MICROPROCESSOR BASED STEPPING MOTOR CONTROL

by

Ali ÖZKAYA

BS. in EE Boğaziçi University, 1982

Submitted to the Institute For Graduate Studies in Science and Engineering in partial fullfillment of the requirements for the degree of

Master of Science

in

Electrical Engineering

Boğaziçi University

1985

ACKNOWLEDGEMENTS

I would like to thank those people who have extended their utmost support to me throughout the realization of my thesis.

I would especially like to express my sincere gratitude to my thesis advisor Dr. Ahmet DENKER for his understanding co-operation and guidance.

I would also like to thank Assistant Sami SARPTÜRK for his valuable suggestions and the photographs which have been taken by himself.

I am also very grateful to Yılmaz GÜNEYLİOĞLU for his help in developing printed circuit boards.

ABSTRACT

The purpose of this thesis is to design and realize a microcomputer based education kit which is used to control a stepping motor in all conditions, such that its capabilities can be studied under software control by the kit's user.

The operation principle and types of the stepping motors have been studied and given in the first two chapters. The performance of the stepping motor is determined, to a large extent, by the type of the drive circuit, so various drive systems have been studied and compared with each other. As a new approach "multi-level drive with programmable power supply" has been developed and tested.

Monitor and motor drive program provide the user to run the stepping motor with his own program which is developed with the use of monitor facilities or with motor drive program by entering all motor related conditions from the keyboard.

ÖZETÇE

Adımlayıcı motorlar günümüzde gittikçe yaygınlaşmaktadır. Bunların mikrobilgisayarlarla kullanımı ise bilgisayar kontrollu takım tezgâhlarının ve robot teknolojisinin gelişimine yol açmıştır. Bu tezin amacı mirkro işlemci kullanan bir adımlayıcı motor eğitim kiti geliştirmektir.

Adımlayıcı motorun çalışma prensibi, tipleri ve olanakları incelenerek, ayrıntılı bir şekilde sunuldu. Adımlayıcı motorların yetenekleri onları sürmek için kullanılan devreler ile oldukça ilgilidir. Çeşitli sürücü devreler incelendi, yeni bir yaklaşımla çok seviyeli programlanabilir güç kaynaklı sürücü geliştirildi, denendi ve sonuçlar sunuldu.

İşletim ve motor sürücü programları, kullanıcıya motoru kendi programı ile sürme olanağı sağladığı gibi, kullanıcı dilerse motoru yalnızca gerekli koşulları (hız, ivme, adım tipi, adım sayısı) klavyeden girerekte motoru sürebilir.

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF TABLES

LIST OF SYMBOLS

CNC	Computerized Numerically Controlled
VR	Variable-Reluctance
H	Hybrid
SS-VR	Single-Stack Variable-Reluctance
MS-VR	Multi-Stack Variable-Reluctance
mc	Microcomputer
mp	Microproccessor
CTC	Clock-Timer Chip

I. INTRODUCTION

Accurate positioning is a common mechanical control problem. For positioning, an actuator should exist. This actuator is commonly a motor. DC or AC motors are widely used when the settling points are far from the starting points. But, if the positioning requires very small movements, conventional motor capabilities fail.

Accurate positioning with very small movements have been achieved after the development of the stepping motors. The earliest forms of the stepping motors appeared in the 1930s as elements in remote positioning systems of the naval vessels and later in the control mechanism of torpedeos. Commercial exploitation of these motors began in 1960's when transistor technology is improved such that they are capable of switching large D.C. currents in motor windings. The rapid growth of digital electronics in 1970's assured the stepping motor's future and today there is a world wide interest in its manufacture and application.

Nowadays the stepping motors are widely used in CNC

(Computerized numerically controlled) machine tools. Developing robotics technology cause to more demand for the stepping motor. In both CNC machine tools and robots there are more than one motors and related positional control is achieved in more than one axis. Some applications for one axis control may be direction control of aerials and valve controllers.

The aim of this work is to develope a microcomputer based education kit which is designed to control a stepping motor in one axis such that its capabilities can be studied under software control by the kit's user. All conditions related to stepping motor can be entered from the key-board and the user can run the motor with this preentered conditions. The user can also develop his own motor control program and drive the motor with a RUN command. By using this capability of the system, the user can see the responses of motor to various conditions and test the motor's specifications.

II. THE STEPPING MOTOR

2.1. General Information About Stepping Motor

The stepping motor is a form of syncronous motor which is designed to rotate a specific number of degrees for each eleatrical pulse.

Stepping motors are usually designed with a multipole, multiphase stator windings. The rotors are either of the variable reluctance type or the permanent magnet type. Although there is wide range of stepping motor designs, the two most important types are variable-reluctance (VR) and hybrid (H). The iron teeth on the stationary and rotating parts of the motor are magnetically aligned such that an accurate positioning of the rotor is achieved. In the case of H motor, the main source of magnetic flux is a permanent magnet and d.c. currents flowing in one or more windings direct the flux along alternative paths. For VR motor, the magnetic field is produced solely by the winding currents. (1)

2.1.1. Multi-Stack Variable-Reluctance Stepping Motors

The MS-VR stepping motor is divided along its axial length into magnetically isolated stacks, each of which can be excited by a separate winding (phase). Each stack consists of a staionary and a rotating element. The rotor elements are single unit (rotor). The rotor position relative to the stator in a particular stack is accurately defined, such that the stator and rotor teeth are fully aligned the circuit reluctance is minimised and the magnetic flux in the stack is at its maximum value.

For MS-VR motor, there is a simple relationship between the step length and the number of stacks and stator/rotor teeth. If there is N stacks(and phases), each stack is excited in turn, producing a total rotor movement of N steps. The same stack is excited at the beginning and end of the sequence and the rotor meves one tooth pitch. Since one tooth pitch is equal to 360/p degrees (P = the number of rotor teeth) the step length should be

$$\text{Step Length} = 360/N_P \text{ degrees.}$$

The motor shown in Figure 2.1. has three stack and eight rotor teeth, so the step length is 360/3x8 = 15 degrees. For the MS-VR motor, typical step lengths are in the range of 2-15 degrees.(1)

Smaller step lengths are obtained with additional stacks and rotor teeth but more stacks (phases) require more drive circuits and drive costs gets higher.

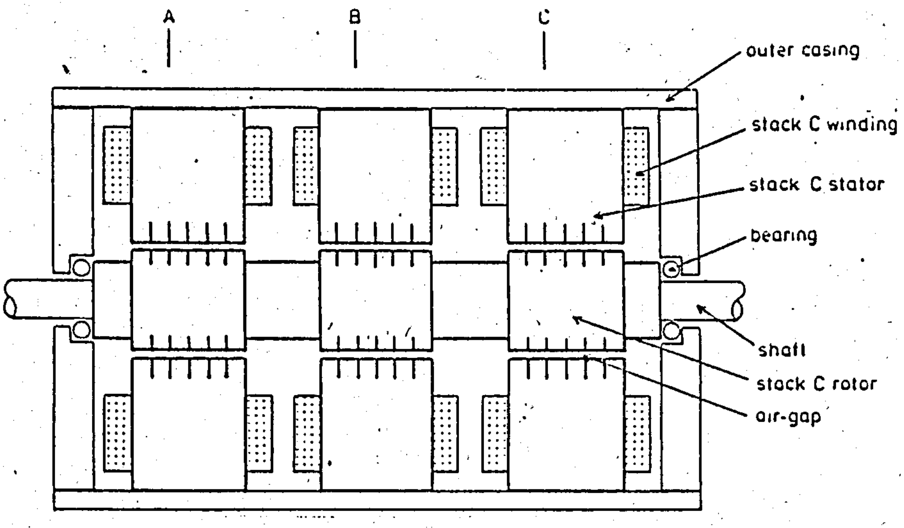


Figure 2.1. Cross-section a three-stack VR stepping motor.

There are four poles and four pole windings in each stack. These all four windings are interconnected to form one phase. The four pole of the three stack motor and the interconnection of pole windings are shown in Figure 2.2. (1)

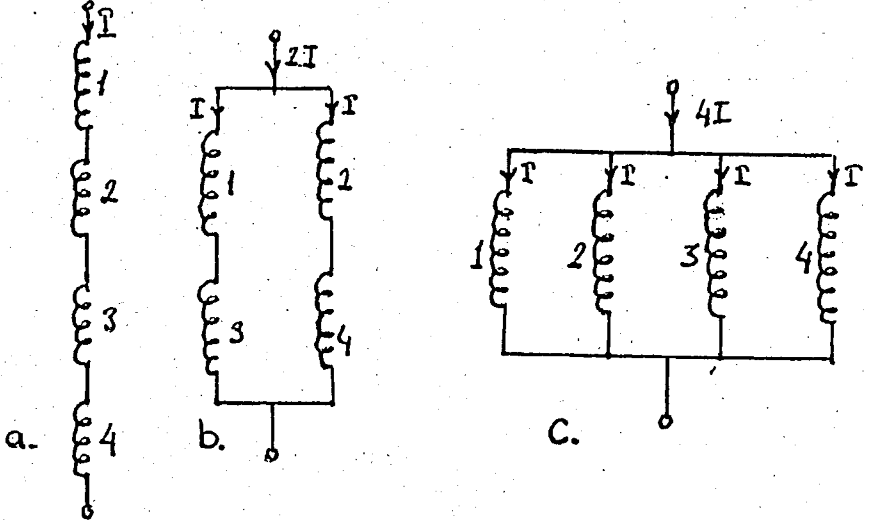


Figure 2.2. a. Series b. Series/parallel c. Parallel interconnection of pole windings

2.1.2. Single-stack Variable Reluctance Stepping Motors.

This motor is constructed as a single unit and the cross-section perpendicular to the shaft shown in figure 2.3 reveals the essential differences between the MS and SS types. As it can be seen, each stator tooth has a separate winding which produces radial magnetic field. The windings on opposite teeth are connected together to form one phase. Since there are six stator teeth, there are three phases in this motor. Another important difference from MS-VR motor is that the rotor has a different number of teeth to the stator.

The step length calculation is the same with the MS-VR motor. N, number of phases, p number of rotor teeth. The tooth pitch is 360/P degrees corresponding to a movement of N steps, so:

$$\text{Step Length} = 360/N_P = 30^\circ$$

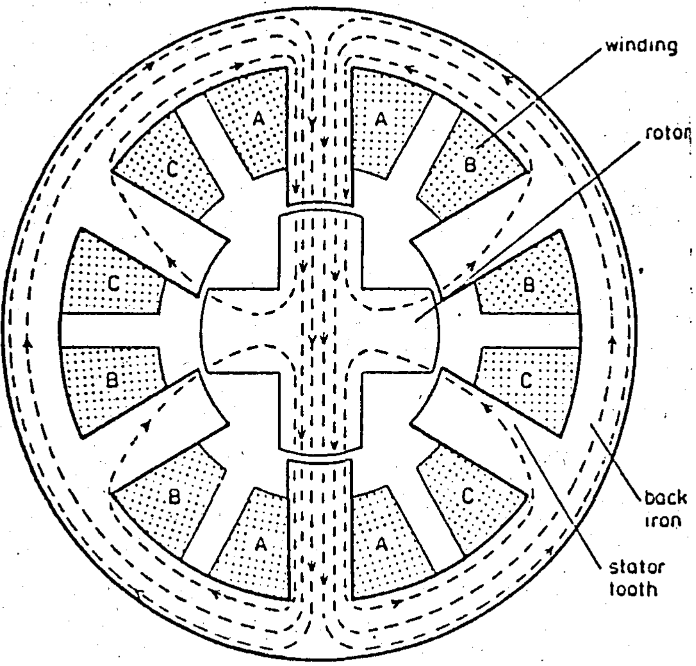
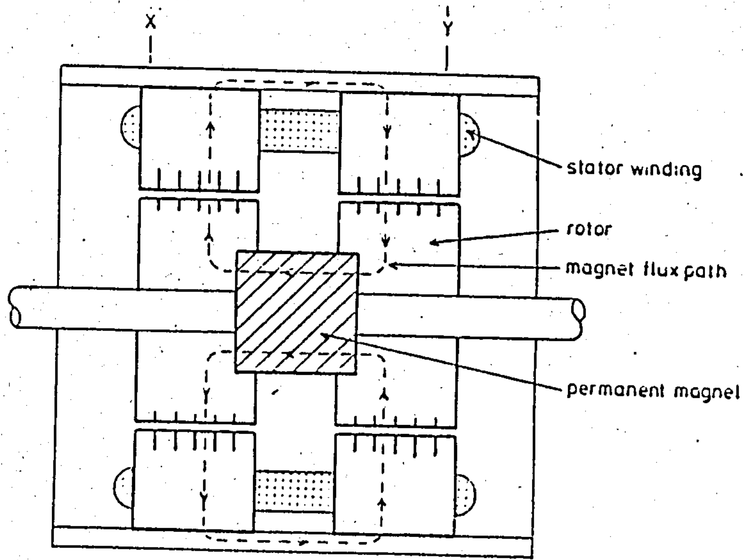


Fig. 2.3. SS-VR Stepping Motor

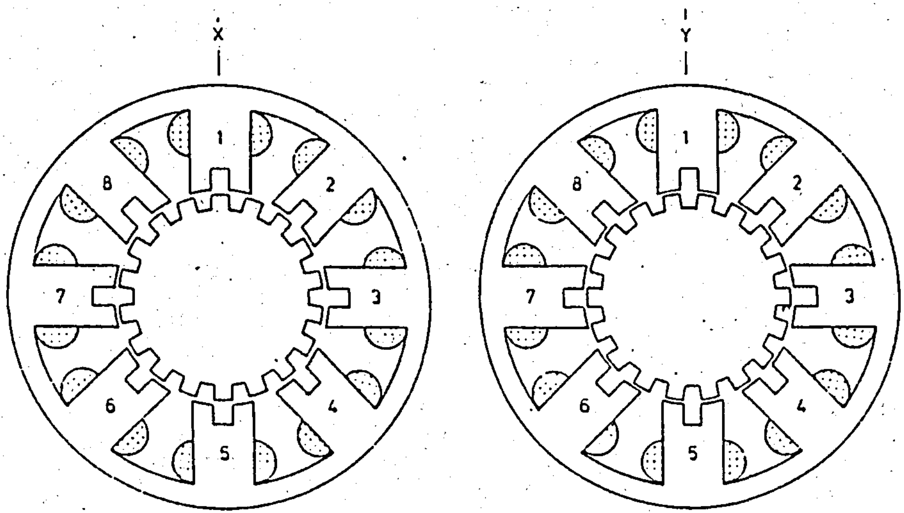
2.1.3. Hybrid Stepping Motors

Since the motor which is used in the thesis application is a hybrid type motor, the explanation about this type will be more detailed.

This type of motors have a permanent magnet on their rotor. The main flux path for the magnet flux is illustrated in Figure 2.4(a). There are typically eight stator poles, as in Figure 2.4 (b), and each pole has between two and four teeth. The stator poles have windings which are used to direct the flow of magnet flux through certain poles according to the rotor position required. There are two windings (phases), winding A is placed on poles 1,3,5,7 and winding B is on 2,4,6,8. Successive poles of each phase are wound in the opposite sense,



(a) Cross-section of a hybrid motor parallel to the shaft



b) Cross-Section of hybrid motor perpendicular to the shaft.

Figure 2.4. Cross sections of an H motor

Sequential excitation of phase windings provides continous rotation of the motor. If the excitation of A is removed and B excited with positive current then alignment of the stator and rotor teeth has to occur under poles 4, 8 of section X and poles 2,6 of section Y in the Figure 2.4(b). The rotor moves one step clockwise to attain the correct position. Clockwise rotation can be obtained by the sequence A-, B-, A+, B+, ... The A+, B-, A-, B+, A+, B-, ... sequence cause the motor to turn in the direction of anticlockwise.

The step length is related to the number of rotor teeth, p. A complete cycle of excitation for the hybrid motor consists of four states and produces four steps of rotor movement. The excitation sequence is the same before and after these four steps, so the alignment of stator/rotor teeth must occur under the same stator poles. Therefore four steps correspond to a rotor movement of one tooth pitch (= 360/P degrees) and for the hybrid motor:

$$\text{Step Length} = 360/4.p \text{ degrees.} \quad (1)$$

The motor which is given in the Figure 2.4.b has 16 rotor teeth resulting a step length of 5 degrees. H motors are usually produced with smaller step lengths than this. For example, a H motor having 50 rotor teeth takes a step with 1.8 degrees.

As being different from these two types of stepping motor discussed in previous sections, there are available some other type of motors capable of stepping action. These are permanent magnet stepping motor and electrohydraulic stepping motor. The detailed information about these rarely used motors can be found in the reference "1".

2.1.4. Comparison of Motor Types

It is not possible to specify any type of motor which is proper for all type of applications. The system designer should detect the requirements of his particular application.

H motors have small step angles (typically 1.8 degrees) which is very important when high resolution angular positioning required. The torque producing capability for a given motor volume is greater in the H than in the VR motor.(1) For applications requiring small step length and high torque, an H motor is natural choice. In the case of H motor, the unexcited magnet flux produces a small detent torque which is useful in applications where the rotor position must be preserved during a power failure.

Since VR motors has longer step lengths and lower mechanical inertia than H motors, they should be chosen where the applications require longer distance movements and faster acceleration.

When a stepping motor is to be chosen, the following information should be determined. Operating speed, torque and load inertia, required step angle, time to accelerate, time to decelerate, type of drive system to be used, size and weight considerations.

2.2. How to Drive a Stepping Motor

It is well known that the performance of the stepping motor is determined, to a large extent, by the type of the drive circuit. There are two main types of the drive circuits.

The VR motor phase currents need only be switched on or off, so a simple unipolar drive circuit is suitable for this type of motors. For the H motors, there are only two phases, but the current polarity is important and a bipolar drive is required to give bidirectional currents.

2.2.1. Unipolar Drive Circuit

The simplest system is the resistance limited (R/L) drive, the essential of which is shown in Figure 2.5.

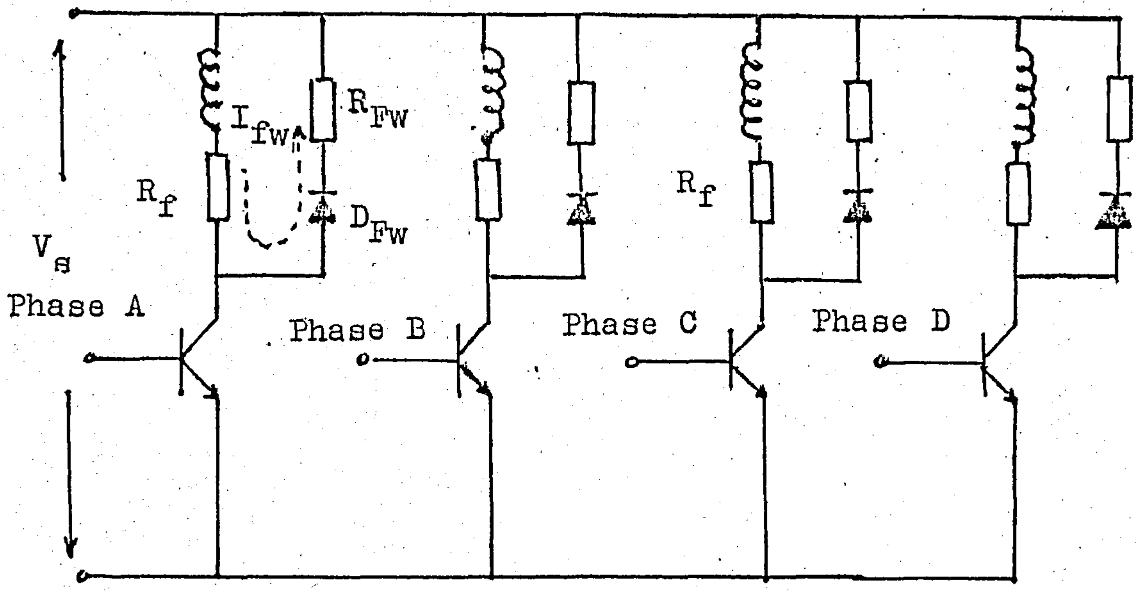


Figure 2.5. Unipolar Drive Circuit

The phase winding is excited when related transistor is saturated. The phase winding has a considerable inductance, so the natural time constant (L/R) is long. At high speed, the phase current can not attain the rated phase current. For satisfactory result, a forcing resistance should be added to reduce electrical time constant. $(L/(R+R_f))$ of course, a proportional increase in supply voltage V_s is required.

Because of the finite phase inductance, the phase current cannot be switched off instantaneously. When transistor is turned off, the current decays through a free wheeling diode and resistor, so that the transistor may be protected from the inductive voltage spikes.

2.2.2. Bipolar Drive Circuit

Bipolar drive circuits are developed for use with H and PM stepping motors. One phase of a transistor bridge bipolar drive circuit is shown in Figure 2.6. The transistors are switched in pairs according to the current pola-rity.

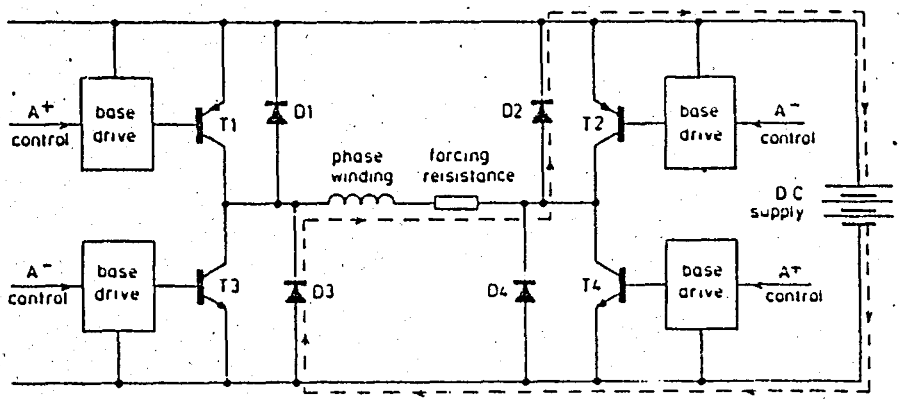


Figure 2.6. One phase of transistor bridge bipolar drive circuit

For positive phase current, transistor T1 and T4 are turned on. In the opposite case the transistors T2 and T3 are turned on so that the current direction in the phase winding is reversed.

A bridge of four diodes, connected in reverse parallel with the transistors provides the path for freewheeling currents. Freewheeling currents in the bipolar drive decay more rapidly than in the unipolar drive, because they are opposed by the dc supply voltage. (1)

In the case of H and PM stepping motors, the drive circuit cost is very high. The bridge configuration base drive circuits has the additional complication since they need optical isolation for the pair of transistors cannected to the positive supply rail. As far as drive costs are concerned, the H and PM motors have considerable disadvantage. To overcome this drawback, motor manufacturers have developed "bifilar wound" hybrid motors, which can be operated with a unipolar drive.

2.2.3. Bifilar Windings

A bidirectional field should be produced in the H motor for stepping. By using bifilar windings, the same result can be obtained with a unidirectional current on the two pole windings in opposite senses, as illustrated for one pole in Figure 2.7. The effect of the negative current in the conventional winding is then achieved by positive excitation of the bifilar (-) winding. The bifilar (+) winding is in the place of conventional winding. Bifilar windings increase the manufacturing cost but simplfy the drive circuit and reduce the cost of it. Because the motor which is used for the application of the thesis has bifilar windings, the drive circuit which is developed is for this type. The detailed information about drive circuits used with bifilar winding H motors will be given in the next chapter.

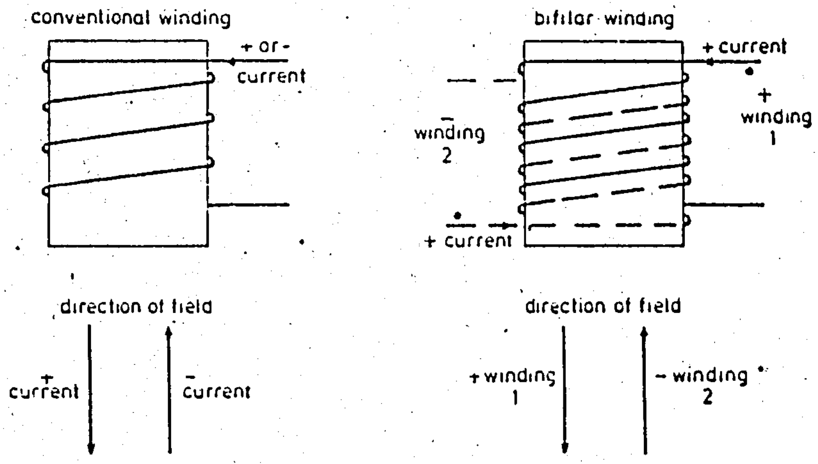


Figure 2.7. Comparison of conventional and bifilar windings

2.3. Some Important Characteristics of the Stepping Motors

As it is known, the stepping motor is developed for the accurate positioning of a mechanical load. Since a mechanical load is concerned, static and dynamic torque characteristics of these motors are very important to use them accurately.

2.3.1. Static Torque Characteristics

External load torques cause small positional errors when the motor is stationary. This type of position error is non-cumulative, i.e. it is not dependent on the number of steps previously taken.

The maximum allowable static error should be determined before the choice of motor. Manufacturers generally give the static torque/motor characteristics as shown in Figure 2.8. These characteristics shows the torque developed by the motor as a function of rotor position for several values of winding currents.

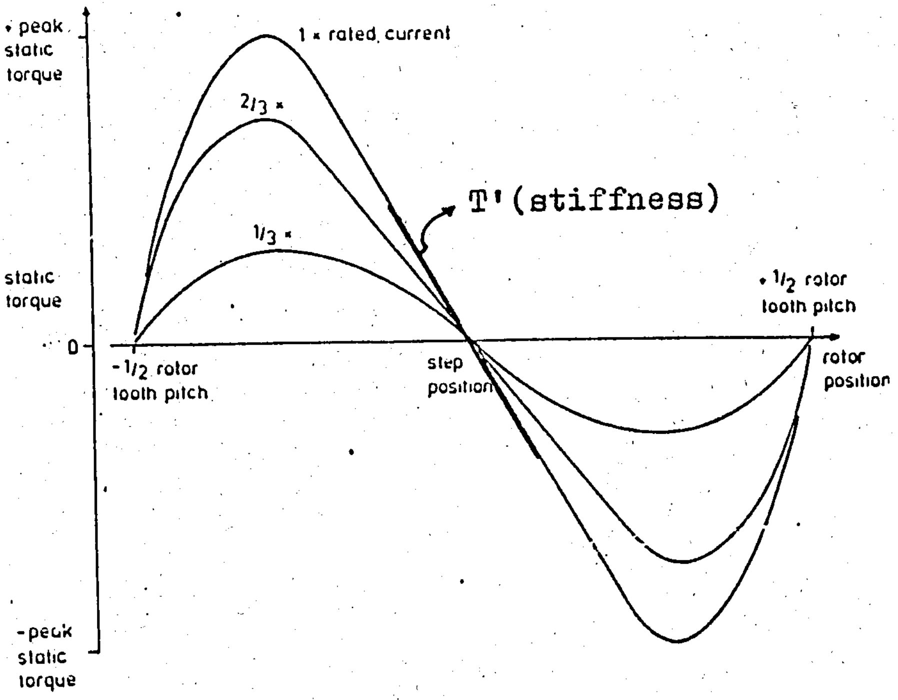


Figure 2.8. Static torque/rotor position characteristics at various phase currents

The peak static torque is a torque which is developed by the rated current. The maximum load which can be applied under static conditions should be equal to the peak static torque. If the load exceeds the peak static torque the motor cannot hold the load at the position demanded by the phase excitation. A static position error produced by any load can be deduced directly from the static torque/rotor position characteristics.

For a motor with P rotor teeth and a peak static torque T_{pk} at a rotor displacement θ from the step position the torque produced by the motor: $T = T_{pk}\,\text{Sin}\,p\theta$. When a load torque T_L is applied the rotor is displaced from the demanded position by the angle θe at which the load and motor torques are equal: $T_L = T = - T_{pk}\,\text{Sinp}\,\theta e$ and the static position error is:

$$\theta e = \text{Sin}^{-1}(-T_L/T_{pk})/p \qquad (1)$$

As it can be seen, the static position error can be reduced by increasing the peak static torque. This improvement can be achieved by using multi-phase excitation.

In the case of H motor, If the motor is bifilar wound there are four phases. For each phase, the static torque/ rotor position characteristics are shown in Figure 2.8. The characteristics can be approximated by the sinusoidal functions.

$$T_A+= -T_{pk}\mathrm{Sin}(p\Theta) \qquad T_B+ = -T_{pk}\mathrm{Sin}(p\theta-\pi/2)$$

$$T_A-= -T_{pk}\mathrm{Sin}(p\theta-\pi) \qquad T_B+ = -T_{pk}\mathrm{Sin}(p\theta-3\pi/2)$$

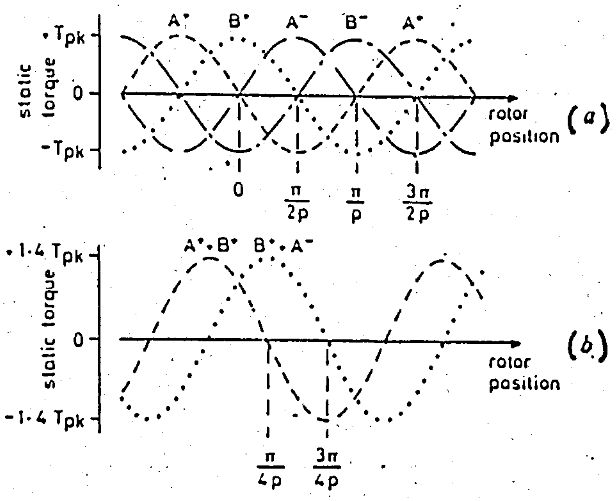


Figure 2.9. Static torque/rotor position characteristics for a hybrid motor
a. one-phase-on excitation
b. two-phases-on excitation

When a pair of phases are excited, the peak static torque is improved by a factor 1.4 over one phase-on excitation.

$$T_A+B+ = T_{A+}+T_{B+} = -T_{pk}\mathrm{Sin}(p\Theta-\pi/4)\mathrm{Cos}\,\pi/4$$

$$= -1.4\,T_{pk}\mathrm{Sin}(p\Theta-\pi/4) \qquad (1)$$

The static positional error can be reduced by connecting the motor to the load by a gear . or a leadscrew.

2.3.2. Torque/Speed Characteristics

The most important characteristic of the stepping motors is the pull-out torque/speed (step rate) characteristic showing the maximum torque which can be developed at each operating step rate. As the stepping rate is increased the motor can provide less torque because the rotor has less time to drive the load from one position to the next as the stator winding current is shifted.

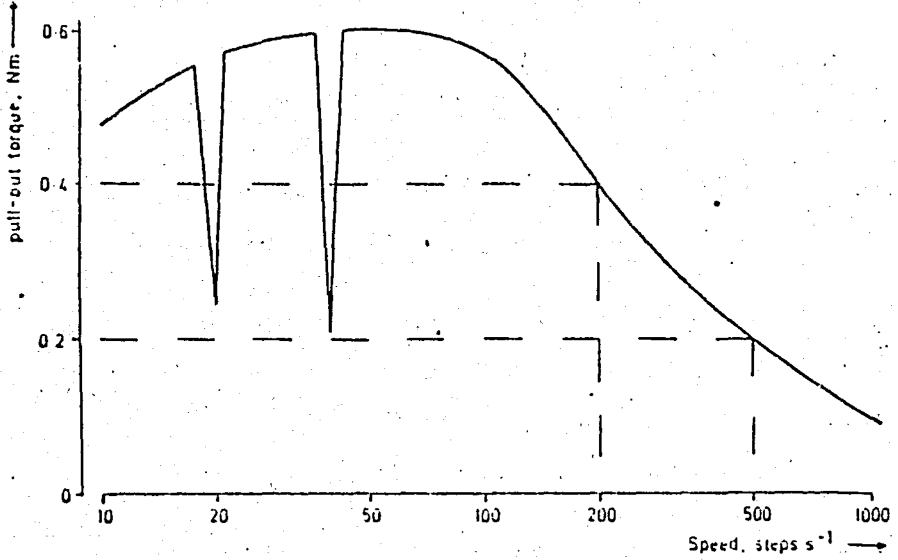


Figure 2.10 Pull-out Torque Speed Characteristic

In the start range, the load position follows the pulses without losing steps. The slew range is that in which the load velocity follows the pulse rate without losing steps, it can not start, stop or reverse on command.

III. THE SYSTEM USED TO DRIVE THE GIVEN MOTOR

3.1. The Stepping Motor Which is Used

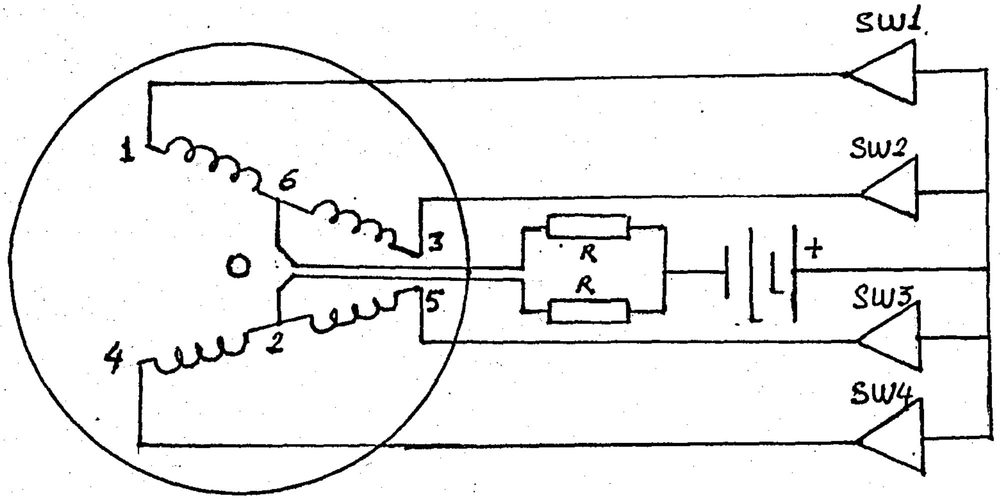
The motor used in this thesis application is a Slo-Syn M092-FD09 Stepping Motor. The related ratings and specifications of this type motor is on the following table.

ELECTRICAL SPECIFICATIONS		
Step Angle	1.8	Degrees
Step Accuracy	∓ 5 %	Percentage
Typical Time for Single Step	3.9	M.Second
Nominal DC. Voltage	2.5	Volt
Rated Current Per Winding	4.6	Amperes
Nominal Resistance Winding	0.55	Ohms
Nominal Inductance Winding	2.76	M Henries
MECHANICAL SPECIFICATIONS		
Minimum Holding Torque	21.6	Kg.Cm
Minimum Residual Torque	0.29	Kg.Cm
Typical Torque to Inertia Ratio	17.2	
Number of Leads	8	
Shaft Diameter	9.53	mm
Max Overhang Load	11.3	Kg
Max Thrust Load	22.7	Kg
Approximate Weight	2.5	Kg
Nominal Rotor Inertia	1.23	Kg.Cm^2

Table 3.1. Ratings and Specifications of The M092-FD09 Motor

The rated time for single step is measured with 24 V. DC. drive. These type of motors have permanent magnet rotors and eight pole stators. They have bifilar windings.

Using the full step drive mode on Table 3.2, the motor step angle is 1.8° with ∓ 5 % precision. The half step drive mode gives a step angle 0.9°. The motor shaft advances 200 steps per revolution (1.8° per step) when a four-step input sequence (full-step mode) is used and 400 steps per revolution (0.9° per step) when an eight step input sequence (half-step mode) is used. The four-step and eight-step input sequences is given on the Tables 3.2. Connection diagram given by the manufacturer is in Figure 3.1.



Unipolar type connection

Figure 3.1. D.C. Stepping Motor Connection Diagram

Four Step Sequence (Full-Step)

STEP	SW1	SW2	SW3	SW4
1	ON	OFF	ON	OFF
2	ON	OFF	ON	OFF
3	OFF	ON	OFF	ON
4	OFF	ON	ON	OFF

To reverse direction read from bottom to top

↓ DIRECTION ↑

Clock Wise   Counter-Clock Wise

STEP	SW1	SW2	SW3	SW4
1	ON	OFF	ON	OFF
2	ON	OFF	OFF	OFF
3	ON	OFF	OFF	ON
4	OFF	OFF	OFF	ON
5	OFF	ON	OFF	ON
6	OFF	ON	OFF	OFF
7	OFF	ON	ON	OFF
8	OFF	OFF	ON	OFF
1	ON	OFF	OFF	ON

Eight Step Sequence (Half Step)

Table 3.2. Drive Sequences

3.2. How to Design a Proper Drive Circuit

As it is known from previous chapter, the performance of motor is very much affected by the drive system which is used. Especially it is difficult to drive the stepping motor in a wide range of speed. Because high speed requires small time constant meaning high forcing resistance and to obtain rated current with this resistor requires high D.C. voltage, but this circuit at low speeds causes unstable operation. The induced voltage on every phase is proportional to the frequency of the fundamental component of phase current.

This induced voltage which increases with increasing speed means that increasing speed requires more voltage than the value only required for rated current and time constant calculated for a specific speed.

In calculating the induced voltage, first a suitable model is to be established and then using this model the phase currents varying with speed can be calculated. The phase circuit model must include the resistance and inductance of each winding. In the H motors phase inductances of the windings is independent of the rotor position. The circuit model also includes the voltages induced in the phase winding by the rotor motion. Because the permanent-magnet flux linking each winding varies sinusoidally with the position of the rotor, these voltages are induced. If a motor has p rotor teeth, then the flux linking A + and A- can be expressed like that

$$\psi_{A+} = \psi_M \operatorname{Sin}(p\theta) \qquad \psi_{A-} = \psi_M \operatorname{Sin}(p\theta)$$

ψ_M is the maximum flux linking each winding

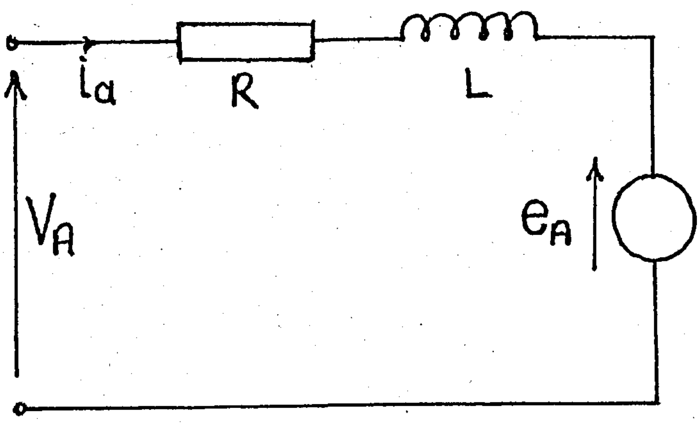


Figure 3.2. Circuit model for one phase of a H motor

When the rotor is at the speed of $d\theta/dt$, the induced voltages in the phase windings are:

$$E_{A+} = d\psi_A/dt = p\psi_M \operatorname{Cos} p\theta \, d\theta/dt$$

$$E_{A-} = d\psi_A/dt = p\psi_M \operatorname{Cos} p\theta \, d\theta/dt$$

As it can be seen easily from above equations, the opposing induced voltages E_A and E_{A-} are proportional to the speed $d\theta/dt$. In order to overcome this speed dependent induced voltage, the d.c. drive voltage should be increased proportionaly. The applied voltage must be considered:

$$\upsilon_{A+} = R\, i_A + (L/2)di_A/dt + (L/2)di_{A-}/dt + E_{A+}$$

where R is one phase total resistance (including forcing) and L/2 is one phase self inductance, the mutual inductance between bifilar windings is also L/2.

This argument reveals that the most important factor in determining the speed range is the phase voltage. The forcing resistance can be regarded as current-limiting resistance. At high speeds the phase current is low, so the voltage drop on the resistance is also law and the remaining voltage from applied voltage balances the induced voltage which increases with increasing speed.

The drive circuit requirements are now clear: a large d.c. voltage is necessary at high speeds, but the phase current at low speeds must be limited to prevent power wastage on the series resistance with the same high d.c. voltage.

There are some circuit configurations providing the above requirements. The two best-known types are bilevel and chopper drive circuits.

3.2.1. Bilevel Drive

In this type of drive there are two supply voltages. A high voltage is applied when the phase current is turned on or off, during continuous excitation a lower voltage is applied to maintain the rated phase current. The circuit diagram is shown in Figure 3.3.

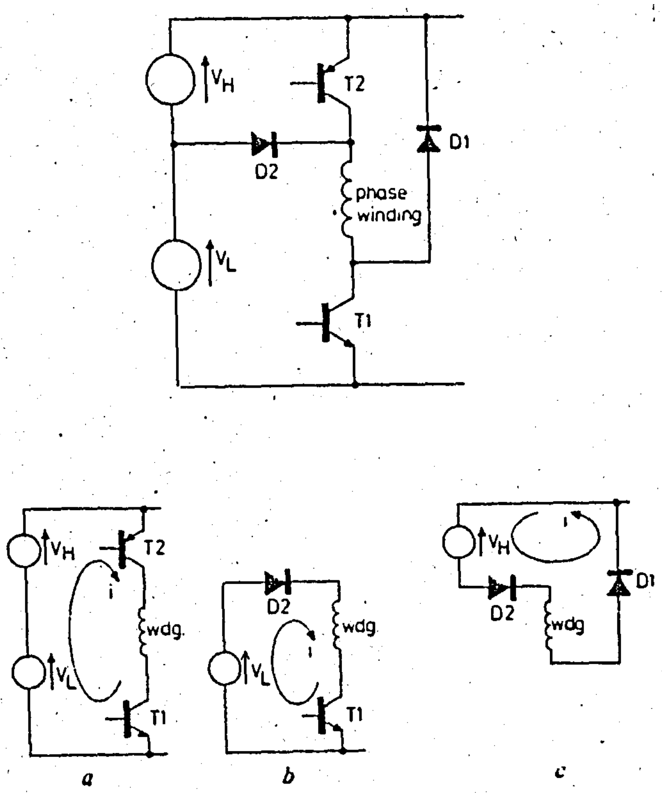


Figure 3.3. The bilevel drive and the effective circuits during the excitation interval
a)At turn-on b)Continuous Excitation c) At turn-of

The main advantage of the bilevel drive is its simplicity. A simple one-shot can control the transistor T_2 at the begining of the each excitation interval for a fixed time. One disadvantage of this type drive is that, during excitation interval the winding current may not overcome the induced voltage.

3.2.2. Chopper Drive

This type of drive circuit has only one supply voltage being high. This voltage is applied to the phase winding whenever current falls below its rated value.

The operating principle of the circuit which is shown in Figure 3.4 can be easily understood by studying this circuit and the diagrams shown in Figure 3.5.

The chopper drive requires more sophisticated control circuitry which increases the cost of the drive circuit. The T_2 base drive requires a Schmitt triggering of the voltage V_c to determine transition levels. If these levels are not well separated the transistor T_a switches on and off at a very high frequency, causing interference with adjacent equipment and additional iron losses in the motor. However the chopper drive have the advantage that the available supply voltage is fully utilized, enabling operation over the widest possible speed range and the power losses in forcing resistors are eliminated, giving a good system

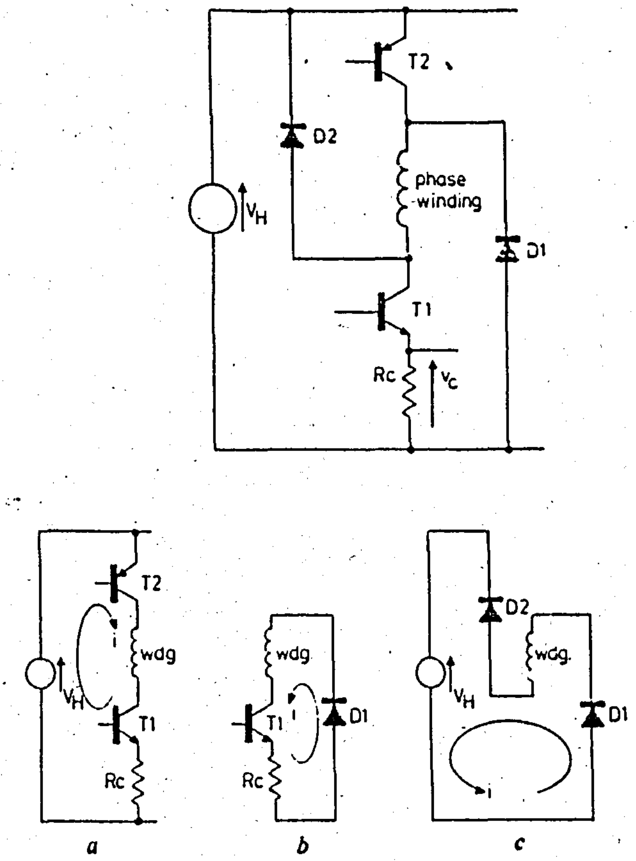


Figure 3.4. The chopper drive and the effective circuits during the excitation interval;a.Current less than rated b.Current greater than rated.

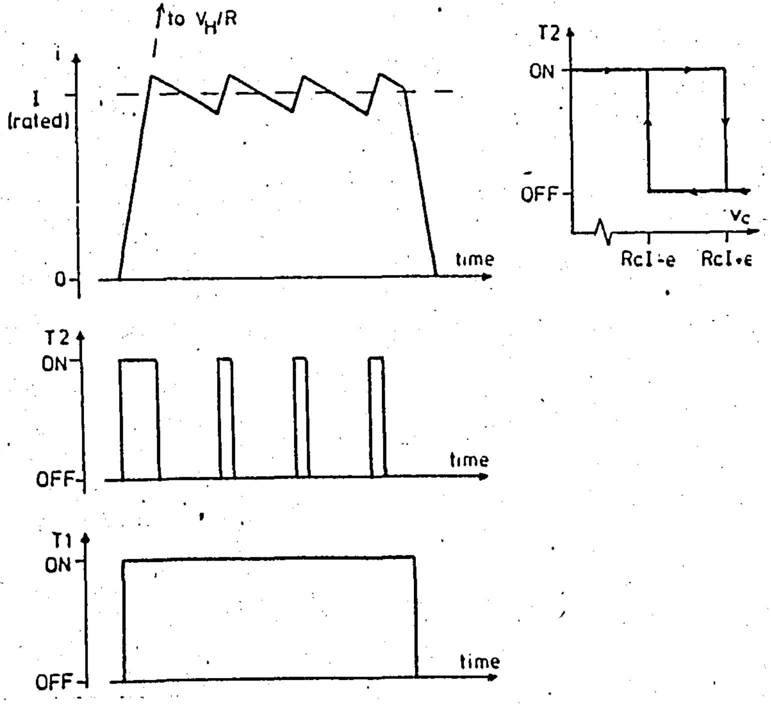


Figure 3.5. Chopper drive current waveform and transistor switching times.

efficiency.(1)

At this stage, it should be asked which configuration is used in this thesis. None of them. A new appoach is developed and used to obtain a wide range of speed with the given motor.

A dedicated microcomputer is developed to control the motor operation. Since there is a microcomputer for control purpose, it is considered that the d.c, drive voltage can be adjustable with speed. So, a programmable power supply is developed and used in stead of bilevel and chopper drives. Then a cost effective conventional drive circuit shown in Figure 3.6. is designed.

3.3. Drive Circuit and Related Calculations.

A unipolar drive circuit is used because the given H motor is a bifilar wound motor. Forcing and freewheeling resistances are shared by two windings of the same phase since the only one of them is activated in both drive modes. The resistance values should be calculated for desired speed range. The desired speed range is 2-2000 step/second. The d.c. drive voltage can be programmed in the range of 5V-47 volts.

Nominal phase resistance: 0.55

Nominal phase industance: 2.76 mH

Rated Current per Winding: 4.6 Amperes

In the case of 2000 step/s. sum of the turn-on and turn off electrical time constants should be at least 1/1000 sec.

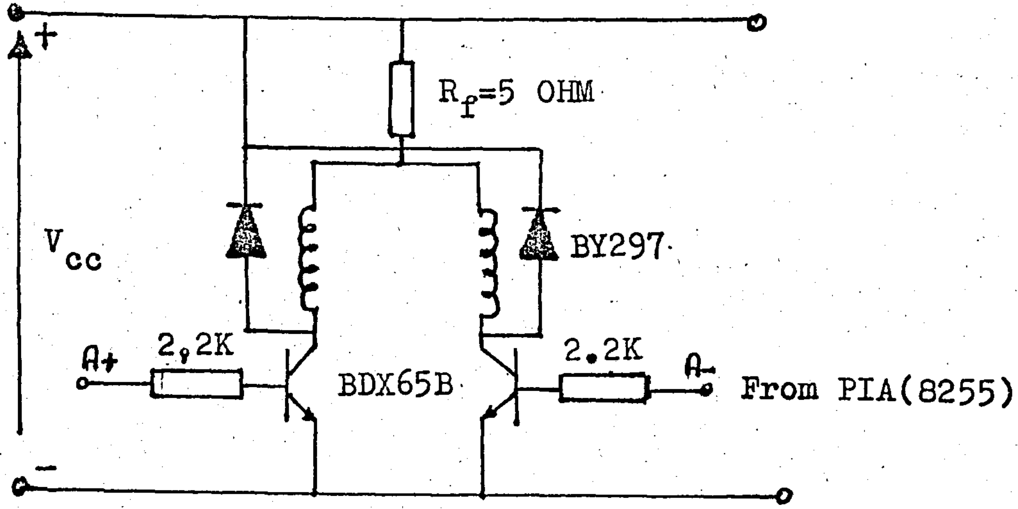


Figure 3.6. Unipolar drive circuit for one phase of the motor.

Turn-on time should not exceed 0.5 mS. Then total winding resistance R_T can be calculated like that:

$$T = L/R_T = 0.5 \text{ mS}$$
$$R_T = 2.76.10^{-3}/0.5.10^{-3}$$
$$R_T = 5.52 \text{ Ohms}$$

Since the winding resistance R_w is 0.55 ohms the forcing resistance R_f should be:

$$R_f = R_T - R_w = 5.52 - 0.55$$
$$R_f = 4.97 \text{ Ohms.}$$

A 5 ohms 30 watts resistor is used for this purpose.

If the power dissipated on this resistor at the rated current is considered, it is calculated as $W_R = RI^2 = 5 \times (4.6)^2 =$ 105.8 Watts. This is very high than it is required, but at low speeds the voltage is decreased. For example, at 2 step/sec. the voltage is adjusted to 5 Volts. Then the power dissipated is approximately 5 watts. At high speeds the current is switched with a high frequency and so, the average power dissipated is two times lower than the calculated value for continous operation at the same d.c. drive voltage.

The freewheeling resistance may be omitted,then turn-off time constant is equal to turn-on time. But the absence of this resistance overloads the freewheeling diode, so a powerfull switching diode (BY297) is used.

Since the drive transistors are darlington type, they can be driven directly from PIO output.

3.4. Programmable Power Supply.

A simple 6 digit D/A converter is designed and the output of the D/A converter is boosted to be able to supply rated current of the motor.

A hex. D type latch chip is used to latch the digital input of the D/A converter. The D/A converter is a simple R-2R ladder network. The converter-resistor array of Figure 3.7. a uses resistors only two sizes, R and 2R. It is to be understood that when Sk=1, the corresponding resistor is connected to a voltage V_R and when $S_R = \emptyset$, the resistor input is grounded. If the simplest situation is considered where So = 1 and the others S_1, S_2, S_3 = 0. Applying Thevenin's theorem, the Figure 3.7.c is obtained. At the output, each digital input contributes its proper relative binary weight. For the circuit shown in Figure 3.7.a, the output analog voltage will be:

$$Vo= V_R/2^6(S_5 2^5+ S_4 2^4+ S_3 2^3+ S_2 2^2+ S_1 2^1+ S_0 2^0)$$

More generally:

$$Vo= \frac{V_R}{2^n} (S_n + 2^{n-1}+ S_{n}-2^{2^{n-1}} + \; . \; .... +So2^o)$$

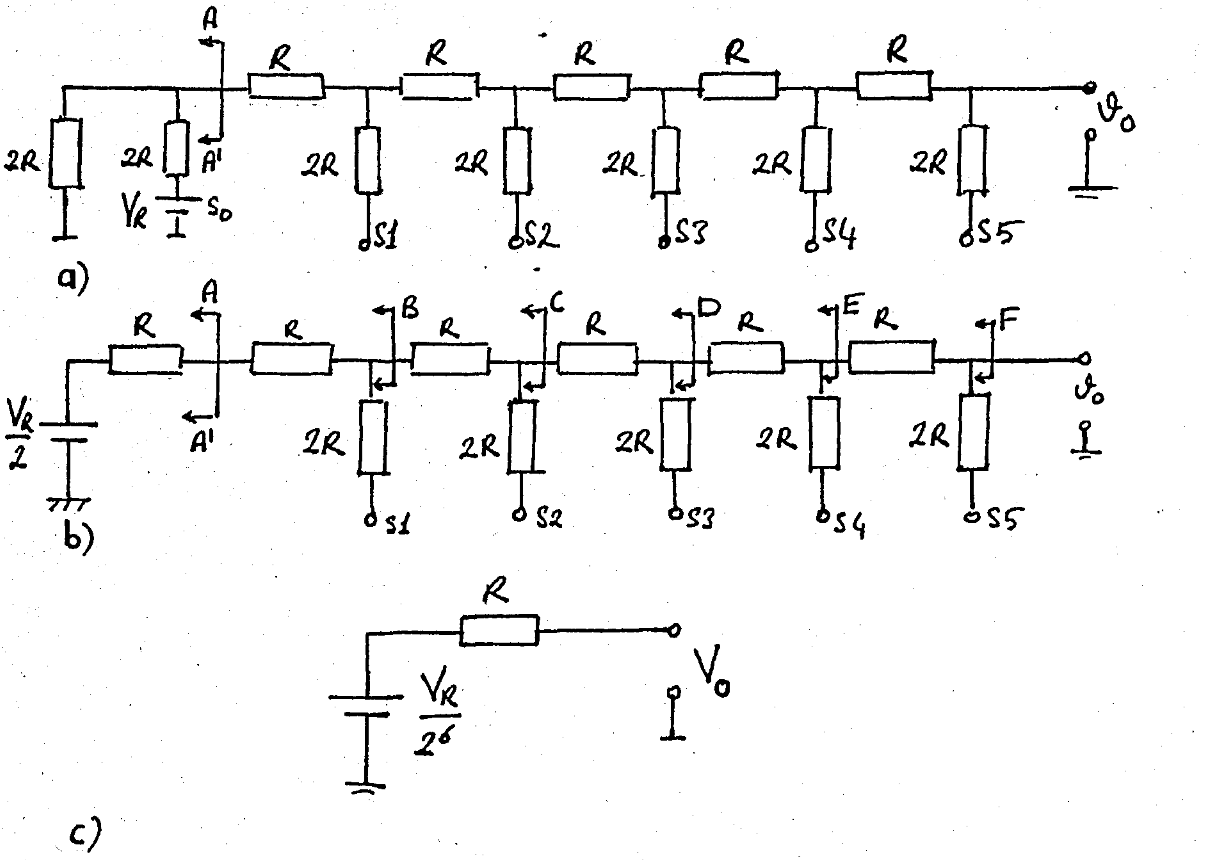


Figure 3.7. R-2R Ladder D/A Converter.

This D/A converter output is amplified with the use of op-amp. The output current of the op-amp is also boosted to provide the rated current to the drive circuit. The complete circuit diagram of the programmable power supply is given in Figure 3.8.

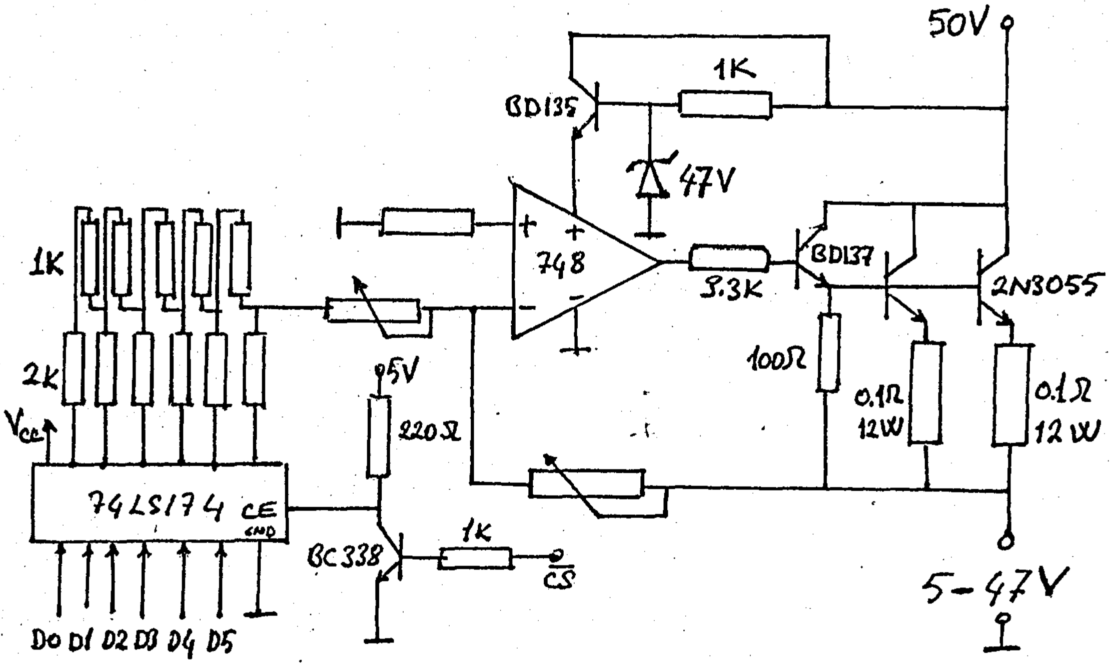


Figure. 3.8. Programmable Power Supply.

A careful study of the above D/A converter reveals that it is an active low converter. The circuit especially designed because TTL IC's low output levels are more consistent than the high output levels. The output current of the op-amp drive a BD137 transistor which forms a darlington pair with parallely connected pass block. The pass block consists of two 2N3055 power transistors.

3.5. How to Control the Stepping Motor

In previous chapter and sections, the choice of stepping motor and design of a proper drive circuit is carefully studied to obtain a good performance. At this stage, the question which will be answered is how the motor and the drive system are to be controlled. Since the aim of the thesis is to study the capabilities of the stepper motor, a microprocessor based control system is designed such that software capability provides required flexibility for research and development.

A block diagram for a typical open loop mp based control system is given in Figure 3.9. Digital phase control signals are generated by the mp. and applied to the drive circuit. In an open-loop control scheme there is no feed back of load position to the controller and therefore it is important that motor responds correctly to each excitation change. If the load parameters are constant with time, the optimum open loop performance can be easily obtained. However, in most applications.the load is not constant and so an optimal control can not be obtained easily.

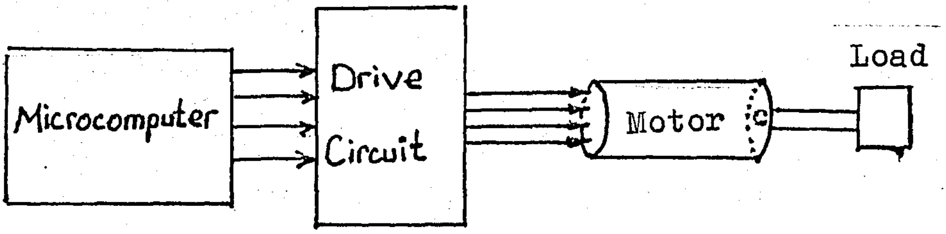


Figure 3.9. A microprocessor based open-loop control.

3.5.1. Time Optimal Control

The main function of any stepper motor control circuit is to generate phase control signals at correct sequence with a correct timing. Generally in most applications the task of the motor is to run in the given direction with the given step size. This given step size can be run in any time, but the desired running is to complete required step size in a minimum time interval.

In general the maximum starting rate of a stepping motor system is much lower than its slew rate (pull-out), so positioning time can be reduced by accelerating the motor over several steps until the maximum slew rate is reached. As the target position is approached the speed is decelerated to the maximum starting/stopping rate in order to be able to stop the motor at target position.

Acceleration and deceleration are necessary to improve move time and setling. Thus timings of the phase control signals are generated by a microprocessor using stored acceleration/deceleration tables, where acceleration/deceleration profiles are linear or piece-wise linear segments determined experimentally. The mp. control acceleration up to the maxinum frequency (N_A steps) or deceleration from the maximum frequency to the settling (N_D steps), for motor move lengths shorter than $N_A + N_D$, the program must look a head and change from acceleration to deceleration at the proper point.(2)

In order to study the acceleration/deceleration capability of any given stepper motor under various conditions, the microcomputer control program provides to the user to enter all stepper motor related conditions; direction, step mode, start speed, step size, acceleration, intermediate half stepping. If any impossible condition is entered, related error messages informs the user. For example, acceleration must not be wanted without entering a finite step size. (Error 2) For error messages look at the appendix D which is microcomputer user manual.

Since acceleration could be given in the range of 1-7F step/sec^2 and any external timer (CTC) is not used, acceleration could be piecewise. If a programable external timer could have been used acceleration profile might be more linear. In the calculation routine, the given step size (N) is divided by two and $N_A = N_D = N/2$. If the maximum slew speed rate is reached-starting from the given start rate-with the entered acceleration in a step length (n) smaller than N_A, the motor attain the maximum slew speed rate along the step length of ($N_A - n + N_D - n = N_A + N_D - 2n$) and then decelerate along the step length of n. Then, the entered step size is completed; $N = n + N_A + N_D - 2n + n = N_A + N_D$. If the maximum slew speed rate is not reached, the given step size is completed by changing acceleration to deceleration at the end of the N_A step.

The acceleration is taken into account at every second, e.g. if a acceleration is 3 step/sec^2 and the starting speed rate is 20 step/sec then in the first one second the motor runs at 20 step/s., in the second one second the motor runs at 23 step/s and so on.

The above explanations are summarized in Figures 3.10.a.b.c and a detailed information about this subject is given in the control program of the stepping motor.

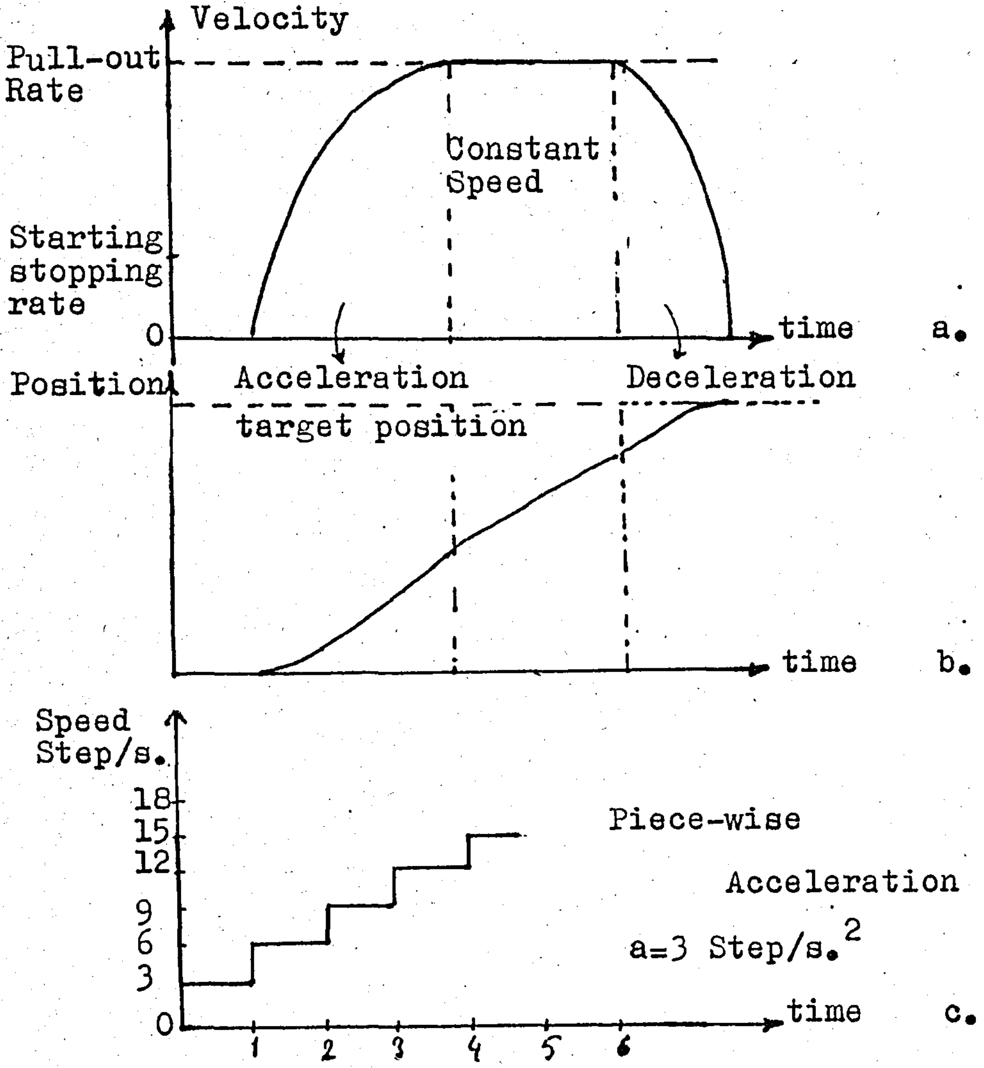


Figure 3.10. a,b)Time Optimal Positioning c) Piece-wise acceleration.

IV. MICROCOMPUTER PART OF THE CONTROL CIRCUIT

4.1. Selected CPU and Information About It

4.1.1. Why Z80?

The ZILOG Z80A microprocessor has been used as a CPU of the microcomputer unit. The choice of Z80 is on purpose. It has a very powerful instruction set which makes program development easy and Intel 8080, 8085 instruction set is sub-set the Z80's, providing that Intel CPU users can operate the microcomputer and also the programs developed for these Intel CPUs can be run directly on Z80.

4.1.2. Internal Structure of the Z80

The Z80 is an 8 bit processor. Its address bus is 16 bits wide and specifies an external memory address 0 to 65535, since the Z80 has no memory mapped I/O. In memory mapped I/O, a position of the memory address must be dedicated to addresses of I/O devices.

The Z80 has 14 general purpose 8 bits registers designated A,B,C,D,E,H,L and A',B',C',D',E',H',L'. Only one set of seven registers and related flag registers F and F' can be activated. A special instruction selects AF or A'F' while another instruction selects B,C,D,E,H,L or B',C',D',E',H',L'. There are four possible combinations which provide process switching and more register storage in CPU. The remaining CPU registers I,R,Ix,IY, SP and PC are special purpose. The index registers Ix and IY are two 16 bit registers that provide indexed addressing. The interrupt vector register I is an 8 bit register that can be loaded with 8 bit of data specifying a memory address. This address is combined with lower order 8 bits of address supplied by the interrupting device. The I register is used with one of three interrupt mode which the Z80 may utilize under program control. The R register is the 8 bit memory refresh register. When external memory is made up of dynamic memories, the R register allows automatic refreshing. (4)

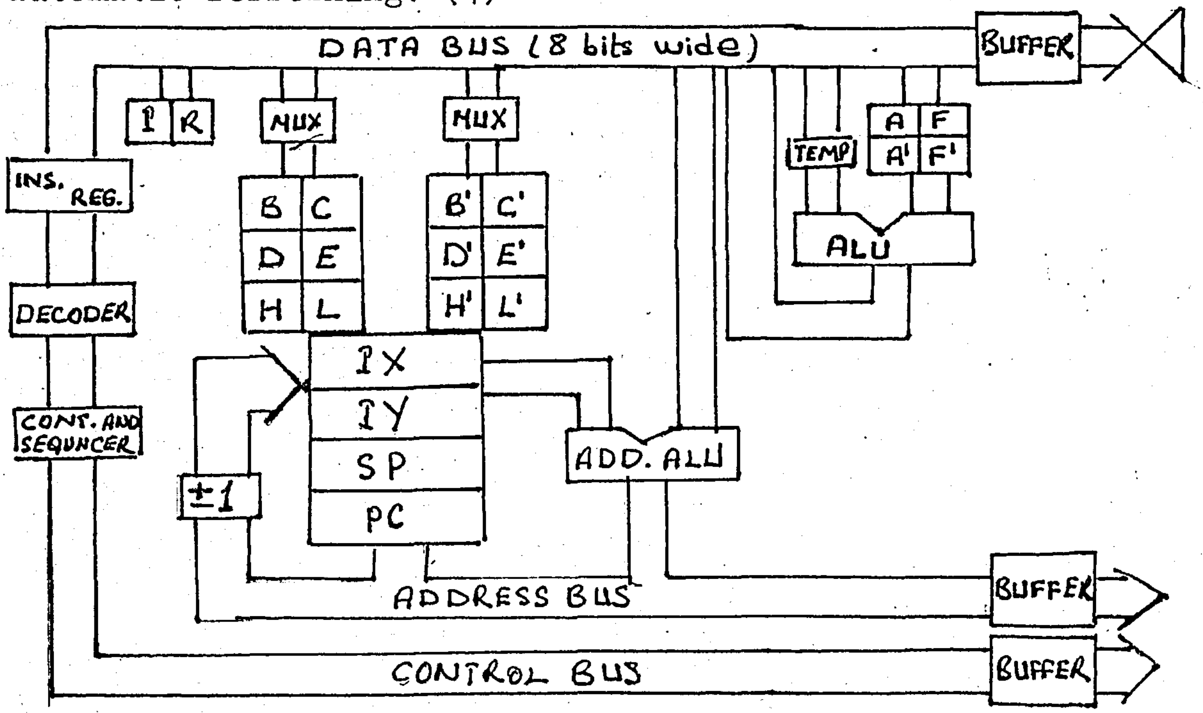


Figure 4.1. Internal structure of the Z 80

4.1.3. Interface Signals and Timing of the Z80

Address Bus is 16 bits wide, and 64 Kbytes of memory can be addressed directly. When I/O instruction executed lower 8 bits A0-A7 contain I/O address. During refreshing, contents of R register appears on A0-A7.

$\overline{\text{BUSRQ}}$ (input), $\overline{\text{BUSAK}}$ (output) pins are used for DMA purposes. $\overline{\text{MRQ}}$ is tristate active low signal indicating that the address bus holds a valid memory address. The $\overline{\text{RD}}$ and $\overline{\text{WR}}$ signals are tristate active low outputs indicating that whether the memory or I/O operation is to be read or write. The $\overline{\text{RFSH}}$ is not used with normal memory operation. When $\overline{\text{RFSH}}$ and $\overline{\text{MRQ}}$ are both active, R register content is on the lower portion of address bus, and external dynamic memory use the A0-A7 to implement one of the refresh cycles.

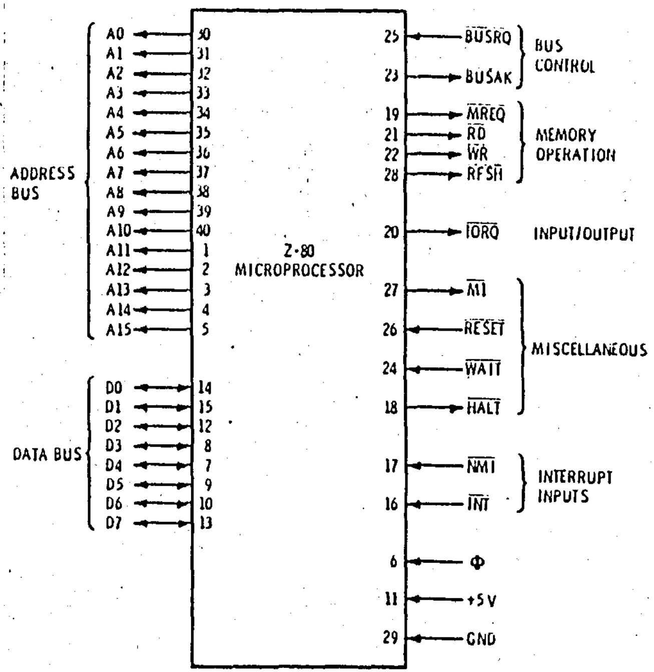


Figure 4.2. Interface Signals of the Z80

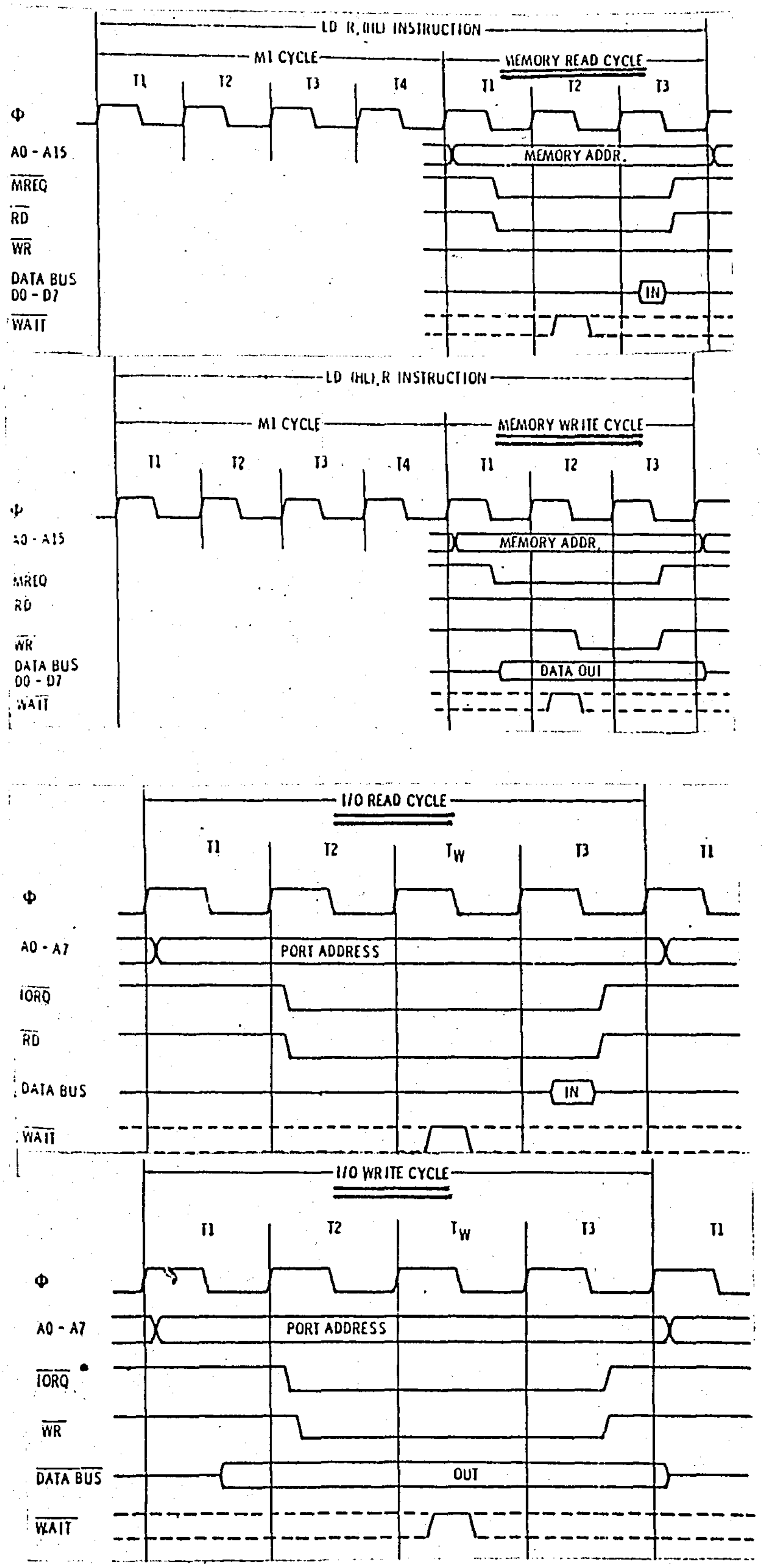


Figure 4.3. Timings of the read and write operations.

The $\overline{IORQ}$ signal is a tristate active low output, indicating that the address bus now contains a I/O address. $\overline{IORQ}$ is also used together with $\overline{M1}$ for interrupt responses. The $\overline{M1}$ signal is active low output signal that indicates the microprocessor is in the fetch cycle of the instruction. The $\overline{WAIT}$ signal is an input signal associated with slow memories or I/O devices. The $\overline{HALT}$ signal is an active low output signal that goes low during the execution of halt instruction.

The $\overline{RESET}$ signal is an active low input that is used as a master CPU reset.

The $\overline{NMI}$ is a negative-edge triggered input that specifies a non-maskable interrupt is to be performed. The main interrupt request signal $\overline{INT}$ is an active low input signal that is supplied by external devices to cause an interrupt. The CPU accepts the interrupt if it is not masked and acknowledges the interrupting device by sending out an $\overline{IORQ}$ during the fetch ($\overline{M1}$) time of the next instruction.

4.2. Hardware of The Microcomputer

The hardware of the microcomputer consists of following main parts. Clock an reset circuit, interrupt circuit, memory and I/O decoding, keyboard and display circuit.

4.2.1. Clock and Reset Circuit

A crystal controlled circuit is used to maintain consistant execution time. There is a 390 ohms pull up resistor that the clock signal satisfies both AC and DC clock signal requirements of the Z80. The reset circuit allows the computer to start program execution immediately after power is turned on. The program execution can be stopped and restarted by using reset button.

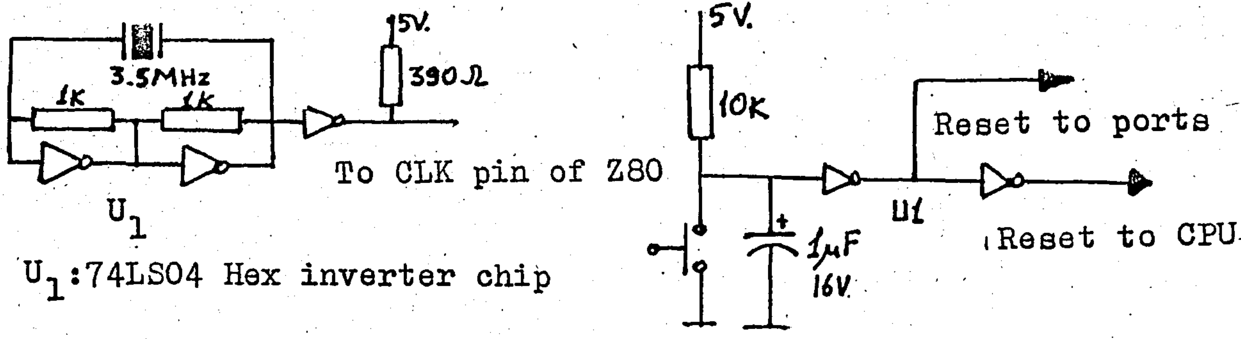


Figure 4.4. Clock and Reset Circuits

4.2.2. Memory and I/O Decoding Circuits

Eight different $\overline{CS}$ signals for memory devices are obtained by using a 1 of 8 decoder chip. (74LS138). Each $\overline{CS}$ signal selects 2K bytes memory devices being 2716 as EPROM and 6116 as RAM. Then, there are 8 memory chips on board. Another 1 of 8 decoder chip is used for the purpose of I/O decoding. Each I/O $\overline{CS}$ signal includes three successive internal port addresses, i.e. when one of these three internal ports is addressed, related $\overline{CS}$ signal is activated automatically.

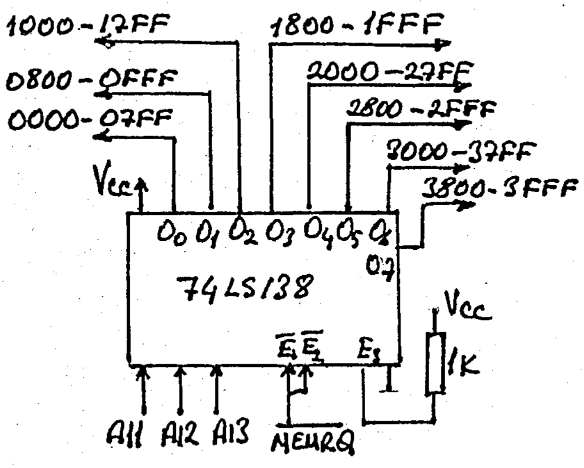


Figure 4.5. Memory and I/O Ports Decoding Circuits

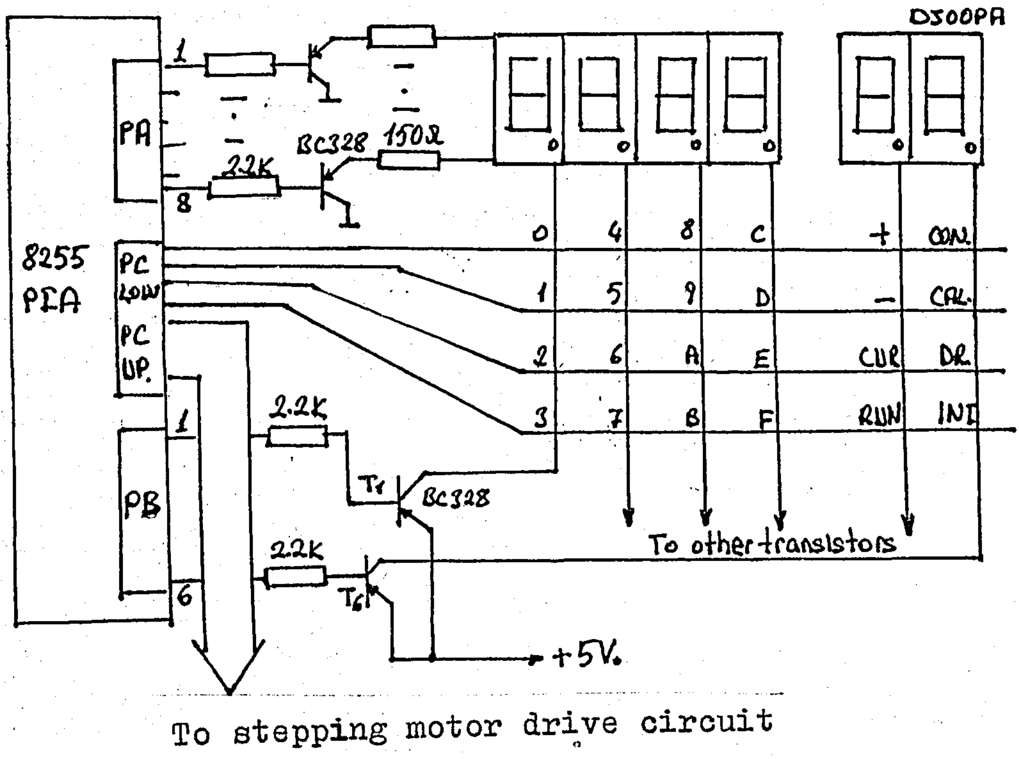


Figure 4.6 Keyboard and Display Circuit

4.2.3. Interrupt Circuit

In order to detect some slow mechanical displacements, interrupt facility of the CPU can be used. But slow, i.e. a long active low pulse on CPU interrupt pin can cause multiple interrupt because of the automatic masking of the interrupt. To overcome this problem a dual One Shot IC is used to shorten the long interrupting pulses for both $\overline{\text{INT}}$ and $\overline{\text{NMI}}$.

4.2.4. Keyboard and Display Interface

A Keyboard consists of pressure or touch activated switches. A combination of hardware and/or software means are required to detect which key has been pressed. Encoded and nonencoded keyboards are available. Encoded keyboards include the hardware necessary to detect which key was pressed and to hold that data until a new key stroke. The encoded ones are very easy to use but they are expensive. Non-encoded keyboards have no hardware and must be analyzed by a software routine. (5)

A non-encoded keyboard is used on this microcomputer. As a display, six 7 segment display digits are used. Key columns and display digits are scanned together. Whenever a digit is selected, an input operation is made from port C upper whether there is a key stroke on the related column or not. If there is no key pressed on the selected digit column, CPU reads all 0 from the port C upper. Common anode digits (columns) are scanned by a walking one. If there is a key pressed, the related one is read from one of the rows of port C upper. Keyboard routine recognizes the key which is pressed. This key identification tecnique is known as "row scanning". After this first key stroke detection; monitor waits untill the same column (digit) is activated such that this next detection and check provides key-debounce time (6x delay time between each digit selection = 10 mS). This is a software solution to key bounce problem which causes multible data entry. After the second detection, required key operation is made and monitor waits for release of the same key. Release detection of the key is also verified in order to prevent key bounce problem.

Release detection and verification provides n key lock out facility. Rollover is the problem caused when more than one key is pressed at the same time. The two main techniques used to solve this problem are the n-key rollover and n-key lock out. N-key rollover either ignore all keys pressed until only one remains down. N-key lock out takes into account only one key pressed. The first key pressed generates the code, the other ones are ignored. Each key should be released before the next one is pressed down.

Transistor buffer stages are used in the keyboard display circuit. This is because, when multiplexing, each display must be 6 times bright as when it operates alone, since it is an 1/6 times as long. Thus, currents needed for each digit are 6 times as large. PIO IC can not provide this current, so external discrete transistors are used.

V. EXPERIMENTAL RESULTS, DISCUSSION AND CONCLUSION

5.1. Drawbacks of The Microprocessor Based Stepping Motor Control

Microprocessor based control can be achived by using two different approach: software-intensive, hardware-intensive. In software-intensive system (the system which is used) phase control signals are generated by a dedicated microprocessor, but in the case of hardware-intensive system, there is a hardware controller which operates only with given target position information and start command by the microprocessor. In applications involving the real-time control of several other devices the hardware-intensive approach is the more realistic choice because of programming constraints.

The software intensive control provides accurate and detailed timing. If the time taken to execute one cycle of delay routine is T_1 and the time occupied in changing excitation, step count and delay pointing is T_2 , then, for a delay count "d", the time between excitation changes is:

$$\text{Step Interval: } dT_1 + T_2$$

T_1 and T_2 are fixed by the number of processor instruction cycles required to execute the corresponding section of software. The Table 5.1. illustrates the weakness of the software based system. As it can be seen from the table, at low speeds a double decrement of delay count value corresponds one increment at step rate, but at high speeds a double decrement of delay count corresponds more and more increment at step rate with increasing speed.

Stepping Rate(Step/s)	Delay Count Value.
240	0222
241	0220
-	-
512	00FE
517	00FC
-	-
899	0082
1023	0080

Table 5.1. Stepping Rate and Delay Count Value Relationship.

Constant and nonavoidable T_1 and T_2 creates the above problem. In order to overcome this drawback, hardware-intensive approach should be chosen and used. In the case of hardware-based control, the maximum operating speed of the motor is no longer restricted by its ability to jump between discrete stepping rates. So, the maximum operating speed of the motor can be increased by using hardware-based control.

The acceleration profile for given conditions is calculated and stored in a look-up table. Since a real-time timing is necessary and this is also provided by microprocessor, each delay count is used along 1 sec. then the next delay count is used along the second 1 sec. So acceleration and deceleration is performed with 1 sec. intervals.

The speed condition entered from keyboard can not be realized at the given value because of the time (T_2) occupied by the drive routine itself. T_2 disturbes a wide range real time timing. To optimize this effect, an average speed (600 step/sec.) is selected and speed timing is adjusted to this value. The speed values higher and smaller than this optimized value do not correspond to real speed. When the difference between the given speed value and the optimized speed value (600 step/s.) increases, the difference given and realized speed rates also increases.

If an external programmable timer is used to generate real timing, then more linear acceleration and deceleration can be achieved.

5.2. About Drive Circuit Which is Used

In order to run a stepping motor in a wide range of speed, bi-level and chopper drives are generally used, as it is mentioned in Chapter III. As a new approach, a conventional drive circuit with programmable power supply is designed, developed and tested. Although the desired speed range is reached with this type of drive circuit, the result is worse than the result obtained from the use of chopper drive, better than bi-level drive. This is because the D/A converter bit number is restricted by the time occupied by drive routine. An 8 bit D/A converter provides 256 different supply voltage level, but it requires a long comparison routine to obtain proper power code for given delay (speed) count. This increments the drive routine time and speed adjustment becomes more difficult. So, in order to optimize time and power requirements, only 10 level adjustment is provided although the D/A is designed as 6 bit.

Actually this type of drive system may be called as multi-level drive and it operates better than bi-level, but worse than chopper drive systems.

Chopper drive action can be obtained by using a power supply having current limit facility as the main supply of the drive system.

5.3. Step Response Related Considerations

Since stepping motors run step by step, step response of the motors have great importance as far as correct positioning is concerned. When a step signal is taken by a stepping motor, the motor makes the single step angular rotation within a period of time which is called "step response time". This time is a function of the torque to inertia ratio of the motor and of the characteristics of the drive circuits.

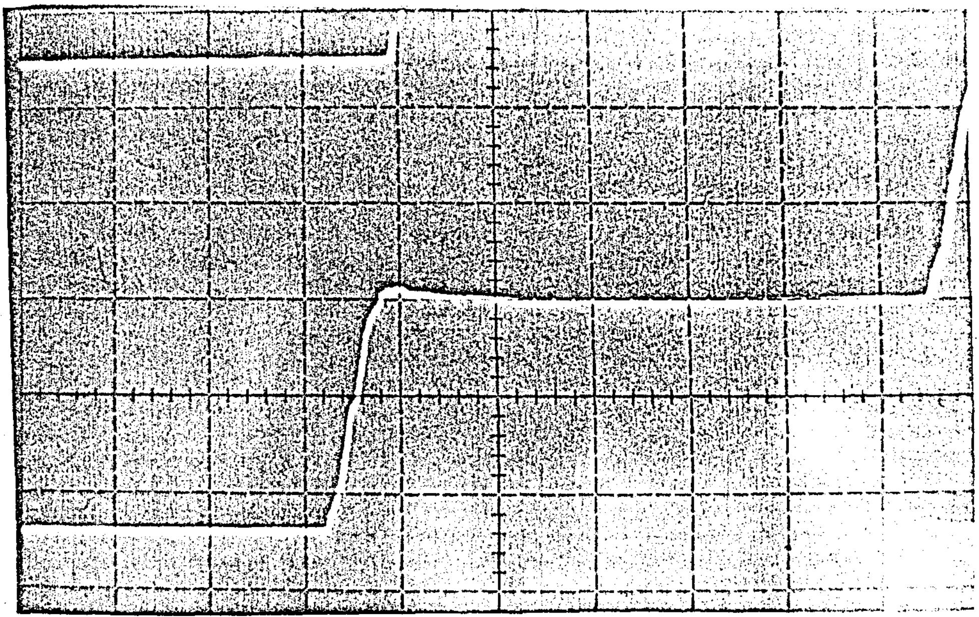
DC drive voltage level has the most important effect on the step response. The photographs given in the next page reveals that, there is no overshoot with 7 Volt DC drive level, but overshoot increases and rise time decreases with increasing DC drive level(12V, 17V). First three photographs are taken at different DC voltages without load. The last one is taken with load at 17 V DC drive voltage. With load, as it can be seen from last photograph; overshoot decreases and rise time increases.

The results obtained from above tests are summarized in the Table 5.2.

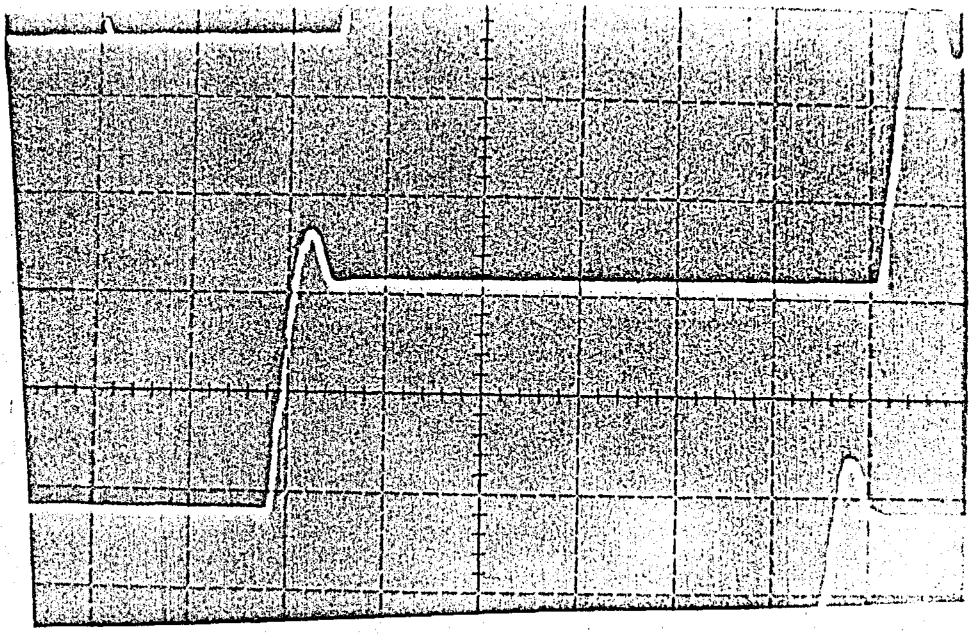
DC level	Load	Overshoot %	Rise Time	Settling Time (5 % of st
7 Volt	No	3	6 ms	25 ms
12 Volt	No	20.4	4 ms	24 ms
17 Volt	No	55.5	2.5 ms	30 ms
17 Volt	Yes	9.3	8 ms	35 ms

Table 5.2. Step Response Test Results.

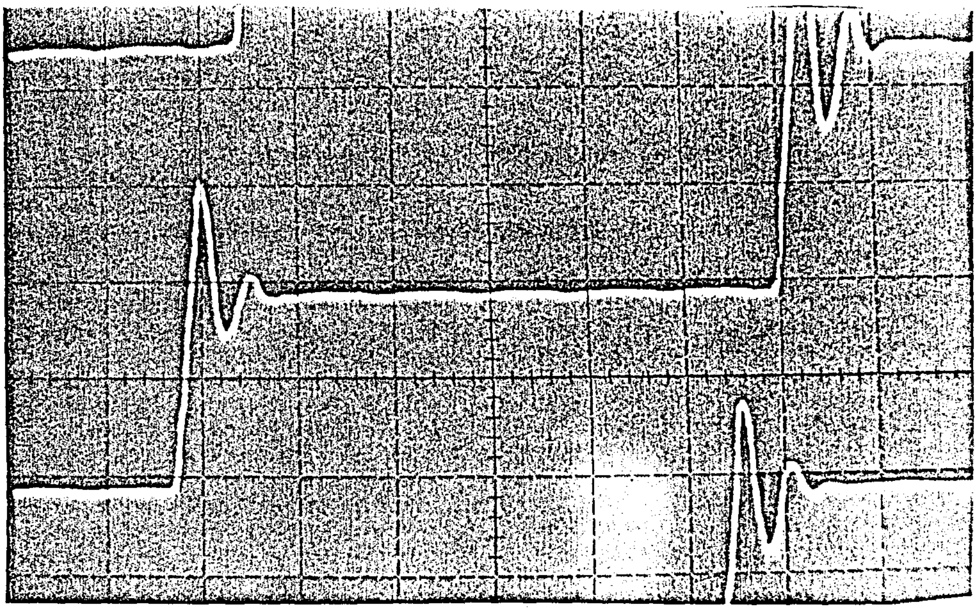
STEP RESPONSE PHOTOGRAPHS AT VARIOUS CONDITIONS



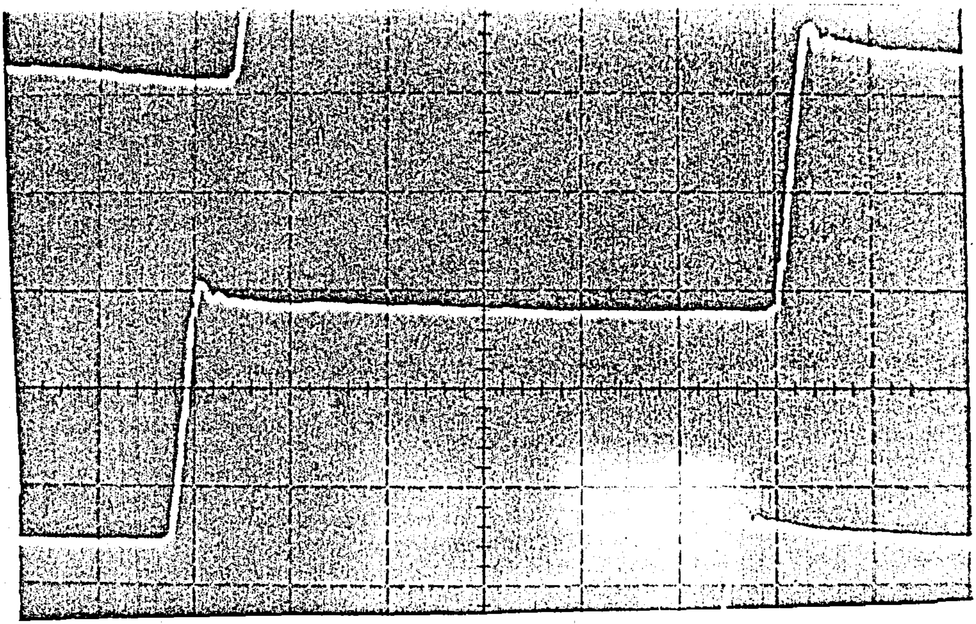
a. V_{cc} = 7 V. , No load



b. V_{cc} = 12 V. , No load



c. V_{cc}= 17 V. , No load



d. V_{cc}=17 V. , With load

As it can be seen from photographs, the single step response is very oscillatory. In applications requiring frequent accurate positioning, this poorly-damped response can be a great disadvantage. The photographs reveals that the stepper motor-load combination can be represented by second order differential equation. The torque generated by the motor should be equal to;

$$T_g = T_L + DW + J\ dw/dt \tag{5.1}$$

where T_L is load torque, D is coefficient of viscous friction, J is total moment of inertia, W is angular velocity.

Motor torque (T_g) at a rotor position θ is $- T'\theta$; where T' is the stiffness of the static torque/position characteristic. Stiffness is the slope of the mentioned characteristic at the equilibrium position. (Figure 2.8). θ is angular displacement and $W = d\theta/dt$. Then the equation 5.1 will be:

$$- T'\theta = T_L + DW + Jdw/dt \tag{5.2}$$

$$- T'\theta = T_L + Dd\theta/dt + Jd^2\theta/dt^2 \tag{5.3}$$

At no load conditions:

$$J\ d^2\theta/dt + Dd\theta/dt + T'\theta = 0 \tag{5.4}$$

Frequency of oscillation and the damping ratio can be derived from above equation.

$f = (T'/J)^{1/2}/2\pi$; the undamped natural frequency of oscillation

$\delta = D/2\,(T' \cdot J)^{1/2}$; damping ratio

The value of the damping ratio is important. It shows whether the oscillation dies or not.

If $\delta < 1$; oscillation will not die (undamped)
If $\delta = 1$; critically damped
If $\delta > 1$; oscillation is overdamped.

In order to obtain damped step response, δ should be equal to or greater than unity and this means; D should be equal to or greater than $2(T' \cdot J)^{1/2}$ Damping ratio can be increased by increasing the coefficient of viscous friction (D) and by decreasing total inertia (J) and the stiffness (T').

D can be increased by introducing additional viscous friction, so that the rotor oscillations decay at a faster rate. If motor and load have been chosen, decreasing total inertia is impossible.

T' (stiffness) is very much affected from the drive system which is used. This parameter provides to the user to run the motor with steps having no overshoot. As it can be seen in Figure 2.8, the T' decreases with decreasing phase current. If a specific load is to be driven at a specific speed, it is possible to obtain a step response without any oscillation by adjusting the drive circuit to give proper current to the motor. It is the case in photographs

For the damping purpose, electrical and mechanical means are used. One mechanical method of damping is to increase viscous friction. However the use of straight forward viscous friction is undesirable because the operation of the motor at high speeds is severely limited by the friction torque. More detailed information about damping can be found in references (1,7).

When a stepping motor is operated at its natural frequency, an increase in the audio and vibration level of the motor may occur. The resonant behavior of the motor causes the loss of torque at specified stepping rates, as it can be seen with the dips in the Figure 2.10. Resonance usually occur when the motor is excited where the rotor is in advance of the equilibrium position and has a positive velocity, as indicated in Figure 5.1.

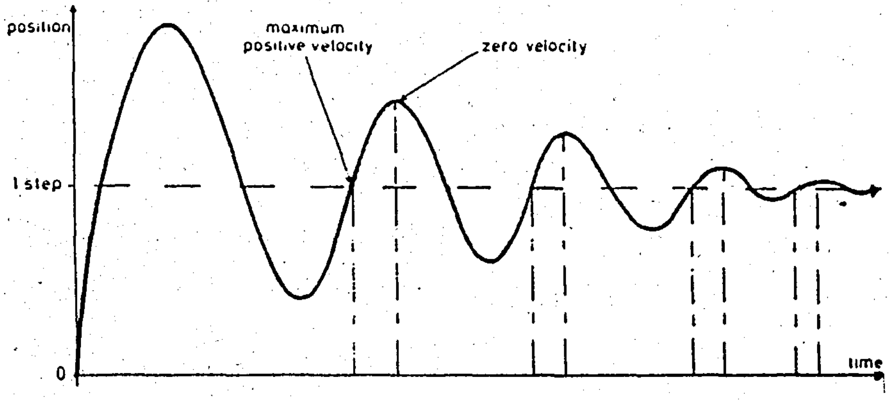


Figure 5.1. Regions of the single step response in which phase switching leads to resonance.

5.4. Conclusion

If it is summarized, the important points to drive a stepping motor are the followings.

a. Drive system has very much effect on the behaviour of the motor. If the motor is to be driven in a wide speed range, chopper drive system should be used. It doesn't need any software and short software means more reliable speed adjustment. If the load and speed is constant and the speed is in the range of start/stop speed of the motor, then a conventional unipolar drive system with a constant dC supply can be used.

b. If a software-intensive microprocessor control is to be used, control program should be shortened. It provides more reliable timing in a wider speed range. If it is possible external clock-timer chip (CTC) should be used then acceleration and deceleration will be more linear and the control software will be also shortened since the delay counts will be omitted.

In application requiring control of several other devices the hardware-intensive approach should be preffered.

c. In applications requiring frequent accurate positioning the step response of the motor is very important and the poorly damped response can be a big problem. For example, if a stepping motor is used to drive the carriage of a teletype then the system must come to rest for the printing of each letter. The operating speed of the teletype is limited by the time taken for the system to settle to within the required accuracy at each letter position. For this applications, either a motor having some type damping should be selected or a carefully designed drive system (for that load-speed combination) should be used.

Stepping motors offer many advantages as an actuator in a digitally controlled positioning system. It is easily interfaced with a microcomputer or microprocessors to provide opening, closing, rotating, reversing, cycling and highly accurate positioning in a variety of applications. Robots and CNC machines use the stepping motor. Then, we can say that " stepping motors have the future"

APPENDIX A

MONITOR PROGRAM OF THE M. COMPUTER

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC        COMMENTS
0000  0000  ST     ØØ         NOP           ; Start here after RESET
0001  0001         C3 50 01   JP  INT       ; Jump to initilization
0002                                          routine
0003  0004         CØ F9 A4   0,1,2         ; Display Look Up Table
0004  0007         BO 99 92   3,4,5
0005  000A         82 F8 80   6,7,8
0006  000D         90 88 83   9,A,B
0007  0010         C6 A1 86   C,D,E
0008  0013         8E A3 8C   F,O,P
0009  0016         AF C1 C7   r,U,L
0010  0019         AB 91 10   n,y,.
0011                This routine output the seven segments
0012                code of the data which will be displayed:
0013  0020  ODC    1A         LD A,(DE)       DE holds the data add.
0014  0021         01 04 00   LDBC,SADC       Disp. Code Tab. Start Ad
0015  0024         81         ADDC            is on BC.
0016  0025         4F         LD C, A       ; Code Add. is found and
0017  0026         OA         LD A,(BC)       then it is outputted.
0018  0027         D3 00      OUT 00,A
0019  0029         C3 2D 01   JP MLP        ; Jump to main disp.loop.
0020                The following routine provides a delay bet-
0021                ween successive digits providing nonflashing
0022                display.(there will be also no ghost from
                    previous digit):
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC       COMMENTS
0023  0030  DISDL  3E BF      LD A, DCON   ; DCON = Delay Constant
0024  0032  LP     3D         DECH
0025  0033         C2 32 00   JP NZ LPI
0026  0036         C9         RET          ; Ret if DCON = 0
0027                Multible key input is prevented with the
0028                use of n-key lock out facility.
0029  003C  LOUT   3A FB OF   LD A,(PKINP); Check whether there
0030  003F         B7         OR A           is a previous key input
0031  0040         CA 3D 01   JP Z DISP      or not.
0032  0043         3A EF OF   LD A,(KIC)
0033  0046         E1         POP HL       ; Whether the key pressed
0034  0047         BD         CP L           is the same with previou
0035  0048         CA 4F 00   JP Z RKINP     detection or not.
0036  004B         E5         PUSH HL
0037  004C         C3 3D D1   JP DISP
0038  004F RKINP   E5         PUSH HL
0039  0050         3E 00      LD A, 00     ; reset previous key inpu
0040  0052         32 FB ØF   LD PKINP,A     register (PKINP)
0041  0055         C3 3D 01   JP DISP      ; to display routine
0042                HL is address pointer, (HL) is data which
0043                will be displayed. The content of HL and (HL)
0044                are firstly loaded in to temporary register and
0045                then, they are outputted from here and
0046                displayed.
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC       COMMENTS
0047  0070  LDTMP  32 FA OF   LD(TTMP),A   ; the data on A is
0048  0073         E6 OF      AND OF         loaded to temp. regis-
0049  0075         DD 77 00   LD(IX+Ø),A     ters as two 4 bit data
0050  0078         3A FA OF   LD A,(TTMP)    being  H nibble and
0051  007B         E6 FO      AND FO         L nibble
0052  007D         CB 3F      SRL A
0053  007F         CB 3F      SRL A
0054  0081         CB 3F      SRL A
0055  0083         CB 3F      SRL A
0056  0085         DD 77 01   LD(IX+01),A
0057  0088         C9         RET
0058               The function of the key (+) is provided.
0059  0096  INC    RA F6 OF   LD HL,(HTMP)
0060  0099         23         INC HL       ; the address pointer is
0061  009A         22 F6 OF   LD(HTMP),HL    incremented.
0062  009D         C3 E5 01   JP DISPR
0063               The function of the key (-) is provided.
0064  00A0  DEC    2A F6 OF   LD HL,(HTMP)
0065  00A3         2B         DEC HL       ; the address pointer is
0066  00A4         22 F6 OF   LD(HTMP),HL    decremented.
0067  00A7         C3 E5 01   JP DISPR
0068               The cursor is added to the related data.
0069  00B0  ODPC   1A         LD A, (DE)   ; data from temp.registers
0070  00B1         01 04 00   LD BC,0004   ; BC= start address of cod
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC        COMMENTS
0071  00B4         81           AOD C
0072  00B5         4F           LD C,A
0073  00B6         0A           LD A,(BC)
0074  00B7         CB BF        RES A,7        ; put the point (cursor)
0075  00B9         D3 00        OUT 00,A       ; output the data
0076  00BB         C3 2D 01     JP MLP
0077                Main display and key recognition routine.
0078  00C0  PON    21 00 00     LD HL,0000     ; some temporary registers
0079  00C3         22 FD 0F     LD(EPC),HL     ; are initialized
0080  00C6         22 F8 0F     LD(CURSOR),HL
0081  00C9         22 FB 0F     LD(KINP),HL
0082  00CC         31 E0 0F     LD 3P,0FDØ     ; initialize stack pointer
0083  00CF         3E 88        LD A, 88       ; Control word of PIO
0084  00D1         D3 03        OUT 03, A
0085  00D3         21 00 08     LD HL,0800     ; reset value of add.disp.
0086  00D6         22 F6 0F     LD(HTMP),HL
0087  00D9  DISP   2A F6 0F     LD HL,(HTMP)
0088  00DC         3A F9 0F     LD A,(EP)
0089  00DF         B7           OR A
0090  00E0         C2 EB 00     JP NZ LPX
0091  00E3        DD 21 F2 0F   LD IX,TMPØD
0092  00E7         7E           LD A,(HL)
0093  00E8         CD 70 00     CALL LDTMP
0094  00EB  LPX    7D           LD A, L
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC        COMMENTS
0095  00EC         DD 21 F2 OF  LD IX,TMP1L
0096  00F0         CD 70 00     CALL LDTMP
0097  00F3         7C           LD A, H
0098  00F4         DD 21 F4 OF  LD IX,TMP2H
0099  00F8         CD 70 00     CALL LDTMP
0100  00FB  EBSTA  2E 20        LD L, 2Ø        ; L is digit pointer.
0101  00FD         11 F6 OF     LD DE, OFF6
0102  0100  OD     1D           DEC E
0103  0101         7B           LD A, E
0104  0102         FE EF        CP EF           ; Whether the 6 digit disp.
0105  0104         CA D9 OF     JP Z DISP         is completed or not.
0106  0107         E6 OF        AND OF
0107  0109         C2 19 01     JP NZ LPO
0108  010C         DD 46 04     LD B,(IX+04)
0109  010F         BO           OR B
0110  0110         C2 16 01     JP NZ LP1
0111  0113         C3 BO 00     JP ODP
0112  0116  LP1    C3 20 00     JP ODC          ; jump to output data
0113  0119  LPO    FE 02        CP 02             code.
0114  011B         C2 2A 01     JP NZ LP
0115  011E         DD 46 04     LD B,(IX+04)
0116  0121         3E 00        LD A, 00
0117  0123         BO           OR B
0118  0124         CA 2A 01     JP Z LP2
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS
0119  0129         C3 B0 00   JP ODP
0120  012A  LP2    C3 20 00   JP ODC
0121  012D         7D         LD A, L
0122  012E         2F         CPL
0123  012F         D3 01      OUT 01,A      ; out digit select code
0124  0131         3A FD OF   LD A,(EPC)
0125  0134         B7         OR A
0126  0135         C2 47 01   JP NZ DECA
0127  0138         D5         PUSH DE       ; preserve DE data pointer
0128  0139         E5         PUSH HL       ; preserve HL digit pointer
0129  013A         C3 80 01   JP KR         ; jump to key recognition
0130  013D         E1         POP HL
0131  013E         D1         POP DE
0132  013F         CD 30 00   CALL DBPDL    ; call disp. delay
0133  0142  SRL    CB 3D      SRL L         ; new digit
0134  0144         C3 00 01   JP OD
0135  0147  DECA   3D         DEC A
0136  0148         C8         RET Z
0137  0149         32 FD OF   LD(EPC),A
0138  014C         C3 3F 01   JP SRL
0139                          This routine checks whether there is a key
0140                          stroke or not. If there is, related func-
0141                          tion is provided.
0142  0180  KR     DB 02      INA, 02       ; port C is checked.
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS
0143  0182         E6 OF      AND OF
0144  0184         CA 3C 00   JP Z KINPC    ; jump if zero to key inp.
0145  0187         CD 30 00   CALL KDL        check routine.
0146  018A         DB 02      IN A, 02      ; check for multible entry.
0147  018C         E6 FO      AND FO
0148  018E         CA 3C 00   JP Z KINPC
0149  0191         06 00      LD B, 00
0150  0193  LX     04         INC B
0151  0194         B7         OR A          ; clear carry
0152  0195         CB 07      RLC A         ; check which row
0153  0197         D2 98 01   JP NC LX
0154  019A         E1         POP HL
0155  019B         7D         LD A, L       ; find which column
0156  019C         E5         PUSH HL
0157  019D         OE 00      LD C, 00
0158  019F  LY     OC         INC C
0159  01A0         B7         OR A          ; clear carry
0160  01A1         CB 07      RLC A
0161  01A3         D2 9F 01   JP NC LY
0162  01A6         CB 20      SLA B
0163  01A8         CB 20      SLA B
0164  01AA         CB 20      SLA B
0165  01AC         CB 20      SLA B
0166  01AE         78         LD A, B
```

```
LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS
01AF         B1         OR C
01B0         4F         LD C, A       ; the key code add on C
01B1         C3 86 02   JP KPV        ; check the key with
01B4         06 07      LD B, 07        the previous one.
01B6         3E 17      LD A, 17      ; check whether the key
01B8         B9         CP C            is a function key or data
01B9         CA 96 00   JP Z (+)      ; INC key.
01BC         3C         INC A
01BD         B9         CP C
01BE         CA A0 02   JP Z CON      ; conditions key.
01C1         3E 27      LD A, 27
01C3         B9         CP C
01C4         CA A0 00   JP Z,(-)      ; DEC key.
01C7         3C         INC A
01C8         B9         CP C
01C9         CA C7 05   JP Z,CAL      ; calculation key.
01CC         3E 37      LD A, 37
01CE         B9         CP C
01CF         CA 70 02   JP Z CUR      ; cursor memory/data change
01D2         3C         INC A
01D3         B9         CP C
01D4         CA 82 06   JP Z DRIVE    ; drive key
01D7         3E 47      LD A, 47
01D9         B9         CP C
```

```
LOC.  LABEL   OBJ.CODE    MNEMONIC       COMMENTS
01DA          CA 60 02    JP Z RUN     ; RUN key
01DD          3C          INC A
01DE          B9          CP C
01DF          CA 9E 06    JP Z INT     ; initialize motor pos.
01E2          CD FF 01    CALL RDA     ; replace data or add.
01E5  RETKR   C3 3D 01    JP DISIPOP.    portion of the display.
              The following routine replaces first data
              or add. digit by shifting previous digit
              values to the left.
01FF  RDA     3A F8 OF    LD A, CURSOR; determine whether the
0202          B7          OR A           data or address digit
                                                   re
0203          CA 44 02    JP Z RD        values are dreplaced
0206         DD 21 F2 OF  LD IX,OFF2   ; replace address digit
020A          DD 7E 02    LD A,(IX+02)   content. In order to
020D          DD 77 03    LD(IX+03),A    replace first digit,shift
0210          DD 7E 01    LD A,(IX+01)   all digit values to
0213          DD 77 02    LD(IX+02),A    the left ones.
0216          DD 7E 00    LD A, (IX+00)
0219          DD 77 01    LD(IX+01),A
021C          OA          LD A, (BC)
021D          DD 77 00    LD(IX+00),A
0220          DD 46 01    LD B, (IX+01)
0223          CB 20       SLA B
0225          CB 20       SLA B
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC         COMMENTS
0215  0227         CB 20      SLA  B
0216  0229         CB 20      SLA  B
0217  022B         BO         OR  B
0218  022C         DD 77 04   LD(IX+04),A  ;  now new data is placed
0219  022F         DD 7E 02   LD A,(IX+02)    in to first digit of the
0220  0232         DD 46 03   LD B,(IX+03)    address diplay.
0221  0235         CB 20      SLA,B
0222  0237         CB 20      SLA  B
0223  0239         CB 20      SLA  B
0224  023B         CB 20      SLA  B
0225  023D         BO         OR  B
0226  023E         DD 77 05   LD(IX+05),A
0227  0241         C3 56 02   JP  PET
0228  0244  RD     2A F6 OF   LD HL,(HIMP);   Replace the first digit
0229  0247         7E         LD A,(HL)       of the data display and
0230  0248         CB 27      SLA  A          shift the previous value
0231  024A         CB 27      SLA  A          to left digits.
0232  024C         CB 27      SLA  A
0233  024E         CB 27      SLA  A
0234  0250         08         EX AF
0235  0251         OA         LD A,(BC)
0236  0252         47         LD B, A
0237  0253         08         EX AF
0238  0254         BO         OR  B
```

```
LINE  LOC.  LABEL  OBJ.CODE  MNEMONIC     COMMENTS
0239  0255         77        LD(HL),A
0240  0256         C9        RET
0241               This routine provides program run facility
0242               to the user; starting from the displayed
0243               address value.
0244  0260  RUN    2A F6 OF  LD HL,(HIMP)
0245  0263         E9        JP (HL)
0246               The following routine changes the corsor's
0247               place.
0248  0270  CRSOR  3A F8 OF  LD A, CURSOR
0249  0273         B7        OR A
0250  0274         CA 7E 02  JP Z ADD    ; cursor to first add dig.
0251  0277         3D        DEC A
0252  0278         32 F8 OF  LD CURSOR,A
0253  027B         C3 82 02  JP RETKR
0254  027E  ADD    3C        INC A
0255  027F         32 F8 OF  LD CURSOR,A
0256  0282         C3 E5 01  JP RETKR
0257               This  routine holds the key value pressed if
0258               there is a key release detection before
0259               it.
0260  0286  KPV    3A FB OF  LD A, KINP
0261  0289         B7        OR A
0262  028A         CA 92 02  JP Z SKINP
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS

0263  020D         E1         POP HL
0264  020E         D1         POP DE
0265  020F         C3 42 01   JP DISP
0266  0212  SKIMP  3C         INC A
0267  0213         32 FB OF   LD(KIMP),A
0268  0216         E1         POP HL
0269  0217         7D         LD A, L      ; holds column address.
0270  0218         32 EF OF   LD(KPV),A      in KPV temporary RAM
0271  021B         E5         PUSH HL        register
0272  021C         C3 B4 01   JP
```

APPENDIX B

THE DRIVE PROGRAM OF THE STEPPING MOTOR

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC        COMMENTS
                   The following routine accepts the conditions
                   required to drive motor.
0001  02A0  CON    21 01 01   LD HL, CONST;  condition status
0002  02A3         22 F8 OF   LD(CSOR),HL
0003  02A6         3A FC OF   LD A,(ADD)   ; which condition
0004  02A9         B7         OR A
0005  02AA         C2 BB 02   JP NZ SP     ; if NZ to speed
0006  02AD         21 12 OD   LD HL, ODI2
0007  02B0         22 F0 OF   LD(TMPOL),HL
0008  0233         3E 01      LD A, 01
0009  02B5         32 FC OF   LD(ADD), A
0010  02B8         C3 E5 01   JP RET/DISP
0011  02BB  SP     3D         DEC A
0012  02BC         C2 D3 02   JP NZ SP     ; if NZ to step size
0013  02BF         2A F6 OF   LD HL,(HTMP)
0014  02C2         22 E9 OF   LD(DR),HL    ; load direction and
0015  02C5         21 11 05   LD HL, 0511    step mode
0016  02C8         22 F0 OF   LD(TMPDL),HL
0017  02CB         3E 02      LD A, 02
0018  02CD         32 FC OF   LD(ADD), A
0019  02D0         C3 E5 01   JP RET/DISP
0020  02D3  SS     3D         DEC A
0021  02D4         C2 EB 02   JP NZ
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC         COMMENTS

0022  02D7         2A F6 OF   LD HL,(HTMP)
0023  02DA         22 EB OF   LD (SP),HL   ; load speed (step/s)
0024  02DD         21 05 05   LD HL, 0505
0025  02E0         22 F0 OF   LD (TMPOL),HL
0026  02E3         3E 03      LD A, 03
0027  02E5         32 FC OF   LD(ADD),A
0028  02E8         C3 E5 01   JP RET/DISP
0029  02EB  EDR    3D         DEC A
0030  02EC         C2 06 03   JP NZ ACC
0031  02EF         2A FB OF   LD HL,(HTMP)
0032  02F2         22 ED OF   LD(SS), HL   ; load step size
0033  02F5         22 D6 OF   LD (S'S'), HL
0034  02F8         21 OC OA   LD HL, OAOC
0035  02FB         22 F0 OF   LD (TMPOL),HL
0036  02FE         3E 04      LD A, 04
0037  0300         32 FC OF   LD(ADD), A
0038  0303         C3 E5 01   JP RET/DISP
0039  0306         2A F6 OF   LD HL,(HTMP)
0040  0309         22 E7 OF   LD (AC),HL     ; load acceleration
0041  030C         21 00 00   LD HL, 0000
0042  030F         22 F8 OF   LD (Cursor),HL
0043  0312         22 FC OF   LD(ADD),HL
0044  0315         C3 E5 01   JP RET/DISP
```

The following routine decides whether the

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC     COMMENTS
0045               reset action is due to power on or the
0046               result of the pushed reset button.
0047  0151  RST    21 05 0A   LD HL,0A05  ; test location start
0048  0153  SRCH   2D         DEC 1         address.
0049  0154         7E         LD A,(HL)
0050  0155         B7         OR A
0051  0156         C2 60 01   JP NZ PON   ; to power-on action.
0052  0159         BD         CP L
0053  015A         CA 74 01   JP Z RBT    ; jump if zero to reset
0054  015D         C3 53 01   JP SRCH       button action.
0055  0160  PON    3E 00      LD A, 00
0056  0162  LD     77         LD(HL), A
0057  0163         2D         DEC L
0058  0164         C2 62 01   JP NZ LD
0059  0167         21 DA OF   LD HL,OFDA  ; initialize temporary
0060  016A         3E 00      LD A, 00      registers.
0061  016C         77         LD (HL), A
0062  016D         2C         INC L
0063  016E         77         LD (HL), A
0064  016F         2C         INC L
0065  0170         77         LD (HL), A
0066  0171         C3 C0 00   JP DISP     ; to display routine.
0067  0174  RBT    3E FF      LD A, FF    ; prog. power supply is
0068  0176         D3 08      OUT 08        adjusted to "0"volt .
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC       COMMENTS
0069  0178         C3 C0 00   JP DISP
0070               This routine calculates delays related with
0071               given step rates and places them in to proper
0072               tables.
0073  05C7  CAL    2A D6 OF   LD HL,(S'S')
0074  05CA         22 ED OF   LD (S'S'), HL
0075  05CD         CD 29 03   CALL ERROR   ; call error search routin
0076  05D0         21 00 OB   LD HL, OB00
0077  05D3         22 FE OF   LD(DELAD),HL
0078  05D6         21 00 00   LD HL,0000   ; initialize some tempo-
0079  05D9         22 E1 OF   LD(S), HL      rary registers.
0080  05DC         22 E5 OF   LD (S'), HL
0081  05DF         22 E3 OF   LD (SS'), HL
0082  05E2         22 DF OF   LD (T), HL
0083  05E5         CD 01 04   CALL DELCL   ; call delay calculation
0084  05E8         CD OF 04   CALL LDADD   ; call delay loading
0085  05EB         CD 31 04   CALL TOTAL   ; call step summing
0086  05EE         3A E7 OF   LD A,(ACC)
0087  05F1         B7         OR A
0088  05F2         CA C0 00   JP DISP      ; jump if zero to display
0089  05F5  LOOP   2A ED OF   LD HL,(SS)     routine
0090  05F8         C5         PUSH BC
0091  05F9         CD 60 03   CALL DVDB2   ; call division routine
0092  05FC         C1         POP BC         to calculate N_A and N_D.
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC        COMMENTS
0093  05FD  LOOP   2A E7 0F     LD HL,(ACC)
0094  0600         09           ADD BC        ; speed + acc.
0095  0601         44           LD B, H
0096  0602         4D           LD C, H
0097  0603        ED 5B E1 0F   LD DE,(s)
0098  0607         2A DF 0F     LD HL,(T)     ; check T>S
0099  060A         00           NOP             if  T>S, calculation
0100  060B         CB 7A        BIT 7,D         is over; if T<S, conti
0101  060D         C2 1B 06     JP NZ OP-       nue calculation.
0102  0610         7C           LD A, H
0103  0611         BA           CP D
0104  0612         CA 60 06     JP Z CPL
0105  0615         F2 19 03     JP P MONITOR
0106  0618         C3 20 06     JP ROUT       ; H<D then check L>E
0107  061B  CP-    CB 7C        BIT 7, H        or not
0108  061D         F2 19 03     JP P MONITOR; jump to monitor
0109  0620  ROUT   CD 01 04     CALL DELCL
0110  0623         CD 0F 04     CALL LDADD
0111  0626         CD 31 04     CALL TOTAL
0112  0629         3E 03        LD A, PMAX    ; check with max.speed
0113  062B         B8           CP B
0114  062C         F2 FD 05     JP P LOOP
0115  062F         CD 3D 04     CALL SS
0116  0632         C3 19 03     JP MONITOR    ; jump to monitor
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC        COMMENTS
0117  0660  CPL    7D         LD A, L
0118  0661         CB 7D      BIT 7, L
0119  0663         C2 6E 06   JP NZ CHK
0120  0666         CB 7B      BIT 7, E
0121  0668         CA 7A 06   JP Z CMP+
0122  066B         C3 20 06   JP ROUT         ; continue to calculate
0123  066E  CHK    CB 7B      BIT 7, E
0124  0670         CA 19 03   JP MONITOR
0125  0673         BB         CP E
0126  0674         FA 19 03   JP MONITOR
0127  0677         C3 20 06   JP ROUT         ; continue calculation
0128  067A  CMP+   BB         CP E
0129  067B         F2 19 03   JP P MONITOR
0130  067E         C3 20 06   JP ROUT
0131               The following routine is used to obtain $N_A$
0132               (S, accelerating step size) and $N_D$(S', decele-
0133               rating step size)
0134  0360  DVDB2  11 00  00  LD DE,0000
0135  0363         01 02 00   LD BC,0002
0136  0366         2A ED OF   LD HL,(SS)
0137  0369         CB 45      BIT Ø, L
0138  036B         CA 7C 03   JP Z EVEN
0139  036E         2D         DEC L
0140  036F         CD 58 00   CALL DIV
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC       COMMENTS

0141  0372         ED 53 E5 OF  LD(S'),DE
0142  0376          13          INC DE
0143  0377         ED 53 E1 OF  LD(S),DE      ; S=S'+1 if SS is add.
0144  037B          C9          RET
0145  037C  EVEN    CD 58 00    CALL DIV
0146  037F         ED 53 E5 OF  LD(S'), DE
0147  0383         ED 53 E1 OF  LD(S),DE      ; S = S'
0148  0387          C9          RET
0149                HL content is divided by BC content and
0150                result is on DE register pair.
0151  0058  DIV     B7          OR A          ; clear carry
0152  0059  LPDIV   ED 42       SB BC
0153  005B          D8          RET C
0154  005C          13          INC DE
0155  005D          C3 59 00    JP LPDIV
0156                TOTAL(T) temporary register holds the current
0157                step size calculated in CALC routine
0158  0431  TOTAL   2A DF OF    LD HL,(T)
0159  0434          09          ADD BC
0160  0436          22 DF OF    LD (T), HL
0161  0439          C9          RET
0162                In the CALC routine if maximum speed is
0163                reached with N steps <N_A, then SS (step
0164                size to be run with const.max. speed) should
0165                be calculated as follows.
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC        COMMENTS
0166  043D    SS    2A E1 OF    LD HL,(S)      ; NA is on HL
0167  0440         ED 4B DF OF  LD BC,(T)      ; n is on BC
0168  0444          ED 43 E1 OF LD (S),BC      ; (S) and (S') is loaded
0169  0448         ED 43 E5 OF  LD(S'),BC        with  n.
0170  044C          B7          OR A           ; clear carry
0171  044D          ED 42       SUB BC         ; NA-n = (SS)
0172  044F          2B          DEC HL
0173  0450          22 E3 OF    LD (SS),HL
0174  0453          C9          RET
0175                The following routine checks the conditions
0176                entered from keyboard and if there is impossibl
0177                condition, related error message is displayed.
0178  0329  ERROR ED 4B EB OF   LD BC,(SPEED)
0179  032D          79          LD A, C
0180  032E          BO          OR B
0181  032F          CA EB 01    JP Z Error 0; if speed is zero
0182  0332          78          LD A, B
0183  0333          B7          OR A
0184  0334          C2 3D 03    JP NZ CON
0185  0337          79          LD A, C
0186  0338          FE 01       CP 01
0187  033A          CA EB 01    JP Z Error 0; if speed is smaller
0188  033D  CON     78          LD A, B         than 2 step/sec.
0189  033E          FE 05       CP 05
0190  0340          F2 F4 01    JP P Error 1; if speed is higher
0191  0343          CB 7F       BIT 7, A        than 04FF step/sec.
```

```
LOC.  LABEL  OBJ.CODE  MNEMONIC      COMMENTS
0345         C2 F4 01  JP NZ Error 1
0348         3A E7 OF  LD A,(ACC)    ; check acceleration
034B         B7        OR A
034C         C8        RET Z
034D         FE 7F     CP 7F         ; if acc.is higher
034F         F2 57 02  JP P Error2     then 7E step/sec^2,
0352         CB 7F     BIT 7,A         display Error2 message
0354         C2 57 02  JP NZ Error 2
0357         2A ED OF  LD HL,(SS)    ; if an acc. is
035A         7D        LD A, L         given without giving
035B         B4        OR H            step size, display
035C         CA 67 02  JP Z Error 3    Error 3 message.
035F         C9        RET
                   This routine calculates speed related delay
                   contants.
0401  DELCAL B7        OR A          ; clear carry.
0402         21 FF FF  LD HE,FFFF
0405         11 00 00  LD DE,0000
0408  LP     ED 42     ZBC BC        ; speed constant is
040A         D8        RET C           on BC.
040B         13        INC DE        ; delay constant is on
040C         C3 08 04  JP LP           DE register pair.
                   This routine loads the delay constants into
                   related tables.
040F  LDADD  EB        EX DE,HL      ; double the calculated
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC         COMMENTS
0217  0410         29         ADD HL           delay constant.
0218  0411         EB         EX DE, HL
0219  0412         3E 02      LD A, 02
0220  0414         2A FE OF   LD HL,(OFFE);    start address of
0221  0417  LPW    73         LD(HL), E        table is on HL
0222  0418         23         INC HL           register pair.
0223  0419         72         LD(HL), D
0224  041A         23         INC HL
0225  041B         71         LD (HL),C      ; repeat number of the
0226  041C         23         INC HL           delay constant is on
0227  041D         70         LD(HL),B         BC register pair.
0228  041E         3D         DEC A
0229  041F         CA 2A 04   JP Z DC
0230  0422         2A FE OF   LD HL,(OFFE)
0231  0425         24         INC H          ; deceleration table
0232  0426         24         INC H            address adjustment.
0233  0427         C3 17 04   JP LPW
0234  042A  DC     25         DEC H
0235  042B         25         DEC H
0236  0423         23         INC HL
0237  042D         22 FE OF   LD (OFFE),HL
0238  0430         C9         RET
0239               Error message   first bytes loading are perfor-
0240               med in the following routines.
0241  01EB  ERO    21 FO OF   LD HL, OFF0
```

```
LINE  LOC.  LABEL  OBJ.CODE  MNEMONIC       COMMENTS
0242  01EE         3E 00     LD A, 00     ; ERROR 0 message.
0243  01FO         77        LD(HL),A
0244  01F1         C3 8C 03  JP ERROR
0245  01F4  ER1    21 FO OF  LD HL,OFFO
0246  01F6         3E 01     LD A, 01     ; ERROR 1 message.
0247  01F8         77        LD (HL),A
0248  01FA         C3 8C 03  JP ERROR
0249  0257  ER 2   2F FO OF  LD HL,OFFO
0250  025A         3E 02     LD A, 02     ; ERROR 2 message.
0251  025C         77        LD(HL), A
0252  025D         C3 8C 03  JP ERROR
0253  0267  ER3    21 FO OF  LD HL,OFFO
0254  026A         3E 03     LD A, 03     ; ERROR 3 message.
0255  026C         77        LD (HL), A
0256  026D         C3 8C 03  JP ERROR
0257               ERROR word is loaded in the following
0258               routine.
0259  038C  ERROR  2C        INC L
0260  038D         3E 12     LD A, "r"    ; "r" is loaded to
0261  038F         77        LD(HL),A       display table.
0262  0390         2C        INC L
0263  0391         3E 10     LD A, "O"    ; "O" is loaded to
0264  0393         77        LD(HL), A      display table.
0265  0394         2C        INC L
0266  0395         3E 12     LD A, "r"    ; "r" is loaded to
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS
0267  0397         77         LD(HL),A       display table.
0268  0398         2C         INC L
0269  0399         77         LD(HL),A
0270  039A         2C         INC L
0271  039B         3E OE      LD A,"E"     ; "E" is loaded to
0272  039D         77         LD(HL),A       display table.
0273  039E         C3 BD 03   JP LP DISP   ; jump to display
                                             "ERROR" message.
0274               "READY" word is loaded in to the display
0275               look-up table and displayed three times.
0276  03A3  READY  21 FO OF   LD HL, OFFO
0277  03A6         3E 17      LD A, "."
0278  03A8         77         LD(HL),A     ; point is loaded into
0279  03A9         2C         INC L          display look-up table.
0280  03AA         3E 16      LD A, "y"
0281  03AC         77         LD(HL),A     ; "y" is loaded into
0282  03AD         2C         INC L          display look-up table.
0283  03AE         3E OD      LD A, "d"
0284  03B0         77         LD(HL),A     ; "d" is loaded.
0285  03B1         2C         INC L
0286  03B2         3E OA      LD A, "A"
0287  03B4         77         LD(HL),A     ; "A" is loaded.
0288  03B5         2C         INC L
0289  03B6         3E OE      LD A, "E"
0290  03B8         77         LD(HL),A     ; "E" is loaded.
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC        COMMENTS
0291  03B9         2C         INC L
0292  03BA         3E 12      LD A, "r"
0293  03BC         77         LD(HL),A      ;"r" is loaded.
0294  03BD  LPDSP  3E 03      LD A, 03      ; flashing repeat
0295  03BF  LPX2   32 DE OF   LD(RC),A        number.
0296  03C2         3E FF      LD A, FF      ; error display delay.
0297  03C4  LPX    32 DD OF   LD(ERC),A
0298  03C7         3E 06      LD A, 06      ; digit counter.
0299  03C9         32 FD OF   LD (EPC),A
0300  03CC         CD FB 00   CALL EPS      ; display message.
0301  03CF         CD 30 00   CALL DISP DEL
0302  03D2         CD DD OF   LD A,(ERC)
0303  03D5         3D         DEC A
0304  03D6         C2 C4 03   JP NZ LPX
0305  03D9  LPX1   32 DD OF   LD(ERC), A
0306  03DC         CD 30 00   CALL DISPDEL; display blank.
0307  030F         3A DD OF   LD A,(ERC)
0308  03E2         3D         DEC A
0309  03E3         C2 D9 03   JP NZ LPX1
0310  03E6         3A DE OF   LD A,(ERC)
0311  03E9         3D         DEC A
0312  03EA         C2 BF 03   JP NZ LPX2
0313  03ED         3E FF      LD A,FF       ;adjust power to zero.
0314  03EF         D3 08      OUT 08,A
```

```
LOC.  LABEL   OBJ.CODE   MNEMONIC       COMMENTS
03F1          C3 C0 00   JP MONITOR    ; after three flashing.
              "RUN" routine outputs the drive codes accor-
              ding to the given conditions.
0456  RUN     3E 03      LD A, 03      ; start power code.
0458          D3 08      OUT 08,A      ; to programmable
045A          16 07      LD D, 07        power supply.
045C          CD A4 04   CALL LCD      ; DE register pair is
045F          3A DC 0F   LD A,(LCA)      the pointer of the
0462          5F         LD E, A         drive codes.
0463  DR      3A EA 0F   LD A,(DR)     ; Check direction.
0466          B7         OR A
0467          CA 83 04   JP Z CW       ; if zero, clock wise.
046A          1D         DEC E
046B          3A E9 0F   LD A,(SM)     ; check step mode.
046E          B7         OR A
046F          C2 73 04   JP NZ ROUT1   ; if NZ, half stepping.
0472          1D         DEC E         ; counter clock-wise.
0473  ROUT 1  3E FF      LD A, FF      ; check start of drive
0475          B8         CP E            table.
0476          CA 7E 04   JP Z TOUT
0479          3D         DEC A
047A          BB         CP E
047B          C2 99 04   JP NZ OUT
047E  TOUT    1E 08      LD E, 08
0480          C3 63 04   JP DR
```

```
LOC.   LABEL   OBJ.CODE   MNEMONIC        COMMENTS
0483   CW      1C         INC E         ; clock-wise.
0484           3A E9 0F   LD A,(SM)     ; check step mode.
0487           B7         OR A
0488           C2 8C 04   JP NZ ROUT2
048B           1C         INC E
048C   ROUT2   3E 08      LD A, 08      ; check end of drive
048E           BB         CP E            table.
048F           CA 97 04   JP Z INT 2    ; if zero, initialize
0492           3C         INC A           pointer.
0493           BB         CP E
0494           C2 99 04   JP NZ OUT
0497   INT2    1E 0D      LD E, 00      ; initialize pointer.
0499   OUT     CD D5 04   CALL OUT
049C           C3 A0 0Z   JP 1HS        ; check intermediate
                                          half stepping.
                       Previous step mode and last code address
                       are found in the following routine.
04A4   LCAD    3A DB 0F   LD A,(PSM)
04A7           DD BE 0E   CP(IX+0E)
04AA           C8         RET Z
04AB           B7         OR A          ; half step after
04AC           C8         RET Z           full step.
04AD           3A DC 0F   LD A,(LCA)    ; full step after
04B0           CB 4F      BIT 1, A        half step.
04B2           C8         RET Z
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC       COMMENTS
0364  04B3         3A EA OF     LD A,(DR)    ; check direction and
0365  04B6         B7           OR A           correct the LCA.
0366  04B7         CA BE 04     JP Z DLCA
0367  04BA         DD 34 01     INC(IX+01)
0368  04BD         C9           RET
0369  04BE  DLCA   DD 35 01     DEC(IX+01)
0370  04C1         C9           RET
0371               The following routine outputs the code
0372               pointed by (DE) and decrement the step
0373               size to check whether it is completed or
0374               not.
0375  04D5  OUT    1A           LD A,(DE)    ; drive code is on A.
0376  04D6         D3 02        OUT 02,A
0377  04D8         7B           LD A,E       ; LCA = E(last code
0378  04D9         32 DC OF     LD(LCA),A      address),reserve LCA.
0379  04DC         3A E7 OF     LD A,(ACC)   ; check acceleration.
0380  04DF         B7           OR A         ; set flags.
0381  04E0         CA 40 06     JP Z SSDEC   ; no acceleration.
0382  04E3         ED 4B E1 OF LD BC,(S)     ; there is acceleration.
0383  04E7         79           LD A,C       ; check s  is zero.
0384  04E8         BO           OR B
0385  04E9         C2 04 05     JP NZ SDEC   ; if not, decrement S.
0386  04EC         ED 4B E3 OF LD BC,(SS')   ; check S' S' is zero.
0387  04F0         79           LD A,C
0388  04F1         BO           OR B
```

```
LINE  LOC.   LABEL  OBJ.CODE     MNEMONIC        COMMENTS
0389  04F2          C2 0A 05     JP NZ SSDC    ; if not, decrement SS'.
0390  04F5          ED 4B E5 OF  LD BC,(S')    ; check S' is zero.
0391  04F9          79           LD A,C
0392  04FA          BO           OR B
0393  04FB          C2 10 05     JP Z SVST     ; if it is zero, then
0394  04FE          OB           DEC BC          go to save status
0395  04FF         ED 43 E5 OF   LD(S'),BC       routine, if not,
0396  0503          C9           RET             decrement S':
0397  0504   SDEC   OB           DEC BC        ; decrement "S".
0398  0505         ED 43 E1 OF   LD (S),BC
0399  0509          C9           RET
0400  050A   SSDC   OB           DEC BC        ; decrement "S".
0401  050B         ED 43 E3 OF   LD (SS),BC
0402  050F          C9           RET
0403  0510   SVST   7B           LD A, E       ; save LCA.
0404  0511          32 DC OF     LD(LCA),A
0405  0514          3A E9 OF     LD A,(SM)     ; save step mode.
0406  0517          32 DB OF     LD(PSM), A
0407  051A-E        00           NOP
0408  051F          CD C5 04     CALL STOP     ; stop power is given.
0409  0522          3E 00        LD A, 00
0410  0524          32 F8 OF     LD(CURSOR),A
0411  0527          C3 A3 03     JP READY      ; display "ready".
0412  0640   SSDEC  ED 4B ED OF  LD BC,(SS)    ; there is no
0413  0644          79           LD A,C          acceleration.
```

```
LINE  LOC.  LABEL  OBJ.CODE     MNEMONIC       COMMENTS
0414  0645         B0           OR B
0415  0646         C8           RET Z
0416  0647         0B           DEC BC
0417  0648       ED 43 ED 0F    LD (SS),BC     ; decrement SS if
0418  064C         79           LD A,C           it is not zero.
0419  064D         B0           OR B
0420  064E         C0           RET NZ
0421  064F         C3 10 05     JP SVST        ; jump to same status.
0422               The following routine is delay routine
0423               which provides timing of drive codes.
0424  0534  DELAY  3A E7 0F     LD A,(ACC)     ; check, there is acc.
0425  0537         B7           OR A             or not.
0426  0538         C2 49 05     JP NZ CHKS
0427  053B       ED 4B 00 0B    LD BC,(0B00);  delay constant on BC.
0428  053F         CD 50 07     CALL PWCD      ; call power code rout.
0429  0542  LP     0B           DEC BC         ; without acceleration.
0430  0543         79           LD A,C
0431  0544         B0           OR B
0432  0545         C2 42 05     JP NZ LP
0433  0548         C9           RET            ; return from non-acc.rou
0434  0549  CHKS ED 4B E1 0F    LD BC,(S)      ; with acceleration.
0435  054D         79           LD A,C
0436  054E         B0           OR B
0437  054F         C2 70 05     JP NZ ACCR     ; if S≠ 0, go to ACCR.
0438  0552       ED 4B E3 0F    LD BC,(SS')
```

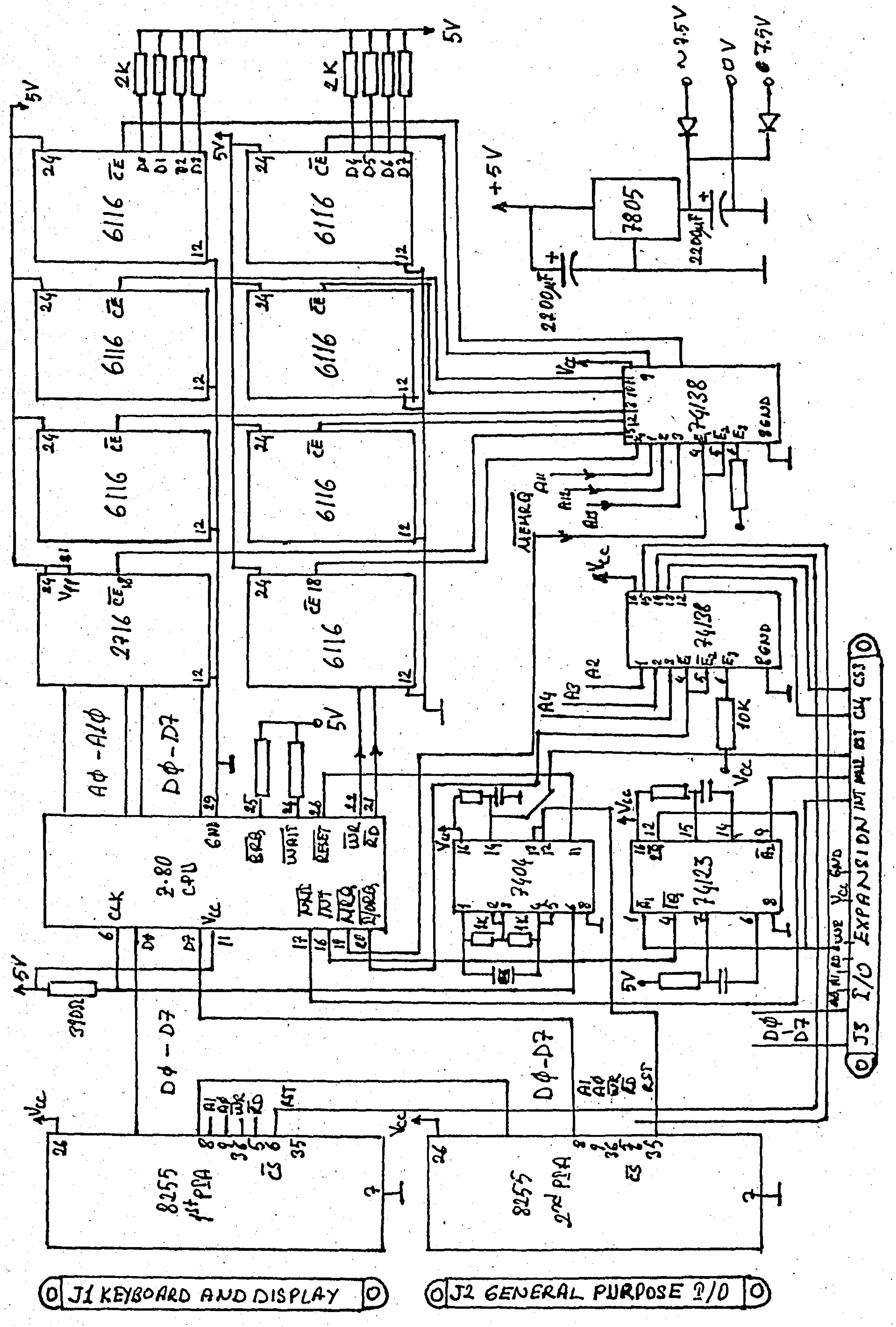
```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS
0439  0556         79         LD A,C      ; check SS' ≟ 0.
0440  0557         BO         OR B
0441  0558         CA 96 05   JP NZ DECR  ; if SS' ≠ 0, jump DECR.
0442  055B         01 3F 00   LD BC, SPMAX
0443  055E         CD 50 07   CALL PWCD
0444  0561  LPXX   OB         DEC BC      ; decrement SPMAX
0445  0562         79         LD A,C        until SPMAX = 0.
0446  0563         BO         OR B
0447  0564         C2 61 05   JP NZ LPXX
0448  0565         C9         RET
0449                          This routine follows the accelerating
0450                          profile.
0451  0570  PCCR   4E         LD C,(HL)   ; load delay constant.
0452  0571         23         INC HL      ; increment table
0453  0572         46         LD B,(HL)     pointer.
0454  0573         CD 50 07   CALL PWCD   ; adjust power.
0455  0576  LPY    OB         DEC BC      ; decrement delay
0456  0577         79         LD A,C        constant.
0457  0578         BO         OR B
0458  0579         C2 76 05   JP NZ LPY
0459  057C         23         INC HL
0460  057D         4E         LD C,(HL)   ; check repeat number.
0461  057E         23         INC HL
0462  057F         46         LD B,(HL)
0463  0580         OB         DEC BC      ; decrement repeat
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC       COMMENTS
0464  0581         79         LD A,C         number.
0465  0582         BO         OR B
0466  0583         CA 8C 05   JP Z INCHL
0467  0586         70         LD(HL),B     ; reload repeat number.
0468  0587         2B         DEC HL
0469  0588         71         LD(HL),C
0470  0589         2B         DEC HL       ; restrore the table
0471  058A         2B         DEC HL         pointer value.
0472  058B         C9         RET
0473  058C         23         INC HL
0474  058D         3E 00      LD A,00
0475  058F         32 09 OF   LD DCR1,A
0476  0592         C9         RET
0477                          This routine follows the decelerating
0478                          profile.
0479  0594         OE 08      LD C, 08
0480  0596         3A D8 OF   LD A,(DR60)  ; check decrement
0481  0599         B7         OR A
0482  059A         C2 AF 05   JP NZ RT
0483  059D         3E OB      LD A, OB     ; first deceleration.
0484  059F         32 D8 OF   LD(DR60),A   ; correct delay pointer
0485  05A2         BC         CP H           value on BC for dece-
0486  05A3         CA AB 05   JP Z INC       leration.
0487  05A6         3C         INC A
0488  05A7         BC         CP H
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC       COMMENTS
0489  05A8         C2 AF 05   JP NZ RT
0490  05AB  INC    24         INC H
0491  05AC         24         INC H
0492  05AD         0E 04      LD C,04
0493  05AF  RT     3A D9 0F   LD A,(DR61);  check decrement registe
0494  05B2         B7         OR A        ; if repeat number is
0495  05B3         C2 BF 05   JP NZ CALL    not zero, use the same
0496  05B6         06 00      LD B,00       delay constant.
0497  05B8         ED 42      SBC BC      ; adjust delay pointer.
0498  05BA         3E 01      LD A, 01
0499  05BC         32 D9 0F   LD(DR61),A
0500  05BF  CALL   CD 70 05   CALL ACCR   ; call delay and delay
0501  05C2         C9         RET           counter routine, then
                                            return.
0502               The following routine outputs power
0503               codes generated according to delay count
0504               value.
0505  0750  PPW    C5         PUSH BC     ; reserve delay count(DC)
0506  0751         E5         PUSH HL     ; reserve pointer.
0507  0752         21 d8 06   LD HL,06D8  ; HL is power table
0508  0755         78         LD A,B        pointer.
0509  0756         B7         OR A        ; set flags.
0510  0757         C2 76 07   JP NZ LPWR  ; if DC  FF; go low
0511  075A         CB 79      BIT 7,C       power routine.
0512  075C         C2 68 07   JP NZ CM    ; jump to C minus sub-
0513  075F         79         LD A,C        routine.
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC     COMMENTS
0514  0760  CPI    BE         CP(HL)      ; Compare DC with table
0515  0761         F2 9A 07   JP P HPWR   ; jump high power
0516  0764         2C         INC L         subroutine.
0517  0765         C3 60 07   JP CP1      ; continue to compare
0518  0768  CM     79         LD A,C      ; C minus subroutine
0519  0769         2F         CPL
0520  076A         4F         LD C,A      ; now C is positive
0521  076B  DECL   2D         DEC L       ; decrement power table
0522  076C         3E D0      LD A, D0      pointer and continue
0523  076E         BD         CP L          to compare.
0524  076F         CA 9A 07   JP Z HPWR
0525  0772         79         LD A, C
0526  0773         BE         CP (HL)     ; Compare DC with table
0527  0774         FA 9A 07   JP M HPWR
0528  0777         C3 6B 07   JP DECL
0529  077A  LPWR   CB 7F      BIT 7,A     ; check DC  07FF
0530  077C         C2 89 07   JP NZ OA    ; JP power code OA.
0531  077F         FE 01      CP 01       ; if DC = 01XX,
0532  0781         CA 90 07   JP Z 08       power code is 08.
0533  0784         FE 02      CP 02       ; if DC = 02XX,
0534  0786         CA 95 07   JP Z 09       power code is 09
0535  0789  OA     3E OA      LD A, OA
0536  078B  OUT    D3 08      OUT 08, A   ; out power code to sup
0537  078D         E1         POP HL      ; restore delay pointer
0538  078E         C1         POP BC      ; restore delay count
```

```
LINE  LOC.  LABEL  OBJ.CODE   MNEMONIC      COMMENTS
0539  078F         C9         RET           value.
0540  0790  08     3E 08      LD A,08     ; power code is 08
0541  0792         C3 8B 07   JP OUT
0542  0795  09     3E 09      LD A, 08    ; power code is 09
0543  0797         C3 8B 07   JP OUT
0544  079A  HPWR   24         INC H       ; power code pointer
0545  079B         7E         LD A,(HL)   ; power code is from
0546  079C         C3 8B 07   JP OUT        table.
0547               Delay comparison and power code table
0548               are given in the following locations
0549               respectively.
      (06D0-06DF):7F 6A 5F 55 47 37 25 07 77 70 6A 65 5F 5A 55 3F
      (07D0-07DF):03 03 04 04 05 05 06 07 02 02 02 01 01 01 00 00
```



CONNECTION DIAGRAM OF THE MICROCOMPUTER

APPENDIX D

USER MANUAL OF THE KIT

The kit has 24 keys on keyboard. 16 of them are for hexadecimal data, 8 of them are function keys. The first four of function keys are related with operating system and the others are related with motor drive program.

0	4	8	C	+	Con.
1	5	9	D	-	Cal.
2	6	A	E	Cur.	Drv.
3	7	B	F	Run	Int.

Keyboard Layout

FUNCTION KEY:

+ Key: The address value which is displayed is incremented by one and data belongs to the new address value.

- Key: The address value which is displayed is decremented by one, and data belongs to the new address.

Cur. : It operates the CURSOR. If the cursor is on the first digit of data, then the data keys are used to change the data portion of the display. Cur. key changes the place of the cursor point.

Run : When it is pushed, the address value on the display is loaded to the program counter.

CON. : All conditions related with motor drive are entered with the succesive use of this key.

1st STROKE: dr is demanded

X	X	X	X.		d	r

Direction Step Mode Condition

Clock-Wise: 00 Full Step:00

Counter Clock-Wise: xx Half Step:xx

3rd STROKE: Previous condition is entered, SS is demande

X	X	X	X.		S	S

Step Size Condition

0000-FFFF, 0000 means limitless.

2nd STROKE: SS is entered, SP is demanded.

X	X	X	X.		S	P

Speed Condition

0002-04FF Step/sec.

04FF Step/sec. is the maximum start-stop speed in half step mode. For full step mode, it is about 0280 step/sec.

4th STROKE: SP is entered, AC is demanded.

0	0	X	X

Acceleration

A	C

Condition

a = 0000-007F Step/sec .

5th STROKE: Acceleration value is entered and program returns to monitor.

CAL.: Delay counts and acceleration table is calculated and after calaulation program returns to initial state.

DRV :Motor is started to run according to the previous calculation values.

INT :Initialize the motor position by running it 16 steps and then, computer knows the step position of the motor at that moment.

MESSAGES:

READY. : After given step size is completed, motor is stopped and "ready" message is displayed three times by flashing.

Error 0 : If the given speed is smaller than 0002 step/sec. then, Error 0 message is displayed three times by flashing.

Error 1 : If the given speed is greater than 04FF step/sec. then, Error 1 message is displayed three times by flashing.

Error 3 : If acceleration is given with no limit (Step size = 0000) then, Error 2 message is displayed three times by flashing.

<u>RESET KEY</u>: Reset can be also used as an emergency stop switch of the motor drive system.

<u>REVERSE KEY</u>:This key is used to reverse the direction of the motor. It is connected to INT pin of the CPU via a one shot chip (interrupt circuit) .

APPENDIX E

Z80 MACHINE CODE LISTING

OBJ CODE	SOURCE STATEMENT	
8E	ADC	A,(HL)
DD8E05	ADC	A,(IX+d)
FD8E05	ADC	A,(IY+d)
8F	ADC	A,A
88	ADC	A,B
89	ADC	A,C
8A	ADC	A,D
8B	ADC	A,E
8C	ADC	A,H
8D	ADC	A,L
CE20	ADC	A,n
ED4A	ADC	HL,BC
ED5A	ADC	HL,DE
ED6A	ADC	HL,HL
ED7A	ADC	HL,SP
86	ADD	A,(HL)
DD8605	ADD	A,(IX+d)
FD8605	ADD	A,(IY+d)
87	ADD	A,A
80	ADD	A,B
81	ADD	A,C
82	ADD	A,D
83	ADD	A,E
84	ADD	A,H
85	ADD	A,L
C620	ADD	A,n
09	ADD	HL,BC
19	ADD	HL,DE
29	ADD	HL,HL
39	ADD	HL,SP
DD09	ADD	IX,BC
DD19	ADD	IX,DE
DD29	ADD	IX,IX
DD39	ADD	IX,SP
FD09	ADD	IY,BC
FD19	ADD	IY,DE
FD29	ADD	IY,IY
FD39	ADD	IY,SP
A6	AND	(HL)
DDA605	AND	(IX+d)
FDA605	AND	(IY+d)
A7	AND	A
A0	AND	B
A1	AND	C
A2	AND	D
A3	AND	E
A4	AND	H
A5	AND	L
CB39	SRL	C
CB3A	SRL	D
CB3B	SRL	E
CB3C	SRL	H
CB3D	SRL	L
96	SUB	(HL)
DD9605	SUB	(IX+d)
FD9605	SUB	(IY+d)
E620	AND	n
CB46	BIT	0,(HL)
DDCB0546	BIT	0,(IX+d)
FDCB0546	BIT	0,(IY+d)
CB47	BIT	0,A
CB40	BIT	0,B
CB41	BIT	0,C
CB42	BIT	0,D
CB43	BIT	0,E
CB44	BIT	0,H
CB45	BIT	0,L
CB4E	BIT	1,(HL)
DDCB054E	BIT	1,(IX+d)
FDCB054E	BIT	1,(IY+d)
CB4F	BIT	1,A
CB48	BIT	1,B
CB49	BIT	1,C
CB4A	BIT	1,D
CB4B	BIT	1,E
CB4C	BIT	1,H
CB4D	BIT	1,L
CB56	BIT	2,(HL)
DDCB0556	BIT	2,(IX+d)
FDCB0556	BIT	2,(IY+d)
CB57	BIT	2,A
CB50	BIT	2,B
CB51	BIT	2,C
CB52	BIT	2,D
CB53	BIT	2,E
CB54	BIT	2,H
CB55	BIT	2,L
CB5E	BIT	3,(HL)
DDCB055E	BIT	3,(IX+d)
FDCB055E	BIT	3,(IY+d)
CB5F	BIT	3,A
CB58	BIT	3,B
CB59	BIT	3,C
CB5A	BIT	3,D
CB5B	BIT	3,E
CB5C	BIT	3,H
CB5D	BIT	3,L
CB66	BIT	4,(HL)
DDCB0566	BIT	4,(IX+d)
FDCB0566	BIT	4,(IY+d)
CB67	BIT	4,A
CB60	BIT	4,B
CB61	BIT	4,C
CB62	BIT	4,D
CB2B	SRA	E
CB2C	SRA	H
CB2D	SRA	L
CB3E	SRL	(HL)
DDCB053E	SRL	(IX+d)
FDCB053E	SRL	(IY+d)
CB3F	SRL	A
CB38	SRL	B
CB63	BIT	4,E
CB64	BIT	4,H
CB65	BIT	4,L
CB6E	BIT	5,(HL)
DDCB056E	BIT	5,(IX+d)
FDCB056E	BIT	5,(IY+d)
CB6F	BIT	5,A
CB68	BIT	5,B
CB69	BIT	5,C
CB6A	BIT	5,D
CB6B	BIT	5,E
CB6C	BIT	5,H
CB6D	BIT	5,L
CB76	BIT	6,(HL)
DDCB0576	BIT	6,(IX+d)
FDCB0576	BIT	6,(IY+d)
CB77	BIT	6,A
CB70	BIT	6,B
CB71	BIT	6,C
CB72	BIT	6,D
CB73	BIT	6,E
CB74	BIT	6,H
CB75	BIT	6,L
CB7E	BIT	7,(HL)
DDCB057E	BIT	7,(IX+d)
FDCB057E	BIT	7,(IY+d)
CB7F	BIT	7,A
CB78	BIT	7,B
CB79	BIT	7,C
CB7A	BIT	7,D
CB7B	BIT	7,E
CB7C	BIT	7,H
CB7D	BIT	7,L
DC8405	CALL	C,nn
FC8405	CALL	M,nn
D48405	CALL	NC,nn
C48405	CALL	NZ,nn
F48405	CALL	P,nn
EC8405	CALL	PE,nn
E48405	CALL	PO,nn
CC8405	CALL	Z,nn
CD8405	CALL	nn
3F	CCF	
BE	CP	(HL)
DDBE05	CP	(IX+d)
FDBE05	CP	(IY+d)
BF	CP	A
B8	CP	B
B9	CP	C
BA	CP	D
BB	CP	E
BC	CP	H
BD	CP	L
FE20	CP	n
EDA9	CPD	
EDB9	CPDR	
EDB1	CPIR	
EDA1	CPI	
2F	CPL	
27	DAA	
35	DEC	(HL)
DD3505	DEC	(IX+d)
FD3505	DEC	(IY+d)
3D	DEC	A
05	DEC	B
0B	DEC	BC
0D	DEC	C
15	DEC	D
1B	DEC	DE
1D	DEC	E
25	DEC	H
2B	DEC	HL
DD2B	DEC	IX
FD2B	DEC	IY
2D	DEC	L
3B	DEC	SP
F3	DI	
102E	DJNZ	e
FB	EI	
E3	EX	(SP),HL
DDE3	EX	(SP),IX
FDE3	EX	(SP),IY
08	EX	AF,AF'
EB	EX	DE,HL
D9	EXX	
76	HALT	
ED46	IM	0
ED56	IM	1
ED5E	IM	2
ED78	IN	A,(C)
ED40	IN	B,(C)
ED48	IN	C,(C)
ED50	IN	D,(C)
ED58	IN	E,(C)
ED60	IN	H,(C)
ED68	IN	L,(C)
34	INC	(HL)
DD3405	INC	(IX+d)
FD3405	INC	(IY+d)
3C	INC	A
04	INC	B
03	INC	BC
0C	INC	C
14	INC	D
13	INC	DE
1C	INC	E
24	INC	H
23	INC	HL
DD23	INC	IX
FD23	INC	IY
2C	INC	L
33	INC	SP
DB20	IN	A,(n)

OBJ CODE	SOURCE STATEMENT	
EDAA	IND	
EDBA	INDR	
EDA2	INI	
EDB2	INIR	
C38405	JP	nn
E9	JP	(HL)
DDE9	JP	(IX)
FDE9	JP	(IY)
DA8405	JP	C,nn
FA8405	JP	M,nn
D28405	JP	NC,nn
C28405	JP	NZ,nn
F28405	JP	P,nn
EA8405	JP	PE,nn
E28405	JP	PO,nn
CA8405	JP	Z,nn
382E	JR	C,e
302E	JR	NC,e
202E	JR	NZ,e
282E	JR	Z,e
182E	JR	e
02	LD	(BC),A
12	LD	(DE),A
77	LD	(HL),A
70	LD	(HL),B
71	LD	(HL),C
72	LD	(HL),D
73	LD	(HL),E
74	LD	(HL),H
75	LD	(HL),L
3620	LD	(HL),n
DD7705	LD	(IX+d),A
DD7005	LD	(IX+d),B
DD7105	LD	(IX+d),C
DD7205	LD	(IX+d),D
DD7305	LD	(IX+d),E
DD7405	LD	(IX+d),H
DD7505	LD	(IX+d),L
DD360520	LD	(IX+d),n
FD7705	LD	(IY+d),A
FD7005	LD	(IY+d),B
FD7105	LD	(IY+d),C
FD7205	LD	(IY+d),D
FD7305	LD	(IY+d),E
FD7405	LD	(IY+d),H
FD7505	LD	(IY+d),L
FD360520	LD	(IY+d),n
328405	LD	(nn),A
ED438405	LD	(nn),BC
ED538405	LD	(nn),DE
228405	LD	(nn),HL
DD228405	LD	(nn),IX
FD228405	LD	(nn),IY
ED738405	LD	(nn),SP
0A	LD	A,(BC)
1A	LD	A,(DE)
7E	LD	A,(HL)
DD7E05	LD	A,(IX+d)
FD7E05	LD	A,(IY+d)
3A8405	LD	A,(nn)
7F	LD	A,A
78	LD	A,B
79	LD	A,C
7A	LD	A,D
7B	LD	A,E
7C	LD	A,H
ED57	LD	A,I
7D	LD	A,L
3E20	LD	A,n
ED5F	LD	A,R
46	LD	B,(HL)
DD4605	LD	B,(IX+d)
FD4605	LD	B,(IY+d)
47	LD	B,A
40	LD	B,B
41	LD	B,C
42	LD	B,D
43	LD	B,E
44	LD	B,H
45	LD	B,L
0620	LD	B,n
ED4B8405	LD	BC,(nn)
018405	LD	BC,nn
4E	LD	C,(HL)
DD4E05	LD	C,(IX+d)
FD4E05	LD	C,(IY+d)
4F	LD	C,A
48	LD	C,B
49	LD	C,C
4A	LD	C,D
4B	LD	C,E
4C	LD	C,H
4D	LD	C,L
0E20	LD	C,n
56	LD	D,(HL)
DD5605	LD	D,(IX+d)
FD5605	LD	D,(IY+d)
57	LD	D,A
50	LD	D,B
51	LD	D,C
52	LD	D,D
53	LD	D,E
54	LD	D,H
55	LD	D,L
1620	LD	D,n
ED5B8405	LD	DE,(nn)
118405	LD	DE,nn
5E	LD	E,(HL)
DD5E05	LD	E,(IX+d)
FD5E05	LD	E,(IY+d)
5F	LD	E,A
58	LD	E,B
59	LD	E,C
5A	LD	E,D
5B	LD	E,E
5C	LD	E,H
5D	LD	E,L
1E20	LD	E,n
66	LD	H,(HL)
DD6605	LD	H,(IX+d)
FD6605	LD	H,(IY+d)
67	LD	H,A
60	LD	H,B
61	LD	H,C
62	LD	H,D
63	LD	H,E
64	LD	H,H
65	LD	H,L
2620	LD	H,n
2A8405	LD	HL,(nn)
218405	LD	HL,nn
ED47	LD	I,A
DD2A8405	LD	IX,(nn)
DD218405	LD	IX,nn
FD2A8405	LD	IY,(nn)
FD218405	LD	IY,nn
6E	LD	L,(HL)
DD6E05	LD	L,(IX+d)
FD6E05	LD	L,(IY+d)
6F	LD	L,A
68	LD	L,B
69	LD	L,C
6A	LD	L,D
6B	LD	L,E
6C	LD	L,H
6D	LD	L,L
2E20	LD	L,n
ED4F	LD	R,A
ED7B8405	LD	SP,(nn)
F9	LD	SP,HL
DDF9	LD	SP,IX
FDF9	LD	SP,IY
318405	LD	SP,nn
EDA8	LDD	
EDB8	LDDR	
EDA0	LDI	
EDB0	LDIR	
ED44	NEG	
00	NOP	
B6	OR	(HL)
DDB605	OR	(IX+d)
FDB605	OR	(IY+d)
B7	OR	A
B0	OR	B
B1	OR	C
B2	OR	D
B3	OR	E
B4	OR	H
B5	OR	L
F620	OR	n
EDBB	OTDR	
EDB3	OTIR	
ED79	OUT	(C),A
ED41	OUT	(C),B
ED49	OUT	(C),C
ED51	OUT	(C),D
ED59	OUT	(C),E
ED61	OUT	(C),H
ED69	OUT	(C),L
D320	OUT	(n),A
EDAB	OUTD	
EDA3	OUTI	
F1	POP	AF
C1	POP	BC
D1	POP	DE
E1	POP	HL
DDE1	POP	IX
FDE1	POP	IY
F5	PUSH	AF
C5	PUSH	BC
D5	PUSH	DE
E5	PUSH	HL
DDE5	PUSH	IX
FDE5	PUSH	IY
CB86	RES	0,(HL)
DDCB0586	RES	0,(IX+d)
FDCB0586	RES	0,(IY+d)
CB87	RES	0,A
CB80	RES	0,B
CB81	RES	0,C
CB82	RES	0,D
CB83	RES	0,E
CB84	RES	0,H
CB85	RES	0,L
CB8E	RES	1,(HL)
DDCB058E	RES	1,(IX+d)
FDCB058E	RES	1,(IY+d)
CB8F	RES	1,A
CB88	RES	1,B
CB89	RES	1,C
CB8A	RES	1,D
CB8B	RES	1,E
CB8C	RES	1,H
CB8D	RES	1,L
CB96	RES	2,(HL)
DDCB0596	RES	2,(IX+d)
FDCB0596	RES	2,(IY+d)
CB97	RES	2,A
CB90	RES	2,B
CB91	RES	2,C
CB92	RES	2,D
CB93	RES	2,E
CB94	RES	2,H
CB95	RES	2,L
CB9E	RES	3,(HL)
DDCB059E	RES	3,(IX+d)
FDCB059E	RES	3,(IY+d)

OBJ CODE	SOURCE STATEMENT	
CB9F	RES	3,A
CB98	RES	3,B
CB99	RES	3,C
CB9A	RES	3,D
CB9B	RES	3,E
CB9C	RES	3,H
CB9D	RES	3,L
CBA6	RES	4,(HL)
DDCB05A6	RES	4,(IX+d)
FDCB05A6	RES	4,(IY+d)
CBA7	RES	4,A
CBA0	RES	4,B
CBA1	RES	4,C
CBA2	RES	4,D
CBA3	RES	4,E
CBA4	RES	4,H
CBA5	RES	4,L
CBAE	RES	5,(HL)
DDCB05AE	RES	5,(IX+d)
FDCB05AE	RES	5,(IY+d)
CBAF	RES	5,A
CBA8	RES	5,B
CBA9	RES	5,C
CBAA	RES	5,D
CBAB	RES	5,E
CBAC	RES	5,H
CBAD	RES	5,L
CBB6	RES	6,(HL)
DDCB05B6	RES	6,(IX+d)
FDCB05B6	RES	6,(IY+d)
CBB7	RES	6,A
CBB0	RES	6,B
CBB1	RES	6,C
CBB2	RES	6,D
CBB3	RES	6,E
CBB4	RES	6,H
CBB5	RES	6,L
CBBE	RES	7,(HL)
DDCB05BE	RES	7,(IX+d)
FDCB05BE	RES	7,(IY+d)
CBBF	RES	7,A
CBB8	RES	7,B
CBB9	RES	7,C
CBBA	RES	7,D
CBBB	RES	7,E
CBBC	RES	7,H
CBBD	RES	7,L
C9	RET	
D8	RET	C
F8	RET	M
D0	RET	NC
C0	RET	NZ
F0	RET	P
E8	RET	PE
E0	RET	PO
C8	RET	Z
94	SUB	H
95	SUB	L
D620	SUB	n
AE	XOR	(HL)

OBJ CODE	SOURCE STATEMENT	
ED4D	RETI	
ED45	RETN	
CB16	RL	(HL)
DDCB0516	RL	(IX+d)
FDCB0516	RL	(IY+d)
CB17	RL	A
CB10	RL	B
CB11	RL	C
CB12	RL	D
CB13	RL	E
CB14	RL	H
CB15	RL	L
17	RLA	
CB06	RLC	(HL)
DDCB0506	RLC	(IX+d)
FDCB0506	RLC	(IY+d)
CB07	RLC	A
CB00	RLC	B
CB01	RLC	C
CB02	RLC	D
CB03	RLC	E
CB04	RLC	H
CB05	RLC	L
07	RLCA	
ED6F	RLD	
CB1E	RR	(HL)
DDCB051E	RR	(IX+d)
FDCB051E	RR	(IY+d)
CB1F	RR	A
CB18	RR	B
CB19	RR	C
CB1A	RR	D
CB1B	RR	E
CB1C	RR	H
CB1D	RR	L
1F	RRA	
CB0E	RRC	(HL)
DDCB050E	RRC	(IX+d)
FDCB050E	RRC	(IY+d)
CB0F	RRC	A
CB08	RRC	B
CB09	RRC	C
CB0A	RRC	D
CB0B	RRC	E
CB0C	RRC	H
CB0D	RRC	L
OF	RRCA	
ED67	RRD	
C7	RST	00H
CF	RST	08H
D7	RST	10H
DF	RST	18H
E7	RST	20H
EF	RST	28H
F7	RST	30H
FF	RST	38H
DE20	SEC	A,n
90	SUB	B
91	SUB	C
92	SUB	D
93	SUB	E

OBJ CODE	SOURCE STATEMENT	
DDCB05E6	SET	4,(IX+d)
FDCB05E6	SET	4,(IY+d)
CBE7	SET	4,A
CBE0	SET	4,B
CBE1	SET	4,C
CBE2	SET	4,D
CBE3	SET	4,E
CBE4	SET	4,H
CBE5	SET	4,L
CBEE	SET	5,(HL)
DDCB05EE	SET	5,(IX+d)
FDCB05EE	SET	5,(IY+d)
CBEF	SET	5,A
CBE8	SET	5,B
CBE9	SET	5,C
CBEA	SET	5,D
CBEB	SET	5,E
CBEC	SET	5,H
CBED	SET	5,L
CBF6	SET	6,(HL)
DDCB05F6	SET	6,(IX+d)
FDCB05F6	SET	6,(IY+d)
CBF7	SET	6,A
CBF0	SET	6,B
CBF1	SET	6,C
CBF2	SET	6,D
CBF3	SET	6,E
CBF4	SET	6,H
CBF5	SET	6,L
CBFE	SET	7,(HL)
DDCB05FE	SET	7,(IX+d)
FDCB05FE	SET	7,(IY+d)
CBFF	SET	7,A
CBF8	SET	7,B
CBF9	SET	7,C
CBFA	SET	7,D
CBFB	SET	7,E
CBFC	SET	7,H
CBFD	SET	7,L
CB26	SLA	(HL)
DDCB0526	SLA	(IX+d)
FDCB0526	SLA	(IY+d)
CB27	SLA	A
CB20	SLA	B
CB21	SLA	C
CB22	SLA	D
CB23	SLA	E
CB24	SLA	H
CB25	SLA	L
CB2E	SRA	(HL)
DDCB052E	SRA	(IX+d)
FDCB052E	SRA	(IY+d)
CB2F	SRA	A
CB28	SRA	B
CB29	SRA	C
CB2A	SRA	D
AA	XOR	D
AB	XOR	E
AC	XOR	H
AD	XOR	L
EE20	XOR	n

OBJ CODE	SOURCE STATEMENT	
9E	SBC	A,(HL)
DD9E05	SBC	A,(IX+d)
FD9E05	SBC	A,(IY+d)
9F	SBC	A,A
98	SBC	A,B
99	SBC	A,C
9A	SBC	A,D
9B	SBC	A,E
9C	SBC	A,H
9D	SBC	A,L
ED42	SBC	HL,BC
ED52	SBC	HL,DE
ED62	SBC	HL,HL
ED72	SBC	HL,SP
37	SCF	
CBC6	SET	0,(HL)
DDCB05C6	SET	0,(IX+d)
FDCB05C6	SET	0,(IY+d)
CBC7	SET	0,A
CBC0	SET	0,B
CBC1	SET	0,C
CBC2	SET	0,D
CBC3	SET	0,E
CBC4	SET	0,H
CBC5	SET	0,L
CBCE	SET	1,(HL)
DDCB05CE	SET	1,(IX+d)
FDCB05CE	SET	1,(IY+d)
CBCF	SET	1,A
CBC8	SET	1,B
CBC9	SET	1,C
CBCA	SET	1,D
CBCB	SET	1,E
CBCC	SET	1,H
CBCD	SET	1,L
CBD6	SET	2,(HL)
DDCB05D6	SET	2,(IX+d)
FDCB05D6	SET	2,(IY+d)
CBD7	SET	2,A
CBD0	SET	2,B
CBD1	SET	2,C
CBD2	SET	2,D
CBD3	SET	2,E
CBD4	SET	2,H
CBD5	SET	2,L
CBD8	SET	3,B
CBDE	SET	3,(HL)
DDCB05DE	SET	3,(IX+d)
FDCB05DE	SET	3,(IY+d)
CBDF	SET	3,A
CBD9	SET	3,C
CBDA	SET	3,D
CBDB	SET	3,E
CBDC	SET	3,H
CBDD	SET	3,L
CBE6	SET	4,(HL)
DDAE05	XOR	(IX+d)
FDAE05	XOR	(IY+d)
AF	XOR	A
A8	XOR	B
A9	XOR	C

APPENDIX F

8255A/8255A-5
PROGRAMMABLE PERIPHERAL INTERFACE

- MCS-85™ Compatible 8255A-5
- 24 Programmable I/O Pins
- Completely TTL Compatible
- Fully Compatible with Intel® Microprocessor Families
- Improved Timing Characteristics
- Direct Bit Set/Reset Capability Easing Control Application Interface
- 40-Pin Dual In-Line Package
- Reduces System Package Count
- Improved DC Driving Capability

The Intel® 8255A is a general purpose programmable I/O device designed for use with Intel® microprocessors. It has 24 I/O pins which may be individually programmed in 2 groups of 12 and used in 3 major modes of operation. In the first mode (MODE 0), each group of 12 I/O pins may be programmed in sets of 4 to be input or output. In MODE 1, the second mode, each group may be programmed to have 8 lines of input or output. Of the remaining 4 pins, 3 are used for handshaking and interrupt control signals. The third mode of operation (MODE 2) is a bidirectional bus mode which uses 8 lines for a bidirectional bus, and 5 lines, borrowing one from the other group, for handshaking.

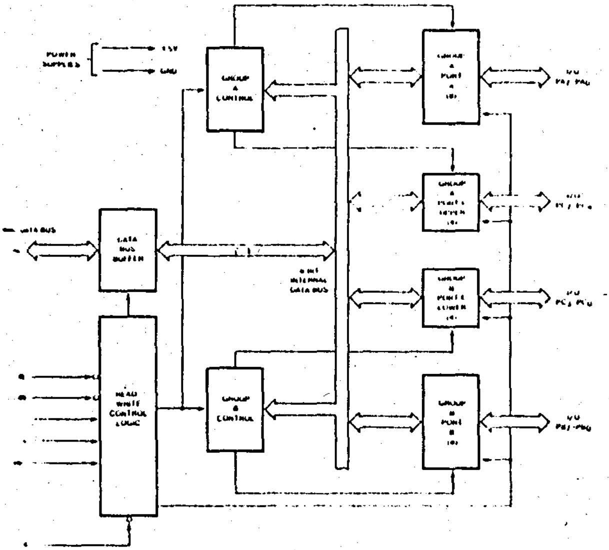


Figure 1. 8255A Block Diagram

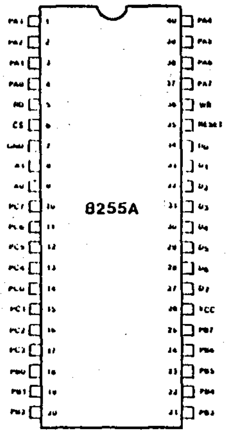


Figure 2. Pin Configuration

8255A FUNCTIONAL DESCRIPTION

General

The 8255A is a programmable peripheral interface (PPI) device designed for use in Intel® microcomputer systems. Its function is that of a general purpose I/O component to interface peripheral equipment to the microcomputer system bus. The functional configuration of the 8255A is programmed by the system software so that normally no external logic is necessary to interface peripheral devices or structures.

Data Bus Buffer

This 3-state bidirectional 8-bit buffer is used to interface the 8255A to the system data bus. Data is transmitted or received by the buffer upon execution of input or output instructions by the CPU. Control words and status information are also transferred through the data bus buffer.

Read/Write and Control Logic

The function of this block is to manage all of the internal and external transfers of both Data and Control or Status words. It accepts inputs from the CPU Address and Control busses and in turn, issues commands to both of the Control Groups.

($\overline{CS}$)

Chip Select. A "low" on this input pin enables the communiction between the 8255A and the CPU.

($\overline{RD}$)

Read. A "low" on this input pin enables the send the data or status information to the CP data bus. In essence, it allows the CPU to "re the 8255A.

($\overline{WR}$)

Write. A "low" on this input pin enables the CPU data or control words into the 8255A.

(A_0 and A_1)

Port Select 0 and Port Select 1. These input sig conjunction with the RD and WR inputs, con selection of one of the three ports or the contr registers. They are normally connected to the significant bits of the address bus (A_0 and A_1).

8255A BASIC OPERATION

A_1	A_0	$\overline{RD}$	$\overline{WR}$	$\overline{CS}$	INPUT OPERATION
0	0	0	1	0	PORT A → DATA BUS
0	1	0	1	0	PORT B → DATA BU
1	0	0	1	0	PORT C → DATA BU
					OUTPUT OPERATI (WRITE)
0	0	1	0	0	DATA BUS → PORT A
0	1	1	0	0	DATA BUS → PORT B
1	0	1	0	0	DATA BUS → PORT C
1	1	1	0	0	DATA BUS → CONTR
					DISABLE FUNCTION
X	X	X	X	1	DATA BUS → 3-STAT
1	1	0	1	0	ILLEGAL CONDITIO
X	X	1	1	0	DATA BUS → 3-STAT

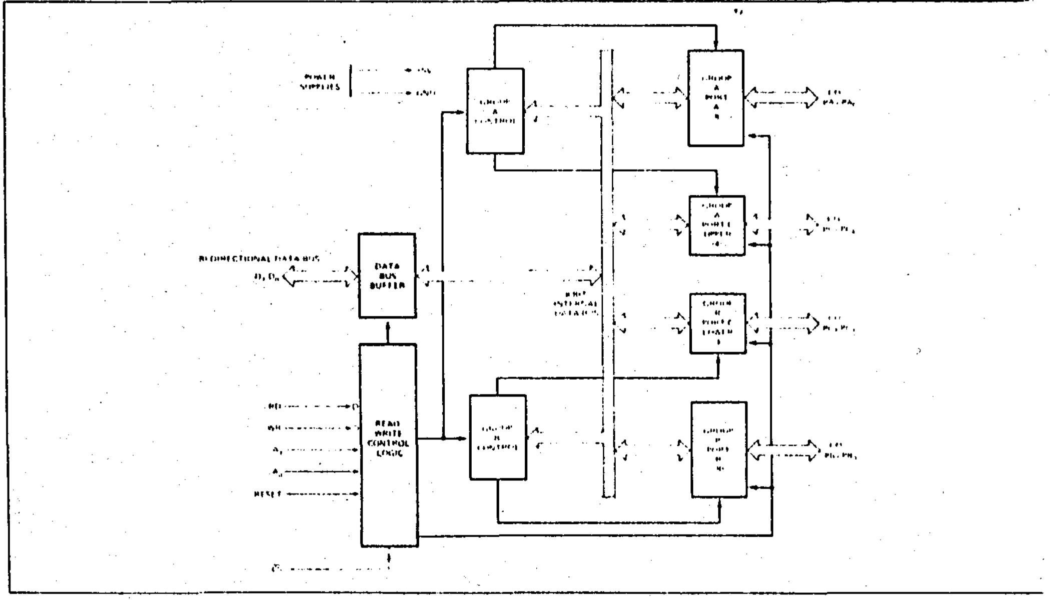


Figure 3. 8255A Block Diagram Showing Data Bus Buffer and Read/Write Control Logic Functions

8255A OPERATIONAL DESCRIPTION

Mode Selection

There are three basic modes of operation that can be selected by the system software:

Mode 0 – Basic Input/Output
Mode 1 – Strobed Input/Output
Mode 2 – Bi-Directional Bus

When the reset input goes "high" all ports will be set to the input mode (i.e., all 24 lines will be in the high impedance state). After the reset is removed the 8255A can remain in the input mode with no additional initialization required. During the execution of the system program any of the other modes may be selected using a single output instruction. This allows a single 8255A to service a variety of peripheral devices with a simple software maintenance routine.

The modes for Port A and Port B can be separately defined, while Port C is divided into two portions as required by the Port A and Port B definitions. All of the output registers, including the status flip-flops, will be reset whenever the mode is changed. Modes may be combined so that their functional definition can be "tailored" to almost any I/O structure. For instance; Group B can be programmed in Mode 0 to monitor simple switch closings or display computational results, Group A could be programmed in Mode 1 to monitor a keyboard or tape reader on an interrupt-driven basis.

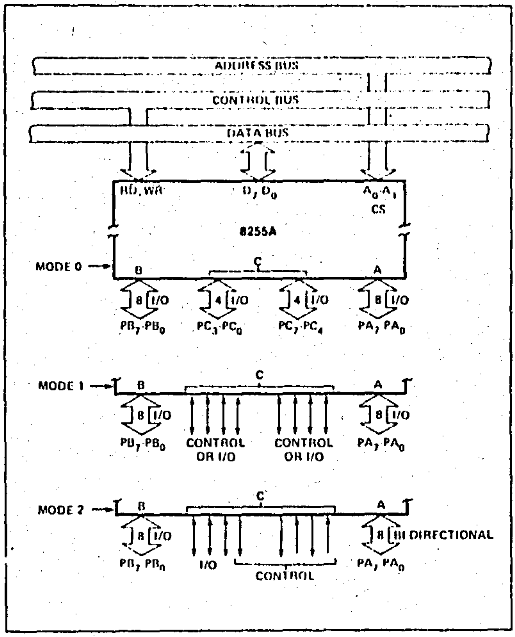


Figure 5. Basic Mode Definitions and Bus Interface

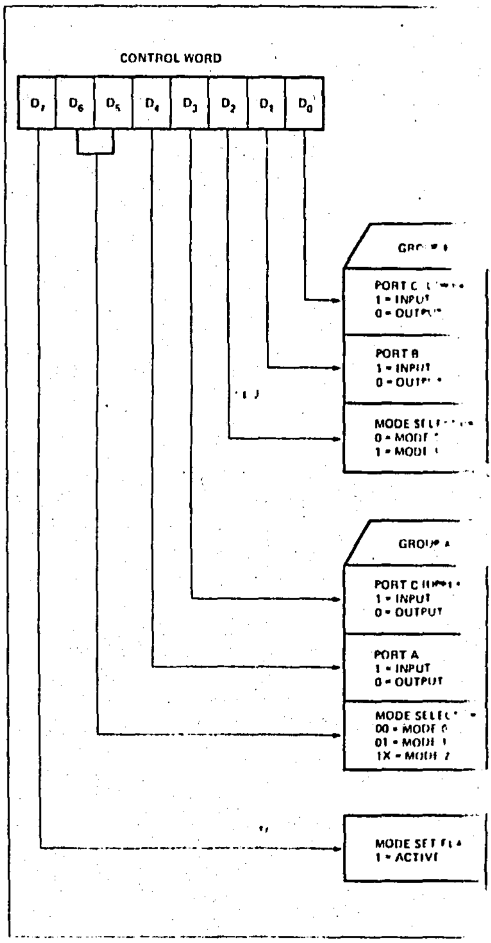


Figure 6. Mode Definition Format

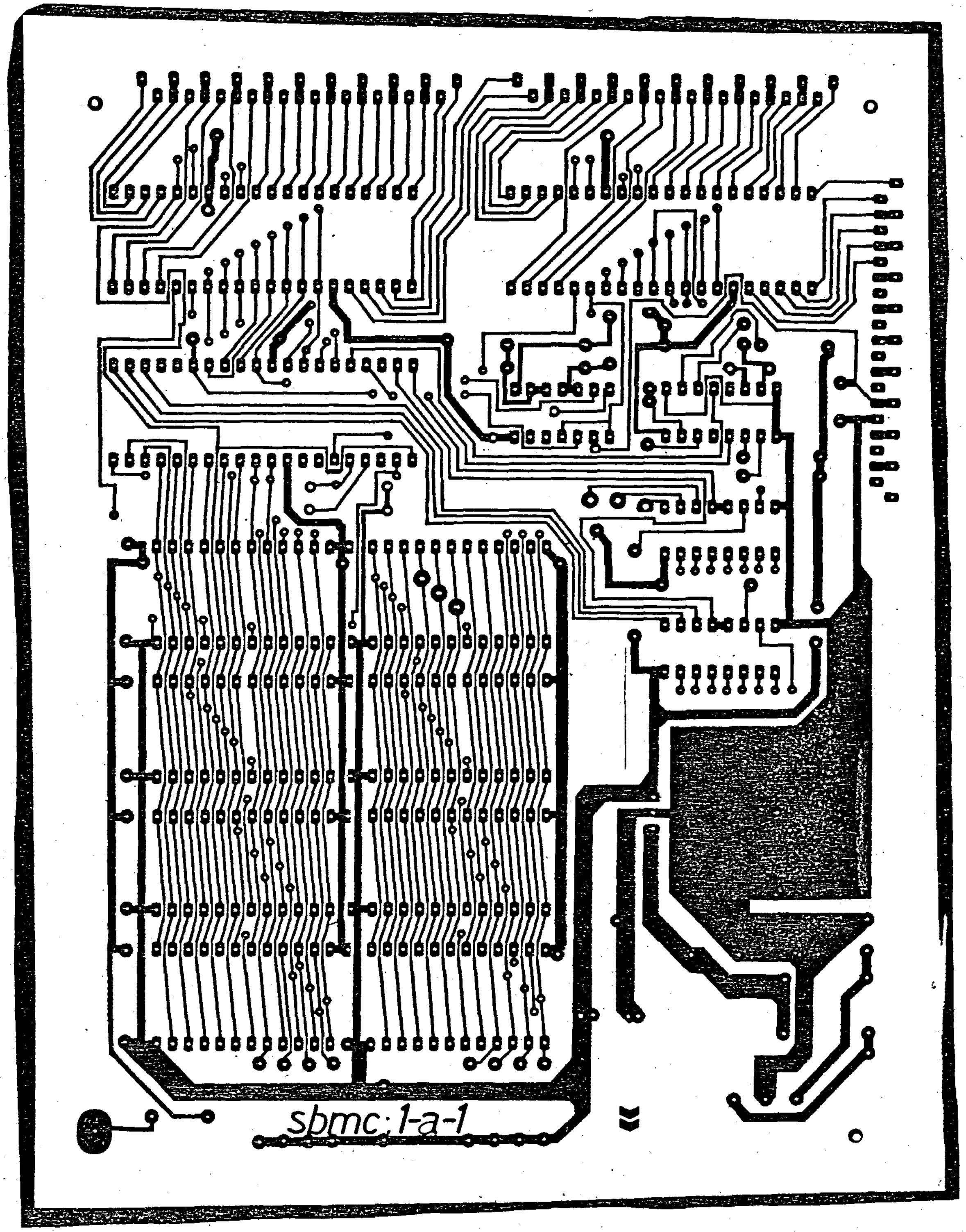
The mode definitions and possible mode combinat may seem confusing at first but after a cursory revie the complete device operation a simple, logical I/O proach will surface. The design of the 8255A has ta into account things such as efficient PC board lay control signal definition vs PC layout and comp functional flexibility to support almost any periph device with no external logic. Such design repres the maximum use of the available pins.

Single Bit Set/Reset Feature

Any of the eight bits of Port C can be Set or Reset usi single OUTput instruction. This feature reduces softw requirements in Control-based applications.

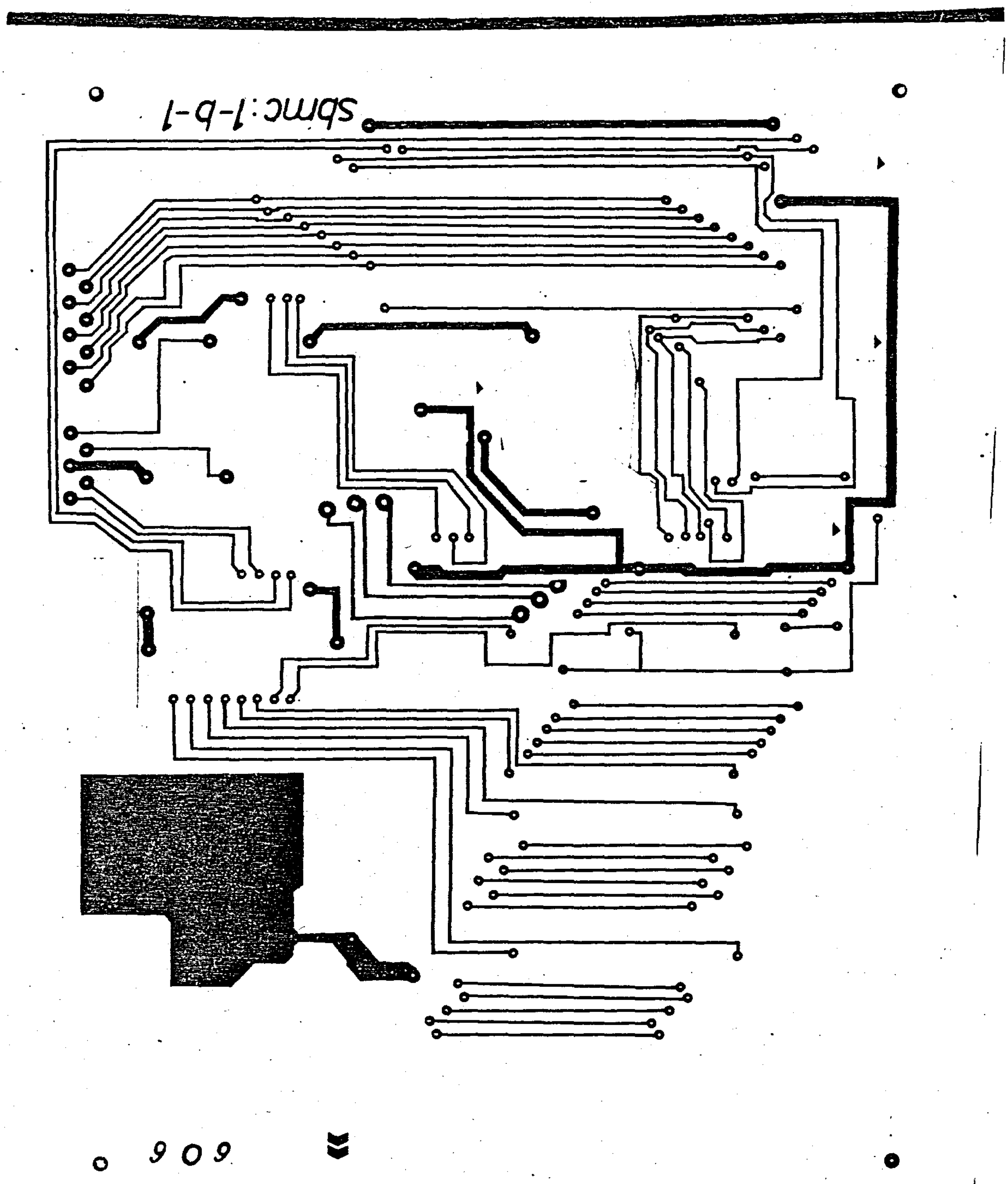
APPENDIX G

PCB LAYOUTS



SOLDER SIDE

COMPONENT SIDE



BIBLIOGRAPHY

1. Acarnley P. P. , Stepping Motors,Peter Peregrinus Ltd. , 1982

2. Miyamoto H. , Goeldel C. , A Microproccessor - Based Time Optimal Control of A Variable-Reluctance Step Motor , IEEE Transactions on Industrial Elect.,Vol.29,No.3,pp 190-196 August 1982

3. Acarnley P.P., Gibbons P., Closed Loop Control of Stepping Motors : Prediction and Realization of Optimum Switching Angle , IEEE Proc. Vol.129,No.4,pp 211-216 , july 1982

4. Barden W. , The Z80 Microcomputer Handbook , Tabs ,1981

5. Zaks R. , Microproccessor Interfacing Techniques , Sybexs , 1982

6. Bailey D. A. , Stepper Motor Drive Circuit , Wireless World , pp 76-81 , February 1983

7. Vraets W. J. , French D. , Minor Closed Loop Control of Stepping Motors For Use In High Speed Positioning , Int. J. Mach. Tool Des. Res. , Vol.21 , No.314, pp. 217-223 , 1981