

ToolsNet Open Protocol ver 3.4

Programmer's Manual

Revision 5
2010-01-15



Table of contents

1. GENERAL	4
2. START UP	4
3. COMMUNICATION	4
4. LOGICAL STRUCTURE.....	5
5. USED TELEGRAMS	6
5.1 GENERAL TELEGRAM LAYOUT	6
5.2 RESULT TELEGRAM LAYOUT	7
5.3 RESULT TELEGRAMS	8
5.4 ERROR EVENT TELEGRAM	10
5.5 ERROR EVENT ACKNOWLEDGE TELEGRAM	10
5.6 ACKNOWLEDGE TELEGRAM.....	11
5.7 PIM VERIFICATION TELEGRAM	11
5.8 PIM INFO REQUEST TELEGRAM.....	11
5.9 PIM INFORMATION TELEGRAM.....	11
5.10 KEEP-ALIVE TELEGRAM	12
5.11 SYSTEM DESCRIPTION TELEGRAM.....	12
5.12 STATION DESCRIPTION TELEGRAM.....	13
6. DATABASE TABLES.....	16
APPENDIX A: PIC FSH PROTOCOL	17
APPENDIX B: COMMUNICATION DIAGRAM	18

Revision History

Ver	Rev.	Date	Sign	Description
	1		SK	Original document
	1.1	2002-01-17	SK	Added a Keep-Alive Telegram. Needed for getting rid of ghost sockets
	1.5	2002-05-27	SK	Result telegram changed – System Number added!!! Manual reorganized.
	3.0	2002-09-23	SK	Error event specification added
	3.1	2003-02-03	SK	Communication at connection changed. At connection to default port information will be sent from server defining port number to be used and at which identification to start communication.
	3.2	2003-03-26	JJ	Changing Event part added needed items. Adding System type to Result and Trace part
	3.3	2003-12-07	SK	Description of PIM. Text updated, System Type introduced, database changes, format of description telegram changed, telegram System type description added
	3.4	2004-01-07	SK	Added Station number to PIM Info Request telegram
	3.5	2004-02-02	SK	Added Event parameter value type and Event parameter ID
	3.6	2004-02-05	SK	System type, System number, Station number, Spindle number and Program number changed to 4 digits ('9999')
	3.7	20040206	SK	Minor corrections in text
	3.8	2004-03-02	SK	Icon definition deleted
3.0	8	2004-04-27	AR	Added protocol version and changed revision number from 3.8 to 8.
3.1	1	2004-10-21	SK	Added additional VIN numbers definition
3.2	1	2005-12-22	SK	Added Parameter info
3.2	2	2006-02-08	SK	Documentation corrected, description of PIM Verification telegram added
3.3	3	2007-06-13	SK	Added MID numbers to Communication diagram
3.3	4	2007-06-14	SK	Added a list of defined System Type
3.4	5	2009-11-10	JJ	Added Graph and Step for Result
3.4	6	2010-01-15	JJ	Fix in graphpart

1. General

Controller manufacturers may use ToolsNet Open Protocol to store rundown results in the ToolsNet database. The protocol is based on standard TCP/IP communication and Ethernet network protocol. All information in the telegrams is in ACSCII format to simplify testing and verification. The ToolsNet Server is acting as a passive host and is listening for connection requests from the Controllers. The Controller shall try to establish a connection to ToolsNet if not already connected.

The PIM (**P**rotocol **I**nterface **M**odule) server listens for connection requests from a controller. When a controller connects the PIM will immediately send a telegram verifying that it is a PIM application. This information can be used to decide if the user application should use the new or the old Open Protocol format. The PIM will then send information about from which sequence numbers the reporting of tightening results and error events shall start and which IP address and port number that should be used for connection to the ToolsNet server. After getting this information the controller will disconnect from PIM and connect to the ToolsNet server. The transfer of tightening results will then start.

The Controller has to store tightening data until the ToolsNet Server acknowledges successful storing in database.

2. Start Up

When a Controller (Station) powers up and is enabled to communicate on the network it shall attempt to open a TCP/IP connection with PIM. If the connection attempt fails the Controller shall wait a defined time (Connection Timeout, default 60 s) before making another attempt to connect. The Controller shall continue to attempt to connect to PIM until a successful connection is established. If the connection is successful but no PIM verification telegram is received before time out then the user application should try to use the old (version 1.5) specification of Open Protocol.

3. Communication

The Controller defines a unique number (a sequence number) in the identifier parameter in each telegram sent. This identifier is returned by the Acknowledge telegram and used to verify successful transmission. If a telegram is not acknowledged in a defined time (Transmission Timeout, default 5 s) the Controller shall continue to retransmit the telegram two times. If the telegram still is not acknowledged the Controller shall close connection and try to reestablish the connection to PIM.

If the Controller has lost connection with ToolsNet Server it should try to reestablish the connection to PIM. If the connection attempt fails the Controller shall wait a defined time (Connection Timeout, default 60 s) before making another attempt to connect. The Controller shall continue to attempt to connect to PIM until a successful connection is established.

When the network is down the Controller shall store the run down results internally until the network is up and running again and results can successfully be transmitted to the ToolsNet Server

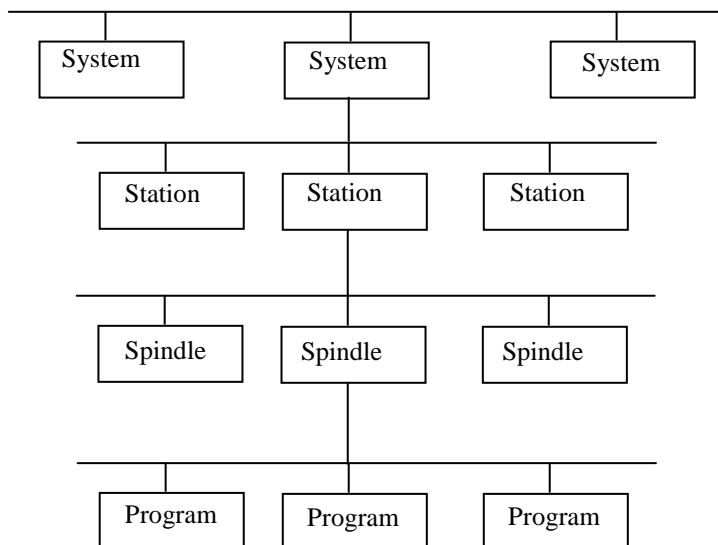
To be able to get rid of ghost sockets (a socket without a partner due to unexpected connection breakdown) the controller must send a Keep-Alive telegram if no information has been transmitted during a defined time (Keep-Alive timeout, default 30 s). If there are no telegrams received from a Controller during a defined time (default 60s) Toolsnet will close the connection to the Controller and the Controller will have to reestablish the connection.

Time synchronization between the ToolsNet server and the attached clients is essential for the efficiency of the database information. The Acknowledge telegram will contain a ToolsNet Server time stamp giving the clients an opportunity to synchronize their clocks with the ToolsNet server clock.

4. Logical structure

All Controllers are grouped in a number of systems. Each System has a number of Stations. Each Station has a number of Spindles. Each Spindle has a number of Programs.

Each System must have a unique System number. Within a System each Station must have a unique Station number. Within a Station each Spindle must have a unique Spindle number and within a Spindle each Program must have a unique Program number.



The Result telegram contains System number, Station number, Spindle number and Program number which will make it possible to build the ToolsNet Map for the non-Atlas Copco Controllers using default names based on the number like System 1, Station 4, Spindle 2, Program 5 and so on.

Data collection and database software, PF3000

The names can be manually modified using the ToolsNet Map Editor or by sending description telegrams like for example Station description telegram from the Controller. For more information see Chapter **16. ToolsNet structure** in the ToolsNet manual.

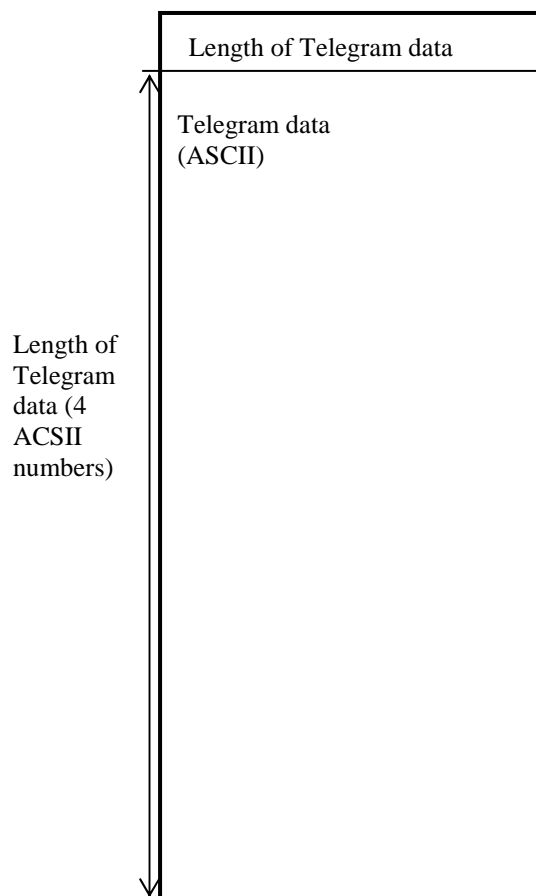
ToolsNet will use the stored information to display a map in the user interface over network structure and all attached units

5. Used telegrams

5.1 General Telegram layout

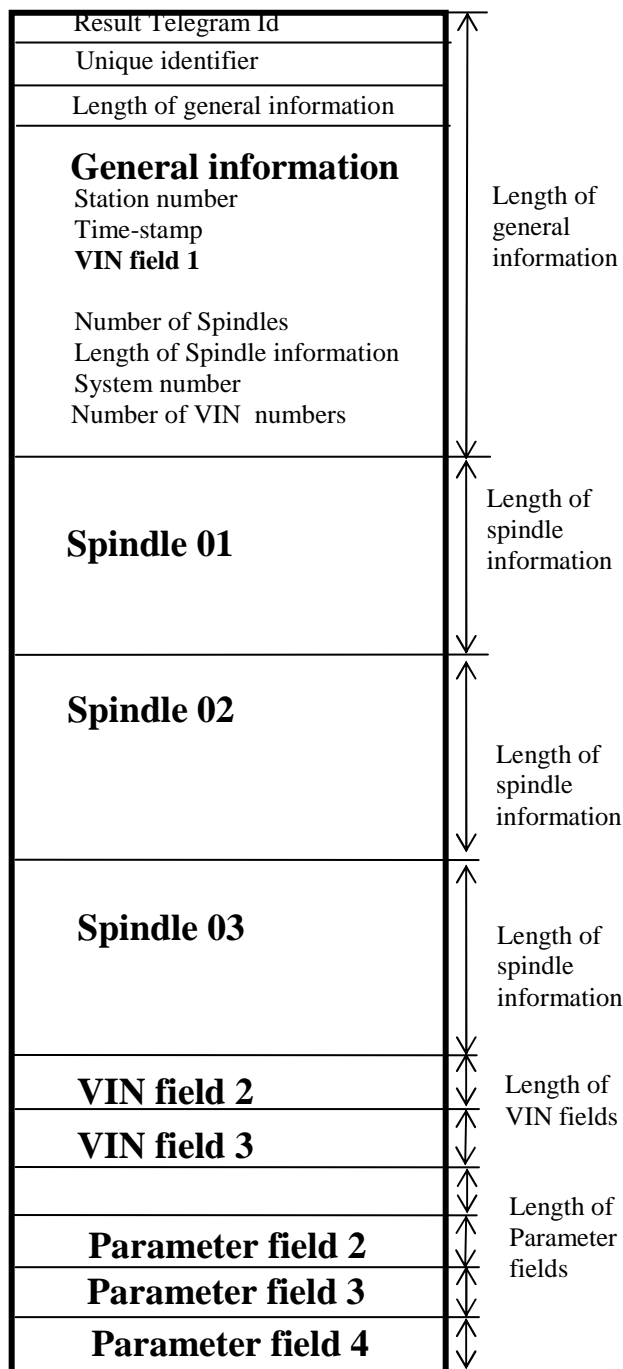
The general layout of the telegrams is four ASCII characters defining the length of the following telegram followed by the data in ASCII format. In the following descriptions of the different telegrams only the data part of the telegram will be defined. To each telegram definition the total length of data should be added.

Length	Contents	Type
4	Length of data	ASCII numbers (0001-9999)
N	Telegram	ASCII data



5.2 Result Telegram layout

The telegram text is divided in three parts – General information and Spindle information (information repeated for each spindle and Additional VIN numbers (information repeated for each VIN number). General information, Spindle information and VIN fields have individual lengths which means that information fields can be expanded in future releases without loosing backward compatibility



5.3 Result Telegrams

For non-Atlas Copco controllers

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'04'	Unique telegram identification number
3-7	5	Identifier	'00000 – 99999'	A unique number to be returned in the Acknowledge telegram. Used to identify the answer ¹⁾
8-9	2	Length of general information	'79 – 99'	Length of general information part from byte 1 (>=80)
10-13	4	System type	'0001 – 9999'	Must be defined
14-17	4	System number	'0001 – 9999'	Must be defined
18-21	4	Station number	'0001 – 9999'	Must be defined
22-35	14	Time-stamp	yyyymmddhh nnss	Year month day hour minutes seconds
36-40	5	Rundown sequence number	'00000-99999'	
41-65	25	VIN	25 ASCII characters	Vehicle Identification Number
66-69	4	Number of spindles	'0001 – 9999'	Maximum number of spindles = 9999
70-71	2	Length of spindle information	'01 – 99'	Length of information for one spindle This information fields are repeated for each spindle. Must be > 69
72-73	2	Number of additional VIN numbers	'00-99'	
74-75	2	Length of VIN field	'29-99'	Must be >28
76-78	3	Number of additional Parameters	'000-999'	
79-80	2	Length of Parameter field	'72-99'	Must be >= 72

Spindle Info

+1-4	4	Spindle number	'0001 – 9999'	
+5-14	10	Spindle serial number	10 ASCII characters	
+15-18	4	Program number	'0001 – 9999'	
+19-19	1	Overall Status	'0-1'	0=OK, 1=NOK
+20-27	8	Torque low limit	'xxxxx.xx'	Nm
+28-35	8	Final torque	'xxxxx.xx'	Nm
+36-36	1	Torque status	'0-3'	0=OK, 1=NOK, 2=Low, 3=High
+37-44	8	Torque high limit	'xxxxx.xx'	Nm
+45-52	8	Angle low limit	'xxxxxx.x'	Degree
+53-60	8	Final angle	'xxxxxx.x'	Degree
+61-61	1	Angle status	'0-3'	0=OK, 1=NOK, 2=Low, 3=High
+62-69	8	Angle high limit	'xxxxxx.x'	
+70-70	1	Time status	'0-3'	0=OK, 1=NOK, 2=Low, 3=High

**Additional
VIN info**

+1-4	4	VIN identifier	'0001-9999'	Identifier 0000 reserved for VIN number in byte 41-65
+5-29	25	VIN number	25 ASCII characters	Additional Vehicle Identification numbers

**Parameter
info**

+1-4	4	Spindle number	'0000 – 9999'	Spindle Number 0 is general Parameter number
+5-8	4	Program number	'0001 – 9999'	
+9-13	5	Parameter ID	'00000-65535'	
+14-38	25	Parameter name	25 ASCII Characters	EX: Step:1 Torque
+39-63	25	Value	25 ASCII characters	A string of 25 characters, a 10 digit signed integer 32bit or a 24-25 digit real with or without decimal point
+64-64	1	Type	'0-2'	Value should be displayed 0 => as a string 1 => as an integer 2 => as a real value
+65-70	6	Unit	6 ASCII characters	Value unit ex: ohm, VA, km/h
+71-72	2	Step no	00-99	Step no 0 are general parameters for the result. All other data is step specific.

Additional VIN numbers are placed after spindle information

Parameter info is placed after Additional VIN numbers

Start of Additional VIN info: byte number = Length of general information+Number of spindles*length of spindle information+1

Start of Parameter info: byte number = Start of Additional VIN info + Number of additional VIN*

Length of VIN field

¹⁾ Unique number (for example a sequence number) for a specific controller

String: Strings should be padded with blanks. Leading blanks will be extracted by the program

5.4 Error Event telegram

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'09'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	A unique number to be returned in the Acknowledge telegram. Used to identify the answer
8-11	4	System type	'0001-9999'	Must be defined
12-15	4	System number	'0001-9999'	Must be defined
16-19	4	Station number	'0001-9999'	Must be defined
20-33	14	Time-stamp	yyyymmddhhnnss	Year month day hour minutes seconds. Report time.
34-38	5	Event sequence number	'00000-99999'	
39-43	5	Error Code	'00000-99999'	Error codes
44-46	3	Event level	'000-999'	
47-48	2	Number of event parameters	'00-99'	
49-51	3	Event parameter ID	'000-999'	Repeated for each parameter value
52	1	Event parameter Value Type	'0-9'	Repeated for each parameter value
53-72	20	Event parameter Value	String	Repeated for each parameter value

5.5 Error Event Acknowledge telegram

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'10'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	The unique number in the Error Event telegram. Used to identify the answer
8-11	4	System type	'0001 – 9999'	Must be defined
12-15	4	System number	'0001 – 9999'	Must be defined
16-19	4	Station number	'0001-9999'	Must be defined
20-33	14	Time-stamp	yyyymmddhhnnss	Year month day hour minutes seconds. Report time.
34-38	5	Error Code	'00000 – 99999'	Error codes

5.6 Acknowledge telegram

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'05'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	A unique number in the acknowledged telegram. Used to identify the answer
8-21	14	Time-stamp	yyyymmddhhnnss	Year month day hour minutes seconds ToolsNet server time stamp
22-24	3	Error code	'000 – 999'	000 => transmission successful

5.7 PIM Verification telegram

	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'12'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	A unique number to be returned in Acknowledge telegram. Used to identify the answer

5.8 PIM Info Request telegram

	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'07'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	A unique number to be returned in PIM Information telegram. Used to identify the answer
8-11	4	System type	'0001-9999'	
12-15	4	System number	'0001-9999'	
16-19	4	Station number	'0001-9999'	

5.9 PIM Information telegram

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'08'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	The unique number in PIM Info Request telegram. Used to identify the answer
8-12	5	Result sequence number	'00000 – 99999'	Sequence number for latest stored result in the TN database
13-17	5	Event sequence number	'00000 – 99999'	Sequence number for latest stored event in the TN database
18-32	15	TN IP address	'192.168.0.15'	ToolsNet IP address
33-37	5	TN Port number	'10111'	ToolsNet port number

5.10 Keep-Alive telegram

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'06'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	A unique number to be returned in Acknowledge telegram. Used to identify the answer

5.11 System description telegram

After a connection is established the Controller may send System Type, System, Station and/or Icon description telegrams to update/modify information about the structure of the connected Controllers. The Description telegrams may be sent at any time and the information sent will overwrite the corresponding information stored in the database.

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'01'	Unique telegram identification number
3-7	5	Identifier	'00000 – 99999'	A unique number to be returned in Acknowledge telegram. Used to identify the answer
8-11	4	System Type	'0001 – 9999'	Unique number for each type of controller
12-15	4	System number	'0001 – 9999'	Unique number within the System Type
16-30	15	IP-address	'10.40.22.112'	
31-55	25	System name	String	
56-57	2	Spare	' '	2 space characters

5.12 Station description telegram

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'02'	Unique telegram identification number
3-7	5	Identifier	'00000 – 99999'	A unique number to be returned in Acknowledge telegram. Used to identify the answer
8-11	4	System Type	'0001 – 9999'	Unique number for each type of controller
12-15	4	System number	'0001 – 9999'	Unique number within the network
16-19	4	Station number	'0001 – 9999'	Unique number within the system
20-34	15	IP-address	'10.40.22.113'	
35-59	25	Station name	String	
60-63	4	Number of spindles	'0001 - 9999'	
64-67	4	Spindle number	'0001 - 9999'	Unique number within the station
68-92	25	Spindle name	String	
93-96	4	Number of programs	'0001 - 9999'	
97-100	4	Program number	'0001 - 999'	Unique number within the spindle
101-125	25	Program name	String	

Byte 97-125 (29 byte) is repeated for each program.

Byte (64 – 96+NumberOfPrograms*29) is repeated for each spindle.

5.13 Graph Telegrams

Byte	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'13'	Unique telegram identification number
3-7	5	Identifier	'00000 – 99999'	A unique number to be returned in the Acknowledge telegram. Used to identify the answer ¹⁾
8-10	3	Length of general information	'000-999'	Length of general information part from byte 1 (>=116)
11-14	4	System type	'0001 – 9999'	Must be defined
15-18	4	System number	'0001 – 9999'	Must be defined
19-22	4	Station number	'0001 – 9999'	Must be defined
23-26	4	Spindle number	'0001 - 9999'	Unique number within the station
27-30	4	Program number	'0001 - 999'	Unique number within the spindle
31-44	14	Time-stamp	Yyyymmddh hnnss	Year month day hour minutes seconds
45-49	5	Rundown sequence number	'00000-99999'	
50	1	Graph Type	'0 – 1'	Type of Graph in Telegram, supported types: 0=Torque Trace 1=Angle Trace
51-60	10	BitShift	'xxxxxxxxxx',	BitShift for trace as signed integer
61-70	10	Scale FactorDom	'xxxxxxxxxx',	Scale Factor Dom for trace as signed integer
71-80	10	Scale FactorNom	'xxxxxxxxxx',	Scale Factor Nom for trace as signed integer
81-88	8	Min Limit	'xxxxxx.x'	Units depends on GraphType
89-96	8	Max Limit	'xxxxxx.x'	Units depends on GraphType
97-104	8	Angle Offset	'xxxxxx.x'	Degree
105-112	8	Sample time	'xxxxxx.x'	Time in ms between samples in Trace Data part
113-116	4	Length of Trace Data	'0000-8000'	Length in bytes
117-X	Dynamic	Trace Data	Special	Trace Data is a packed Array of signed 16bit integers in network order. Scale Factor * 16bit value = actual float value

Scale factor Calculation

Scale factor = 2^{bitshift} * Scale factor Dom / Scale factor Nom

Number:

All leading and trailing blanks will be extracted and the number will be right shifted before reading. If a decimal point exists it will be used. If a decimal point doesn't exist it will be considered being to the right of the number. This will generate a flexible input format for numbers. For example 12.5 will be interpreted as 12.5 and 12 will be interpreted as 12.0 no matter which position it has within the field.

6. Database tables

The tables below will contain information about the network and all attached Controllers including PowerFocus 3000 and PowerMacs

Tables:

TNMapSystemType
TNMapSystem
TNMapStation
TNMapSpindle
TNMapProgram

System type:

PF3000

0

PF2000

1

pMACS

2

Open Protocol

3 => Undefined

16 => Cooper Tools System

17 => Stanley Air Tools

18 => SPX Cooperation

19 => Ingersoll Rand

20 => Hickok

21 => AIMCO

All System type information is defined and inserted in the ToolsNet database by Atlas Copco Tools AB

Within each System type group the System number has to be unique. Within each System the Station number has to be unique. Within each Station the Spindle number has to be unique. Within each Spindle the Program number has to be unique.

PowerFocus 3000 and PowerMacs have separate result tables. This result table above will contain information about PowerFocus 2000 and non-Atlas Copco Controllers

The Icon bitmap information is stored in a database table on the ToolsNet Server in GIF format (16*16).

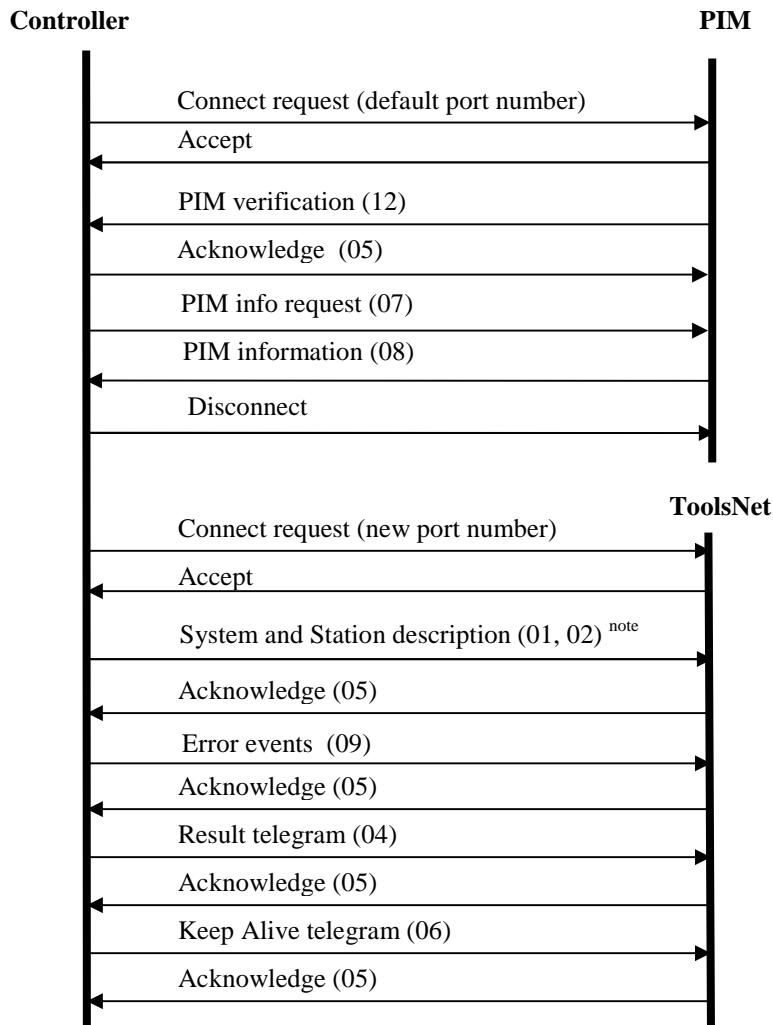
Appendix A: PIC FSH protocol

Protocol used by ToolsNet FSH Plugin

PIM FSH Info Request telegram

	Length	Parameter	Format	Comment
1-2	2	Telegram identification	'11'	Unique telegram identification number
3-7	5	Identifier	'00000 - 99999'	A unique number returned in Acknowledge telegram. Used to identify the answer
8-11	4	Station number	'0001-9999'	
12-26	15	Controller IP address	'192.168.0.20'	

Appendix B: Communication diagram



Note: If System and Station description telegram are not sent to ToolsNet default names will be used. The default names are based on the System/Station/Spindle/Program number for example System 2, Station 10, Spindle 5 or Program 4. These names can be edited manually using the Map Editor.