Withdrawn; however may be applied on special agreement !

EUROMAP 15	PROTOCOL FOR COMMUNICATION BETWEEN INJECTION MOULDING MACHINES AND A CENTRAL COMPUTER
Part 6	GATEWAY TO PERIPHERAL DEVICES (EUROMAP 17)

Version 1.0, Feb. 1992
Document Release Apr. 1992

This recommendation was prepared by the Working Group "Electronic Control of Injection Moulding Machines" of EUROMAP.

It is part of the complete EUROMAP 15 recommendation.

Contents

1. Introduction

2. Types of Telegrams
2.1 Definition of Telegrams from the Central Computer
2.2 Extension of the List with Standard Answers
2.3 Definition of Telegrams from the Machine Controller

3. Communication Flow
3.1 Read Parameters Initiated by Central Computer
3.2 Read Parameters Initiated by Machine Controller
3.3 Write Parameters Initiated by Central Computer
3.4 Write Parameters Initiated by Machine Controller
3.5 Read Blocks of Data Initiated by Central Computer
3.6 Read Blocks of Data Initiated by Machine Controller
3.7 Write Blocks of Data Initiated by Central Computer
3.8 Write Blocks of Data Initiated by Machine Controller
3.9 Procedure Success
3.10 Cancel Option
3.11 Queue Size

1. Introduction

Part 6 of the EUROMAP 15 recommendation, gateway to peripheral equipment via EUROMAP 17, defines the telegrams and data structures necessary to transfer any kind of data which can be transferred between an injection moulding machine and a peripheral device connected by EUROMAP 17 interface also between the central computer and the peripheral device directly.

Remark: EUROMAP 17 is the definition of a protocol for communication between an injection moulding machine and peripheral equipment such as temperature controllers, heating and cooling equipment, dryers, material feeding systems, dosing and mixing units, handling devices, mould changing systems and quality control systems.

The injection moulding machine acts as a gateway. Information from the central computer directed to the peripheral equipment is first sent to the machine controller by EUROMAP 15 telegrams, where the information is converted to EUROMAP 17 telegrams and transmitted to the peripheral equipment. Information from the peripheral equipment directed to the central computer is first sent to the machine controller by EUROMAP17, converted to EUROMAP 15 telegrams and then sent to the central computer.

For the understanding of EUROMAP 15 part 6 the basic knowledge of EUROMAP 17 is mandatory.

The following abreviations are used:

peripheral device	= AA	
starting channel	= SS	
ending channel	= EE	
parameter, Mnemonic	= CC	
transfer direction	= T	(write, download = "1", read, upload = "2")
sequence spec.	= S	(Because in EUROMAP 17 up to 16 channels are possible, up to 128 bytes of data can be transferred with a write or read command. The block length in Euromap 17 is also 128 bytes of data. The first part with 32 bytes has S = "0"; the second part with 32 bytes has S = "1" etc.)
last packet	= L	(L = "1" for last packet, L = "0" otherwise)
data	= D0 .... D31	
channel	= NN	
program name	= P1...P8	
block number of E 17	= BBB	(E 17 = EUROMAP 17)

Optional bytes in EUROMAP 17 are filled with blanks, if not used. Data bytes from EUROMAP 17 are left justified within D0 ... D31. Unused bytes are filled with blanks.

Every communication flow can be suspended by other EUROMAP 15 telegrams without influence on the communication flow itself.

2. Types of Telegrams

2.1 Definition of Telegrams from the Central Computer

Telegram 400: **PARAMETER TRANSFER SPECIFICATION**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07..08	-N02-	starting channel (SS)
09..10	-N02-	ending channel (EE)
11..12	-A02-	parameter (CC)
13	-N01-	transfer direction (T)
	"1"	- Download
	"2"	- Upload

Telegram 401: **REQUEST PARAMETER PACKET**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07	-N01-	sequence spec. (S)

Telegram 402: **REQUEST FOR STATUS**

contains no data bytes

Remark: request for status of parameter or block transfer

Telegram 403: **TRANSMIT PARAMETERS PACKET**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07	-N01-	sequence spec. (S)
08	-N01-	last packet (L)
09..16	-A08-	channel 1+(S*4) (D1...D8)
17..24	-A08-	channel 2+(S*4) (D1...D8)
25..32	-A08-	channel 3+(S*4) (D1...D8)
33..40	-A08-	channel 4+(S*4) (D1...D8)
		unused channels filled with ASCII-blank

Telegram 404: **SPECIFICATION OF BLOCK TRANSFER**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07..08	-N02-	starting channel (SS)
09..10	-N02-	ending channel (EE)
11..12	-A02-	mnemonic (CC)
13..20	-A08-	program name (P1...P8)
21	-N01-	transfer direction (T)
	"1"	- Download
	"2"	- Upload

Telegram 405: REQUEST BLOCK PACKET

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07..09	-N03-	block number (BBB)
10	-N01-	sequence spec. (S)

Telegram 406: TRANSMIT BLOCK PACKET

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07..09	-N03-	block number (BBB)
10	-N01-	sequence spec. (S)
11	-N01-	last packet (L)
	"0"	- no
	"1"	- yes
12..40	-B32-	data (S = "0") (D0...D31)
		or (S = "1") (D32...D63)
		or (S = "2") (D64...D95)
		or (S = "3") (D96...D127)

Telegram 407: CLEAR QUEUES

contains no data bytes

Telegram 408: REQUEST SIZE OF QUEUES

contains no data bytes

Telegram 409 REQUEST FOR TRANSFER SPECIFICATION

contains no data bytes

2.2 Extension of the List with Standard Answers

Telegram 100: STANDARD ANSWER

Byte	cont.	Description
05..08	-N03-	request number
	"402"	- request for status of transfer
	"405"	- request block
	"409"	- request parameter

2.3 Definition of Telegrams from the Machine Controller

Telegram 450: **ACKNOWLEDGE OF TRANSFER SPECIFICATION**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07	-N01-	acknowledge (X)
	"0"	- negative
	"1"	- positive
		if positive, a request must be prepared
		for the peripheral device (see § 2.1
		EUROMAP 17)

Telegram 451: **RECEIVE PARAMETERS PACKET**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07	-N01-	sequence spec (S)
08	-N01-	last packet (L)
	"0"	- no
	"1"	- yes
09..16	-A08-	channel 1+(S*4) (D1...D8)
17..24	-A08-	channel 2+(S*4) (D1...D8)
25..32	-A08-	channel 3+(S*4) (D1...D8)
33..40	-A08-	channel 4+(S*4) (D1...D8)
		unused channels filled with ASCII-blank

Telegram 452: **STATUS OF DEVICE**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07	-N01-	status of device AA (X)
	"1"	- parameter available / accepted by
		peripheral device
	"2"	- no communication
	"3"	- device busy

Telegram 453: **RECEIVE ACKNOWLEDGEMENT**

Byte	cont.	Description
05..06	-N02-	peripheral device (AA)
07	-N01-	acknowledgement
	"0"	- negative
	"1"	- positive

Telegram 455: **RECEIVE BLOCK PACKET**

Byte	cont.	Description	
05..06	-N02-	peripheral device	(AA)
07..09	-N03-	block number	(BBB)
10	-N01-	sequence spec.	(S)
11	-N01-	last packet	(L)
	"0"	- no	
	"1"	- yes	
12..43	-B32-	data (S = "0")	(D0...D31)
		or (S = "1")	(D32...D63)
		or (S = "2")	(D64...D95)
		or (S = "3")	(D96...D127)

Telegram 456: **RECEIVE ACKNOWLEDGEMENT OF BLOCK PACKET**

Byte	cont.	Description	
05..06	-N02-	peripheral device	(AA)
07..09	-N03-	block number	(BBB)
10	-N01-	sequence spec.	(S)
11	-N01-	acknowledgement	(X)
	"0"	- negative	
	"1"	- positive	

Telegram 457: **CLEAR QUEUES ACKNOWLEDGEMENT**

contains no data bytes

Telegram 458: **RECEIVE SIZE OF QUEUES**

Byte	cont.	Description	
05..09	-N05-	- EUROMAP 15 queue size in bytes	(MMMMM)
10..14	-N05-	- EUROMAP 17 queue size in bytes	(NNNNN)

Telegram 459: **PARAMETER TRANSFER SPECIFICATION**

Byte	cont.	Description	
05..06	-N02-	peripheral device	(AA)
07..08	-N02-	starting channel	(SS)
09..10	-N02-	ending channel	(EE)
11..12	-A02-	parameter	(CC)
13	-N01-	transfer direction	(T)
	"1"	- Download	
	"2"	- Upload	

Telegram 460: **SPECIFICATION OF BLOCK TRANSFER**

Byte	cont.	Description	
05..06	-N02-	peripheral device	(AA)
07..08	-N02-	starting channel	(SS)
09..10	-N02-	ending channel	(EE)
11..12	-A02-	mnemonic	(CC)
13..20	-A08-	program name	(P1...P8)
21	-N01-	transfer direction	(T)
	"1"	- Download	
	"2"	- Upload	

3. Communication Flow

3.1 Read Parameters Initiated by Central Computer

```
Transfer specification telegram:
"400"+AASSEECC+"2" | - parameter transfer (read)
                   |
"450"+AA+X         | - acknowledge of transfer specification
                   |   X = "0"  NACK
                   |   X = "1"  ACK
                   | if an ACK is sent, a request must be
                   | prepared for the peripheral device
                   | (see § 2.1 EUROMAP 17).
                   |
Wait for data ready telegram:
"000"              | - standard question
                   |
"100"+"402"        | - standard answer; request for status
                   |   of parameter transfer
                   |
Check parameter availaible and communication:
"402"              | - request for status of parameter transfer
                   |
"452"+AA+X         | - status of device AA
                   |   X = "1"  parameter available / accepted
                   |            by peripheral device
                   |   X = "2"  no communication
                   |   X = "3"  device busy
                   |
Read parameters:
"401"+AA+"0"       | - request parameters, packet 0
                   |
"451"+AA+"0"+L     | - receive parameters, packet 0
+4 times D1...D8   |   L = "1" for last packet
                   |   L = "0" if more than 4 channels
                   |
If more than 32 bytes (4 channels) needed:
"401"+AA+"1"       | - request parameters, packet 1
                   |
"451"+AA+"1"+L     | - receive parameters, packet 1
+4 times D1...D8   |   L = "1" for last packet
                   |   L = "0" if more than 8 channels
                   |
If more than 64 bytes (8 channels) needed:
"401"+AA+"2"       | - request parameters, packet 2
                   |
"451"+AA+"2"+L     | - receive parameters, packet 2
+4 times D1...D8   |   L = "1" for last packet
                   |   L = "0" if more than 12 channels
                   |
If more than 96 bytes (12 channels) needed:
"401"+AA+"3"       | - request parameters, packet 3
                   |
"451"+AA+"3"+L     | - receive parameters, packet 3
+4 times D1...D8   |   L = "1" for last packet
```

3.2 Read Parameters Initiated by Machine Controller

```
Transfer specification telegram:
"000"                 | - standard question
                      |
"100"+"409"           | - receive request
                      |
"409"                 | - request for transfer specification
                      |
"459"+AASSEECC+"2"    | - transfer specification (read)
                      |
Read parameters:
"401"+AA+"0"          | - request parameters, packet 0
                      |
"451"+AA+"0"+L        | - receive parameters, packet 0
+4 times D1...D8      |   L = "1" for last packet
                      |   L = "0" if more than 4 channels
                      |
If more than 32 bytes (4 channels) needed:
"401"+AA+"1"          | - request parameters, packet 1
                      |
"451"+AA+"1"+L        | - receive parameters, packet 1
+4 times D1...D8      |   L = "1" for last packet
                      |   L = "0" if more than 8 channels
                      |
If more than 64 bytes (8 channels) needed:
"401"+AA+"2"          | - request parameters, packet 2
                      |
"451"+AA+"2"+L        | - receive parameters, packet 2
+4 times D1...D8      |   L = "1" for last packet
                      |   L = "0" if more than 12 channels
                      |
If more than 96 bytes (12 channels) needed:
"401"+AA+"3"          | - request parameters, packet 3
                      |
"451"+AA+"3"+L        | - receive parameters, packet 3
+4 times D1...D8      |   L = "1" for last packet
```

3.3 Write Parameters Initiated by Central Computer

```
Transfer specification telegram:
"400"+AASSEECC+"1"  | - parameter transfer specification (write)
                    |
"450"+AA+X          | - acknowledge of transfer specification
                    |   X = "0"  NACK
                    |   X = "1"  ACK
                    |
Write parameters:
"403"+AA+"0"+L      | - transmit parameters, packet 0
+4 times D1...D8    |   L = "1" for last packet
                    |   L = "0" if more than 4 channels
                    |
"453"+AA+X          | - receive acknowledgement for packet 0
                    |   X = "0" NACK
                    |   X = "1" ACK
                    |
If more than 32 bytes (4 channels) needed:
"403"+AA+"1"+L      | - transmit parameters, packet 1
+4 times D1...D8    |   L = "1" for last packet
                    |   L = "0" if more than 8 channels
                    |
"453"+AA+X          | - receive acknowledgement for packet 1
                    |   X = "0" NACK
                    |   X = "1" ACK
                    |
If more than 64 bytes (8 channels) needed:
"403"+AA+"2"+L      | - transmit parameters, packet 2
+4 times D1...D8    |   L = "1" for last packet
                    |   L = "0" if more than 12 channels
                    |
"453"+AA+X          | - receive acknowledgement for packet 2
                    |   X = "0" NACK
                    |   X = "1" ACK
                    |
If more than 96 bytes (12 channels) needed:
"403"+AA+"3"+L      | - transmit parameters, packet 3
+4 times D1...D8    |   L = "1" for last packet
                    |
"453"+AA+X          | - receive acknowledgement for packet 3
                    |   X = "0" NACK
                    |   X = "1" ACK
                    |
Waiting for acknowledgement from peripheral device:
"000"               | - standard question
                    |
"100"+"402"         | - standard answer; request for status of
                    |   parameter transfer
                    |
Receiving acknowledgement:
"402"                | - request for status of parameter transfer
                     |
"452"+AA+X           | - status of device AA
                     |   X = "1"  parameter available / accepted
                     |            by peripheral device
                     |   X = "2"  no communication
                     |   X = "3"  device busy
```

3.4 Write Parameters Initiated by Machine Controller

```
Transfer specification telegram:
"000"                | - standard question
                     |
"100"+"409"          | - receive request
                     |
"409"                | - request for transfer specification
                     |
"459"+AASSEECC+"1"   | - transfer specification (write)
                     |
Write parameters:
"403"+AA+"0"+L       | - transmit parameters, packet 0
+4 times D1...D8     |   L = "1" for last packet
                     |   L = "0" if more than 4 channels
                     |
"453"+AA+X           | - receive acknowledgement for packet 0
                     |   X = "0" NACK
                     |   X = "1" ACK
                     |
If more than 32 bytes (4 channels) needed:
"403"+AA+"1"+L       | - transmit parameters, packet 1
+4 times D1...D8     |   L = "1" for last packet
                     |   L = "0" if more than 8 channels
                     |
"453"+AA+X           | - receive acknowledgement for packet 1
                     |   X = "0" NACK
                     |   X = "1" ACK
                     |
If more than 64 bytes (8 channels) needed:
"403"+AA+"2"+L       | - transmit parameters, packet 2
+4 times D1...D8     |   L = "1" for last packet
                     |   L = "0" if more than 12 channels
                     |
"453"+AA+X           | - receive acknowledgement for packet 2
                     |   X = "0" NACK
                     |   X = "1" ACK
                     |
If more than 96 bytes (12 channels) needed:
"403"+AA+"3"+L       | - transmit parameters, packet 3
+4 times D1...D8     |   L = "1" for last packet
                     |
"453"+AA+X           | - receive acknowledgement for packet 3
                     |   X = "0" NACK
                     |   X = "1" ACK
                     |
Waiting for acknowledgement from peripheral device:
"000"                 | - standard question
                      |
Wait for:             |
"100"+"402"           | - standard answer; request for status of
                      |   parameter transfer
                      |
Receiving acknowledgement:
"402"                 | - request for status of parameter transfer
                      |
"452"+AA+X            | - status of device AA
                      |   X = "1"  parameter available / accepted
                      |            by peripheral device
                      |   X = "2"  no communication
                      |   X = "3"  device busy
```

3.5 Read Blocks of Data Initiated by Central Computer

```
Transfer specification telegram:
"404"+AASSEECC        | - specification of block transfer
     +P1...P8+"2"     |
                      |
"450"+AA+X            | - acknowledge of transfer specification
                      |   X = "0"  NACK
                      |   X = "1"  ACK
                      |   if an ACK is sent, a request must be
                      |   prepared for the peripheral device
                      |   (see § 2.1 EUROMAP 17).
                      |
```

Single block read (repeated until the last block):

```
                      |
Waiting for block ready telegram:
"000"                 | - standard question
                      |
"100"+"405"           | - standard answer with request block
                      |
Read a block:
"405"+AA+BBB+"0"      | - request packet 0
                      |
"455"+AA+BBB+"0"      | - receive packet 0
   +L+D0...D31        |   L = "0" because packet 1 follows
                      |
"405"+AA+BBB+"1"      | - request packet 1
                      |
"455"+AA+BBB+"1"      | - receive packet 1
   +L+D32...D63       |   L = "0" because packet 2 follows
                      |
"405"+AA+BBB+"2"      | - request packet 2
                      |
"455"+AA+BBB+"2"      | - receive packet 2
   +L+D64...D95       |   L = "0" because packet 3 follows
                      |
"405"+AA+BBB+"3"    | - request packet 3
                    |
"455"+AA+BBB+"3"    | - receive packet 3
   +L+D96...D127    |   L = "0" means more blocks to follow
                    |   L = "1" means last block
```

If during the telegram sequence the machine detects any error, it sends as standard answer "100"+"402".

3.6 Read Blocks of Data Initiated by Machine Controller

Transfer specification telegram:

```
"000"               | - transmit standard question
                    |
"100"+"409"         | - receive request
                    |
"409"               | - request for transfer specification
                    |
"460"+AASSEECC      | - specification of block transfer (read)
     +P1...P8+"2"   |
                    |
```

Single block read (repeated until the last block):

```
                    |
```

Waiting for block ready telegram:

```
"000"               | - standard question
                    |
"100"+"405"         | - standard answer with request block
                    |
```

Read a block:

```
"405"+AA+BBB+"0"    | - request packet 0
                    |
"455"+AA+BBB+"0"    | - receive packet 0
   +L+D0...D31      |   L = "0" because packet 1 follows
                    |
"405"+AA+BBB+"1"    | - request packet 1
                    |
"455"+AA+BBB+"1"    | - receive packet 1
   +L+D32...D63     |   L = "0" because packet 2 follows
                    |
"405"+AA+BBB+"2"    | - request packet 2
                    |
"455"+AA+BBB+"2"    | - receive packet 2
   +L+D64...D95     |   L = "0" because packet 3 follows
                    |
"405"+AA+BBB+"3"    | - request packet 3
                    |
"455"+AA+BBB+"3"    | - receive packet 3
   +L+D96...D127    |   L = "0" means more blocks to follow
                    |   L = "1" means last block
```

If during the telegram sequence the machine detects any error, it sends as standard answer "100"+"402".

3.7 Write Blocks of Data Initiated by Central Computer

```
Transfer specification:
"404"+AASSEECC      | - specification of block transfer (write)
     +P1...P8+"1"   |
                    |
"450"+AA+X          | - acknowledge of transfer specification
                    |   X = "0"  NACK
                    |   X = "1"  ACK
                    |
Single block write (repeated until the last block):
                    |
Write a block:
"406"+AA+BBB+"0"    | - transmit packet 0
   +L+D0...D31      |   L = "0" because packet 1 follows
                    |
"456"+AA+BBB+"0"+X  | - receive acknowledgement for packet 0
                    |   X = "0"  NACK
                    |   X = "1"  ACK
                    |
"406"+AA+BBB+"1"    | - transmit packet 1
   +L+D32...D63     |   L = "0" because packet 2 follows
                    |
"456"+AA+BBB+"1"+X  | - receive acknowledgement for packet 1
                    |   X = "0"  NACK
                    |   X = "1"  ACK
                    |
"406"+AA+BBB+"2"    | - transmit packet 2
   +L+D64...D95     |   L = "0" because packet 3 follows
                    |
"456"+AA+BBB+"2"+X  | - receive acknowledgement for packet 2
                    |   X = "0"  NACK
                    |   X = "1"  ACK
                    |
"406"+AA+BBB+"3"    | - transmit packet 3
   +L+D96...D127    |   L = "0" means more blocks to follow
                    |   L = "1" means last block
                    |
"456"+AA+BBB+"3"+X  | - receive acknowledgement for packet 3
                    |   X = "0"  NACK
                    |   X = "1"  ACK
                    |
Waiting for acknowledgement from peripheral device:
"000"               | - standard question
                    |
"100"+"402"         | - standard answer; request for status of
                    |   parameter transfer
                    |
Receiving acknowledgement:
"402"               | - request for status of parameter transfer
                    |
"452"+AA+X          | - status of device AA
                    |   X = "1"  parameter available / accepted
                    |            by peripheral device
                    |   X = "2"  no communication
                    |   X = "3"  device busy
```

If during the telegram sequence the machine detects any error, it sends as standard answer "100"+"402".

3.8 Write Blocks of Data Initiated by Machine Controller

Transfer specification telegram:

```
"000"                  | - transmit standard question
                       |
"100"+"409"            | - receive request
                       |
"409"                  | - request for transfer specification
                       |
"460"+AASSEECC         | - specification of block transfer (write)
     +P1...P8+"1"      |
                       |
```

Single block write (repeated until the last block):

```
                       |
Write a block:
"406"+AA+BBB+"0"       | - transmit packet 0
   +L+D0...D31         |   L = "0" because packet 1 follows
                       |
"456"+AA+BBB+"0"+X     | - receive acknowledgement for packet 0
                       |   X = "0"  NACK
                       |   X = "1"  ACK
                       |
"406"+AA+BBB+"1"       | - transmit packet 1
   +L+D32...D63        |   L = "0" because packet 2 follows
                       |
"456"+AA+BBB+"1"+X     | - receive acknowledgement for packet 1
                       |   X = "0"  NACK
                       |   X = "1"  ACK
                       |
"406"+AA+BBB+"2"       | - transmit packet 2
   +L+D64...D95        |   L = "0" because packet 3 follows
                       |
"456"+AA+BBB+"2"+X     | - receive acknowledgement for packet 2
                       |   X = "0"  NACK
                       |   X = "1"  ACK
                       |
"406"+AA+BBB+"3"       | - transmit packet 3
   +L+D96...D127       |   L = "0" means more blocks to follow
                       |   L = "1" means last block
                       |
"456"+AA+BBB+"3"+X     | - receive acknowledgement for packet 3
                       |   X = "0"  NACK
                       |   X = "1"  ACK
                       |
Waiting for acknowledgement from peripheral device:
"000"                  | - standard question
                       |
"100"+"402"            | - standard answer; request for status of
                       |   parameter transfer
                       |
```

Receiving acknowledgement:

"402"	- request for status of parameter transfer

"452"+AA+X	- status of device AA
	X = "1" parameter available / accepted by peripheral device
	X = "2" no communication
	X = "3" device busy

If during the telegram sequence the machine detects any error, it sends as standard answer "100"+"402".

3.9 Procedure Success

After every data exchange between central computer and machine controller, it is necessary, for the central computer itself, to know if data have been really acquired also by the peripheral device.
This problem arises, of course, just during writing operations to the peripheral device. No problem is related to the read operations requested by the central computer, because in this case received data themselves can be considered as a valid support to check the end of the communication window.
To be sure of your writing operations, you can take these steps:

1) Follow a communication flow to write data or parameters from the central computer to the machine controller.

2) The machine controller will start to fill the queue, towards the peripheral device, with your data. A writing flag will be set to TRUE.

3) Data will be sent from the machine controller to the peripheral device until the queue will be empty: set writing flag to FALSE.

4) Next poll to the machine controller will return: "100"+"402".

5) The central computer sends "402" and receives "452"+AA+"1". This means that your last writing request has been completed successfully.

3.10 Cancel Option

To offer the possibility to reinitialize the whole gateway system by clearing queues present in the machine controller (i.e. a queue towards EUROMAP 15 and the other one towards EUROMAP 17) the following communication flow can be used:

"407"	- clear queues
"457"	- clear queues acknowledgement

3.11 Queue Size

The central computer can get the size of the queues in the machine controller, using the following communication flow:

```
"408"                | - request size of queues
                     |
"458"+MMMMM+NNNNN    | - receive size of queues (in bytes)
                     |   MMMMM: EUROMAP15 queue size
                     |   NNNNN: EUROMAP17 queue size
```

EUROMAP

Europäisches Komitee der Hersteller von Kunststoff- und Gummimaschinen

European Committee of Machinery Manufacturers for the Plastics and Rubber Industries

Comité Européen des Constructeurs de Machines pour Plastiques et Caoutchouc

Comitato Europeo Costruttori Macchine per Materie Plastiche e Gomma

See you again

http://www.euromap.org