Pixsys ELECTRONICS

PLC

EPL101-N

Datasheet

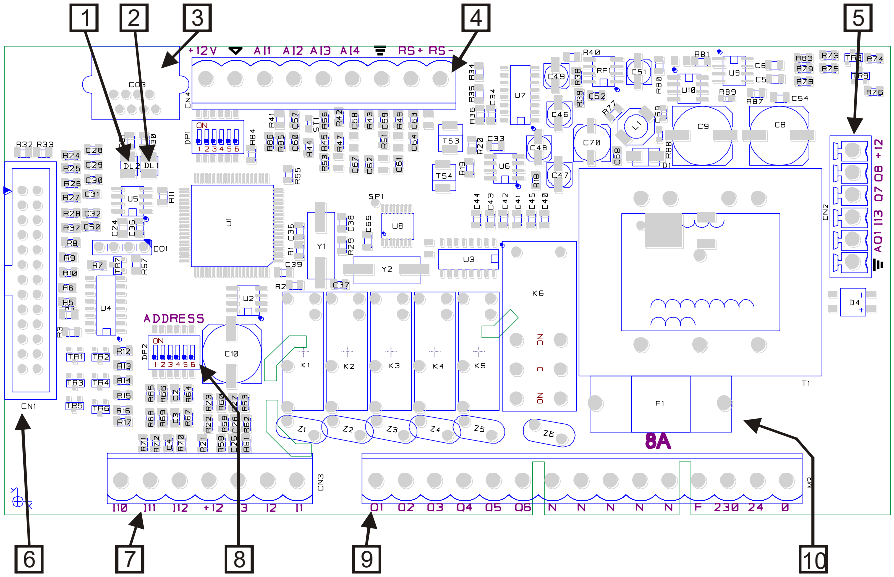
Sommario

1 Data acquisition + Power unit EPL101

Ordering codes:

EPL101-	□	□	-N
In/Out	1		6 relays + 4 analog inputs 16 bit
Power supply		AB	24Vac / 230Vac ± 10%
		C	127 Vac ± 10%
		D	12Vdc ± 15%

1.1 Description of PCB



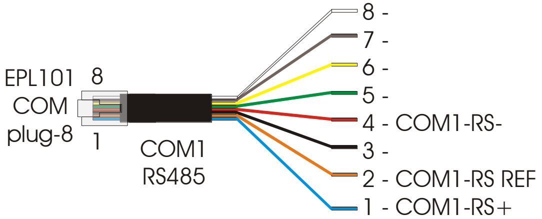
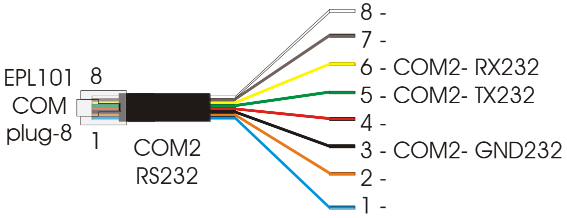
NO.	Description
1	Led **RUN** green : • **ON** → PLC is in RUN mode and is executing the instructions programmed by ladder language • **Slowly flashing** → (0,5 s on / 0,5 s off) PLC is used as I/O module (no ladder program loaded) • **Fast flashing** → (0,2 s on / 0,2 s off) only boot program is loaded; no main program and no ladder application.
2	Led **COM** yellow **:** • **ON** → for 50mS during transmission of each frame on one of the available serial ports. • **ON** → always during ladder programming or maintenance of PLC (updating main program)
3	COM. Connector plug-8pins for serial ports COM1 (RS485) and COM2 (RS232).
4	Terminal block for analog inputs AI1…4 and serial port COM1 (RS485).
5	Terminal block for input I13, outputs Q7,Q8 and analog output AQ1.
6	Connector for terminal (keyboard, display, led, infrared receiver)
7	Terminal block for digital inputs I1…I3, I10…I12
8	Dip-switch for the selection of slave address
9	Terminal block for outputs and supplì
10	Outputs fuse

Hardware data		
Supply	24Vac / 230Vac 5VA.	**Model EPL101-1AB**
	127Vac 5VA	**Model EPL101-1C**
	12Vdc 5VA	**Model EPL101-1D**
Analog inputs	AI1	- Tension 0-20mV (16 bit). - Tension 0-1V (16 bit). - Current 0-20mA (16 bit). - Current 4-20mA (16 bit). - Thermocouples K, S, T, R, J, E. - PT100,NI100 (2 or 3 wires) - NTC-10K (β=3435)
	AI2	- Tension 0-20mV (16 bit). - Tension 0-1V (16 bit). - Thermocouples K, S, T, R, J, E. - PT100,NI100 (2 wires or compensation) - NTC-10K (β=3435)
	AI3	- Tension 0-20mV (16 bit). - Tension 0-1V (16 bit). - Thermocouples K, S, T, R, J, E. - PT100,NI100 (2 wires or compensation) - NTC-10K (β=3435)
	AI4	- Tension 0-20mV (16 bit). - Tension 0-1V (16 bit). - PT100,NI100 (2 or 3 wires) - NTC-10K (β=3435)
Digital inputs	I1÷I3	- **Model EPL101-1AB / EPL101-1C** PNP inputs or 0..10V 10bit - **Model EPL101-1D** PNP inputs or 0..15V 10bit
	I12	- PNP input
	I4÷I9	- Keyboard keys
Digital inputs/ encoder inputs	I10/A1 I11/B1	- PNP inputs or bidirectional encoder no. 1 (1 KHz)
	I13/A2	- **Model EPL101-1AB / EPL101-1C** NPN input or monodirectional encoder no. 2 (2 KHz) - **Model EPL101-1D** Magnetic pick-up input for monodirectional counter n° 2 (4 KHz)
Relay outputs	U1÷U5	- Relay 5A-250 Vac.
	U6	- Relay 16A-250 Vac.

Hardware data			
Digital outputs	U7÷U8	-	open-colletor outputs (closing to ground if active) 20 mA max
Analog output	AQ1	-	Tension 0-5V (8 bit) 20 mA
Serial ports	COM1	-	RS485 on terminal block and connector COM plug-8 poles (not isolated).
	IR	-	Infrared receiver for remote switch.
	COM2	-	RS232 on connector COM plug-8 poles (not isolated).

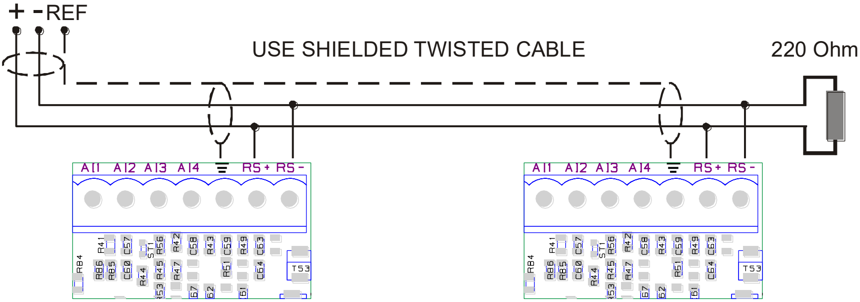
1.2 Electrical wirings

Name	Description
0	**Model EPL101-1AB** Common pin for power supply 24 / 230 VAC. To improve noises immunity, the secondary of dedicated transformer is highly recommended **Model EPL101-1C** Common pin for power supply 127VAC. To improve noises immunity, the secondary of dedicated transformer is highly recommended **Model EPL101-1D** Negative power supply 12Vdc.
24	**Model EPL101-1AB** Supply 24 VAC. Use this pin for supply 24 VAC **Model EPL101-1C** Not connect. **Model EPL101-1D** Positive power supply 12Vdc.
230	**Model EPL101-1AB** Supply 230 VAC. Use this pin for supply 230 VAC. **Model EPL101-1C** Supply 127 VAC. Use this pin for supply 127 VAC. **Model EPL101-1D** Not connect.
F	Phase input for relay outputs. Fuse 8A -250V provided for safety
N	Neutral pins of outputs.
Q1	Relay output Q1. If output is active, F signal reported on Q1
Q2	Relay output Q2. If output is active, F signal reported on Q2

Q3	Relay output Q3. If output is active, F signal reported on Q3
Q4	Relay output Q4. If output is active, F signal reported on Q4
Q5	Relay output Q5. If output is active, F signal reported on Q5
Q6	Relay output Q6. If output is active, F signal reported on Q6
+12	Common positive signal for digital inputs. Connect this signal to one of the digital inputs I1÷I3 or I10÷I12, to activate the input. Signal available on these pins can supply sensors (current/tension) connected to the analog inputs (Warning: on these pins the available tension is Vcc, not stabilized!).
I1	Digital input PNP.
I2	Digital input PNP.
I3	Digital input PNP.
I10	Digital input PNP.
I11	Digital input PNP.
I12	Digital input PNP.
I13	Digital input NPN.
▽	Reference signal for analog inputs
AI1	Positive signal for analog input AI1.
AI2	Positive signal for analog input AI2.
AI3	Positive signal for analog input AI3.
AI4	Positive signal for analog input AI4.
⏚	Ground for circuit and for tension +12. Reference signal for serial COM1 (RS485 not isolated).
RS+	Signal RS485+ for COM1.
RS-	Signal RS485- for COM1.
COM1 **RS485**	 EPL101 COM plug-8 8 1 COM1 RS485 8 - 7 - 6 - 5 - 4 - COM1-RS- 3 - 2 - COM1-RS REF 1 - COM1-RS+
COM2 **RS232**	 EPL101 COM plug-8 8 1 COM2 RS232 8 - 7 - 6 - COM2- RX232 5 - COM2- TX232 4 - 3 - COM2- GND232 2 - 1 -

1.2.1 Example of connection to RS485

Below an example of connection for several modules EPL101 to RS485 for communication with a master device.



1.3 Configuration of analog inputs

Analog inputs of EPL101 must be properly set both at software level (selecting choosen value on special-markers SM40..43) and by correct setting of internal jumpers. Check the following table to get the options allowed for each input.

Selection of inputs		AI1	AI2	AI3	AI4[1]
0	Desabled	○	○	○	○
1	0..1 V	○	○	○	○
2	0..20 mV	○	○	○	○
3	0..20 mA	○	×	×	×
4	4..20 mA	○	×	×	×
5	TC K	○	○	○	×
6	TC S	○	○	○	×
7	TC T	○	○	○	×
8	TC R	○	○	○	×
9	TC J	○	○	○	×
10	TC E	○	○	○	×
11	TC B[2]	○	○	○	×
12	PT100	○	○[3]	○[4]	○
13	NI100	○	○[5]	○[6]	○
14	Compens. PT/NI	×	○	○	×
15	NTC-10K	○	○	○	○

○ = selection allowed
× = selection not allowed

According to the table, the EPL101can read following inputs:

- Up to 3 thermocouples (K, S, T, R, J, E, B)
- Up to 4 PT100/NI100 2wires or up to 2 PT100/NI100 3wires
- Up to 4 NTC-10K
- Up to 4 inputs 0..1 V or 0..20 mV
- 1 input 0..20 mA or 4..20 mA

[1] Input AI4 cannot be used if thermocouples are connected to one of the channels AI1, AI2 or AI4 because it is internally connected to cold junction.

[2] Thermocouple B available starting with firmware 2.00

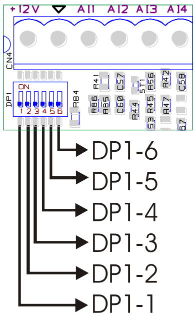
[3] Only for PT100 2wires

[4] Only for PT100 2wires

[5] Only for NI100 2wires

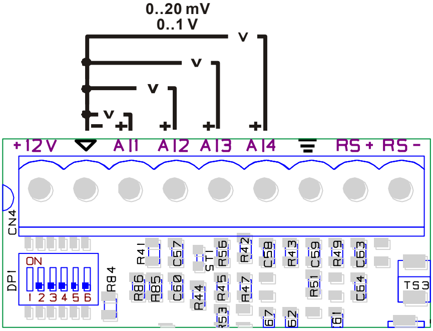
[6] Only for NI100 2wires

Check the table below for the setting of internal jumpers

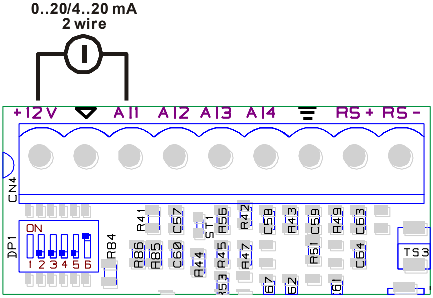
Switch DP1	DP1-	Function of switch
	2	Place the switch on for PT100, NI100 or NTC-10K on AI1. Do not place switch on for tension or current signals
	3	Place the switch on for PT100 and NI100 2wires or NTC-10K on AI2. Do not place switch on if you connect the compensation wire (for PT100/NI100 on AI1) or tension signals
	4	Place the switch on for PT100 and NI100 2wires or NTC-10K on AI3. Do not place switch on if you connect the compensation wire (for PT100/NI100 on AI4) or tension signals
	5	Place the switch on for one or more thermocouples on AI1, AI2 or AI3.
	6	Place the switch on for current signals (0..20 mA or 4..20 mA) on AI1.

1.3.1 Examples of wirings ffor the most common sensors/signals

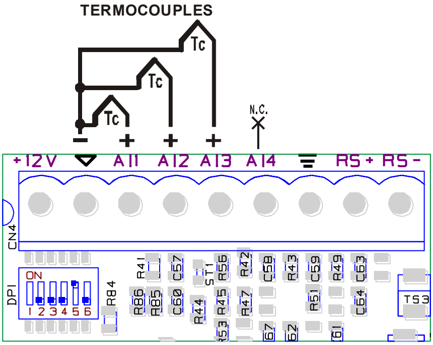
- Tension 0..20 mV or 0..1 V:



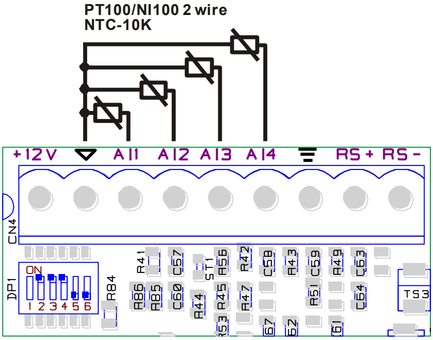
- Current 0..20 mA or 4..20 mA:



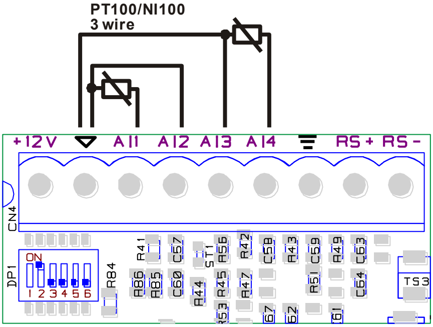
- Reading of thermocouples:



- Reading of PT100/NI100 2wires or NTC-10K:

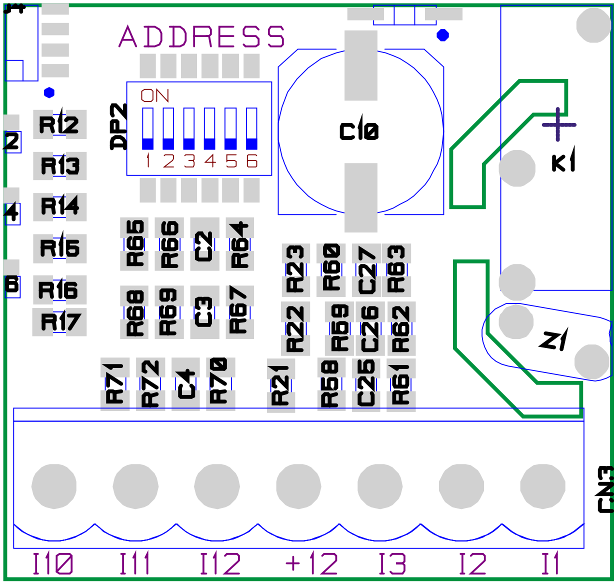


- Reading of PT100/NI100 3wires:



1.4 Entering slave address of EPL101

A jumper is provided to configurate the address of module for serial communication with a master device. There are 64 possible combinations of this jumper. To connect more than 64 devices to the same network, it is necessary to change the value of offset. . The address is given as follows:
MODULE ADDRESS = ADDRESS OFFSET + JUMPERS COMBINATION
The address offset, which is a value stored on memory of EPL101 (default ="1"), can be modified by writing on Word SM82.
A multiplying value is assigned by the EPL101 to each of the 6 dips
(dip set to ON = 1, dip set to OFF = 0) and values are added according to this formule:
Combination = (DIP1*1) + (DIP2*2) + (DIP3*4) + (DIP4*8) + (DIP5*16) + (DIP6*32)



1.5 Programming the Memory-card

To program the Memory card it is necessary to have one EPL101-1AB, the programming software PLProg and serial cable PLC to PC. Follow the steps below:

1. Connect EPL101-1AB to power supply
2. Enter memory card on connector CO1. Make sure that pin1 of the connector corresponds exactly to pin1 of Memory Card
3. Start PLProg and open the file which you want to download on Memory Card.
4. Compile the project.
5. From “PLC”, select “Create memory card” and wait for the end of programming
6. Memory is now ready for the programmino of other EPL101 (as explained in the following paragraph).

1.6 Using the Memory-card

EPL101 may be programmed by Memory Card. This enables the software upgrade (both firmware and Ladder application) without connection to a PC. Follow the steps below:

1. Switch-off the PLC to be programmed
2. Enter Memory card (already programmed as explained in previous paragraph) on connector CO1, Make sure that pin1 of the connector corresponds exactly to pin1 of Memory Card.
3. Connect EPL101-1AB to power supply and wait until boot program can trace the memory Card and starts the programming. During the programming, the led RUN and COM will alternately switch-on, indicating the program advancement
4. After programmino has been completed (when led RUN stays ON), remove the Card
5. The program saved on memory Card is now stored on the PLC.

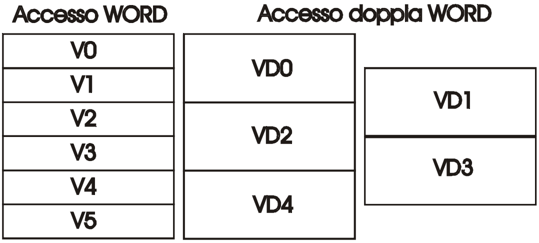
1.7 Memory areas of EPL101

EPL101 is provided with memory areas for the reading and/or writing of program data. Memory areas can be accessed by instructions which enter single bit (B), by word (W) or double word (D).

SIGLA	AREA	ACCESS
V	Area Variables V	B, W, D
SM	Area Special Marker	B, W, D
I	Area Digital inputs	B, W
AI	Area Analog inputs	B, W
Q	Area Digital outputs	B, W
M	Area Marker	B, W
AQ	Area Analog outputs	B, W
T	Area Timer	B, W
PT	Area Preset Timer	B, W
C	Area Counters	B, W
PV	Area Preset Counters	B, W
EEP	Area EEPROM	W

1.7.1 Memory area Variables V

The memory area "Variables V" is the retaining memory which is used by the program to store data of operations. It is composed by 200 locations, type word (therefore 100 double words). This area may be entered by operations on bit, word or double word. Number of double word still refers to the words structure, therefore to enter consecutive variables in double word the number must be put forward of 2 units. Values are reset at starting or at reset of PLC.



1.7.2 Area memoria special marker SM

This area is the retaining memory which contains all data used by the ladder program to interact with the hardware resources of EPL101. Some of these data are initialized at starting of PLC with default values described in the table below. This area contains all data related to analog inputs and also some bits controlled by the PLC for the ladder application as well as the settings for serial ports. The table below includes the content of each single location of the are “Special markers”, giving the address for Modbus protocol and the type of operation allowed for each location (R=reading, W=writing, R/W=reading/writing).

SM n°	Mod. word	Description/Meaning		
SM0	**2000**	**Status bit**		
		Bit 0	RUN/STOP bit (1=run). At starting this bit is always forced to ON, obtaining the RUN mode of PLC. In STOP mode, the relay outputs of PLC are disabled.	R/W
		Bit 1	This bit is always ON for first scanning cycle of main program. Ex. It is used to execute initialization sub-program.	R
		Bit 2	This bit makes available a clock impulse of 60 seconds (ON for 30 seconds, OFF for 30 seconds).	R
		Bit 3	This bit makes available a clock impulse of 1 second (ON for 0,5 second, OFF for 0,5 second)	R
		Bit 4	This bit is clock of scanning cycle, which is active ON for one cycle and desabled OFF for following cycle. It may be used as counting input for scanning cycle.	R
		Bit 5	Bit TEST. Setting this bit as ON, the reading of digital inputs is desabled. Status of inputs is read on words SM8 and SM9. Setting these words, it is possibile to execute debug/test of program simulating the starting.	R/W
		Bit 6	This bit is ON during the transmission of data to serial port COM1. It is automatically set to OFF at the end of transmission	R

		Bit 7	-	
		Bit 8	This bit is ON during the transmission of data to serial port COM2. It is automatically set to OFF at the end of transmission.	R
		Bit 9	If set to ON this bit enables mode “modem” for serial port COM1. This means that Timeout between one character and the other character in receiving mode is automatically set to 40mS.	R/W
		Bit 10	-	
		Bit 11	If set to ON this bit enables mode “modem” for serial port COM2. This means that Timeout between one character and the other character in receiving mode is automatically set to 40mS.	R/W
SM1	**2001**	**Diagnostic Bit (faults/anomalies)**		
		Bit 0	-	
		Bit 1	-	
		Bit 2	This bit is ON in case that the retaining data of the area EEPROM are lost.	R/W
		Bit 3	This bit is ON in case that calibration data are lost	R/W
		Bit 4	This bit is ON in case of CPU reset or of watch-dog intervention	R/W
		Bit 5	This bit is ON in in case of stack overflow on the area reserved to RAM	R/W
		Bit 6	This bit is ON in case of error during calibration proceeding.	R/W
		Bit 7	This bit is ON in case of error/fault of serial Eeprom.	R/W
		Bit 8	-	
		Bit 9	This bit is ON in case of error/fault of analog digital converter 16 bit	R/W
		Bit 10	This bit is ON in case of stack overflow of timed interrupts .	R/W
		Bit 12	ON in case of Analog input AI1 out of range.	R
		Bit 13	ON in case of Analog input AI2 out of range.	R
		Bit 14	ON in case of Analog input AI3 out of range.	R
		Bit 15	ON in case of Analog input AI4 out of range.	R

SM2	**2002**	**Bit for management of encoders 1 - 2**		
		Bit 0	Loading of counter for bidirectional encoder 1. Setting this bit to “1”, at the end of scanning cycle, the counter of encoder1 (32 bit) is loaded with the value which is set on special marker n°24 (most significant part) and on special marker n°25 (less significant part). The bit is automatically set to OFF at the end of operation.	R/W
		Bit 1	Loading of counter for bidirectional encoder 2. Setting this bit to “1”, at the end of scanning cycle, the counter of encoder 2 (32 bit) is loaded with the value which is set on special marker n°26 (most significant part) and on special marker n°27 (parte bassa). (less significant part). The bit is automatically set to OFF at the end of operation	R/W
SM3	**2003**	**Cycle time**		
		This word gives the time of last scanning cycle of the program (resolution 100 uS).		R
SM4	**2004**	**Minimum cycle time**		
		This word gives the minimum time of scanning cicle for noticed program (resolution 100 uS).		R
SM5	**2005**	**Max. cycle time**		
		This word gives the max. time of scanning cicle for noticed program (resolution 100 uS).		R
SM6	**2006**	**Interval of timed Interrupt no. 1**		
SM7	**2007**	**Interval of timed Interrupt no. 2**		
		These words define the interval for timed interrupts. Value of interval may be set between 1 and 100 ms (example: SM6=1 → 1 ms SM6=100 → 100 ms). For SM6 and SM7 values not included between 1 and 100, the default setting of relevant interrupt is 100 ms. At starting the values are fixed as 100 → 100 ms.		R/W
SM8	**2008**	**Status digital inputs I1÷I16 for test cycle**		
		This word gives the status of digital inputs during test (SM0.5=1). Each bit of these words refers to the status of one digital input, starting with less significant bit (SM8.0→I1, SM8.15→I16). This word is automatically set to zero at the starting of EPL101.		R/W

SM9	**2009**	**Min. value analog output AQ1**	
		The value of analog output) for which the Volt output must be 0,0V. This word is directly modified using the instruction RANGE(AQx,Min,Max). Automatically set to zero at starting.	R/W
SM10	**2010**	**Max value analog output AQ1**	
		The value of analog output AQ1) for which the volt output must be 5,0V. This word is directly modified using the instruction RANGE(AQ1,Min,Max). Automatically set to 500 at starting	R/W
SM11	**2011**	**Value of analog output AQ1**	
		*Values of this word definse the value of output AQ1. The tension value of output is given by the formule below AQ1(volt) = ((SM10-SM11)/(SM10-SM9))*5,0* *Setting the output value equals to minimum limit, output will be 0,0 volt; setting the value equals to max. limit, the output will be 5,0 volt.* *If output value is not included in the interval Minimum<Value<Maximum, the value for rating volt output is automatically reset within the minimum and maximum limit.* *Automatically set to zero at starting*	R/W
SM20	**2020**	**Countings of counter bidirectional encoder 1 (high area)**	
SM21	**2021**	**Countings of counter bidirectional encoder 1 (low area)**	
SM22	**2022**	**Countings of counter bidirectional encoder 2 (high area)**	
SM23	**2023**	**Countings of counter bidirectional encoder 2 (low area)**	
		These words contain the value of bidirectional conters for encoders 1 and 2. Counting is NOT STORED in case of power failure and it is automatically updated at each program scanning.	R
SM24	**2024**	**Loading value for counter of encoder 1 (high area)**	
SM25	**2025**	**Loading value for counter of encoder 1 (low area)**	
		These words contains the value (expressed as countings) which is loaded on counter of encoder 1 when loading bit SM2.0 is set to "1".	R/W
SM26	**2026**	**Loading value for counter of encoder 2 (high area)**	
SM27	**2027**	**Loading value for counter of encoder 2 (low area)**	
		Countings value loaded on counter of encoder 2 when loading bit SM2.1 is set to "1".	R/W

SM28	**2028**	**Countings per second of encoder 1**		
SM29	**2029**	**Countings per second of encoder 2**		
		Number of countings completed by the encoders during last second. These words are automatically updated every second		R
SM30	**2030**	**Countings per tenth of second for encoder 1**		
SM31	**2031**	**Countings per tenth of second for encoder 2**		
		Number of countings read by encoders during last 100 ms. These words are automatically updated every 100 ms..		R
SM32	**2032**	**Configuration COM1 in free-port mode**		
SM33	**2033**	**Configuration IR in free-port mode**		
SM34	**2034**	**Configuration COM2 in free-port mode**		
		These words enable the FREE-PORT mode for serial ports, setting also the relevant parameters. Enabling this mode, the protocol of serial communication is desabled with consequent direct access to transmission and receipt of data on port. These parameters are initialized at 0 on starting (free-port desabled).		R/W
		Bit 0÷3	Baud rate of serial port in free-port mode according to following values 0 → 4800 baud 6 → 110 baud 1 → 9600 baud 7 → 150 baud 2 → 19200 baud 8 → 300 baud 3 → 28800 baud 9 → 600 baud 4 → 38400 baud 10 → 1200 baud 5 → 57600 baud 11 → 2400 baud	R/W
		Bit 4÷7	Format of data for communication of serial port in free-port mode. 0 → 8,N,1 6 → 8,N,2 1 → 8,O,1 7 → 8,O,2 2 → 8,E,1 8 → 8,E,2 3 → 7,N,1 9 → 7,N,2 4 → 7,O,1 10 → 7,O,2 5 → 7,E,1 11 → 7,E,2	R/W
		Bit 8	Set this bit to “1” to enable mode free-port, or set it to “0” to restore standard mode for serial port, allowing communication according to the protocol selected in the program.	R/W

SM35	**2035**	**No. of characters on RX buffer of COM1**		
SM36	**2036**	**No. of characters on RX buffer of IR**		
SM37	**2037**	**No. of characters on RX buffer of COM2**		
		These words contain the number of valid character on RX buffer of each serial port. These words are used to check the no. of received characters in mode free-port. Any writing on these words will empty the RX buffer, setting the value at zero.		R/W
SM38	**2038**	**Filter on analog inputs *(default 5 averages)***		
		A filter may be applied to the signals of analog inputs, selecting the number of values to consider in the average for the rating of final input value. The filter can also be excluded for each input.		R/W
		Bit 0÷3	These bit set the number of values to calcolate the average for analog input. 1 → average of 1 value 2 → average of latest 2 values 3 → average of latest 3 values 4 → average of latest 4 values 5 → average of latest 5 values	R/W
		Bit 4	Exclude software filter for analog input 1. 0 → filter enabled 1 → filter excluded	R/W
		Bit 5	Exclude software filter for analog input 2 0 → filter enabled 1 → filter excluded	R/W
		Bit 6	Exclude software filter for analog input 3 0 → filter enabled 1 → filter excluded	R/W
		Bit 7	Exclude software filter for analog input 4 0 → filter enabled 1 → filter excluded	R/W
SM39	**2039**	**Filter on digital inputs *(default 10 ms)***		
		A filter may be applied to the signals of digital inputs entering a delay time. If the state of input changes, new state will be confirmed only if input will keep this state for the given time. Data will be confirmed after that filter will have eliminated noises and stabilized the inputs lines. *PLC accepts delay time values between 0 and 50 ms.*		R/W

SM40	**2040**	**Configuration Analog input AI1**	
SM41	**2041**	**Configuration Analog input AI2**	
SM42	**2042**	**Configuration Analog input AI3**	
SM43	**2043**	**Configuration Analog input AI4**	
		These special marker words define the type of sensor connected to analog inputs AI1, AI2, AI3 and AI4 (select properly the jumpers for configuration of inputs). 0 → Desabled 1 → Input 0÷1V 2 → Input 0÷20mV 3 → Input 0÷20mA (AI1 only) 4 → Input 4÷20mA (AI1 only) 5 → Thermocouple K (only AI1..AI3) 6 → Thermocouple S (only AI1..AI3) 7 → Thermocouple T (only AI1..AI3) 8 → Thermocouple R (only AI1..AI3) 9 → Thermocouple J (only AI1..AI3) 10 → Thermocouple E (only AI1..AI3) 11 → Thermocouple B (only AI1..AI3) 12 → Input PT100 13 → Input NI100 14 → Compensation PT100/NI (only AI2 + AI3) 15 → NTC-10KΩ β=3435	R/W
SM44	**2044**	**Min. value for Analog input AI1 linear**	
SM45	**2045**	**Min. value for Analog input AI2 linear**	
SM46	**2046**	**Min. value for Analog input AI3 linear**	
SM47	**2047**	**Min. value for Analog input AI4 linear**	
SM48	**2048**	**Max value for Analog input AI1 linear**	
SM49	**2049**	**Max value for Analog input AI2 linear**	
SM50	**2050**	**Max value for Analog input AI3 linear**	
SM51	**2051**	**Max value for Analog input AI4 linear**	
		Set the minimum and maximum numeric limits for analog conversion of inputs AI configured as current or tension. These words are modified using the instruction RANGE (AIx,Min,Max). Default settings at starting is 0 for minimum value and 1000 for maximum value.	R/W

SM52	**2052**	**Offset calibration Analog input AI1**	
SM53	**2053**	**Offset calibration Analog input AI2**	
SM54	**2054**	**Offset calibration Analog input AI3**	
SM55	**2055**	**Offset calibration Analog input AI4**	
SM56	**2056**	**Gain calibration Analog input AI1**	
SM57	**2057**	**Gain calibration Analog input AI2**	
SM58	**2058**	**Gain calibration Analog input AI3**	
SM59	**2059**	**Gain calibration Analog input AI4**	
		These words define the calibration of conversion for AI1, AI2, AI3 , AI4. They are used to correct eventual mistakes of reading. The formule is as follows: *Value AIx = Value AIx + (Value AIx * Gain calibration AIx) / 1000 Offset calibration AIx.* *At starting all calibration values are set to zero.*	R/W
SM60	**2060**	**Value Analog input AI1**	
SM61	**2061**	**Value Analog input AI2**	
SM62	**2062**	**Value Analog input AI3**	
SM63	**2063**	**Value Analog input AI4**	
		Numeric values of analog inputs AI obtained rating between minimum, maximum values, the conversion and the values of offset and gain.	R
SM64	**2064**	**Volt value for input I1**	
SM65	**2065**	**Volt value for input I2**	
SM66	**2066**	**Volt value for input I3**	
		Value of tension which is measured on digital inputs I1..I3. Range is 0..10V and it is available with two decimal points.	R
SM82	**2082**	**Offset of protocol address EPL101**	
		This value is added to the value obtained from the combination of jumpers used to select the address. Default setting at starting is 1.	R/W
SM83	**2083**	**Status of dip switches for address selection**	
		Value obtained by the combination of dip switches for the address selection.	R

SM84	**2084**	**Status of COM1**	
SM85	**2085**	**Status of IR**	
SM86	**2086**	**Status of COM2**	
		These words contain the status of communication serials COM1,IR,COM2. Each bit of each word notices the condition of failed communication (off line) or error for data sent or received by means of instructions COM_1÷16 (example: SM84.0=1 means error in the istruction COM_1(…)). If serial is configured with slave protocol, fault condition is noticed setting to "1" all bit of relevant word.	R
SM87	**2087**	**Baudrate COM1 (default 9600 baud)**	
SM91	**2091**	**Baudrate IR (default 1200 baud)**	
SM95	**2095**	**Baudrate COM2 (default 19200 baud)**	
		The value entered for this word defines the baudrate of serial port [1]*.* 0 → 4800 baud 1 → 9600 baud 2 → 19200 baud 3 → 28800 baud 4 → 38400 baud 5 → 57600 baud 6 → 110 baud 7 → 150 baud 8 → 300 baud 9 → 600 baud 10 → 1200 baud 11 → 2400 baud	R/W
SM88	**2088**	**Format of COM1**	
SM92	**2092**	**Format of IR**	
SM96	**2096**	**Format of COM2** [2]	
		Select format of communication data for serial port [3]*.* 0 → 8,N,1 (default at starting) 1 → 8,O,1 2 → 8,E,1 3 → 7,N,1 4 → 7,O,1 5 → 7,E,1 6 → 8,N,2 7 → 8,O,2 8 → 8,E,2 9 → 7,N,2 10 → 7,O,2 11 → 7,E,2	R/W

[1] In order to confirm and activate the changes, this word must be set during initialisation. In case that no change is made or it is made in other parts of the program, the baudrate will remain same as the default value of starting.

[2] Format cannot be modified (8,N,1).

[3] To activate the new settings, this word must be modified in the initialization function. In case that no change has been made or any change has been made in other parts of the program, format will remain the same as default at starting.

SM89	**2089**	**Answer delay/Waiting data on receipt COM1 *(default 20 ms)***	
SM97	**2097**	**Answer delay/Waiting data on receipt COM2 *(default 0 ms)***	
		The value of this word defines: • ***Slave protocol:*** *minimum delay between end of serial receipt from a master device and the start of answer broadcast from PLC (max 100ms).* • ***Master protocol:*** *max. waiting between start of polling broadcast from PLC and complete receipt of answer from slave device.* • *Value is expressed as ms.*	R/W
SM90	**2090**	**Number of errors for signal about status of COM1 *(default 10)***	
SM98	**2098**	**Number of errors for signal about status of COM2 *(default 10)***	
		Value entered for this word defines the number of consecutive communication errors after which the anomaly is notified in the relevant bit of “serial status”.	R/W
SM94	**2094**	**Number of characters RX on IR *(default 8)***	
		Number of characters for each data package received on IR port.	R/W

1.7.3 Memory area “digital inputs I”

This memory area “digital inputs I” is the area in which the state of digital inputs is stored. It is organized in words; each of the 16 bit of the word represents the state of an input. For instance: the state of digital input I20 is stored on bit no. 3 of word 2 in area I. The area consists of 4 words. The first and the second are updated with actual state of inputs at the starting of each cycle, while the last two words may contain the state of inputs read via serial communication from an expansion module.

1.7.4 Memory area “digital outputs Q”

This memory area stores the state of digital outputs. It is organized in words; each of the 16 bit of the word represents the state of an output. . For instance: the state of digital output Q1 is stored on bit no. 0 of word 1 in the area Q. The area consists of 8 words. The first one is transferred to the outputs of PLC at the end of each cycle, the other ones may contain the state of further outputs to write them via serial communication on an expanded module.

1.7.5 Memory area “support marker M”

The memory area M contains teh status of all markers (bit contacts) used in the program. It is organized in word; each of 16 bit of the word means the status of a marker. Example: status of marker M1 is stored on bit number 0 of word 1 in the area M. The area is composed of 8 words.

1.7.6 Area memoria “analog inputs AI”

In this memory area “analog inputs AI” the PLC stores the value which is assumed by analog inputs. The value is rated according to min. and max. limits which have been set as range of analog input.

1.7.7 Memory area ”timer T”

Timers are stored in this area. If timer is enabled, the value contained in this area will increase or decrease according to type of time, with the resolution chosen at the starting of timer.

1.7.8 Memory area “preset timer PT”

Memory area “preset timer PT” is the memory where the preset values of timers are stored.

1.7.9 Memory area “counters C”

This memory area contains values of counters. Accordng to type of counter, at each counting operation, the value container in this area will be updated.

1.7.10 Memory area preset values of counters PV

In this area the preset values of conters PV are stored.

1.7.11 Memory area EEPROM

EEPROM area is the self-retaining memory to store all data which cannot be lost even if PLC is switched-off for long periods (over 6 months). Data stored in this area are tested at the starting of PLC to check their integrity and in case of anomaly, the fault condition is notified (SM1.2) and the whole area is initialized at 0 (zero). Access and writing in this area requires longer time than any other area, therefore it is recommended not to use this area for frequent access, but only at starting to copy the stored data , for example in area V , and then to use these for quick access, guaranteeing a faster program.

N.B.: EEPROM memory allows a max number of writigs for each single location (1000000 are guaranteed). Above this number, data are not guaranteed, therefore it is recommended to avoid continuous writings in this area.

1.7.12 Memory area MMC

MMC area is the optional external memory which can be used to save big amounts of data which must be stored even by lack of power supply. Memory is Eeprom, therefore access to this area is slower than access to area V and SM. PLC does not check integrity of data saved in this area.

This area is divided in words (0÷32767) and can be entered also by Modbus protocol.

N.B.: MMC memory allows a max number of writigs for each single location (1000000 are guaranteed). Above this number, data are not guaranteed, therefore it is recommended to avoid continuous writings in this area.

1.7.13 Memory area COMx_SEND

Memory area COMx_SEND are used to load data which will be sent to the relevant serial port. They are used only in free-port mode. In standard mode these areas are handled with the selected communication protocol.

These ares are divided in byte (8 bit).

1.7.14 Memory area COMx_RECEIVE and IR_RECEIVE

Memory areas COMx_RECEIVE and IR_RECEIVE are used to save data received by the relevant serial port. They are used only in mode free-port. In standard mode these areas are handled with the selected communication protocol.
These ares are divided in byte (8 bit).

1.8 Protocol Modbus RTU slave

EPL101 is conceived for the use with SCADA systems or Operator panels via Modbus protocol RTU. Serial communication enables reading and modifying of data on available memory areas, entering and visualizing any data concerning the PLC. Module PL250A is provided with 2 serial communication ports which are enabled to operate as **slaves** with MODBUS protocol:

- COM1 - RS485 available on connector plug-8 and on pins
- COM2 - RS232 available on connector plug-8

Both serials support protocol modbus RTU as described below. Therefore the PLC may be connected and may communicate with 2 master devices at the same time.

Main features of protocol Modbus RTU	
Baud-rate	9600 bits/sec (default)
Format	8,N,1 (8 bit, no parity, 1 stop) (default)
Supported function	BITS READING (0x01, 0x02) WORDS READING (max 30 word) (0x03, 0x04) SINGLE BIT WRITING (0x05) SINGLE WORD WRITING (0x06) MULTIPLE BITS WRITING (0x0F) MULTIPLE WORDS WRITING (max 30 word) (0x10)
Error codes	ILLEGAL FUNCTION CODE (0x01) ILLEGAL DATA ADDRESS (0x02) ILLEGAL DATA VALUE (0x04)
Broadcast	Simultaneus writing to all connected slaves using address 0x00 and no answer by slaves.
Polling with unknown slave address	Polling using address 0xFF, any connected slave can answer

1.9 Addresses word/bit of EPL101 for protocol Modbus RTU

The following tables give all data (word and bit) which may be entered via Modbus protocol. For all data the table gives reading/writing elements and the value assumed at starting of PL250. According to initialization value at styarting, the following options are given:

1. "**ROM**" fixed value defined by program.
2. "**EEP**" value stored on Eeprom for 10years even in absence of power supply
3. "**TAMP**" value stored on Ram memory with buffer battery. These data are stored in absence of power supply for a limited time (approx. 4 months)
4. "**?**" value of these data is unknown at starting
5. Defined value. The value assumed at starting is the value defined in the table.

WORD			
MODBUS ADDRESS	**DESCRIPTION**	**READ/ WRITE**	**RESET VALUE**
0	Device type	R	ROM
1	Software release EPL101	R	ROM
2	Protocol activated on COM1	R	ROM
3	Protocol activated on IR	R	ROM
4	Protocol activated on COM2	R	ROM
5	Protocol address	R	TAMP
1000 ÷ 1199	Word area variables V	R/W	TAMP
2000 ÷ 2099	Word area special marker SM	R/W	TAMP
3000 ÷ 3095	Word area timer T	R/W	0
4000 ÷ 4095	Word area preset timer PT	R/W	0
5000 ÷ 5031	Word area counters C	R/W	0
6000 ÷ 6031	Word area preset counters PV	R/W	0
7000 ÷ 7126	Word area EEPROM	R/W	EEP

WORD			
10000 ÷ 10031	Word % prop/integr/deriv/outputs PID	R	
10000	% proportional action PID1		0
10001	% integral action PID1		TAMP
10002	% derivative action PID1		TAMP
10003	% output PID1		TAMP
10004	% proportional action PID2		0
…	…		…
10031	% output PID8		TAMP
32768 ÷ 65535	Word area MMC	R/W	EEP
100	N.O. contacts digital inputs I1÷I16	R	?
101	N.O. contacts digital inputs I17÷I32	R	?
102	N.O. contacts digital inputs I33÷I48	R	0
103	N.O. contacts digital inputs I49÷I64	R	0
110	N.O. contacts digital outputs Q1÷Q16	R	0
111	N.O. contacts digital outputs Q17÷Q32	R	0
112	N.O. contacts digital outputs Q33÷Q48	R	0
113	N.O. contacts digital outputs Q49÷Q64	R	0
114	N.O. contacts digital outputs Q65÷Q80	R	0
115	N.O. contacts digital outputs Q81÷Q96	R	0
116	N.O. contacts digital outputs Q97÷Q112	R	0
117	N.O. contacts digital outputs Q113÷Q128	R	0
120	N.O. contacts bistable relays B1÷B16	R/W	0
121	N.O. contacts bistable relays B17÷B32	R/W	0
122	N.O. contacts bistable relays B33÷B48	R/W	0
123	N.O. contacts bistable relays B49÷B64	R/W	0
130	N.O. contacts timer T1÷T16	R	0
131	N.O. contacts timer T17÷T32	R	0
132	N.O. contacts timer T33÷T48	R	0
133	N.O. contacts timer T49÷T64	R	0
134	N.O. contacts timer T65÷T80	R	0
135	N.O. contacts timer T81÷T96	R	0
140	N.O. contacts timer C1÷C16	R	0
141	N.O. contacts timer C17÷C32	R	0
170	N.O. contacts support markers M1÷M16	R	0
171	N.O. contacts support markers M17÷M32	R	0
172	N.O. contacts support markers M33÷M48	R	0
173	N.O. contacts support markers M49÷M64	R	0
174	N.O. contacts support markers M65÷M80	R	0

WORD			
175	N.O. contacts support markers M81÷M96	R	0
176	N.O. contacts support markers M97÷M112	R	0
177	N.O. contacts support markers M113÷M128	R	0

BIT			
MODBUS ADDRES S	**DESCRIPTION**	**READ/ WRITE**	**RESET VALUE**
1600 ÷ 1663	N.O. contact digital input I1 ÷ N.O. contact digital input I64	R/W	?
1760 ÷ 1887	N.O. contact digital output Q1 ÷ N.O. contact digital output Q128	R/W	0
1920 ÷ 1983	N.O. contact bistable relay B1 ÷ N.O. contact bistable relay B64	R/W	0
2080 ÷ 2175	N.O. contact timer T1 ÷ N.O. contact timer T96	R	0
2240 ÷ 2271	N.O. contact counter C1 ÷ N.O. contact counter C32	R	0
2720 ÷ 2847	N.O. contact support marker M1 ÷ N.O. contact support marker M128	R/W	0
16000 ÷ 19199	Bit 0 area marker V0 ÷ Bit 15 area marker V199	R/W	TAMP
32000 ÷ 33599	Bit 0 area special marker SM0 ÷ Bit 15 area special marker SM99	R/W	TAMP

2 Ladder programming of EPL101

2.1 Introduction

The software tool PLProg for Windows allows to program the EPL101. The ladder programming enables to draw the logic diagram of the application and to store it on the PLC.

2.2 Elements of Ladder programming

Available elements with relevant features are listed below.

2.2.1 Contacts digital inputs I

Contacts I contains the state of digital inputs of PLC and eventually of one more expansion. The N.O. contact is closed (on) when bit value is 1 (active input). The N.C. contact is open (on) when bit value is 0 (input not active).

2.2.2 Relay outputs/auxiliary Q

EPL101 is provided with 128 outputs type “Q”, each composed of one coil and the relevant logic contact N.O. and N.C. which may be used to draw the ladder diagram. Hardware is provided with only 6 “physical” relay outputs Q, the other ones are available as auxiliary relays
Energizing coil “Q” , the relevant logic contact will close (N.O.) or open (N.C.). The contacts of physical outputs are all N.O. and at starting all contacts N.O. are open.

2.2.3 Bistable relays B

64 bistable relays are available. Each consists of one coil and the relevant logic contact N.O. and N.C. .
Energizing coil “B”, the relevant logic contact will change its state (it opens if previously closed, it closes if previously open). The N.O. contact is closed (on) when bit value is 1. The N.C. contact is open (on) when bit value is 0. At starting of PLC the N.O. contact is open.

2.2.4 Timer T

Three different operating modes are available for Timers:

- **TON**. Mode **Start timing as delay at insertion** counts time when coil is active (ON). Timing bit (contact T) is activated when actual value (T) is bigger or equals to preset time (PT). When coil is deactivated (OFF), actual value of timer “delay at insertion” is reset. Timer keeps on counting after reaching preset value and it stops reaching the max. value 32767.
- **TOFF**. Mode **Start timing as delay at disconnection** allows to delay the deactivation of an output for a certain time after that the input has been deactivated. When coil is activated, the timing bit (contact T) is immediately activated and the actual value (T) is set to zero. At deactivation of coil, the timer counts until elapsed time is equals to preset timer (PT). After reaching the preset timer, timing bit is deactivated and actual value does not increase. If input is deactivated for a time which is lower than preset time, the timing bit is still active. To start counting, the function TOF must notice a transition from activated to deactivated (ON → OFF).
- **TONR**. Mode **Start timing as delay at insertion with memory** counts time when coil is activated (ON). Timing bit (contact T) is activated when actual value (T) is bigger or equals to preset time (PT). When coil is deactivated (OFF), actual value of timer “delay at insertion with memory” is stored. This value allows to accumulate time for more activation peroids of the coil. Actual value of timer can be reset with operation MOV(Tx = #0). Timer keeps on counting after reaching preset value and it stops reaching the max. value 32767.

Timers with operating modes TON, TONR and TOF are available in three different resolutions, not depending from numer of timer; they can be activated with time basis 10 ms, 100ms and 1s. Each counting of actual value is multiple of time base. Examplea counting of 50 in a timer with time base 10 ms is equals to 500 ms.
Preset time (PT) can be directly loaded with a value or by means of a variable in the area VW, SMW, AI , TR.

2.2.5 Counters C

Counters are available with two operating modes:

- **MUP**. In mode **Count Up** the counting bit (contact C) is activated when present value (C) is >= to preset value (PV). Counter counts up every time that the counting-up input Cx(UP) goes from Off to On and it counts down every time that the counting-down inputs Cx(DOWN) goes from Off to On.

The counter is reset when the reset input Cx(RESET) is activated or when the operation MOV(Cx = #0) is executed. At reaching of max. value

(32.767), the next edge-up of the counting-up input will keep unchanged the actual value. Similarly at reaching of minimum value (-32.768) the next edge-up of the counting-down input will keep unchanged the actual value. The up-counters have an actual value which keeps the actual counting (T). They also have a preset value (PV) which is compared to the actual value at the end of each program cycle. If actual value is bigger or equal to preset value, the counting bit is activated (contact C), otherwise it is deactivated. Please use number of counter to refer both to actual value and to Contact C of the counter itself.

- **MDOWN**. In mode **Count Down,** the counting bit (contact C) is activated when present value is equals to zero. The counter counts down starting from a preset value (PV) on the edge-up of the counting-down input Cx(DOWN) and it counts up on the edge-up of the counting-up input Cx(UP). At reaching of max. value (32.767), the next edge-up of the counting-up input will keep unchanged the present value. The counter resets the counting bit (contact C) and load present value with preset value (PV) when loading input Cx(RESET) is activated. The counter in mode count-down stops counting when it reaches zero. Please use number of counter to refer both to actual value and to Contact C of the counter itself

Preset value (PV) may be directly loaded with a value or it may be loaded by one the variables in the area VW, SMW, AI, TR.

2.2.6 Function Math formules FM

The function FM allows to perform math operations (+, -, *, /, |, &, ^, <<, >>) between two operators and to save the result in another memory location. The operators can be numeric or they may also refers to available memory areas.

2.2.7 Assignement Function MOV

The function MOV allows to assign to the specified memory location a numeric value or a value assumed by another memory location.

2.2.8 Assignement Function BLKMOV

The function BLKMOV allows to assign to the memory block a numeric value or the value assumed by another block of memory locations

2.2.9 Indexed Assignement Function MOVIND

The indexed assignement function MOVIND allows to assign to a certain memory location specified by another memory location a numeric value or the value assumed by another memory location selected in the area specified by a memory location which is the index. This type of assignement allows to consider the memory areas as vectors of a certain number of locations: by means of the value assumed by another location called “index” it is possible to enter the value of the area n=0, n=1, , n=N-1

2.2.10 Assignement function MOVTXT

Starting with the specified memory location, the assignement function MOVTXT allows to save the characters of a string sent as parameter to the function. Following types of format are available for characters of string in the memory area:

- ONE_CHARACTER_PER_WORD : in this format each word of destination area will contain only one character of string
- TWO_CHARACTERS_PER_WORD in this format each word of destination area will contain two characters of string, starting with high area of the word
- ONE_ CHARACTER_7_SEGMENTS_PER_WORD

2.2.11 Contacts II immediate digital inputs

Contacts II allow to read immediately the state of digital input. The N.O. contact is closed (On) when bit value is 1 (active input). The N.C. contact is open (On) when bit value is 0 (deactivated input).

2.2.12 Immediate outputs QI

By means of direct access to outputs QI, PLC allows to directly work on outputs Q during the ladder program execution, even without waiting the end of program. The command is allowed only on hardware outputs of PLC (QI1..QI12).

2.2.13 Contact IF

The operation “conditional comparison IF” allows to compare the values of 2 variables in any memory area. Following comparisons are allowed:
=, >=, <=, >, <, <>. Contact is active when comparison is true.

2.2.14 Functions SBIT and RBIT

Function SBIT sets to “1” a bit of a memory area when the coil of function is activated.
Function RBIT sets to “0” a bit of a memory area when the coil of function is activated.
Number of bit is 0 to 15, bit 0 is the less significant bit (LSB).

2.2.15 Contact BIT

This operation rates the value of a bit in a memory area. N.O contact is closed (On) when bit value is 1. N.C. contact is open (On) when bit value is 0. Number of bit is 0 to 15, bit 0 is the less significant bit (LSB).

2.2.16 Function RANGE

The RANGE fuction defines minimum and maximum limits for analog inputs AI and for PID outputs.

Concerning analog inputs AI, the minimum and maximum values allow to traslate the countings value of the analog-digital conversion into a value which can be used inside the program. Below an example:

RANGE(AI1, Min 10, Max 200)

For analog input AI1 the function defines the minimum limit as 10 and the maximum limit as 200. In case that analog input AI1 would be connected to a potentiometer to fix a preset time (PT) of a timer with time-base 100ms, the result would be a variable time between 1.0 and 20.0 seconds, accordino to the position of potentiometer.
Concerning PID outputs, minimum and maximum values allow to rate the value for the output of PID control algorithm. Below an example:

RANGE(PID1, Min 100, Max 500)

For PID1 output , minimum limit is fixed as 0 and maximum limit is 500. This means that for output 0% the PID output will be equals to minimum fixed value and for output 100% the PID output will be equals to maximum fixed value. For each PID [1..8], minimum and maximum values of output are initialized at starting as 0 and 10000.

2.2.17 Contact NOT

Contact NOT modifies the state of current flow. Current flow stops if it reaches contact NOT and it works if it does not reach contact NOT. The operation NOT modifies logic value from 0 to 1 or from 1 to 0.

2.2.18 Contact P and N

The contact “positive transition P” activates the current flow for a scanning cycle at each transition Off / On. The contact “negative transition N” activates the current flow for a scanning cycle at each transition On / Off. When the operation “positive transition P” detects a transition of logic value from 0 to 1, it sets this value to 1, otherwise to 0. When the operation “negative transition N” detects a transition of logic value from 1 to 0, it sets this value to 1 , otherwise to 0.

2.2.19 Function SEND and Free-port mode

Function SEND allows to activate data transmission by means of serial ports in mode free-port. In this mode, which can be activated by special marker SM32, SM33 and SM34, the protocol which usually handles the serial ports is desabled and the ports as well as the relevant TXT and RX buffers are controlled by the ladder program. After loading on buffer the data to send and activating function SEND (which uses serial port and no. of characters to send as parameters) these data will be sent on serial line. During the transmission of data, bit SM0.7, SM0.7 or SM0.8 (according to transmission port) will be set to “1”, while it will be set to “0” at the end of transmission. It is possible to check the answer of a connected device by means of SM35, SM36 and SM37, which contain the number of characters received and saved on RX buffer of each serial port. Any writing on each of these special marker will empty the RX buffer. Calls to function SEND before the end of previous transmission or with mode free-port desabled will be ignored by the program.

2.2.21 Serial communication function COM

Communication functions COM enables the programming of the serial ports (COM1-RS485) for reading/writing of data from connected slave devices using the master protocol selected in the project. These functions are active only when in the project a master communication protocol has been selected for the relevant serial port. A Master protocol means a protocol which enables PLC to control the serial line communication, controlling the data flow towards slave devices. Instruction COM operates with interface RS485, allowing to connect more devices on the same line. Instructions are active until the relevant coil is active. Consider also that, according to communication protocol, the time required for data upgrade can be quite different and read data are not immediately available at activation of coil, but only after a certain time depending from communication delays.
Instruction COM requires the configuration of following parameters:

- Index (max. 16 different serial pollings)
- Type of operation:
- Reading: PLC will read continuously data from slave device and will store them in an internal memory area
- Writing: PLC will write continuosly data on an internal memory area of the slave device
- Reading/Writing: PLC will read data on slave device and will store them in an internal memory area; when these internal data on PLC will be modified by the program, any change will be automatically sent also to slave device by means of a writing instruction (this instruction can work only on a single data each time).
- Slave number (communication address of slave device)
- Data type (word or bit)
- Number of data (or starting number in case of more data)
- Internal memory area of PLC where data must be read/written
- Word number (the same reading / writing instruction can work simoultaneously on more consecutive data

2.2.22 Protocol for REMOTE CONTROLLER on IR PORT

The protocol for management of data on infrared port (for data received from remote switch) is checking that data on RX buffer are correct. After correct receipt of data from remote control, it notices new data to special marker SM85II.
Remote controller sends two strings: one for ON/OFF of the plant and one for the programming. Bit SM85.1 is set to 1 anytime that data are correctly received on serial port IR. Bit SM85.0 is set to 1 anytime that data are correctly received and with same address of the machine which received it. Lenght of sdata strings sent by Remote controller is always 8 byte. First byte (0xAA) is a symchronous character (start od string) , second byte is the address of Slave (with offset 100). Additional 4 data byte follow, and finally 2 byte are added by the checksum. The 4 data byte are all set to 0x65 in case of on-off, or they contain the value to program (still with offset0x65) if programmino instruction is coming from remote controller. After checking the data on RX buffer, it is necessary to reset manually word SM85. Protocol accepts max one correct data string each second.

2.2.23 Functions StartPID , PID , SetOutPID

Functions StartPID, PID and SetOutPID enables control of a process by means of PID algorithm (proportional, integral, derivative).

Function StartPID starts the relevant control block and sets parameters as required. Function can be activated once at the starting or it may be recalled later to modify quickly control parameters. PID integral action is initialized only calling this function and setting integral time to 0, otherwise even in case of switch-off the system will start control action keeping the same percentage of integral action and consequently limiting the transient times. Parameters required by function StartPID are the following:

- Proportional band
- Integral time
- Derivative time
- Dead band

Parameters can be entered with numeric format or referring to internal variables. Integral time is expressed in time unit used for PID function (ex. PID function recalled every 1 sec., integral time expressed in seconds). Derivative time is expressed with one decimal digit more than integral time. Proportional band and dead band are expressed with a numeric value as setpoint and process.

PID function requires following parameters:

- Setpoint
- Process
- Output value
- Type of control action
- Type of output

After acquisition of necessary parameters values, PID function will enter the value obtained by control algorithm in the variable “Output value”. This value will have been obtained rescheduling the percentage value 0 – 10000 (0.00% ÷100.00%) between minimum and maximum value of PID output as entered using the function RANGE.

For optimal results, PID function should be recalled at regular intervals. A timer can be used or even an internal Interrupt, to get shorter intervals.

Function SetOutPID is used for control actions which foresee both automatical and manual functions. The function allows to avoid process oscillations when changing from manual to automatical PID control. Following parameters are required:

- Output value

It allows to set the value of PID output, automatically rating the single percentages of proportional and integral action. Changing from manual to automatical

operation, PID output will assume the value entered in manual function and it will start control action.
Consequently the function must be used only in manual control, to keep PID output in line with value of manual control.
This function automatically cancels derivative action.
If the function is used with process value outside proportional band, integral action will be set to 0.

2.2.24 Function GENSET

Function GENSET allows to generate automatically a variable setpoint, with option to enter an acceleration and deceleration ramp. Function GENSET operates on a series of adjoining variables in double word, starting with the location which is given as parameter of the function.

Address area VD	Contents
+0	State of function GENSET 0 → Stop or end motion 1 → Function initialization 2 → Acceleration ramp 3 → Motion at constant speed 4 → Deceleration ramp
+2	Starting Setpoint / Setpoint rated at end of GENSET function (countings)
+4	Final setpoint (countings)
+6	Motion speed (countings * 1000 / time units)
+8	Duration of acceleration ramp (time units)
+10	Duration of deceleration ramp (time units)
+12	Instant speed of setpoint (countings * 1000 / time units)

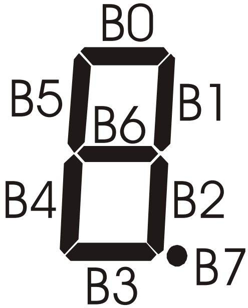
To use this function, proceed as follows:

- Enter on location VD+2 the starting setpoint
- Enter on location VD+4 the final setpoint
- Enter on location VD+6 max. motion speed as countings*1000/time units (to have 3 decimals. Example: setting 12345 will give a speed of 12.345 countings/time units).
- Enter on location VD+8 the duration of accelaration ramp (expressed as time units; if duration of acceleration ramp must be 1 second and function GENSET is recalled by an interrupt of 1 ms, enter 1000)
- Enter on location VD+10 the duration of deceleration ramp.

- Write “1” on location VD to start the function which will automatically start to write on location VD+2 the generated setpoint. Location VD will also be updated with current state of function, while location VD+12will be updated with instant speed of setpoint, expressed with 3 decimals. This value may be used to generate action “F” for function POSPID.
- At end of motion, when location VD+2 will reach value of final setpoint, the function will automatically enter standby modde, indicated by value “0” on location VD. This means that function GENSET can be always active, even when motion is not required

2.2.25 Function CONV

The function CONV provides the conversion of source data in one of the available formats. Conversion type “TO_7SEG_SIGNED” converts input data (one word with sign -32768..32767) in a specified number of digits alredy suitable for a display with 7 segments. The number of digits to convert, starting from less significant digit, will be sent to the function as parameter.
The codes will be saved (one digit for each word) starting from destination word and then in the following words accordino to the required number of digits.
Coversion type “TO_7SEG_UNSIGNED” is similar to the above described conversion. The difference is that the source data is considered as unsigned word (0..65535). the code is composed by one bit set to 1 if the segment must be ON and by one bit set to if the segment must be Off. The correspondance between bit and segment is the following:



Conversion type “TO_ASCII_SIGNED” converts the entering data (one word with sign -32768..32767) in a specified number of ascii characters. The number of characters to save will be assigned to this function as parameter. Codes of these characters will be saved (one character per word) starting with the destination word and then in the following ones, according to the no. of required characters.

Conversion type “TO_ASCII_UNSIGNED” is similar to the one described above, but the source data is meant as unsigned word (0..65535).

PIXSYS
Via Po, 16
30030 Mellaredo di Pianiga (VE)
www.pixsys.net
e-mail: sales@pixsys.net - support@pixsys.net

Software Rev. 1.13 (firmware)

2300.10.204-RevI 090516