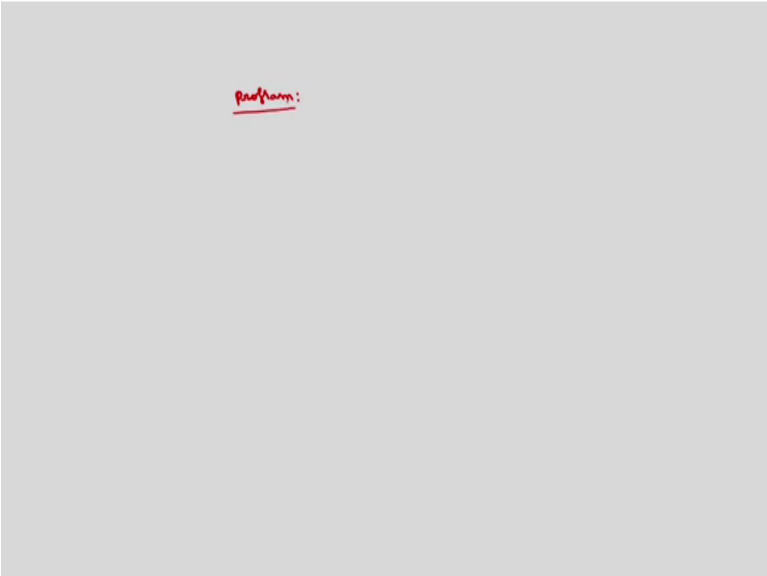
Microprocessors and Interfacing

Prof. Shaik Rafi Ahmed
Department of Electronics and Electrical Engineering
Indian Institute of Technology, Guwahati

Lecture – 25
Electronic weighing machine

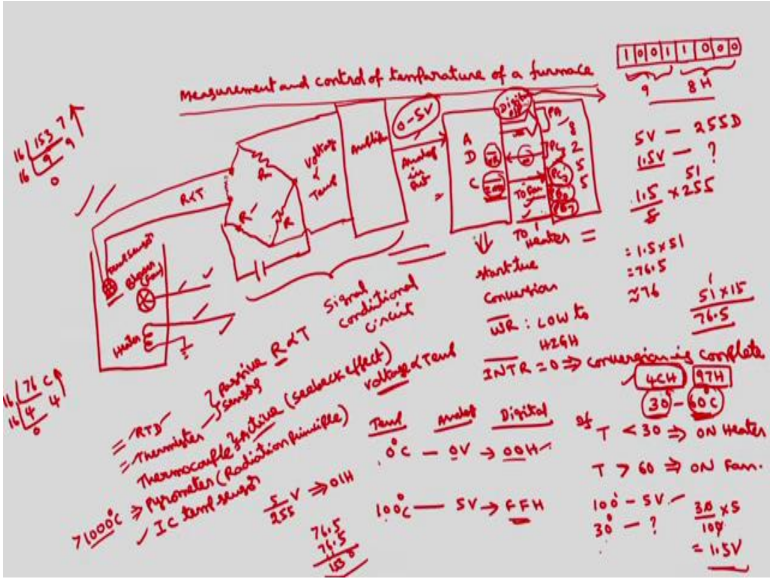
In the last class we were discussing about the temperature control of furnace. So, we have discussed about the connection details.

(Refer Slide Time: 00:40)



Now, if I come for the program.

(Refer Slide Time: 00:50)



So, we are using this port A as output port; port C lower as output port this is input port, this is output port, this is output port. So, what is the control word register for this so if you want the control word register in I/O mode. This is for port C lower. So, port C lower; this is port C lower we are using as a output port.

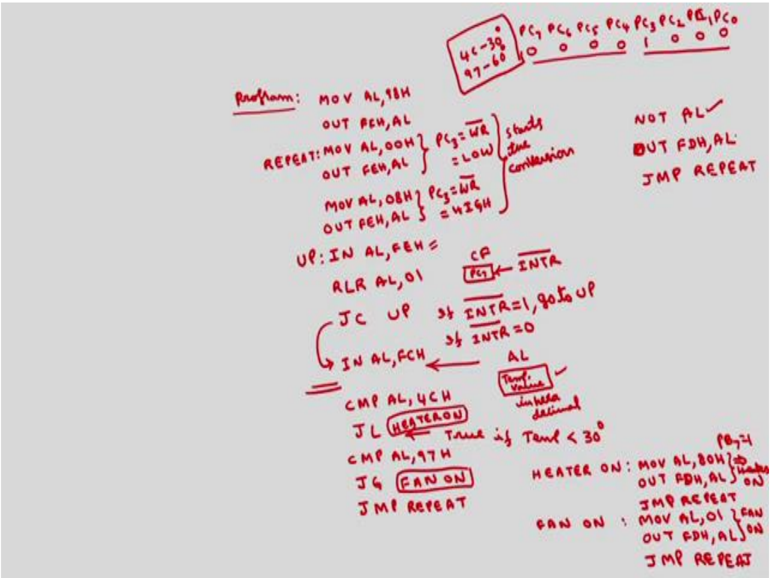
So, this will be 0. This is port B; port B also you are using these two pins. Output port 0 this is mode selection for port B I am using mode 0. This is port C upper; port C upper is this one. We are going to take this as input port. So, one this is port A. Port A we are using as input port 1 mode selection for port A.

I am taking 0; mode 0 and this is for I/O mode this is 1. So, what will be control word 9, 8; 98 H. So, you have to first load 98 H into control word register, after that the process is this right bar you have to make as a low to high transition. You have to wait until INTR bar is equal to 0. Once the INTR bar is 0 then you have to read the digital equivalent of this temperature. Then I have to compare with 30 degrees and 60 degrees if it is less than 30 degrees.

We have to ON the heater and if it is greater than 60 degrees you have to ON the fan. So, for that you have to write the program. So, 98 H AL comma, 98 H OUT control word register this is FFH comma, AL. Then, MOV AL comma, 0 0. Then OUT so, this right bar is connected to port C lower right?

This is port C lower right bar; right bar you have to make as 0 to 1 transition.

(Refer Slide Time: 03:37)



So, port C address will be FEH AL. This will make PC3 which is equal to WR bar Low. Then again I have to make high. MOV AL comma, PC3 you have to make high. Means PC3 bit will be PC0, PC1, PC2, PC3 the MSB of the lower nibbile.

So, that means it will be 08 H. Is it clear? This is PC7, PC6, PC5, PC4, PC3, PC2, PC1, PC naught. This should be 1 remaining all can be 0's. This is 0 hexadecimal this is 8; 08 H then OUT onto the same port C FE. So, this will make PC3 is equal to right bar these two are connected is high. So, this low to high transition will start the conversion. Then you have to check for the end of conversion, because the microprocessor is a faster device A to D converter will take some time to convert.

So, once the conversion is over this INTR bar becomes 0, INTR bar is connected to PC7; this INTR bar is connected to PC7. So, you have to input port C, IN port C AL comma, port C address is FEH. Then PC7 is MSB bit. So, MSB bit will be you have to check whether this is 0 or 1. Rotate Left Accumulator RLR AL comma, 01. So, that the MSB bit will come on to the carry flag, this is PC7 which is the status of INTR bar.

If this is 0, jump on carry. If this is one means carry; that means, conversion is not yet over. So, you have to wait here itself; you have to input this again jump on carry UP. So, you have to continuously monitor this PC7 which is equal to INTR bar if this is equal to

1 if INTR is equal to 1 it will go to UP; is equal to 1 go to UP, means conversion is not yet over you have to wait in a loop. If INTR bar is 0.

So, this will come out of the loop then you have to read the data. So, where this digital equivalent of the temperature is available? In port A. So, we have to write IN AL comma, port A address is FCH.

So, after this instruction AL will contain; the digital equivalent of temperature value; temperature value in hexadecimal form. Now, you have to compare this with 30 degrees the hexadecimal equivalent of 30 degrees hexadecimal equivalent of 60 degrees. 30 degrees yesterday we have calculated this is 4C H and 60 degrees 97 H.

4C and 97. So, 4C is corresponding to 30 degrees, 97 H is corresponding to 60 degrees. Compare AL which contains the present temperature value then 4 CH; the lower limit, jump on less than.

If AL is less than 4 CH means no need to ON the heater. If it is less than 30 degrees you have to ON the heater. Less than we have to if jump less. So, this is true if temperature is less than 30 degrees, then you have to ON the heater. Jump less than HEATER ON. So, HEATER ON you have to write the program.

Otherwise you have to compare with; compare AL with 97 H which is corresponding to 60 degrees. Jump greater than FAN ON. Otherwise if it is greater than 30 and if it is less than 60. So, you do not have to I mean ON and OFF fan and heater right.

So, you have to unconditional jump to monitoring this UP. May REPEAT you can call as you have to REPEAT the process. So, where should be this REPEAT again starting from this REPEAT. So, you have to continuously again monitor this you have to I mean start for the conversion then you have to check the end of conversion the same process will repeats.

So, at this HEATER ON you have to write some program FAN ON you have to write some program. HEATER ON, HEATER is connected to this is the pin PB7 is connected to a HEATER. PB7 you have to make as 1. To make PB7 as 1; MOV AL comma, 80 H. Now, you have to output on to port B FDH, OUT FD comma, FDH comma, AL.

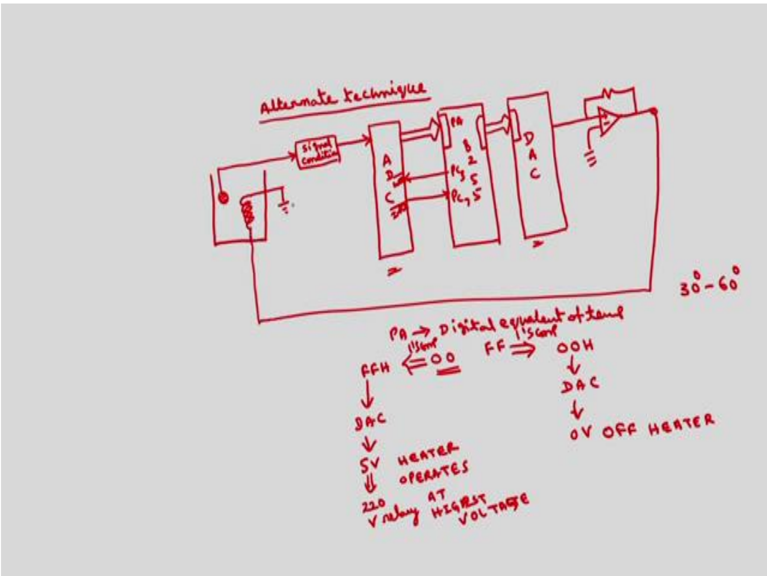
So, this will make PB7 as one which is connected to the HEATER. PB7 is equal to 1 in place heater is ON. Then you have to unconditional jump and now you have to repeat the process; JMP to same REPEAT process. And FAN ON MOV AL comma, FAN is connected to PB0. PB0 means you have to make a 01 AL comma 01. So, that PB0 becomes 1 OUT FDH comma AL.

So, these two instructions will ON FAN then unconditional jump to again repeat. This is the program which controls the temperature and maintain the temperature between 30 degrees and 60 degrees for the connections given here.

So, these are the connections. So, you have connected this I mean digital equivalent of temperature to port A, start of conversion to PC3, end of conversion to PC7 and the HEATER to PB7 and fan to PB0. If want to display the temperature value.

So, whatever these digital value that you have to read we can apply through BCD into 7 segment decoder or directly to the 7 segment decoder and you can display the values. So, we can do this in a similar another way also instead of using this both heater and blower or fan, we can use only single heater only or you can use the fan only. If I use D to A converter. So, the setup will be this entire setup will be same I will just write from the connections of ADC onwards.

(Refer Slide Time: 13:37)



So, alternate way is; alternate technique, this is the furnace. So, I will use only either fan or heater this is the sensor; temperature sensor and I will use this signal conditioning I will call as signal conditioning.

I will give to A to D converter means this digital equivalent value are given to some of the ports of 8255 this is same as the previous thing. The start of conversion, end of conversion is similar this is right bar; this is INTR bar this is port A, this is PC3, this is PC7.

Now, what I will do is, now I will connect this another port say port B this I will connect to D to A converter. This output through resistance it will convert into the voltage; this I am going to give to the fan. So, the remaining connections of this A to D and D to A converters I am not showing, because those details I have already discussed in the earlier lectures.

Now, the basic principle here is, everything is same except for this that. So, we will read here in port A the digital equivalent of the temperature. So, here we do not have the limits now. So, you do not want to control between 30 degrees and 60 degrees. I want to maintain at a particular temperature only. So, based on this I want to maintain a fixed temperature. So, whatever the value that we are going to get here, if this is FF means temperature is large.

So, I want to reduce the temperature for that what I will do is I will complement this I will take the ones complement of this one. So, that it becomes 00 H this I will give to the D to A converter. So, you will get 0 volts. So, thereby these 0 volts will OFF the FAN. On the other hand, if this is 00 H your digital equivalent is 00 H means temperature is almost 0. So, I will again take the ones complement of this one.

So, that it becomes FFH, this I will convert into DAC, you will get the maximum voltage plus 5 volts and this will run the fan at highest speed. Of course, this can be 5 volts to 220 volts we are going to have relay 5 volts to 220 volts it will operate 220 volt relay. FAN is ON, FAN operates at maximum speed.

Because temperature is less there is 00 another way you have you can have the reverse. This will be FAN has to be operated it. So, no need of this I mean a compliment also; otherwise if this is HEATER you can call as instead of FAN you can use a HEATER. So, the heater operates at the maximum temperature. Highest voltage thereby it will produce more heat. So, instead of this I am using the HEATER this is the process. So, you can write in a similar manner the program, the program is previous program only the only modification is here after reading this temperature value in AL ok.

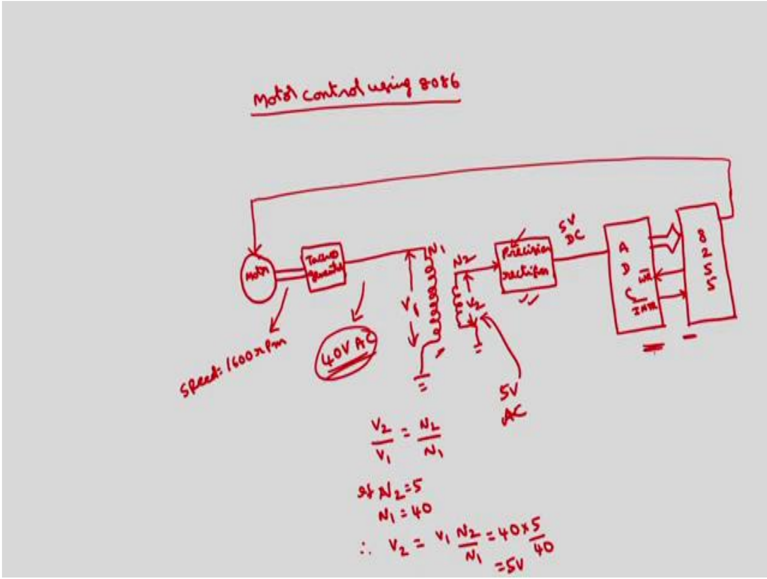
Simply you have to complement this value and you have to give that value to the digital to analog converter. So, up to here this is same I am writing the remaining program.

So, in AL you have to complement this value NOT AL. If temperature is highest then the heater will be operated at low voltages. If temperature is low, then you have to; the operate the heater at highest voltages that is why we are taking NOT AL. So, this AL contents will be complemented this I have to send to on to D to A converter.

So, this any how AL is there. So, MOV AL is there you have to OUT on to D to A converter I am going to connect to, this is D to A converter I have connected to port B. This you have to program as output port you know in the previous example also you have program the port B as output port. So, out port B is FDH comma, AL. Then you have to do this repeat process you jump on to repeat.

So, just we are just complimenting this AL value that temperature value we are giving to the port B which is connected to the heater. So, thereby heater temperature will be controlled. So, which controls the temperature of the furnace, this is the another way if I want to use both A to D converter and D to A converter. So, in a similar manner you can have lot of applications. Like if you want to use DC motor control.

(Refer Slide Time: 21:44)



A motor control using 8086. So, you can use a motor this is connected to tachogenerator. So, this produces a voltage which is proportional to the speed of this.

If this is 1600 rpm speed; here it will generate 40 volts AC voltage at the tachogenerator. So, this 40 volts is a high voltage which cannot be directly connected to microprocessor. So, what you have to do is you have to step down this voltage using step down transformer. Then you have to connect to a precision rectifier; to convert AC into DC because this is AC. Here the voltage will be depends upon the this number of turns. If I call this primary voltage here this voltage is 40 volts in this case.

In general if I call this as V1, and if I call this as V2. This number of turns here are N1, number of turns here are N2. Then V2 by V1 is equal to N2 by N1. If I take V1 as 40 volts in this case if it is this is 40 volts. And if I take this turns ratio as N2 is equal to; if I take N2 number of turns here are 5, N1 the number of turns here; if I take 40. Then what will be V2? V2 is equal to V1 into N2 by N1. Which is equal to V1 is 40 volts; this V1 is 40 volts here; 40 into N2 is 5. N1 is again 40 I have taken, so that we will get 5 volts. So, here we will get 5 volt AC. So, to convert into DC we will use precision rectifier.

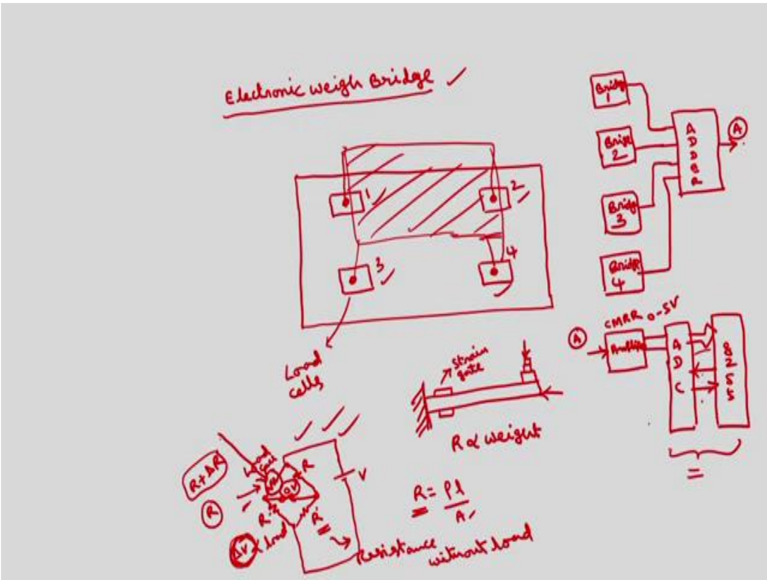
So, the precision rectifier will be same as this ordinary rectifier except for that here we will be using op-amps. So, we can construct this rectifier by using just simply diodes also. So, if it is a half wave rectifier 2 diodes fully rectifier 4 diodes.

So, the name precision is if I use this op-amp operational amplifier then it is called as precision rectifier. Without operational amplifier also you can construct this rectifier using just simple diodes, that is called just rectifier. If we use op-amps also instead of precision rectifier then this 5 volts will be given to A to D converter. So, it will produce the corresponding digital value this will be given to 8255.

So, here again right bar interrupt bar whenever A to D converter is connected this is 0800 then these two signals you have to connect. So, this 8255 another port of this one, we can use to control the speed of this motor using some feedback mechanism. We set some value here set point.

So, I want to operate at some speed if it is more than that one we can activate. So, we can activate here some sort of break or something like. So, we can control the speed of the motor. So, similarly I will discuss one more example before going for the other interfacing devices is electronic weighing bridge.

(Refer Slide Time: 26:40)



So, these are all actually is a minor project type of thing. So, if you want you can do this projects. So, the hardware details I have discussed here. So, only thing that you need is some sort of this signal conditioning like precision rectifier. So, you can refer any linear integrated circuits book, you can find the circuit of precision rectifier. If any amplifier is required amplifiers also you can find in any linear integrated circuits books.

So, electronic weighing machine so here basically I want to weight; here basically I want to weigh a vehicle there will be 4 load cells. These are 4 load cells. So, vehicle will sit on these 4 wheels will be placed on these 4 load cells. So, below this there will be some sort of this type of mechanism. So, there is a cantilever beam, if a load is placed on this one if I place a load here what happens this side is fixed. So, if I connect here strain gauge.

So, this load will change the dimensions of this cantilever beam thereby the resistance will changes. Basically, here resistance is proportional to the weight, that change in resistance can be measured by using bridge. So, here we have 4 different load cells will give 4 different changes. So, the basic circuitry here will be you have to connect these 4 in 4 arms of a bridge. This is load cell corresponding to 1; this is 1 2 3 4.

So, without any load this will have some resistance depends upon R is equal to rho l by A. Rho is the resistivity, l is the length of this rod A is the area. If I apply some load here. So, this resistance will changes; I will set the other 3 arms R R R this resistance is equivalent to resistance without load; this R is equal to resistance without load.

So, this also will be having initially without load cell application this is also R. If I apply the load here, what happens? This becomes R become R plus delta R. So, when this is also R this bridge will be balanced. If I connect these 2 points to some voltage. At these 2 points you will get some voltage which is equal to 0 volts in case all are 4 R's. This is R, this is R, this is R, this is R because this voltage will be V by 2, this voltage will be V by 2. So, the difference becomes 0 volts.

If I apply the load this becomes R plus delta R which causes the unbalance of the bridge. Thereby, the voltage here will be other than 0. So, this delta V volts that is generated here is proportional to the load. Whereas, in case of this electronic weighing machine we will be having 4 such loads.

So, these 4 voltages will be having 4 such bridges. So, we will be having 4 such bridges this is 1 bridge, this is 1 bridge. So, inside this bridge we will be having this circuitry. So, 4 such bridges; this is Bridge 1 corresponding to this load cell. Bridge 2 corresponding to this load cell, Bridge 3 corresponding to this, Bridge 4 corresponding to this.

So, these 4 voltages you have to connect to the adder. You can implement the adder circuit also using operational amplifier. You can find the circuit also in any one of the standard linear integrated circuit books.

You have to basically add these. Still this voltage will be very small this delta V will be very small. So, that small voltage we cannot directly apply to this microprocessor; this voltage of this adder can be applied to the amplifier. So, I am connecting here, this is A point; this is A point there is a link between these 2.

So, you have to connect to a amplifier. Normally, here we will use instrumentation amplifier, which will be having high gain high CMRR Common Mode Rejection Ratio, because while measuring these voltages there may be some interference to the common mode signals such as noise which can be eliminated if your amplifier is having high CMRR Common Mode Rejection Ratio, high gain, high input impedance so that the loading effect will be less.

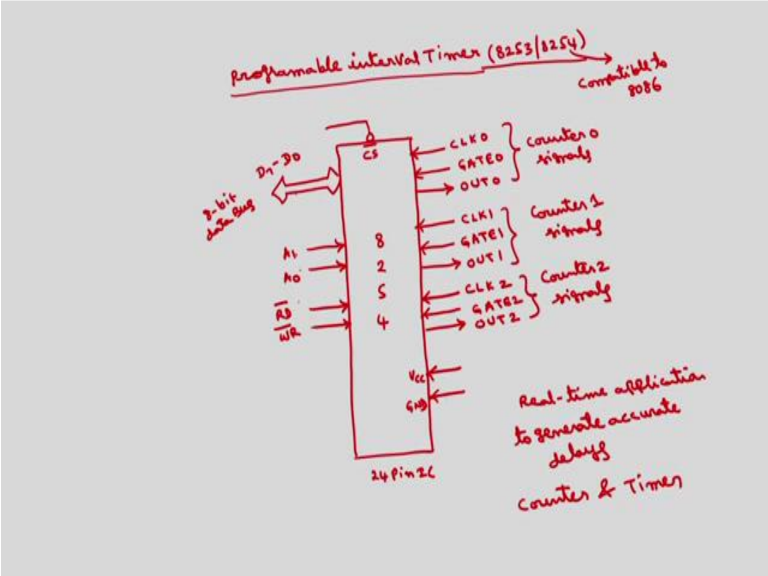
Low output resistance. So, these are the some characteristics of the amplifier that is to be used in the signal conditioning circuitry. Choose this amplifier again such that so, the output will be in the range of 0 to 5 volts which you can directly apply to the A to D converter, then to the microprocessor. Then this will be connected to ADC, which is connected to 8255 which in turn connected to 8086. This is start of conversion. this is end of conversion. This is the complete block diagram of electronic weighing machine.

So, we can write the program the main contribution here is interfacing the A to D converter and how to start the conversion, how to end the conversion, how to read the digital equivalent of this load. So, the program that we have used for the temperature measurement is almost same here also.

So, in a similar manner you can write the program of electronic weighing machine and if want you can display the weight of that particular vehicle here we are going to lay the vehicle. So, the weight of the vehicle can measure. You can display on the display board this is the vehicle.

So, like that we have plenty of applications of 8086 using this 8255 ADC and DAC. Now, we will discuss some more the peripheral devices till now we have discussed about only one peripheral device which is 8255. There are some other peripheral devices which will be having some other applications. So, the next peripheral device is 8254 which is programmable interval timer.

(Refer Slide Time: 35:27)

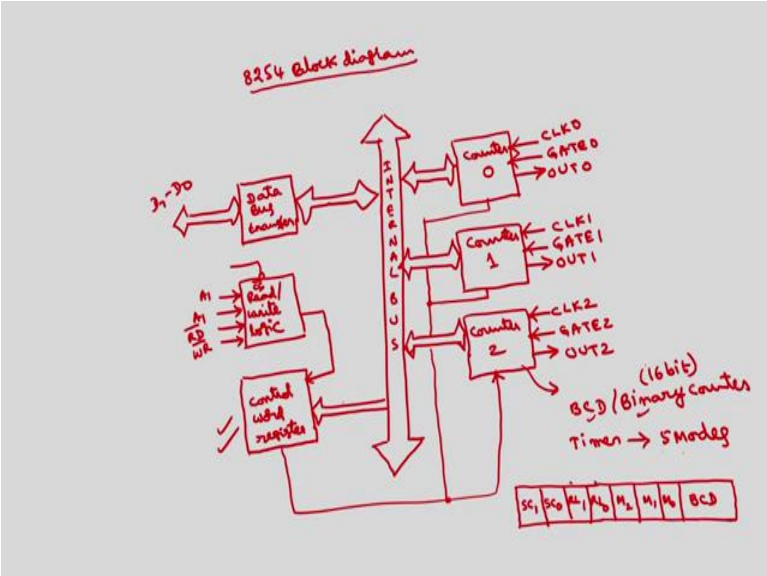


This is 8253 or 8254. The pin configuration is same, but 8254 is compatible to 8086. 8253 not compatible. So, this is basically a 24 pin IC which will be having 3 timers and 3 16 bit counters. So, there are 3 control signals for each counter one is CLOCK 0, GATE 0, OUT 0 this is counter 0 signals.

Similarly, for counter one we have CLOCK one, GATE one, OUT one. These are all counter 1 signals. Counter 2 similarly, CLOCK 2, GATE 2, OUT 2. So, any IC will be having BCC and ground and there will be 1 chip select signal ok. And there is 8 bit data bus D7 to D0, then we have A1 and A0 two signals which will select the counters and we have read bar write bar. So, total 24 signals this is 8 9 10 11 12 13 14 and these are total 9 plus 1 10 is 24 pin this is 24 pin IC.

Basically this programmable interval timer will be used in real time applications to generate accurate delays, using the delay programs we cannot generate the accurate delays. But there are some applications where you have to generate the delays precisely very accurately presented the accurate delays. We have to use 8254. In addition this also will acts as counters. So, this 8254 will acts as counter as well as timer; we have both the applications.

(Refer Slide Time: 39:36)



So, if you take inside this pin diagram 8254 architecture diagram or block diagram. So, you have database buffer this is internal bus, this is internal bus. There are 3 counters as I have told this is 16 bit counters, counter 0, counter 1, counter 2. There are 3 signals; 2 input, 1 output OUT 2, GATE 2, CLOCK 2, OUT 1, GATE 1 what are the functions of these signals we will discuss we will discussing about the different modes.

So, I will discuss the functions of each of these signals OUT 0 GATE 0 CLOCK 0. There will be a data bus transfer. So, you have 8 bit data bus as we have told D7 to D0. Then we have a read write logic. For this block we have chip select signal A1 A0 read bar write bar. And there is one control word register which is similar to 8255 to program this counters timers we need a control word register. So, this control word register will control all the counters there will be connection to all the counters.

This is the complete block diagram of 8254 timer. So, here also we have 9 signals here this is 8, 17 signals. 18, 19, 20, 21 and we have VCC ground total 24 signals. Now, here we can program this there are 16, 3 16 bit counters. We can count this using BCD count or binary counting; and you can program this as a timers also.

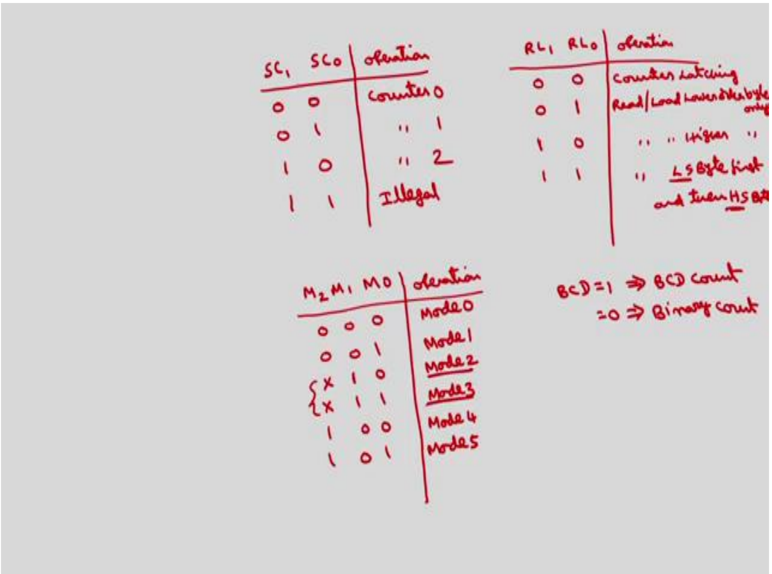
So, the timers can operate in 5 modes. I will explain each mode in detail. So, in order to operate this in 5 modes and whether this is a binary counter or BCD counter there we are going to these are all 16 bit counters; in 16 bit, but you can also use as a 8 bit counters also. In case of 8 bit counters so, you have to a load only 8 bits only in case of 16 bit counter first you have to load the 8 bit lower order 8 bits then higher order 8 bits.

So, these are all depends upon the format of the control word register. So, the format of the control word register is similar to the 8255. We have 8 bit control word register; here also we have 8 bit control word register. Which is this will be SC1 SC2 SC1 SC0 this will control select counter SC stands for Select Counter 00 means counter 0, 01 means counter 1, 10 means counter 2, 11 means it is illegal operation.

Similarly, we have RL0 RL1. RL1, RL0 read or load either byte or word. So, byte means again it is a lower order 8 bits or higher order 8 bits. So, depends upon the status of these 2 pins we can select either lower order byte or higher order byte or both the bytes. I will explain the function of that. Then we have 3 modes in order to select the 3 modes; we require 3 control signals M2 M1 M0 there are 5 modes. So, in order to select these 5 modes we require 3 control signals M2 M1 M0.

Then there this will be last bit which is BCD total 8 bits. So, if this is for BCD count this will be 1 for binary count this will be 0.

(Refer Slide Time: 46:20)



So, the exact functions of this one if I come here the MSB 2 bits SC1 SC0 as the name implies select counter. Operation 00 means counter 0 will be selected, 01 means counter

1 will be selected, 10 means counter 2 will be selected, 11 is illegal these are about the first 2 bits next we have RL1 RL0.

So, the operation is we have 4 combinations 00 means counter latching operation. I will discuss this what is this counter latching operation while discussing about the modes, 01 means it will read or load lower order byte only means for 8 bit operation lower order 8 bit only. If this is 10 read or load higher order byte only if this is 11 read or load lower order byte first then higher order byte; high significant byte.

Low significant byte high significant byte it will read both means this is first 16 bit operations and M2 M1 M0 is mode selection hence I have told there are 5 modes. And this is 0 0 0 mode 0, 0 0 1 mode 1, x 1 0 is mode 2 means this can be 0 1 0 or 1 1 0 both are mode 2, this is x 1 1 this can be 0 1 1 or 1 1 1 is mode 3, then we have 1 0 0 mode 4, 1 0 1 mode 5 here these 2 are do not cares. So, we have two possibilities for mode 2 and mode 3.

And then the last bit is BCD; if this bit is 1 BCD count. This is 0 binary count this is about the control word register. Now, how to interface this 8254 to the 8086 after that how to operate in 5 modes and how to generate the precise delays? And how to generate this square wave and different waveforms that we will discuss in the next lecture.

Thank you.