1:CYCL_EXC

Cycle Execution

Name:		Time stamp	Code:	28/06/09
Author:			Interface:	20/01/04
Family:		Lengths	Block:	00980
Version:	0.0		Code:	00772
Code version:	2		Data:	00028

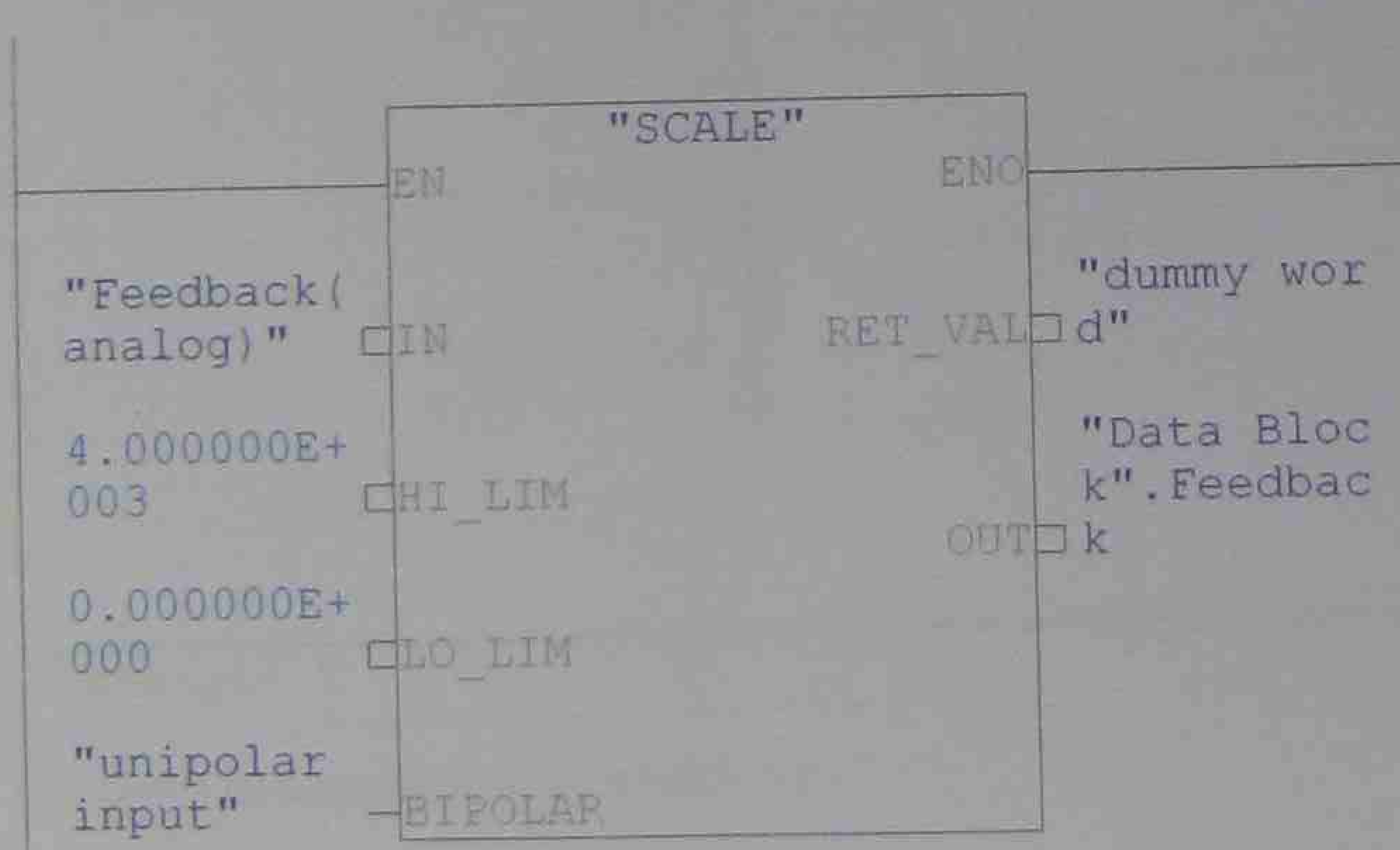
Block: OB1 "Main Program Sweep (Cycle)"

DC Motor speed controller

Address	Declaration	Name	Type	Start value	Comment
0.0	temp	OB1_EV_CLASS	BYTE		Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
1.0	temp	OB1_SCAN_1	BYTE		1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
2.0	temp	OB1_PRIORITY	BYTE		1 (Priority of 1 is lowest)
3.0	temp	OB1_OB_NUMBR	BYTE		1 (Organization block 1, OB1)
4.0	temp	OB1_RESERVED_1	BYTE		Reserved for system
5.0	temp	OB1_RESERVED_2	BYTE		Reserved for system
6.0	temp	OB1_PREV_CYCLE	INT		Cycle time of previous OB1 scan (milliseconds)
8.0	temp	OB1_MIN_CYCLE	INT		Minimum cycle time of OB1 (milliseconds)
10.0	temp	OB1_MAX_CYCLE	INT		Maximum cycle time of OB1 (milliseconds)
12.0	temp	OB1_DATE_TIME	DATE_AND_TIME		Date and time OB1 started

Network: 1

Feedback input, scale speed from 0 to 4000 RPM



CONCLUSION QUESTION

. Hysteresis is used for avoiding noise around setpoint that could lead to continuous switching therefore improve lifetime of switches and components.

2. Proportional control generates an output proportional to error signal and attemp to correct the final output closer to setpoint. The higher gain the more accuracy it acts and final output closer to setpoint.

3. Temperature inertia causes overshoot and only be eliminated by derivative control.

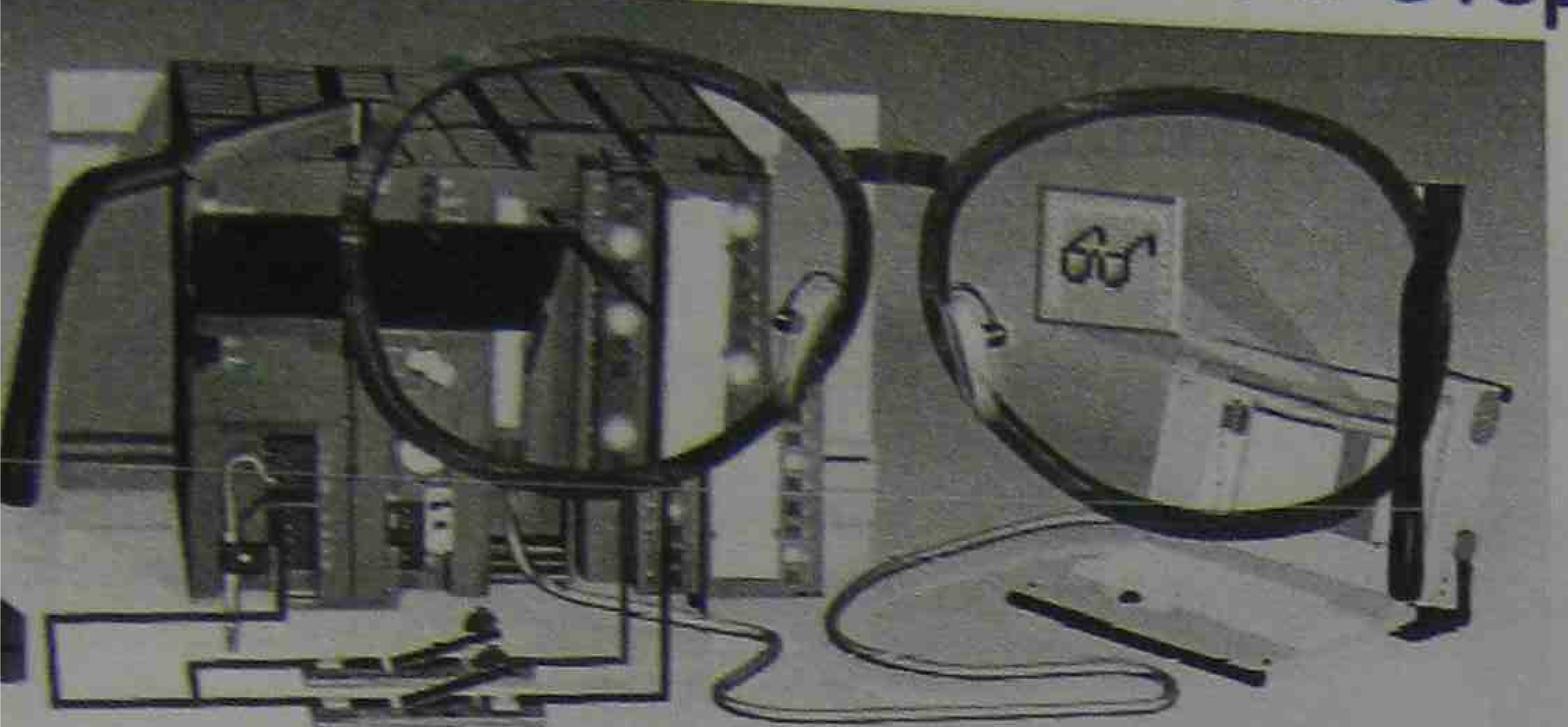
5. RTD : Resistance Temperature Detector
based on property of metallic conductor whose resistance increases with rising of temperature

Others: thermocouple : 500°C ÷ 1500°C
thermister : -100°C ÷ 300°C

4. A 4 - 20 mA current loop is used for some reasons:
- The system needs a current to operate as analog input/output
- Not to be confused between low limit of current as 0°C and non-operate system such as stopped by safety components In other word 0 mA should be understand as not operating or stopped rather than lower limit of analog input.

Minh Van

...LC System Applications -Major Project
Temperature Control with a Step 7 PLC



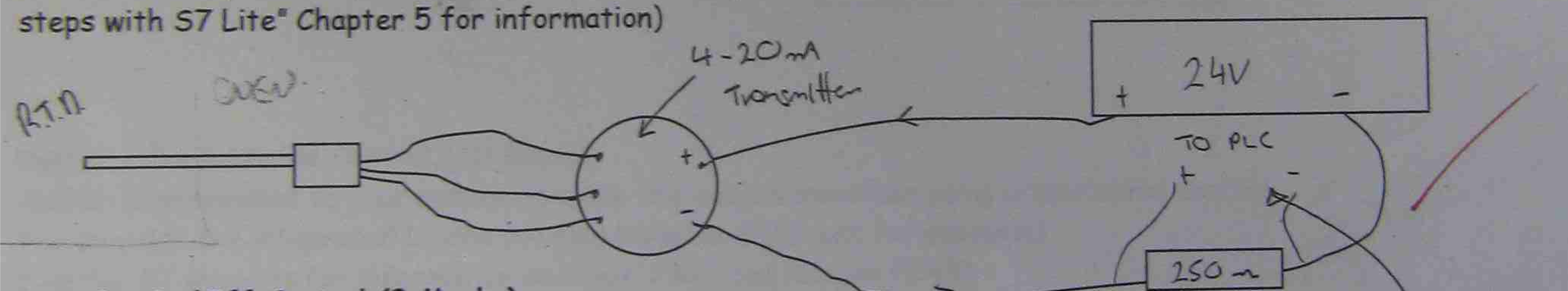
Aim: the aim of this major project is to assess student abilities to apply their knowledge to a new system. Building on previous knowledge the students will adapt this to a newer model PLC and software.

Procedure :

Examine the temperature control system that was used in a previous assignment.
Demonstrate that you are capable of connecting this system to your S7 313C plc.
Familiarize yourself with the S7 "lite" software , use the supplied PDF documents for reference material.

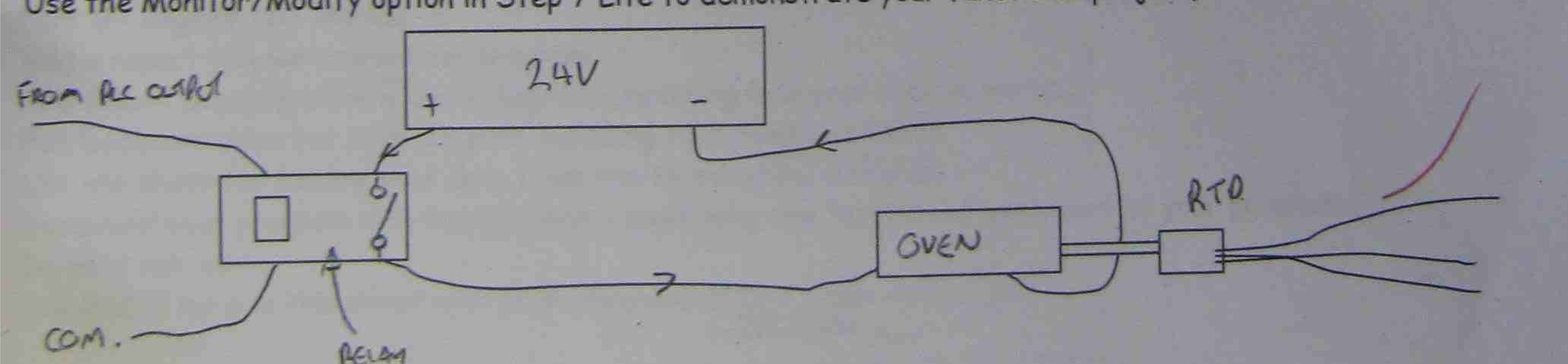
Part A - Analog Inputs.(0 Marks)

Connect the RTD input and scale this input to accurately display the temperature in a variable block or 7 segment display.
Create a full Symbolic table for the project and maintain the Symbolic table as your project develops. (See "First steps with S7 Lite" Chapter 5 for information)



Part B- On/Off Control.(0 Marks)

Use the scaled analog input as a feedback signal to control the temperature in an oven.
Sketch a block diagram of your system and explain the operation in your report.
Use the Thumbwheel switch as your setpoint.
Use the Monitor/Modify option in Step 7 Lite to demonstrate your functional project)

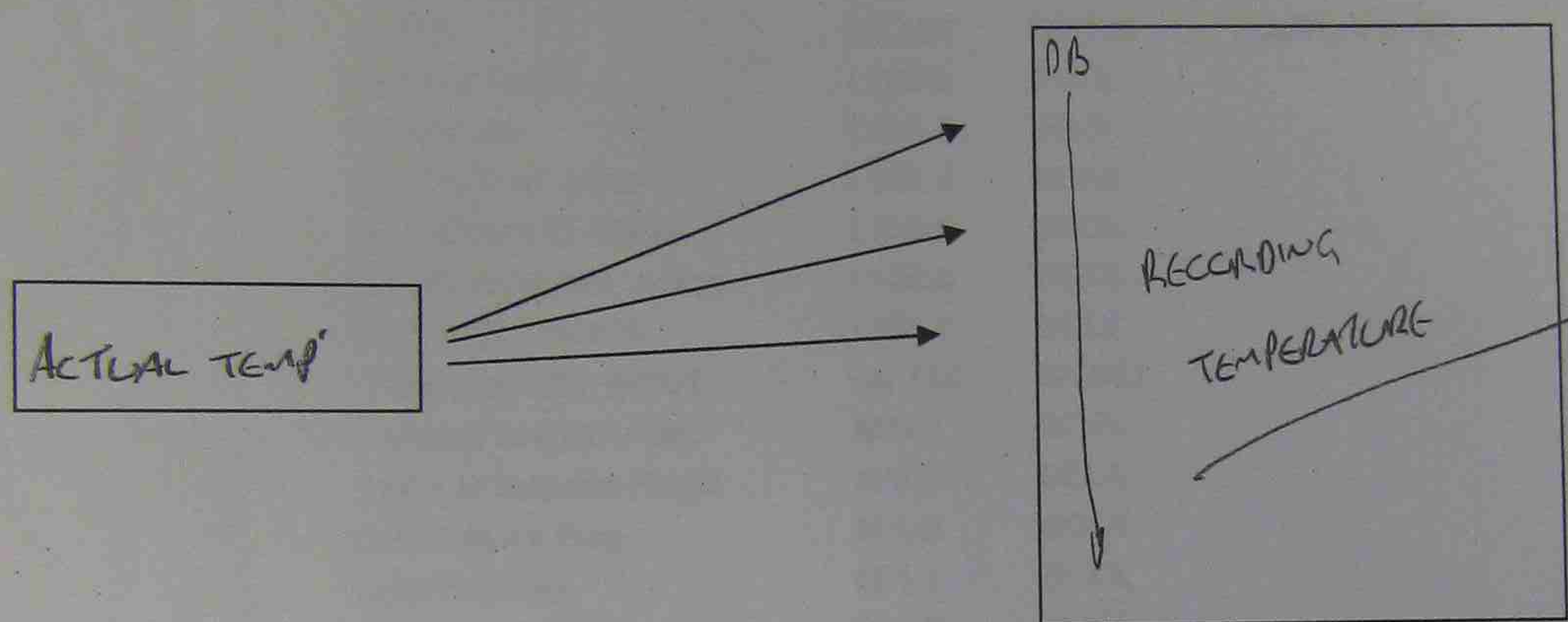


C- Hysteresis.(10 Marks)

dd some Hystersis to you your control system , the Hystersis temperature is set by the thumbwheel switch.
Make sure your symbolic table is up to date and use the Monitor/Modify option in S7 lite to demonstrate your program.

Part D- Data logging.(10 Marks)

Add a data logging system where the temperature can be recorded in a data block over a period of time.
Make the reading every 0.4 seconds for testing purposes.
(You may use a SCADA system to perform this function using trends as an alternative to using a data block in the PLC , it is your choice.)
Using the PLC to record the data must include the use of a "pointer" for full marks.
See the S7 manual for information on Address registers (LAR 1)



Part E - Proportional Control.(10 Marks)

Add an improvement to your system to make the output smoother using proportional control.
You may use the integrated blocks for this purpose. (Pulsegen for example)
See the S7 Manuals for information on Con-C FB41 and Pulsgen FB43)

Part F- PID control.(10 Marks)

Using the integrated special function blocks to control the temperature using PID control.

Part G- Report.(10 Marks)

Add a report to your printed out program.
The report should include a block diagram explaining how your system works.
Full Documentation for your program including your Symbolic Table.
Use one shots for loading your data from the thumbwheel switches.
Document your program with line comments explaining the function of each part of your program.
Do your own work.
You **MUST** hand in this sheet with your assignment! Don't lose it!

Symbol table

Status	Symbol	Address	Data Type	Comment
	DATA RECORDING	DB 10	DB 10	
	CONT_C DB	DB 30	FB 41	
	PULSEGEN DB	DB 40	FB 43	
	SCADA MONITOR	DB 60	DB 60	
	CONT_C	FB 41	FB 41	Continuous Control
	PULSEGEN	FB 43	FB 43	Pulse Generation
	INDEX ADDRESS	FC 1	FC 1	
	DATA RESET	FC 2	FC 2	
	SCALE	FC 105	FC 105	Scaling Values
	Setpoint Select	I 126.0	BOOL	
	Reset Data	I 126.2	BOOL	
	DERIVATIVE Action	I 126.4	BOOL	
	INTEGRATIVE Action	I 126.5	BOOL	
	PROPORTIONAL Action	I 126.6	BOOL	
	DATA RECORDER	I 126.7	BOOL	
	THUMBWHEEL INPUT	IW 124	WORD	
	Oneshot Setpoint Flag 1	M 0.1	BOOL	
	Oneshot Setpoint Flag 2	M 0.2	BOOL	
	Data Record Flag	M 1.0	BOOL	
	Data Full Flag	M 1.1	BOOL	
	Oneshot Data Reset 1	M 1.2	BOOL	
	Oneshot Data Reset 2	M 1.3	BOOL	
	UNUSED_1	M 3.0	BOOL	
	UNUSED_2	M 3.1	BOOL	
	UNUSED_3	M 3.2	BOOL	
	UNUSED_4	M 3.4	BOOL	
	UNUSED_5	M 3.5	BOOL	
	UNUSED_6	M 3.6	BOOL	
	UNUSED_7	M 3.7	BOOL	
	UNUSED_8	M 4.0	BOOL	
	UNUSED_9	M 4.1	BOOL	
	UNUSED_11	M 4.2	BOOL	
	UNUSED_12	M 4.3	BOOL	
	UNUSED_13	M 4.4	BOOL	
	UNUSED_14	M 4.5	BOOL	
	UNUSED_15	M 4.6	BOOL	
	UNUSED_16	M 4.7	BOOL	
	SCALE's Bipolar	M 80.0	BOOL	
	P out	MD 50	DWORD	
	I out	MD 54	DWORD	
	D out	MD 58	DWORD	
	UNUSED_17	MD 62	DWORD	
	UNUSED_18	MD 66	DWORD	
	TEMPERATURE FEEDBACK	MD 100	DWORD	
	Rounded Temp	MD 104	DWORD	
	Temp in BCD	MD 108	DWORD	

Status	Symbol	Address	Data Type	Comment
	Double Integer Setpoint	MD 200	DWORD	
	SETPOINT	MD 204	DWORD	
	Double Integer Gain	MD 208	DWORD	
	GAIN	MD 212	DWORD	
	LMN/INV	MD 220	DWORD	
	Thumbwheel Value 1	MW 20	WORD	
	Masked 2 left digits	MW 24	WORD	
	Shift Righted	MW 26	WORD	
	Integer Setpoint	MW 28	WORD	
	Thumbwheel Value 2	MW 30	WORD	
	Masking Gain	MW 32	WORD	
	Integer Gain	MW 34	WORD	
	SCALE Return Value	MW 40	WORD	
	BCD Temp to Display	MW 110	WORD	
	UNUSED CONT_C Out	MW 120	WORD	
	Index Addressing	MW 140	WORD	
	CYCL_EXC	OB 1	OB 1	Cycle Execution
	CYC_INT5	OB 35	OB 35	Cyclic Interrupt 5
	SCALE Input	PIW 752	WORD	
	HEATER	Q 125.0	BOOL	
	Data Full Indicator	Q 125.2	BOOL	
	7-seg display	QB 124	BYTE	
	Recording Step	T 1	TIMER	
	Data Full Flash 1	T 2	TIMER	
	Data Full Flash 2	T 3	TIMER	

OB35:CYC_INT5

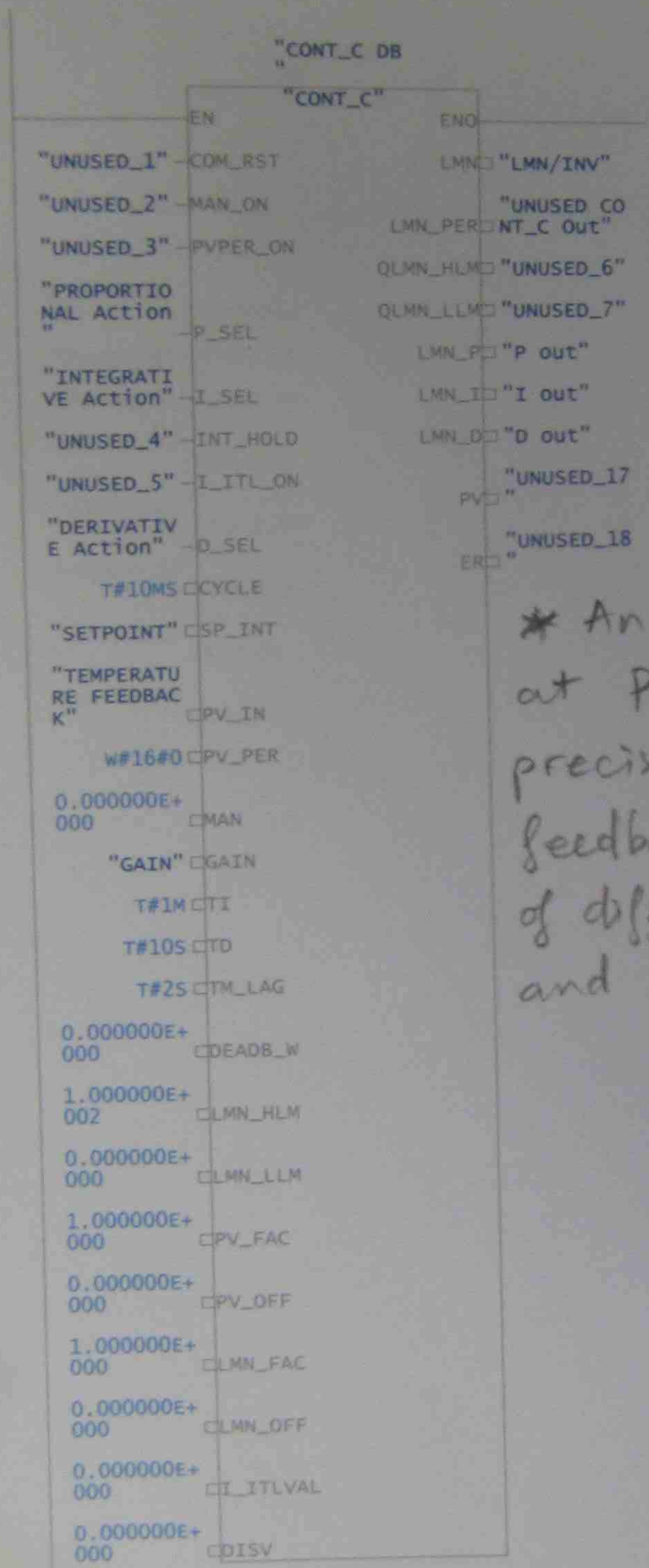
Cyclic Interrupt 5

```
Name:                    Time stamp Code:       27/05/09
Author:                             Interface:  20/01/04
Family:                  Lengths    Block:      00702
Version:       1.0                  Code:       00588
Code version: 2                     Data:       00028
```

Block: OB35 "Cyclic Interrupt"

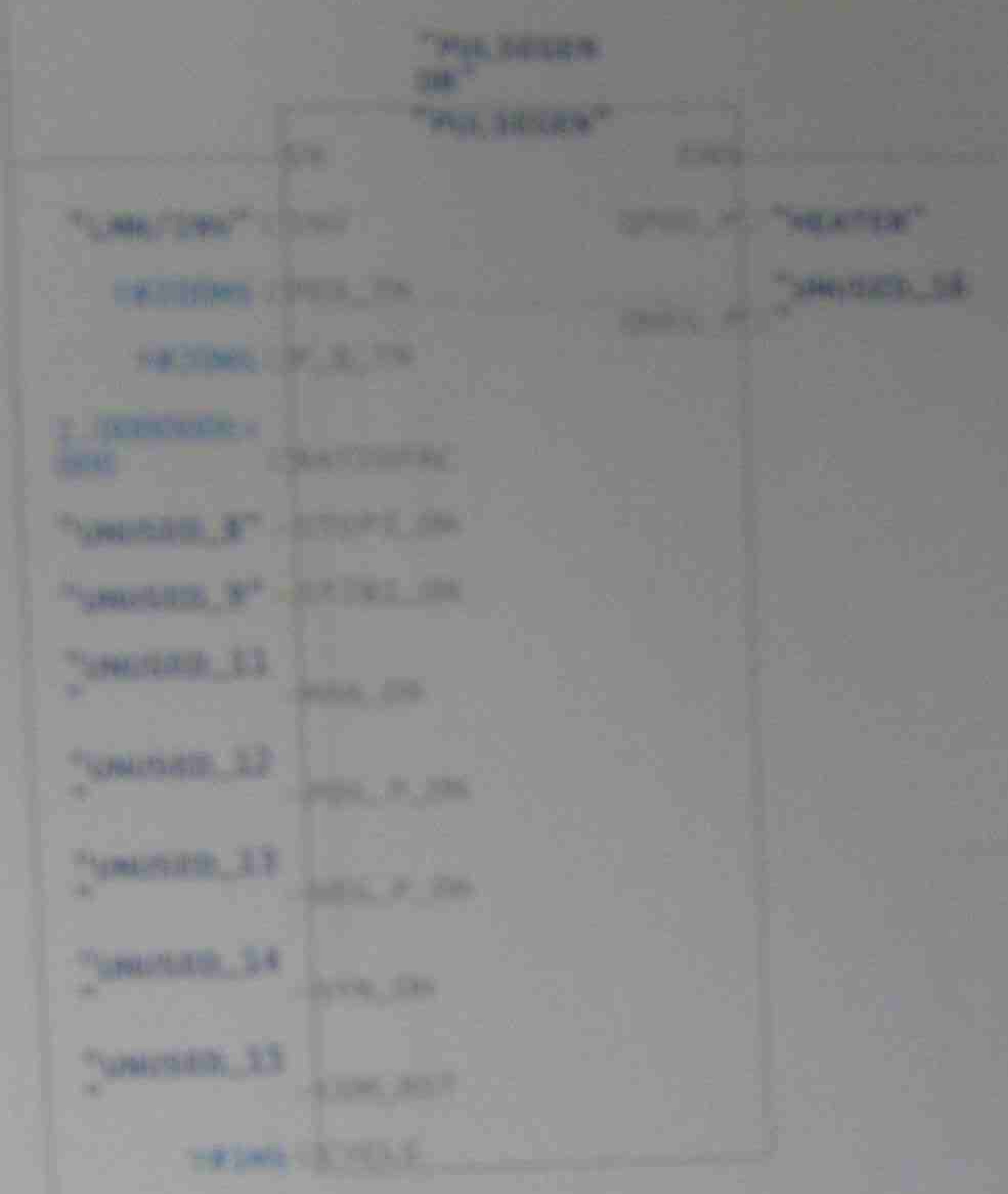
Address	Declaration	Name	Type	Start value	Comment
0.0	temp	OB35_EV_CLASS	BYTE		Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
1.0	temp	OB35_STRT_INF	BYTE		16#36 (OB 35 has started)
2.0	temp	OB35_PRIORITY	BYTE		11 (Priority of 1 is lowest)
3.0	temp	OB35_OB_NUMBR	BYTE		35 (Organization block 35, OB35)
4.0	temp	OB35_RESERVED_1	BYTE		Reserved for system
5.0	temp	OB35_RESERVED_2	BYTE		Reserved for system
6.0	temp	OB35_PHASE_OFFSET	WORD		Phase offset (msec)
8.0	temp	OB35_RESERVED_3	INT		Reserved for system
10.0	temp	OB35_EXC_FREQ	INT		Frequency of execution (msec)
12.0	temp	OB35_DATE_TIME	DATE_AND_TIME		Date and time OB35 started

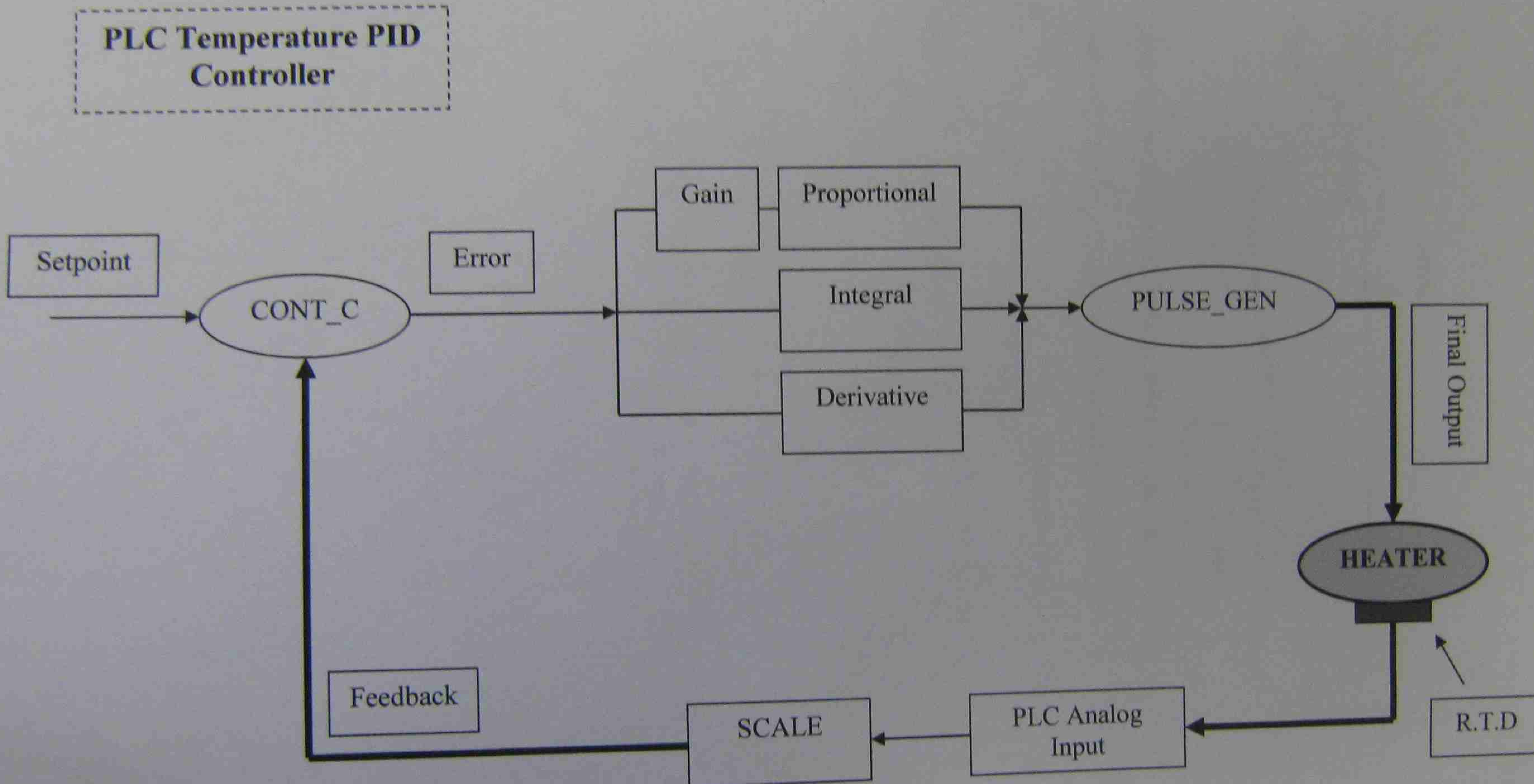
Network: 1



* An analog input should take place at PV_IN directly for more precision rather than using feedback from "SCALE" because of different cycles between OB1 and OB35.

Network: 2




PLC Temperature PID Controller
Setpoint
CONT_C
Error
Gain
Proportional
Integral
Derivative
PULSE_GEN
Final Output
HEATER
R.T.D
PLC Analog Input
SCALE
Feedback

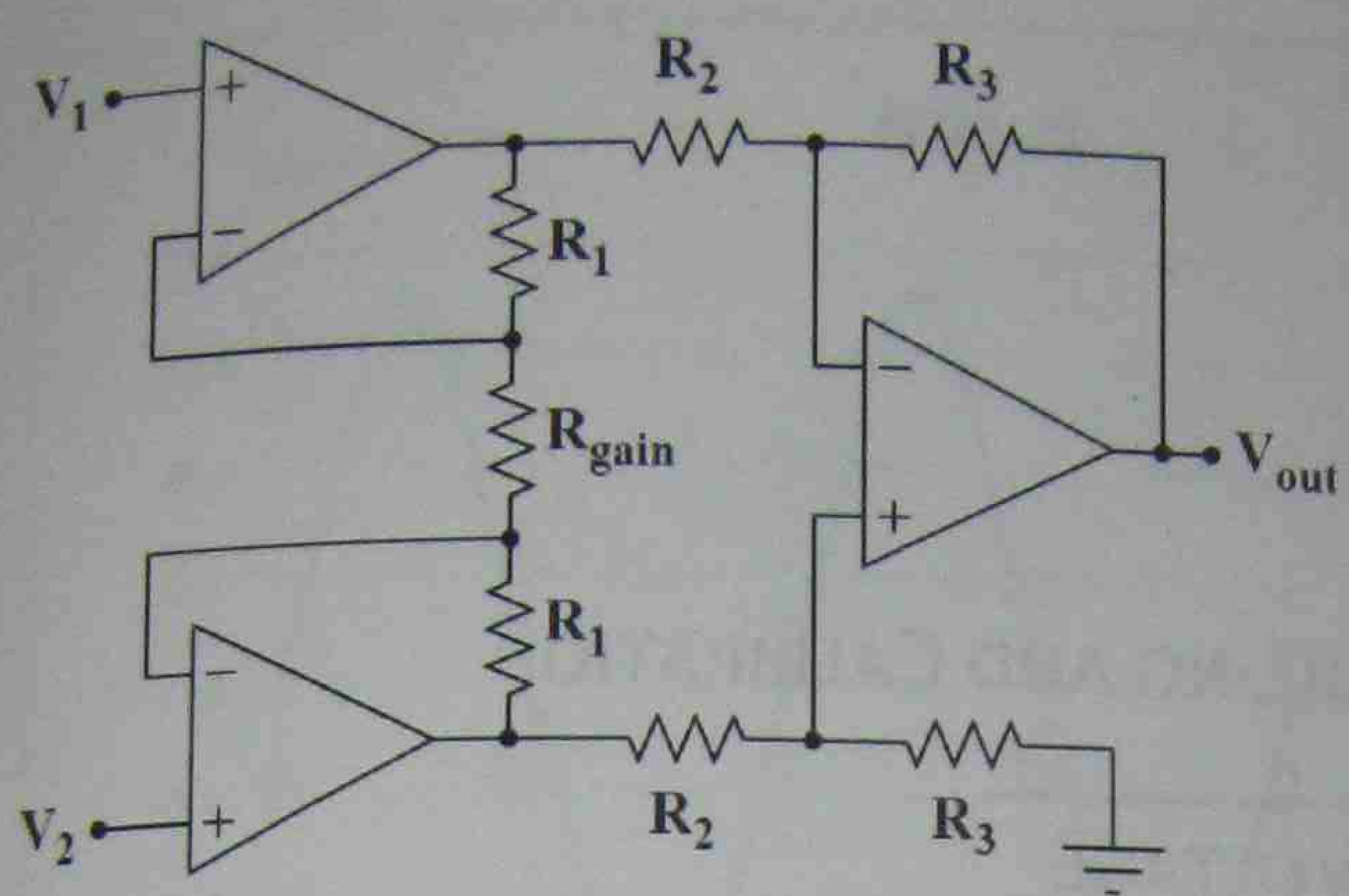
Analog Electronics Report

Instrumentation Amplifier

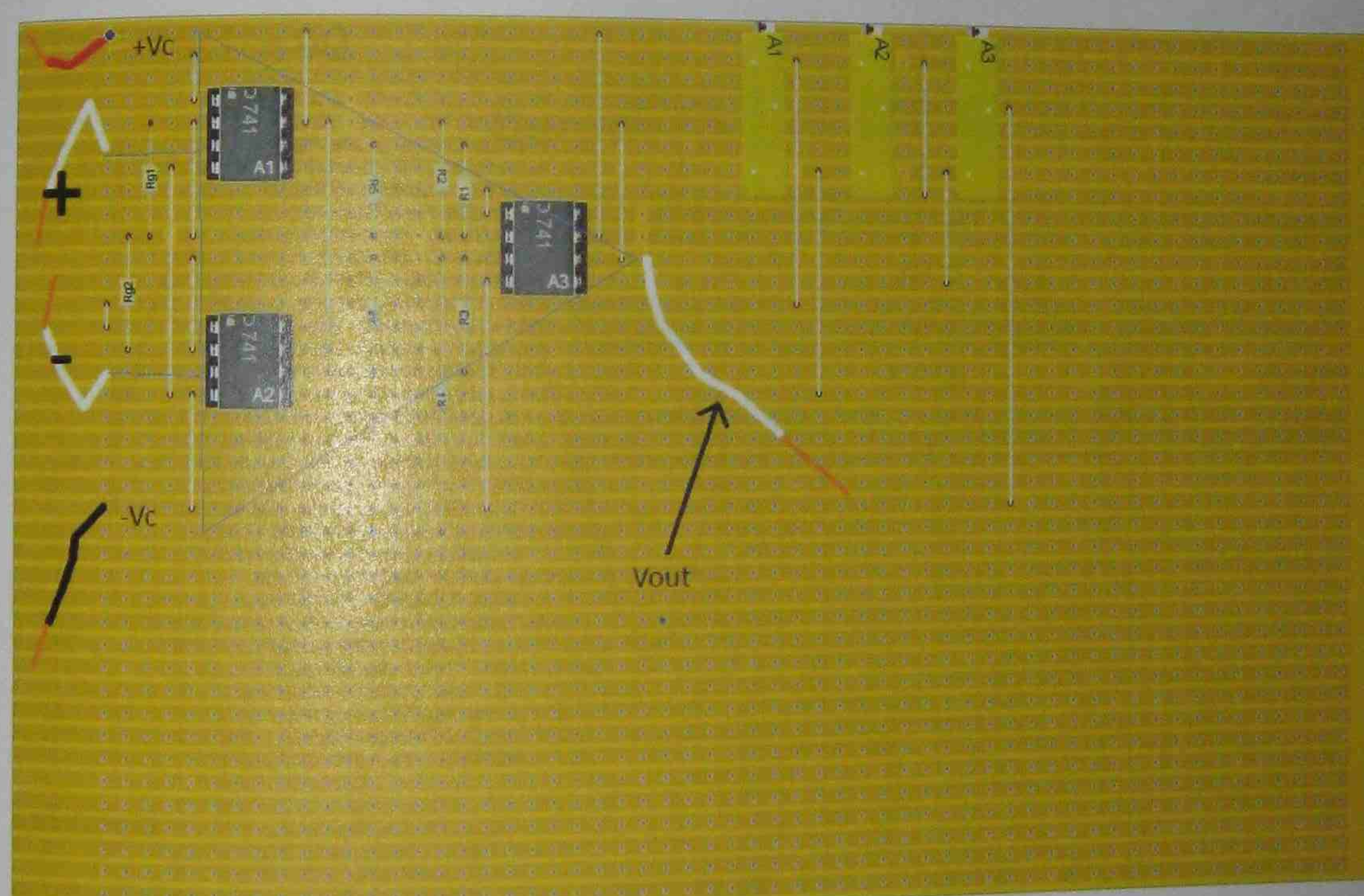
Ali Rezaeisarlak

2009

CIRCUIT OPERATION



Circuit diagram



Circuit Board

Results

Av = 6.1V / 175mV

= 34.85 (Gain)

Rg = 2 x Rf / Av – 1

= 590Ω (Gain Resistance)

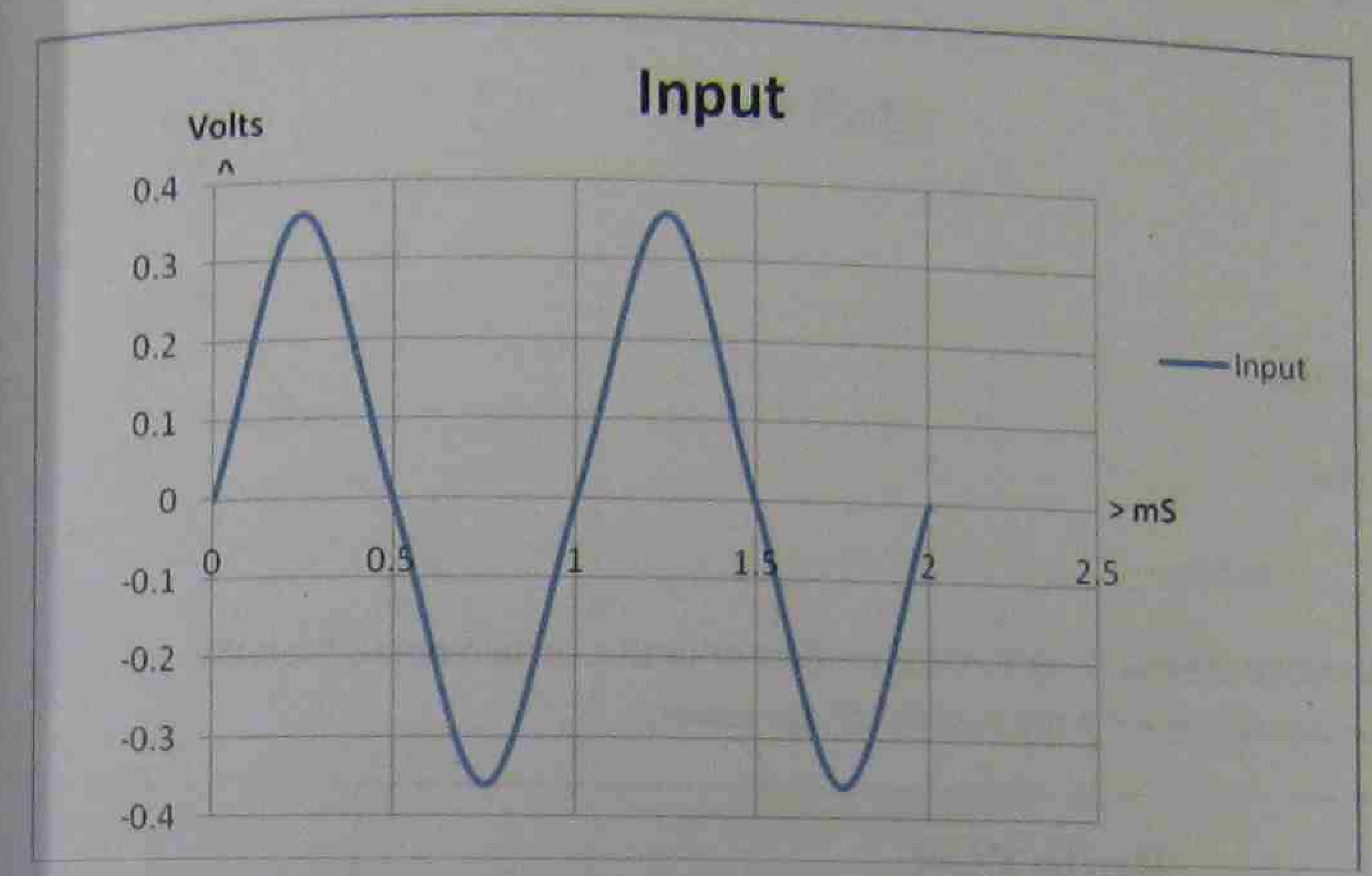
Step 1 – Basic Operation, Nulling and Calibration

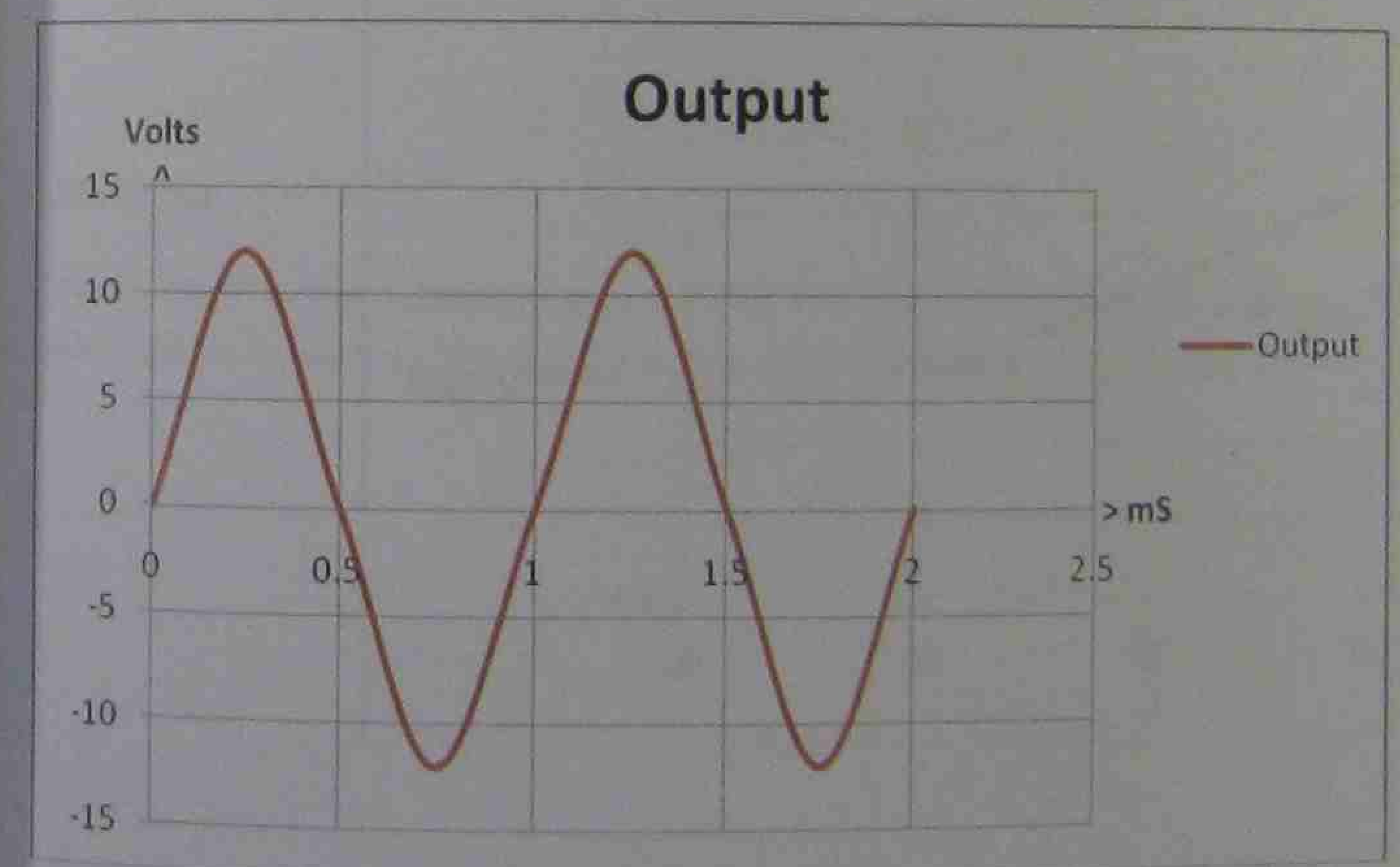
Input Signal	Output Voltage
0mV	13mV
20mV	700mV
40mV	1.4V
60mV	2.1V
80mV	2.8V
100mV	3.5V
120mV	4.2V
140mV	4.9V
160mV	5.6V
180mV	6.3V
200mV	7V

0V = 13mV

175mV = 6.17V

Step 2 – AC Signal Response





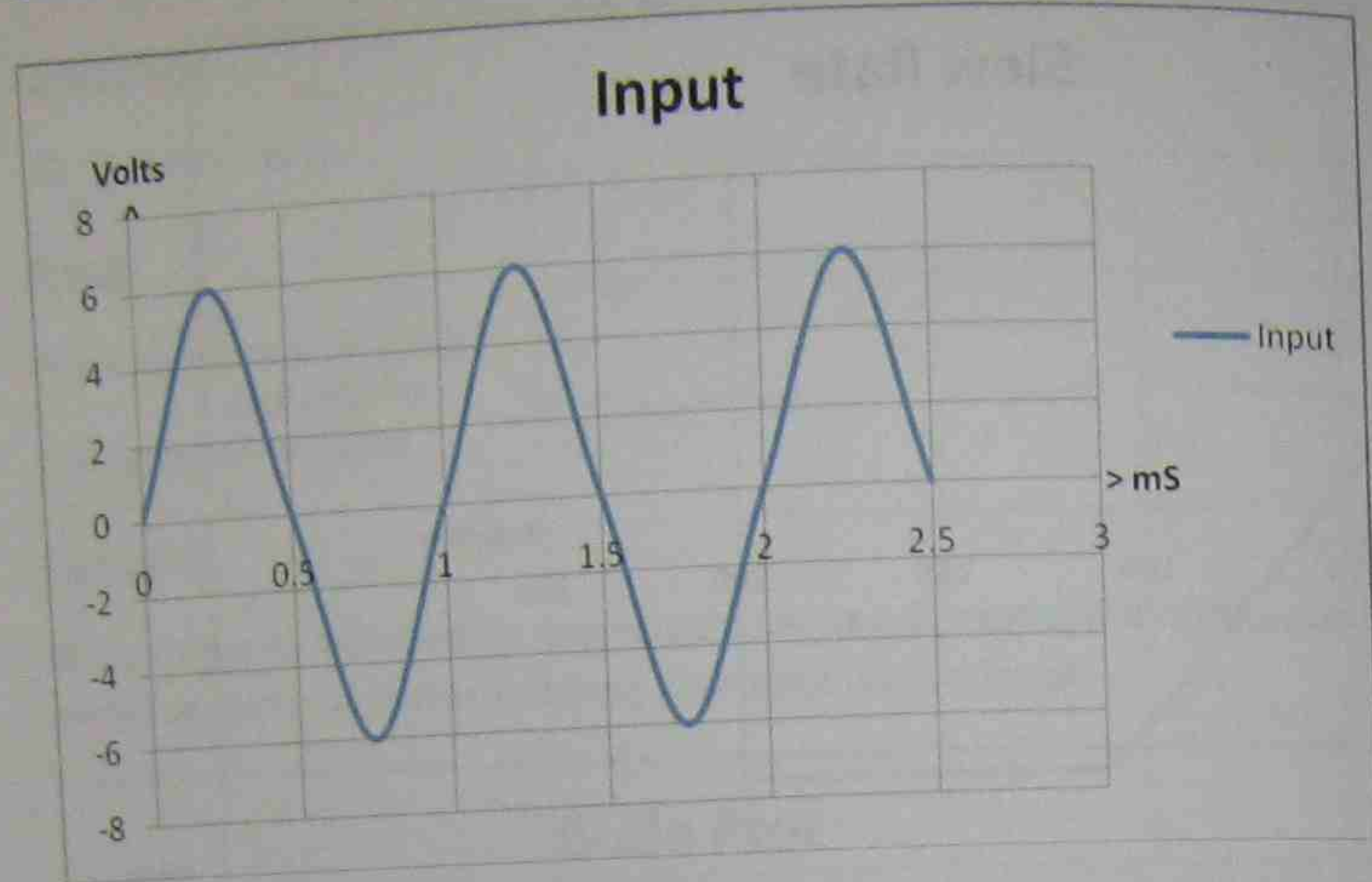
Calculation

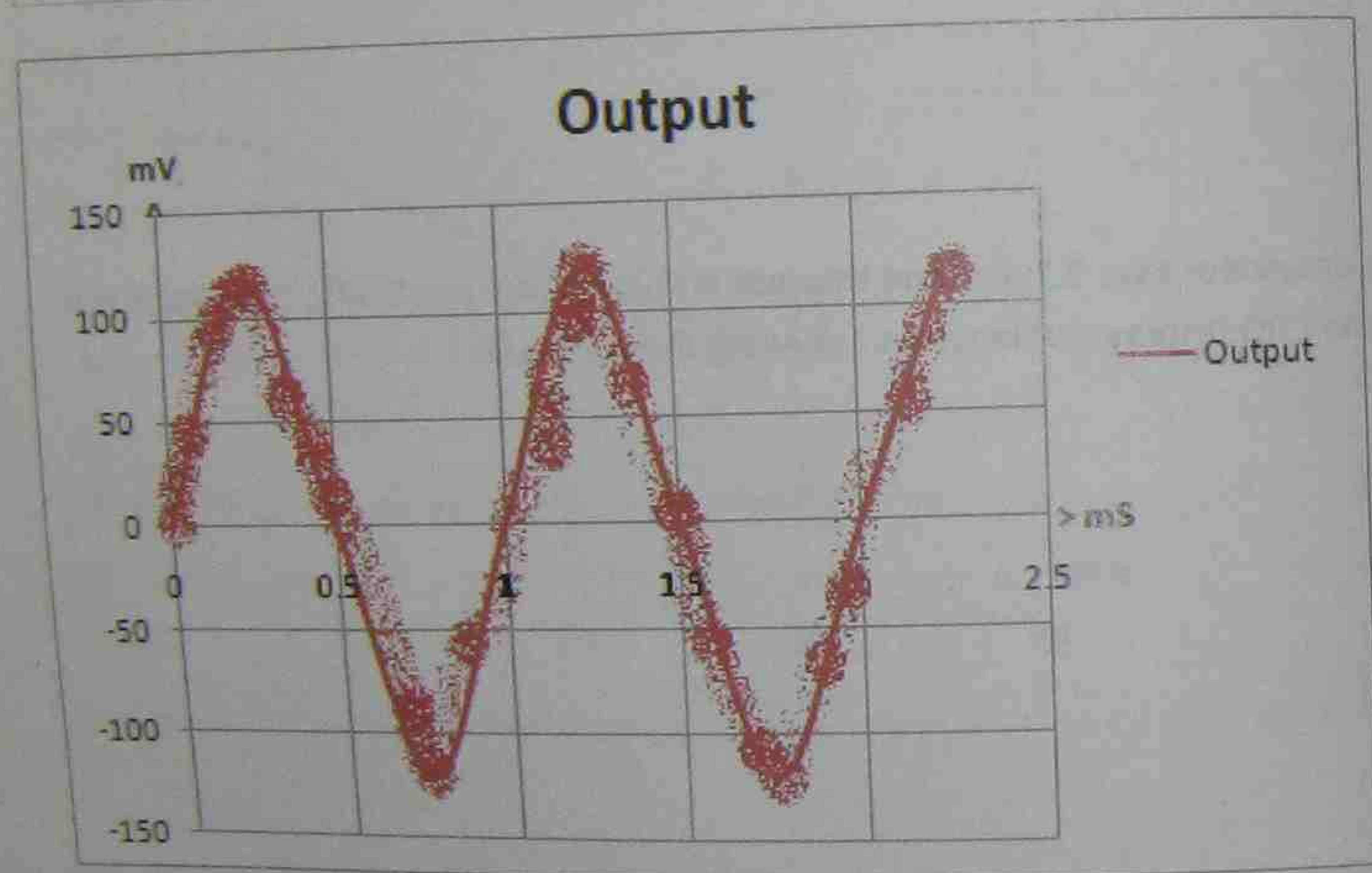
Differential gain = output / input = 12V / 0.36V = 33.33

Comments

My calculated gain was 34.85 however the actual gain was 33.33, I think this is due to the offset voltage.

I had clipping on my output waveform, so I increased the supply voltage from 12Vpk to 18Vpk, and this made the waveform smooth again.





Comments

The output waveform was very fuzzy compared to the input which was completely stable.

Calculations

CMRR = Av / Acm

- where Av is Differential Gain
- and Acm is Common-Mode Gain

Acm = Vo / Vcm

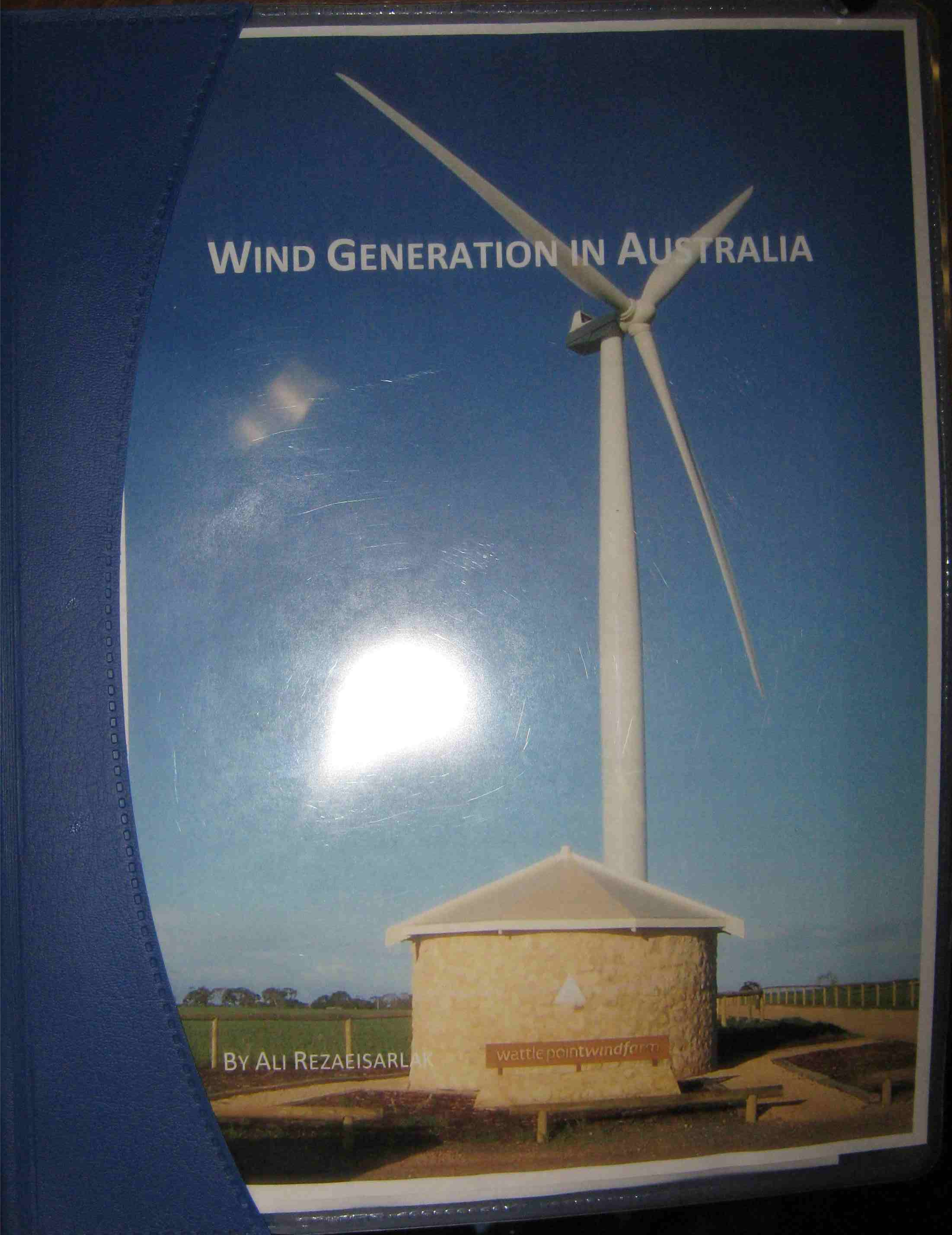
- where Vo is output voltage
- and Vcm is the supply voltage

Acm = 120mV/1[illegible]

= 0.01

CMRR = 20log (34.29/0.0[illegible]

= 70.70db


WIND GENERATION IN AUSTRALIA
BY ALI REZAEISARLAK
wattlepointwindfarm

PURPOSE & OPERATION

The purpose of wind generation is to convert wind energy to electricity. However, the winds energy itself can be used for other purposes such as sailing or flying a kite.

The operation begins by the sun heating the air and the earths surface unevenly which causes wind. Converting the wind energy to electricity is done by using wind turbines. See Figure 1.1

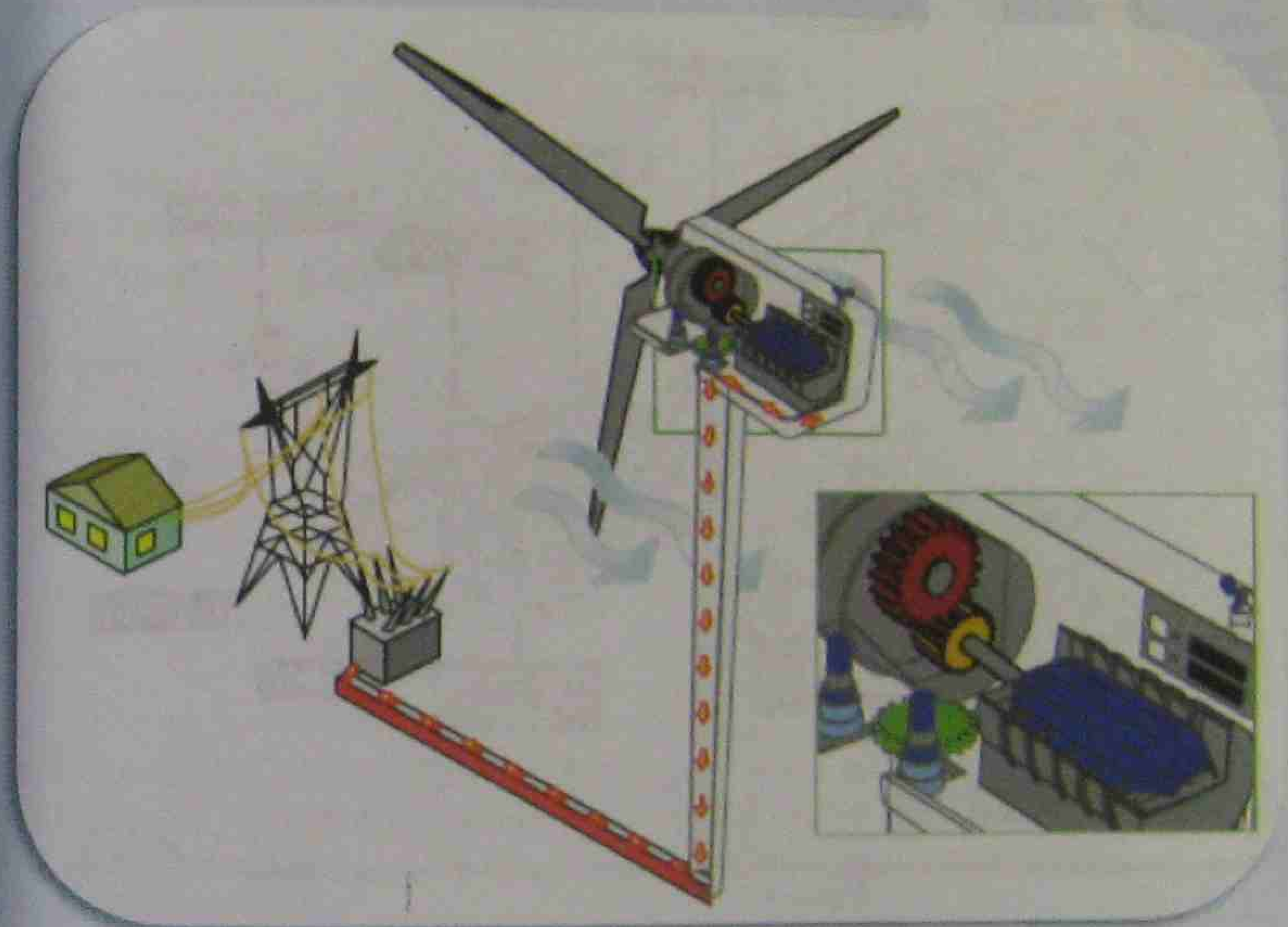


Figure 1.1

Wind turbines work like the opposite of a fan. Instead of using electricity to make wind, wind turbines use wind to make electricity. In a standard wind turbine, the kinetic energy from the wind's moving air molecules is turned into rotational motion by the rotor. The rotor sits at the front of the wind turbine. Wind causes the rotor to turn a shaft that transfers the motion to the nacelle, the housing at the top of a wind turbine tower. Inside the nacelle, the rotating shaft enters a gearbox that increases the shaft's speed of rotation. The fast rotating shaft is connected to a generator that converts the movement into electricity. See Figure 1.2.

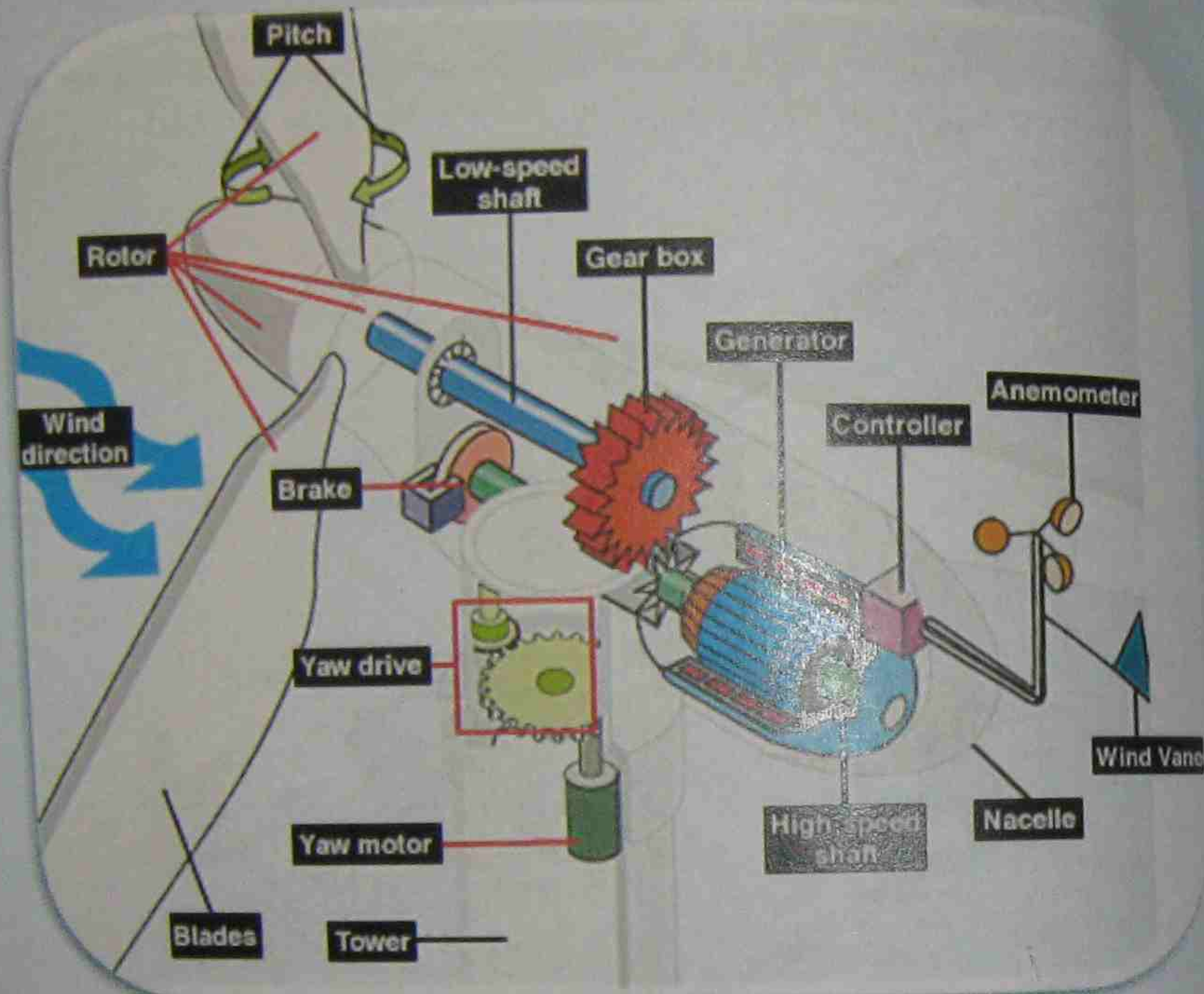


Figure 1.2

Wind turbines are mounted on a tower to capture most of the energy. At 30 meters or more above ground, they can take advantage of faster and more stable wind.

Once the wind has been converted into medium voltage electricity, equivalent to a few hundred volts, that electricity goes down through heavy electric cables in the tower to a transformer which is able to increase the voltage up to a few thousand volts - the distribution voltage. This voltage power travels through underground lines to a collection area where the power can be combined with other wind turbines.

This electricity from wind power is usually sent to nearby farms, homes and towns for usages. If it is not sent to towns and homes is it sent to a substation where the voltage is increased greatly to a few hundred thousand volts (called transmission-voltage power) to be sent through transmission lines to cities and factories.

SUB-TYPES

There are two types of wind machines used today: horizontal–axis wind turbines (HAWT) and vertical-axis wind turbines (VAWT). Most windmills are the horizontal-axis type and are also the only type used in Australia.

Horizontal-axis wind machines have blades like airplane propellers. See Figure 2.1. A typical horizontal wind mill stands about 90 metres tall and has three blades that span 60 metres across. Wind machines stand tall and wide to capture more wind. The more wind the more power.

Advantages of HAWT:

- High Efficiency
- Tall base allows access to stronger winds
- Variable blade pitch

Disadvantages of HAWT:

- Difficult to Transport
- Difficult to Install

Figure 2.1

Vertical–axis wind machines have blades that go from top to bottom and look like giant egg beaters. See Figure 2.2. The typical vertical wind machine stands 30 metres tall and 15 metres wide. Vertical-axis wind machines make up just five percent of the wind machines used today.

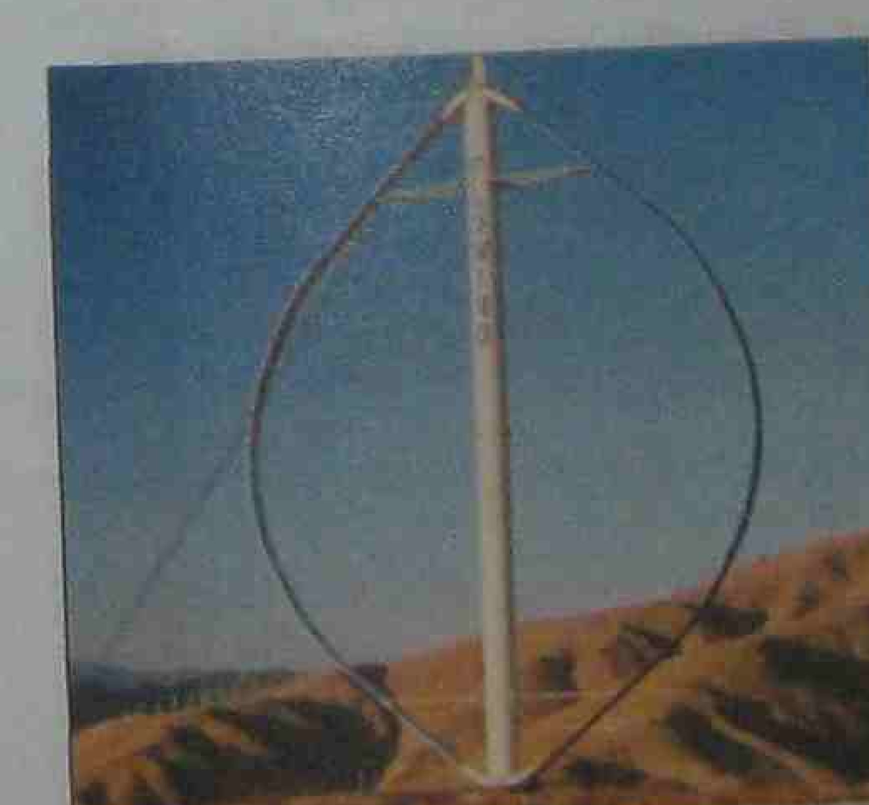


Figure 2.2

Advantages of VAWT:

- Easy to maintain
- Makes less noise
- Has lower start-up speeds

Disadvantages of VAWT:

- 50% less efficient to HAWT
- Doesn't produce as much energy as HAWT

FINANCIAL IMPORTANCE

The difference between wind and other sources of energy can be difficult to calculate because of the costs associated with wind developments. The majority of the costs related to wind developments are mainly capital costs. The operating and maintenance costs are low, and the production cost of each additional unit of wind power is very little compared to gas and coal developments which have large capital costs, as well as expensive operating and maintenance costs. The difference in costs makes it hard when trying to compare the cost of alternative energy sources.

Other than these differences, most of the data indicate that wind energy is one of the most cost efficient sources of renewable energy and that when the costs relates with pollution are factored in it is competitive with coal and gas power stations.

The average cost per windmill includes the following costs:

- the construction of the wind turbine
- the borrowed funds
- the return to investors (including the risk)
- the estimated annual production

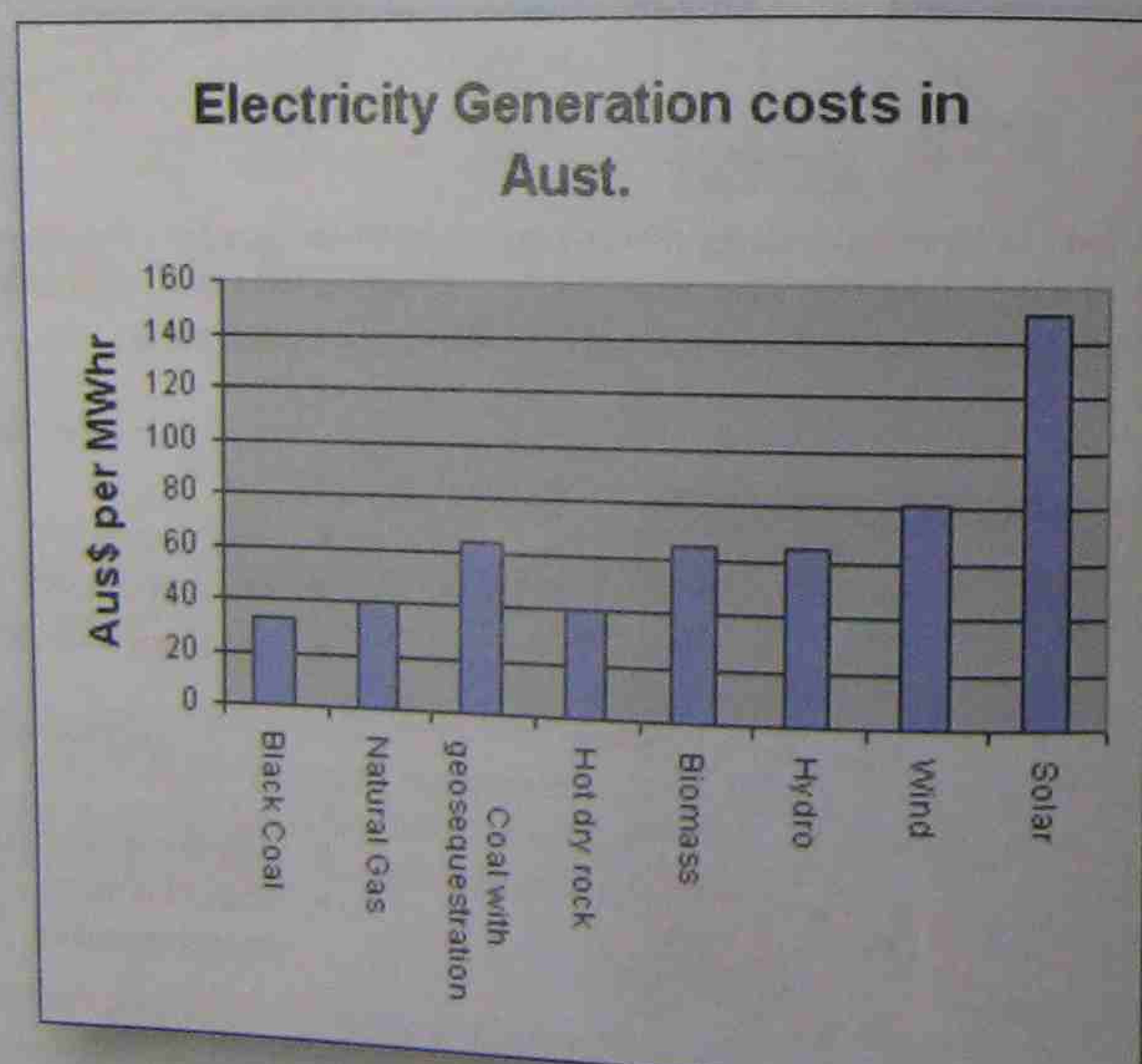


Figure 3.1

In Figure 3.1 above, we see that black coal and natural gas are cheaper alternate ways to produce electricity compared to wind, however this is because the fossil fuel industry is subsidised and environmental pollution costs are not paid by the fossil fuel industry.

INDUSTRIAL IMPORTANCE

The Howard Federal Government had a Mandatory Renewable Energy Target (MRET) which aimed at Australia having about 2% of its electricity generated by renewable means. In the UK Sustainable Development Commission has announced that the UK is aiming at 10% by 2010 and 20% by 2020 and that there "are no major technical barriers to meeting these targets". Scientists have warned that we must reduce the world's greenhouse gas production rates by 60%. The Rudd government promised twenty percent renewable energy by 2020 in the 2007 election campaign.

At the moment there are five operating wind farms In Victoria, five in Western Australia, four in NSW, and in South Australia there are nine which also produces almost 60% of Australia's wind power.

Both the NSW and Victorian governments have legislated much larger mandatory renewable energy targets than the Howard government did. These will make electricity retailers buy significant percentages of renewable energy. For NSW to fulfil its commitment to renewable energy it will have to buy wind generated electricity from SA.

The carbon dioxide reduction from one typical (2MW) wind turbine in Australia can be expected to produce over 6000 megawatt hours of electricity each year. If this replaces coal-fired power, then the CO_2 released to the atmosphere will be reduced by 6000 tonnes each year, if it replaces oil or gas-fired power, CO_2 released each year is reduced by about 3000 tonnes.

There are also a lot of issues involved with wind generation, the following demonstrate these issues:

- Noisy turbines
- Television reception
- Erosion
- Bird deaths
- Wind is intermittent
- Turbine shutdown after 43°C

These are just a few problems that come up when debating about the use of wind generators. Figure 4.1 below is a graph of wind generator installations all over the world, and implies that the issues above are not interrupting the future of wind generation.

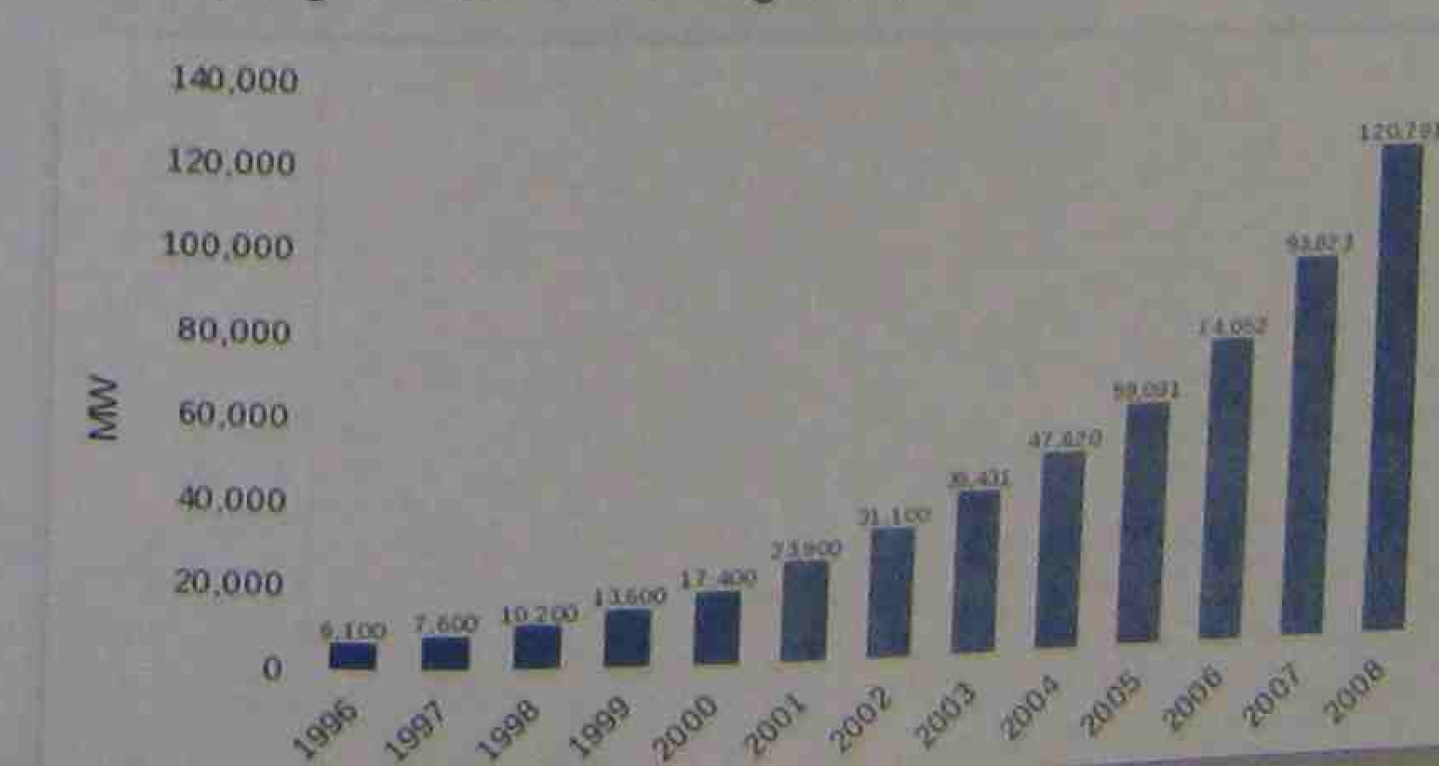


Figure 4. 1

CONCLUSION

Climate change is happening and must be minimised. Australia and the world must move away fro fossil fuels. Wind, at present, is the only economically competitive form of sustainable energy. However, just because wind is a good form of energy and is renewable, doesn't mean we can turn up our usage of electricity. We still need to subsidise our Co^2 emissions. For climate change, wind not the answer, but it is part of the answer.

REFERENCES

http://en.wikipedia.org/wiki/Wind_power_in_Australia

http://en.wikipedia.org/wiki/Wind_power

http://www1.eere.energy.gov/windandhydro/wind_ani[illegible].html

http://www.newsdial.com/science/energy/wind-powe[illegible].html

http://www.green-trust.org/wind.htm

http://www1.eere.energy.gov/windandhydro/wind_how[illegible]

http://en.wikipedia.org/wiki/Vertical_axis_wind_turbine

http://en.wikipedia.org/wiki/Nacelle

http://en.wikipedia.org/wiki/Renewable_energy

http://en.wikipedia.org/wiki/Vertical_axis_wind_turbine

http://www.geocities.com/daveclarkecb/Australia/WindPower.html

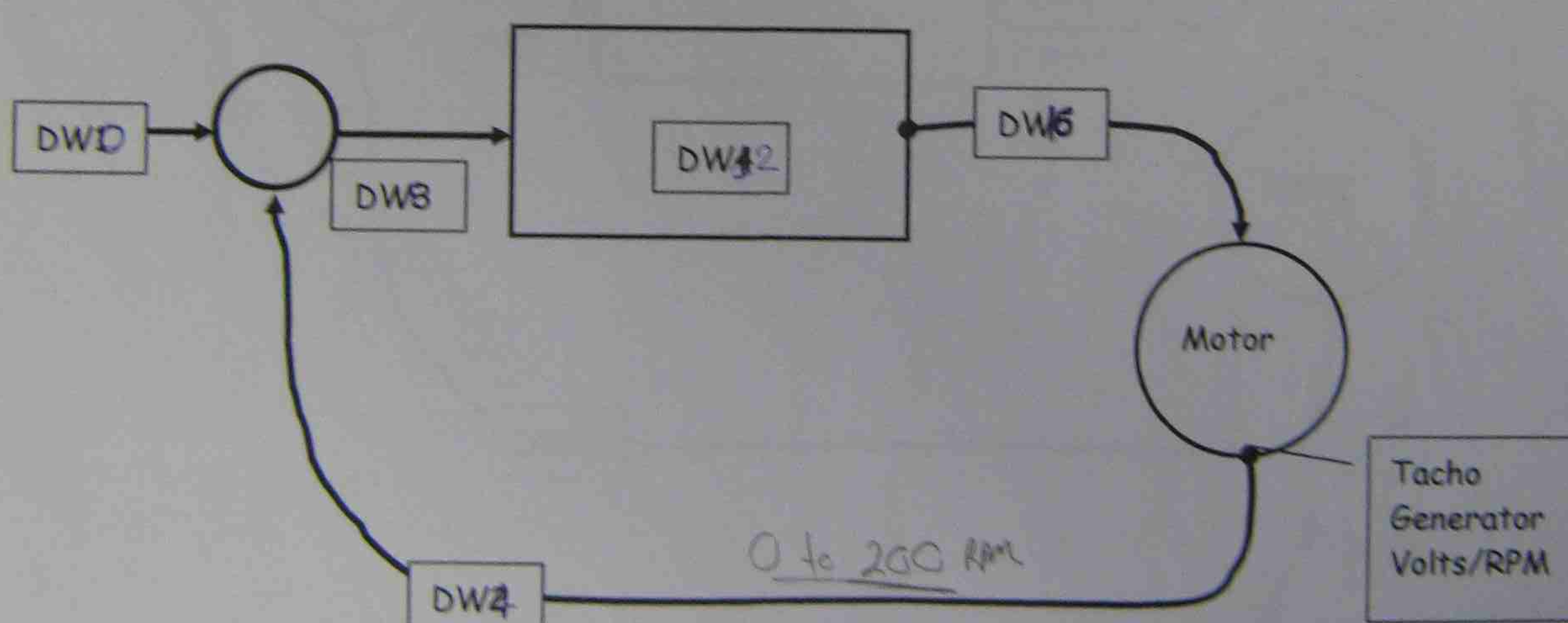
Aim :-

- To design a program using analog inputs and outputs that will control the speed of a DC Motor using Proportional Control.
- Observe and record the operation of the system.
- Add some Integral control to improve system performance.
- Add a control system for operator interfacing.

Procedure A - Proportional Control (10 Marks)
Use Data Block 10 for storing all settings.

- The speed of the motor is set by the thumbwheel switch after a one shot from an input. The setpoint is to be stored in data word 1 (Setpoint = DW0)
- The feedback is taken from a tacho generator connected to an analog input.(Feedback = DW4)
- The feedback is compared to the setpoint and the error is produced.(Error = DW8)
- The gain of the system is set by the same thumbwheel switch after a one shot. (Gain = DW12) (use I 2.1)
- The output of the controller is sent to the servo drive from an analog output .(Output = DW16)

Motor Drive Parameters	PLC Address	Set by One Shot
Motor Speed Set point	DW1	I 126.0
Proportional Gain	DW4	I 126.1



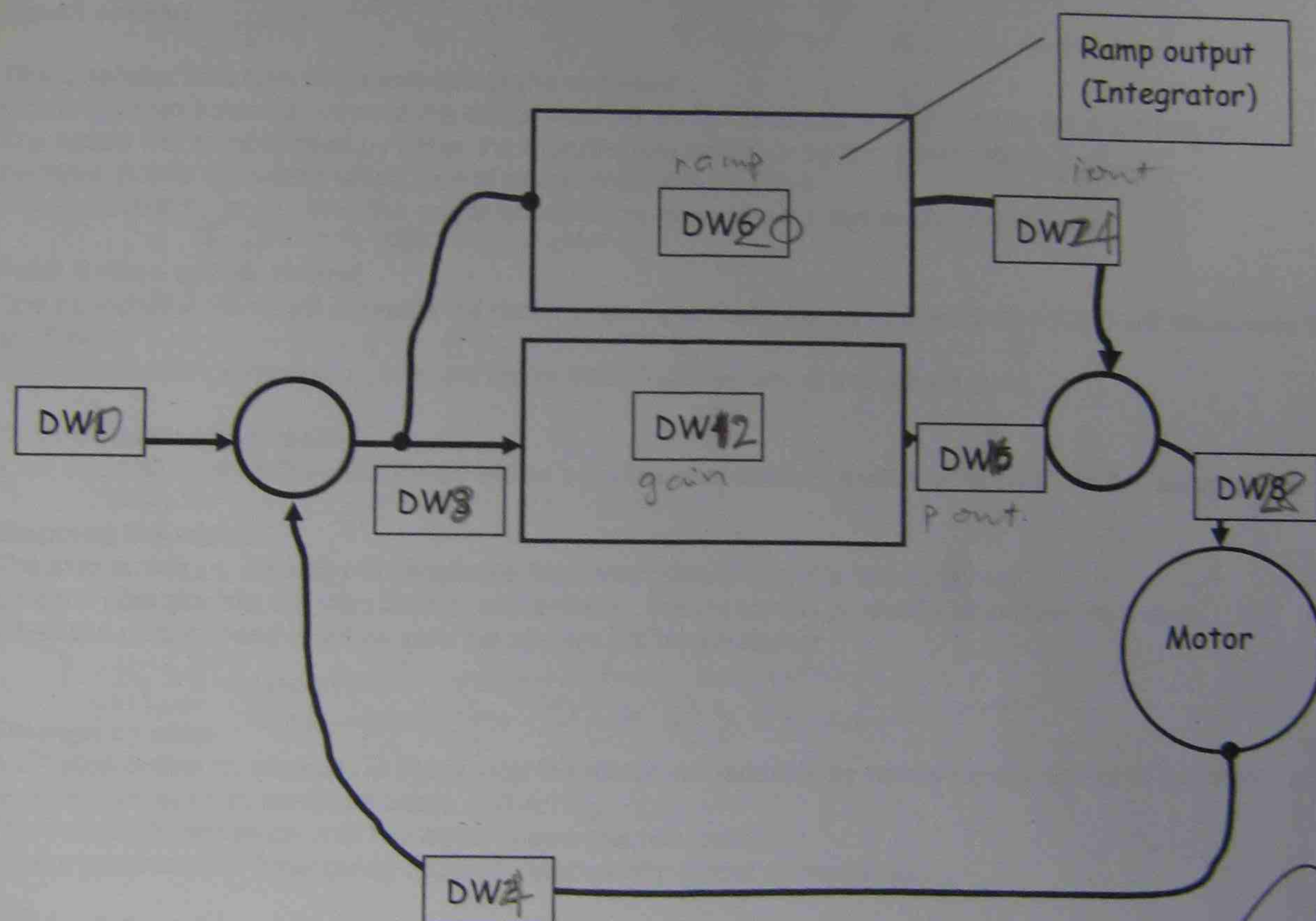
- Demonstrate your program's function and record the offset when the setpoint is set to 100 RPM with a range of gains.

Gain	Offset
2	64 – 36
4	78 – 22
6	84 – 16
8	88 – 12
10	90 – 10
12	94 – 6

Procedure B - Proportional + Integral action (20 Marks)

Aim - To add integral action to the control system and remove the offset.

Design a system that will improve the performance of your control system by adding some form of Integral control. The Integral action will remove the offset , we will use a scada system to test your project for changed in load and see if the offset can be removed.
The Integral action rate should be adjustable via the thumbwheel switch and a one shot.
Marks will be awarded for accuracy , simplicity and reliability of your system.



error = sp − feedbak

error >0 → sp

error <0 → sp fb

~~set limit for ramping~~

Part C - Complete control system (20 Marks)

Aim - to add operator control to the stopping and starting of the motor by gradual acceleration and deceleration and an emergency stop and braking procedure.
The operator will have the option to control the speed of the motor via a thumbwheel switch or two pushbuttons.

Starting and stopping the motor
The motor is be started and stopped by a control station.
The start button (I2.2) is normally open and the stop button (I2.3) is normally closed.

Speed control

The Operator has two ways to control the setpoint
When the start button is pressed the motor start will be enabled and a "run" light at Q4.5 will turn on.
The speed will be controlled by either the thumbwheel switch or by two pushbuttons.
An input (I2.6) will select which type of speed control is be used ,
If the input(I2.6) is on , then the motor speed is controlled by the two pushbuttons .

Push Button speed control
One pushbutton (I2.4) will increase the motor speed gradually the other pushbutton (I2.5) will decelerate the motor gradually.
If the pushbutton is released , then the motor speed will remain at that speed level.

Thumbwheel speed control
If the input (I2.6) is off then the speed will be set via a thumbwheel switch as in part 1 of this assignment.

Stopping the motor
The stop button (n, closed) will decelerate the motor slowly until the speed is zero.
So if the operator hits the stop button momentarily , the motor will gradually slow down to a stop.
When the motor speed reaches zero the system will be shutdown.

Emergency stop.
A 2nd stop button (n, closed) (I2.7) will stop the motor immediately by removing any power to the motor and applying power to an electrical brake at Q 4.7.
The brake will remain on until the motor speed reaches zero.
In your program this Emergency Stop will remove any output to the motor.

Extra – Adjustable acceleration. (5 marks)
Make the acceleration and deceleration rates individually adjustable via the thumbwheel switch and a one shot.

Extra Part 2 –Working with S7 Lite (5 marks)
Complete the assignment using an S7 PLC and the S7 lite software.

Notes
Document your program thoroughly explain each point clearly.
Do your own work.
Marks are awarded for ingenious designs !
Marks are deducted for poor program comments and explanations.
Marks are awarded for the accuracy of your control system and your ability to demonstrate it.

Symbol table

Status	Symbol	Address	Data Type	Comment
	CYCL_EXC	OB 1	OB 1	Cycle Execution
	SCALE	FC 105	FC 105	Scaling Values
	INTEGRAL RATE	I 126.2	BOOL	
	I rate oneshot 1	M 0.5	BOOL	
	I rate oneshot 2	M 0.6	BOOL	
	Offset exists	M 0.7	BOOL	
	Integral Rate BCD	MW 14	WORD	
	UNSCALE	FC 106	FC 106	Unscaling Values
	Gain DI	MD 120	DWORD	
	Scale/Unscale Bipolar	M 0.0	BOOL	
	Scale return value	MW 10	WORD	
	GAIN	I 126.1	BOOL	
	Analog input	PIW 752	WORD	
	SETPOINT	I 126.0	BOOL	
	Thumbwheel input	IW 124	WORD	
	SP oneshot 1	M 0.1	BOOL	
	SP oneshot 2	M 0.2	BOOL	
	SP BCD	MD 104	DWORD	
	DATA STORING	DB 10	DB 10	
	G oneshot 1	M 0.3	BOOL	
	G oneshot 2	M 0.4	BOOL	
	SP DI	MD 108	DWORD	
	Gain BCD	MD 112	DWORD	
	Gain masked	MD 116	DWORD	
	Unscale return value	MW 12	WORD	
	Integral Timing Rate	T 1	TIMER	
	Analog output	PQW 752	WORD	
	1 cycle Reset	M 1.0	BOOL	
	DECELERATE	I 124.6	BOOL	
	START	I 126.6	BOOL	
	MANUAL	I 126.7	BOOL	
	Master Control	M 1.1	BOOL	
	ACCELERATE	I 124.7	BOOL	
	E-STOP	I 126.4	BOOL	
	STOP	I 126.5	BOOL	
	Light	Q 124.0	BOOL	
	S-Stop Flag	M 1.5	BOOL	
	Stop @ Zero	M 1.2	BOOL	
	Manual Speed Rate BCD	MD 124	DWORD	
	MSR masked	MD 128	DWORD	
	MSR DI	MD 132	DWORD	
	MSR Real	MD 136	DWORD	
	Setpoint BCD masked	MD 140	DWORD	
	Manual Time Rate	M 1.3	BOOL	
	Manual Speed Rate	T 4	TIMER	
	Electrical Brake	Q 124.1	BOOL	

Status	Symbol	Address	Data Type	Comment
	E-Stop Flag	M 1.6	BOOL	

OB1:CYCL_EXC

Cycle Execution

```
Name:                      Time stamp Code:       25/06/09
Author:                               Interface:  20/01/04
Family:                    Lengths    Block:      01024
Version:       0.0                    Code:       00796
Code version: 2                       Data:       00028
```

Block: OB1 "Main Program Sweep (Cycle)"

FILE NAME: motor1.k7p

KNOWN FACTS & ISSUES:

- When switching from too high manual speed (comparing to thumbwheel control setpoint) to PI control, P+I action will decrease it fast to correct error by large P out, I out and that might be too large that produces 0 at analog output to cause the system misunderstanding shutdown.

- SCALE and UNSCALE are isolated from master control flag for precision and continously monitoring regardless of power control unless PLC loses its own power supply.

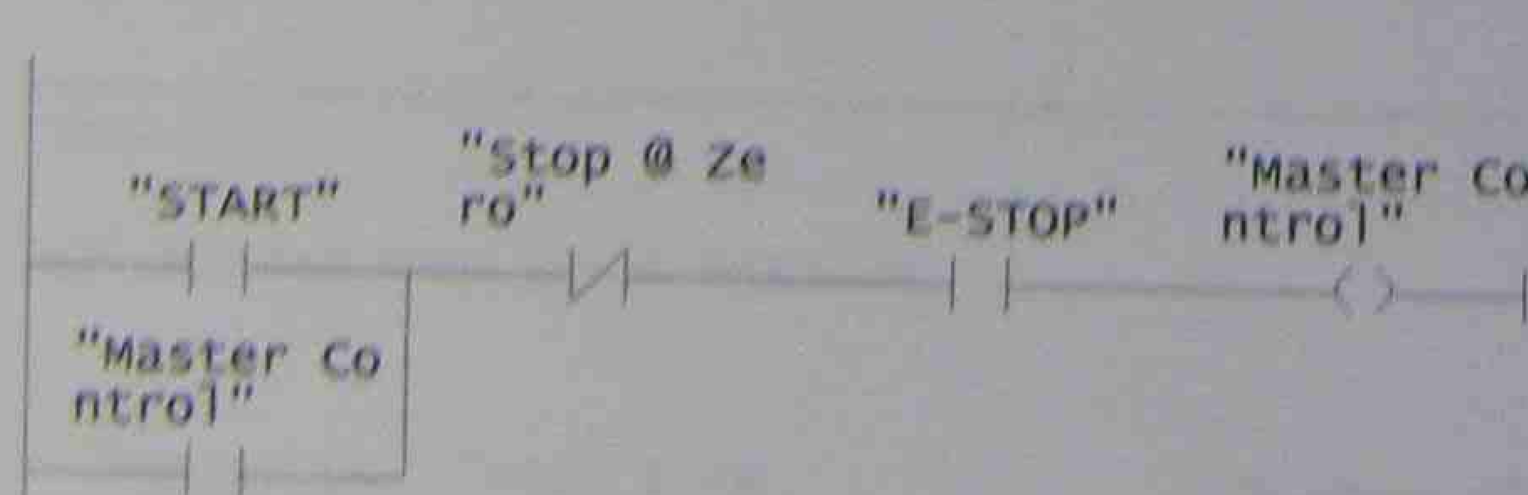
- PI and manual control operate simutanously without interfering each other but outputs are separated by manual switch. Manual Switch is used to select which output will be produced at final output to be unscaled. In other word, specifications can be pre-set before switching modes.

- Stop buttons must be wired/connected/switched on before starting.

Address	Declaration	Name	Type	Start value	Comment
0.0	temp	OB1_EV_CLASS	BYTE		Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
1.0	temp	OB1_SCAN_1	BYTE		1 (Cold restart scan 1 of OB 1), 3 (Scan 2-n of OB 1)
2.0	temp	OB1_PRIORITY	BYTE		1 (Priority of 1 is lowest)
3.0	temp	OB1_OB_NUMBR	BYTE		1 (Organization block 1, OB1)
4.0	temp	OB1_RESERVED_1	BYTE		Reserved for system
5.0	temp	OB1_RESERVED_2	BYTE		Reserved for system
6.0	temp	OB1_PREV_CYCLE	INT		Cycle time of previous OB1 scan (milliseconds)
8.0	temp	OB1_MIN_CYCLE	INT		Minimum cycle time of OB1 (milliseconds)
10.0	temp	OB1_MAX_CYCLE	INT		Maximum cycle time of OB1 (milliseconds)
12.0	temp	OB1_DATE_TIME	DATE_AND_TIME		Date and time OB1 started

Network: 1

master control



Network: 2

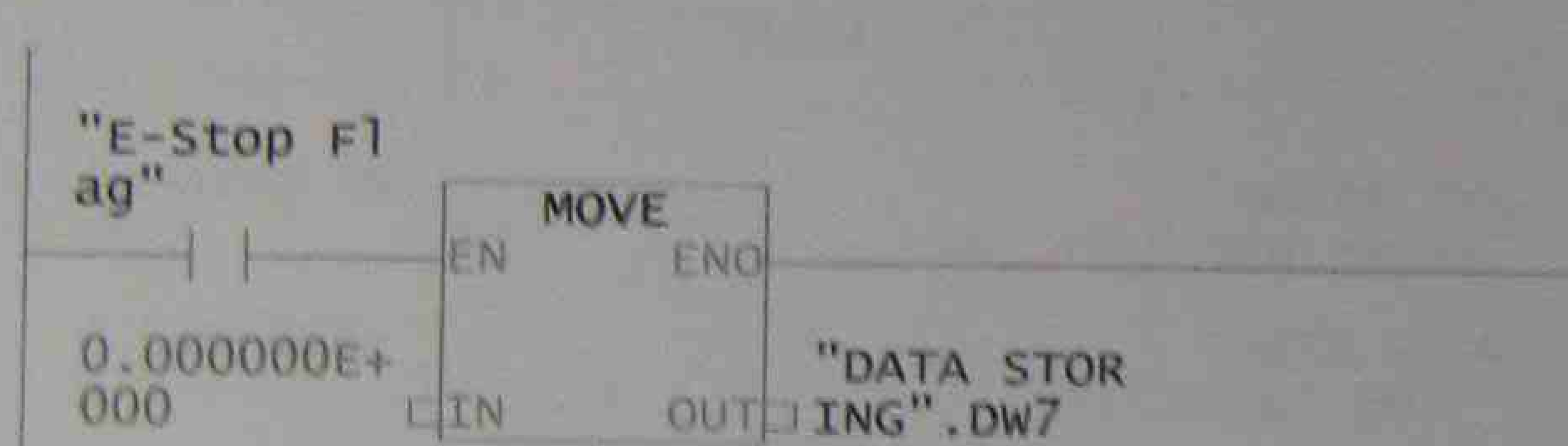
soft stop set flag

Network: 3

emergency stop set flag

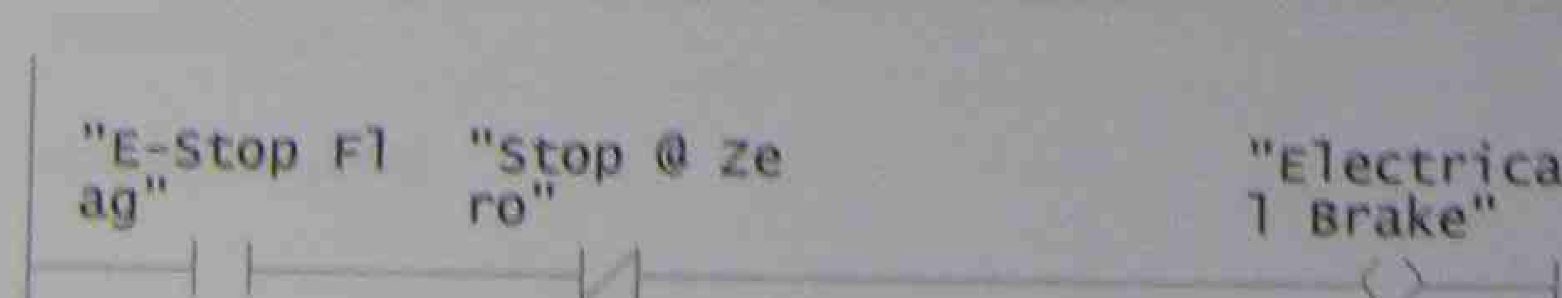
Network: 4

e-stop moves 0.0 to internal output to send out 0 voltage at analog output.
- depends on type of motor that it will stop immediately or free spinning after cutting out the source.



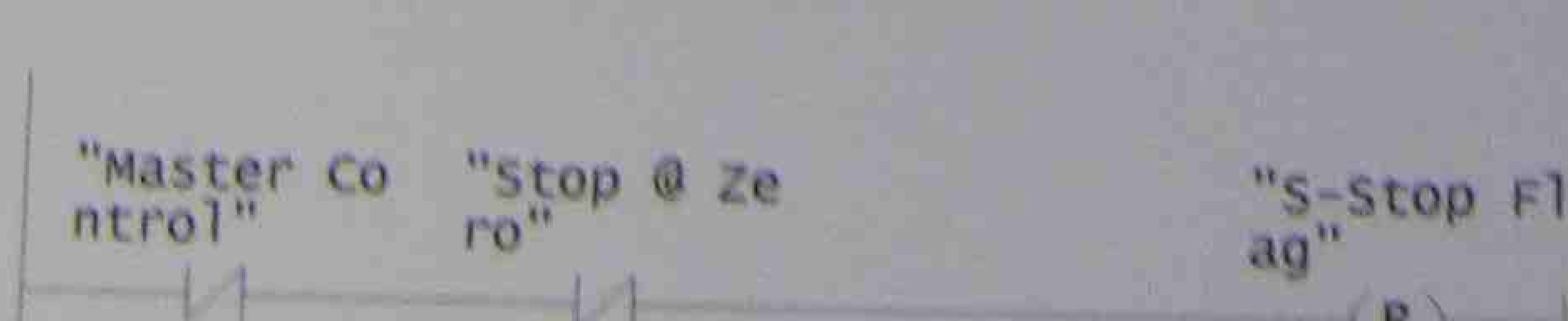
Network: 5

electric brake on when e-stop set and off when feedback equal 0 - in this practical case, the motor stops immediately at analog output of 0 so it may not be easy to distinguish the moment of braking the free spinning motor



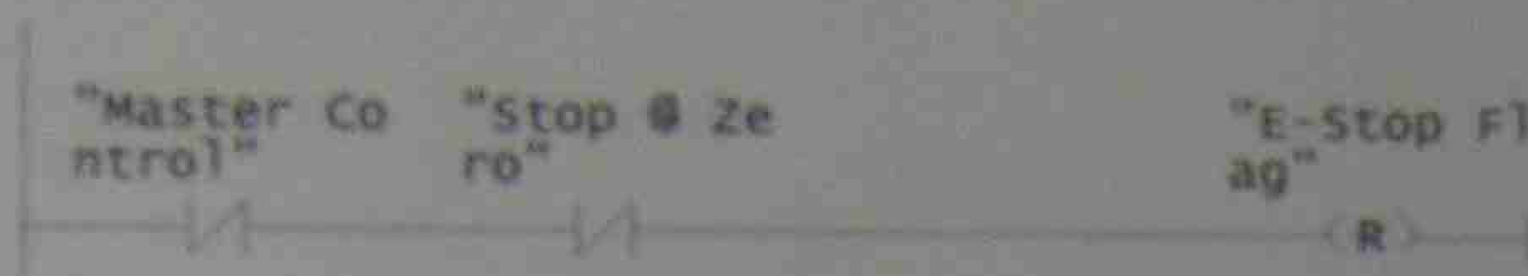
Network: 6

soft stop reset when system shutdown



Network: 7

e-stop reset when system shutdown and speed at zero

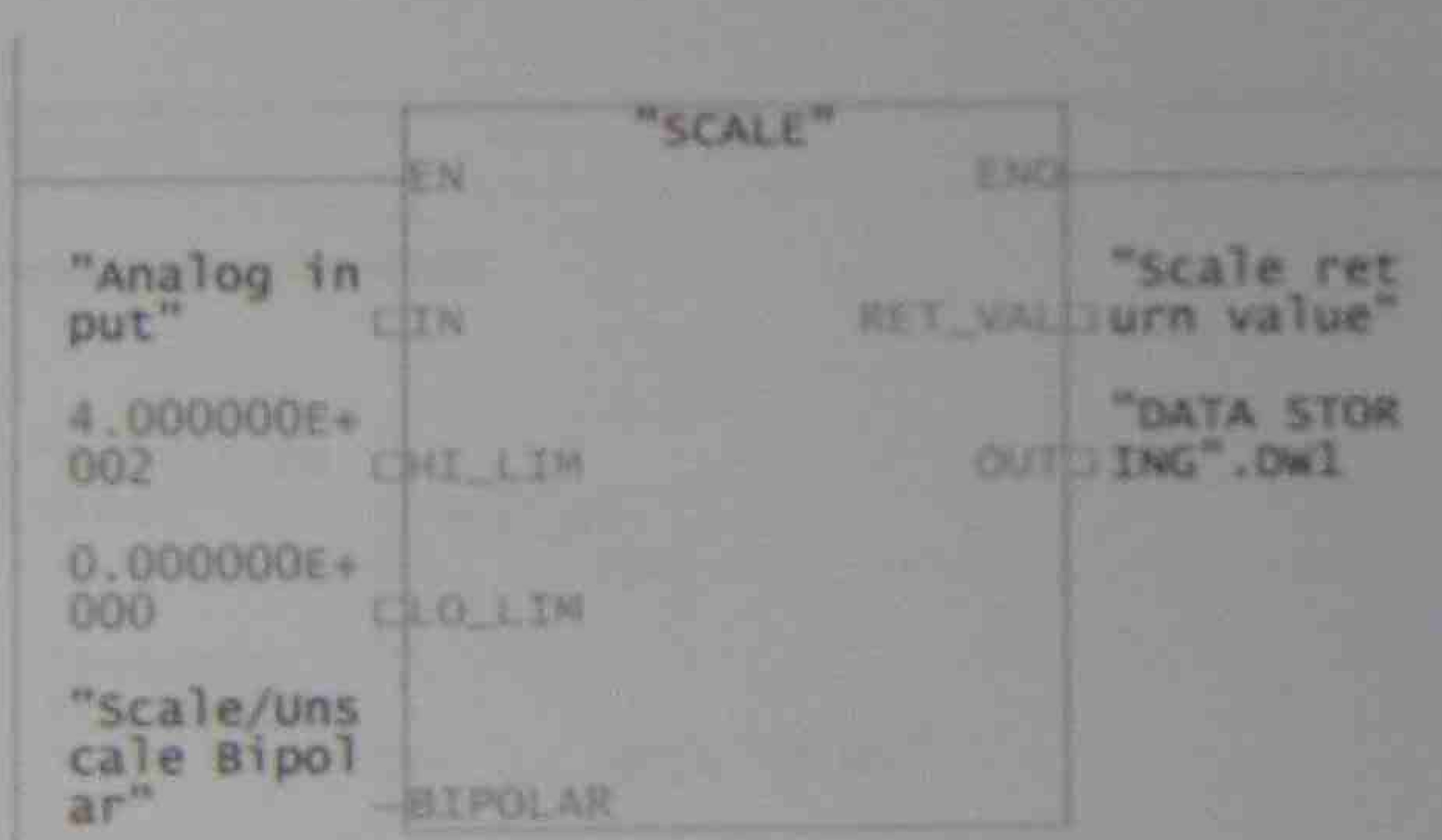


Network: 8

run light

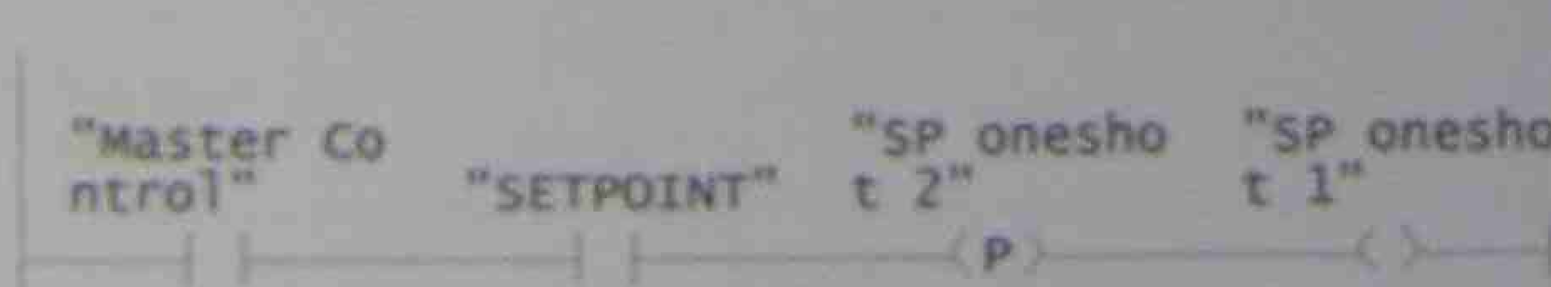
Network: 9

feedback stored



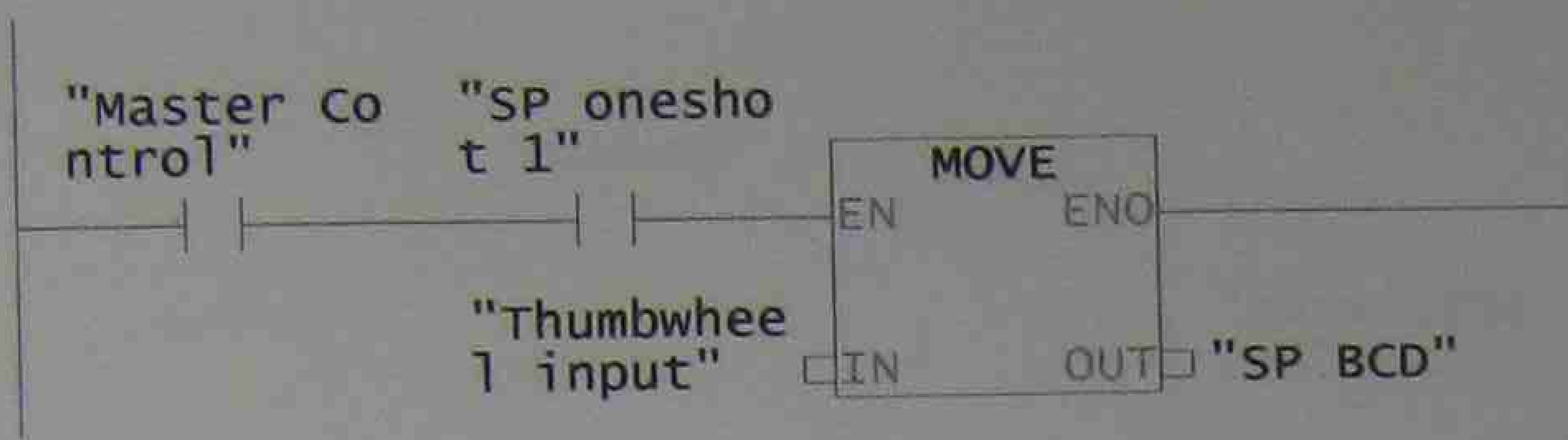
Network: 10

oneshot setpoint



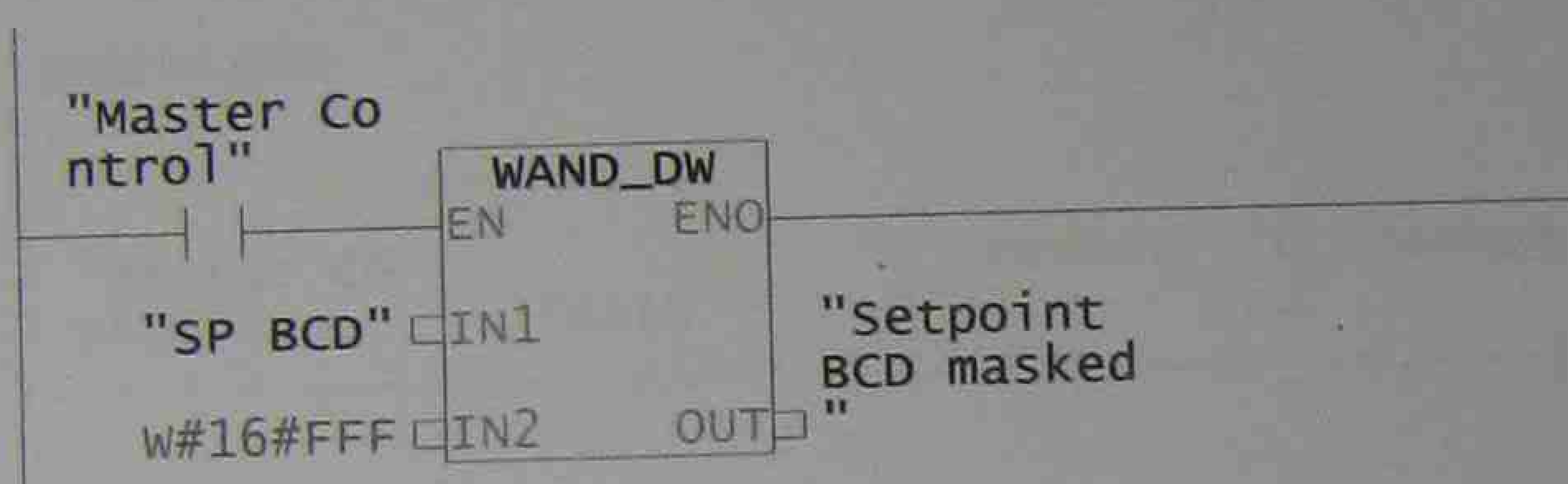
Network: 11

taking setpoint thumbwheel input



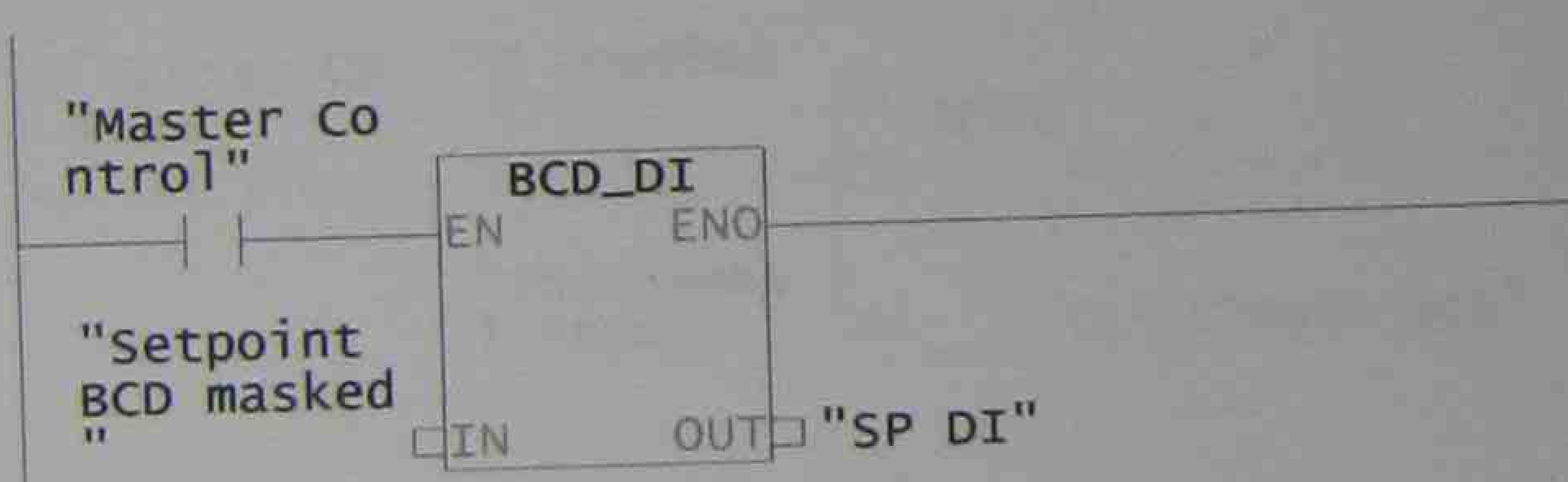
Network: 12

setpoint masking



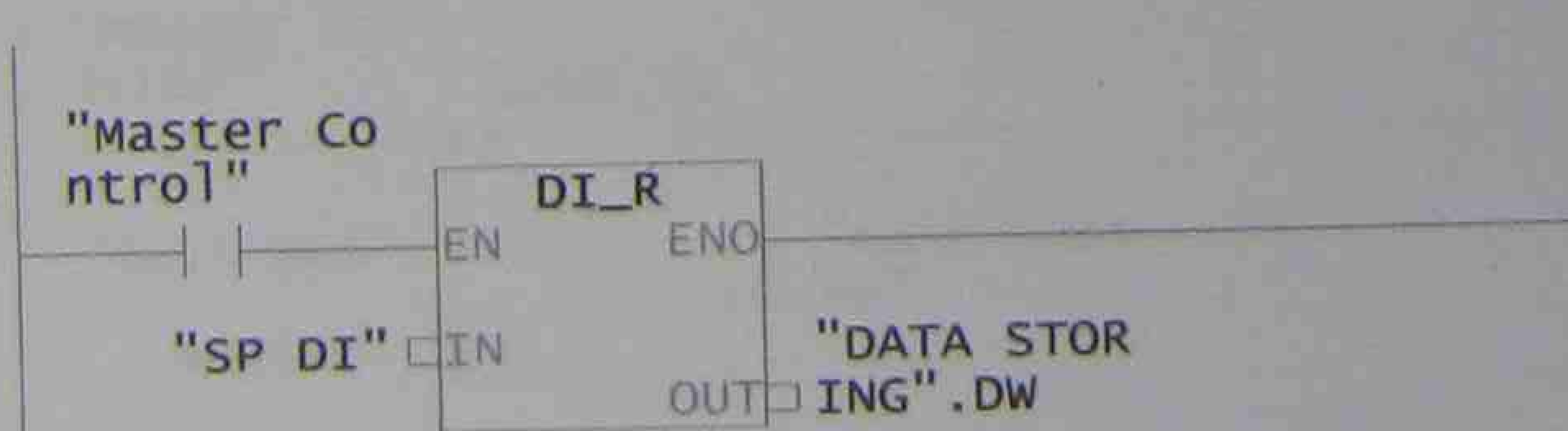
Network: 13

setpoint double-integer



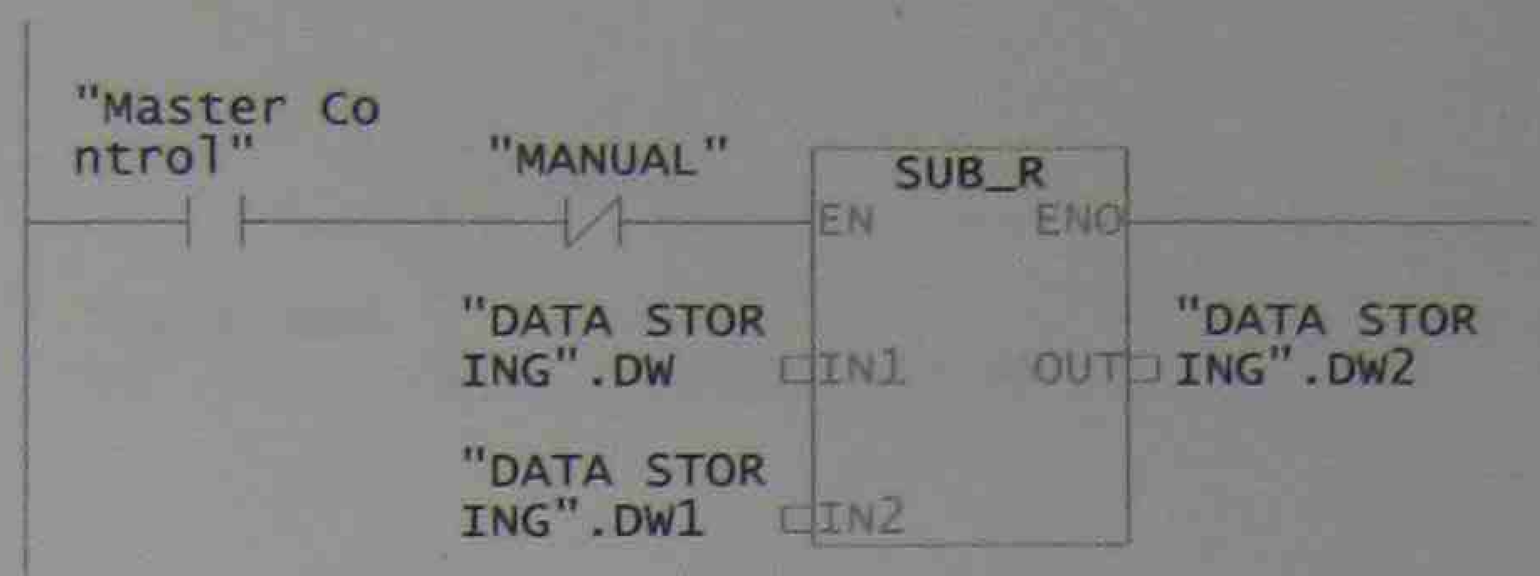
Network: 14

setpoint real stored



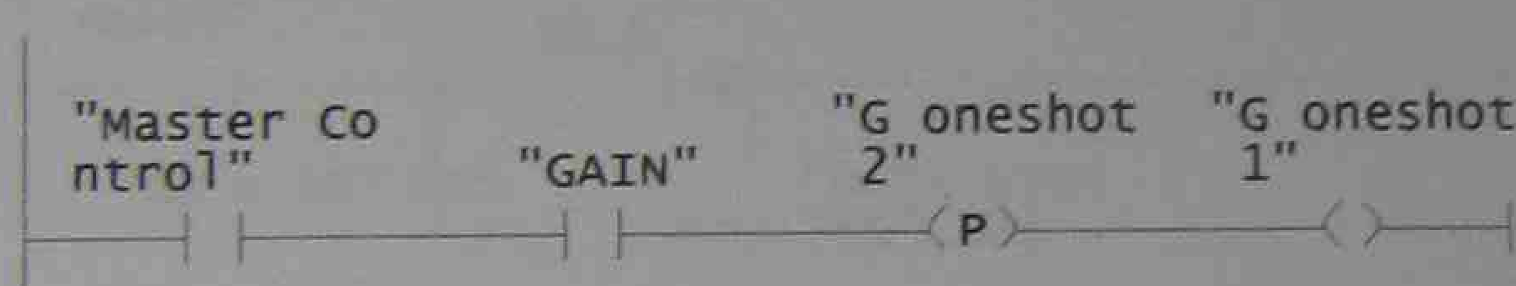
Network: 15

error



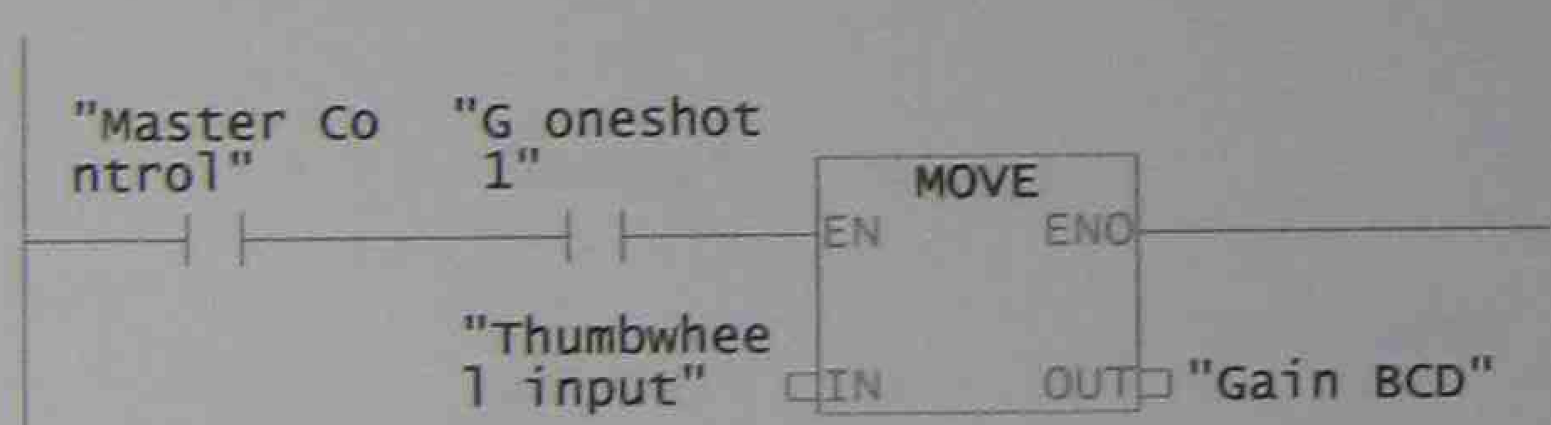
Network: 16

oneshot gain



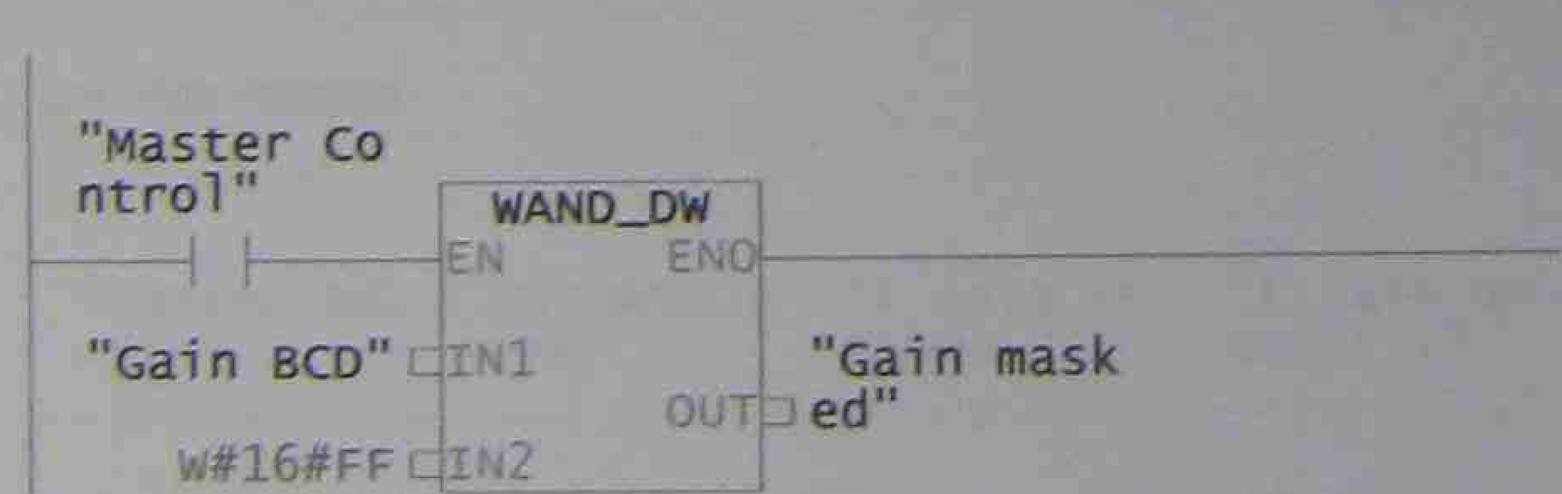
Network: 17

taking gain thumbwheel input



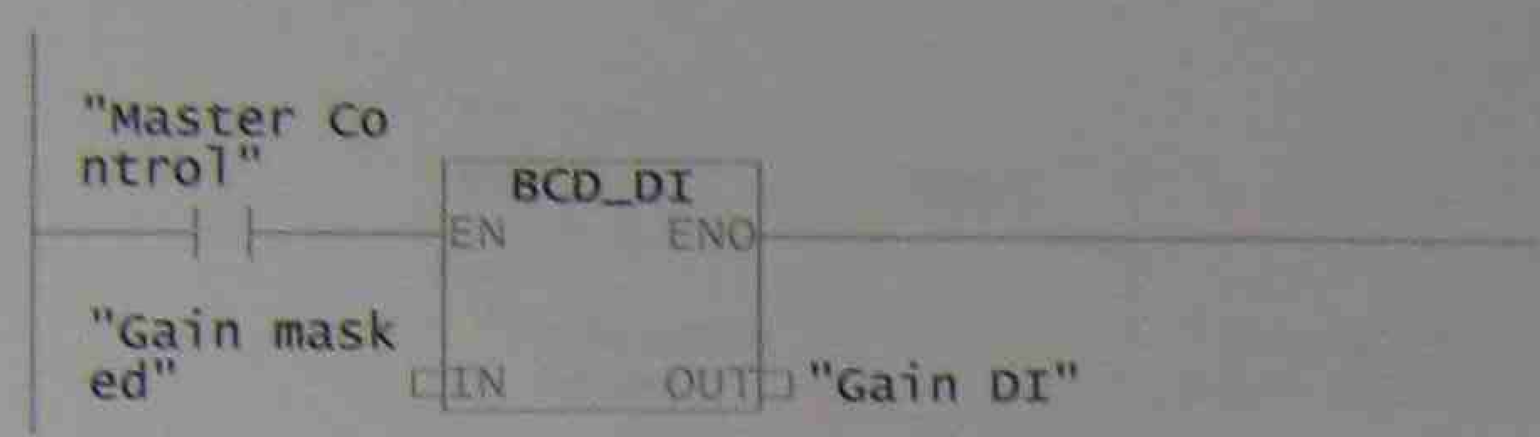
Network: 18

gain masking



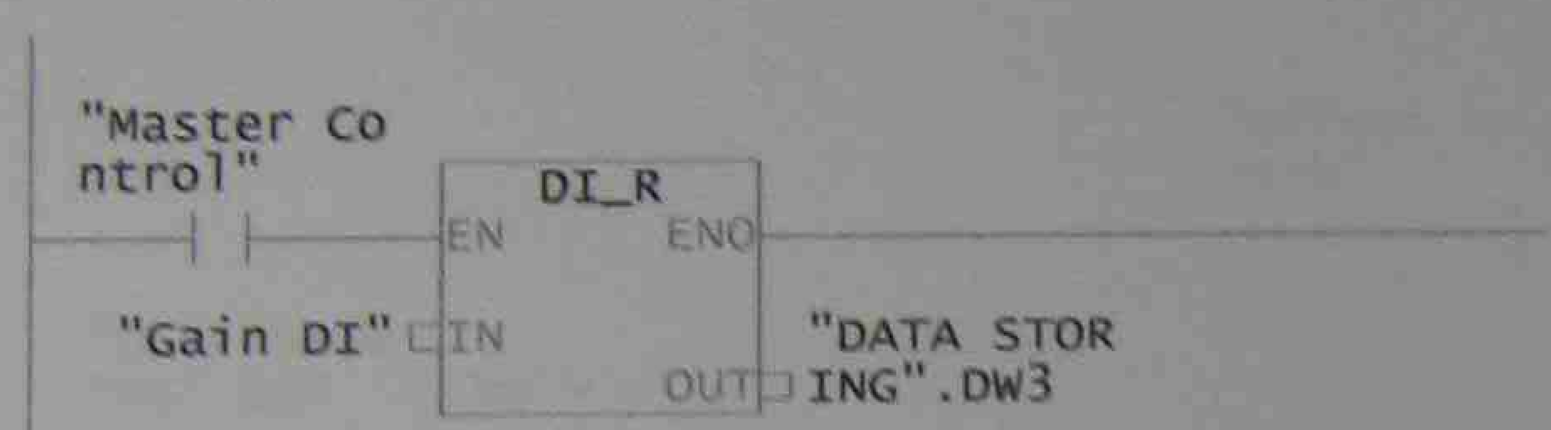
Network: 19

gain BCD - DI



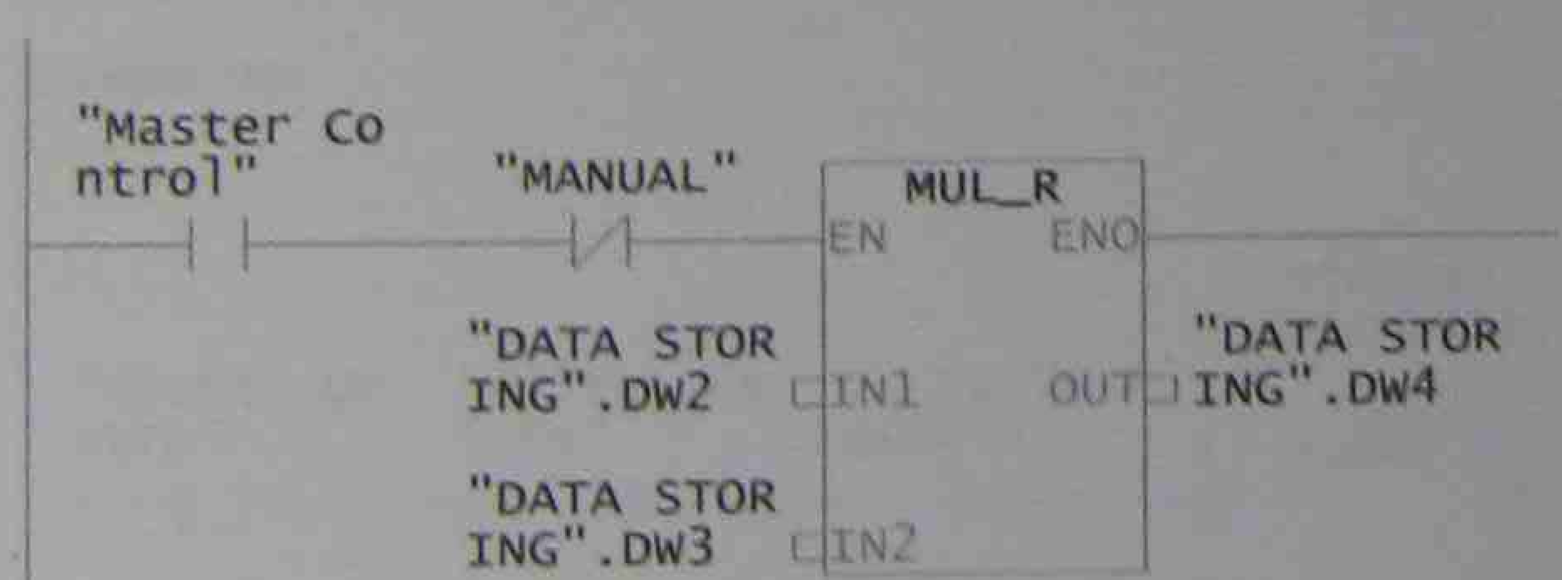
Network: 20

gain real stored



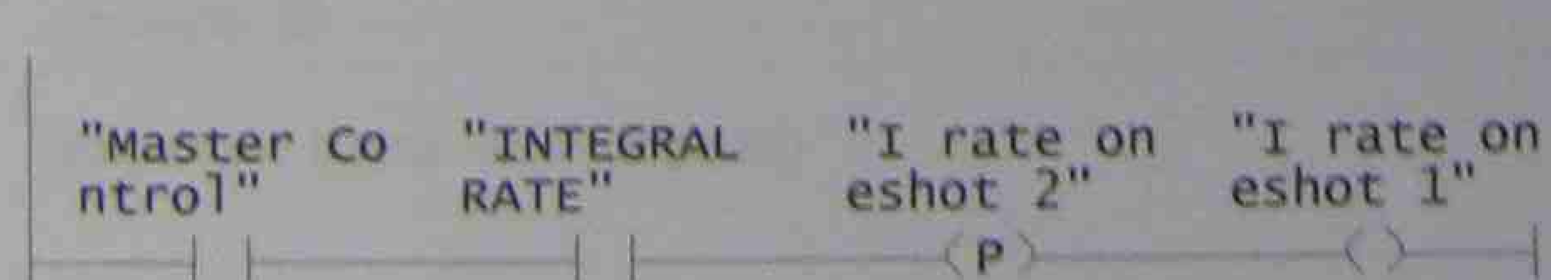
Network: 21

proportional output



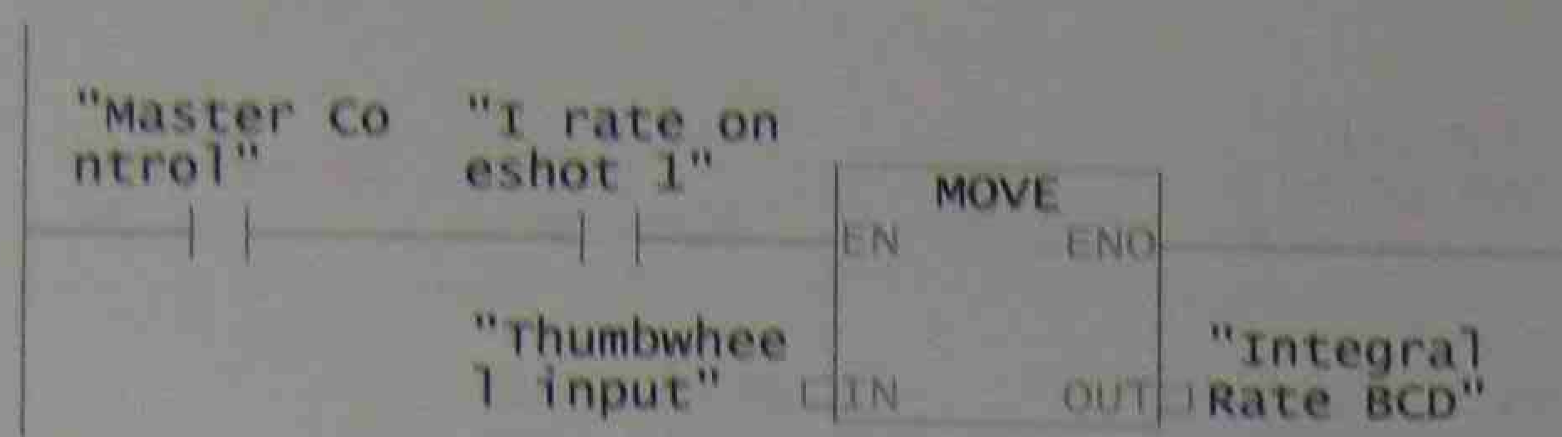
Network: 22

oneshot integral



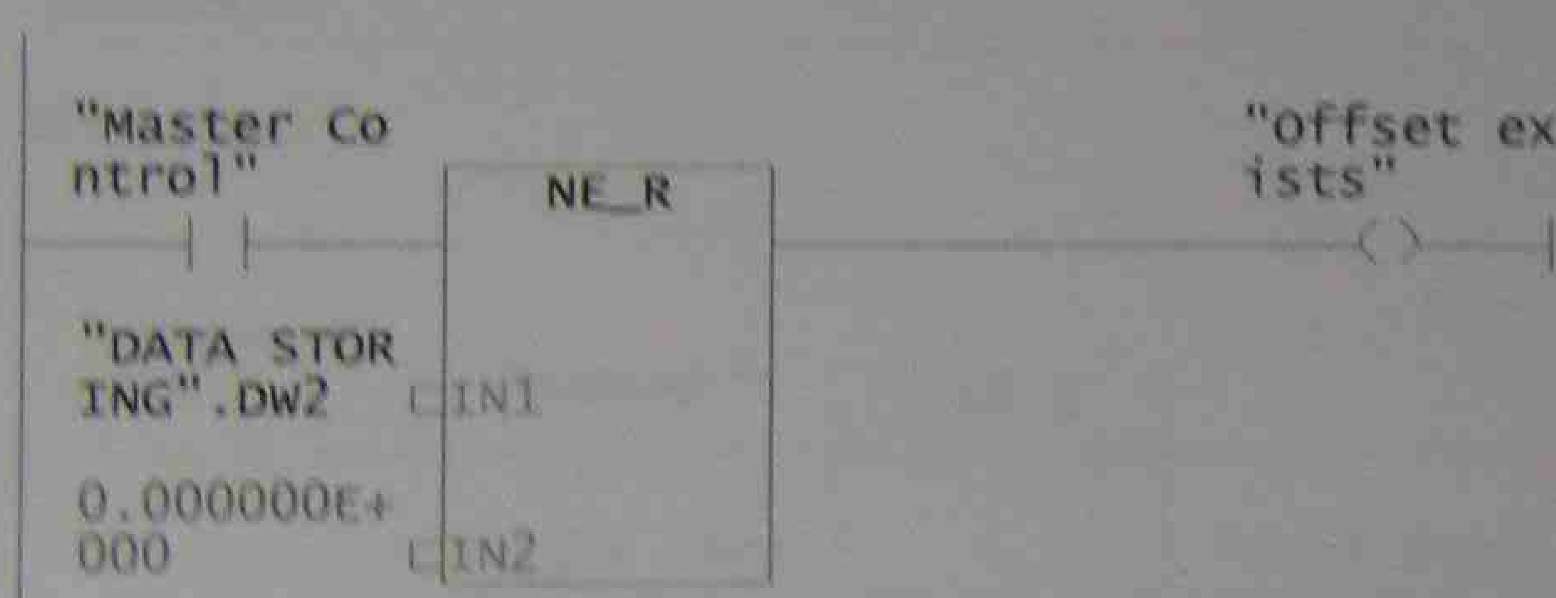
Network: 23

taking integral rate thumbwheel input



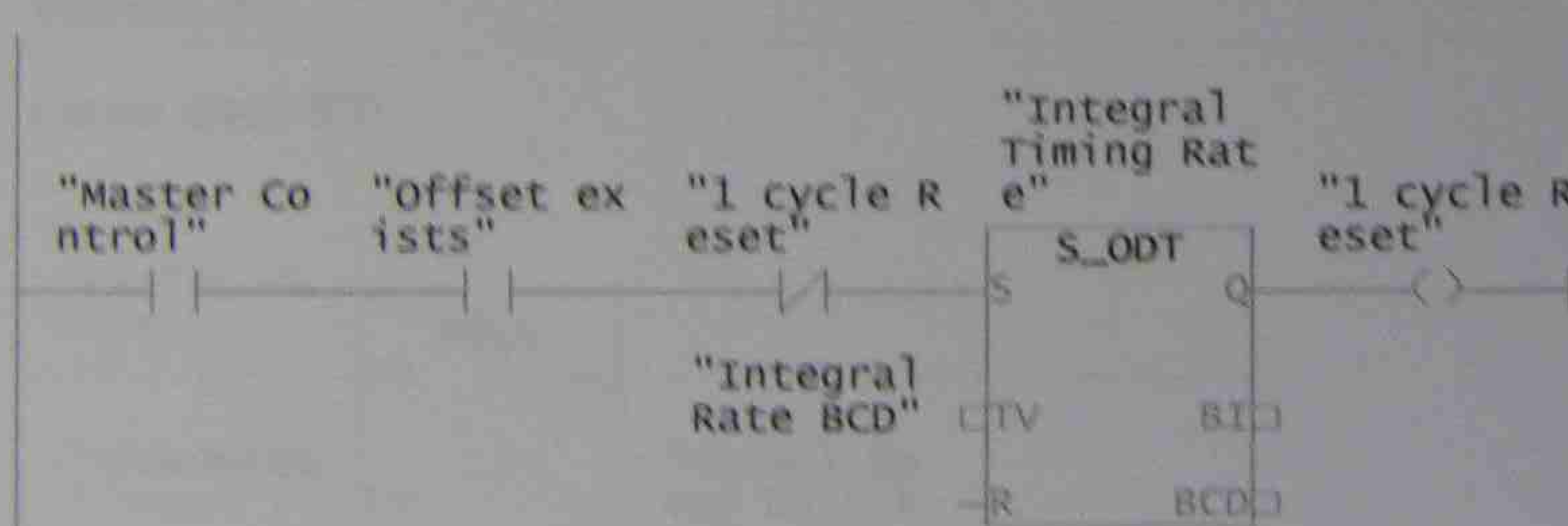
Network: 24

offset (error not 0)



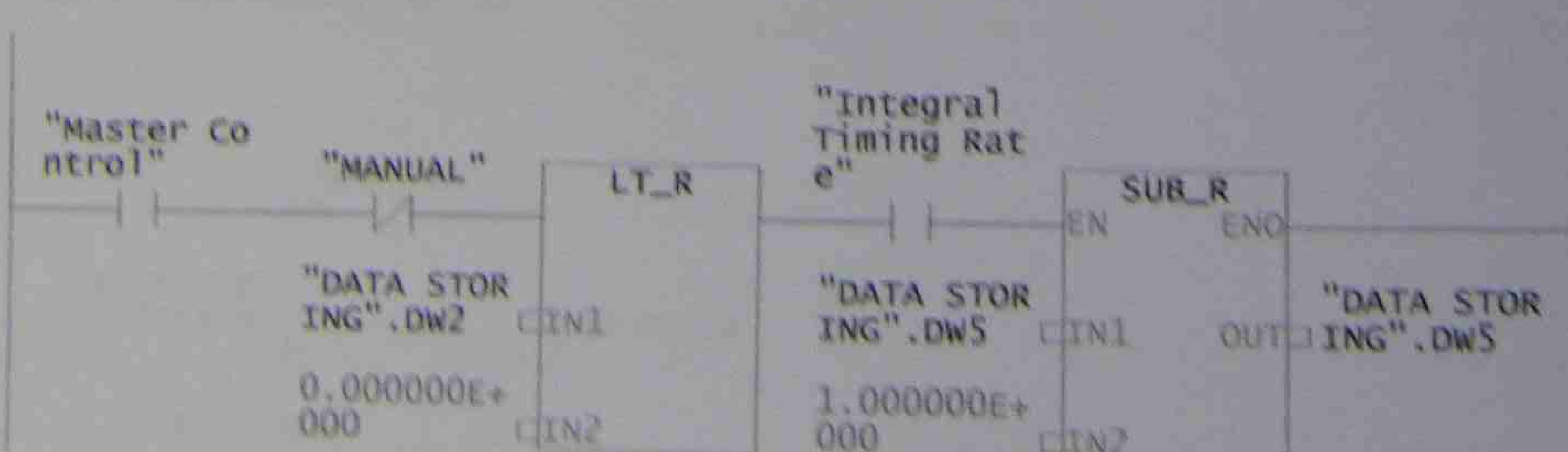
Network: 25

integral rate



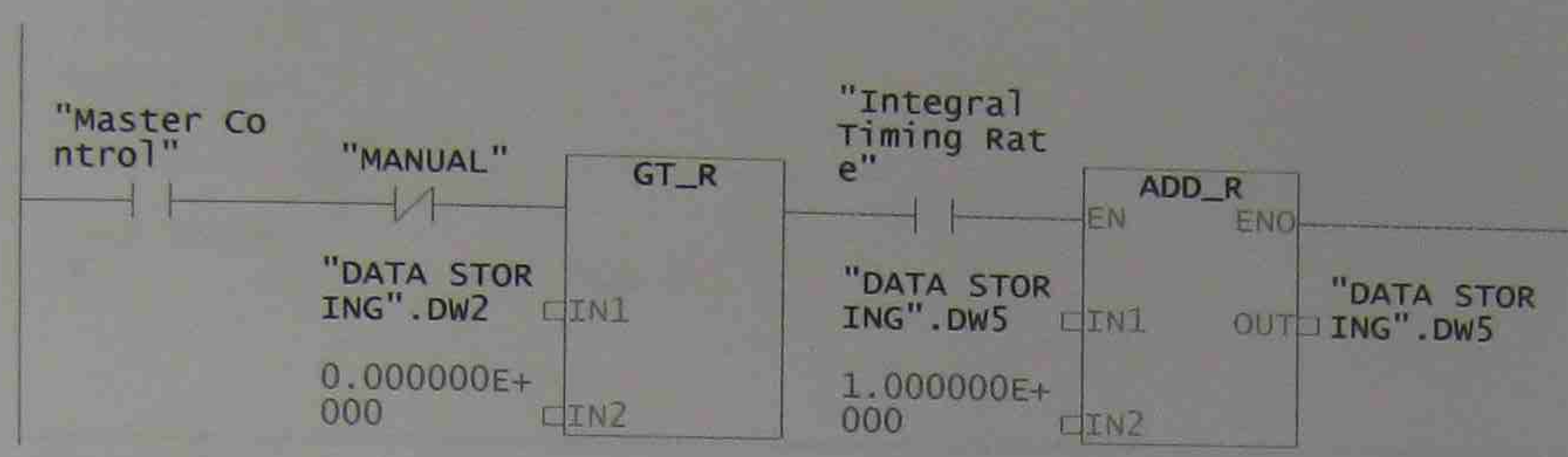
Network: 26

ramp down (feedback > setpoint)



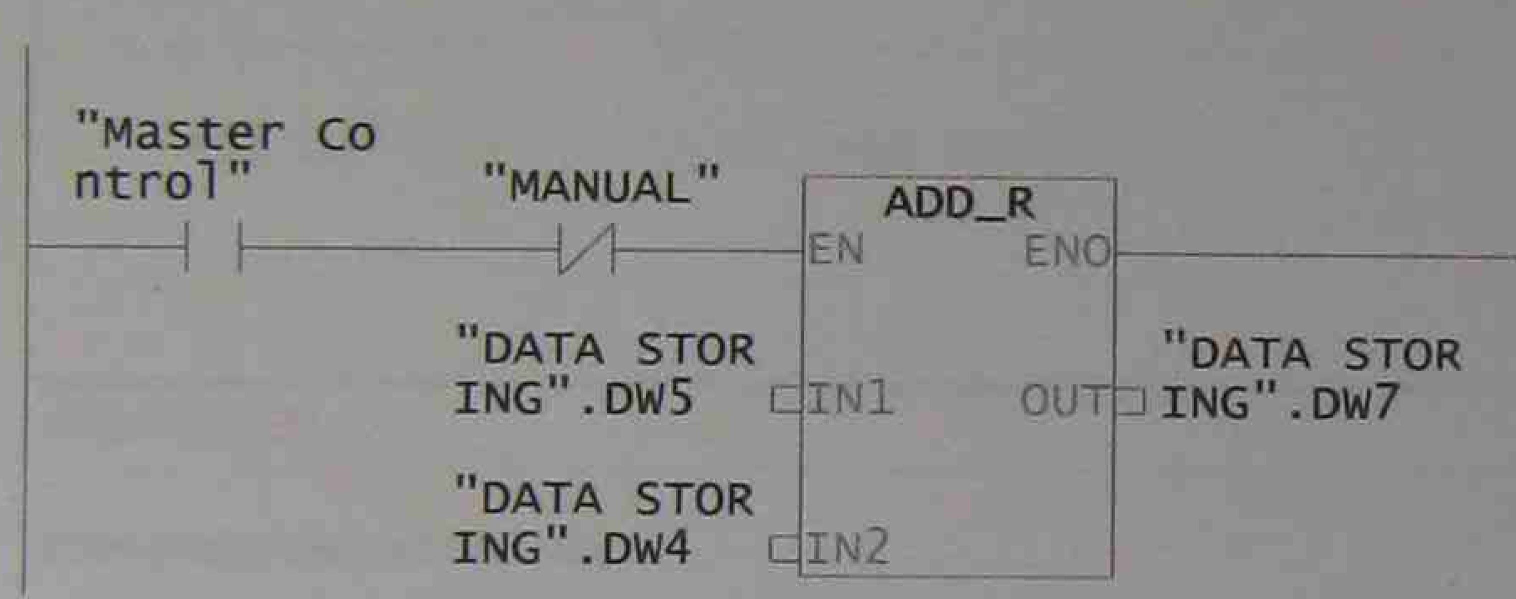
Network: 27

ramp up (feedback < setpoint)



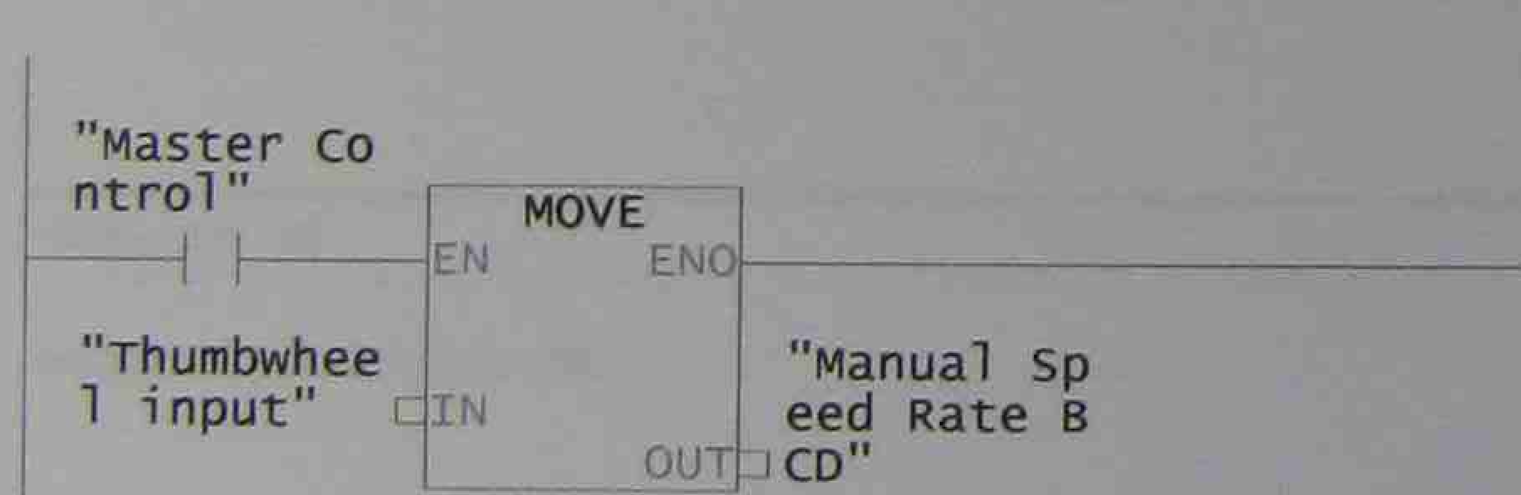
Network: 28

final output



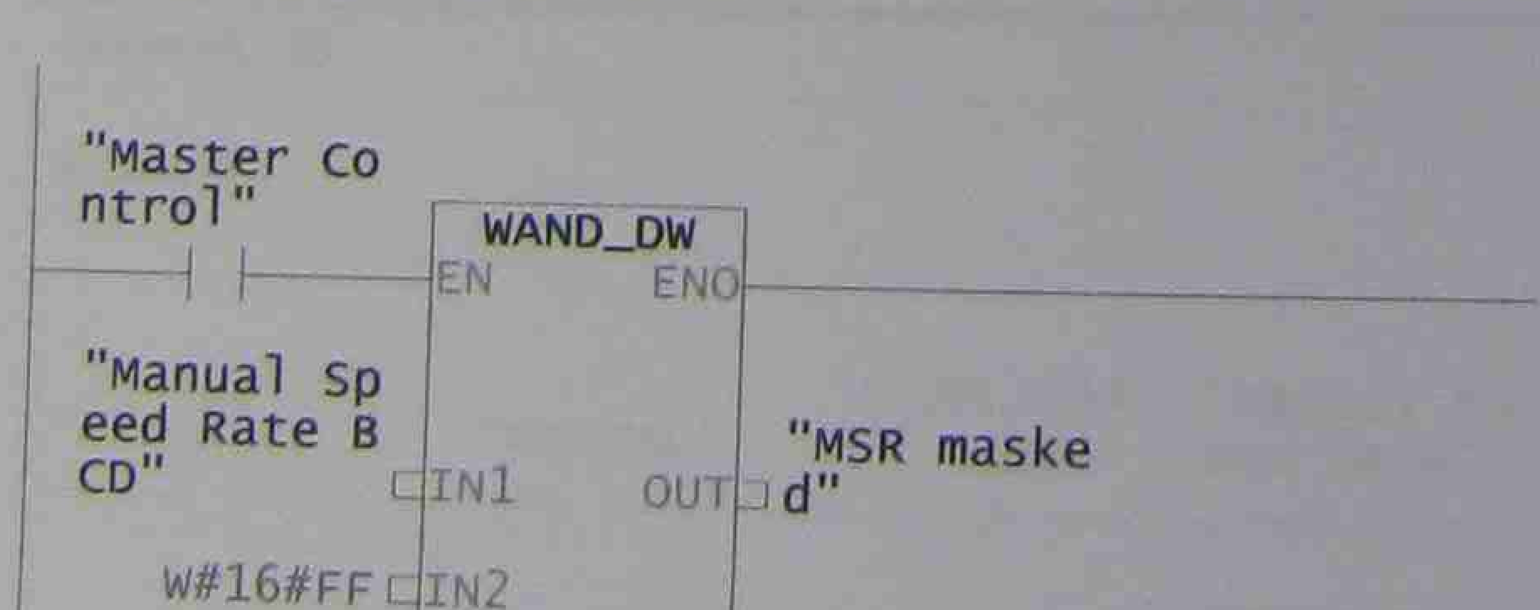
Network: 29

manual speed BCD



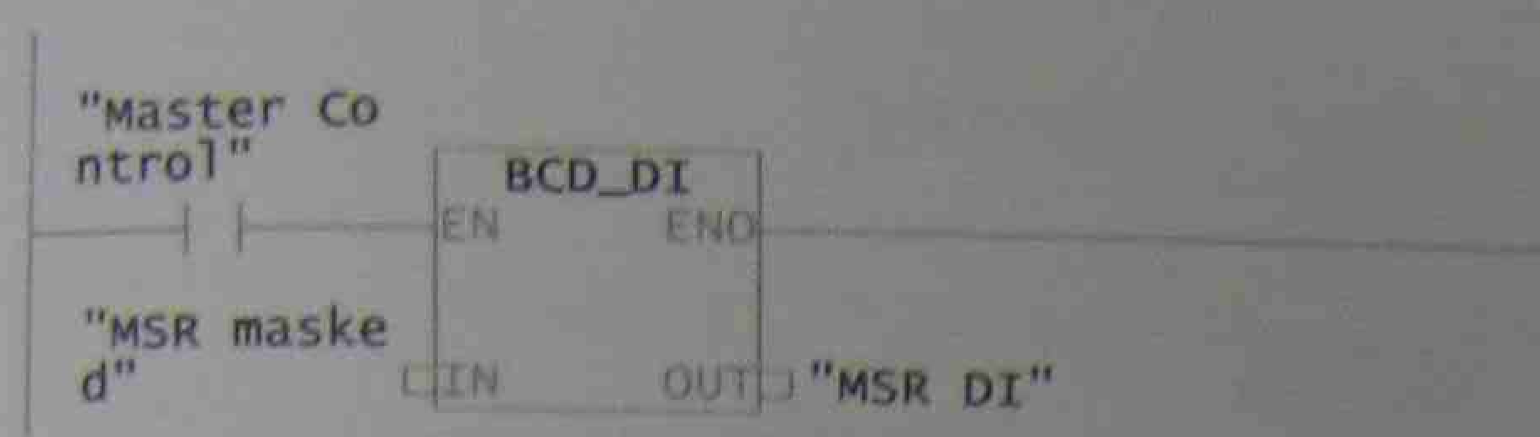
Network: 30

manual speed masking (right-2)



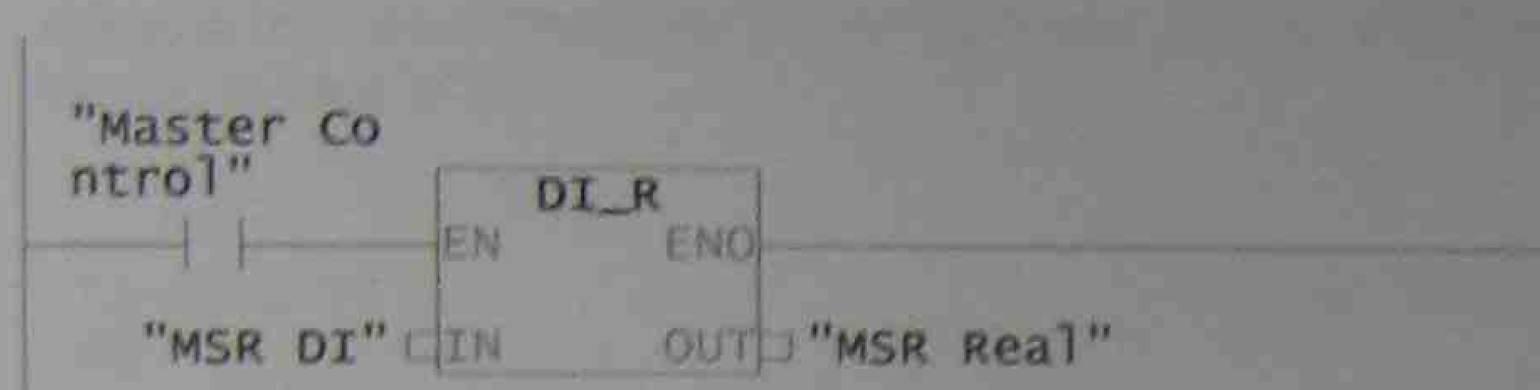
Network: 31

manual speed BCD-DI



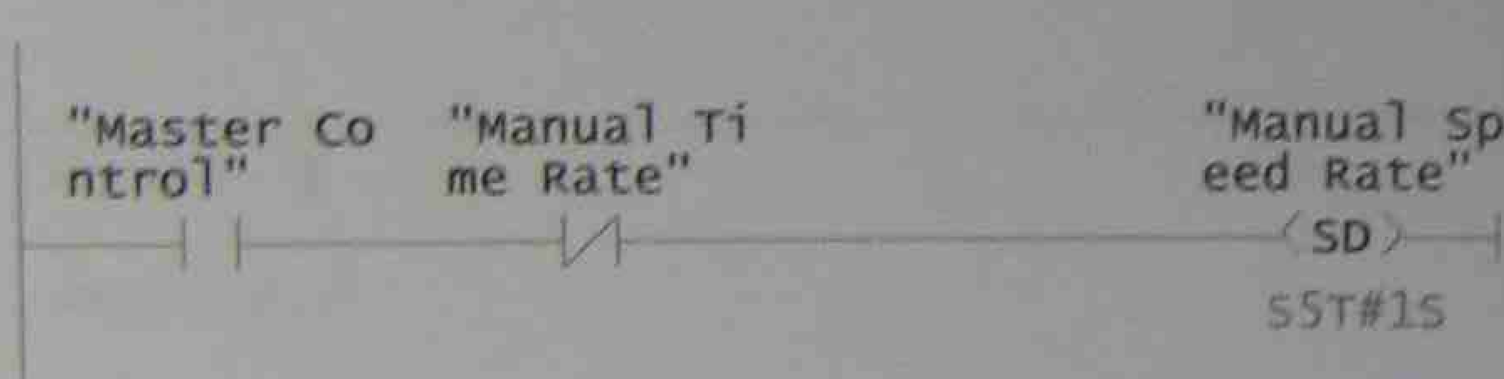
Network: 32

manual speed DI-R



Network: 33

accelerate/decelerate rate

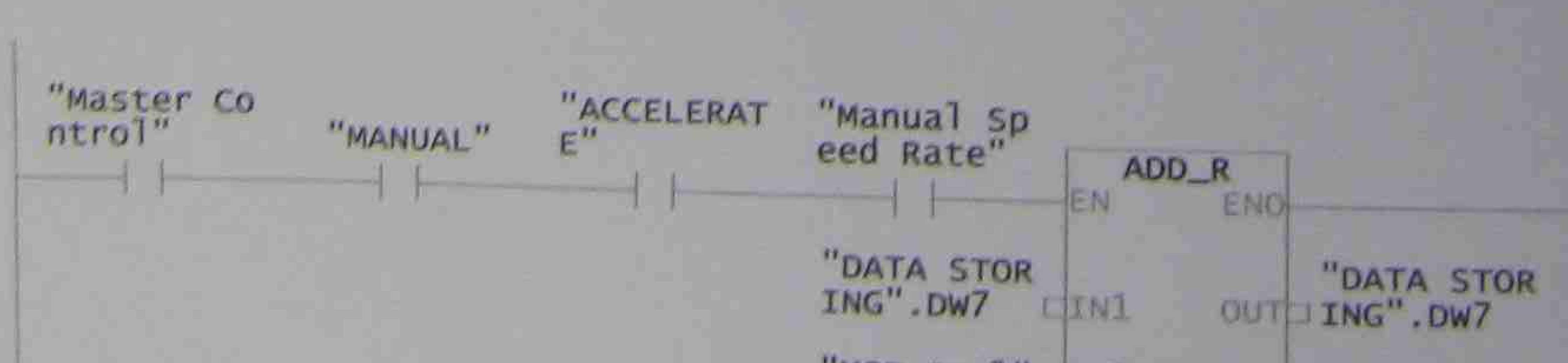


Network: 34

accelerate/decelerate rate

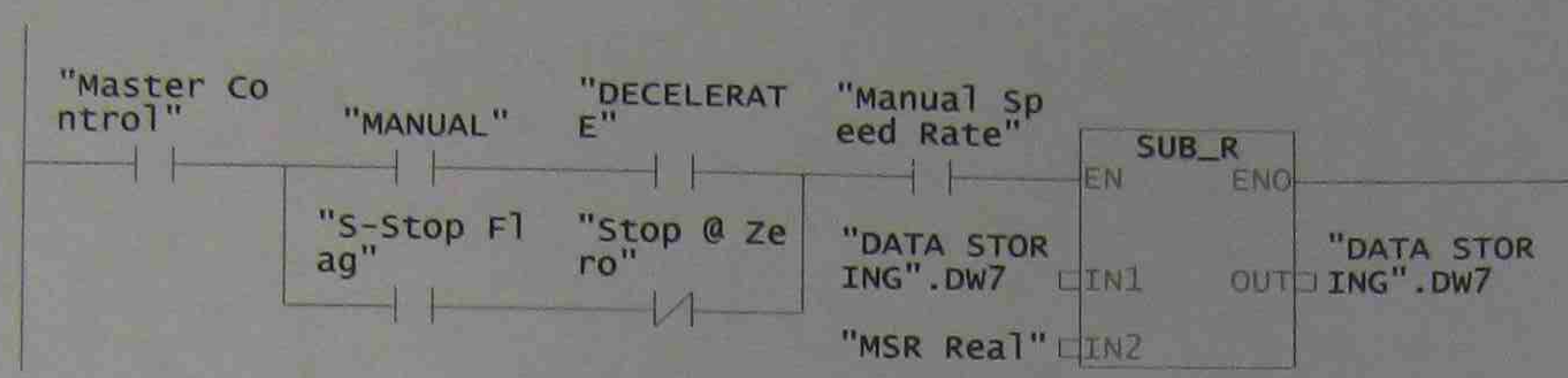
Network: 35

accelerate



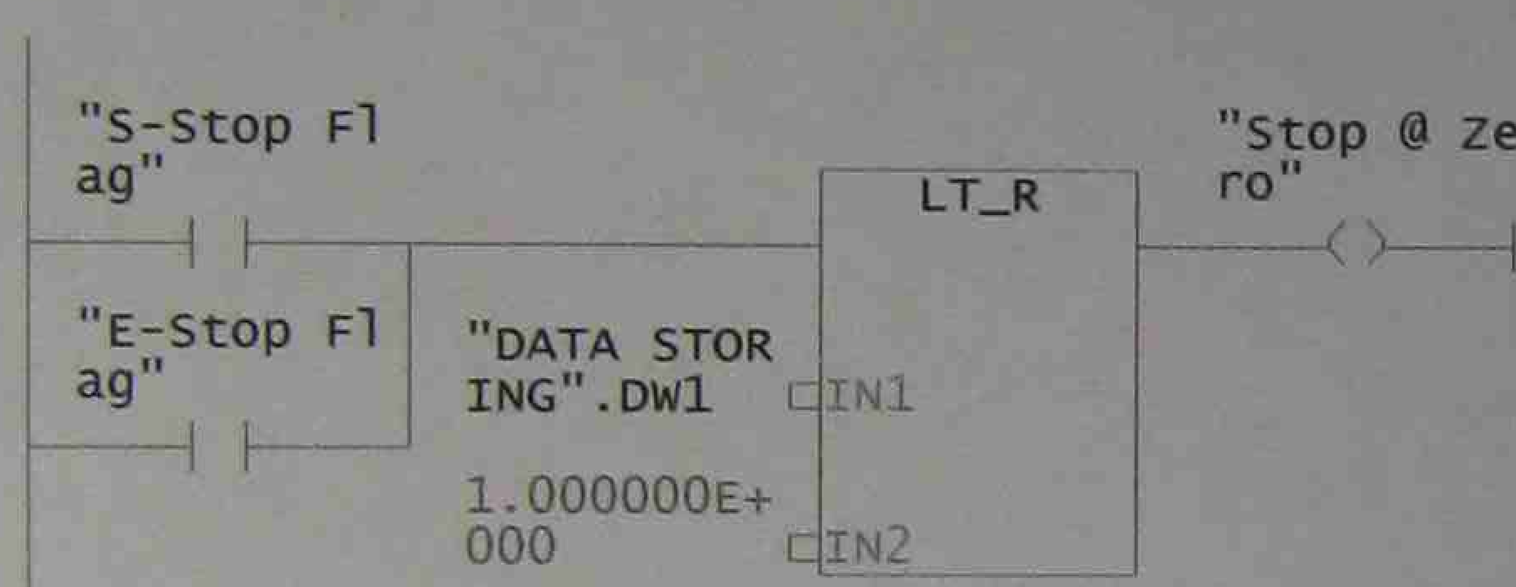
Network: 36

decelerate/soft stop



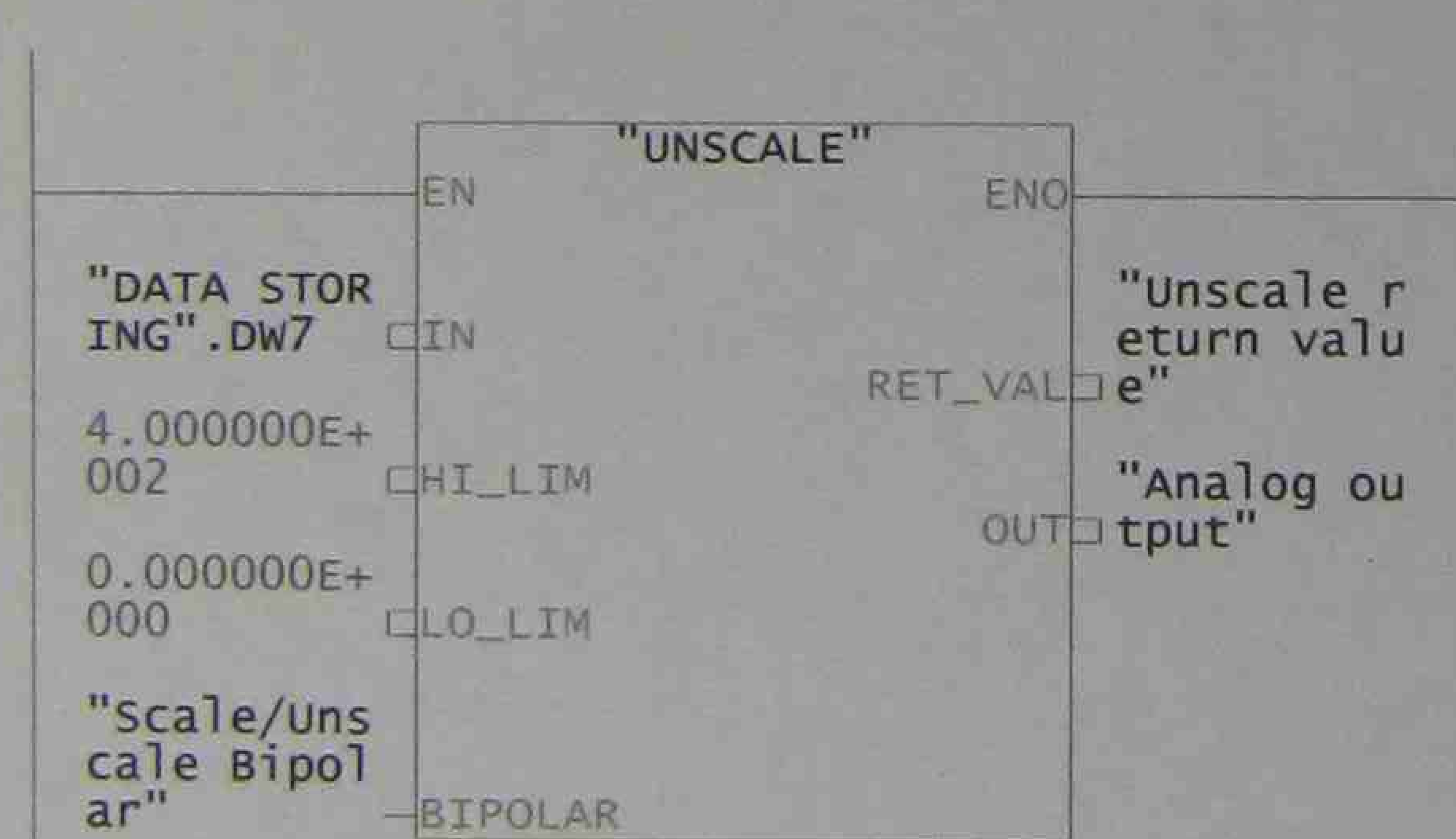
Network: 37

soft stop shut down (feedback hardly reach zero at stop -> compare less than 1 for stop instead of 0)



Network: 38

analog output



Name: File name:

PLC System Applications Temperature control Assignment

On/Off Control

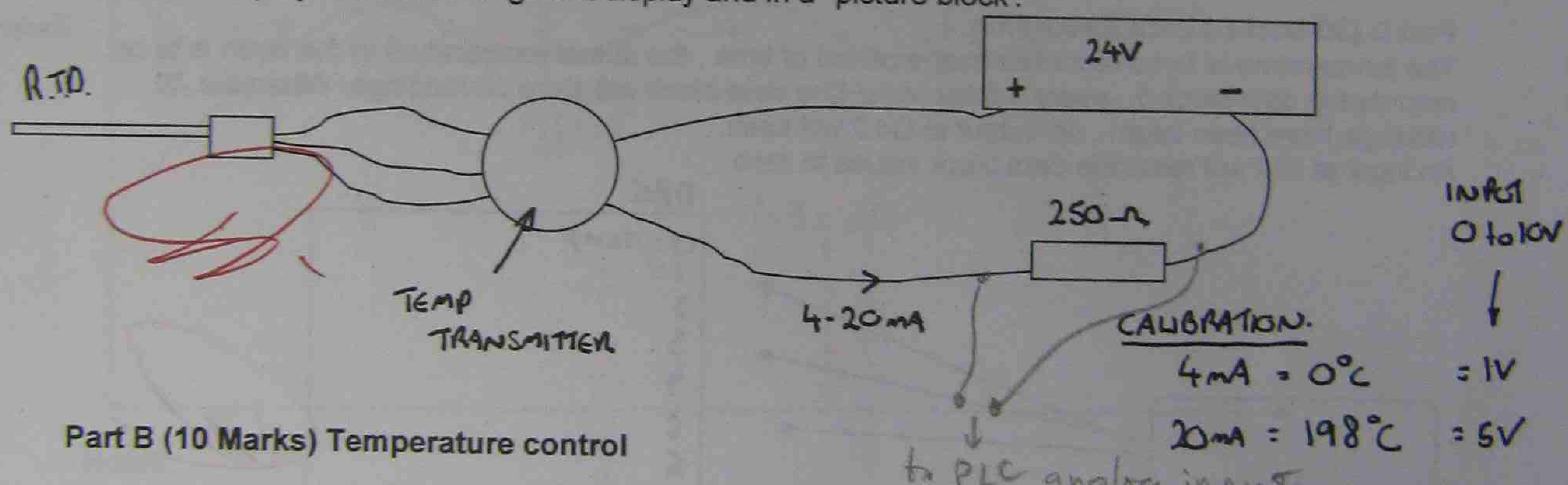
An on-off controller is the simplest form of temperature control device. The output from the device is either on or off, with no middle state. An on-off controller will switch the output only when the temperature crosses the setpoint. For heating control, the output is on when the temperature is below the setpoint, and off above setpoint. Since the temperature crosses the setpoint to change the output state, the process temperature will be cycling continually, going from below setpoint to above, and back below

n cases where this cycling occurs rapidly, and to prevent damage to contactors and valves, an on-off differential, or "hysteresis," is added to the controller operations. This differential requires that the temperature exceed setpoint by a certain amount before the output will turn off or on again. On-off differential prevents the output from "chattering" or making fast, continual switches if the cycling above and below the setpoint occurs very rapidly. On-off control is usually used where a precise control is not necessary, in systems which cannot handle having the energy turned on and off frequently, where the mass of the system is so great that temperatures change extremely slowly, or for a temperature alarm.

Use the analog input to your PLC to perform the following tasks

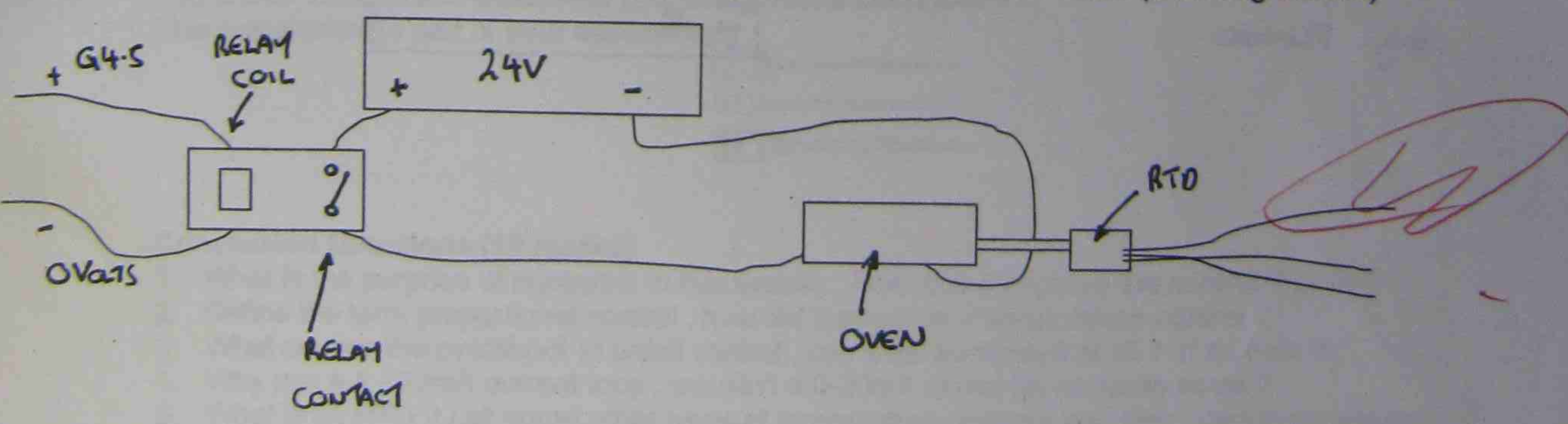
Part A (10 Marks) Temperature measurement

Connect a 4-20mA current loop as shown and measure the temperature in the room. The temperature should be displayed on the 7 segment display and in a "picture block'.



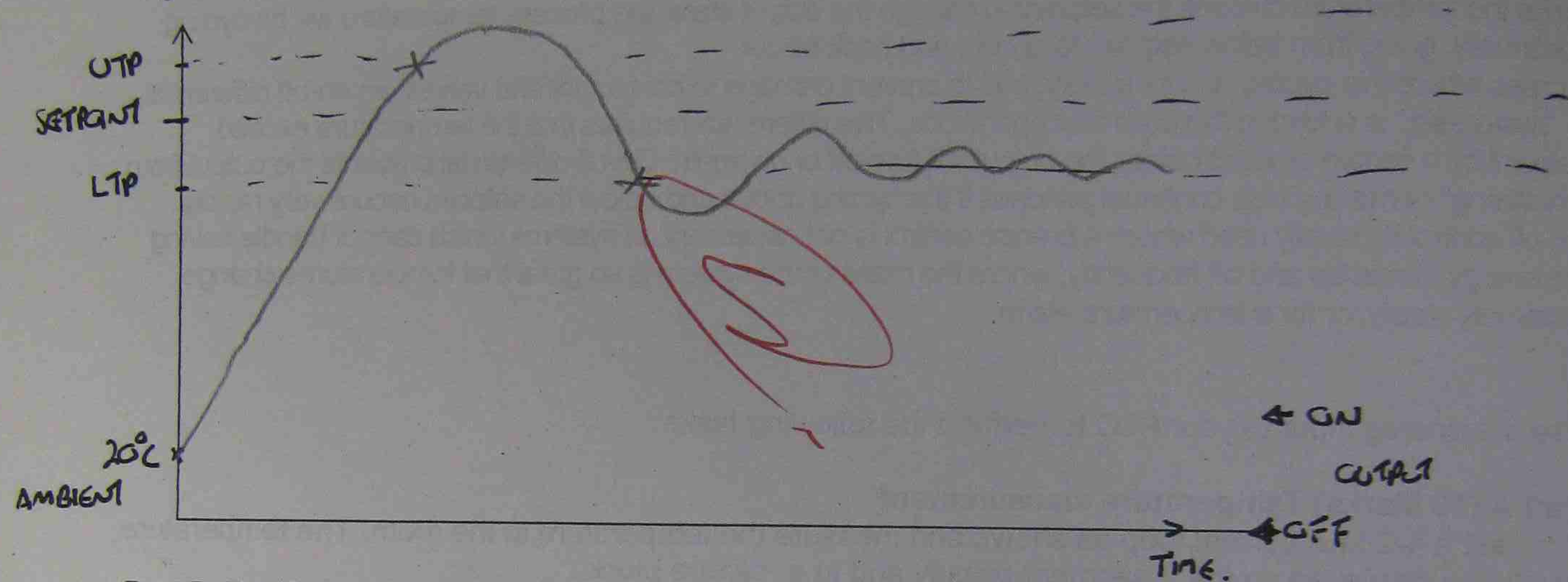
Part B (10 Marks) Temperature control

The temperature in an oven is to be controlled by your PLC , a digital output will control the power to the heater element at Q4.5.
The temperature will be set by the left two digits of the thumbwheel switch. (0-99 deg celcius)



Part C (10 Marks) Hystersis

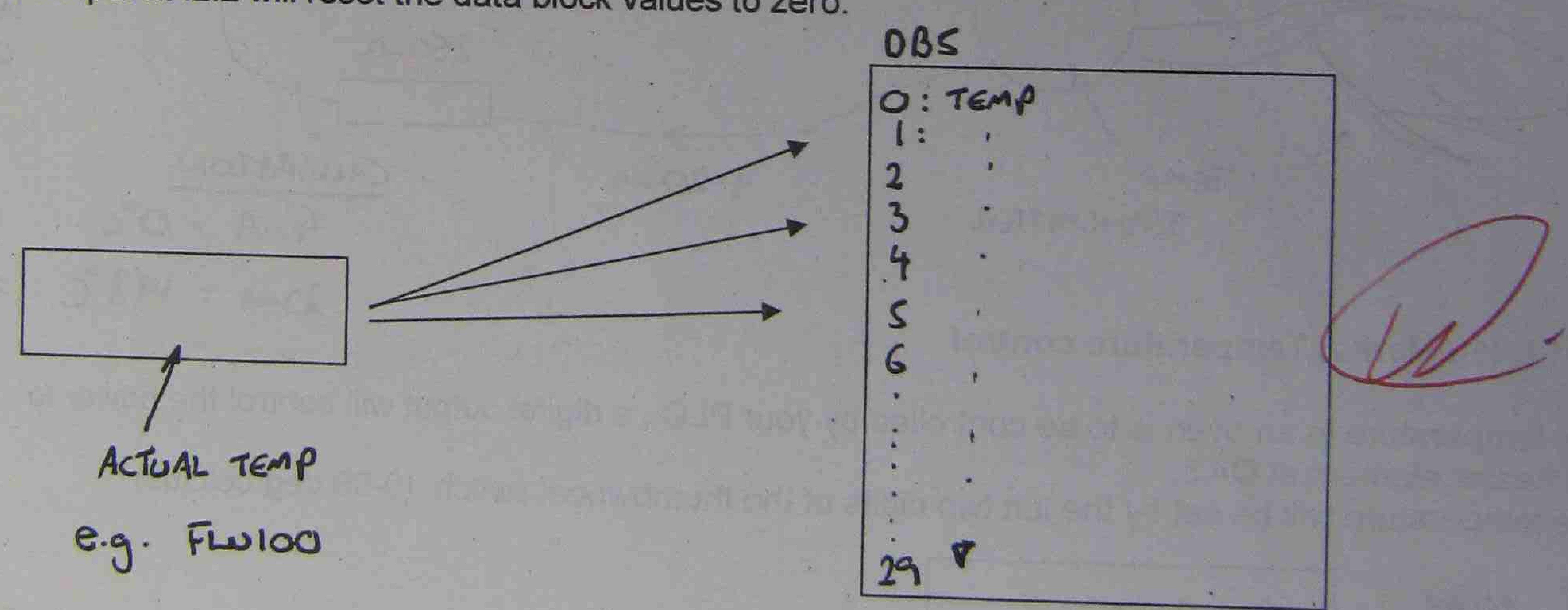
Add some Hystersis to you your control system , the Hystersis temperature is set by the right two digits of your thumbwheel switch.



Part D (30 Marks) Data Recording

The temperature is to be recorded over a period of time , the actual temperature in the oven is to be recorded in data block 5 , every 0.5 seconds/ The data block will store 30 readings. When the 30 readings have been taken , an output at Q4.2 will flash.

An input at I2.2 will reset the data block values to zero.



Proportional Control

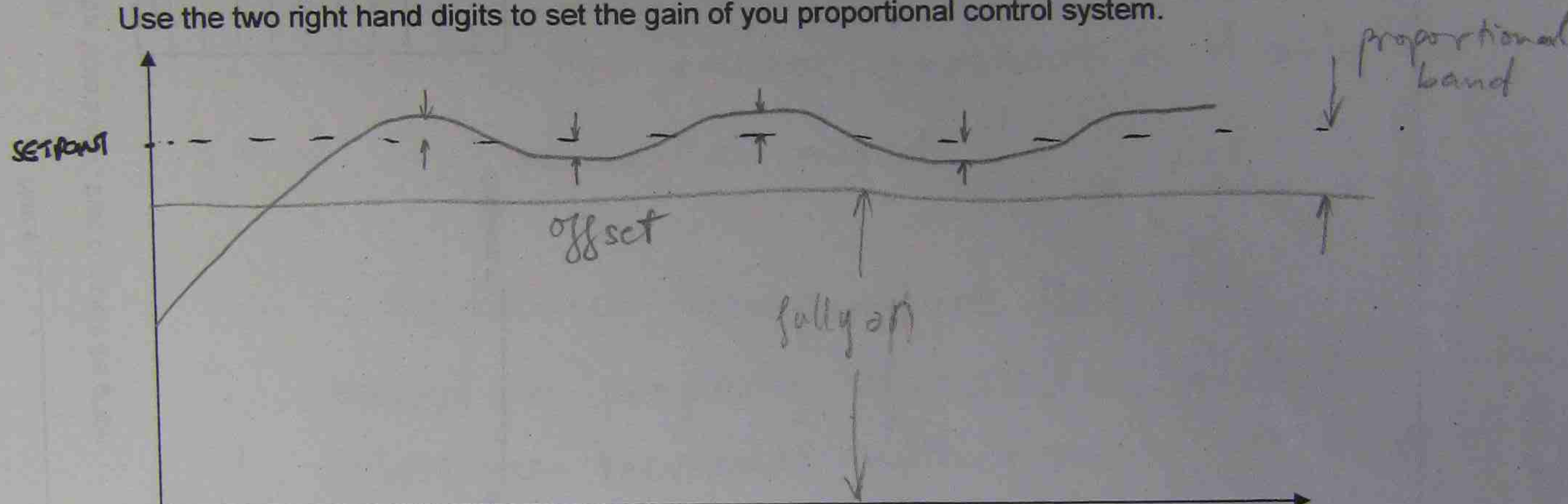
Proportional controls are designed to eliminate the cycling associated with on-off control. A proportional controller decreases the average power supplied to the heater as the temperature approaches setpoint. This has the effect of slowing down the heater so that it will not overshoot the setpoint, but will approach the setpoint and maintain a stable temperature. This proportioning action can be accomplished by turning the output on and off for short time intervals. This "time proportioning" varies the ratio of "on" time to "off" time to control the temperature. The proportioning action occurs within a "proportional band" around the setpoint temperature. Outside this band, the controller functions as an on-off unit, with the output either fully on (below the band) or fully off (above the band). However, within the band, the output is turned on and off in the ratio of the measurement difference from the setpoint. If the temperature is further from the setpoint, the on- and off-times vary in proportion to the temperature difference. If the temperature is below setpoint, the output will be on longer; if the temperature is higher, the output will be off longer.

Part E (30 Marks) Proportional Control.

Remove the Hystersis and On/Off control from your program and design a proportional control system that will average the output using pulse width modulation. Therefore controlling the temperature more accurately.

This is a design question and marks will be awarded for accuracy and simplicity of you system.

Use the two right hand digits to set the gain of you proportional control system.



Notes

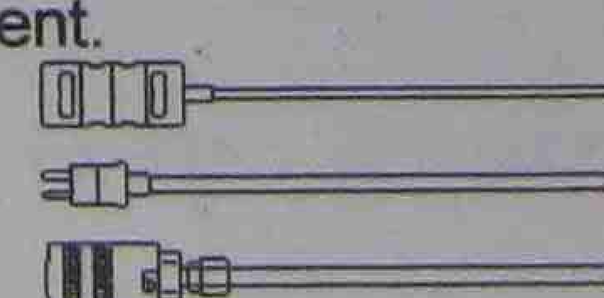
Use one shots for loading your data from the thumbwheel switches.

Document your program with line comments explaining the function of each part of your program.

Do your own work.

You **MUST** hand in this sheet with your assignment! Don't lose it !

This assignment is part of your assessment.



Conclusion Questions (10 marks)

1. What is the purpose of Hysteresis in this system , how dies it improve the control if at all ?
2. Define the term proportional control , how did it affect your temperature control ?
3. What causes the overshoot in on/off control , can it be eliminated at all ? If so how ?
4. Why use a 4 - 20mA current loop , wouldn't a 0-20mA signal be easier to scale ?
5. What is an RTD ? List some other types of temperature sensors and their operating ranges.

CONCLUSION QUESTION

1. Hystersis is used for avoiding noise around setpoint that could lead to continuos switching therefore improve lifetime of switches and components.

2. Proportional control generates an output proportional to error signal and attemp to correct the final output closer to setpoint. The higher gain the more accuracy it acts and final output closer to setpoint.

3. Temperature inertia causes overshoot and only be eliminated by derivative control.

5. RTD: Resistance Temperature Detector
   based on property of metallic conductor whose resistance increases with rising of temperature

   Others: thermocouple : 500°C ÷ 1500°C
   thermister : -100°C ÷ 300°C

4. A 4-20 mA current loop is used for some reasons:
   - The system needs a current to operate as analog input/output
   - Not to be confused between low limit of current as 0°C and non-operate system such as stopped by safety components. In other word 0 mA should be understand as not operating or stopped rather than lower limit of analog input.

LC System Applications

Name :________________________

Temperature Proportional Control reading Part E Practical 1

Record the temperature in your oven every 30 seconds in the table below , then complete the graph

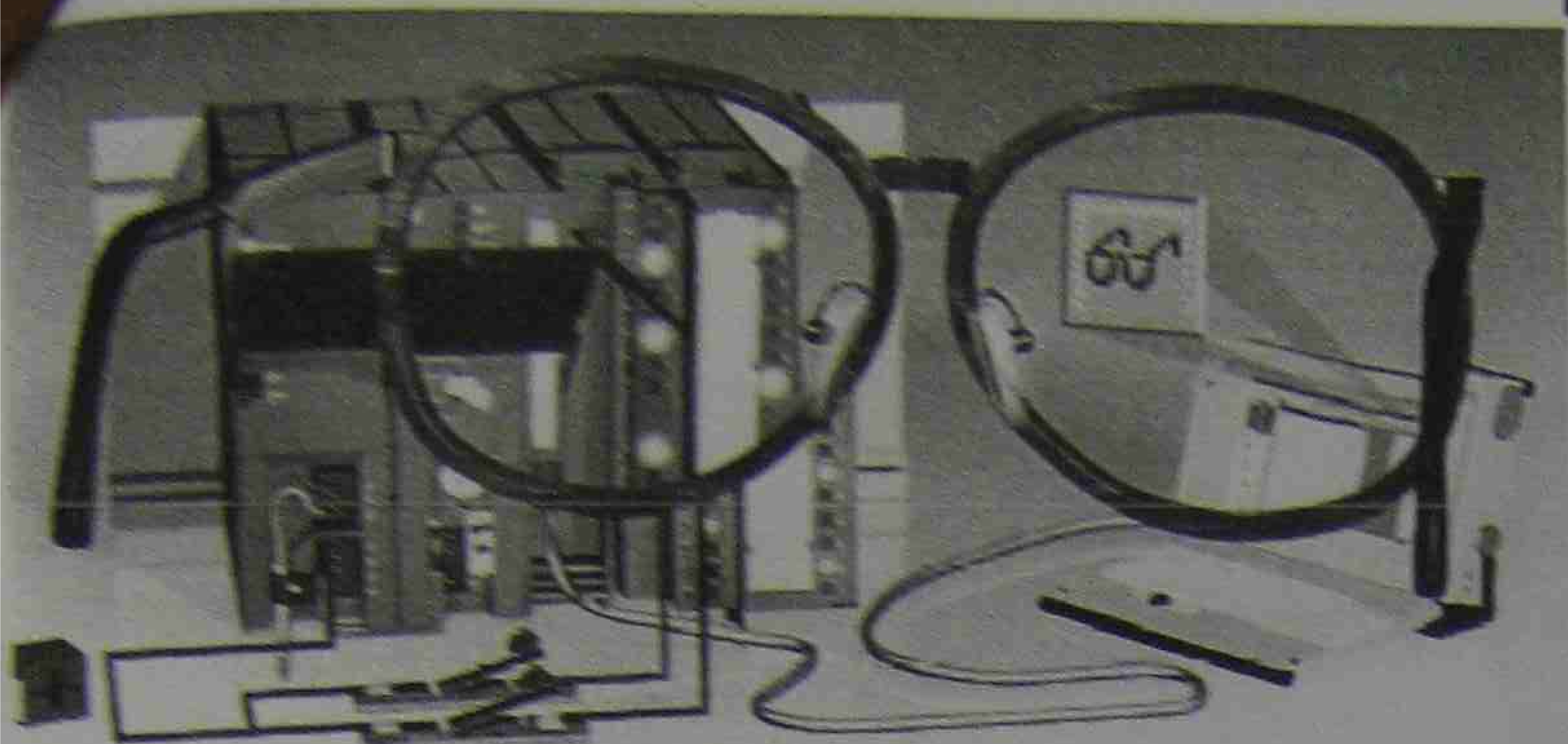
Hand this in with your file

Reading #	Temp	Reading #	Temp	Reading #	Temp
1		11		21	
2		12		22	
3		13		23	
4		14		24	
5		15		25	
6		16		26	
7		17		27	
8		18		28	
9		19		29	
10		20		30	

Setpoint = ______________ deg

Temp

LC System Applications -Major Project
Temperature Control with a Step 7 PLC



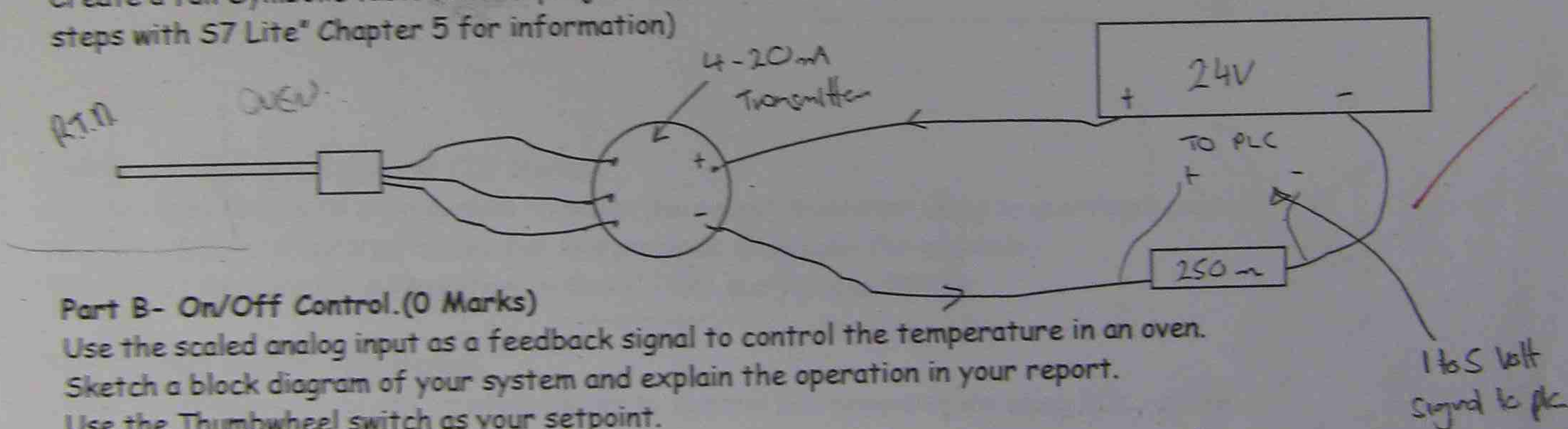
Aim: the aim of this major project is to assess student abilities to apply their knowledge to a new system. Building on previous knowledge the students will adapt this to a newer model PLC and software.

Procedure :

Examine the temperature control system that was used in a previous assignment.
Demonstrate that you are capable of connecting this system to your S7 313C plc.
Familiarize yourself with the S7 "lite" software , use the supplied PDF documents for reference material.

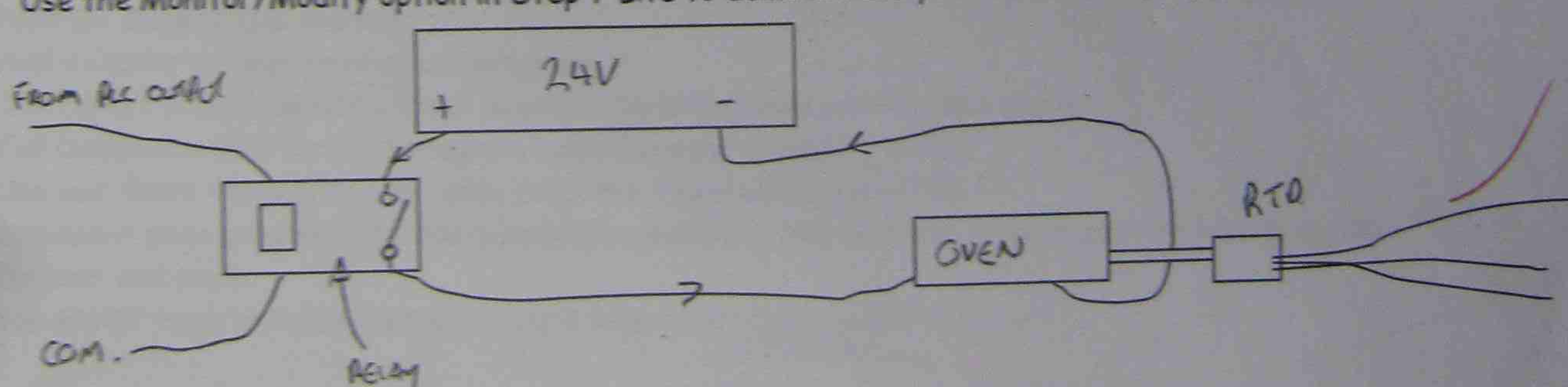
Part A - Analog Inputs.(0 Marks)

Connect the RTD input and scale this input to accurately display the temperature in a variable block or 7 segment display.
Create a full Symbolic table for the project and maintain the Symbolic table as your project develops. (See "First steps with S7 Lite" Chapter 5 for information)



Part B- On/Off Control.(0 Marks)

Use the scaled analog input as a feedback signal to control the temperature in an oven.
Sketch a block diagram of your system and explain the operation in your report.
Use the Thumbwheel switch as your setpoint.
Use the Monitor/Modify option in Step 7 Lite to demonstrate your functional project)

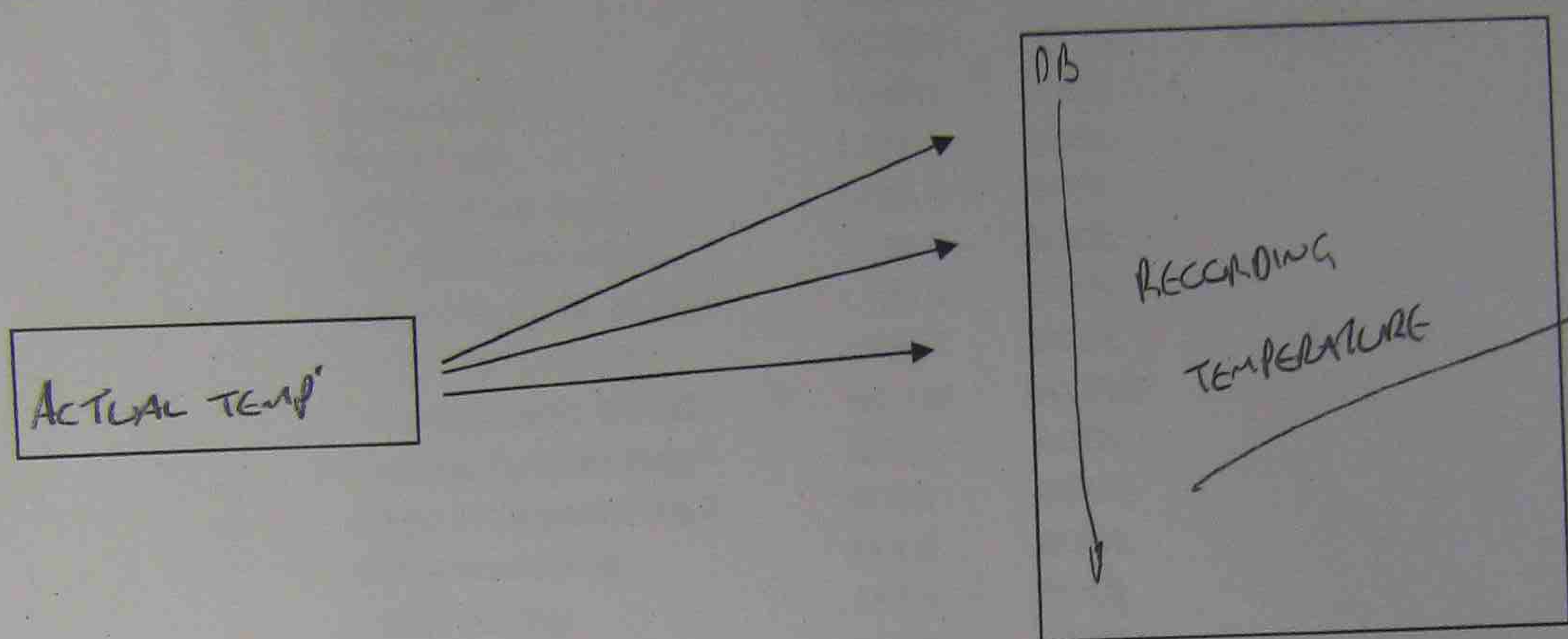


C- Hystersis.(10 Marks)

Add some Hystersis to you your control system , the Hystersis temperature is set by the thumbwheel switch.
Make sure your symbolic table is up to date and use the Monitor/Modify option in S7 lite to demonstrate your program.

Part D- Data logging.(10 Marks)

Add a data logging system where the temperature can be recorded in a data block over a period of time.
Make the reading every 0.4 seconds for testing purposes.
(You may use a SCADA system to perform this function using trends as an alternative to using a data block in the PLC , it is your choice.)
Using the PLC to record the data must include the use of a "pointer" for full marks.
See the S7 manual for information on Address registers (LAR 1)



Part E - Proportional Control.(10 Marks)

Add an improvement to your system to make the output smoother using proportional control.
You may use the integrated blocks for this purpose. (Pulsegen for example)
See the S7 Manuals for information on Con-C FB41 and Pulsgen FB43)

Part F- PID control.(10 Marks)

Using the integrated special function blocks to control the temperature using PID control.

Part G- Report.(10 Marks)

Add a report to your printed out program.
The report should include a block diagram explaining how your system works.
Full Documentation for your program including your Symbolic Table.
Use one shots for loading your data from the thumbwheel switches.
Document your program with line comments explaining the function of each part of your program.
Do your own work.
You **MUST** hand in this sheet with your assignment! Don't lose it!

Symbol table

Status	Symbol	Address	Data Type	Comment
	DATA RECORDING	DB 10	DB 10	
	CONT_C DB	DB 30	FB 41	
	PULSEGEN DB	DB 40	FB 43	
	SCADA MONITOR	DB 60	DB 60	
	CONT_C	FB 41	FB 41	Continuous Control
	PULSEGEN	FB 43	FB 43	Pulse Generation
	INDEX ADDRESS	FC 1	FC 1	
	DATA RESET	FC 2	FC 2	
	SCALE	FC 105	FC 105	Scaling Values
	Setpoint Select	I 126.0	BOOL	
	Reset Data	I 126.2	BOOL	
	DERIVATIVE Action	I 126.4	BOOL	
	INTEGRATIVE Action	I 126.5	BOOL	
	PROPORTIONAL Action	I 126.6	BOOL	
	DATA RECORDER	I 126.7	BOOL	
	THUMBWHEEL INPUT	IW 124	WORD	
	Oneshot Setpoint Flag 1	M 0.1	BOOL	
	Oneshot Setpoint Flag 2	M 0.2	BOOL	
	Data Record Flag	M 1.0	BOOL	
	Data Full Flag	M 1.1	BOOL	
	Oneshot Data Reset 1	M 1.2	BOOL	
	Oneshot Data Reset 2	M 1.3	BOOL	
	UNUSED_1	M 3.0	BOOL	
	UNUSED_2	M 3.1	BOOL	
	UNUSED_3	M 3.2	BOOL	
	UNUSED_4	M 3.4	BOOL	
	UNUSED_5	M 3.5	BOOL	
	UNUSED_6	M 3.6	BOOL	
	UNUSED_7	M 3.7	BOOL	
	UNUSED_8	M 4.0	BOOL	
	UNUSED_9	M 4.1	BOOL	
	UNUSED_11	M 4.2	BOOL	
	UNUSED_12	M 4.3	BOOL	
	UNUSED_13	M 4.4	BOOL	
	UNUSED_14	M 4.5	BOOL	
	UNUSED_15	M 4.6	BOOL	
	UNUSED_16	M 4.7	BOOL	
	SCALE's Bipolar	M 80.0	BOOL	
	P out	MD 50	DWORD	
	I out	MD 54	DWORD	
	D out	MD 58	DWORD	
	UNUSED_17	MD 62	DWORD	
	UNUSED_18	MD 66	DWORD	
	TEMPERATURE FEEDBACK	MD 100	DWORD	
	Rounded Temp	MD 104	DWORD	
	Temp in BCD	MD 108	DWORD	

Status	Symbol	Address	Data Type	Comment
	Double Integer Setpoint	MD 200	DWORD	
	SETPOINT	MD 204	DWORD	
	Double Integer Gain	MD 208	DWORD	
	GAIN	MD 212	DWORD	
	LMN/INV	MD 220	DWORD	
	Thumbwheel Value 1	MW 20	WORD	
	Masked 2 left digits	MW 24	WORD	
	Shift Righted	MW 26	WORD	
	Integer Setpoint	MW 28	WORD	
	Thumbwheel Value 2	MW 30	WORD	
	Masking Gain	MW 32	WORD	
	Integer Gain	MW 34	WORD	
	SCALE Return Value	MW 40	WORD	
	BCD Temp to Display	MW 110	WORD	
	UNUSED CONT_C Out	MW 120	WORD	
	Index Addressing	MW 140	WORD	
	CYCL_EXC	OB 1	OB 1	Cycle Execution
	CYC_INT5	OB 35	OB 35	Cyclic Interrupt 5
	SCALE Input	PIW 752	WORD	
	HEATER	Q 125.0	BOOL	
	Data Full Indicator	Q 125.2	BOOL	
	7-seg display	QB 124	BYTE	
	Recording Step	T 1	TIMER	
	Data Full Flash 1	T 2	TIMER	
	Data Full Flash 2	T 3	TIMER	

OB35:CYC_INT5

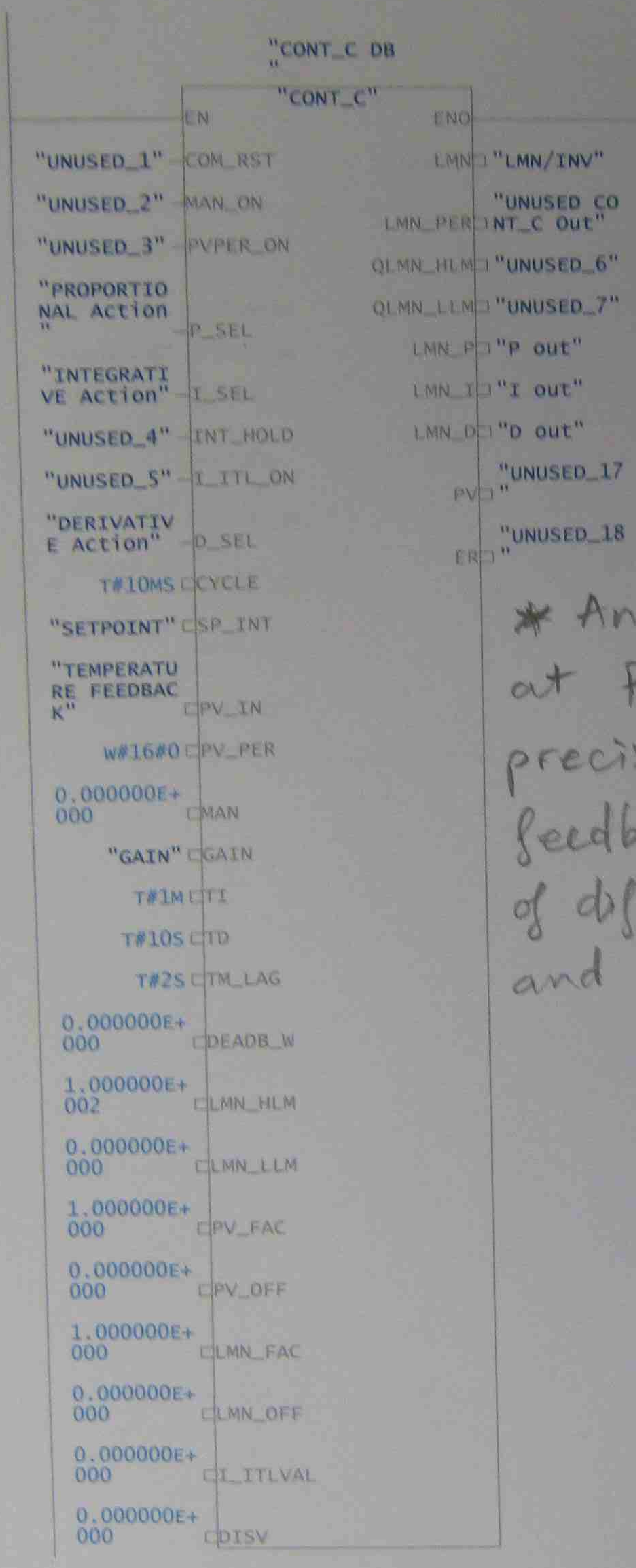
Cyclic Interrupt 5

```
Name:                     Time stamp Code:       27/05/09
Author:                              Interface:  20/01/04
Family:                   Lengths    Block:      00702
Version:        1.0                  Code:       00588
Code version: 2                      Data:       00028
```

Block: OB35 "Cyclic Interrupt"

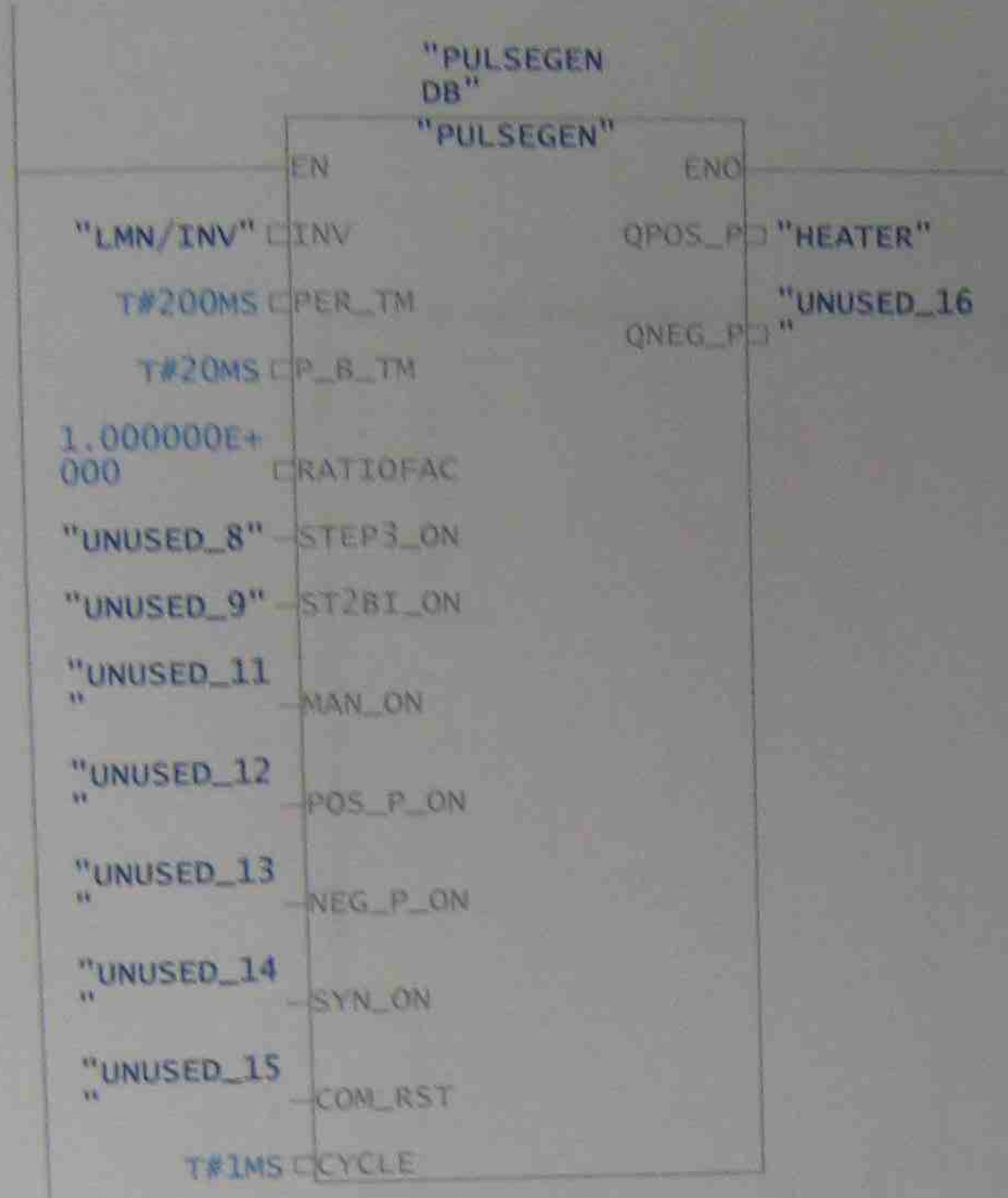
Address	Declaration	Name	Type	Start value	Comment
0.0	temp	OB35_EV_CLASS	BYTE		Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
1.0	temp	OB35_STRT_INF	BYTE		16#36 (OB 35 has started)
2.0	temp	OB35_PRIORITY	BYTE		11 (Priority of 1 is lowest)
3.0	temp	OB35_OB_NUMBR	BYTE		35 (Organization block 35, OB35)
4.0	temp	OB35_RESERVED_1	BYTE		Reserved for system
5.0	temp	OB35_RESERVED_2	BYTE		Reserved for system
6.0	temp	OB35_PHASE_OFFSET	WORD		Phase offset (msec)
8.0	temp	OB35_RESERVED_3	INT		Reserved for system
10.0	temp	OB35_EXC_FREQ	INT		Frequency of execution (msec)
12.0	temp	OB35_DATE_TIME	DATE_AND_TIME		Date and time OB35 started

Network: 1

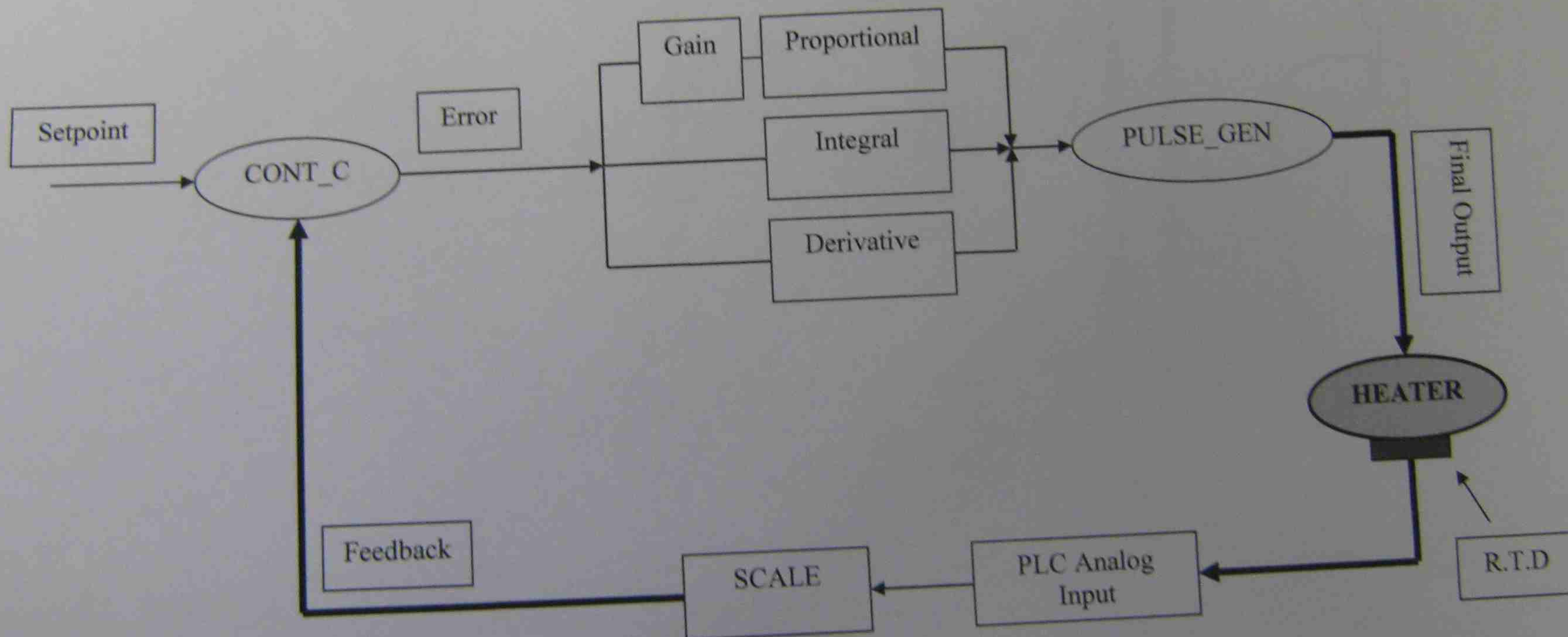


* An analog input should take place at PV_IN directly for more precision rather than using feedback from "SCALE" because of different cycles between OB1 and OB35.

Network: 2


"PULSEGEN DB"
"PULSEGEN"
EN
ENO
"LMN/INV" INV
QPOS_P "HEATER"
T#200MS PER_TM
QNEG_P "UNUSED_16"
T#20MS P_B_TM
1.000000E+000 RATIOFAC
"UNUSED_8" STEP3_ON
"UNUSED_9" ST2BI_ON
"UNUSED_11" MAN_ON
"UNUSED_12" POS_P_ON
"UNUSED_13" NEG_P_ON
"UNUSED_14" SYN_ON
"UNUSED_15" COM_RST
T#1MS CYCLE

PLC Temperature PID Controller



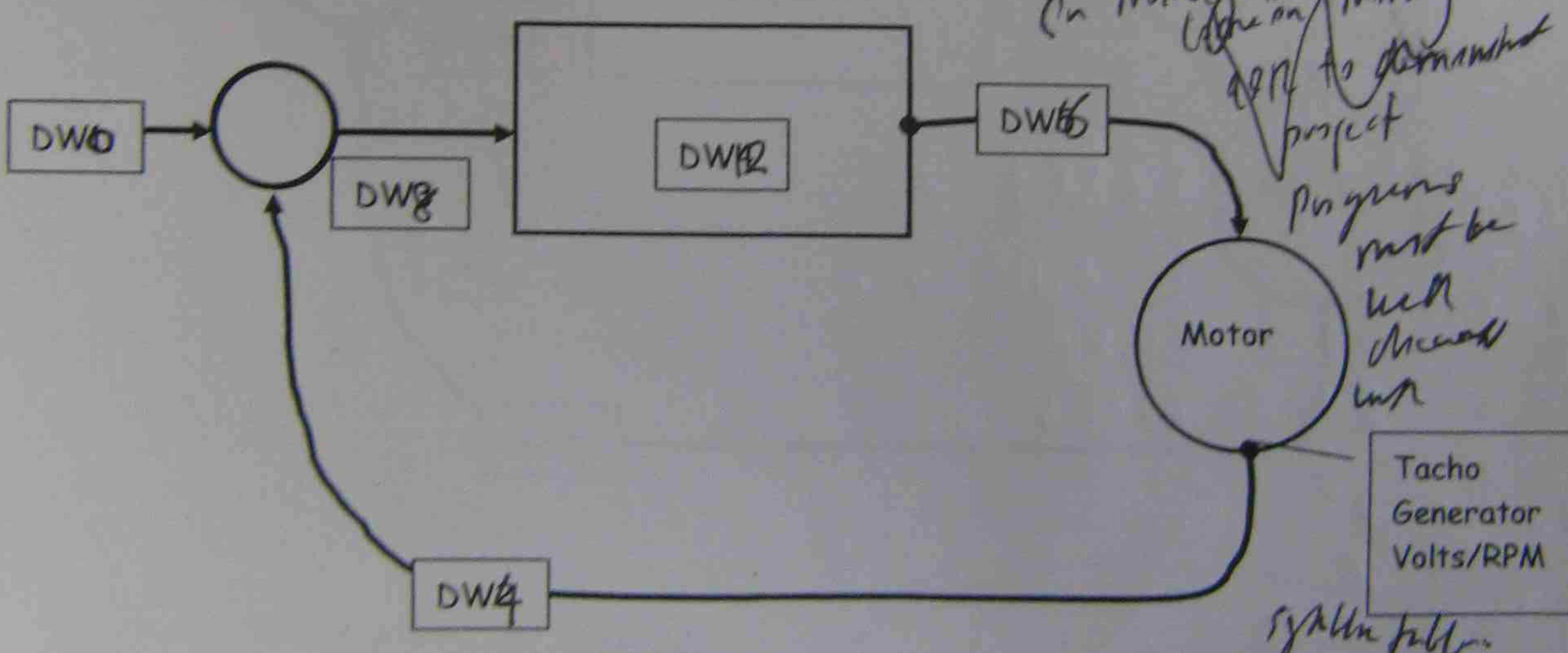
Aim :-

- To design a program using analog inputs and outputs that will control the speed of a DC Motor using Proportional Control.
- Observe and record the operation of the system.
- Add some Integral control to improve system performance.
- Add a control system for operator interfacing.

Procedure A - Proportional Control (10 Marks)
Use Data Block 10 for storing all settings.

- The speed of the motor is set by the thumbwheel switch after a one shot from an input. The setpoint is to be stored in data word 1 (Setpoint = DW1)
- The feedback is taken from a tacho generator connected to an analog input.(Feedback = DW2)
- The feedback is compared to the setpoint and the error is produced.(Error = DW3)
- The gain of the system is set by the same thumbwheel switch after a one shot. (Gain = DW4) (use I 2.1)
- The output of the controller is sent to the servo drive from an analog output .(Output = DW5)

Motor Drive Parameters	PLC Address	Set by One Shot
Motor Speed Set point	DW1	I 126.0
Proportional Gain	DW4	I 126.1



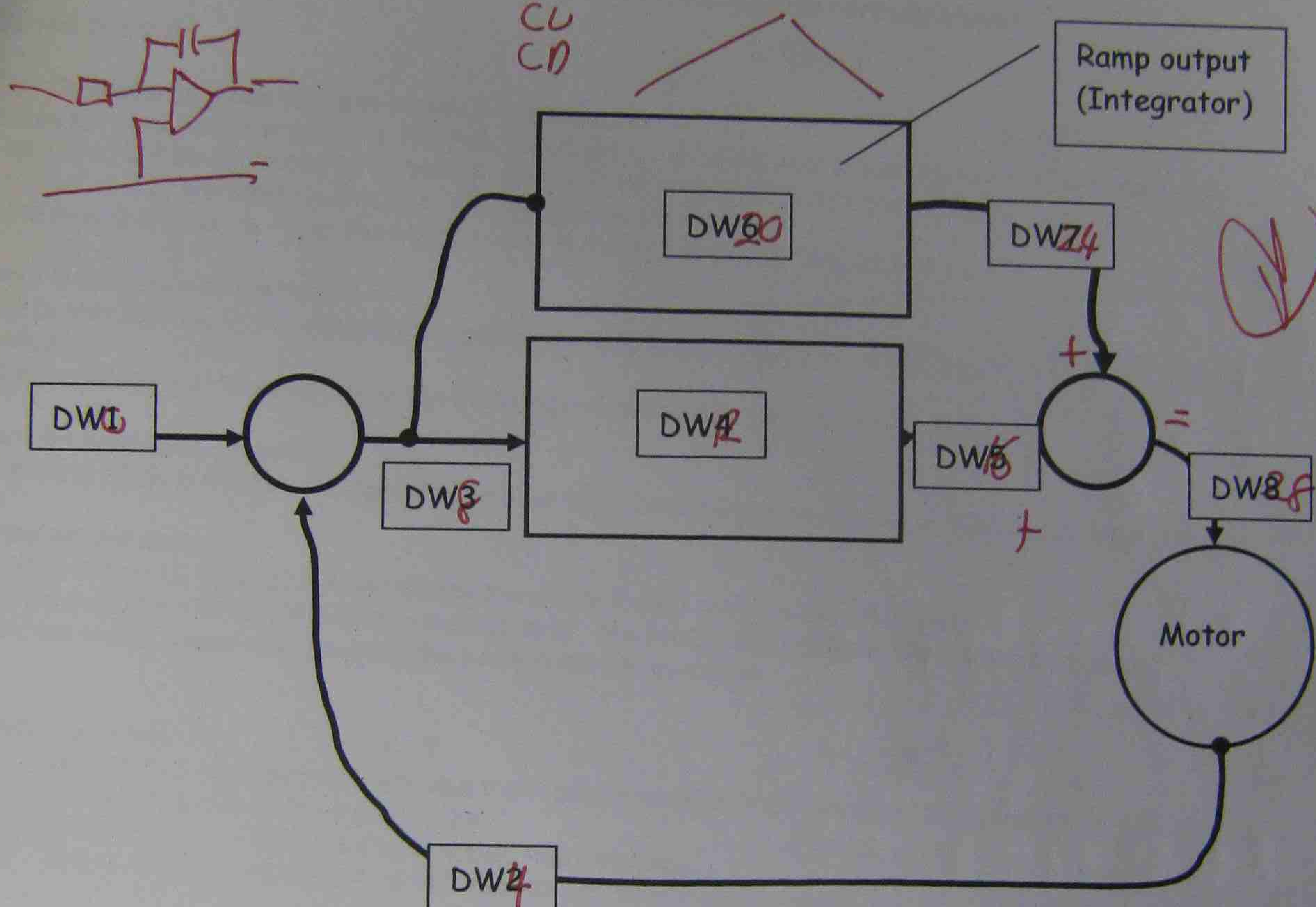
- Demonstrate your program's function and record the offset when the setpoint is set to 1000 RPM with a range of gains.

Gain	Offset/Error
2	354
4	270
6	180
8	145
10	110
12	96

dure B - Proportional + Integral action (20 Marks)

- To add integral action to the control system and remove the offset.

esign a system that will improve the performance of your control system by adding some form of Integral control. The Integral action will remove the offset , we will use a scada system to test your project for changed in load and see if the offset can be removed.
The Integral action rate should be adjustable via the thumbwheel switch and a one shot.
Marks will be awarded for accuracy , simplicity and reliability of your system.



Use cycler + counter
count up when error >0
count down when error <0

⇒ if error is higher should the integral action ramp up faster?
should I divide integral time by proportional action?

Complete control system (20 Marks)

to add operator control to the stopping and starting of the motor by gradual acceleration and deceleration and mergency stop and braking procedure.
operator will have the option to control the speed of the motor via a thumbwheel switch or two pushbuttons.

arting and stopping the motor
he motor is be started and stopped by a control station.
The start button (I2.2) is normally open and the stop button (I2.3) is normally closed.

Speed control

The Operator has two ways to control the setpoint
When the start button is pressed the motor start will be enabled and a "run" light at Q4.5 will turn on.
The speed will be controlled by either the thumbwheel switch or by two pushbuttons.
An input (I 2.6) will select which type of speed control is be used ,
If the input(I 2.6) is on , then the motor speed is controlled by the two pushbuttons .

Push Button speed control
One pushbutton (I2.4) will increase the motor speed gradually the other pushbutton (I2.5) will decelerate the motor gradually.
If the pushbutton is released , then the motor speed will remain at that speed level.

Thumbwheel speed control
If the input (I2.6) is off then the speed will be set via a thumbwheel switch as in part 1 of this assignment.

Stopping the motor
The stop button (n, closed) will decelerate the motor slowly until the speed is zero.
So if the operator hits the stop button momentarily , the motor will gradually slow down to a stop.
When the motor speed reaches zero the system will be shutdown.

Emergency stop.
A 2nd stop button (n, closed) (I2.7) will stop the motor immediately by removing any power to the motor and applying power to an electrical brake at Q 4.7.
The brake will remain on until the motor speed reaches zero.
In your program this Emergency Stop will remove any output to the motor.

Extra – Adjustable acceleration. (5 marks)
Make the acceleration and deceleration rates individually adjustable via the thumbwheel switch and a one shot.

Extra Part 2 –Working with S7 Lite (5 marks)
Complete the assignment using an S7 PLC and the S7 lite software.

Notes
Document your program thoroughly explain each point clearly.
Do your own work.
Marks are awarded for ingenious designs !
Marks are deducted for poor program comments and explanations.
Marks are awarded for the accuracy of your control system and your ability to demonstrate it.