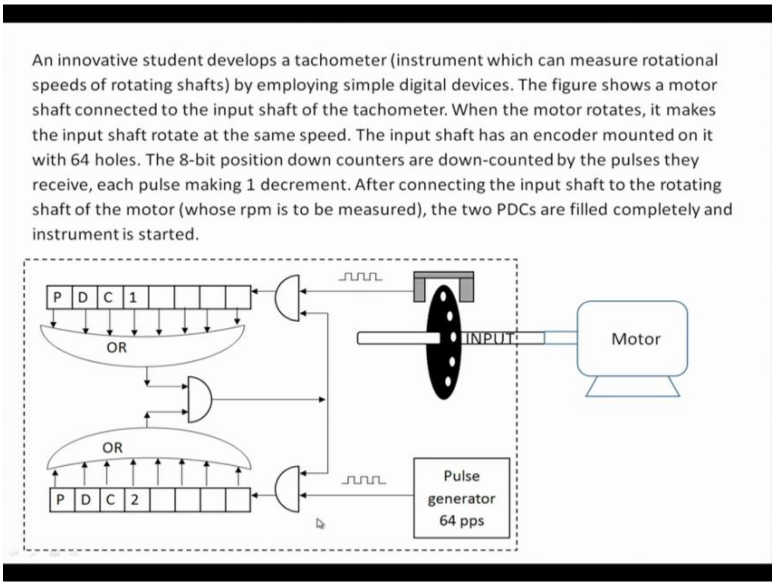
Computer Numerical Control of Machine Tools and Processes
Professor A Roy Choudhury
Department of Mechanical Engineering
Indian Institute of Technology Kharagpur
Lecture 07
Binary Circuits and Decoder

Welcome viewers to the 7th lecture for the open online course "Computer numerical control of machine tools and processes". Today, we are going to have a look at different kinds of binary circuits, decoders, et cetera, DDAs that we are employing in case of CNC and how they can be utilized to control not only CNC machine tools, but also different kinds of processes. They might be quite varied and having different fields of applications, so let's have a quick look what kind of applications that we can have of these binary circuits.

(Refer Slide Time: 01:12)



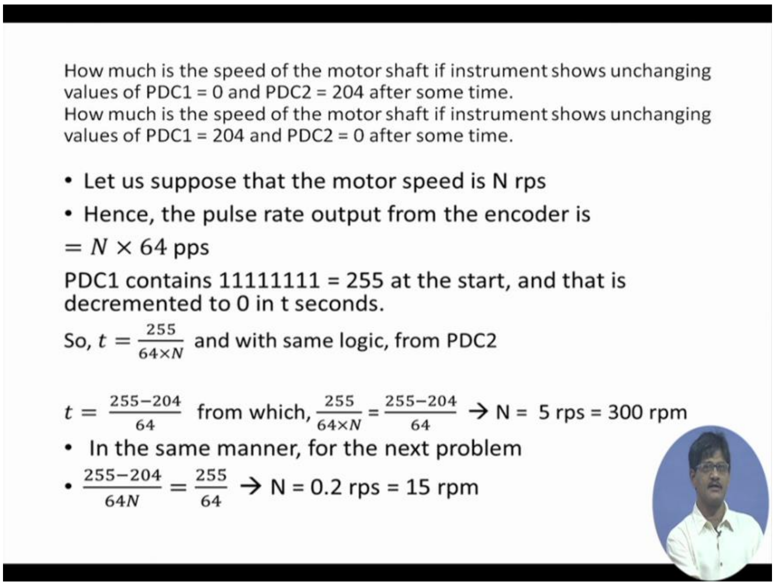
So let's start right away with a problem. The question reads, an innovative student develops a techno-meter, which is basically an instrument which can measure rotational speed of rotating shafts by employing a simple by employing simple digital devices. The figure shows a motor shaft connected to the input shaft of a techno-meter, so this dotted line shows the basically the device that the student has built up. So from the device is sticking out and input shaft I think you can see it, this is the input shaft and we are supposed to connect it to the motor which is on the outside just on the right side of the figure.

Once it is connected, when the motor is rotating, the student claims that he can find out the rotational speed of the motor by this device. How does it work? The input shaft, when the motor rotates, it makes the input shaft rotate at the same speed that is fine. The input shaft has an encoder mounted on it with 64 holes, so if the in input shaft rotates, the encoder will also rotate and it has it has 64 holes on its periphery, which means that it will be sending out 64 pulses per rotation and those pulses are shown and they have an AND gate through which they are passing and moving on to a position down counter and they are down counting it, the content of the down counter is getting reduced by these pulses.

The 8-bit position down counters are down counted by the pulses they receive each pulse making one decrement. After connecting the input shaft to the rotating shaft of the motor, whose RPM is to be measured, the 2 position down counters are filled completely and the instrument is started. So when we start at t = 0, the position down counter 1 and position down counter 2 will be completely filled up that means they will contain 8 1s amounting to 255 each in the instrument will be started. The moment instrument is started, there will be 2 inputs one is from the motor rotation and another is from the pulse generator which is at the bottom emitting 64 pulses per second.

So that that seems to be quite simple because the position down counter will be down counted by these 2 inputs as shown in the figure, pulse generator is sending inputs to position down counter 2 and motor is sending pulses to the position down counter 1, so this is the statement problem statement, let’s see what is being asked.

(Refer Slide Time: 04:48)



How much is the speed of the motor shaft if instrument shows unchanging values of position down counter 1 = 0 and position down counter 2 = 204 after sometime. So the statement is that the student finds that after sometime whatever be the time, there is a study state where position down counter 1 is showing 0 and position down counter 2 is stuck at 204 and is not changing, so in that case, how much is the speed of the motor? That is the 1st question. 2nd question is, how much is the speed of the motor shaft if the instrument shows unchanging values of position down counter 1 = 204 and position down counter 2 = 0 after sometime, just the opposite, so let's see how we can solve the problem.

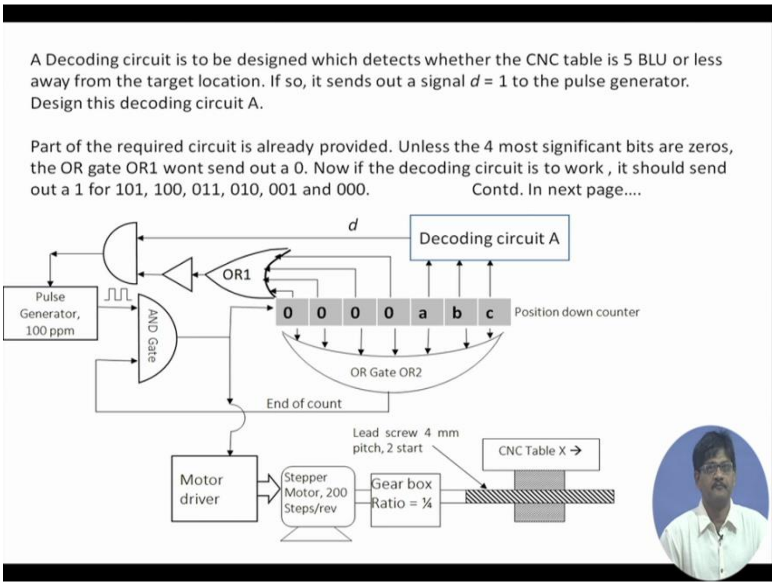
We can understand this that once one of the position down value is becoming 0, further change in values is not possible because the system is configured in such a way that if one of them is becoming 0 here, the OR gates will be emitting 0 and the and the AND gate will be emitting 0 and further pulses will not be entering into this system of 2 position down counter, so this is established one of them becomes 0 the values becomes steady. Once we have understood that, let's say the motor speed is N rotations per second and therefore, the pulse rate output from the encoder, there is only one encoder here, the pulse rate output from the encoder must be 64×N pulses per second because it is rotating N times emitting 64 pulses per rotation therefore N×64 will be the number of pulses per second.

Position down counter 1 contains 255 at the start, so in the 1st problem 255 pulses must have been emitted by the encoder after which the position down counter became 0. So position down counter 1 became 0 after 255 was down counted, so let's say how much time it was taking to do that, t must be equal to that means time at which the position down counter 1 became 0, it must be 255 divided by the rate of pulses entering per second this must be the time in seconds. And with the same logic we can say by the same time position down counter 2 was decremented by 51 bits that is 255 - 204, these many bits had been reduced by that same time.

So the time can also be computed as 51 divided by the pulse rate emitted by the pulse generator, which is 64 versus per second. Having understood that these 2 times must be the same because after this both of them reached a steady state, they have not changed after this point in time therefore, we can equate these two times which means 255/64N must be equal to 51/ 64, so let 64 will cancel out and N will be solved as 5 rotation per second, which is 300 rpm, so the motor must have been rotating at 300 rpm.

Second problem, 2nd problem says just opposite, position down counter 1 is 204 while position down counter 2 is 0. Here again following the same logic we can we can equate the times in the two cases so that now 51 divided by 64N must be = 255/ 64, which shows that if you cancel out 64 from 2 sides then 255 into N must be = 51 therefore, N = 51/255, which is 0.2 rotations per seconds, which means 15 rpm. So in the 2^{nd} case, the motor must have been rotating at 15 rpm. Let's take another problem.

(Refer Slide Time: 09:47)



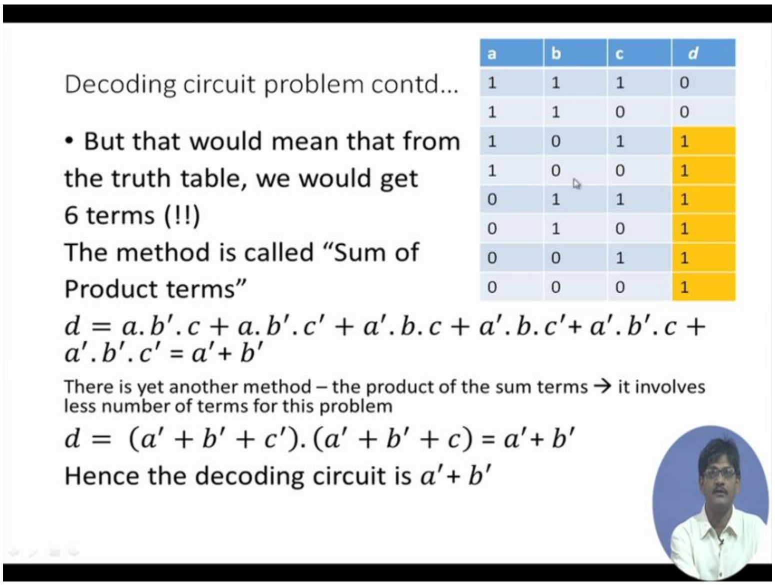
Another problem states; a decoding circuit is to be designed which detects whether the CNC table is 5 basic length unit or less away from the target location. So what do we understand, here there is a motor driver and a stepper motor and all these things that we have studied so many times now through so many problems and theory discussions, so this is the CNC table. The CNC table has to reach a particular position and when it is 5 basic length unit away or less, the decoding circuit here has to sense that and accordingly send out a signal d = 1 during that particular happening.

What is that, that is the table is 5 basic length units or less away from the target, this is the condition. So d has to be = 1 for those events, so this decoding circuit has to be designed, how do we do that and what are the inputs to this decoding circuit? Parts of the required circuit is already provided, in what way? It is argued that when this particular position down counter of this point-to-point control circuit, when the position down counter is having 4 bits, 0 the most significant 4 bits when they are all 0, then only we can be considering the cases of 5 basic length units or less.

If they are non-zeros, the question of being 5 basic length units away does not arise. So 1^{st} of all, this part of the circuit is already designed, all these are bunched together, sent to an OR gate and only when they are all 0, the OR gate will meet 0 that is passed through an inverter or NOT gate so that we will always get a 1 when there is a chance of the basic length unit being 5 or less from the target. So this becoming 1, it is an automatic check for all those cases which we need not consider. With this 1, the decoding circuit has to send its information, so the decoding circuit therefore has to be, has to just check whether we are 5 or less here or not, so let's see how this can be worked out.

From the truth tables that we have discussed previously, we have noticed that whenever a particular a signal or variable is developing values of 1, it can be represented by some of the products terms and in this case we can definitely say therefore, the circuit has to send out a 1 for that means d has to be 1 for 101, abc = 101 or abc = 100 that means 4 or 3 011 or abc = 010 or abc = 001 and abc = 000, these are the cases where decoding circuit has to pass out d = 1.

(Refer Slide Time: 13:45)

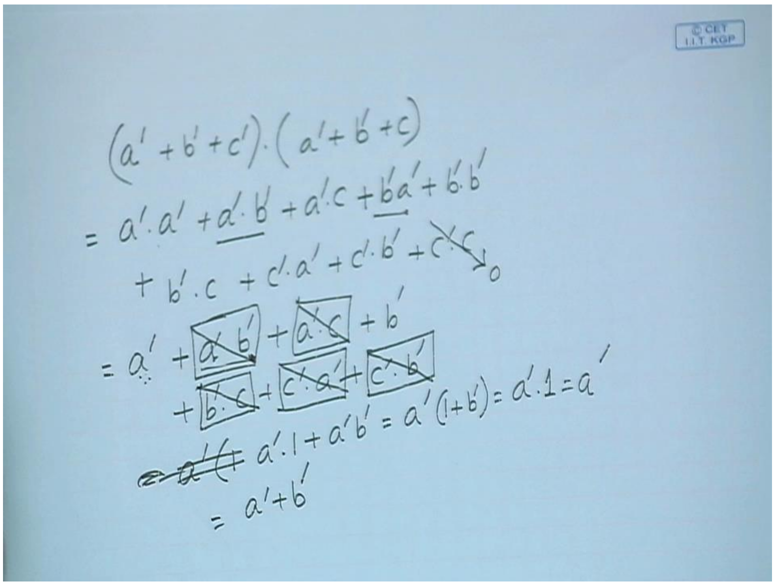


So if we if we look at this, I have highlighted all those cases and we have 6 such cases and therefore, we can say that d will be represented by this particular logic circuit. I mean if it is physically realized as a logic circuit and it can be simplified to a′+ b′ ,. But obviously, even this being strain to deal with find out from this table 6 such terms and simplify them with Boolean algebra and ultimately get a′+ b′, is there any other method existing for such cases where almost all the cases have to be considered?

There is, and the method is called product of the sum terms. The previous one was sum of the products terms and the present case is product of the sum term. How does it work, it works this way just like I can make building blocks representing all the 1 values in the truth table, I can have a building block representing each of these 0 values and then AND them, that means it has to give 0 in this case. Let’s see how it is built up, this one, a′+ b′+ c′ representing the 1st case.

What we mean is that if a signal is going to be 0 for a = 1,b= 1, c = 1 and none of the others in that case we can have a circuit represented by a′ OR b′OR c′ and it is going to give a 0 only in this particular combination, in other combinations it is not. So, let’s look at the other conditions, so just 2 conditions have to be considered this way and this can be simplified very fast, just will have to multiply these terms and you will get a ′ +b′. Shall we try this out actually; let’s do that on this particular piece of paper.

(Refer Slide Time: 16:26)

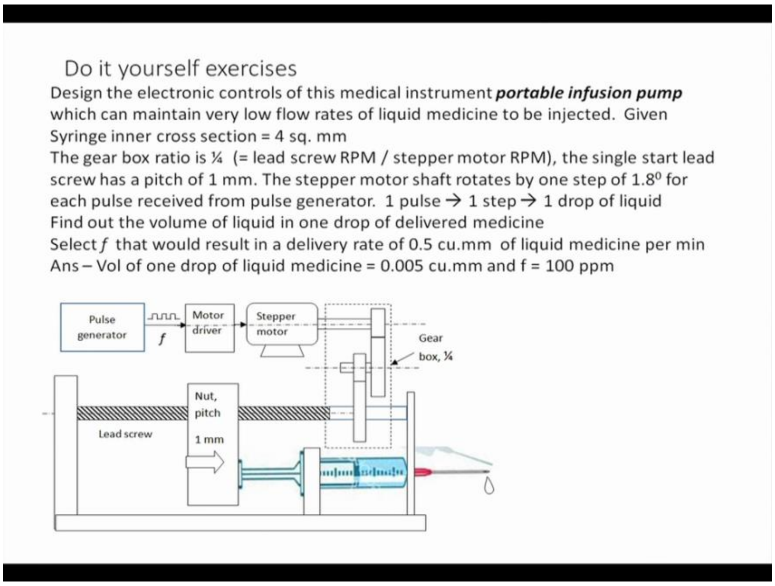


So we are having a′ + b′ + c′ as the 1st place multiplied by a′ + b′ + c, so if we simply find this we will get a′×a′ + a′×b′ + a′×c + b′×a′ + b′×b′ + b′×c + c′×a′ + c′×b′ + c′ ×c. This one cancels out; c′×c goes out and we can further simplify this as, a′×a′ is nothing but a′ + a′×b′ + a′×c+ , see we have two of these a′ b′ terms, so we need not repeat it again, so this is gone, + b′×b′ is nothing but b′ + b′×c+ c′×a′ + c′×b′.

So in this case, this one a′ +a′b′, a′ is this term can be removed, why? We can argue that this can be expanded as a′ + a′× a′ + b′ okay, so we can have simply a′ + b′ in that case. How further can we reduce this? Let’s see we are having okay sorry a′×1 + a′×b′, I have written down the 1st two terms and therefore I can write it as a′×1 +b′ = a′×1, which means a′ only. So that means this term is gone, a′ +a′×b′ is nothing but a′ therefore, this term will also be gone, a′ +a′×c, this term will also be gone, so let's remove them, this is gone, this is gone, this is gone. This is gone because b′ +b′×c′, this is gone and this is also gone.×

Having removed all of these by this argument that a′ +a′×b′ can simply be reduced to a′×1 +a′×b′ and therefore this = a′ only, so this sort of a′ is gulping down all its subsets hence, this is equal to a′ +b′, so this way simplification of these circuits can be done and we are starting with only 2 terms and it is becoming a faster calculation. So the decoding circuit which should be used in order to check whether machine is 5 basic length units away from the target position or not or less can be done by a′ +b′, where a and b they are two bits as shown in the figure.

(Refer Slide Time: 20:43)

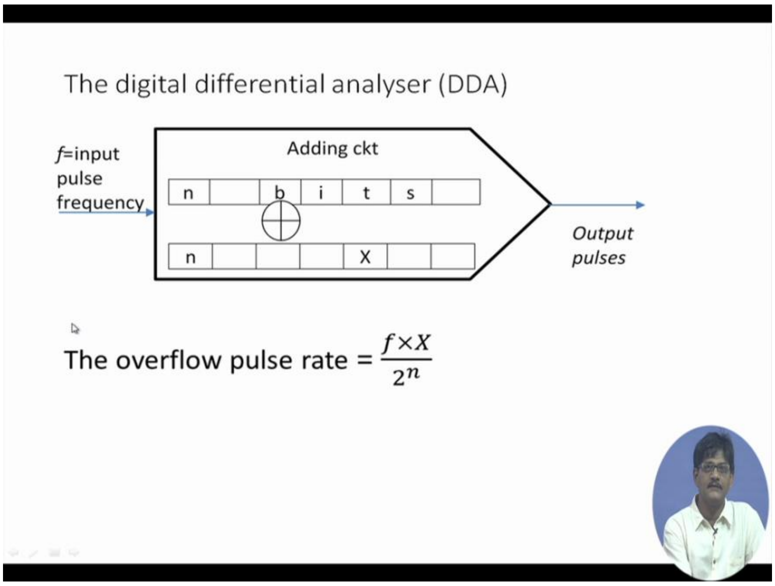


. Design the electronic control of this medical instrument portable infusion pump, which can maintain very low flow rates of liquid medicine to be injected, given the syringe; it has a syringe here okay. There is a syringe and it is getting pushed by this nut, which is acting as a piston, it is pushing on the piston of the syringe and the screw is rotating because of which the nut is moving and pushing.

How is the screw rotating? Through a gearbox by a stepper and with a pulse generator also. The whole thing is a very small portable item, which can administer medicines at very low rate and it can be carried on the pocket of the patient. The syringe inner cross-section have been given to be 4 square millimeters, the gearbox ratio which is shown in the figure is one fourth and the single start lead screw has a pitch of 1 millimeter, the motor the stepper motor shaft rotates by one step, these things are known 1.8 degrees. 1 pulse creates 1 step and that gives rise to one drop of liquid medicine getting administered through the syringe into the body of the patient. So what are we supposed to find out?

Find out the volume of liquid in one drop of delivered medicine and select f that means the frequency of the pulse generator that would result in the delivery of 0.5 cubic millimeters of liquid medicine per minute. Answers are given to be 0.005 cubic millimeter and f = 100 pulses per minute, you should try it yourself.

(Refer Slide Time: 23:08)

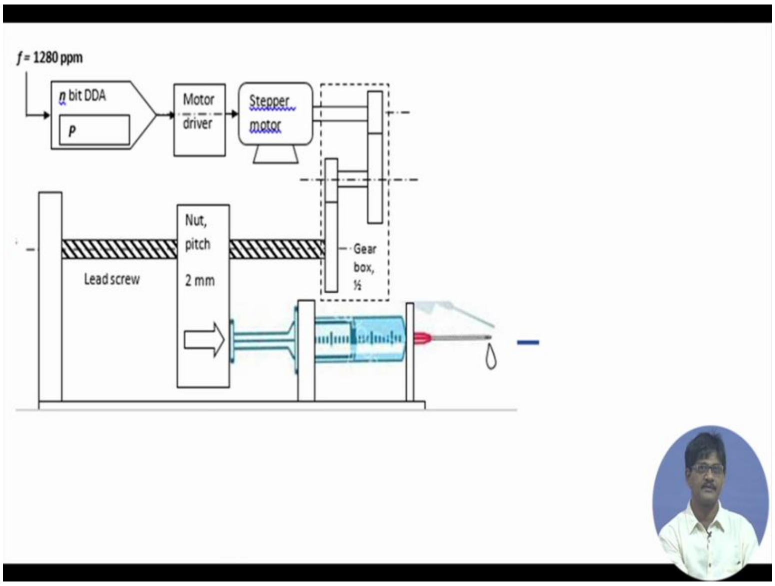


Last of all we are going to discuss a binary circuit, which has quite a wide application in many digital equipment and in CNC machine tools as well for it is a building block of interpolator, hardware interpolator and has been given a very good treatment in Yoram Koren's computer controller of manufacturing systems published by McGraw-Hill, so let's look at this digital differential analyzer the building block of many circuits used in many applications, CNC machine tools and others. 1st of all there will be an input pulse frequency going into the digital device and it energies an adding circuit that means 2 terms will be added to each other.

There will be 2 counters containing 2 numbers and these 2 numbers will be added together. Initially, the upper counter will be empty and the lower counter will be containing a number. In this case, we have designated it as X, so X gets added to 0, then X gets added to X then X gets added to 2X, this way X will go on getting added to itself as these additions take place, one addition for 1 pulse of the input pulse train. And it can be shown that the output pulses will be caused by the overflow of the counter at the top because if you go on adding something to itself and that has a finite size okay, it is a binary device having a definite number of binary bits.

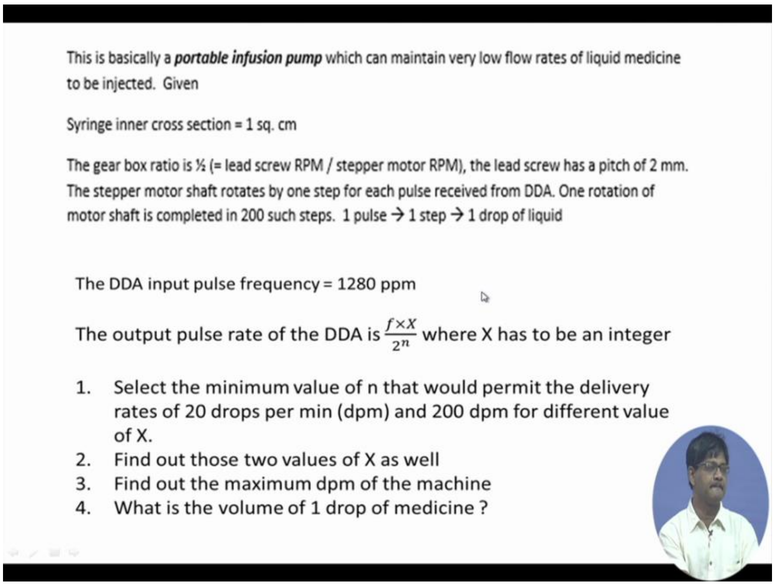
So after some time it will overflow and these overflow pulses are collected as output pulses and their rate can be shown to be f×X divided by 2^n. So we will accept this particular rate and see how this is utilized. So the overflow or output pulse rate = f× X/ 2^n. Let's take the same very same problem and apply this particular DDA here.

(Refer Slide Time: 25:57)



What are we having, instead of a pulse generator we are having a digital differential analyzer here with X = P some value P is put here and input pulse rate is 1280 pulses per minute, all the other things which we discussed they are the same. Let's see what the problem statement is.

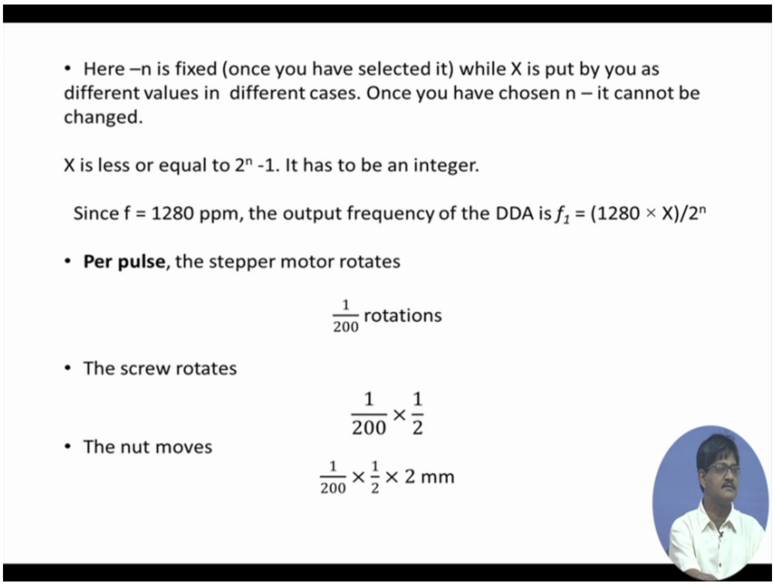
(Refer Slide Time: 26:23)



Problem says, this in this particular portable infusion pump, the syringe inner cross-section is 1 square centimeter, the gear ratio is half and the lead screw pitch is 2 millimeters and everything else is the same, the DDA input pulse frequency is 1280 pulses per minute. The output pulses of the DDA = $f \times X/ 2^n$, which we already have come across. Select the minimum value of n that would permit delivery rates of 20 drops per minute and 200 drops per minute for different values of X.

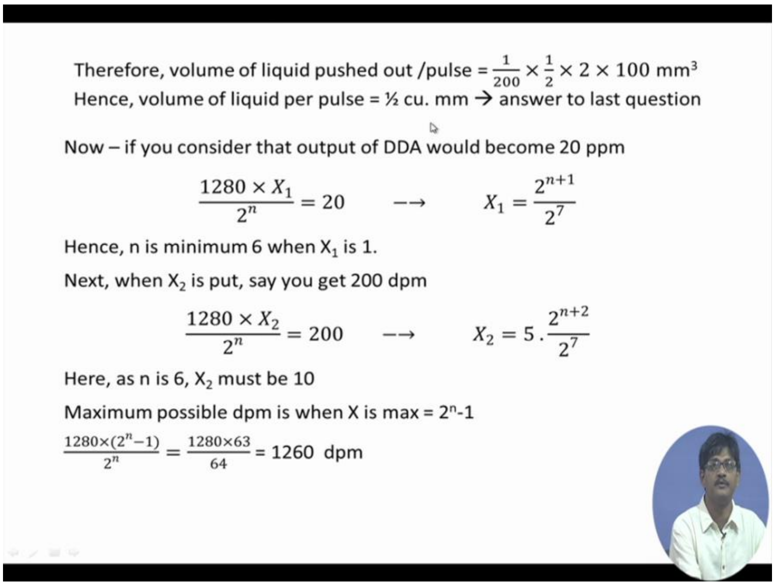
That means we are not designing the DDA, what kind of DDA would serve the purpose, I want 20 drops per minute as well as 200 drops per minute. And find out what are those values of X that will give us these rates, and the maximum drops per minute possible on such a machine and what is the volume of one drop of medicine, so there are 4 questions, so let's look at the answers.

(Refer Slide Time: 27:41)



For example, here n is fixed that means once you have selected the DDA, it is a hardwired circuitry so after that you cannot change. You can again and again for different values for X, X can be = P, X can be = 2 , X can be = 3, this can be set and reset desire, whatever values you want to put. X can be changed, but X has to be an integer and n is fixed once you have selected it. So let’s see what are our requirements and of course X has to be less than the size of these counters, these counters are n bit counters, so the maximum number that they can store is 2 to the power n - 1, X can be less or = 2^n-1 and has to be a whole number that means positive integer.

(Refer Slide Time: 29:19)



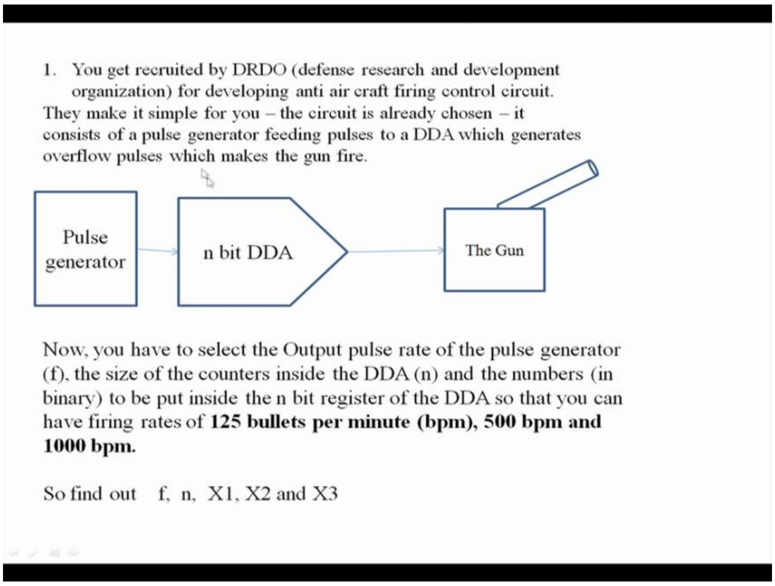
So let's do the calculation that is 1st of all the drops per minute can be calculated very easily, we have done so many calculations up till now of that type. Motor rotates per pulse 1 by 200, screw rotates 1/200 × 1/2, nut rotates 1 / 200 × 1/2 × 2 millimeters from which we can calculate how much is the liquid pushed out multiplied by the cross-sectional area giving rise to half cubic millimeters and that is the volume of liquid medicine administered per drop, so this is answer to the last question. Now if I want 20 pulses per minute to be achieved, what sort of relation do we have?

We have $f \times X$ that is X 1 say / $2^n = 20$. F is given, X is not given neither is n given, but we can argue this way, n should be as low as possible to make the system small and cheap okay, if you take n to be a huge number unnecessarily, you will be incurring more expense, so let's see what is the smallest value of n that we can work with. So for that obviously let's put X 1 to be 1, so if that serves the purpose then we will end up with the smallest value of n. So if X1 = 1. $(1280 \times X1)/2^n=20$ or, $X1=2^{n+1}/2^7$ or, X1=6.

So if X 1 = 1, then n must be = 6 so the minimum possible value of n is 6, when X 1 is 1, mind you X 1 has to be a whole number a non-zero positive integer. That is fine, so is n = 6, let's see what should be the value of X 2 in order to achieve 200 drops per minute. So once again we set this equation and solve for X 2, now n = 6, we cannot work with any other value because as we discussed it is a hardwired circuitry, it has to have the previous value as decided upon in the previous case. So therefore, X 2 will be simply = $(2^6 \times 200) / 1280$ and then we can write $X2=(2^6 \times 20) / 128 = (2^7 \times 10) / 2^7 = 10$. So as n is 6 for 200 dpm, X must be = 10, this is the answer.

So X 1 is 1, X 2 is 10 and the maximum possible dpm will be achieved when X is having its maximum value, which is $2^n - 1$, as we discussed. So f we set it to $2^n - 1$ which means 63, 2^6 is 64 and $2^6 - 1$ is 63. So 1280×63 / 64 = 1260 drops per minute, this is the answer to the maximum possible dpm. We have I think just enough time for discussing one more interesting problem. We are having a DDA, let's see…

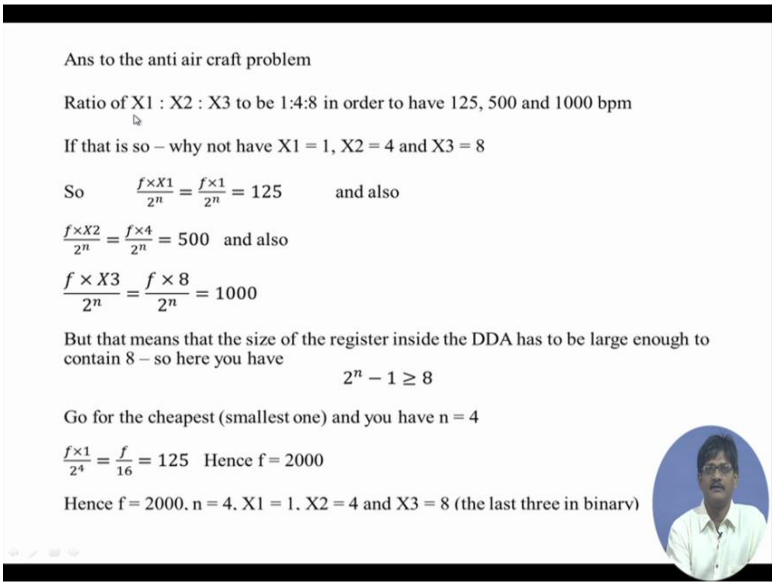
(Refer Slide Time: 33:25)



You are recruited by DRDO (Defense Research Development Organisation) for developing anti-aircraft firing control circuit. They make it simple for you, the circuit is already chosen, it consists of a pulse generator there and n bit DDA and then that huge gun. It consists ofapulse generator reading pulses to a DDA, which generates overflow pulses which will make the gun fire. So whenever you develop overflow pulses, out goes a pulse here and the gun fires, that is quite simple. What do you have to do?

Now you have to select the output pulse rate of the pulse generator, the size of the counter inside DDA and the numbers in binary to be put inside in the n bit register of the DDA so that you can have firing rates of 125 bullets per minute, 500 bpm and 1000 bpm, these many bullets per minute, these 3 options should be available. What is given? Nothing practically, I do not know the pulse generator frequency, I do not know n bit, I do not know what are the numbers to be put nothing, but I only know that I have to make it as cheap and inexpensive and simple as possible, so I will find out f, n, X 1, X 2, X 3 corresponding to 125, 500 and 1000. pulses per minute output.

(Refer Slide Time: 35:13)



So coming to the question, 1st we decide that since X 1, X 2, X 3 their ratio decides the firing rates because ultimately they are = F×X 1 / 2^n, so f is the same for all three and 2^n is the same for all 3, so the only thing which is different is X1, X 2, X 3, so that the ratio of the firing rate essentially be X 1 is to X 2 is to X 3 and which means they should be in the ratio of 1**:** 4 **:** 8, so why not choose X 1 = 1, X 2 = 4 and X 3 = 8 that will solve my purpose, since they have to be integers lets us take the smallest values, so that is good.

So we have f × 1 divided by 2 to the power n = 125, so that makes it extremely simple but we still have not decided upon F neither have we decided upon n, so how do we do that? So but that means that 8 has to be accommodated inside the binary counter, which is having n bits, so we understand that 8 has to be either = 2 to the power n - 1 or less than that and what can be the cheapest case when n = 4, n = 4 can accommodate the maximum value of 15 1111 and that can definitely accommodate 8. Why not go for n = 3 because n = 3 will accommodate maximum 111, which is 7 it cannot accommodate 8 and therefore we go for n = 4.

Now last of all we have f left to be calculated, so we understand that f × 1 divided by 2 to the power 4 must be = 125 and that gives us F = 125 × 16, which means simply f= 2000, so we have f = 2000, n = 4 and X 1, X 2, X 3 = 1, 4 and 8 respectively. Okay, I think the time is up so I will find another time for discussing this, thank you very much.