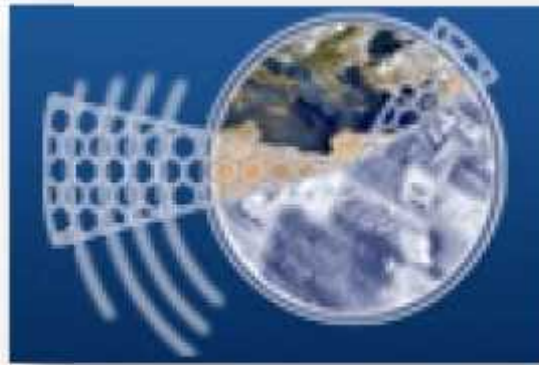




μ

μ

μ μ



:

μ μ μ , μ μ  
μ μ (PLC) μ open source μ  
μ .

- 
- 

:

:

μ

:1587

, μ 2014

μ μ , μ , μ μ

μ μ μ ο μ , . μ .  
μ μ , μ .

## **Abstract**

The automated monitoring and control of the manufacturing process of a product requires systems with increased capabilities. One way to check is visual. The thesis develops autonomous control and monitoring system objects along a path. The tracking of the object is dynamic and follow the object along the route. This was achieved by using a system of sensors that drive the mechanism of the camera in the direction of the object. The control of the system is through a desktop application. The hardware I used was Siemens PLC S7-300 which was already installed in the Industrial Automation Laboratory (ENPET) and for the creation of the desktop application, the Microsoft Visual Studio 2010 with the particular language, Visual Basic. The demonstration of the thesis was to model depicting the route which follows the object and the operation of the automation.

# Σύνοψη

Το παρόν έγγραφο περιγράφει τη διαδικασία ανάπτυξης και δοκιμής ενός προγράμματος αυτοματισμού. Η διαδικασία ξεκινά με τον ορισμό των απαιτήσεων και την επιλογή του κατάλληλου υλικού και λογισμικού. Τα χρησιμοποιούμενα εργαλεία είναι:

- Visual Studio 2010 (Microsoft)
- Hardware, PLC S7-300 (Siemens)
- Visual Basic



..... μ , μ μ , μ μ

μ	.....	1
μ	.....	1
μ μ	.....	1
	.....	2
Abstract.....		3
	.....	4
1	.....	9
	.....	9
1.1	.....	9
1.2	.....	9
1.3	.....	9
1.4	.....	9
2.....		10
PLC	.....	10
2.1 PLC: μ μ	.....	10
2.2 μ PLC	.....	12
2.3 μ	.....	13
2.3.1 Industrial Layers ( μ )	.....	13
2.3.2 Fieldbus	.....	14
2.3.3 μ Fieldbus	.....	15
2.3.4 AS-Interface.....		16
2.3.5 PROFIBUS - PROFINET.....		17
2.3.6 INDUSTRIAL ETHERNET.....		19
2.3.7 INDUSTRIAL ETHERNET	.....	20
2.3.8	.....	21
3	.....	22
PLC	.....	22
3.1 PLC	.....	22
3.2 (RACK)	.....	24
3.3 PS (Power Supply)	.....	26
3.4 (CPU)	.....	27
3.4.1 CPU	.....	28
3.4.2 CPU	.....	30
3.5 PLC (Digital Input)	.....	31
3.6 PLC (Analog Input)	.....	32
3.7 PLC (Digital Output)	.....	33
3.8 PLC (Analog Output)	.....	34

	μ	μ	,	μ		, μ		μ
		4	.....					35
4.1		Simatic Manager Step 7	.....					35
4.2	μ	μμ	.....					36
4.3		μ Block	.....					38
4.4	μ	Block	.....					41
4.5		STEP 7	.....					43
4.6	μ	μ PLC	.....					45
4.7		μ Project	.....					46
4.7.1		μ	.....					48
4.7.2	μ	– Hardware Configuration	.....					49
4.7.3	μ	Block	.....					55
4.8	μ	μ LAD-STL-FBD	.....					56
4.8.1		μμ PLC	.....					60
4.9		OB1-FC-DB1	.....					61
		5	.....					80
			.....					80
6.1		Visual Basic	.....					80
6.2		Desktop Application	.....					82
6.3		VB	.....					84
		7	.....					94
7.1			.....					94
7.2			.....					95
			.....					96

Εικόνα 1: PLC SIMATIC S7-300.....	10
Εικόνα 2: Πίνακας ελέγχου με ρελέ (1965) και Πινάκας ελέγχου με PLC.....	11
Εικόνα 3: Βιομηχανικά Επίπεδα Αυτοματισμού.....	13
Εικόνα 4: Διασύνδεση Fieldbus.....	14
Εικόνα 5: AS-I protocol.....	16
Εικόνα 6: Διασύνδεση σε δίκτυο PROFIBUS.....	17
Εικόνα 7: Διασύνδεση σε δίκτυο PROFINET.....	18
Εικόνα 8: Διασύνδεση Industrial Ethernet.....	19
Εικόνα 9: Βασικά χαρακτηριστικά των Βιομηχανικών Δικτύων.....	21
Εικόνα 10 :Δομή PLC S7-300.....	22
Εικόνα 11: Στοιχεία PLC.....	23
Εικόνα 12: RACK.....	24
Εικόνα 13: Επέκταση Module S7-300.....	25
Εικόνα 14: PS.....	26
Εικόνα 15: PS 307: 5A Wiring Diagram.....	26
Εικόνα 16: CPU.....	27
Εικόνα 17: Μνήμη ρολογιού.....	29
Εικόνα 18: CPU 314C-2 DP.....	30
Εικόνα 19: Digital Input.....	31
Εικόνα 20: Analog Input.....	32
Εικόνα 21: Digital Output.....	33
Εικόνα 22: Analog Output.....	34
Εικόνα 23: Γραφικό Περιβάλλον Simatic Manager.....	35
Εικόνα 24: Γραμμικό Πρόγραμμα.....	36
Εικόνα 25: Τμηματοποιημένο Πρόγραμμα.....	37
Εικόνα 26: Δομημένο Πρόγραμμα.....	37
Εικόνα 27: Κυκλική επεξεργασία προγράμματος μιας CPU.....	38
Εικόνα 28: Πίνακας με τα διαθέσιμα block και την προτεραιότητα τους.....	39
Εικόνα 29: Δομή της FC1 μου.....	41
Εικόνα 30: Δομή του DB1 μου.....	42
Εικόνα 31: Διάγραμμα σύνδεσης Hardware και Software.....	43
Εικόνα 32: Set PG/PC Interface.....	45
Εικόνα 33: Επιλογή δημιουργίας νέου project.....	46
Εικόνα 34: Δημιουργία νέου project.....	47
Εικόνα 36: Hardware.....	49
Εικόνα 38: Rail.....	50
Εικόνα 39: Ιδιότητες του CP.....	51
Εικόνα 40: Ορισμός της IP.....	51
Εικόνα 41: Έλεγχος HW Config.....	52
Εικόνα 42: Σωστό HW Config.....	52
Εικόνα 43: Εύρεση Netpro.....	53
Εικόνα 44: Netpro.....	53
Εικόνα 45: Save and Compile.....	54
Εικόνα 46: Download Hardware.....	54
Εικόνα 47: Project Blocks.....	55
Εικόνα 48: Εισαγωγή Block.....	55
Εικόνα 49: Γλώσσα LAD.....	56
Εικόνα 50: Γλώσσα STL.....	57
Εικόνα 51: Γλώσσα FBD.....	57
Εικόνα 52: Αλλαγή γλώσσας προγραμματισμού.....	58
Εικόνα 53: Αποθήκευση προγράμματος.....	58
Εικόνα 54: Τύποι μεταβλητών.....	59
Εικόνα 55: PLC Download.....	60
Εικόνα 56: Απόδοση τιμής για χειροκίνητη δεξιά κίνηση # 1-2.....	61

Εικόνα 57: Απόδοση τιμής για χειροκίνητη δεξιά κίνηση # 3-4-5.....	62
Εικόνα 58: Απόδοση τιμής για χειροκίνητη δεξιά κίνηση # 6 .....	62
Εικόνα 59: Απόδοση τιμής για χειροκίνητη αριστερή κίνηση # 1-2.....	63
Εικόνα 60: Απόδοση τιμής για χειροκίνητη αριστερή κίνηση # 5-6.....	64
Εικόνα 61: Δήλωση FC1 και FC2 .....	65
Εικόνα 62: Αυτόματη δεξιά κίνηση #1 .....	66
Εικόνα 63: Αυτόματη δεξιά κίνηση #2 .....	66
Εικόνα 64: Αυτόματη δεξιά κίνηση #3 .....	67
Εικόνα 65: Αυτόματη δεξιά κίνηση #4 .....	67
Εικόνα 65: Αυτόματη δεξιά κίνηση #5 .....	68
Εικόνα 66: Αυτόματη δεξιά κίνηση #6 .....	68
Εικόνα 67: Αυτόματη αριστερή κίνηση #1 .....	69
Εικόνα 69: Αυτόματη αριστερή κίνηση #3 .....	70
Εικόνα 70: Αυτόματη αριστερή κίνηση #4 .....	70
Εικόνα 71: Αυτόματη αριστερή κίνηση #5 .....	71
Εικόνα 72: Αυτόματη αριστερή κίνηση #6 .....	71
Εικόνα 73: Ορισμός της FC1 .....	72
Εικόνα 74: Ορισμός στη FC2 για χειροκίνητη δεξιά κίνηση #1 .....	73
Εικόνα 75: Ορισμός στη FC2 για χειροκίνητη δεξιά κίνηση #2 .....	74
Εικόνα 76: Ορισμός στη FC2 για χειροκίνητη δεξιά κίνηση #3 .....	75
Εικόνα 76: Ορισμός στη FC2 για χειροκίνητη αριστερή κίνηση #1.....	76
Εικόνα 77: Ορισμός στη FC2 για χειροκίνητη αριστερή κίνηση #2.....	77
Εικόνα 78: Ορισμός στη FC2 για χειροκίνητη αριστερή κίνηση #3.....	78
Εικόνα 79: Περιεχόμενα του DB.....	79
Εικόνα 80: Συνέχεια των περιεχομένων του DB .....	79
Εικόνα 81: Εύρεση του project στη VB.....	80
Εικόνα 82: Συνέχεια Εύρεσης του project.....	81
Εικόνα 83: Άνοιγμα project .....	81
Εικόνα 84: VideoCapture.vb[Design] .....	82
Εικόνα 85: VideoCapture.vb[Code] .....	82
Εικόνα 86: Application, CAMERA .....	83





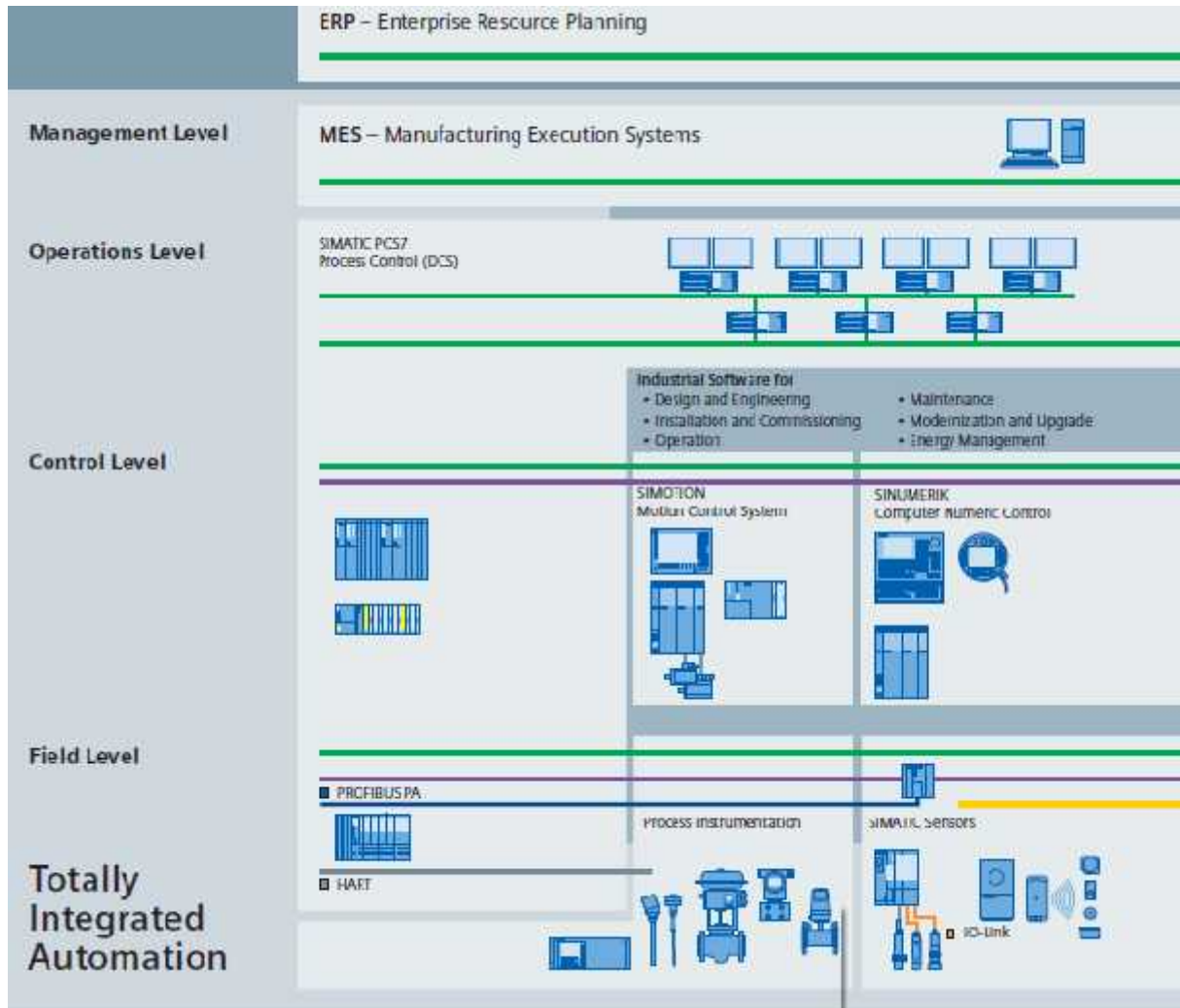






## 2.3 μ

### 2.3.1 Industrial Layers ( μ )

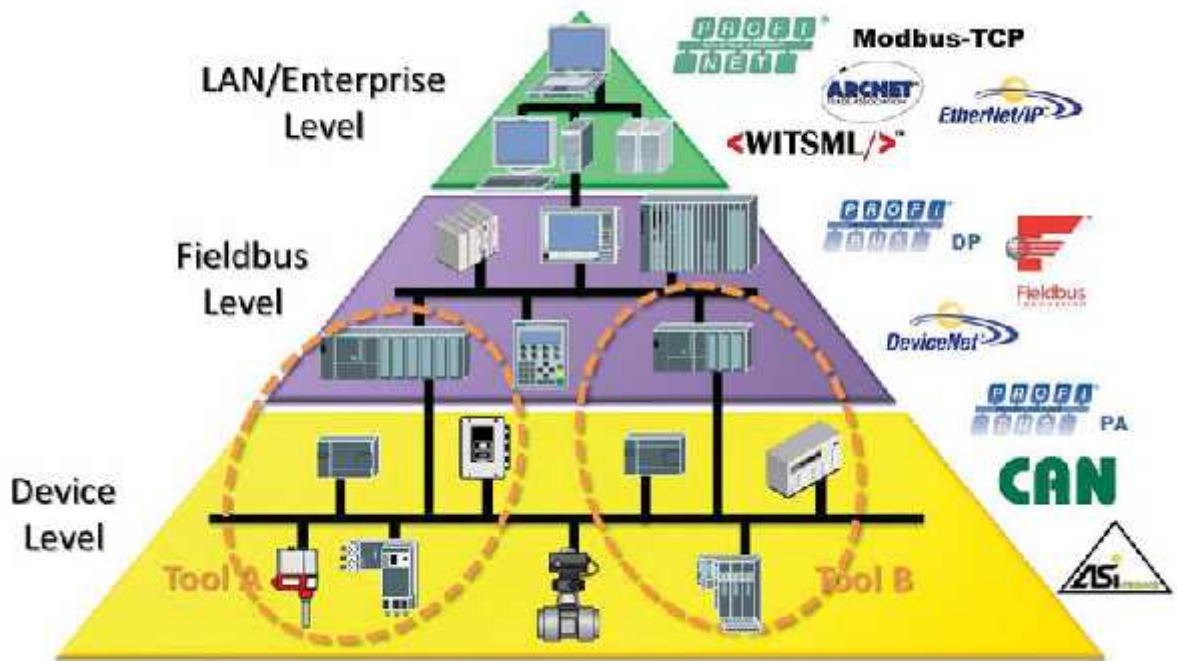


Εικόνα 3: Βιομηχανικά Επίπεδα Αυτοματισμού

- **Field Level:** \_\_\_\_\_ μ .
- **Control Level:** \_\_\_\_\_ μ (0,1) .
- **Operation Level:** \_\_\_\_\_ μ μ .
- **Management Level:** \_\_\_\_\_ μ μ .

## 2.3.2 Fieldbus

Fieldbus, μ, μ, multidrop, μ, μ (field devices), μ « », μ Fieldbus.



Εικόνα 4: Διασύνδεση Fieldbus



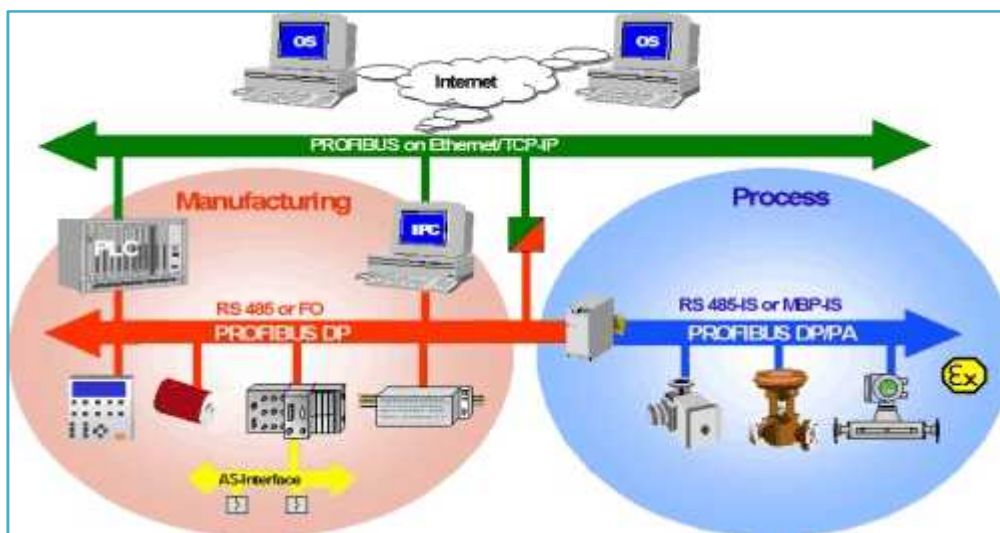




## 2.3.5 PROFIBUS - PROFINET

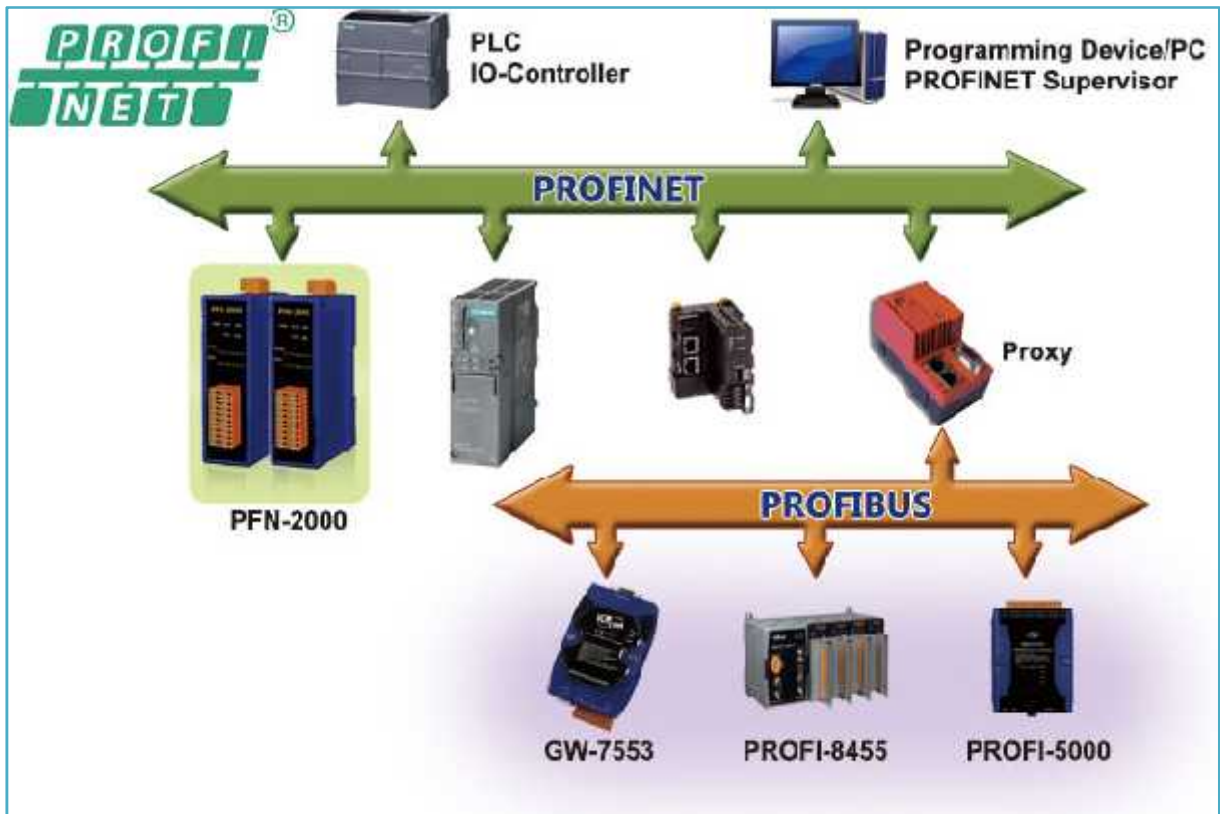
PROFIBUS is a Fieldbus project. It was developed in 1989 by the PROFIBUS project.

- PROFIBUS-FMS** (Fieldbus Message Specification) is a Client-Server protocol.
- PROFIBUS-DP** (Decentralised Periphery) is used for connecting field devices to PLCs. It supports data rates up to 12 Mbps and cycle times down to 2ms. It uses a master/slave (token) communication method.
- PROFIBUS-PA** (Process Automation) is used for connecting field devices to PLCs. It supports data rates up to 31.25 Kbps and cycle times down to 9500 μs. It uses a client/server communication method.



Εικόνα 6: Διασύνδεση σε δίκτυο PROFIBUS

PROFINET, Profibus  
 Ethernet IO (Input- Output) PLC. PROFIBUS  
 (bus), PROFIBUS-PA

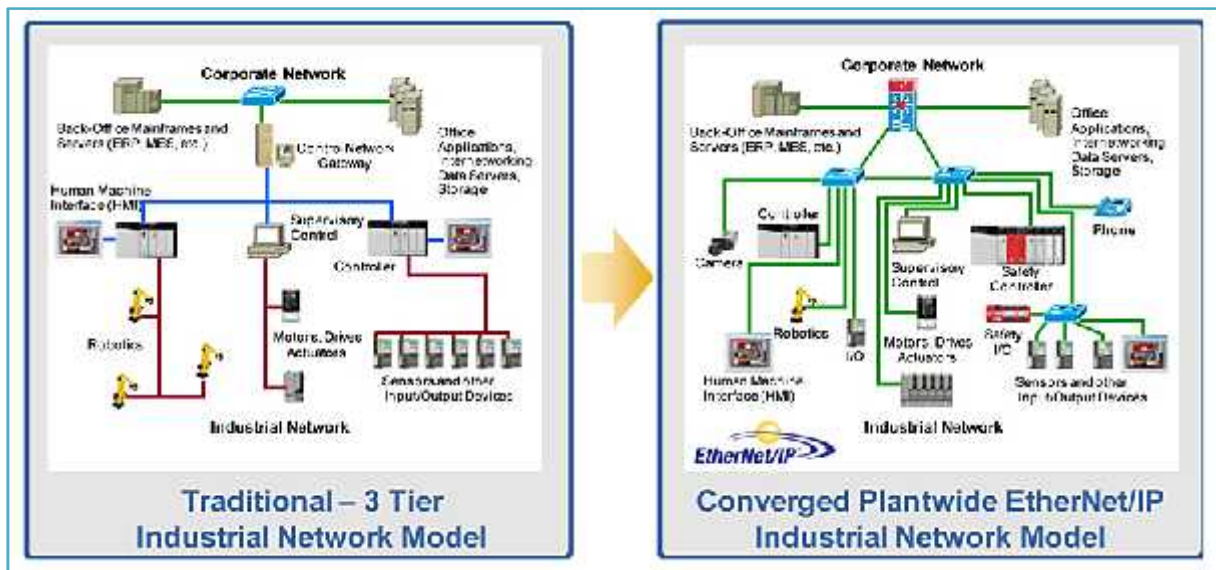


Εικόνα 7: Διασύνδεση σε δίκτυο PROFINET

## 2.3.6 INDUSTRIAL ETHERNET

μ Ethernet μ Ethernet (Industrial Ethernet). Ε μ  
 μ , μ μ μ PLC μ μ slave  
 μ μ μ μ μ , μ μ Profibus.  
 μ μ μ μ Ethernet μ .

1. μ μ 9.6 kbit/s μ RS232 1 Gbit/s μ 802  
 Cat5e/Cat6 .
2. μ .
3. μ .
4. μ « μ » μ ,  
 μ , μ μ , hubs  
 « μ » μ μ .
5. μ μ μ (peer-to-peer).
6. .



Εικόνα 8: Διασύνδεση Industrial Ethernet

### 2.3.7

## INDUSTRIAL ETHERNET

Ethernet :

1. (
2. TCP
3. UDP ( UDP TCP, TCP/IP

Ethernet CSMA/CD (carrier sense multiple access with collision detection). « » jam signal back-off delay milliseconds. Ethernet time-critical Fieldbus masters Ethernet H/Y

### 2.3.8

Δίκτυο	Ταχύτητα	Φυσική Τοπολογία	Λογική Τοπολογία	Αριθμός κόμβων	Ακτίνα	Φυσικό μέσο	Κόστος
ProfibusDP	12Mbits/sec	Διαυλος (bus)	Κεντρική, master/slave, peer-to-peer	Devices 32/124 with repeaters	1200m	twisted pair, RS485 or fiber	Μέτριο
DeviceNet	500Kbits/sec	Διαυλος (bus)	Κεντρική, master/slave, peer-to-peer, multi-master	64	500m	CAN Διαυλος/bus, twisted pair	Χαμηλό
CANopen	1Mbits/sec	Διαυλος (bus)	Κεντρική Master/Slave, Peer-to-peer, multi-cast, multi-master	127	100m	CAN Διαυλος/bus, twisted pair	Χαμηλό
Interbus	500Kbits/sec	Διαυλος (bus)	Κεντρική	512 devices, 4096 I/O ports	13km	Twisted-pair or fibre	Υψηλό
Modbus RTU	Πολύ αργό	Στοθμός-αε-σταθμό Διαυλος (bus)	Κεντρική Master / Slave	255	Εξαρτάται από την υλοποίηση	RS232, RS485	Πολύ Χαμηλό
BACnet	Εξαρτάται από την υλοποίηση	Διαυλος (bus)	Κεντρική Peer-to-peer	Εξαρτάται από την υλοποίηση	Εξαρτάται από την υλοποίηση	RS485, Ethernet, LonTalk & ptt et	Χαμηλό
SLM	2.5Mbits/sec	στοθμός-αε-σταθμό	Κεντρική	2	100m	RS485 twisted pair	Μέτριο
CTNet	5Mbits/sec	Διαυλος (bus)	Αποκεντρωμένη	255	250m	Twisted pair or Fibre	Μέτριο
CTSync	895Kbits/sec	Διαυλος (bus)	Master - Slave, Simplex	16	100m.	RS485, twisted pair	Χαμηλό
Sercas II	16Mbits/s	Δεκτικός (Ring)	Κεντρική	255	100m	Fibre	Υψηλό

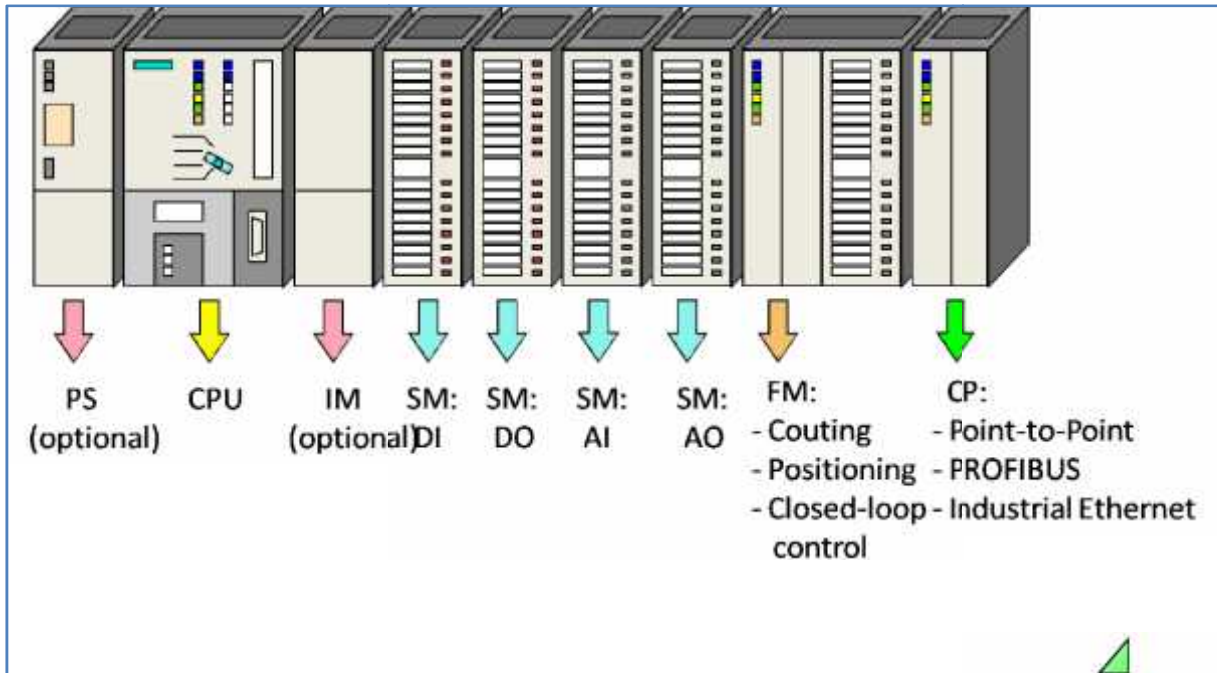
Εικόνα 9: Βασικά χαρακτηριστικά των Βιομηχανικών Δικτύων

### 3

## PLC

### 3.1

### PLC



Εικόνα 10 :Δομή PLC S7-300

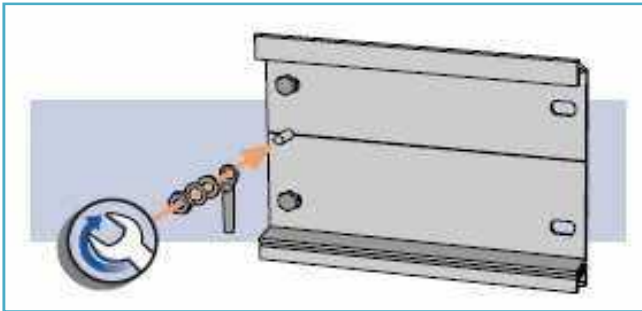
- PLC, S7-300
- (Rack).
- PS (Power Supply).
- CPU (Central Processing Unit).
- ( ) Digital or Analog.
- S7-300.
- FM (Function Module).
- CP.
- Rack Interface Module IM.
- Profibus Bus Connector.
- PROFIBUS
- PG. CPU

RACK	
POWER SUPPLY (PS)	
CPU	
ANALOG-DIGITAL SM	
PROFIBUS CABLE	
PG CABLE	
RACK INTERFACE IM	
CP MODULE	
FM MODULE	

Εικόνα 11: Στοιχεία PLC

### 3.2

### (RACK)

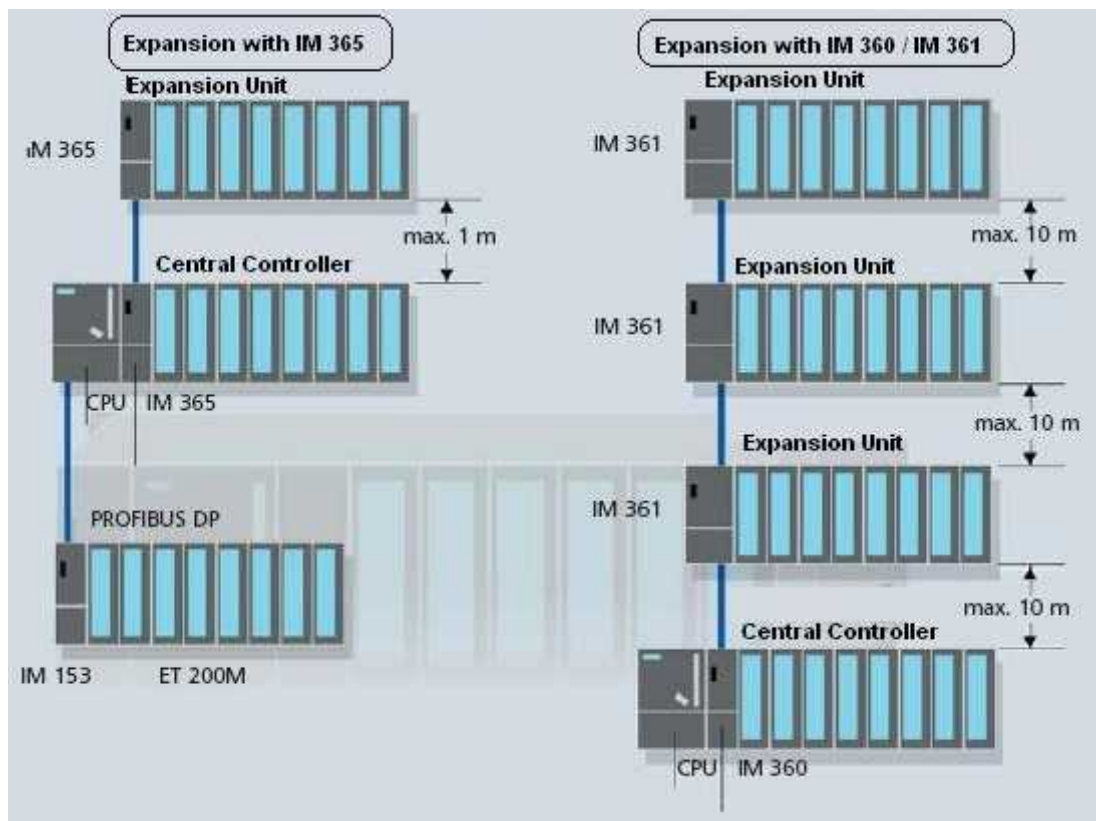


Εικόνα 12: RACK

μ μ μ . μ μ μ μ μ “ μ ”  
μ μ μ . Rack, 12, μ μ μ  
μ . :  
• μ , PS.  
• μ μ , CPU.  
• T “ μ ” μ  
Rack (IM).  
• Rack μ μ .  
Rack Rack (rack0). Rack μ  
PS, CPU, RACK(IM), μ  
Rack,  
CPU . PLC S7-300 μ  
Rack , μ μ μ  
μ , 13.  
O , μ μ μ  
“ ” .  
• **P-Bus** (Peripheral Bus), μ  
PLC. P-Bus μ  
μ 1.5 Mbps  
• **K-Bus** (Communication Bus) CPU μ  
( FM). K-Bus μ  
187.5 Kbps ( μ P-Bus).

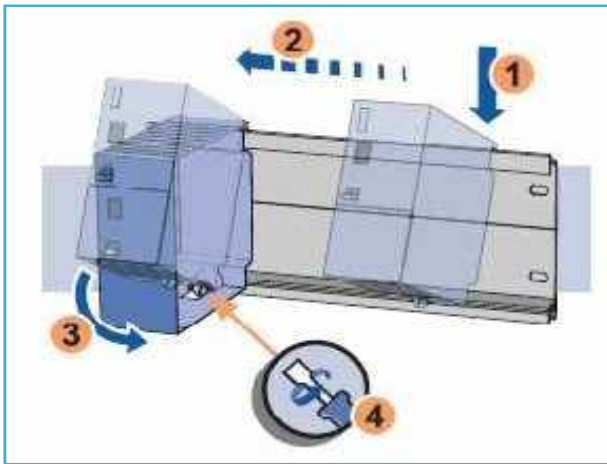


Επέκταση S7-300...



Εικόνα 13: Επέκταση Module S7-300

### 3.3 PS (Power Supply)



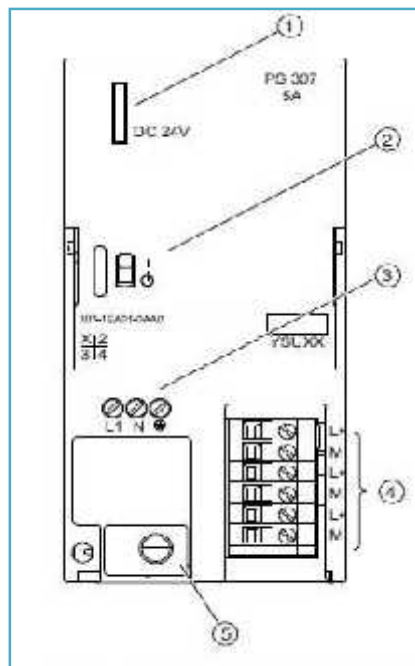
Εικόνα 14: PS

S7 300

24 V DC,

14.

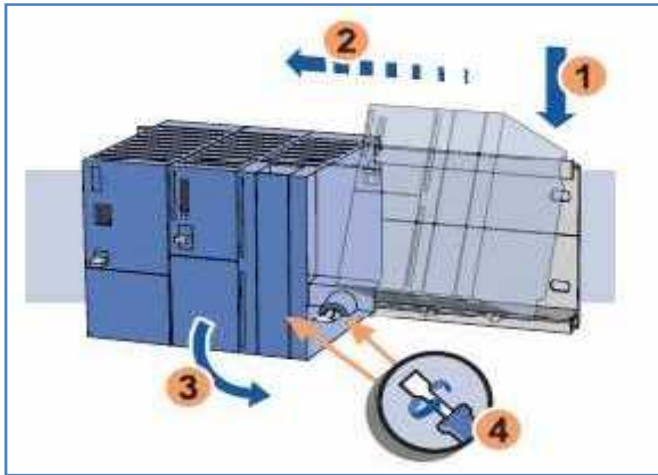
PLC.



Εικόνα 15: PS 307: 5A Wiring Diagram

- :
- μ (L1,N)
  - μ 24 volt (L+,M ).
  - ON/OFF.
  - LED 24 VDC.

### 3.4 Κεντρική Μονάδα Επεξεργασίας (CPU)



Εικόνα 16: CPU

μ CPU (Central Processing Unit), 16,

μ PLC. μ μ

:

- μ μ μ μ .
- μ μ μ .
- μ .
- μ .

μ , μ CPU μ μ , CPU

« μ μ . PLC μ « μ » . PLC



Συνέχεια των κατηγοριών...

4. RUN STOP μ μ CPU
5. μ μ μ STEP 7. PLC μ μ ( . . ) μ μ μ μ 8 bit μ μ μ μ

7	6	5	4	3	2	1	0
0.5 Hz	0.625 Hz	1 Hz	1.25 Hz	2 Hz	2.5 Hz	5 Hz	10 Hz

Εικόνα 17: Μνήμη ρολογιού

### 3.4.2

### CPU



Εικόνα 18: CPU 314C-2 DP

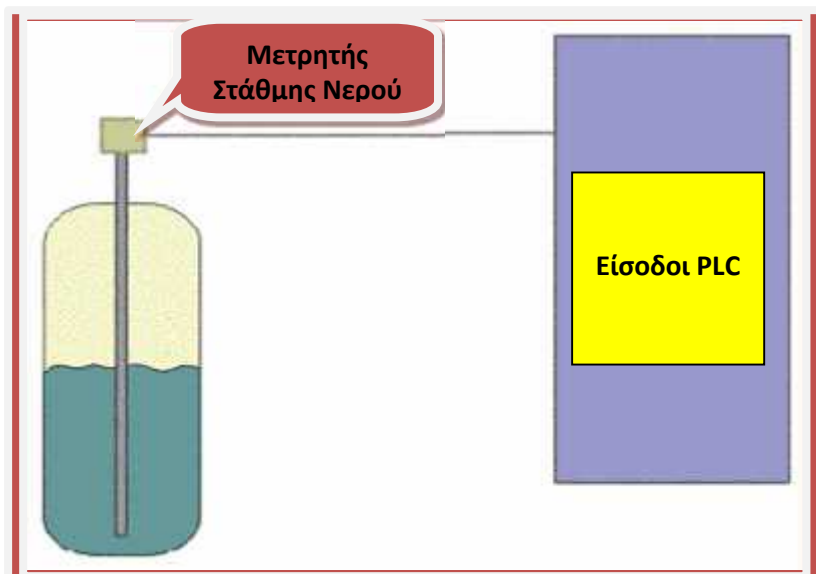
μ μ μ μ CPU, 18 :

1. LED
2. μ μ (FEPROM).
3. RUN/STOP/MRES.
4. μ PC μ MPI
5. μ PC μ DP
6. CPU μ 24 VDC
7. (Analog Input)
8. (Analog Output)
9. (Digital Output)
10. (Digital Input)



### 3.6 PLC (Analog Input)

PLC. PLC  
 0-10V +10V,-10V 0-20 mA 4-20 mA  
 16 bits. 20, "μ" "μ" 4-20 mA  
 "μ" "μ" "μ" "μ" "μ" "μ" "μ" "μ"



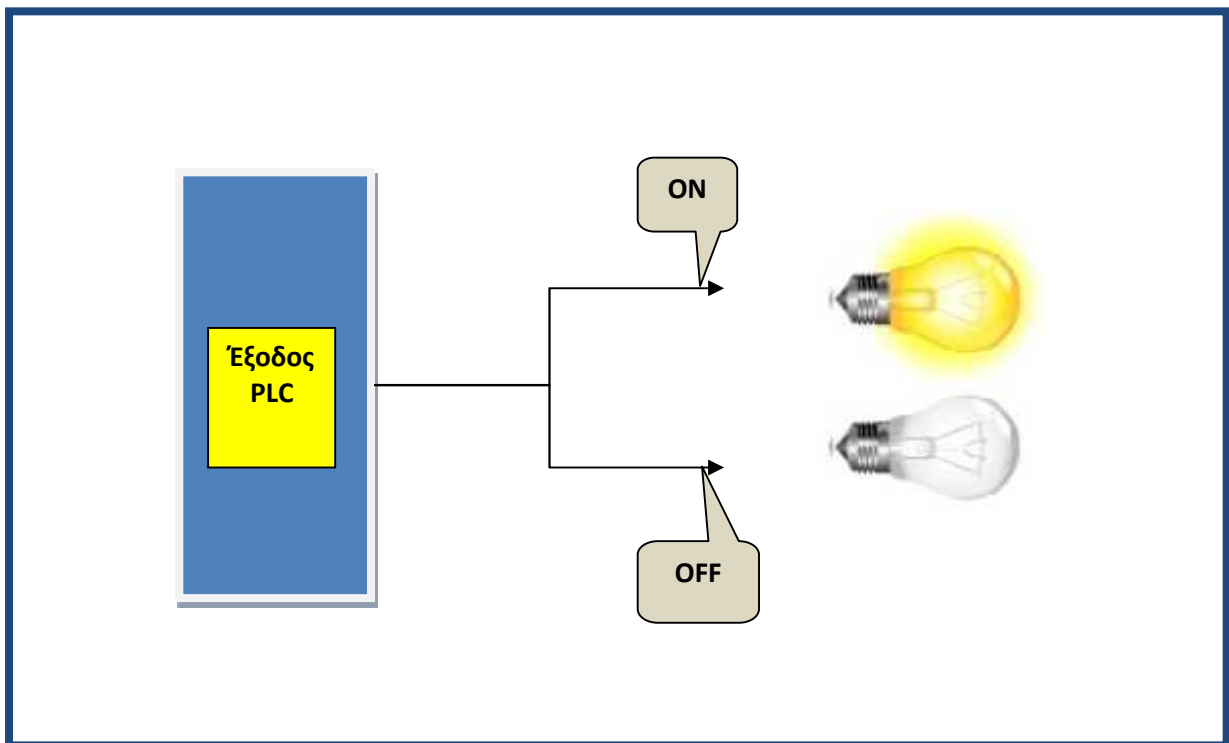
Εικόνα 20: Analog Input

Digital Converters) "μ" "μ" "μ" "μ" "μ" "μ" "μ" "μ" (Analog to  
 μ 0 1, "μ" "μ" "μ" "μ" "μ" "μ" "μ" "μ" CPU. μ



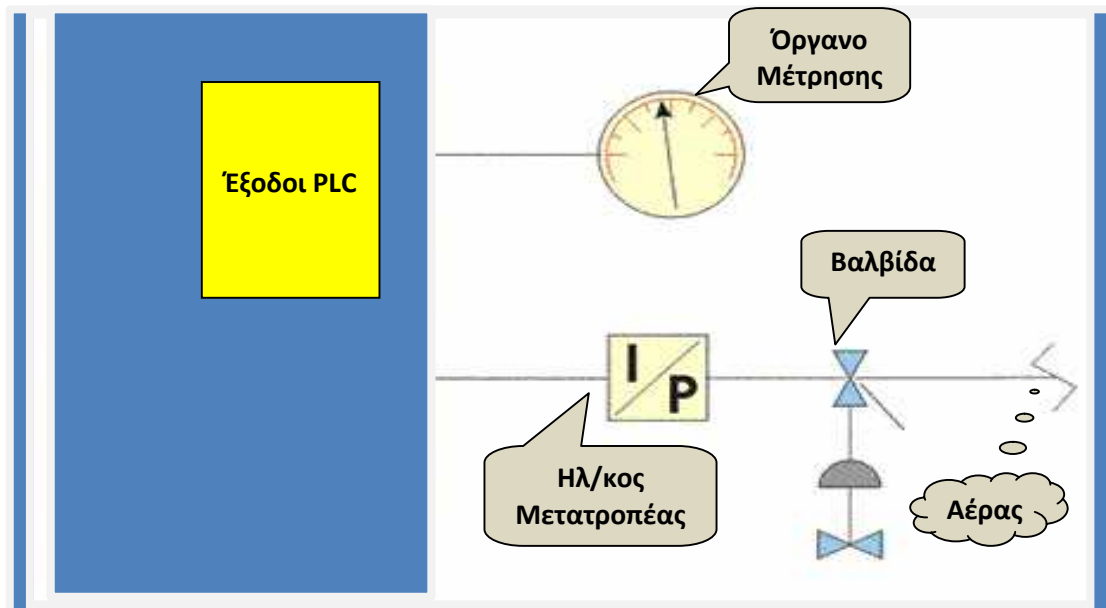
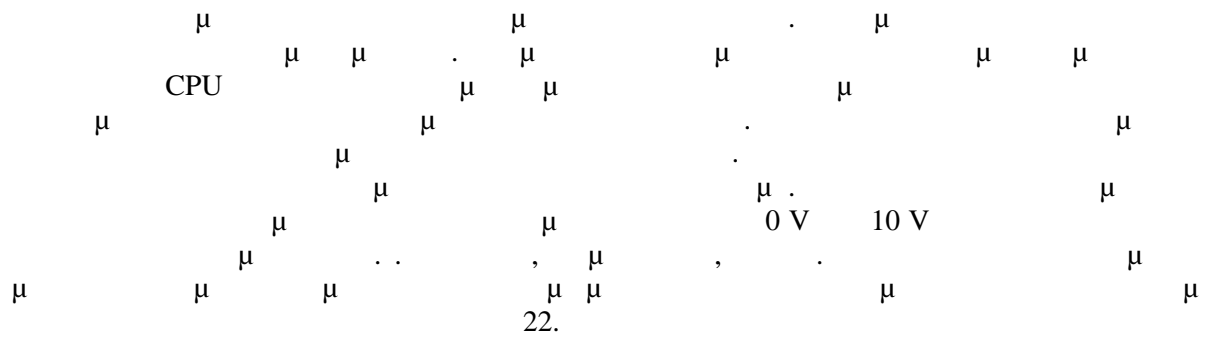
### 3.7 PLC (Digital Output)

μ μ . μ CPU ON OFF.  
μ μ . μ  
8,16 32 μ μ  
μ 21, μ  
ON μ OFF.  
PLC μ .



Εικόνα 21: Digital Output

### 3.8 PLC (Analog Output)



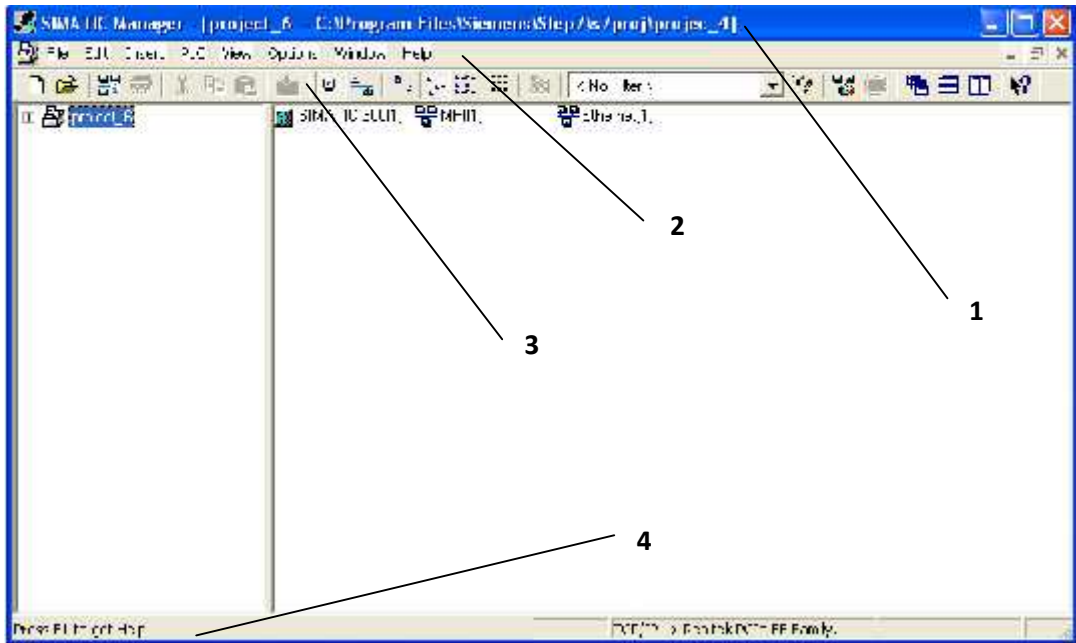
Εικόνα 22: Analog Output

# 4

## SIMATIC MANAGER

### 4.1 Simatic Manager Step 7

μμ μ PLC μ Step 7. Simatic Manager  
 online offline μ μ (projects, block, μμ  
 ). Simatic Manager μ μ :  
 μ projects ( μ libnodave ).  
 μ Step 7.  
 μ online μ PLC.  
 μ μ μ μ .



Εικόνα 23: Γραφικό Περιβάλλον Simatic Manager

- 1. μ :
- 2. button
- 3. μ μ μ μ button μ μ .
- 4. μ .

❖ μ . μ μ μ μ μ μ μ μ F1.

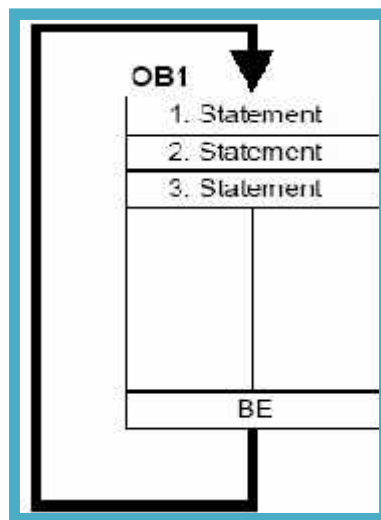
## 4.2

project block CPU

1. CPU, CPU (RUN, STOP, RESET), Led CPU MPI Interface.

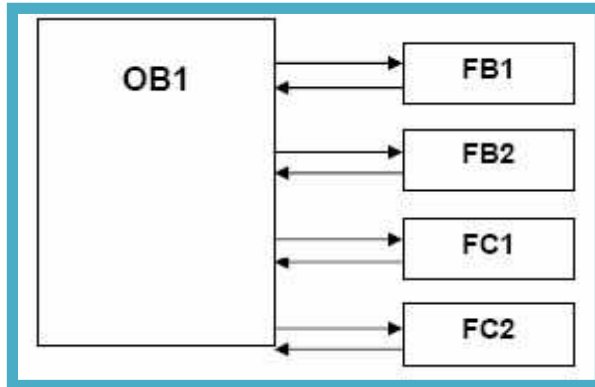
2. PLC block block PLC. block

I. OB1. CPU block.



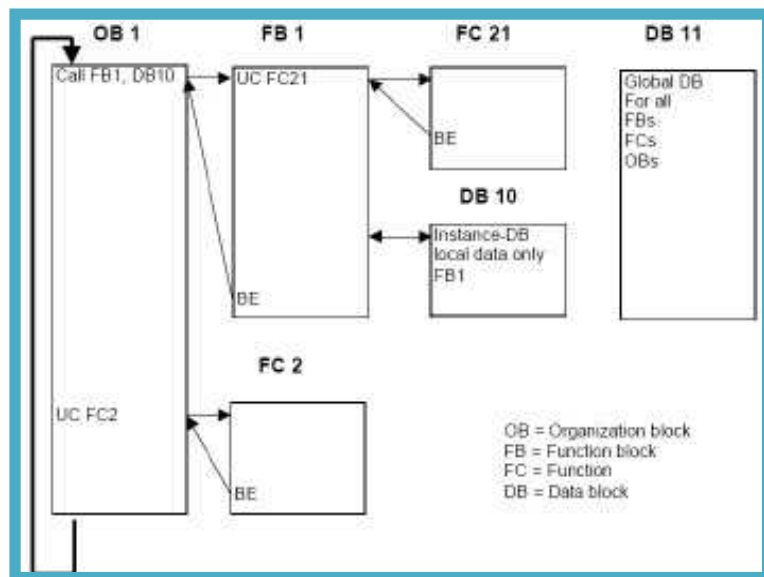
Εικόνα 24: Γραμμικό Πρόγραμμα

II. μ μ μ μ μ . μ μ block  
 μ μ block OB1.



Εικόνα 25: Τμηματοποιημένο Πρόγραμμα

III. μ μ μ μ . μ μ μ μ μ μ  
 μ μ block. μ μ μ μ μ μ  
 block OB1 μ μ block.  
 μ μ μ μ μ μ μ μ μ μ μ μ  
 μ μ μ μ μ μ μ μ μ μ μ μ  
 μ μ μ μ μ μ μ μ μ μ μ μ



Εικόνα 26: Δομημένο Πρόγραμμα

### 4.3 Block

block

μμ μ . μ μ μ μ μ . μ .

μ block :

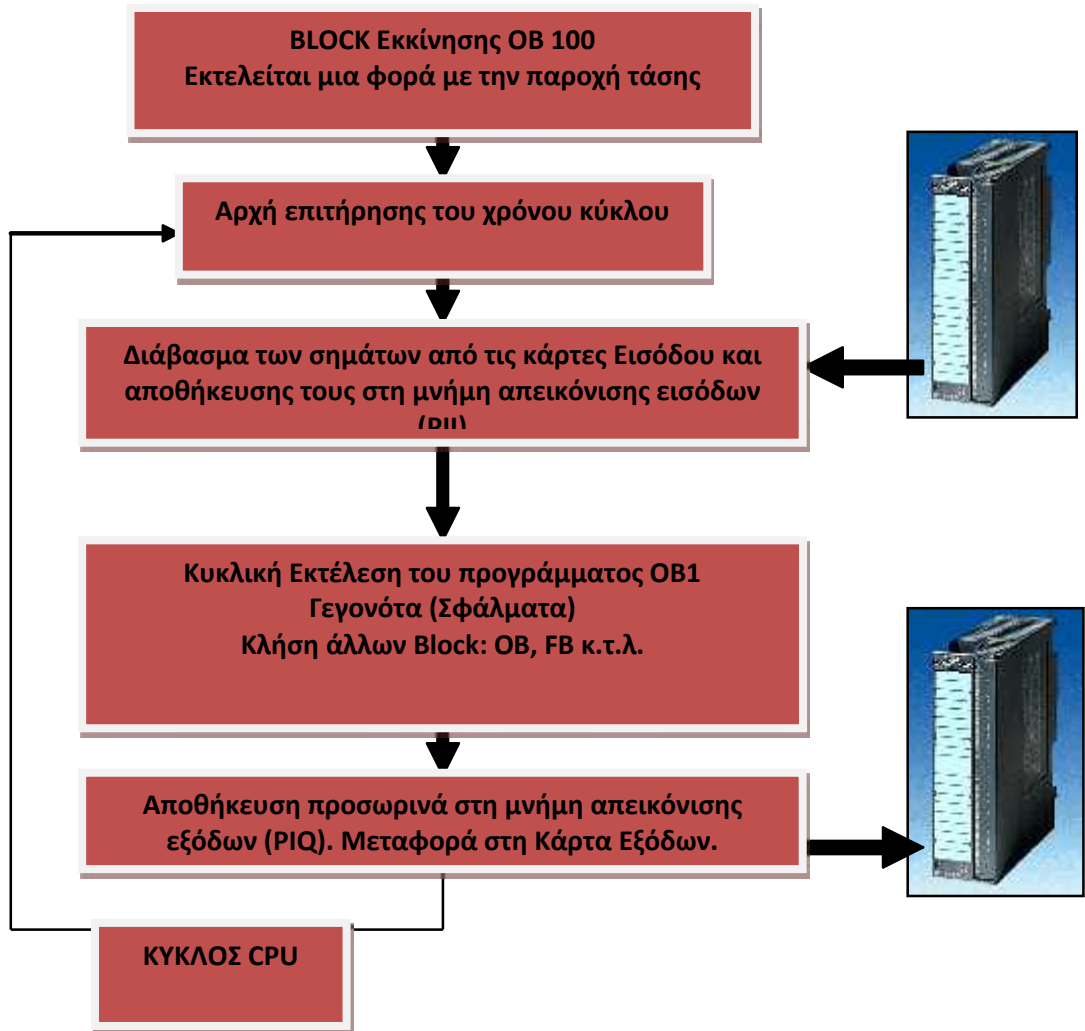
➤ **OB (Organization Block).**

( . . μ μ ) μμ μ . μ CPU block .

μ block OB1. CPU μ

μ block 100. μ μμ block . μ μ 27,

μ CPU. 1.



Εικόνα 27: Κυκλική επεξεργασία προγράμματος μιας CPU

OB (1) μ 0 29. μ μ OB1

Μπλοκ οργάνωσης	Συνθήκες κλήσης	Προτεραιότητα	
		Προεπιλεγμένη	Τροποποιησιμη
Ελεύθερος κύκλος OB 1	Κυκλικά μέσω του λειτουργικού συστήματος	1	Όχι
Χρονικές διακοπές (TOD) OB 10 ως OB 17	Σε συγκεκριμένη ώρα της ημέρας ή σε τακτά χρονικά διαστήματα (π.χ. μηνιαίως)	2	2 ως 24
Διακοπές καθυστέρησης OB 20 ως OB 23	Μετά από προγραμματισμένο χρόνο, ελεγχόμενο από το πρόγραμμα χρήστη	3 ως 6	2 ως 24
Διακοπές χρονιστή επιτήρησης OB 30 ως OB 38	Τακτικά σε προγραμματισμένα χρονικά διαστήματα (π.χ. κάθε 100ms)	7 ως 15	2 ως 24
Διακοπές επεξεργασίας OB 40 ως OB 47	Σε σήματα διακοπών από τις βαθμίδες I/O	16 ως 23	2 ως 24
Διακοπή πολυεπεξεργασίας OB 60	Κλήση υπό συνθήκες μέσω του προγράμματος χρήστη σε κατάσταση πολυεπεξεργασίας	25	Όχι
Εφεδρικά σφάλματα OB 70,	Στην περίπτωση απώλειας εφεδρικού στοιχείου που απορρέει από σφάλμα I/O,	25	2 ως 26
OB 72,	Στην περίπτωση εφεδρικού σφάλματος της CPU,	28	2 ως 28
OB 73	Στην περίπτωση εφεδρικού σφάλματος επικοινωνιών	25	2 ως 26
Ασύγχρονα σφάλματα OB 80, OB 81 ως OB 84, OB 86, OB 87, OB 85	Λάθη που δεν σχετίζονται με την εκτέλεση του προγράμματος (π.χ. χρονικά σφάλματα, σφάλματα SE, διαγνωστικές διακοπές, διακοπές εγκατάστασης ή απεγκατάστασης βαθμίδων, αποτυχία βάσης στήριξης ή σταθμού)	26 <sup>21</sup> 26 <sup>21</sup> 26 <sup>21</sup>	26 2 ως 26 24 ως 26
Εκτέλεση στο παρασκήνιο OB 90	Ελάχιστη διάρκεια χρονικού κύκλου που δεν έχει επιτευχθεί ακόμα	29 <sup>11</sup>	Όχι
Ρουτίνα εκκίνησης OB 100, OB 101, OB 102	Σε προγραμματιζόμενη εκκίνηση ελεγκτή	27	Όχι
Σύγχρονα σφάλματα OB 121, OB 122	Σφάλματα που σχετίζονται με την εκτέλεση του προγράμματος (π.χ. σφάλματα πρόσβασης I/O)	Η προτεραιότητα των OB που προκαλεί τα σφάλματα	

Εικόνα 28: Πίνακας με τα διαθέσιμα block και την προτεραιότητα τους

➤ **FC (Functions).** FC block (Local Data Stack).  
 DB block (Shared Data Block). FC Block.  
 block :  
 • FC block  
 • FC block  
 • FC block

➤ **FB (Function Block).** FB block (Data Block).  
 block . FB  
 block FB  
 variables) instance data block. (static variables)  
 (temporary variables)  
 FB instance data block (local data stack)  
 FB block FB

➤ **DB (Data Block).** DB

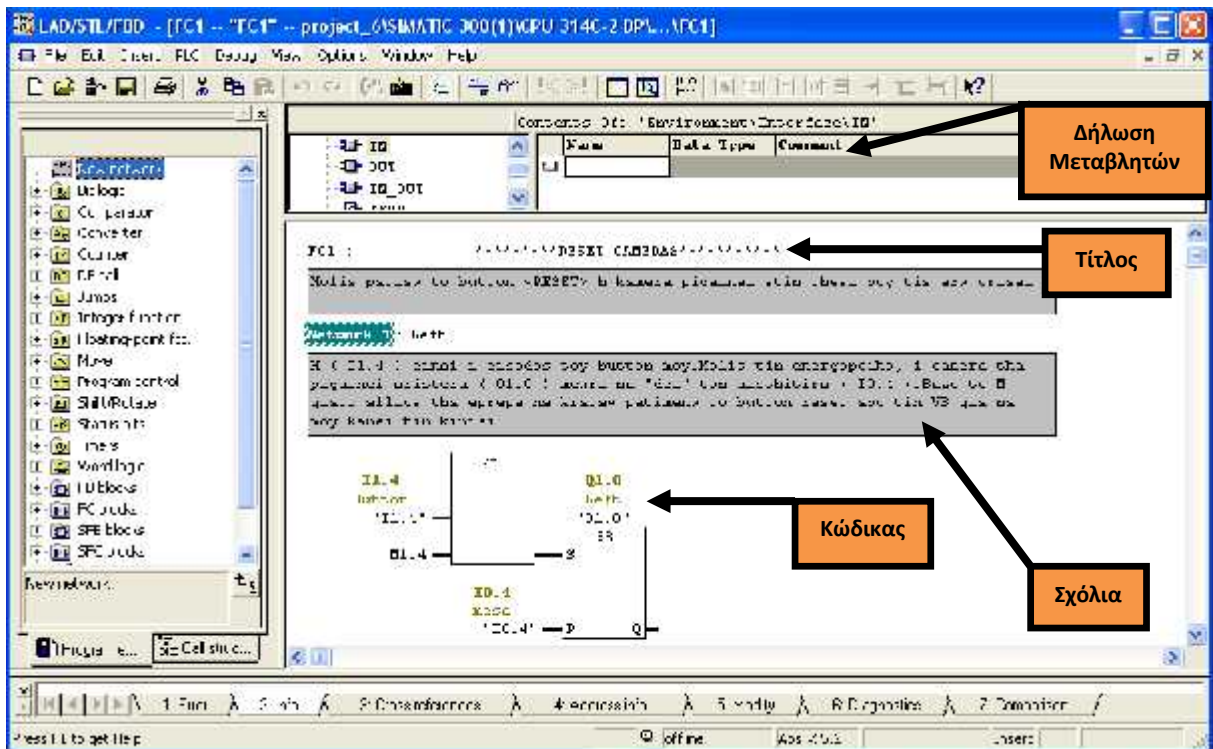
- a) **GD (Global Data Block).** GD
- b) **IDB (Instance Data Block).** IDB  
 FB CPU (FC, FB, OB) (L-Stack) DB. DB

DB. write( GB ) read( IDB )



## 4.4 μ Block

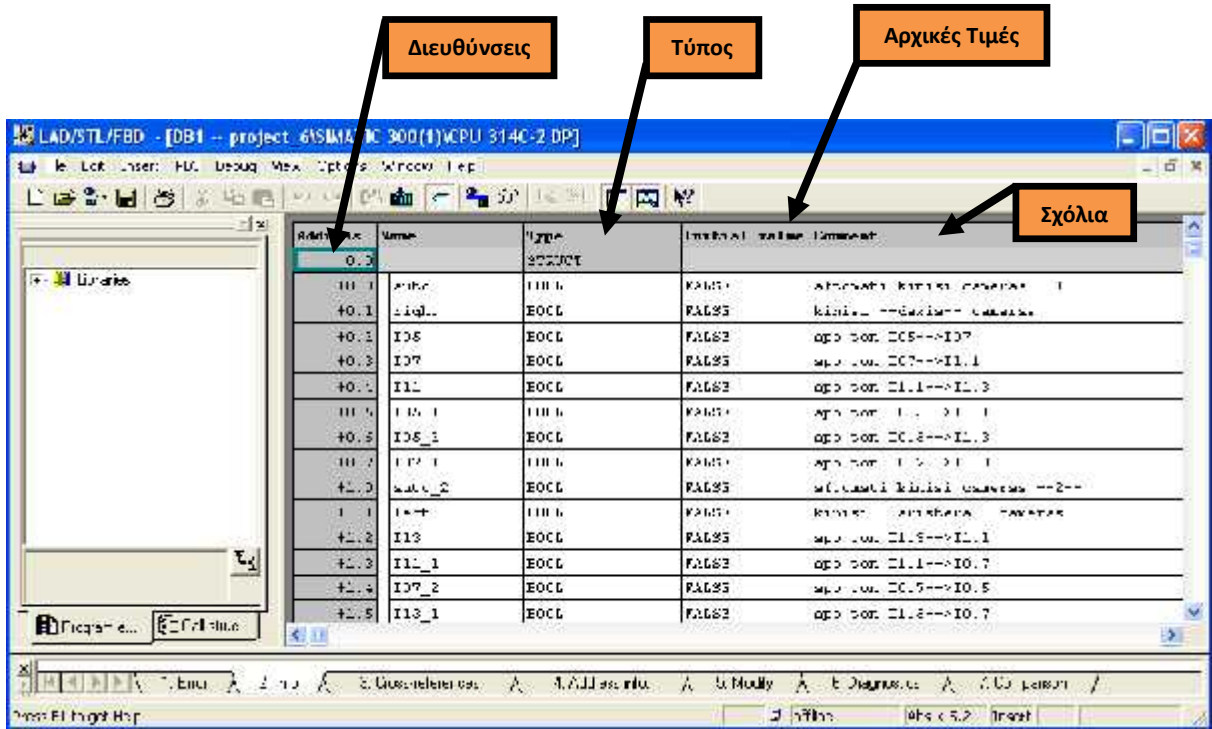
- ✓ μ , . . . FC1, 29 :
- ✓ μ , μ μ μ .
- ✓ μ μ μ .



Εικόνα 29: Δομή της FC1 μου

✓ μ DB μ , 30. :

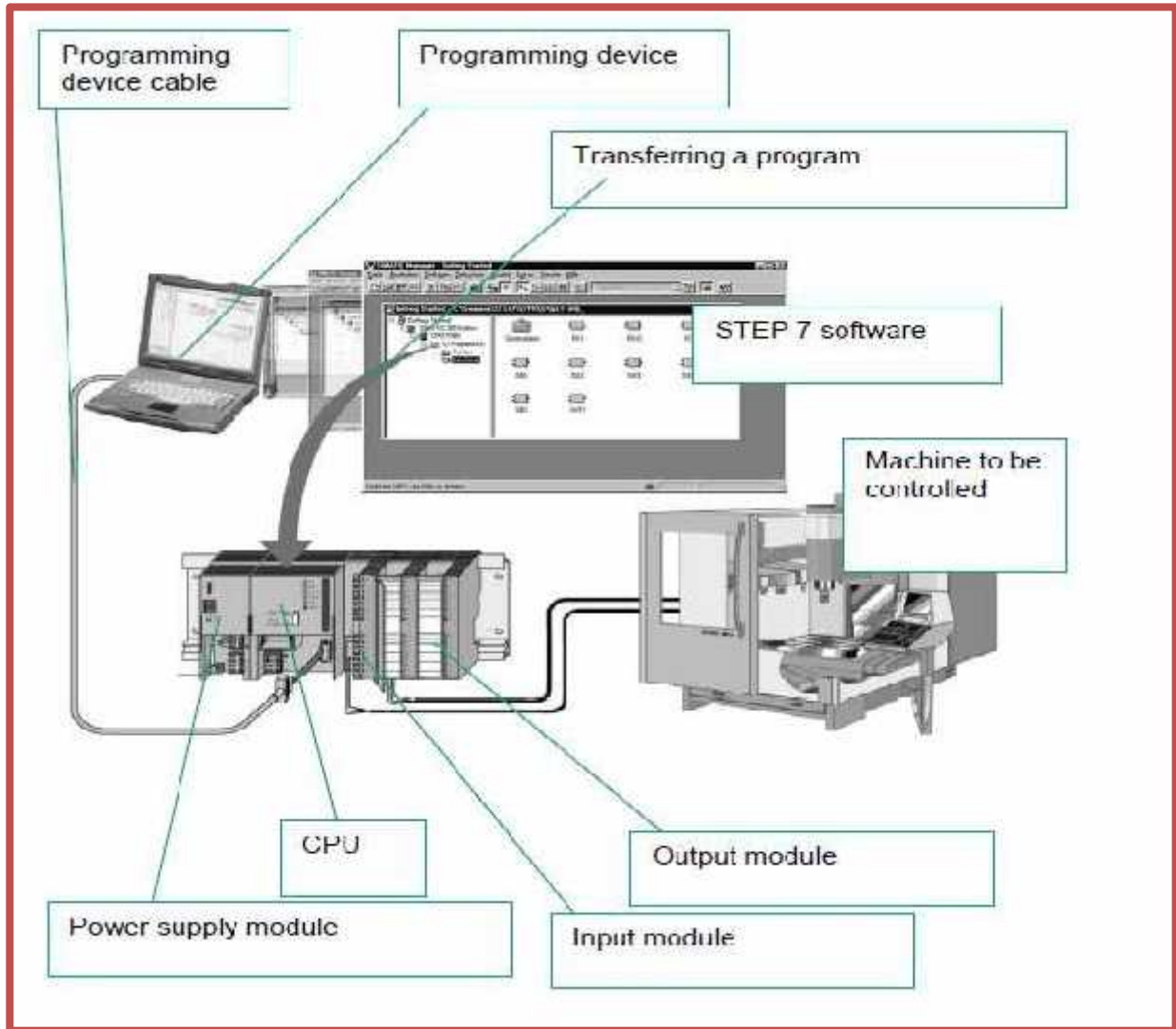
✓ μ μ μ μ μ μ .



Εικόνα 30: Δομή του DB1 μου

## 4.5 STEP 7

STEP 7 : (PS), (Input-Output modules). S7 31. Simatic PLC. project. S7 (CPU)



Εικόνα 31: Διάγραμμα σύνδεσης Hardware και Software

μ μ μ μ μ μ

1. To **Hardware** :

- μ -
- μ μ
- μ rack
- CPU
- μ HMI
- μ

2. To **Software** :

- μ μμ
- μ μ μ μ μ
- μ μ (Configuration)
- μ
- μ μμ

Step 7, Hardware Software μμ μ project μ μ .  
 μ μμ μ μ μ .  
 μ μμ μμ hardware  
 configuration Step 7 μ - μ μ  
 μ μ μμ μ μ .  
 μ μ μ μ μ μ Step 7.

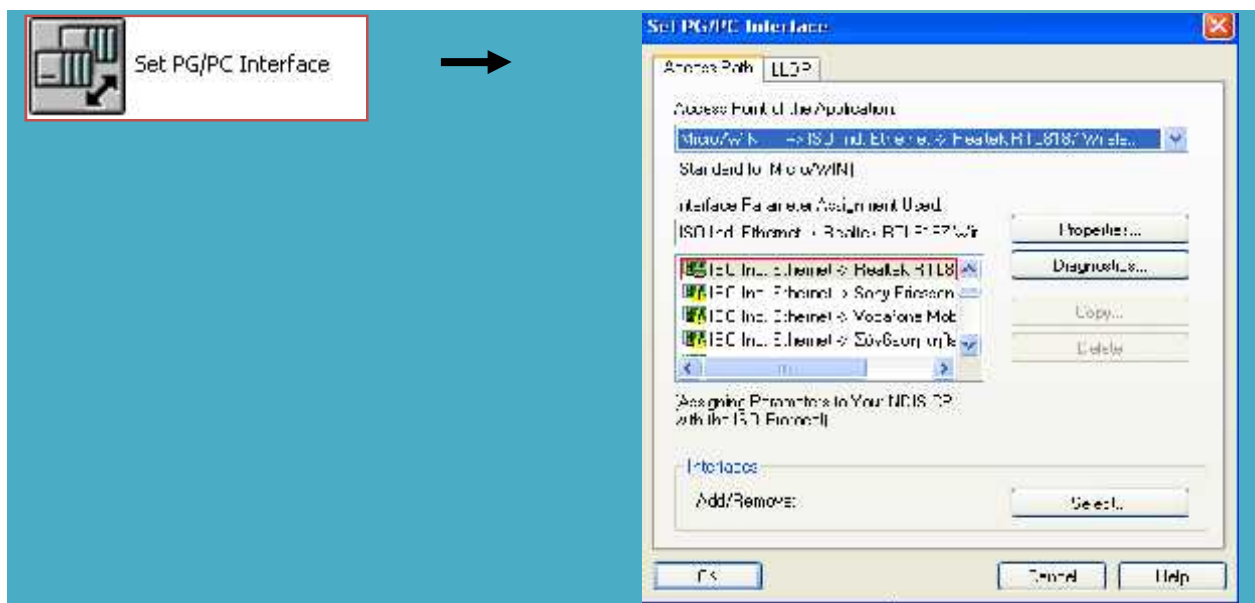
## 4.6 PLC project

PLC project

1) , :



2) :



Εικόνα 32: Set PG/PC Interface

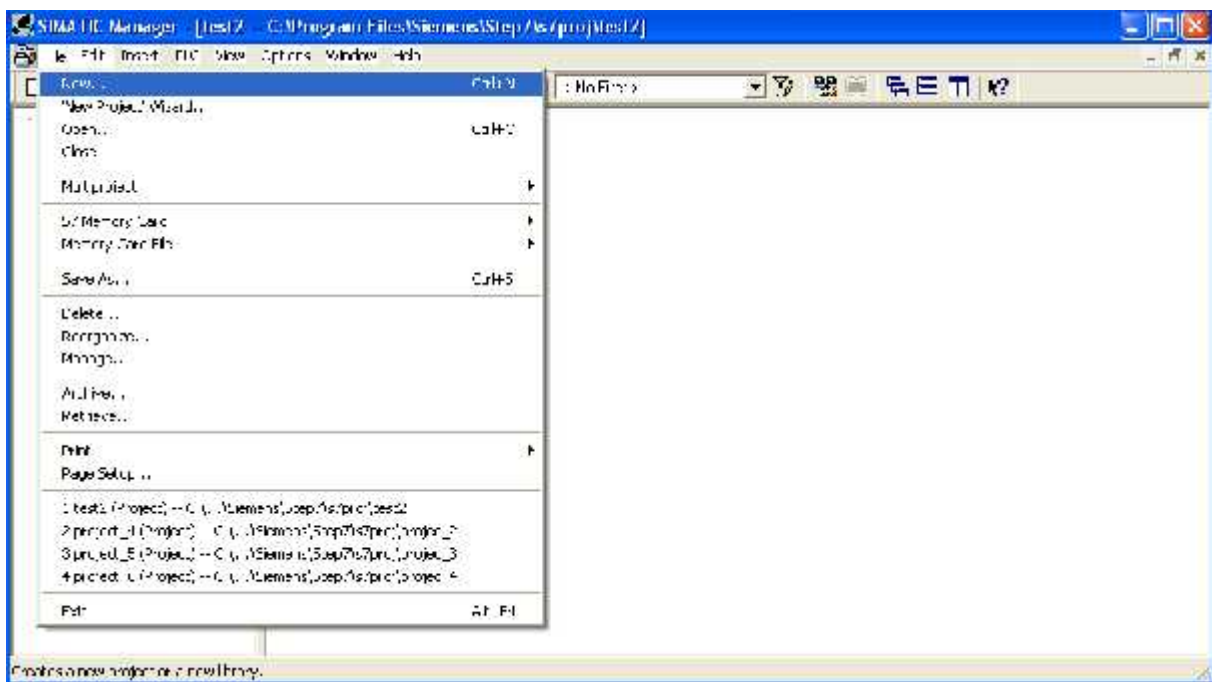
❖ μ μ project μ .

**ISO Ind. Ethernet.**

## 4.7 Project

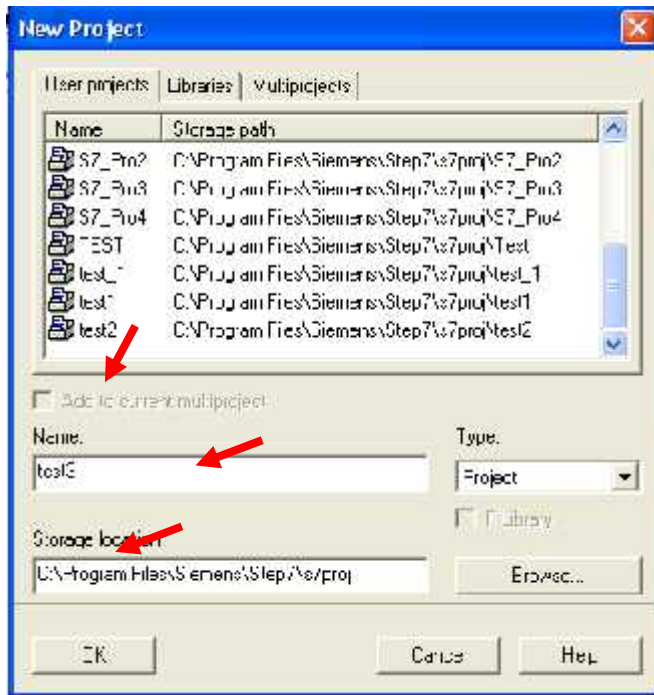
μ 1 :  Double click

μ 2 : Simatic Manager, μ File > New



Εικόνα 33: Επιλογή δημιουργίας νέου project

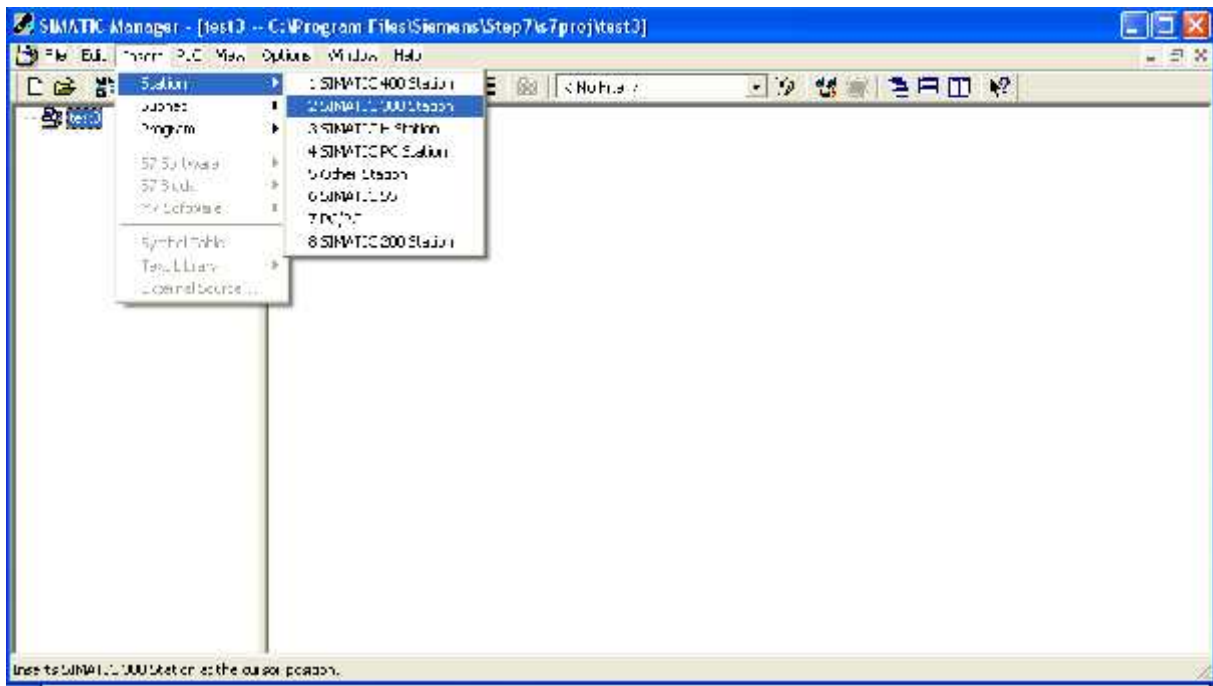
μ 3 : Εμ  
μ μ project  
μ >



Εικόνα 34: Δημιουργία νέου project

## 4.7.1

PLC project μ , project μ μ S7-300.  
: **Insert > Station > SIMATIC 300**

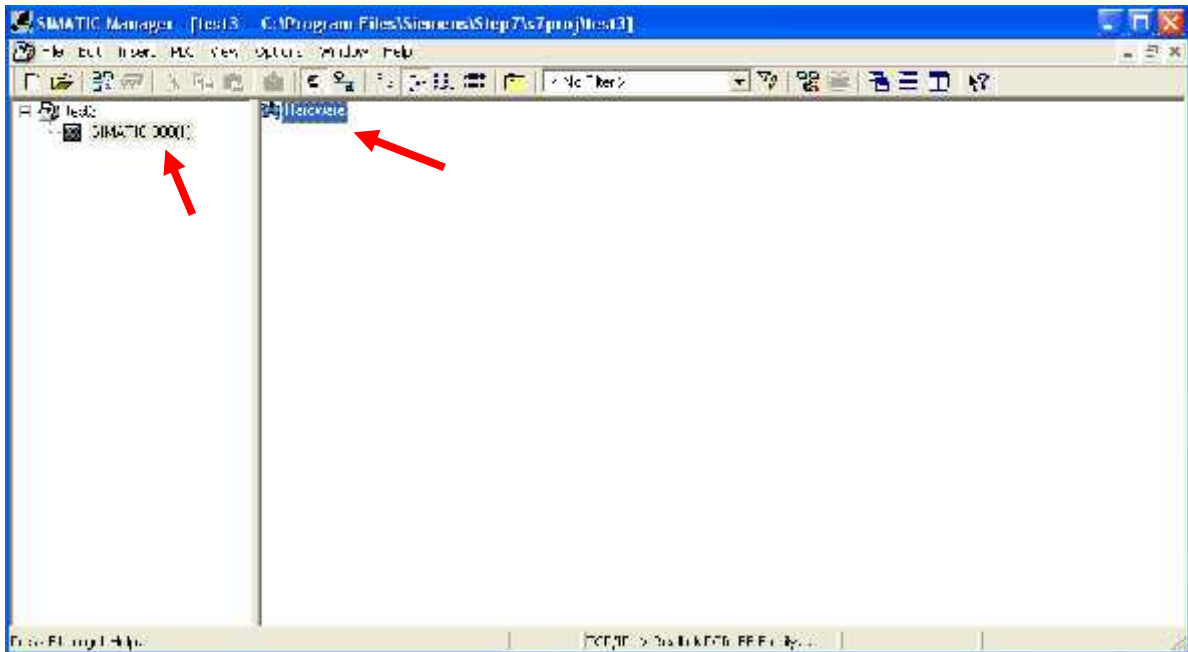


Εικόνα 35: Εισαγωγή σταθμού S7 300

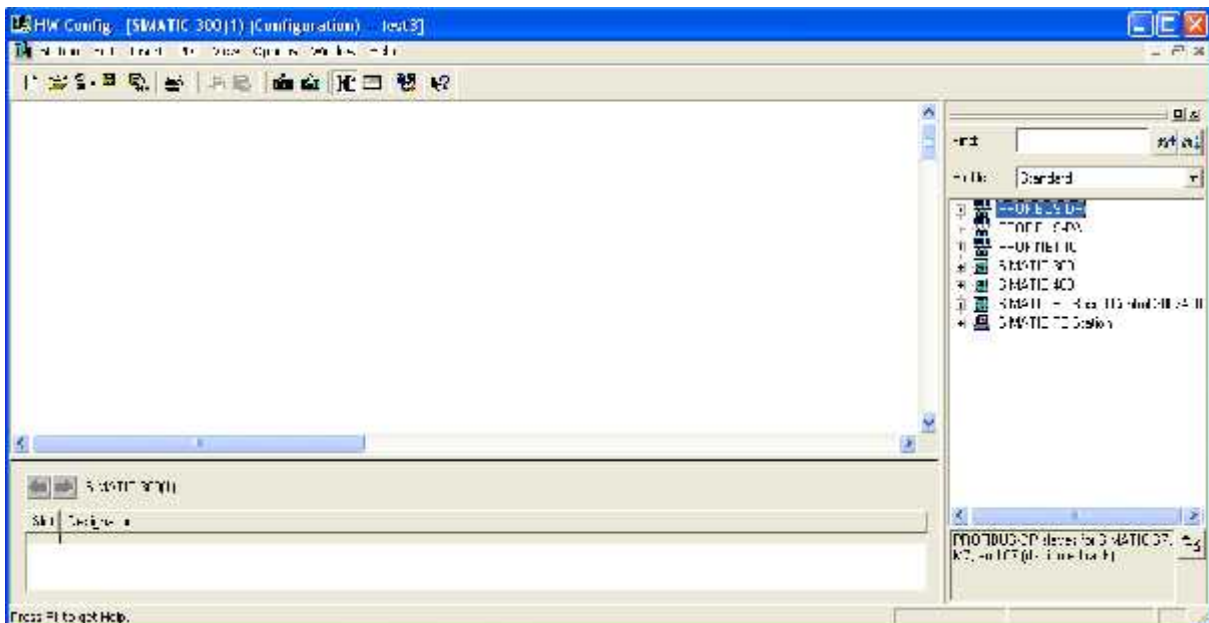


## 4.7.2 – Hardware Configuration

- HW Config.
- (SIMATIC 300) Hardware.
  - ( )
  - Hardware.

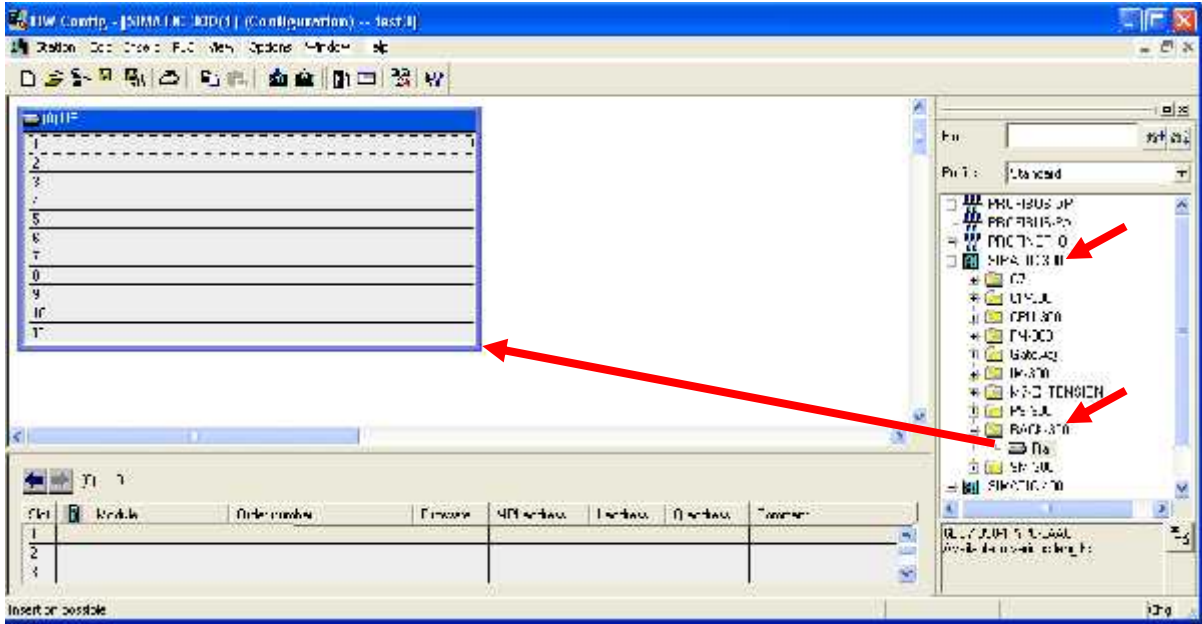


Εικόνα 36: Hardware



Εικόνα 37: HW Config

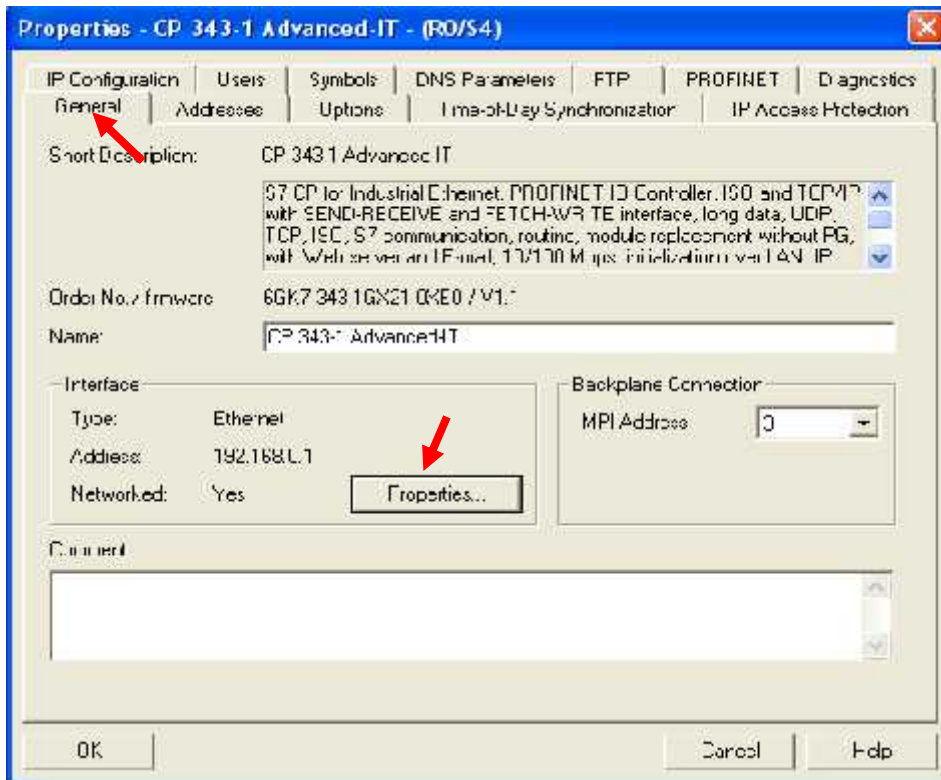
Hardware, Rack  
 μ : **Simatic 300 > Rack-300 > Rail.**



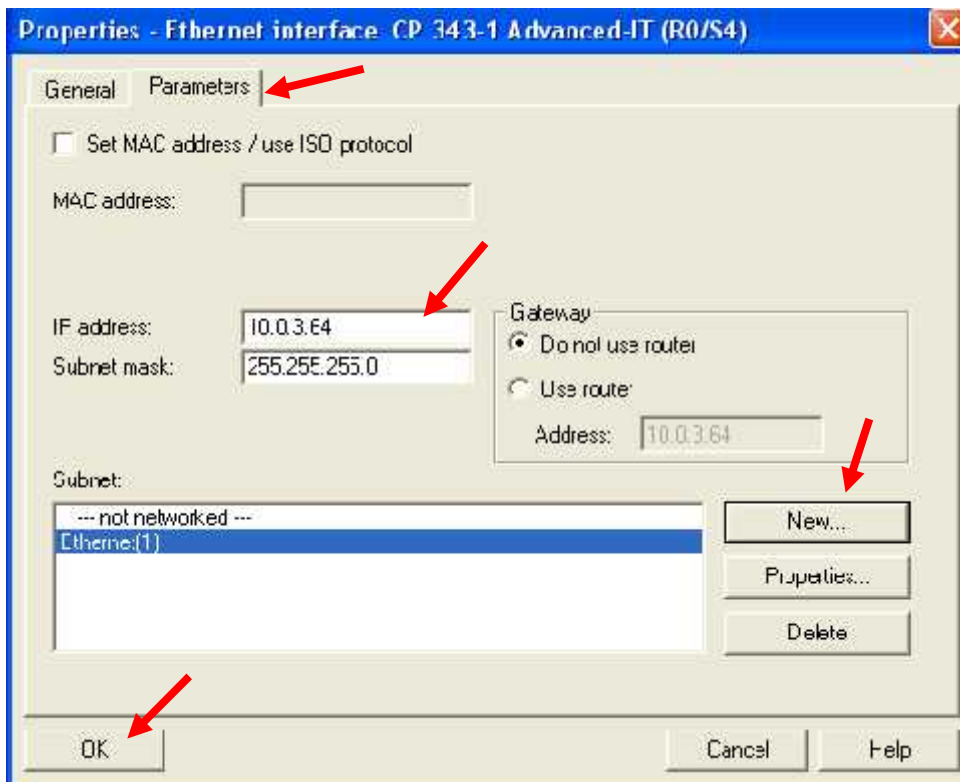
Εικόνα 38: Rail

- rack μ , 1 μ , 2  
 CPU, 3 rack 4 11  
 SM, FM CP.  
 μ project μ :
- 1: **PS-300 > PS 307 5A.**
  - 2: **CPU 300 > CPU 314C-2DP > 6ES7 314-6CF02-0AB0.**
  - 3: **\_\_\_\_\_μ\_\_\_\_\_.**
  - 4: **CP-300 > Industrial Ethernet > CP 343-1 Advanced-IT > 6GK7 343-1 GX21-  
 OXEO > V1.1.**

module (CP) μ μ Ethernet.  
 μ ...General > Properties... > Parameters. ...New,  
 μ **IP 10.0.3.64** μ PLC, μ μ  
 Router ( 39,40).

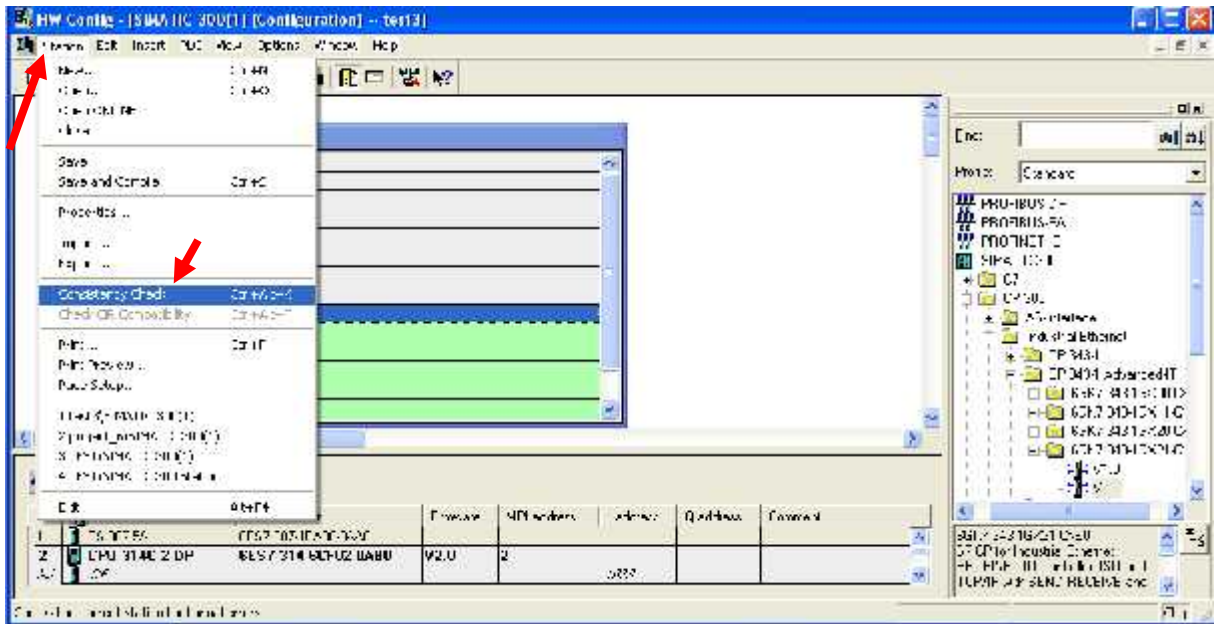


Εικόνα 39: Ιδιότητες του CP



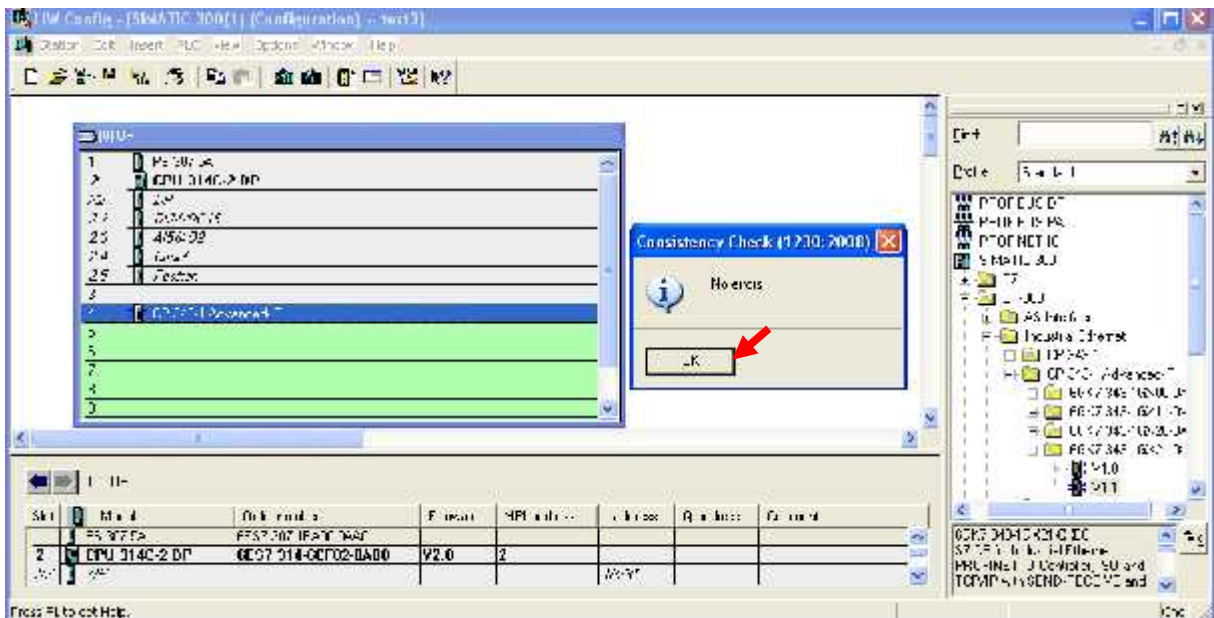
Εικόνα 40: Ορισμός της IP

Hardware :Station > Consistency Check ( 41)



Εικόνα 41: Έλεγχος HW Config

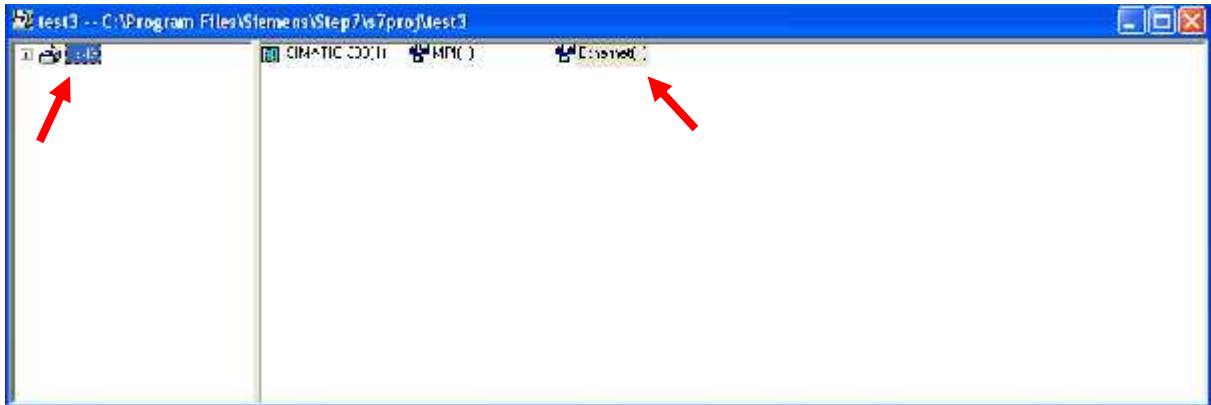
μ μ μ μ μ ( 42)



Εικόνα 42: Σωστό HW Config

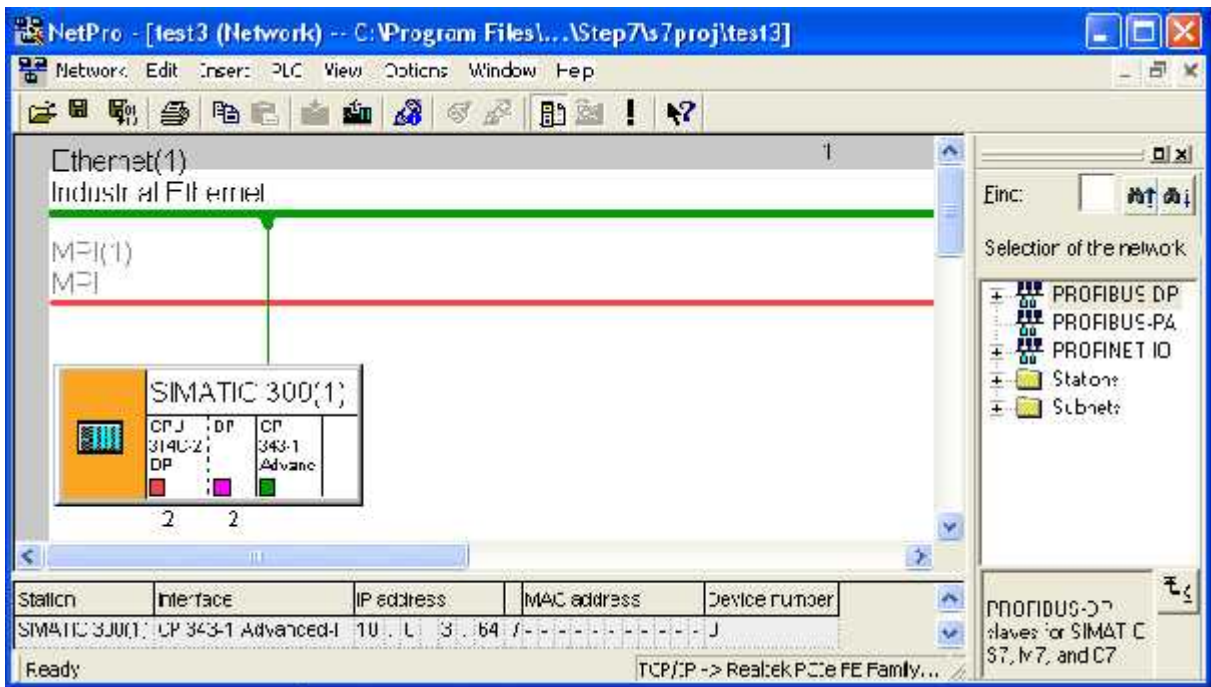
Netpro

project > Ethernet(1)



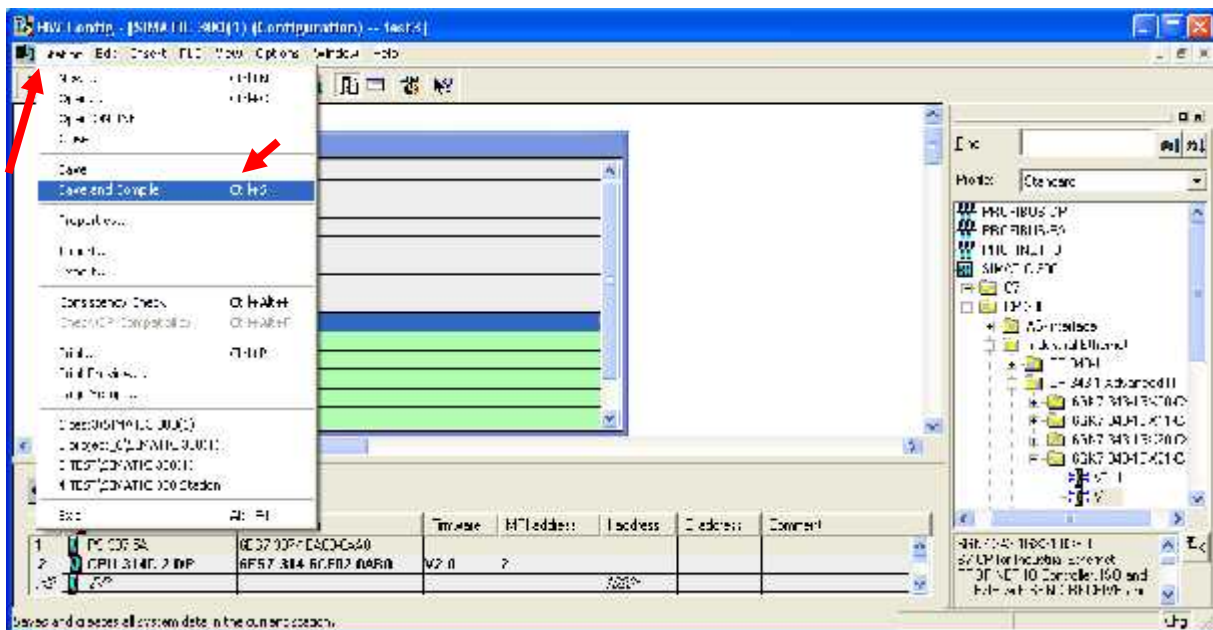
Εικόνα 43: Εύρεση Netpro

Netpro.



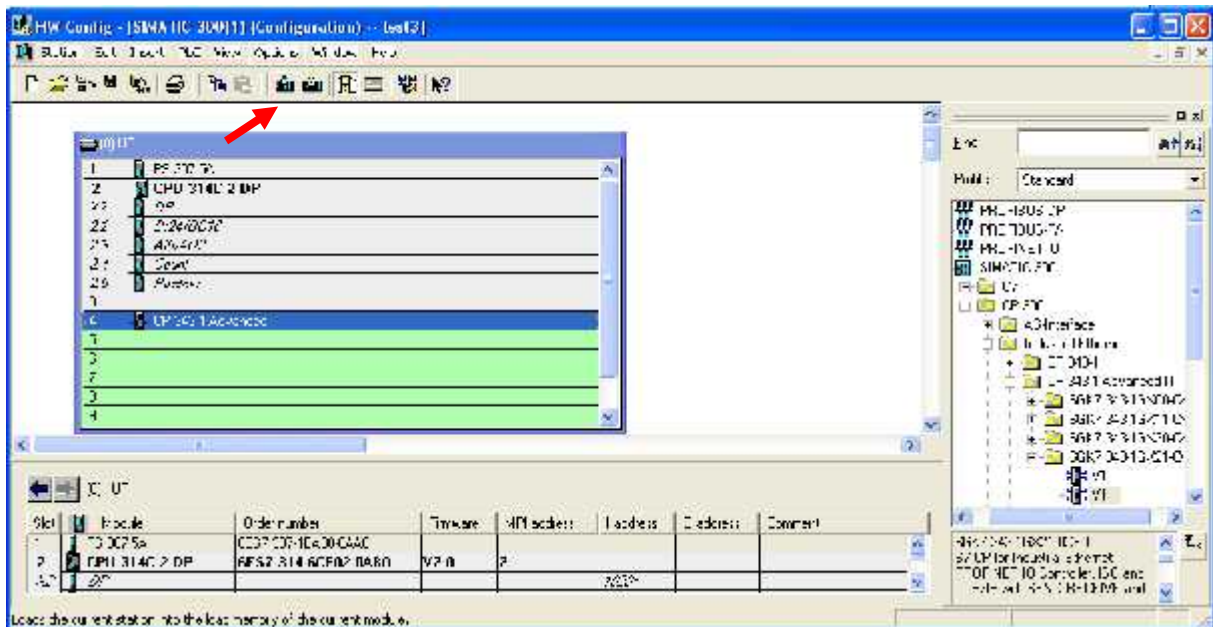
Εικόνα 44: Netpro

**Station > Save and Compile ( 45).**



Εικόνα 45: Save and Compile

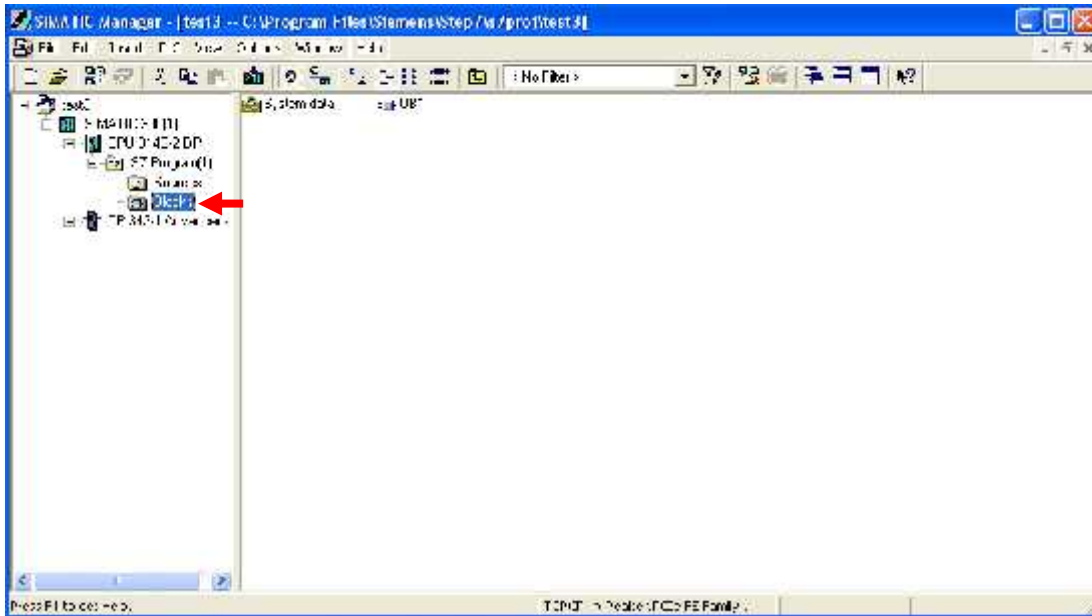
**Download Hardware PLC ( 46)**



Εικόνα 46: Download Hardware

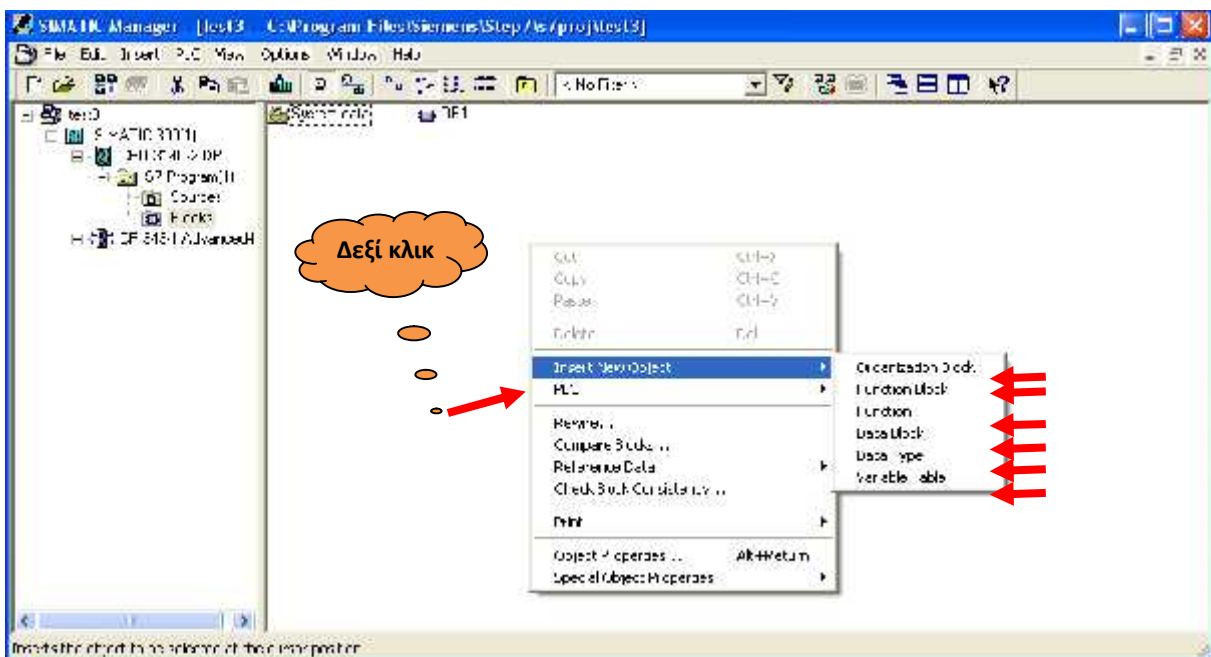
### 4.7.3 Project Blocks

Hardware project Blocks (47).



Εικόνα 47: Project Blocks

OB1 PLC. Insert New Object (48)



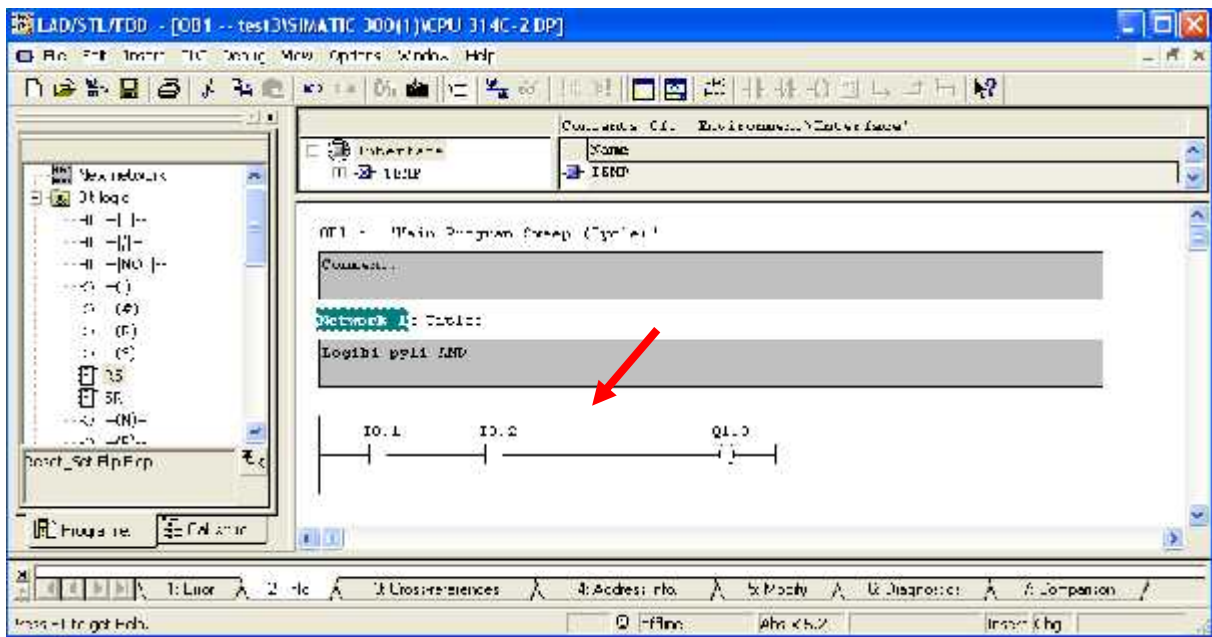
Εικόνα 48: Εισαγωγή Block

## 4.8 LAD-STL-FBD

OB1, FB FC STEP 7.

### ❖ LAD (LADDER Logic)

( 49).

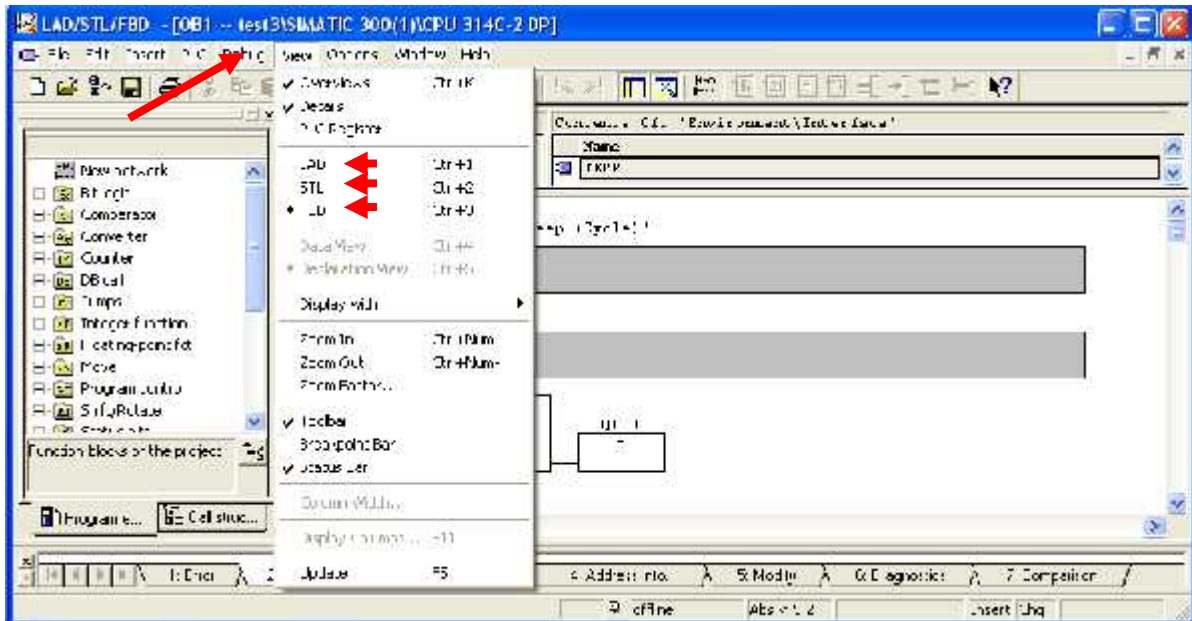


Εικόνα 49: Γλώσσα LAD



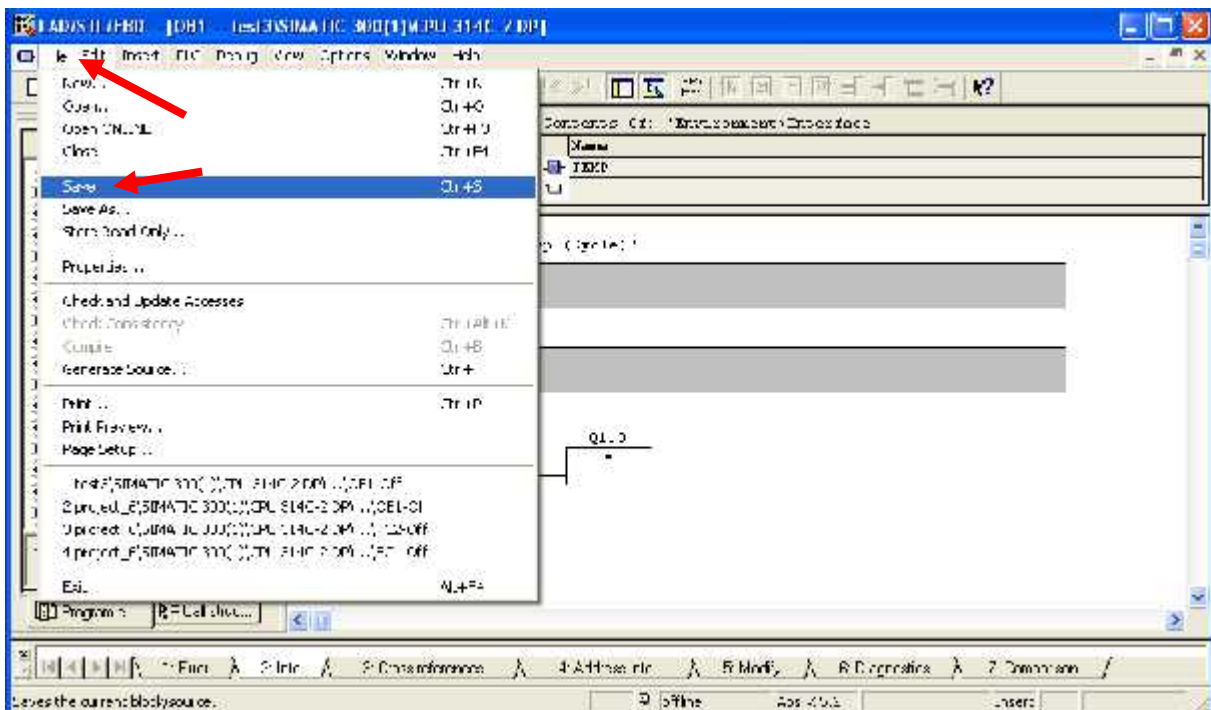


View > LAD/STL/FBD ( 52)



Εικόνα 52: Αλλαγή γλώσσας προγραμματισμού

File > Save ( 53).



Εικόνα 53: Αποθήκευση προγράμματος

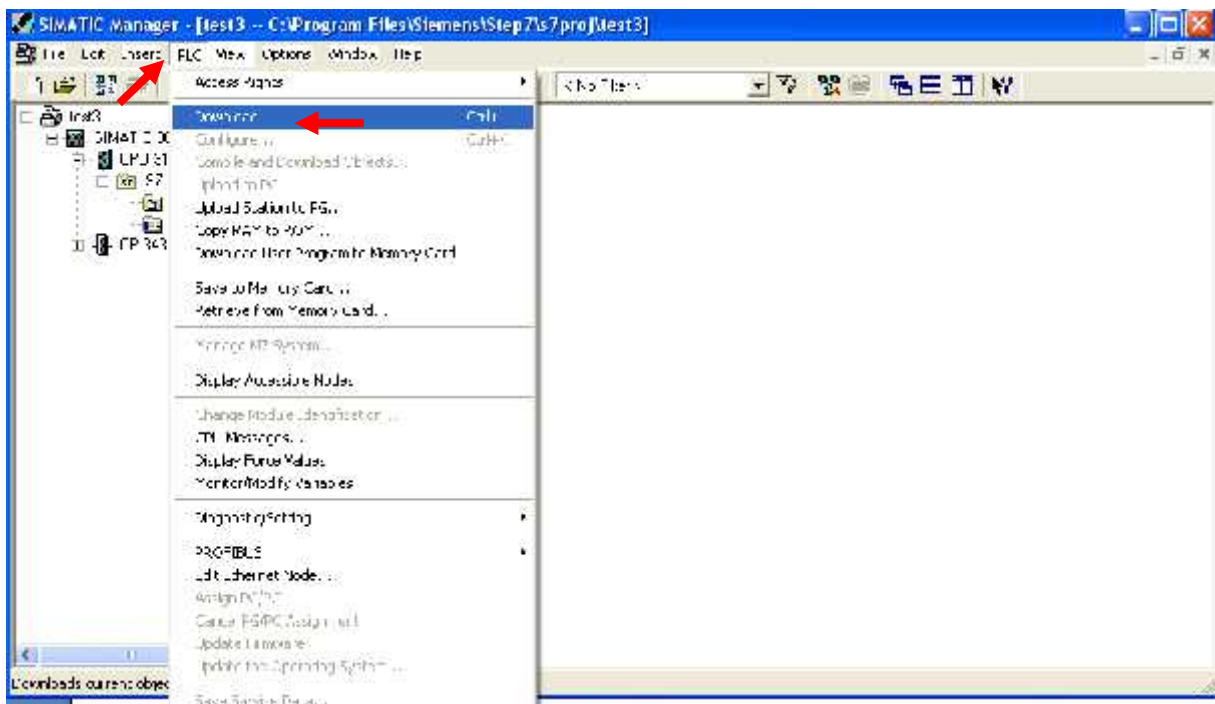
✓ , μ μ μ μ μ

IEC	SIMATIC	Description	Data Type	Address Range
I	E	Input bit	BOOL	0.0 to 65535.7
IB	FB	Input byte	BYTE, CHAR	0 to 65535
IW	EW	Input word	WORD, INT, S5TIME, DATE	0 to 65534
ID	ED	Input double word	DWORD, DINT, REAL, TOD, TIME	0 to 65532
Q	A	Output bit	BOOL	0.0 to 65535.7
QB	AB	Output byte	BYTE, CHAR	0 to 65535
QW	AW	Output word	WORD, INT, S5TIME, DATE	0 to 65534
QD	AD	Output double word	DWORD, DINT, REAL, TOD, TIME	0 to 65532
M	M	Memory bit	BOOL	0.0 to 65535.7
MB	MB	Memory byte	BYTE, CHAR	0 to 65535
MW	MW	Memory word	WORD, INT, S5TIME, DATE	0 to 65534
MD	MD	Memory double word	DWORD, DINT, REAL, TOD, TIME	0 to 65532
PIB	PEB	Peripheral input byte	BYTE, CHAR	0 to 65535
PQB	PAB	Peripheral output byte	BYTE, CHAR	0 to 65535
PIW	PEW	Peripheral input word	WORD, INT, S5TIME, DATE	0 to 65534
PQW	PAW	Peripheral output word	WORD, INT, S5TIME, DATE	0 to 65534
PID	PED	Peripheral input double word	DWORD, DINT, REAL, TOD, TIME	0 to 65532
PQD	PAD	Peripheral output double word	DWORD, DINT, REAL, TOD, TIME	0 to 65532
T	T	Timer	TIMER	0 to 65535
C	Z	Counter	COUNTER	0 to 65535
FR	FB	Function block	FB	0 to 65535
OB	OB	Organization block	OB	1 to 65535
DB	DB	Data block	DB, FB, SFB, UDT	1 to 65535
FC	FC	Function	FC	0 to 65535
SFB	SFB	System function block	SFB	0 to 65535
SFC	SFC	System function	SFC	0 to 65535
VAT	VAT	Variable table		0 to 65535
UDT	UDT	User-defined data type	UDT	0 to 65535

Εικόνα 54: Τύποι μεταβλητών

## 4.8.1 PLC

blocks  
PLC.  
Simatic Manager,  
, Blocks PLC > Download. ( 55)

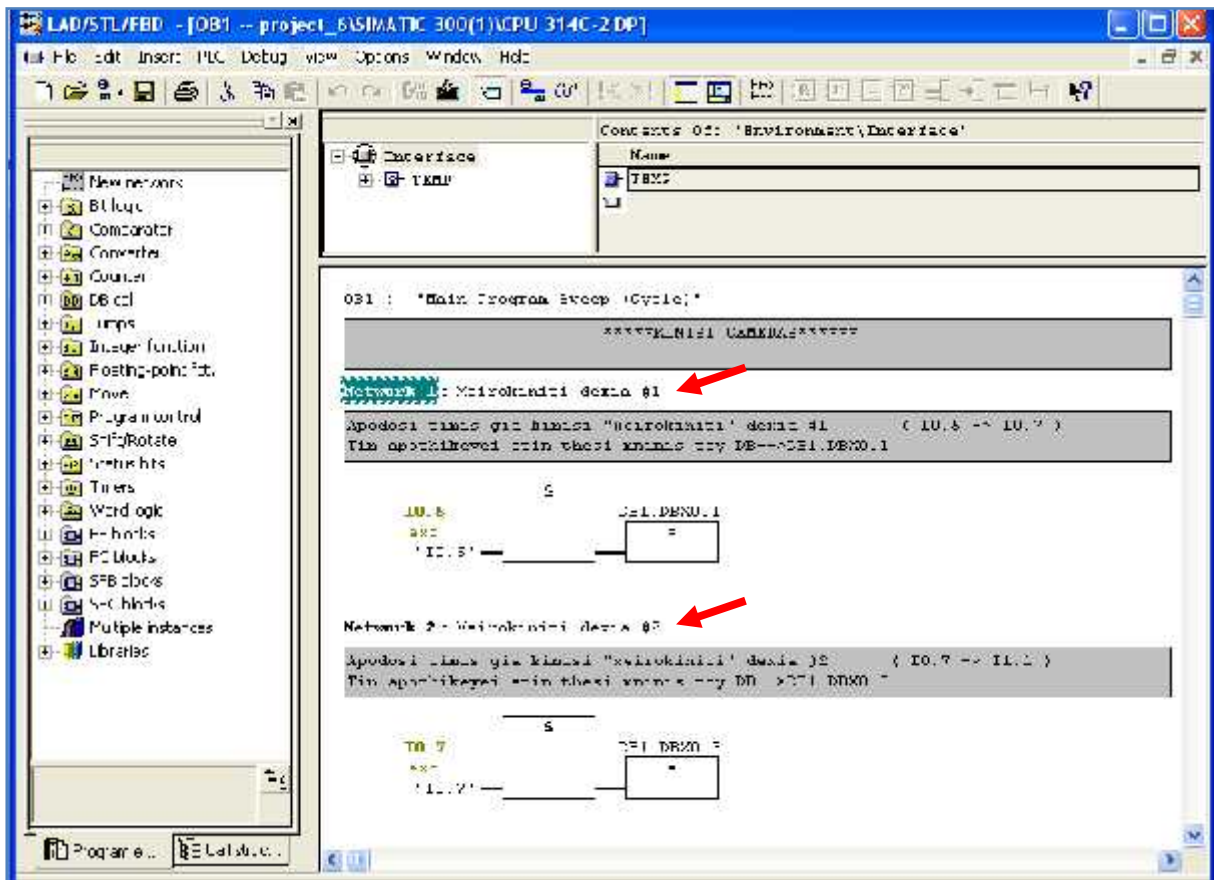


Εικόνα 55: PLC Download

## 4.9

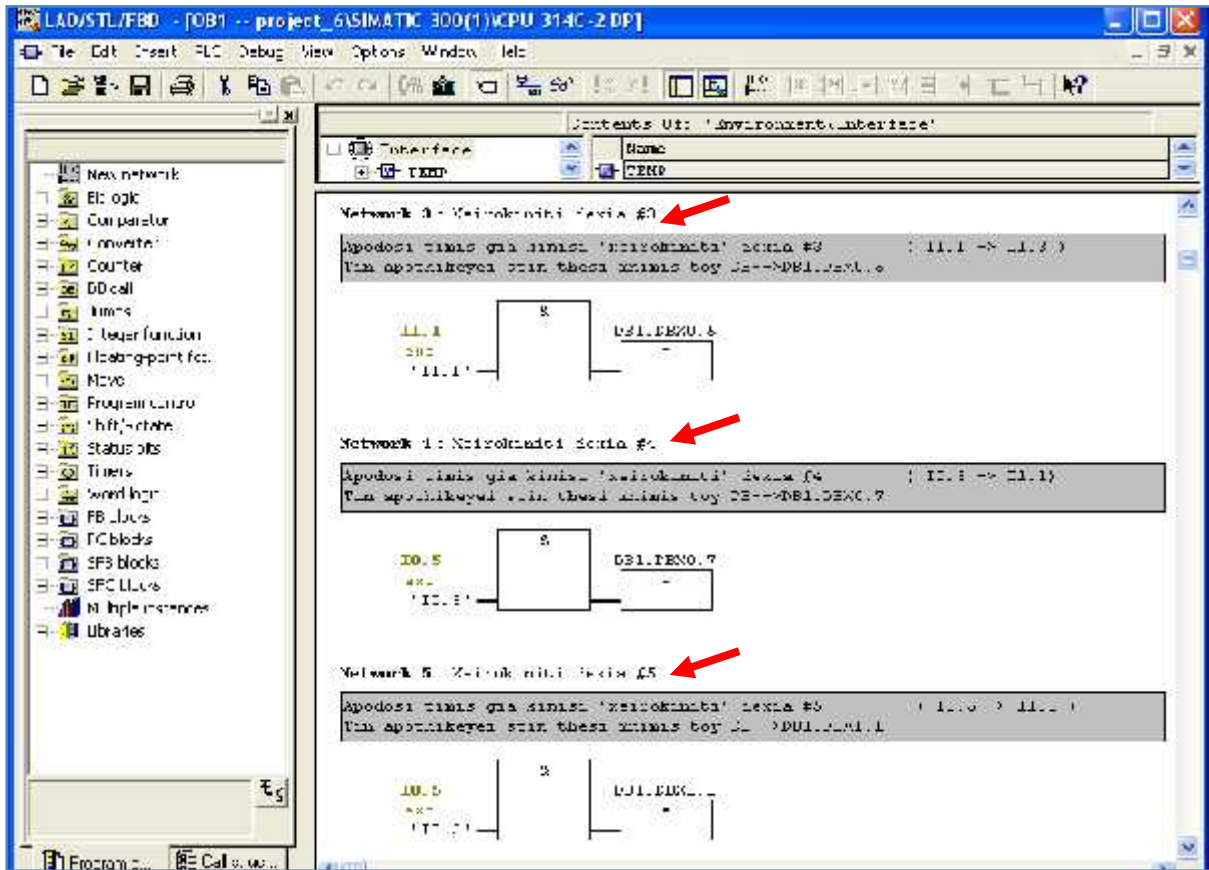
## OB1-FC-DB1

- OB1 μ , μ μ μ . μ .
  - Network 1 μ Network 12, μ μ μ μ
  - Network 20 31 μ μ DB1.
- ✓ μ

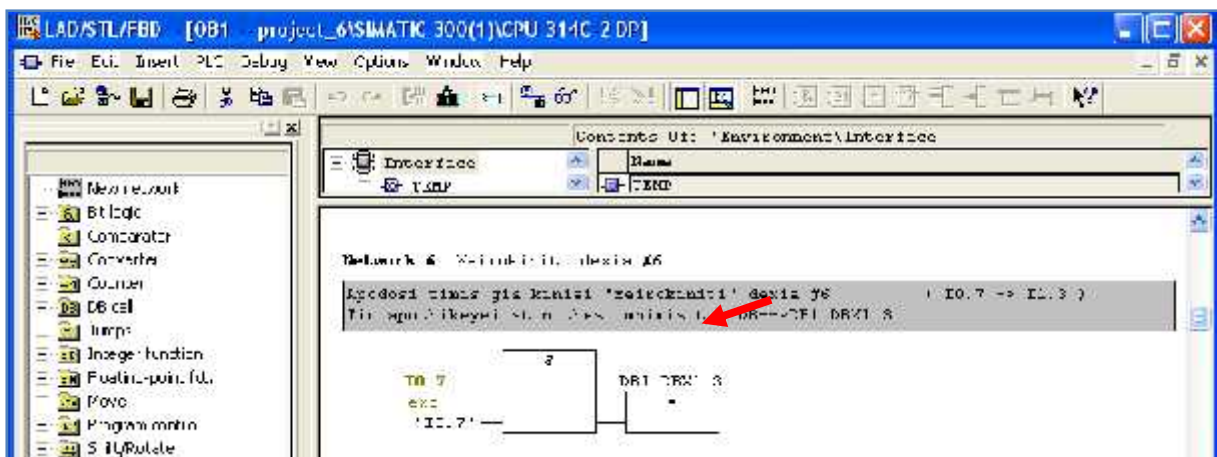


Εικόνα 56: Απόδοση τιμής για χειροκίνητη δεξιά κίνηση # 1-2

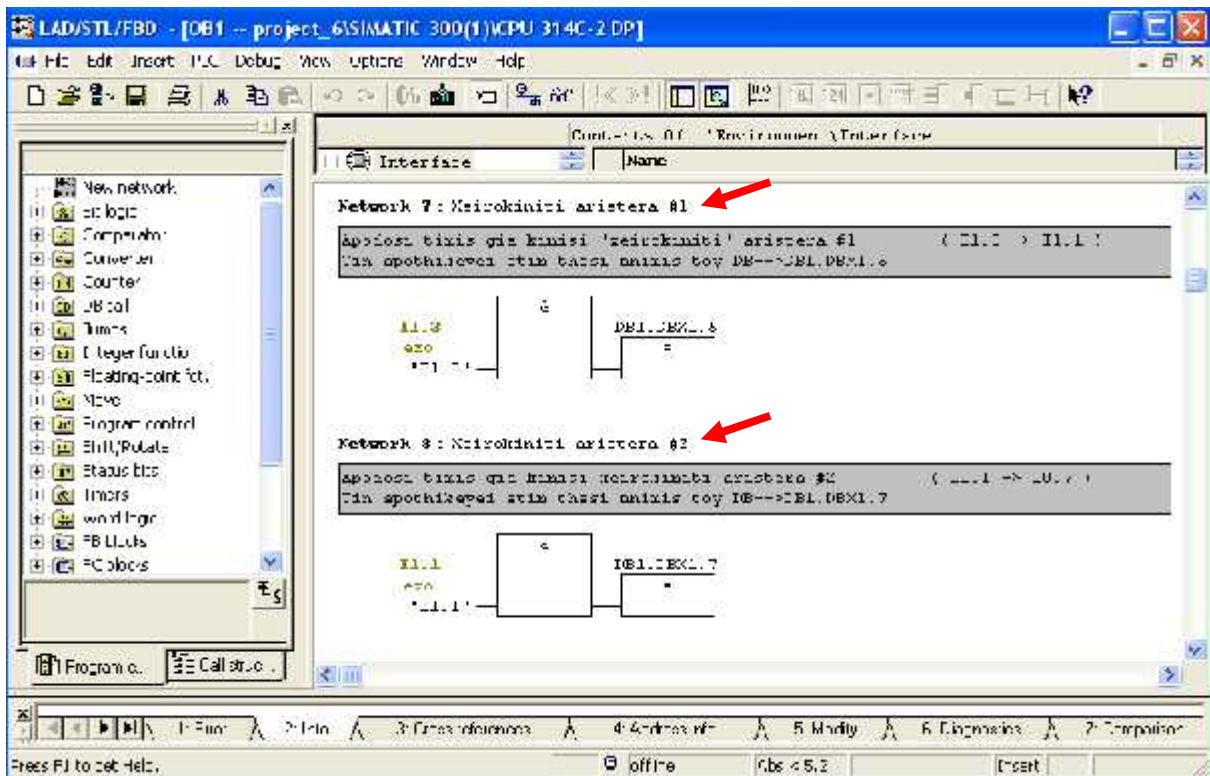
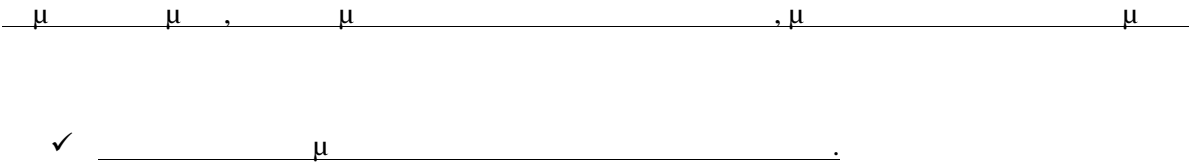
✓



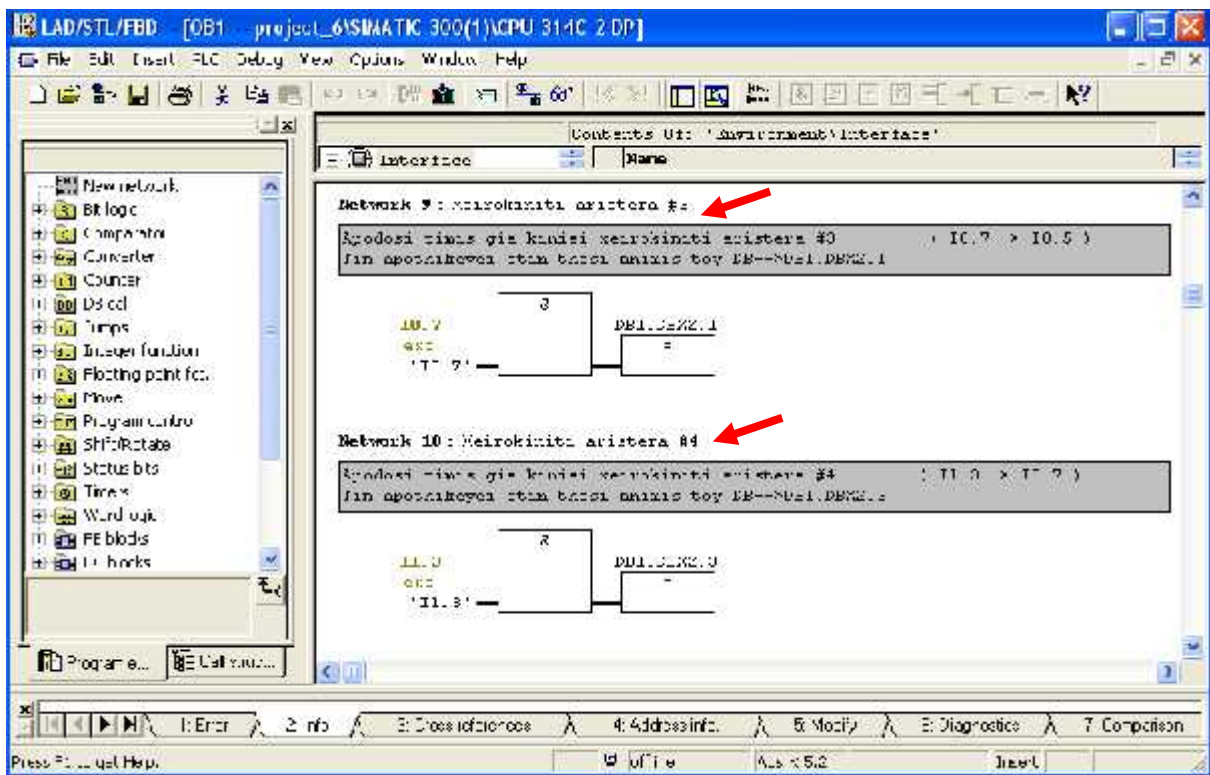
Εικόνα 57: Απόδοση τιμής για χειροκίνητη δεξιά κίνηση # 3-4-5



Εικόνα 58: Απόδοση τιμής για χειροκίνητη δεξιά κίνηση # 6

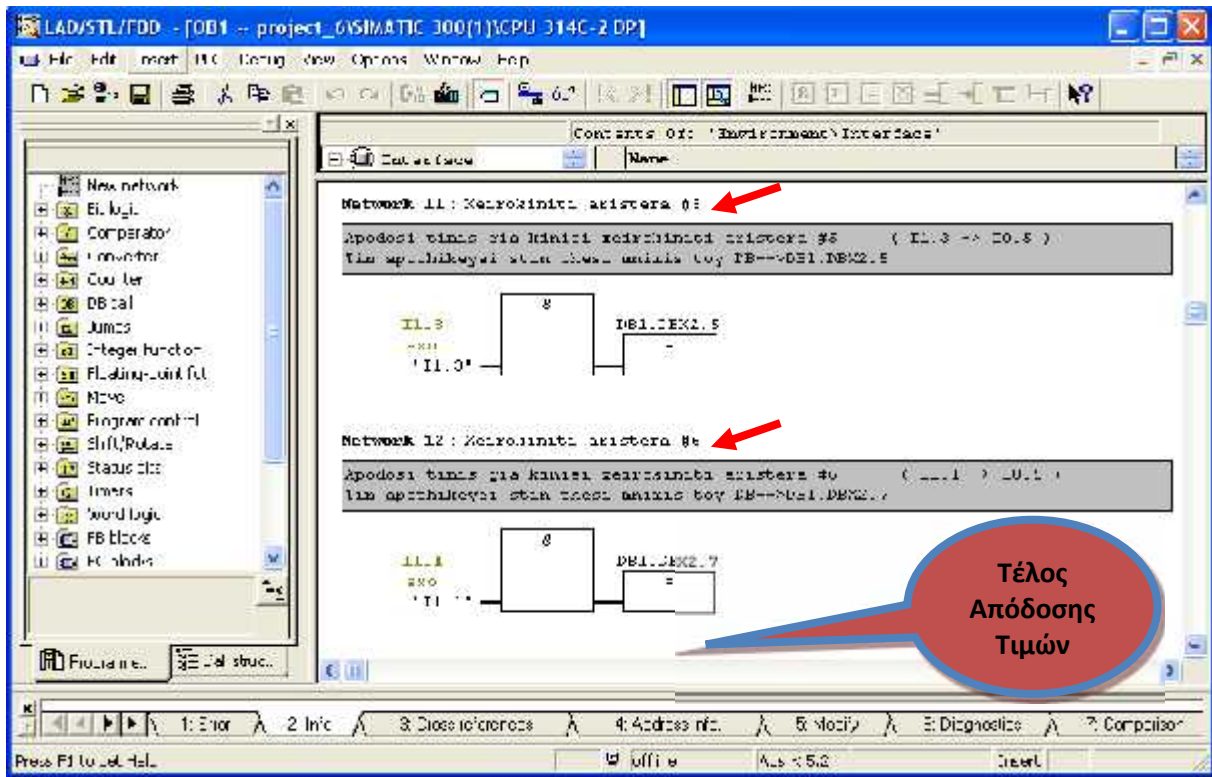


Εικόνα 59: Απόδοση τιμής για χειροκίνητη αριστερή κίνηση # 1-2



Εικόνα 60: Απόδοση τιμής για χειροκίνητη αριστερή κίνηση # 3-4

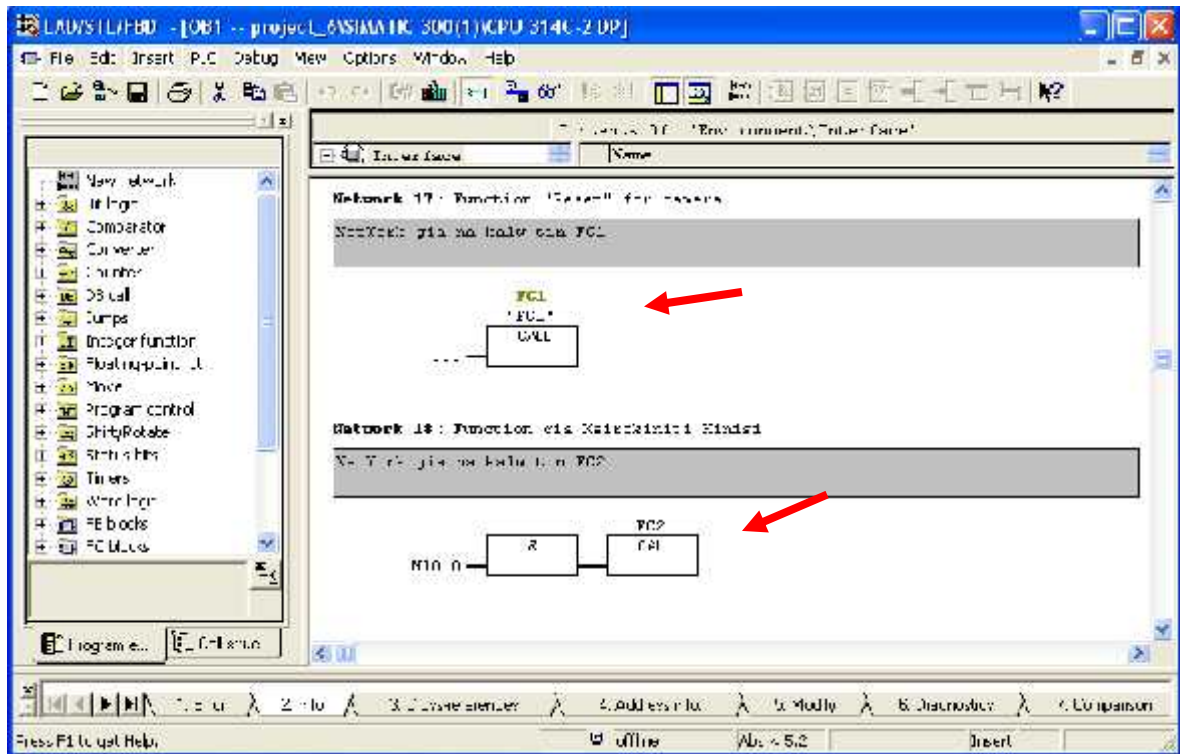
✓ Network 12 ( 60).



Εικόνα 60: Απόδοση τιμής για χειροκίνητη αριστερή κίνηση # 5-6

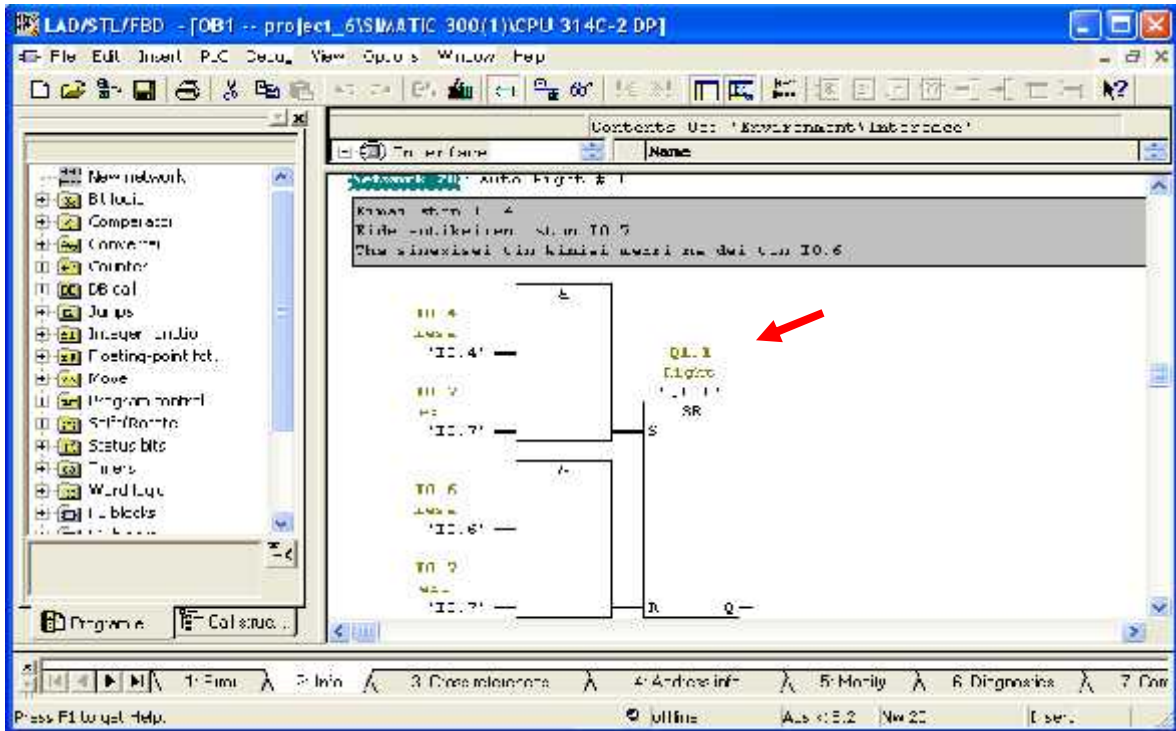


➤ Network 17 18 μ μ  
μ μ μ . FC1 FC2. μ μ μ

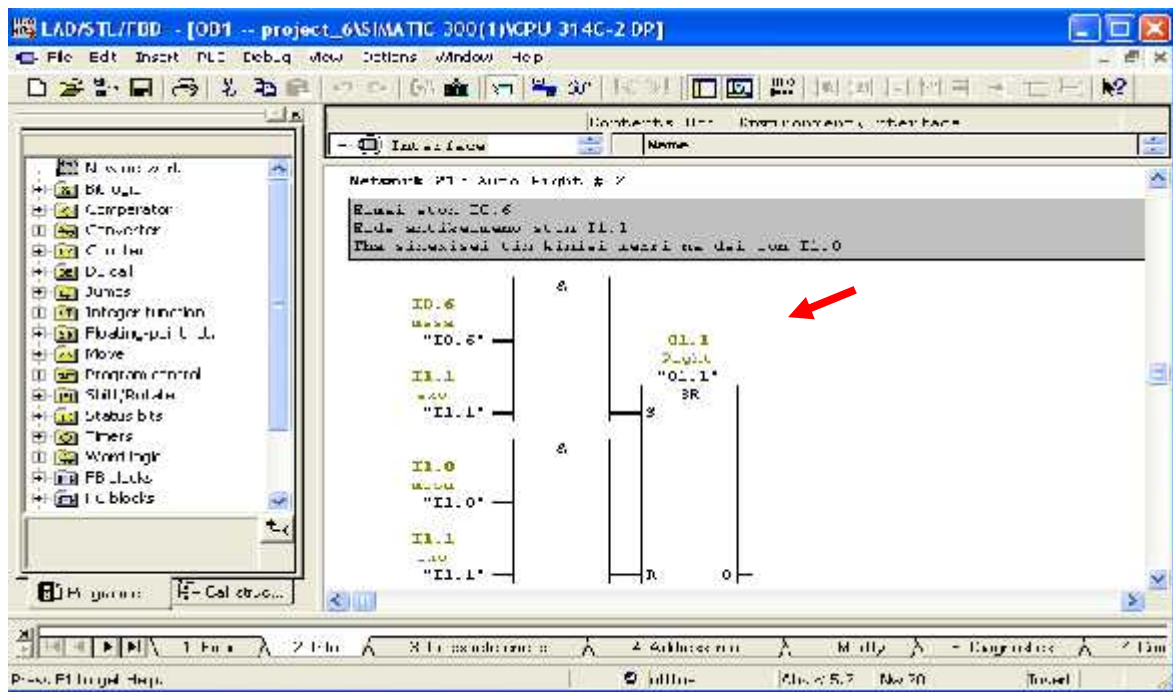


Εικόνα 61: Δήλωση FC1 και FC2

- Network 20 31, μ μ μ μ μ
- ✓ μ

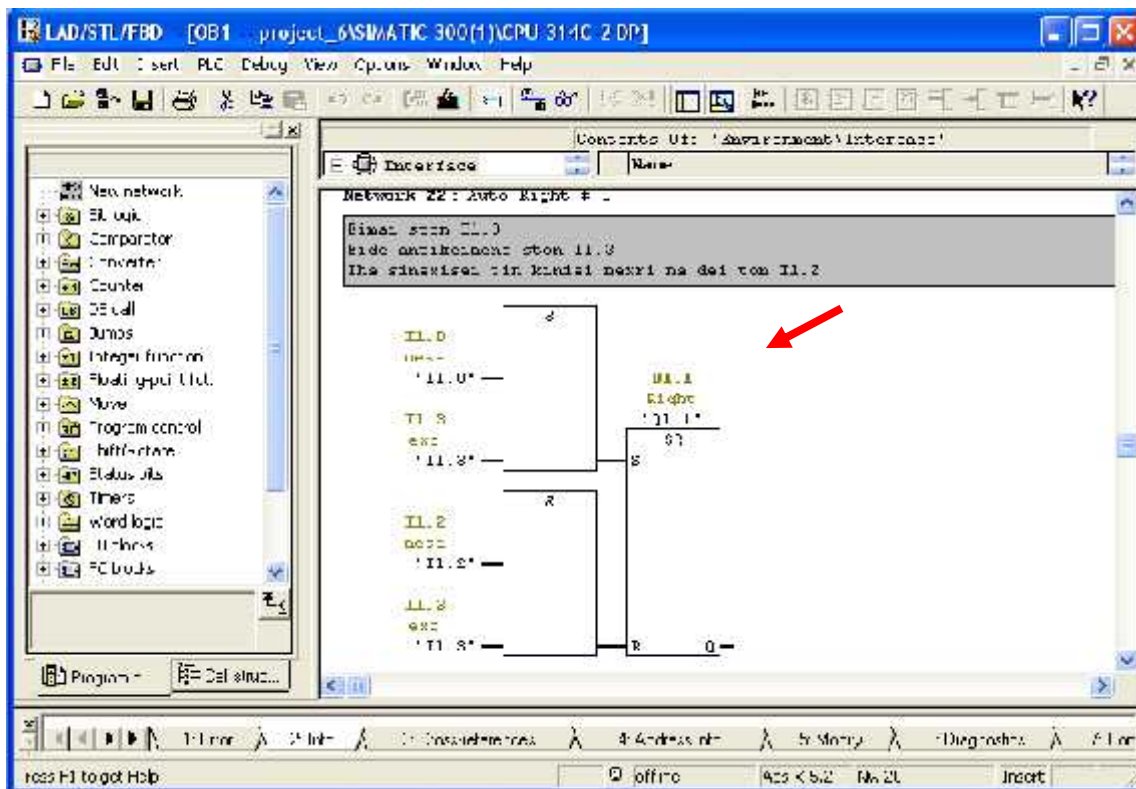


Εικόνα 62: Αυτόματη δεξιά κίνηση #1

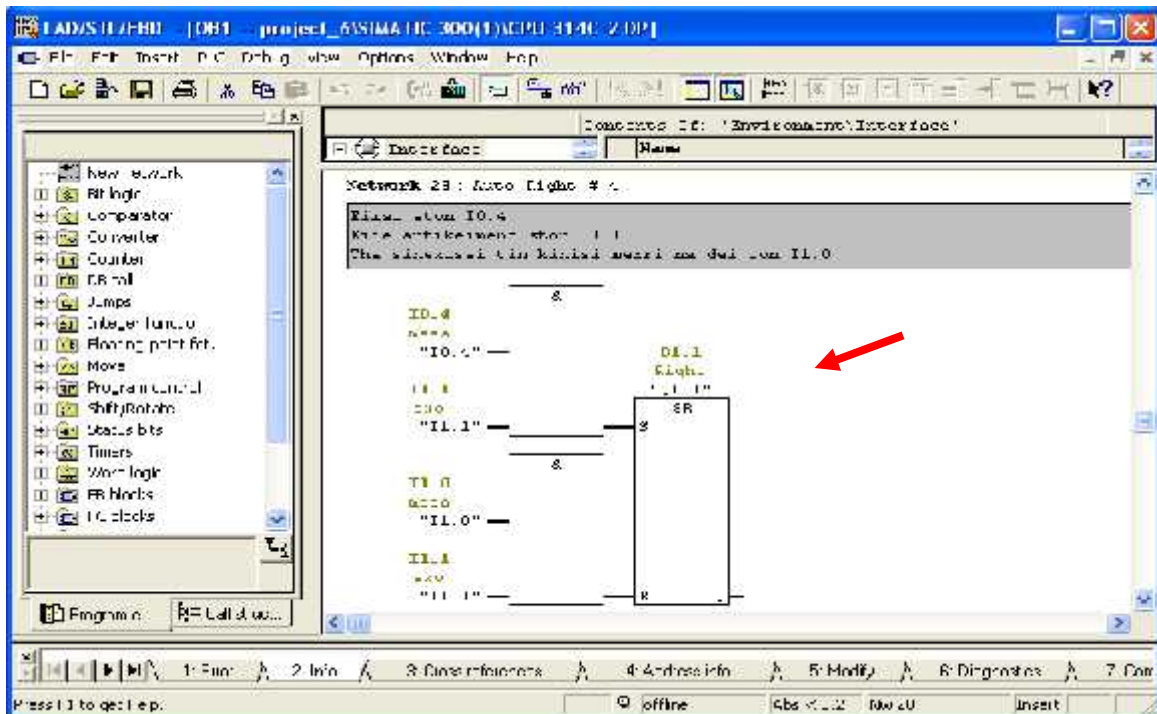


Εικόνα 63: Αυτόματη δεξιά κίνηση #2

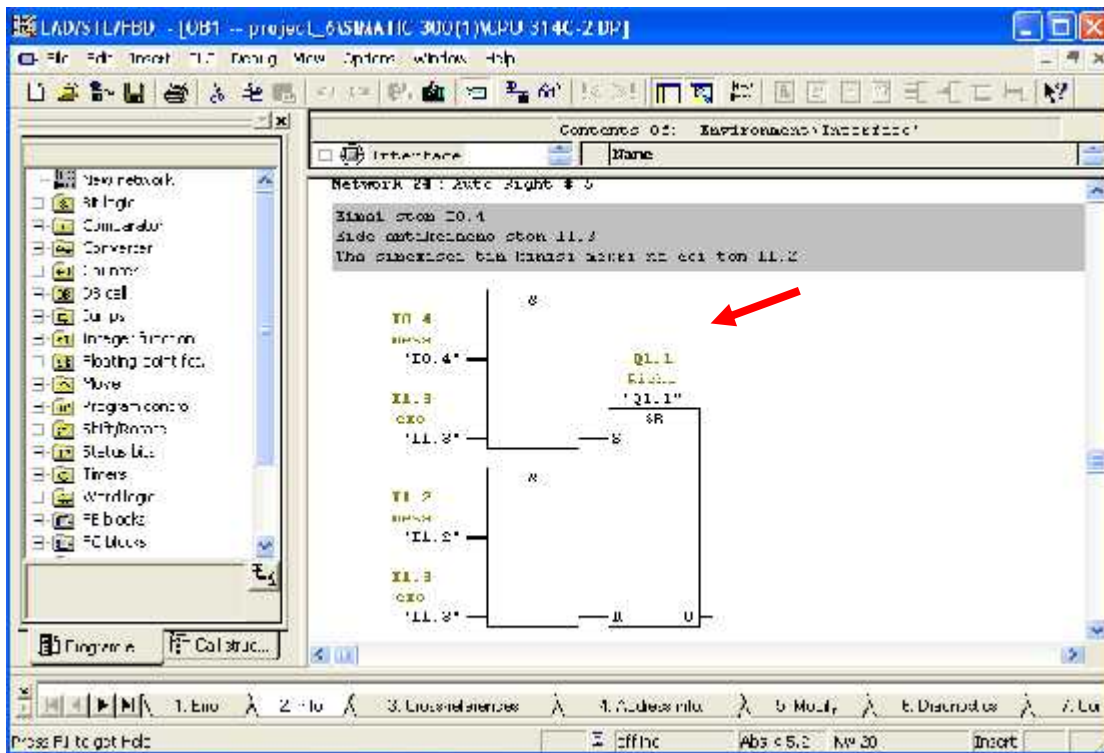
✓ μ



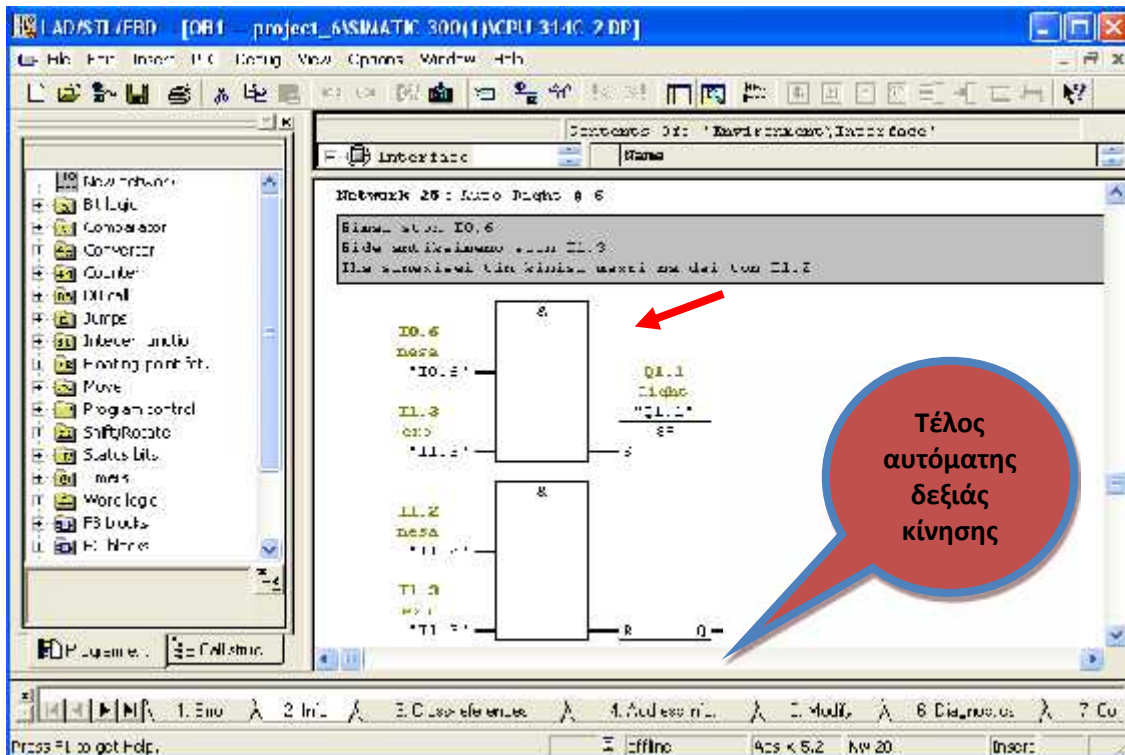
Εικόνα 64: Αυτόματη δεξιά κίνηση #3



Εικόνα 65: Αυτόματη δεξιά κίνηση #4

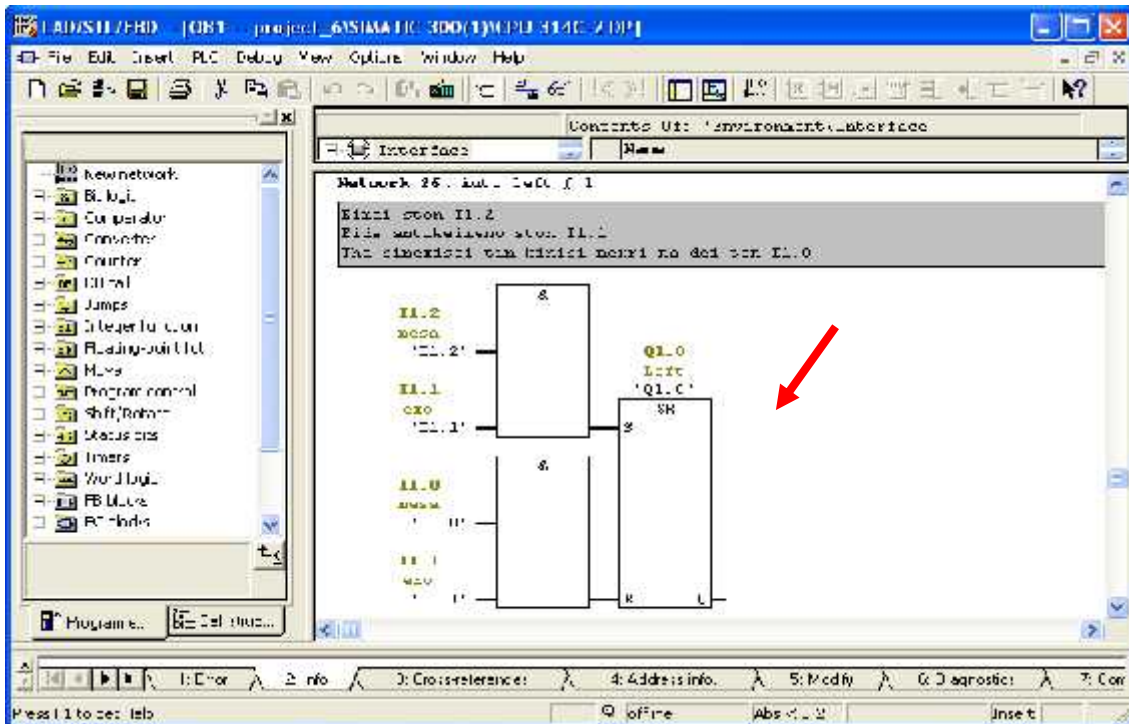


Εικόνα 65: Αυτόματη δεξιά κίνηση #5

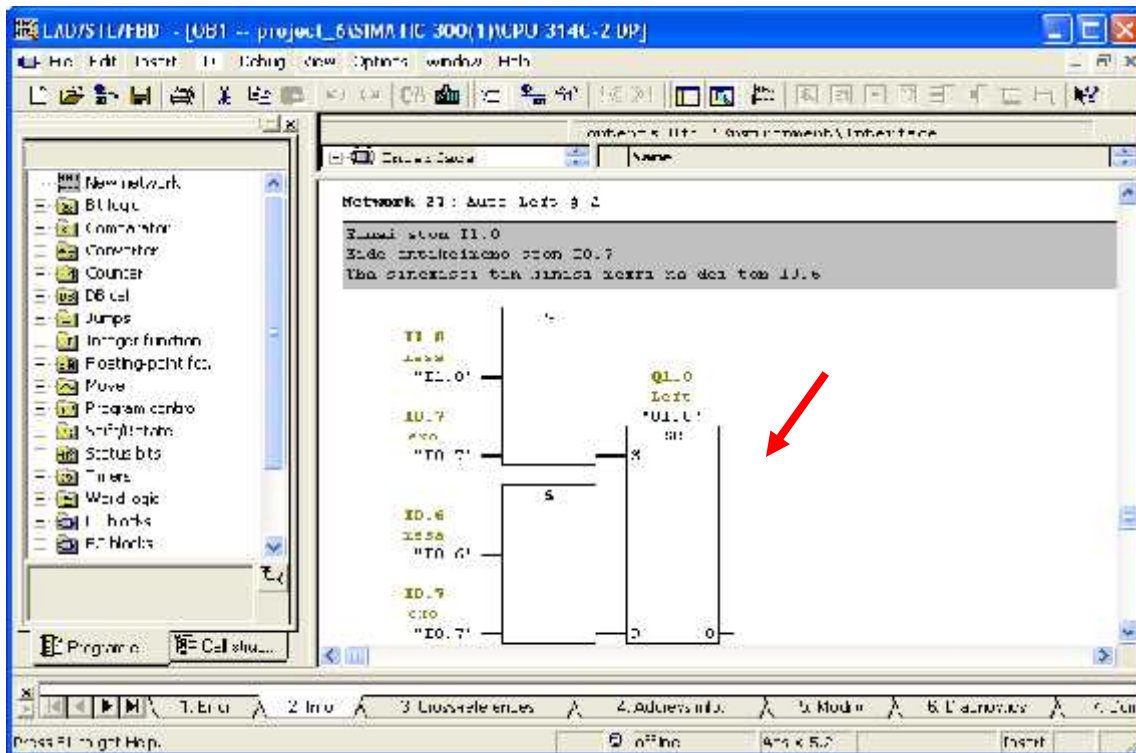


Εικόνα 66: Αυτόματη δεξιά κίνηση #6

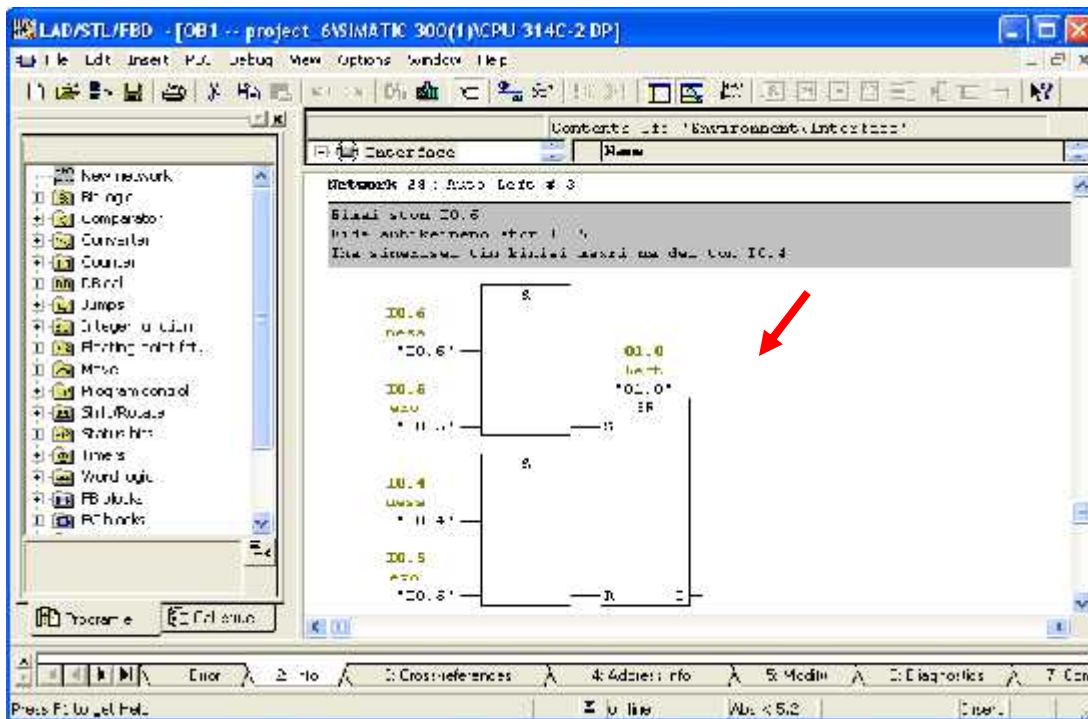
✓ \_\_\_\_\_ μ \_\_\_\_\_



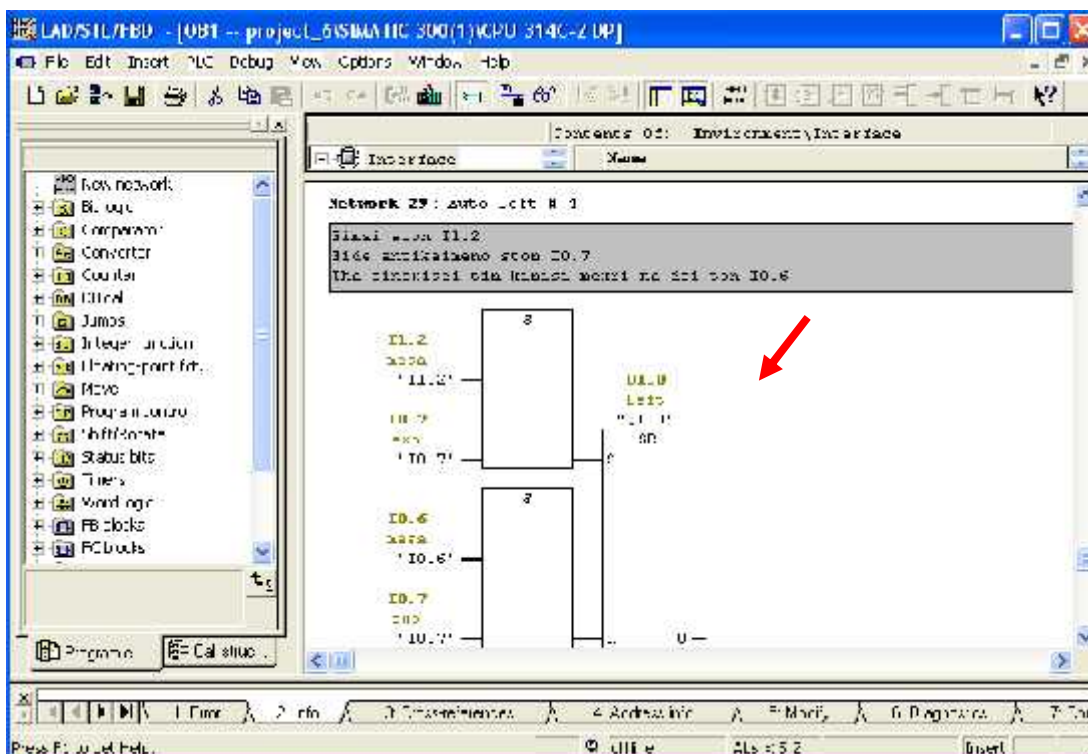
Εικόνα 67: Αυτόματη αριστερή κίνηση #1



Εικόνα 68: Αυτόματη αριστερή κίνηση #2

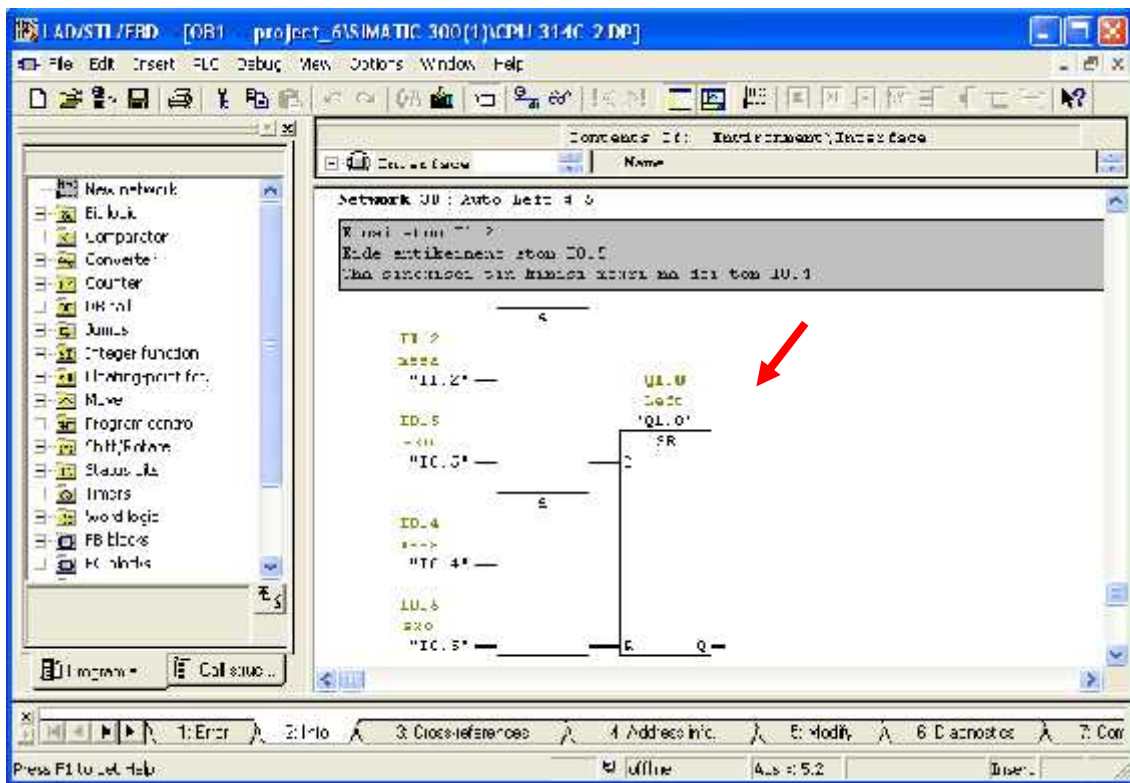


Εικόνα 69: Αυτόματη αριστερή κίνηση #3

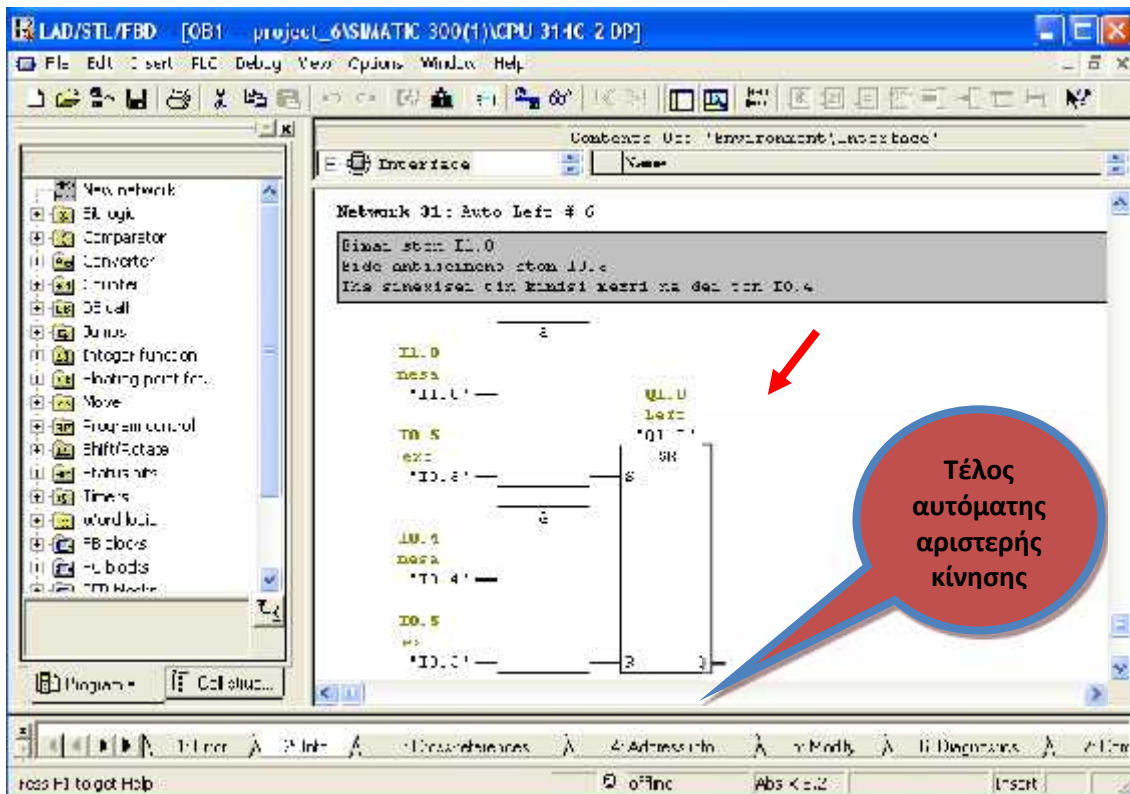


Εικόνα 70: Αυτόματη αριστερή κίνηση #4

✓ Network 31 ( 72)



Εικόνα 71: Αυτόματη αριστερή κίνηση #5

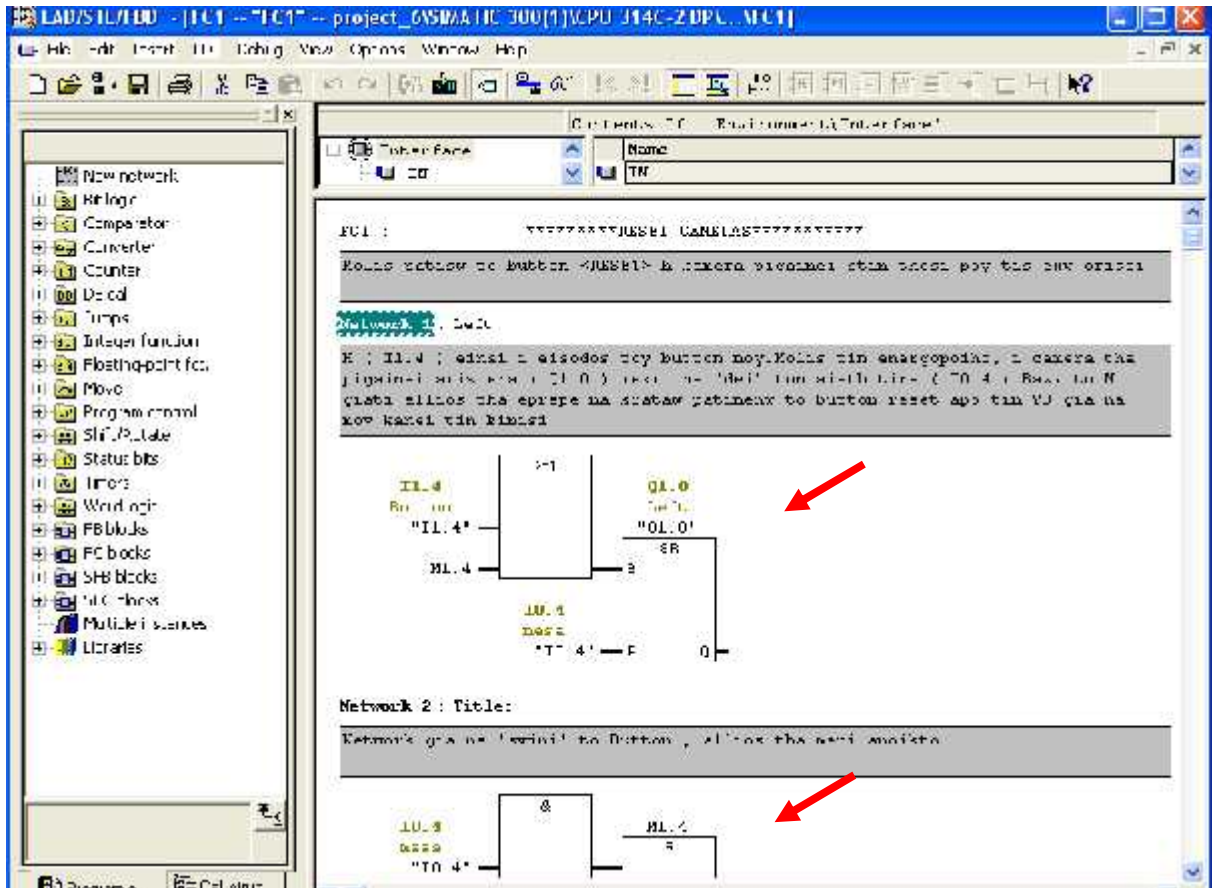


Εικόνα 72: Αυτόματη αριστερή κίνηση #6

➤ **FC1**

Network.

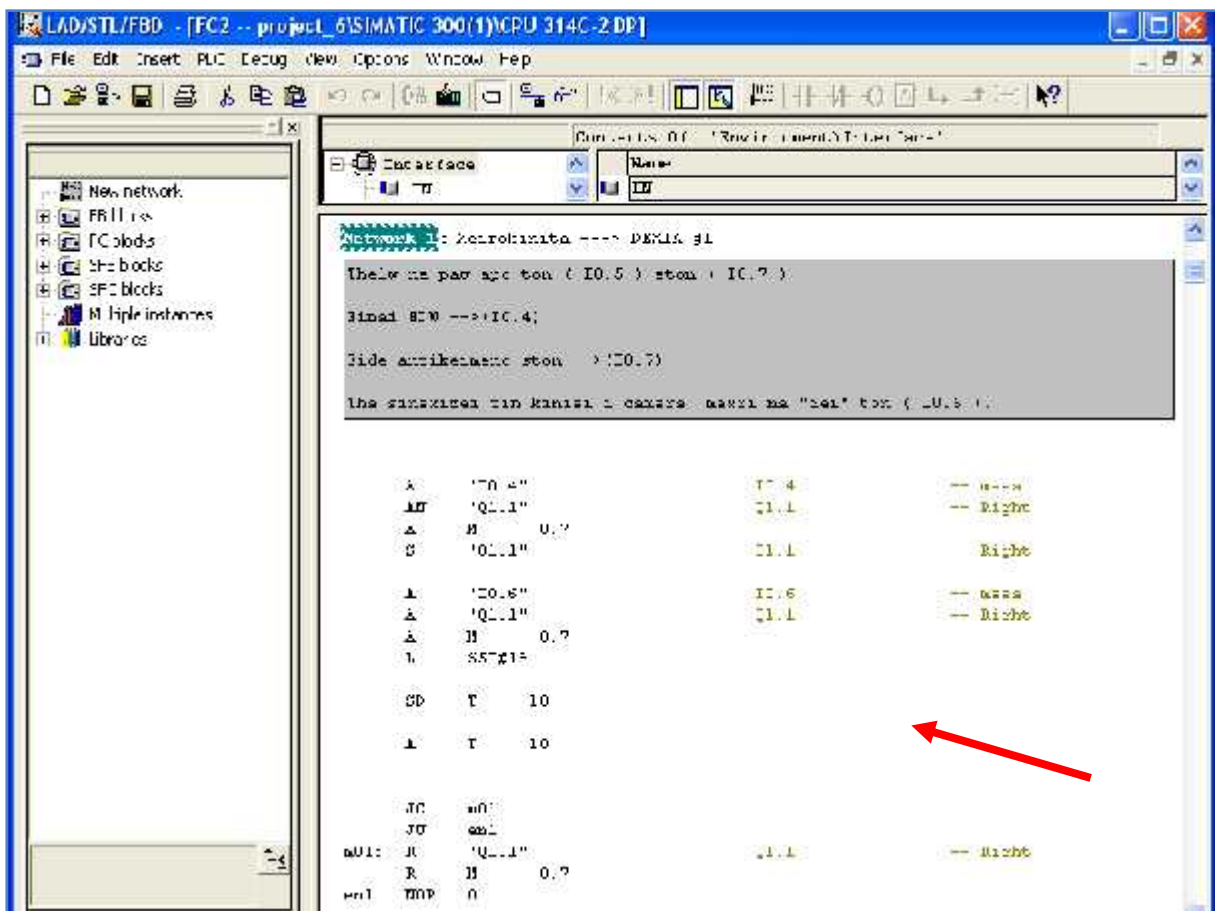
Button application Visual Basic  
Network, 0. Button (73).



Εικόνα 73: Ορισμός της FC1

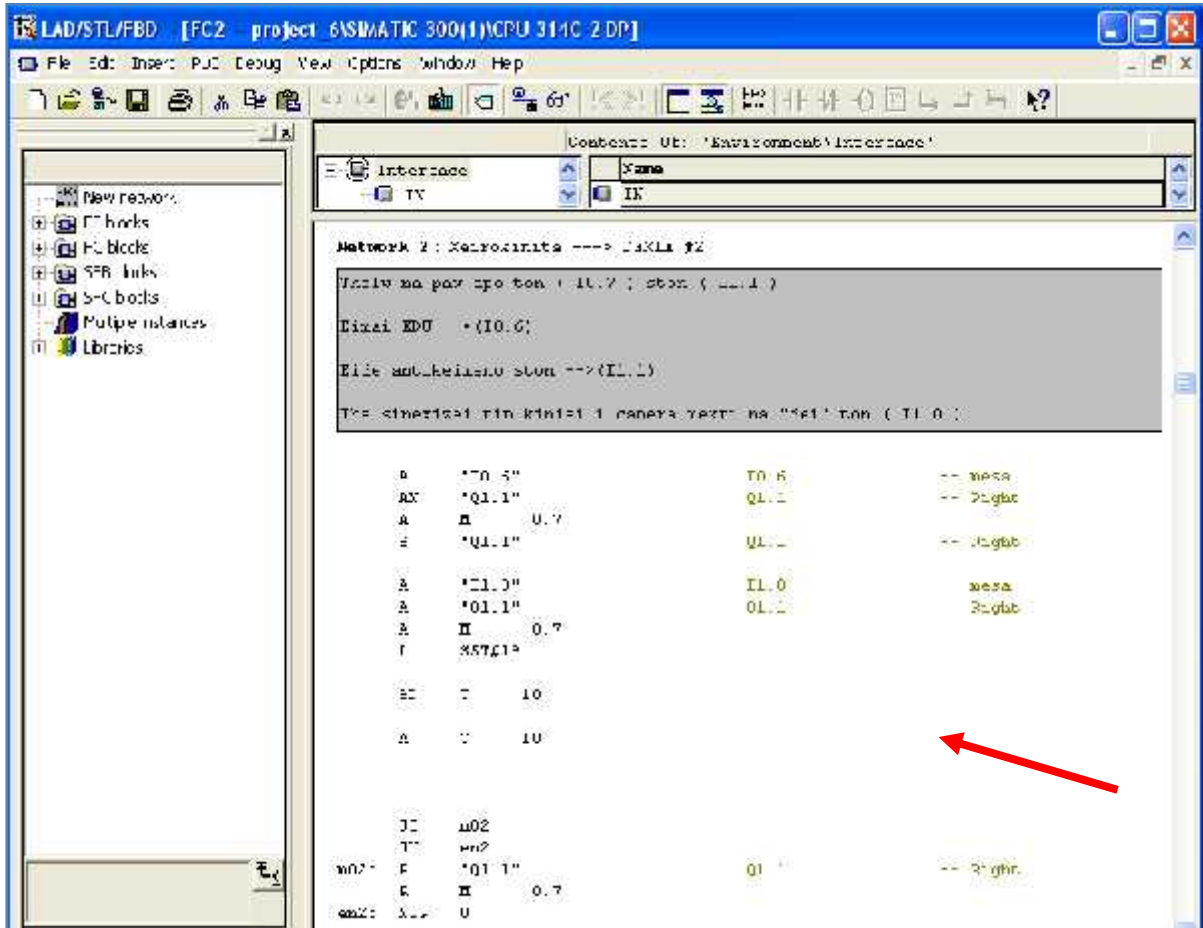


- Η FC2
    - μ Visual Basic.
    - μ application
    - μ Network.
    - μ STL,
    - μ FDB.
- ✓ \_\_\_\_\_



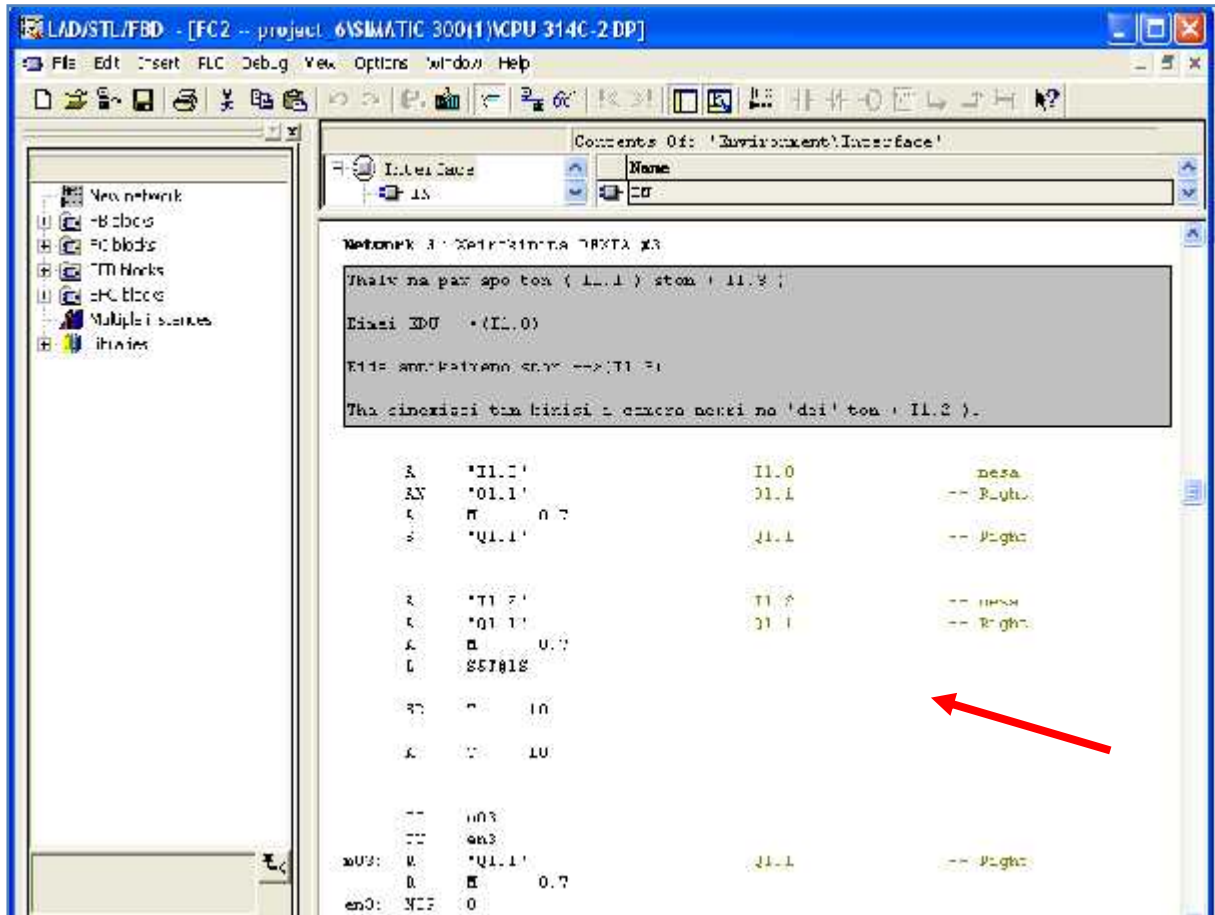
Εικόνα 74: Ορισμός στη FC2 για χειροκίνητη δεξιά κίνηση #1

✓



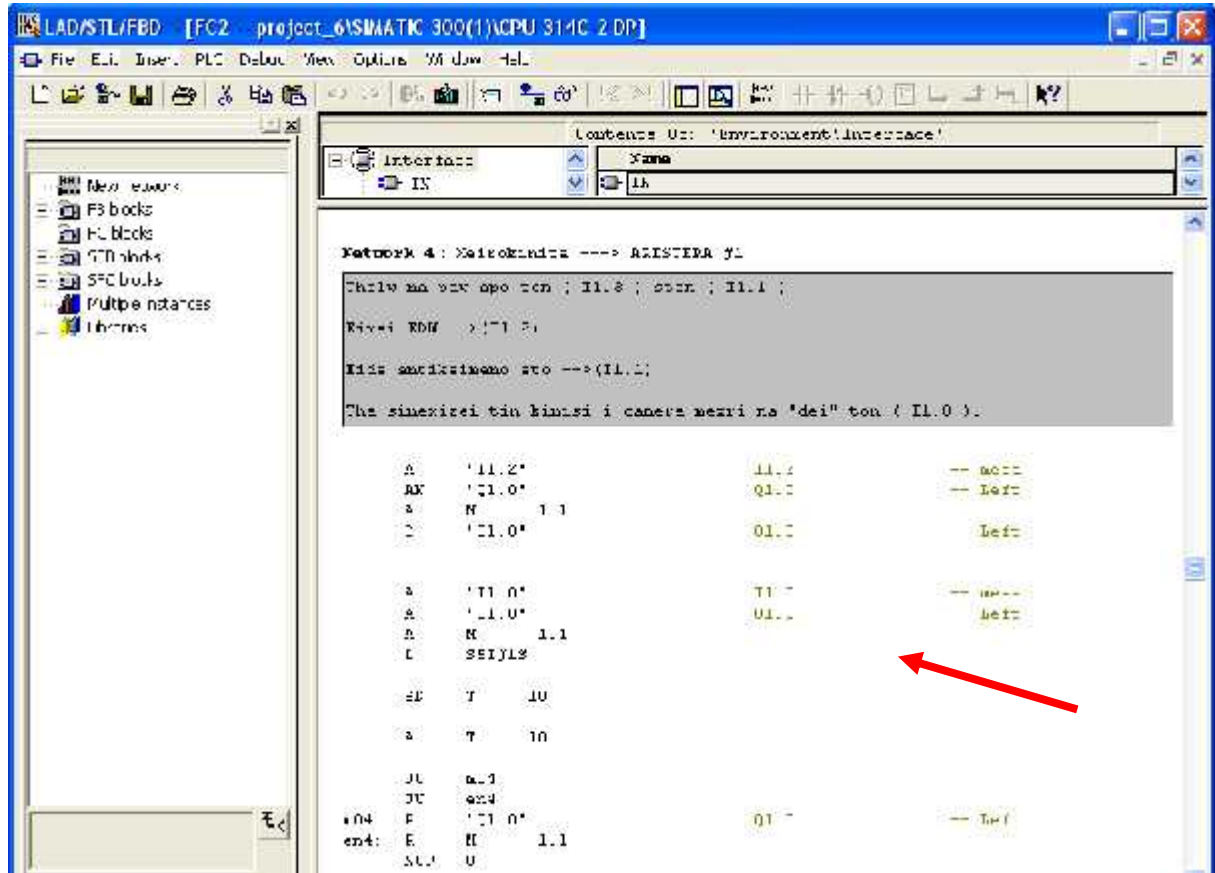
Εικόνα 75: Ορισμός στη FC2 για χειροκίνητη δεξιά κίνηση #2

✓



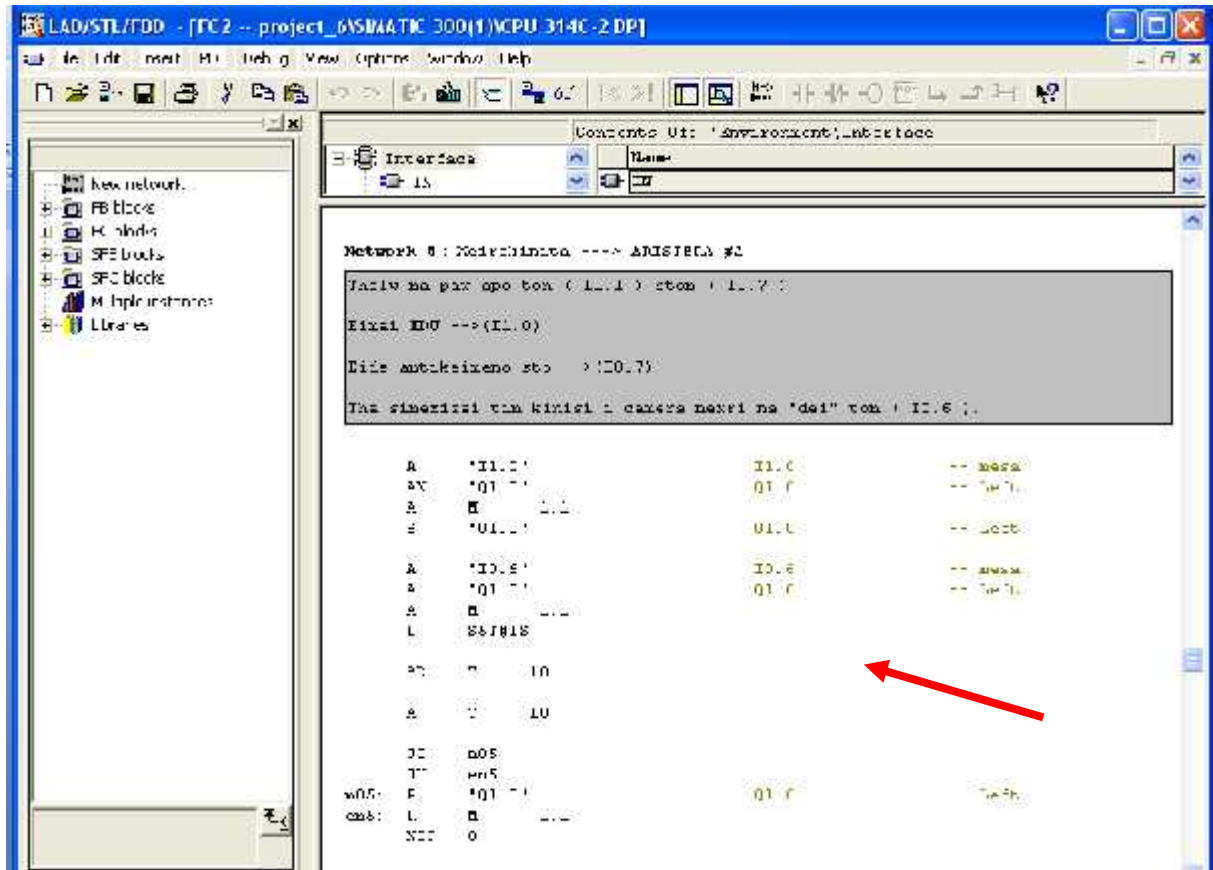
Εικόνα 76: Ορισμός στη FC2 για χειροκίνητη δεξιά κίνηση #3

✓



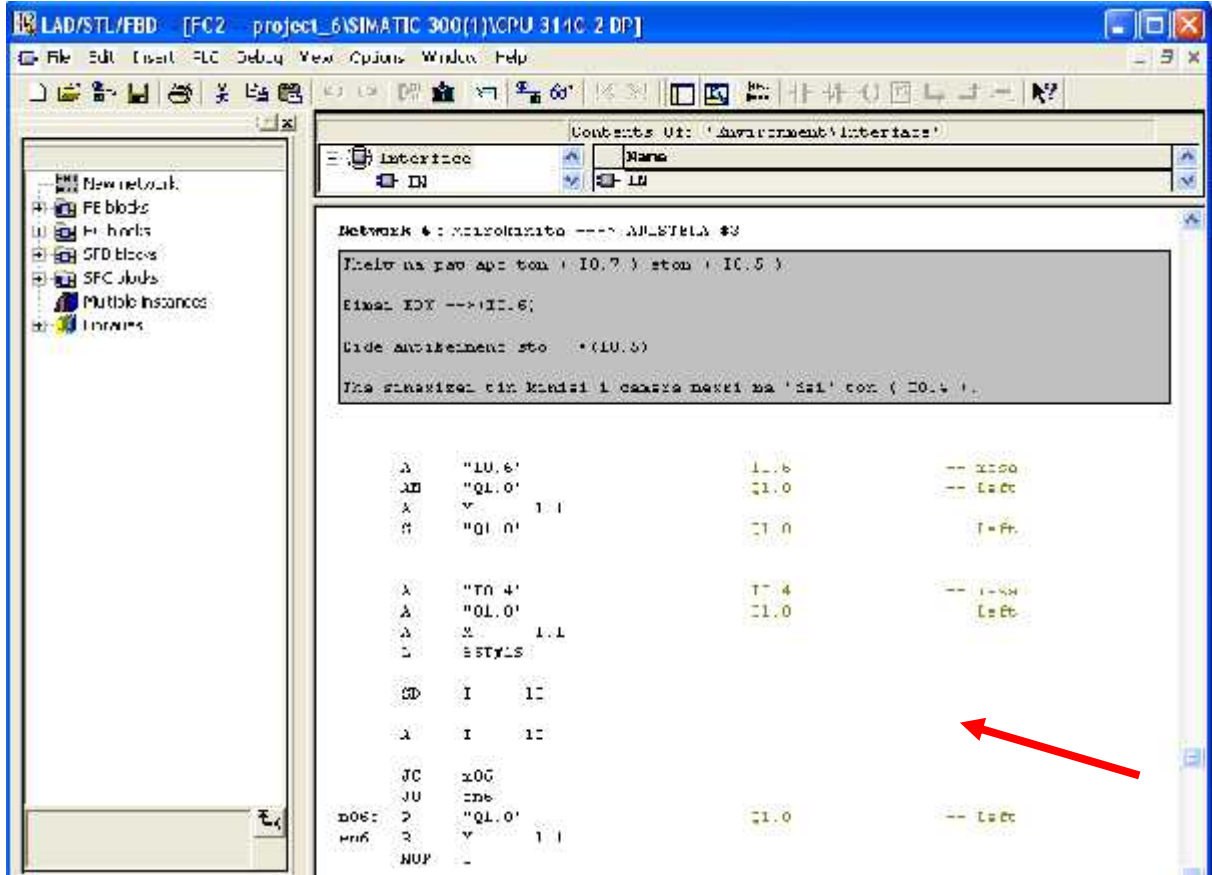
Εικόνα 76: Ορισμός στη FC2 για χειροκίνητη αριστερή κίνηση #1

✓



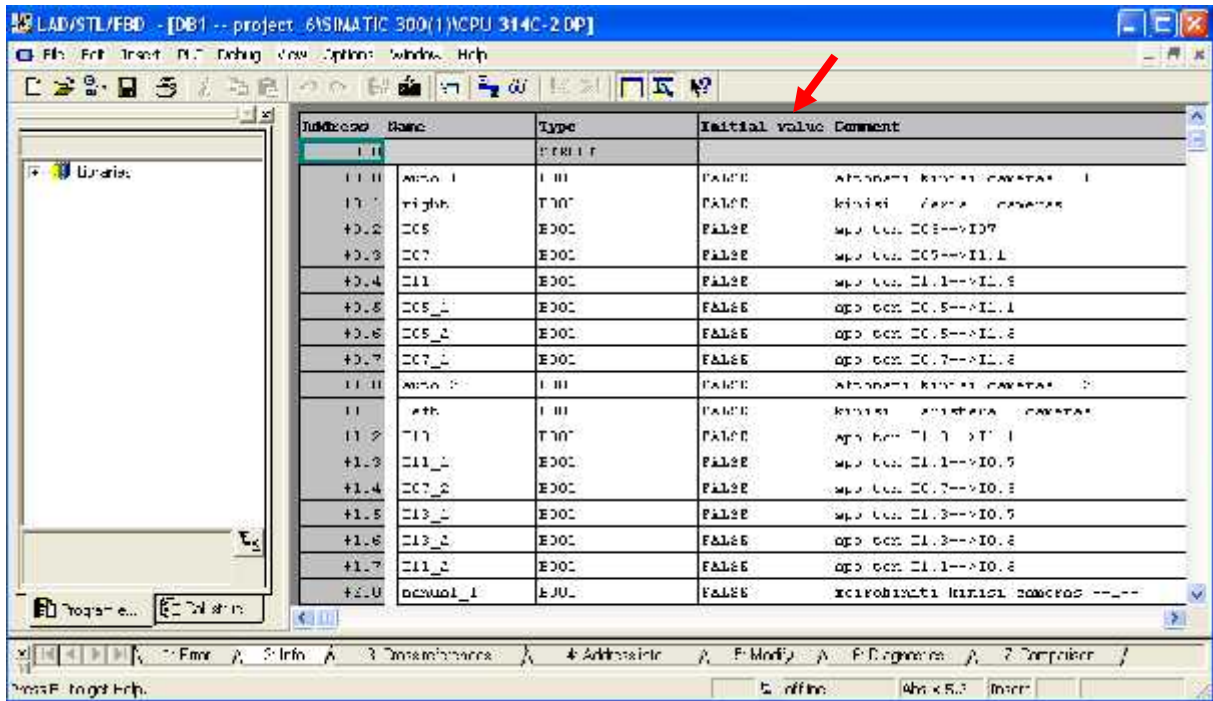
Εικόνα 77: Ορισμός στη FC2 για χειροκίνητη αριστερή κίνηση #2

✓

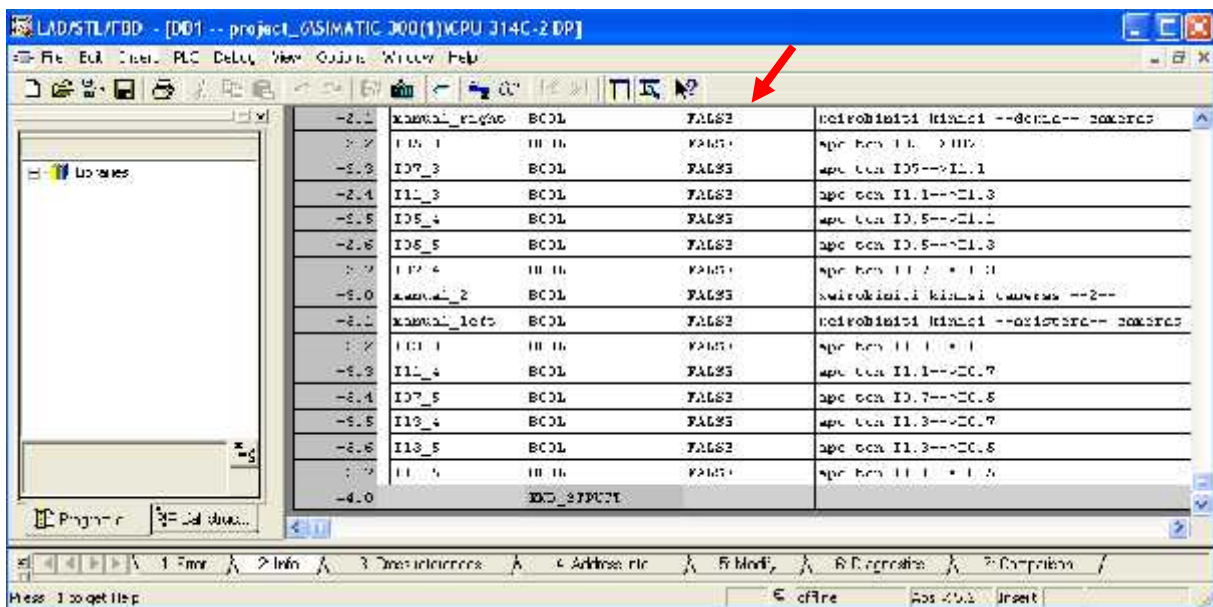


Εικόνα 78: Ορισμός στη FC2 για χειροκίνητη αριστερή κίνηση #3

- DB1 block , Initial Value, FALSE TRUE.



Εικόνα 79: Περιεχόμενα του DB



Εικόνα 80: Συνέχεια των περιεχομένων του DB

5

6.1 Visual Basic

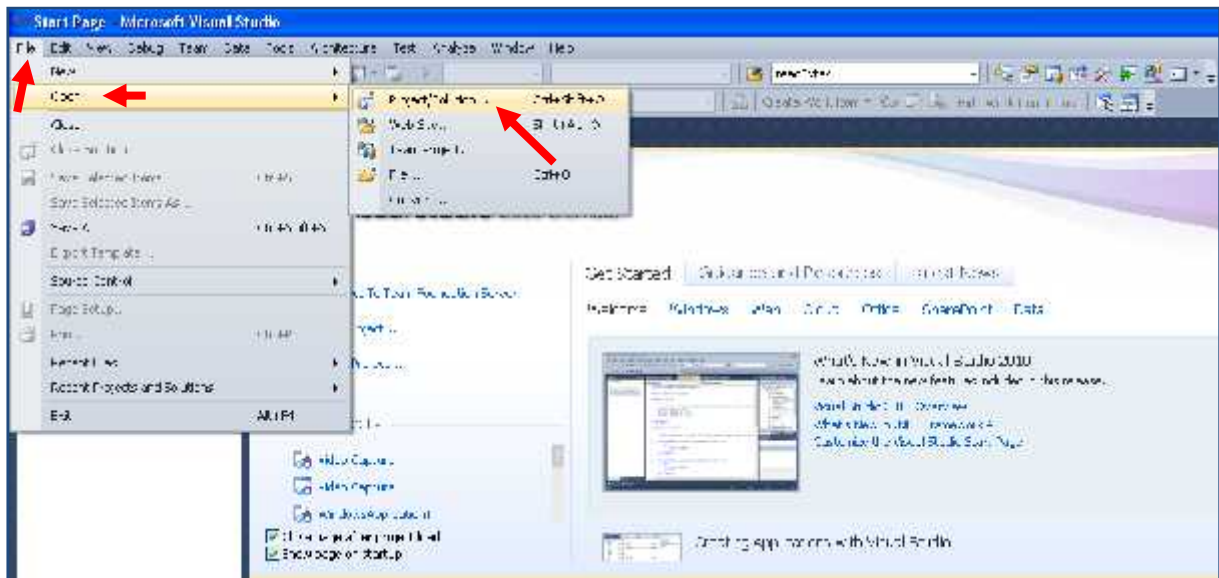
μ application μ μ VB μ μ .

μ 1 :



μ 2 :

μ μ (Start Page), μ : File > Open > Project/Solution...



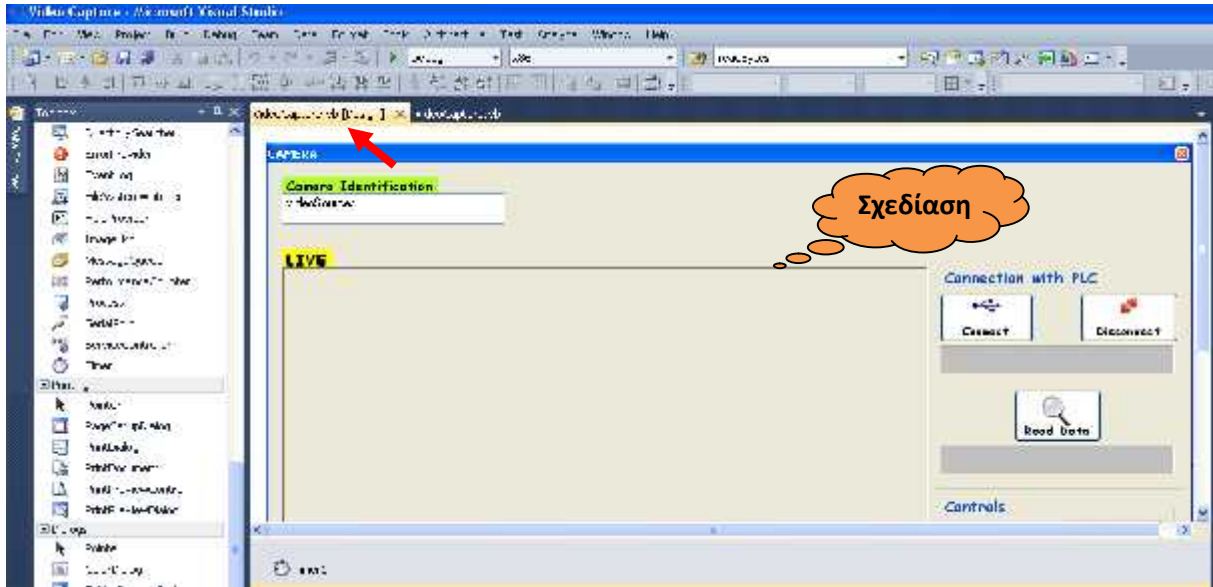
Εικόνα 81: Εύρεση του project στη VB



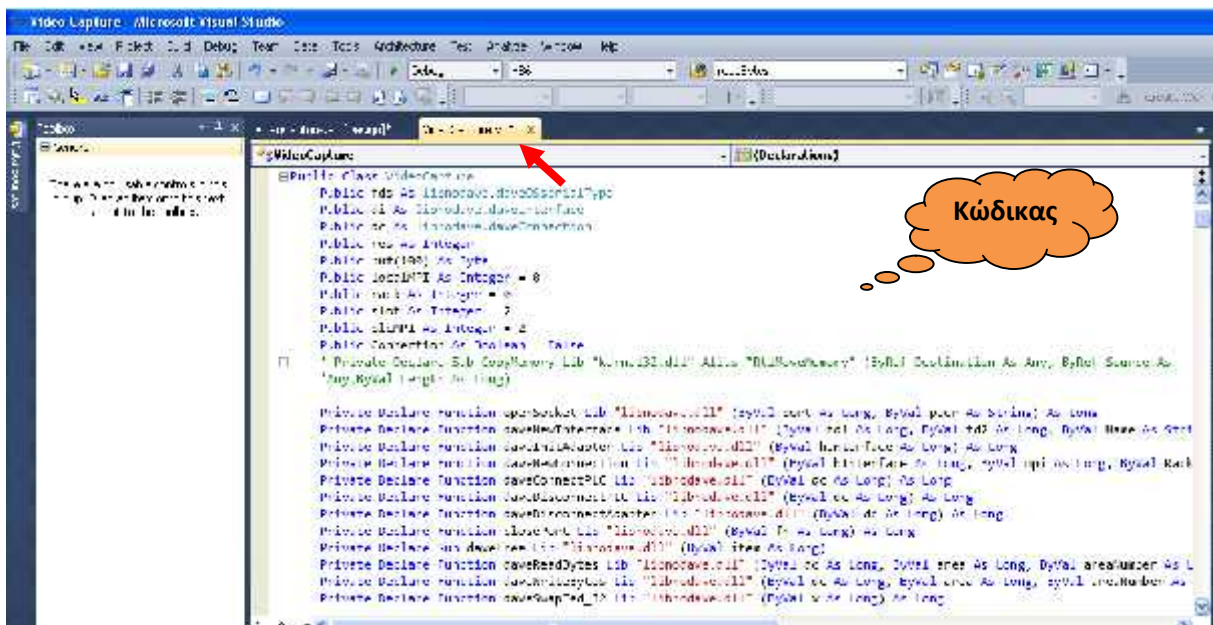


## 6.2 Desktop Application

project μ μ .  
 Forms μ , . . . buttons μ , ο  
 application,  
 Video Capture.vb( 85).  
 Step 7.  
 μ μ .



Εικόνα 84: VideoCapture.vb[Design]



Εικόνα 85: VideoCapture.vb[Code]



ορισμός του κατάλληλου namespace για την επικοινωνία με dll του λειτουργικού συστήματος'  
 'το οποίο παρέχει την δυνατότητα κλήσεων συστήματος για την επικοινωνία με τις διάφορες συσκευές'

```
Imports System.Runtime.InteropServices
```

'Τα μηνύματα που χρησιμοποιούνται είναι κωδικοί αριθμοί που για χάρη ευκολίας'  
 'λαμβάνουν ονόματα αντιστοίχισης υπό μορφή σταθερών.'

```
Public Class VideoCapture
    Public fds As libnodave.daveOSserialType
    Public di As libnodave.daveInterface
    Public dc As libnodave.daveConnection
    Public res As Integer
    Public buf(100) As Byte
    Public localMPI As Integer = 0
    Public rack As Integer = 0
    Public slot As Integer = 2
    Public plcMPI As Integer = 2
    Public Connection As Boolean = False
    ' Private Declare Sub CopyMemory Lib "kernel32.dll" Alias "RtlMoveMemory" (ByRef
    Destination As Any, ByRef Source As
    'Any, ByVal Length As Long)

    Private Declare Function openSocket Lib "libnodave.dll" (ByVal port As Long, ByVal
    peer As String) As Long
    Private Declare Function daveNewInterface Lib "libnodave.dll" (ByVal fd1 As Long,
    ByVal fd2 As Long, ByVal Name As String, ByVal localMPI As Long, ByVal protocol As
    Long, ByVal speed As Long) As Long
    Private Declare Function daveInitAdapter Lib "libnodave.dll" (ByVal hInterface As
    Long) As Long
    Private Declare Function daveNewConnection Lib "libnodave.dll" (ByVal hInterface
    As Long, ByVal mpi As Long, ByVal Rack As Long, ByVal Slot As Long) As Long
    Private Declare Function daveConnectPLC Lib "libnodave.dll" (ByVal dc As Long) As
    Long
    Private Declare Function daveDisconnectPLC Lib "libnodave.dll" (ByVal dc As Long)
    As Long
    Private Declare Function daveDisconnectAdapter Lib "libnodave.dll" (ByVal dc As
    Long) As Long
    Private Declare Function closePort Lib "libnodave.dll" (ByVal fh As Long) As Long
    Private Declare Sub daveFree Lib "libnodave.dll" (ByVal item As Long)
    Private Declare Function daveReadBytes Lib "libnodave.dll" (ByVal dc As Long,
    ByVal area As Long, ByVal areaNumber As Long, ByVal start As Long, ByVal numBytes As
    Long, ByVal buffer As Long) As Long
    Private Declare Function daveWriteBytes Lib "libnodave.dll" (ByVal dc As Long,
    ByVal area As Long, ByVal areaNumber As Long, ByVal start As Long, ByVal numBytes As
    Long, ByRef buffer As Byte) As Long
    Private Declare Function daveSwapIed_32 Lib "libnodave.dll" (ByVal x As Long) As
    Long
```

```

Private Const daveProtoMPI = 0      ' MPI for S7 300/400
Private Const daveProtoMPI2 = 1    ' MPI for S7 300/400, "Andrew's version"
Private Const daveProtoMPI3 = 2    ' MPI for S7 300/400, Step 7 Version, not yet
implemented
Private Const daveProtoPPI = 10    ' PPI for S7 200
Private Const daveProtoAS511 = 20   ' S5 via programming interface
Private Const daveProtoS7online = 50 ' S7 using Siemens libraries & drivers
for transport
Private Const daveProtoISOTCP = 122 ' ISO over TCP
Private Const daveProtoISOTCP243 = 123 ' ISO over TCP with CP243
Private Const daveProtoMPI_IBH = 223 ' MPI with IBH NetLink MPI to ethernet
gateway */
Private Const daveProtoPPI_IBH = 224 ' PPI with IBH NetLink PPI to ethernet
gateway */
Private Const daveProtoUserTransport = 255 ' Libnodave will pass the PDUs of S7
Communication to user defined call back functions.
'
'   ProfiBus speed constants:
'
Private Const daveSpeed9k = 0
Private Const daveSpeed19k = 1
Private Const daveSpeed187k = 2
Private Const daveSpeed500k = 3
Private Const daveSpeed1500k = 4
Private Const daveSpeed45k = 5
Private Const daveSpeed93k = 6
'
'   S7 specific constants:
'
Private Const daveBlockType_OB = "8"
Private Const daveBlockType_DB = "A"
Private Const daveBlockType_SDB = "B"
Private Const daveBlockType_FC = "C"
Private Const daveBlockType_SFC = "D"
Private Const daveBlockType_FB = "E"
Private Const daveBlockType_SFB = "F"
'
' Use these constants for parameter "area" in daveReadBytes and daveWriteBytes
'
Private Const daveSysInfo = &H3      ' System info of 200 family
Private Const daveSysFlags = &H5    ' System flags of 200 family
Private Const daveAnaIn = &H6       ' analog inputs of 200 family
Private Const daveAnaOut = &H7      ' analog outputs of 200 family
Private Const daveP = &H80          ' direct access to peripheral addresses
Private Const daveInputs = &H81
Private Const daveOutputs = &H82
Private Const daveFlags = &H83
Private Const daveDB = &H84 ' data blocks
Private Const daveDI = &H85 ' instance data blocks
Private Const daveV = &H87 ' don't know what it is
Private Const daveCounter = 28 ' S7 counters
Private Const daveTimer = 29 ' S7 timers
Private Const daveCounter200 = 30 ' IEC counters (200 family)
Private Const daveTimer200 = 31 ' IEC timers (200 family)
'
'   Library specific:
'
'   Result codes. Generally, 0 means ok,
'   >0 are results (also errors) reported by the PLC
'   <0 means error reported by library code.

```

```

'
Private Const daveResOK = 0 ' means all ok
Private Const daveResNoPeripheralAtAddress = 1 ' CPU tells there is no
peripheral at address
Private Const daveResMultipleBitsNotSupported = 6 ' CPU tells it does not support
to read a bit block with a
' length other than 1 bit.
Private Const daveResItemNotAvailable200 = 3 ' means a a piece of data is
not available in the CPU, e.g.
' when trying to read a non existing DB or bit bloc of length<>1
' This code seems to be specific to 200 family.
Private Const daveResItemNotAvailable = 10 ' means a a piece of data is
not available in the CPU, e.g.
' when trying to read a non existing DB
Private Const daveAddressOutOfRange = 5 ' means the data address is
beyond the CPUs address range
Private Const daveWriteDataSizeMismatch = 7 ' means the write data size
doesn't fit item size
Private Const daveResCannotEvaluatePDU = -123
Private Const daveResCPUNoData = -124
Private Const daveUnknownError = -125
Private Const daveEmptyResultError = -126
Private Const daveEmptyResultSetError = -127
Private Const daveResUnexpectedFunc = -128
Private Const daveResUnknownDataUnitSize = -129
Private Const daveResShortPacket = -1024
Private Const daveResTimeout = -1025

'Dim localETHERNET As Integer = 0, plcEthernet As Integer = 2

'Dim fds As libnodave.daveOSserialType
'Dim di As libnodave.daveInterface
'Dim dc As libnodave.daveConnection
'Dim res As Integer

'Dim buf(1000) As Byte
Private Const S7_IP As String = "192.167.1.106"
Private hSocket As Long
Private hInterface As Long
Private hConnection As Long

'Sindesi me to PLC
Private Sub connectplc()
Dim RetCode As Long

lblconnect.Text = "connected to " & S7_IP & " ... failed!"

fds.rfd = libnodave.openSocket(102, S7_IP)
fds.wfd = fds.rfd

If fds.rfd > 0 Then ' if step 1 is ok
di = New libnodave.daveInterface(fds, "IF1",
0, libnodave.daveProtoISOTCP,
libnodave.daveSpeed187k)

di.setTimeout(1000000)
res = di.initAdapter
If res = 0 Then ' init Adapter is ok

```

```

' rack amd slot don't matter in case of MPI
dc = New libnodave.daveConnection(di, 0,
                                   rack, slot)

res = dc.connectPLC()
If res = 0 Then
    Connection = True
    lblconnect.Text = "Connected " + S7_IP
End If
End If
End If

End Sub

'Aposyndesi me to PLC

Private Sub DisconnectPLC()
    Dim RetCode As Long
    lblconnect.Text = "Disconnection from" & S7_IP & " failed! "

    dc.disconnectPLC()
    lblconnect.Text = "Disc OK"
    'cmdConnect.Enabled = True
    'cmdDisconnect.Enabled = False
    'cmdRead.Enabled = False
    'cmdWrite.Enabled = False
    'lblconnect.Caption = ""
End Sub

' Read DATA '

Private Function ReadPLCbits(ByVal inputnum, ByVal inputbit, ByVal type) As
Boolean
    res = dc.readBytes(type, inputnum, inputbit, 1, buf)
    If res = 0 Then
        Dim a = dc.getU8
        txtREAD.Text = a
    End If

End Function

'Write Data Manual Left'

Private Sub ManualLeft()

    Dim Adr As Integer
    Dim Par As Boolean = True

    Dim InputNum As Integer = 0 'M 0.5
    Dim BitNum As Integer = 5
    If Par Then
        buf(0) = 255 'write 1
    Else
        buf(0) = 0 'write 0
    End If
    If Connection Then
        Adr = InputNum * 8 + BitNum

        'write DB1.DBX1.2
        res = dc.writeBits(libnodave.daveFlags, 0, Adr, 1, buf)
    End If
End Sub

```

```

Else
    MsgBox("No connection with PLC!")
End If

End Sub

'Write data Manual Right

Private Sub ManualRight()

    Dim Adr As Integer
    Dim Par As Boolean = True

    Dim InputNum As Integer = 0 'M 0.3
    Dim BitNum As Integer = 3
    If Par Then
        buf(0) = 255 'write 1
    Else
        buf(0) = 0 'write 0
    End If
    If Connection Then
        Adr = InputNum * 8 + BitNum

        'write DB1.DBX1.3
        res = dc.writeBits(libnodave.daveFlags, 0, Adr, 1, buf)

    Else
        MsgBox("No connection with PLC!")
    End If

End Sub

Private Sub ResCam()
    Dim Adr As Integer
    Dim Par As Boolean = True

    Dim InputNum As Integer = 1 'inPut I1.0(Button)
    Dim BitNum As Integer = 0
    If Par Then
        buf(0) = 255 'write 1
    Else
        buf(0) = 0 'write 0
    End If
    If Connection Then
        Adr = InputNum * 8 + BitNum

        'write Input I1.0
        res = dc.writeBits(libnodave.daveFlags, 0, Adr, 1, buf) '0=apo pia thesi ,
1=bit

    Else
        MsgBox("No connection with PLC!")
    End If

End Sub

```



'@@@@@@@@@@@@@ APO EDW XEKINAEI O KWDIKAS GIA "CAPTION" @@@@@@@@@@@@@@@@@@'

```
Const WM_CAP_START = &H4005
Const WS_CHILD = &H40000000
Const WS_VISIBLE = &H10000000
```

```
Const WM_CAP_DRIVER_CONNECT = WM_CAP_START + 10
Const WM_CAP_DRIVER_DISCONNECT = WM_CAP_START + 11
Const WM_CAP_EDIT_COPY = WM_CAP_START + 30
Const WM_CAP_SEQUENCE = WM_CAP_START + 62
Const WM_CAP_FILE_SAVEAS = WM_CAP_START + 23
```

```
Const WM_CAP_SET_SCALE = WM_CAP_START + 53
Const WM_CAP_SET_PREVIEWRATE = WM_CAP_START + 52
Const WM_CAP_SET_PREVIEW = WM_CAP_START + 50
```

```
Const SWP_NOMOVE = &H2S
Const SWP_NOSIZE = 1
Const SWP_NOZORDER = &H4S
Const HWND_BOTTOM = 1
```

'Στις δύο Συναρτήσεις παρακάτω έχω...'

'1### Η πρώτη Συνάρτηση χρησιμοποιείται για την ανάκτηση των διαθέσιμων συσκευών που έχουν αναγνωρίσει τα Windows'

```
Declare Function capGetDriverDescriptionA Lib "avicap32.dll" _
    (ByVal wDriverIndex As Short, _
    ByVal lpszName As String, ByVal cbName As Integer, ByVal lpszVer As
String, _
    ByVal cbVer As Integer) As Boolean
```

'2### Η δεύτερη Συνάρτηση χρησιμοποιείται για την δημιουργία του παράθυρου προεπισκόπησης της εικόνας που'

'λαμβάνει εκείνη την στιγμή η webcam '

```
Declare Function capCreateCaptureWindowA Lib "avicap32.dll" _
    (ByVal lpszWindowName As String, ByVal dwStyle As Integer, _
    ByVal x As Integer, ByVal y As Integer, ByVal nWidth As Integer, _
    ByVal nHeight As Short, ByVal hWnd As Integer, _
    ByVal nID As Integer) As Integer
```

'Στις τρεις Συναρτήσεις παρακάτω έχω...'

'1### Η πρώτη Συνάρτηση χρησιμοποιείται για την αποστολή μηνυμάτων σε ένα παράθυρο'

```
Declare Function SendMessage Lib "user32" Alias "SendMessageA" _
    (ByVal hWnd As Integer, ByVal Msg As Integer, ByVal wParam As Integer, _
    <MarshalAs(UnmanagedType.AsAny)> ByVal lParam As Object) As Integer
```

'2### Η δεύτερη Συνάρτηση χρησιμοποιείται για την τοποθέτηση του παραθύρου στον χώρο'

```
Declare Function SetWindowPos Lib "user32" Alias "SetWindowPos" _
    (ByVal hWnd As Integer, _
    ByVal hWndInsertAfter As Integer, ByVal x As Integer, ByVal y As Integer, _
    ByVal cx As Integer, ByVal cy As Integer, ByVal wFlags As Integer) As Integer
```

```
'3### Η τρίτη Συνάρτηση χρησιμοποιείται για την καταστροφή του παραθύρου'  
Declare Function DestroyWindow Lib "user32" (ByVal hwnd As Integer) As Boolean  
  
'Εδώ γίνεται ο ορισμός δύο Global μεταβλητών που θα διαχειρίζονται την εκάστοτε  
επιλεγμένη συσκευή καθώς και  
'το παράθυρο εμφάνισης της πληροφορίας που εμφανίζει εκείνη την στιγμή η web  
camera'  
Dim VideoSource As Integer  
Dim hwnd As Integer  
  
'Κατά την εκκίνηση της εφαρμογής ενεργοποιείται το κουμπί btnStartRecording και  
απενεργοποιείται  
'το κουμπί btnStopRecording.  
'Η εφαρμογή αναζητά και επιστρέφει τις συσκευές που έχουν αναγνωρισθεί στο  
σύστημα'  
Private Sub VideoCapture_Load(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load  
  
    btnStartRecording.Enabled = True  
    btnStopRecording.Enabled = False  
  
    '---list all the video sources---  
    ListVideoSources()  
End Sub  
  
'Η μέθοδος StopPreviewWindow() αναλαμβάνει να αποσυνδέσει την επικοινωνία με  
οποιαδήποτε συσκευή είναι ενεργοποιημένη  
'εκείνη την στιγμή και να καταστρέψει το παράθυρο «χώρο» εμφάνισης της εικόνας που  
λαμβάνεται από την webcam.'  
  
'--disconnect from video source---  
Private Sub StopPreviewWindow()  
    'Σε πρώτη φάση στέλνεται μήνυμα στην webcam ότι η εφαρμογή αποσυνδέεται από  
αυτήν'  
    SendMessage(hwnd, WM_CAP_DRIVER_DISCONNECT, VideoSource, 0)  
    'ακολουθως καλείται η συνάρτηση DestroyWindow'  
    DestroyWindow(hwnd)  
End Sub  
  
'Η μέθοδος ListVideoSources() είναι μία μέθοδος που δημιουργούμε για την ανάκτηση  
των συσκευών.  
'Αυτή η μέθοδος ελέγχει χρησιμοποιώντας την συνάρτηση capGetDriverDescriptionA για  
την ύπαρξη συσκευών,  
'τις οποίες καταχωρεί στο ListBox lstVideoResources. '  
  
'---list all the various video sources---  
Private Sub ListVideoSources()  
    Dim DriverName As String = Space(80)  
    Dim DriverVersion As String = Space(80)  
    For i As Integer = 0 To 9  
        If capGetDriverDescriptionA(i, DriverName, 80, DriverVersion, 80) Then  
            VideoSources.Items.Add(DriverName.Trim)  
        End If  
    Next  
End Sub  
  
'Η μέθοδος PreviewWindow(PictureBox) αναλαμβάνει να συνδεθεί με τον driver της  
webcam και να εμφανίσει την πληροφορία  
'που λαμβάνει η συσκευή εκείνη την στιγμή
```

```

'---preview the selected video source---
Private Sub PreviewVideo(ByVal pbCtrl As PictureBox)
    'Αρχικά, στέλνεται ένα μήνυμα μέσω της συνάρτησης capCreateCaptureWindow,
    'η οποία δημιουργεί ένα παράθυρο στο οποίο θα παρουσιαστεί πληροφορία που
    λαμβάνει η webcam κάθε στιγμή
    hWnd = capCreateCaptureWindowA(VideoSource, WS_VISIBLE Or WS_CHILD, 0, 0, 0, _
    0, pbCtrl.Handle.ToInt32, 0)
    'Αμέσως μετά γίνεται έλεγχος αν ο driver μπορεί να συνδεθεί με την webcam και
    εφόσον κάτι τέτοιο είναι δυνατό τότε
    If SendMessage(hWnd, WM_CAP_DRIVER_CONNECT, VideoSource, 0) Then
        'στέλνεται ένα μήνυμα που καθορίζει τις διαστάσεις της εικόνας που θα
        προβάλλει
        '---set the preview scale---
        SendMessage(hWnd, WM_CAP_SET_SCALE, True, 0)
        'ένα δεύτερο μήνυμα που καθορίζει τον ρυθμό ανανέωσης των εικόνων
        '---set the preview rate (ms)---
        SendMessage(hWnd, WM_CAP_SET_PREVIEWRATE, 30, 0)
        'ένα τρίτο μήνυμα που καθορίζει την έναρξη της λήψης εικόνας στο παράθυρο
        που δημιουργήθηκε
        '---start previewing the image---
        SendMessage(hWnd, WM_CAP_SET_PREVIEW, True, 0)
        'και τέλος την τοποθεσία που θα εμφανιστεί η εικόνα
        '---resize window to fit in PictureBox control---
        SetWindowPos(hWnd, HWND_BOTTOM, 0, 0, _
        pbCtrl.Width, pbCtrl.Height, _
        SWP_NOMOVE Or SWP_NOZORDER)
        'Σε περίπτωση που δεν γίνεται επικοινωνία με την webcam καταστρέφεται το
        παράθυρο
    Else
        '---error connecting to video source---
        DestroyWindow(hWnd)
    End If
End Sub

'Δημιουργία του κώδικα διαχείρισης της επιλογής συσκευής του χρήστη. Ο χρήστης
επιλέγοντας κάποια συσκευή'
'από το ListBox συνδέεται με αυτήν και ξεκινάει την λήψη της πληροφορίας.'
Private Sub ListBox1_SelectedIndexChanged( _
    ByVal sender As System.Object, ByVal e As System.EventArgs) _
    Handles VideoSources.SelectedIndexChanged

    '---stop video in case it is on---
    'αρχικά σταματάμε την εμφάνιση πληροφορίας από οποιαδήποτε συσκευή είναι
    ενεργοποιημένη καλώντας την μέθοδο
    'StopPreviewWindow() '
    StopPreviewWindow()
    '---check which video source is selected---
    'αμέσως μετά λαμβάνουμε την επιλογή του χρήστη από το ListBox'
    VideoSource = VideoSources.SelectedIndex
    '---preview the selected video source
    'τελός ξεκινάμε την εμφάνιση πληροφορίας από την νέα συσκευή καλώντας την
    μέθοδο PreviewVideo(PictureBox1)'
    PreviewVideo(PictureBox1)
End Sub

'Telos με την πίεση του πλήκτρου Stop Camera απενεργοποιείται η προεπισκόπηση της
εικόνας από την webcam
'καλώντας την μέθοδο StopPreviewWindow().'
Private Sub StopCamera_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnStopCamera.Click
    StopPreviewWindow()

```

```

End Sub
'Sτην περίπτωση λήψης video από την webcam (πιέζοντας δηλαδή ο χρήστης το κουμπί
Start Recording) απενεργοποιείται
'το κουμπί Start Recording, ενεργοποιείται το κουμπί Stop Recording
Private Sub btnStartRecording_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnStartRecording.Click
    btnStartRecording.Enabled = False
    btnStopRecording.Enabled = True
    'ενώ στέλνεται μήνυμα στην webcam να ξεκινήσει την εγγραφή video από το
    παράθυρο που γίνεται η προεπισκόπηση
    '---start recording---
    SendMessage(hWnd, WM_CAP_SEQUENCE, 0, 0)
End Sub
'Με την πίεση του πλήκτρου Stop Recording, αυτό απενεργοποιείται, ενεργοποιείται
το κουμπί Start Recording
Private Sub btnStopRecording_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnStopRecording.Click
    btnStartRecording.Enabled = True
    btnStopRecording.Enabled = False
    'και αποθηκεύεται το video στον σκληρό δίσκο με όνομα RecordedVideo.avi
    '---save the recording to file---
    SendMessage(hWnd, WM_CAP_FILE_SAVEAS, 0, "C:\RecordedVideo.avi")
End Sub

Private Sub btnResetCamera_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnResetCamera.Click
    ResCam()
End Sub

Private Sub GroupBox1_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GroupBox1.Enter

End Sub

Private Sub lblconnect_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles lblconnect.Click

End Sub

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnConnect.Click
    connectplc()
End Sub

Private Sub GroupBox3_Enter(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles GroupBox3.Enter

End Sub

Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnDisc.Click
    DisconnectPLC()
End Sub

Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As

```

```

System.EventArgs) Handles btnReadData.Click
    ReadPLC()
End Sub

Private Sub txtREAD_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles txtREAD.Click

End Sub

Private Sub btnLeft_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLeft.Click
    ManualLeft()
End Sub

Private Sub btnRight_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnRight.Click
    ManualRight()
End Sub

Private Sub Button4_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnManual.Click
    Dim Adr As Integer

    Dim a

    Dim InputNum As Integer = 10 'M10.0
    Dim BitNum As Integer = 0
    Adr = InputNum * 8 + BitNum

    If Connection Then
        res = dc.readBits(libnodave.daveFlags, 0, Adr, 1, buf)
        If res = 0 Then
            a = dc.getU8
        End If
    End If
    If a > 0 Then
        buf(0) = 0 'write 1
    Else
        buf(0) = 255 'write 0
    End If
    If Connection Then
        Adr = InputNum * 8 + BitNum

        'write DB1.DBX1.3
        res = dc.writeBits(libnodave.daveFlags, 0, Adr, 1, buf)

    Else
        MsgBox("No connection with PLC!")
    End If
End Sub

Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Timer1.Tick
    Dim a = ReadPLCbits(libnodave.daveFlags, 10, 1)
End Sub
End Class

```





- 
- 1) **Σταμάτης, Μάνεσης Α.** *Συστήματα βιομηχανικών Αυτοματισμών*. Πάτρα : Πανεπιστήμιο Πατρών, 2004.
  - 2) Siemens Simatic Working With Step 7 Getting Started [ Edition 03/2006 ].
  - 3) Siemens Simatic Programming With Step 7 V5.3 Manual [ Edition 01/2004 ].
  - 4) [www.siemens.com/automation](http://www.siemens.com/automation)
  - 5) **Dennis Collins, Eamon Lane.** *Προγραμματιζόμενοι Ελεγκτές*. Θεσσαλονίκη : Α . Τζιόλα, 2003.
  - 6) **Petruzella, Frank.** *Προγραμματιζόμενοι Λογικοί Ελεγκτές*. Θεσσαλονίκη : Α. Τζιόλα, 2000.
  - 7) Τεχνικά Εγχειρίδια : SIEMENS Simatic S7.