAITTM
                 235
00E3 8910        236        ORL P1,#10H ; SWITCH ON THE FIELD TRANSISTOR UNTIL THE FIELD
                 237                    ; CURRENT REACHES THE VALUE DETERMINED BY THE FIELD DAC
00E5 0E          238 FON:    MOVD A,P6 ; CHECK FIELD CURRENT LEVEL
00E6 A9          239        MOV R1,A
00E7 0E          240        MOVD A,P6 ; CONFIRM WITH A DELAYED CHECK
00E8 59          241        ANL A,R1
00E9 12ED        242        JB0 MSEN
00EB 04E5        243        JMP FON
                 244
00ED 0C          245 MSEN:   MOVD A,P4 ; INITIAL POSITION SENSE, SIGNAL IS DUE TO D(IF)/D(T)
00EE 2400        246        JMP NPAGE
0100             247        ORG 100H
0100 1206        248 NPAGE:  JB0 S0BC
0102 720A        249 S0AD:   JB3 SA0 ; SENSED CD MEANING POSITION A
0104 241C        250        JMP SD0 ; SENSED BC MEANING POSITION D
0106 3216        251 S0BC:   JB1 SC0 ; SENSED AB MEANING POSITION C
0108 2410        252        JMP SB0 ; SENSED DA MEANING POSITION B
                 253
                 254        ; DEFINE R7 AS CURRENT PHASE SWITCH, R6 AS NEXT PHASE SWITCH
010A BF01        255 SA0:    MOV R7,#01H
010C BF02        256        MOV R6,#02H
010E 2420        257        JMP S1
0110 BF02        258 SB0:    MOV R7,#02H
0112 BF04        259        MOV R6,#04H
0114 2420        260        JMP S1
0116 BF04        261 SC0:    MOV R7,#04H
0118 BF08        262        MOV R6,#08H
011A 2420        263        JMP S1
011C BF08        264 SD0:    MOV R7,#08H
011E BF01        265        MOV R6,#01H
                 266
                 267        ; STAGE 1
                 268
0120 2304        269 S1:     MOV A,#04H ; SELECT SENSE INTERRUPT
0122 3E          270        MOVD P6,A
0123 FB          271        MOV A,R3 ; MOVE THE NUMBER OF CONSECUTIVE PHASES FOR WHICH A
0124 AC          272        MOV R4,A ; TIME DURATION IS TAKEN IN R3 INTO R4
                 273
0125 B941        274        MOV R1,#41H ; USE LOCATIONS 41H TO 48H TO AID IN SEQUENCING
```

SUMMARY ACCELERATION PROFILE

```
LOC  OBJ        LINE        SOURCE STATEMENT

0127 B110         275         MOV @R1,#10H ; THE EXTERNAL RAM A9A8 ADDRESS BITS FROM 00 TO
0129 19           276         INC R1 ; 01 TO 10 TO 11 TO 00
012A B120         277         MOV @R1,#20H
012C 19           278         INC R1
012D B130         279         MOV @R1,#30H
012F 19           280         INC R1
0130 B100         281         MOV @R1,#0
0132 19           282         INC R1
0133 B190         283         MOV @R1,#90H
0135 19           284         INC R1
0136 B1A0         285         MOV @R1,#0A0H
0138 19           286         INC R1
0139 B1B0         287         MOV @R1,#0B0H
013B 19           288         INC R1
013C B100         289         MOV @R1,#0
013E B940         290         MOV R1,#40H
0140 B800         291         MOV R0,#0 ; INITIALIZE THE 8 LOWER BITS OF THE RAM LOCATION
                  292                   ; COUNTER TO THE 0 POSITION
                  293



0142 FE           294         MOV A,R6
0143 3F           295         MOVD P7,A ; ENABLE NEXT PHASE TO INPUT TO INTERRUPT
0144 FF           296         MOV A,R7
0145 43E0         297         ORL A,#0F0H
0147 39           298         OUTL P1,A ; TURN OFF THE FIELD AND TURN ON THE SELECTED PHASE
0148 990F         299         ANL P1,#00FH ; TURN OFF THE PROHIBITION OF DAC CURRENT REGULATION
                  300
014A 27           301         CLR A
014B A2           302         MOV T,A
014C 55           303         STRT T ; START THE TIMER FROM 0
014D B800         304         MOV R5,#0 ; CLEAR THE UPPER 8 TIME COUNT BITS
                  305
014F 05           306         EN I ; ENABLE THE EXTERNAL INTERRUPT
0150 25           307         EN TCNTI ; ENABLE THE TIMER INTERRUPT
                  308
0151 2451         309 WORKL1: JMP WORKL1
                  310
                  311         END
```

USER SYMBOLS

CLRTF	000A	EXTINT	0003	FON	00E5	HERE1	000F	HERE10	003C	HERE15	00B2	HERE1A	00B4	HERE17	00C0
HERE18	00C2	HERE2	0011	HERE3	0020	HERE4	0022	HERE5	0030	HERE6	0032	HERE9	003A	LASTCT	00D3
LASTPH	00AC	MREST	000F	MSEN	00ED	NEXT1	008A	NEXT2	008D	NPAGE	0100	RESET	000B	S0AD	0102
S0BC	0106	S1	0120	SA0	010A	SB0	0110	SC0	011A	SD0	011C	SERVIC	007D	SYSRST	0000
TIMINT	0007	UPDATE	00A2	WAIT1T	0073	WAITTF	0077	WAITTM	0075	WORKL1	0151	WTMR	007B		

ASSEMBLY COMPLETE, NO ERRORS

CHAPTER 5 CONCLUDING REMARKS

The program titled CONSTANT SWITCHING DELAY using only position feedback that allowed independent speed and current level control ran the motor over a speed range going from 17.6 to 1625 R.P.M. Current level control would allow torque adjustment for any constant load. Because this program has the greatest speed stability and range, it would be the program of choice for any known constant load since a lookup table giving time delays for desired speeds can be easily constructed.

However, for a variable load situation, the program titled VARIABLE SWITCHING DELAY using both position and velocity feedback that allowed independent speed and current level control would be the program of choice. Although the speed range only extends from 100 to 1640 R.P.M. and the rate of time delay updating must be properly adjusted for speed stability, velocity feedback is necessary in a varying load situation. Current level control would allow adjustment for the maximum necessary torque.

The program titled CONSTANT SPEED VIA VARYING CURRENT is of little practical use because the speed range is very limited, only extending from 1400 to 1710 R.P.M., and because speed and current level cannot be independently controlled. It does, however, have the advantage of minimizing power dissipation during high speed operation. It would be interesting to see if using 10 or 12 bit DACs rather than 8 bit DACs would significantly extend the present narrow speed range.

A two phase on program could be written although this would limit the upper limit of speed because in a one phase on scheme

speeds of nearly one chop per phase are obtained while in a two phase on scheme the speed must be slow enough to allow sensing windows of zero current for proper waveform detection.

Acceleration and deceleration profiles could be optimized for different supply voltages, current levels, inertial loads, and frictional loads. Since the air gap between the rotor and stator varies from .003 to .010 inch in the motor used,[1] a more precise motor might be necessary for this undertaking.

The 8 bit timer incremented only every 32 instruction cycles or 43.5 microseconds, thereby limiting the resolution of the phase duration counts. For time resolution less than 43.5 microseconds an external clock can be connected to the T1 input and the counter operated in the event counter mode. Then, ALE divided by 3 or more can serve as this external clock. This would allow a time resolution of 3 instruction cycles or 4.1 microseconds, but a T1 pushbutton input would no longer be possible. Interfacing with an external 16 or more bit timer capable of incrementing with every instruction cycle would be the best solution.

In the future an Intel 8749H, which is similar to the 8039 but has a 2K x 8 EPROM included on a single microcomputer chip, could be used. Since the use of more than 2K of program memory never proved necessary, a microcomputer system could be constructed without using any address latch or external EPROM.

The use of microprocessors belonging to the Intel MCS-48 family, such as the 8039 or 8749H, has one major drawback. The MCS-48 family is primarily designed for switching operations and has a rather weak arithmetic capability. Members of the MCS-48

family have addition instructions but no subtraction, multiplication, or division instructions. Implementation of moderately complicated arithmetic algorithms in the time available would require another type of processor with more arithmetic capability, possibly one used as a slave processor to a MCS-48 master processor. A 16 bit processor would eliminate the need for 2 register arithmetic.

The waveform detection scheme has a major drawback; the field coil must be placed in series with the phase coils. This greatly reduces the maximum possible phase current slew rate and hence increases the minimum possible phase chop duration. Thus, faster speeds could be obtained in an optical detection scheme in which the phase coils were not in series with the field coil.

FOOTNOTES

CHAPTER 1

1. Thornton, Richard D., "Microprocessor Controlled Inductor Motor", Proceedings of IEEE Conference on Variable Speed Drives, September, 1979.

2. Eamsherangkoon, Pipat, "Construction of a Variable Reluctance Machine", M.I.T.B.S.M.E. Thesis, 1979.

3. Wong, William, "A Variable Reluctance Motor with Microprocessor Based Motor Drive", M.I.T.M.S.E.E. Thesis, February, 1980.

4. Maginot, J. and Oliver, W., "Step Motor Drive Circuitry and Open Loop Control", Proceedings: Third Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1974, pp. B-1 to B-39.

5. Beling, T.E., "Permanent Magnet Step Motors", Theory and Application of Step Motors, Ed., B.C. Kuo, St. Paul, Minnesota: West Publishing Company, 1974, pp. 206-251.

6. Lafreniere, B.C., "Interactive Microprocessor-Controlled Step Motor Acceleration Optimization", Proceedings: Seventh Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1978, pp. 97-110.

7. Miyamoto, H. and Goeldel, C., "Optimal Control of a Variable-Reluctance Step Motor by Microprocessor", Proceedings: Eighth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1979, pp. 133-140.

8. Kuo, B.C. and Cassat, A., "On Current-Detection in Variable Reluctance Step Motors", Proceedings: Sixth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1977, pp. 205-220.

9. Kuo, B.C., Lin, W.C., and Goerke, U., "Waveform Detection of Permanent-Magnet Step Motors, Part II", Proceedings: Eighth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1979, pp. 243-256.

10. Unger, J.D., "Adaptive Step Motor Damping Using Current Feedback", Proceedings: Sixth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana,

Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1977, pp. 5-18.

11. Tal, Jacob, "Step Motor Control for Maximum Torque", IEEE Transactions on Automatic Control, April, 1976, pp. 224-227.

12. Kuo, B.C. and Singh, G., "Closed-Loop and Speed Control of Step Motors", Proceedings: Third Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1974, pp. C-1 to C-32.

13. Tal, Jacob, "Torque Boundaries for Permanent Magnet Step Motors", Proceedings: Fourth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1975, pp. M-1 to M-6.

14. Kuo, B.C., "Calculation of Torque-Speed Performance Characteristics of Closed-Loop Control of Permanent-Magnet Step Motors", Proceedings: Fifth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1976, pp. L-1 to L-22.

15. Kuo, B.C., Lin, W.C., and Yen, D.W., "Optimal Torque-Speed Performance Characteristics of Variable-Reluctance Step Motors", Proceedings: Seventh Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering at Urbana-Champaign, 1978, pp. 291-306.

16. Kuo, B.C., "Closed-Loop Control of Step Motors", Theory and Application of Step Motors, Ed., B.C. Kuo, St. Paul, Minnesota: West Publishing Company, 1974, pp. 252-272.

17. Frus, J.R. and Kuo, B.C., "Closed-Loop Control of Step Motors without Feedback Encoders", Proceedings: Fifth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1976, pp. CC-1 to CC-11.

18. Yackel, R.A., "Speed Control of Step Motors", Proceedings: First Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1972, pp. 373-430.

19. Wells, B.H., "Microprocessor Control of Step Motors", Proceedings: Fifth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1976, pp. S-1 to S-9.

20. Wetter, R., Jufer, M., and Imhof, L., "Performance Improvements of Stepping Motors by Means of a Variable Rate of Phase Superposition", Proceedings: Eighth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1979, pp. 307-310.

21. Singh, G. and Kuo, B.C., "A Computer Design and Case Study of a Step Motor Printer System", Proceedings: Third Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1974, pp. D-1 to D-21.

22. Pittet, A. and Jufer, M., "Closed-Loop without Encoder of Electromagnetic Step Motors", Proceedings: Seventh Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1978, pp. 37-44.

23. McKee, K.D., "Load Adaptive Damping of Step Motors by Back EMF Feedback", Proceedings: Eighth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1979, pp. 213-226.

24. Kuo, B.C. and Lin, W.C., "Waveform Detection of Permanent-Magnet Step Motors, Part I", Proceedings: Eighth Annual Symposium on Incremental Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1979, pp. 227-241.

CHAPTER 3

1. Maurice, Bruno, "Fast Recovery Diodes" and "Applications of Fast Recovery Diodes", The Power Transistor in its Environment, Ed., Jean Marie Peter, Aix-En-Provence, France, THOMSON-CSF, SESCOSEM Semiconductors Division, 1978, pp. 337-389.

2. Rischmueller, Klauss, "Base Drive Circuits of Switching Transistors", The Power Transistor in its Environment, Ed., Jean Marie Peter, Aix-En-Provence, France, THOMSON-CSF, SESCOSEM Semiconductors Division, 1978, p. 108.

3. Peter, Jean Marie, "Switching Aid Networks", The Power Transistor in its Environment, Ed., Jean Marie Peter, Aix-En-Provence, France, THOMSON-CSF, SESCOSEM Semiconductors Division, 1978, pp. 181-205.

4. Skanadore, W.R., "Methods for Utilizing High-Speed Switching Transistors in High-Energy Switching Environments", Tempe, Arizona, General Semiconductors, Inc., pp. 1-10.

5. Wong, William, "A Variable Reluctance Motor with Microprocessor Based Motor Drive", M.I.T.M.S.E.E. Thesis,

February, 1980, p.32.

6. Eamsherangkoon, Pipat, "Construction of a Variable Reluctance Machine", M.I.T.B.S.M.E. Thesis, 1979, pp. 14 and 18.

7. Wheeler, Gershon J., The Design of Electronic Equipment: A Manual for Production and Manufacturing, Englewood Cliffs, New Jersey, Prentice-Hall, Inc., 1972, pp. 90 and 164.

8. Peel, M.E., Manager, "Analysis of Wrapping and Rewrapping on Unplated Wire Wrap Posts Exposed to Harsh Environments", U.S.A., Texas Instruments Technical Library.

CHAPTER 4

1. Wells, B.H., "Microprocessor Control of Step Motors", Proceedings: Fifth Annual Symposium on Incremetnal Motion Control Systems and Devices, Ed., B.C. Kuo, Urbana, Illinois: Department of Electrical Engineering, University of Illinois at Urbana-Champaign, 1976, p. S-6.

CHAPTER 5

1. Eamsherangkoon, Pipat, "Construction of a Variable Reluctance Machine", M.I.T.B.S.M.E. Thesis, 1979, p. 23.