HNS SPC

Statistical process and quality control

MPG - Measuring Program

MPG - Measuring Program

2010 May 12

HNS SPC

Statistical process and quality control

***HNS** Technical Development Co. Ltd.*
H-9027 Győr, Gesztenyefa u. 4., Hungary

Phone: +36 (96) 506-930
Fax: +36 (96) 506-931
E-mail: spc@hns.eu
Web: www.hns.eu, www.hns.eu/spc

*The **HNS SPC** program is a statistical process and quality control program with an own database. The program is capable of collecting data and of processing the collected data. The data collection is primarily based on Measuring Programs but it is possible to input individual data one by one too.*

This description explains the Measuring Program on an engineer level so explains terminology of the Measuring Program, its content as well as its creation and rules.

To understand this description you need to know the HNS SPC program database on logical and end user level. It is also needed to know generally the measurement devices supported by the program included their communication capabilities.

*You can edit measuring program in the menu item **Create** or **Modify** of the **MPG** menu too. By using the Measuring Program Editor to create and edit of measuring programs will be simpler, and following of syntax rules will be automatic. By the way to use the Measuring Program Editor it is needed to know definition, functions and structure of the Measuring Program.*

Contents

Figures

1 Definition of the Measuring Program

The Measuring Program is a prescription relating to performing of a predetermined measurement and qualification series. The program makes the operator to done it step by step. After a successful measurement and qualification series the program stores the collected measurement and analysis results in the database and displays the control chart consisting also the new dates for the operator.

The Measuring Program enables fast and more safety data collection, and in parallel with production it makes possible to automate evaluation.

2 Main features of the Measuring Program

- The Measuring Program can be edited by MPG Editor in HNS SPC program, or in an ***ASCII file*** - according to the syntactic rules valid for the Measuring Program.
- The extension of the Measuring Program file is ***mpg*** (obligatory), because the Measuring Program menu searches the measurement files based on their extension.
- The program searches the files containing Measuring Programs (∗.mpg) in the Measuring Program directory. Path of this directory you can set in the ***Select MPG directory*** menu item of the ***File*** menu.
- It is needed to order Measuring Program files by factory records of the database, so file groups relating to different factories will be in different directories. There is an further possibility to order files inside file groups too.
- A Measuring Program can be prescribed ***related to person***. In this case the program ensures that only the code and password identified operator can perform the task on the lowest access level. You can enable to execute a Measuring Program for more operators too (125 maximum).
- In a Measuring Program can be collected dates only from a given ***product***, but this dates can be recorded also for an other product.
- In the scope of a Measuring Program it is possible to perform ***measurement***s and/or ***qualification***s.
- In the scope of a Measuring Program you can measure and/or qualify ***optional number of products***.
- You can measure and qualify ***variable sample size*** characteristics by the same Measuring Program. (It is not necessary to measure/analyze all characteristics on each products.)
- Measured and attributed samples collected in Measuring Program can be ***recorded to more processes*** together.
- It can be done ***mathematic operations*** by measured values collected in a measuring program. It can be defined and measured in the database ***not identified parameters***. It can be done mathematic operations between these parameters, and recorded results only.
- In the Measuring Program you can prescribe only ***static measurements***, dynamic measurements not except measuring of analogue signs.
- Static and dynamic measurements can be defined in the MPG. Dynamic measurement means turning parts can be measured in the MPG.
- A Measuring Program you can prescribe according to the following measurement strategies. Within a Measuring Program only one measurement strategy is valid (from the followings):
    - measurements and qualifications in ***product [A]*** order,
    - measurement and qualifications in ***parameter [P]*** order,
    - measurements simultaneously by parts in measuring ***device [K]***.
- In the case of ***direct data input*** the measured data can be directly measured with the prescribed measurement device during the Measuring Program but of course data input from the keyboard is also possible.
- Attributed samples can be input ***by parts*** or all in one ***by*** the whole ***sample***.
- It is possible to convert the measured value with the function determined in the Measuring Program so the ***converted measurement value*** will be stored in the database. The converting function can be any member of the given function set, in the case of functions with parameters value of the parameters can determine the person prescribing the Measuring Program.

- During performing of the measurement series of an Measuring Program it is possible to abort the series and either to ***measure again*** last measurement or last measurement unit by chance cancel the whole Measuring Program.
- Performing of the measurement series of an Measuring Program is managed by displaying step by step the sequence number of the individual parts and description of the parameters to measure as well as it is possible to display the ***drawing*** of the parameter to measure to help the measurement.
- Measuring Programs make possible to display ***automatic measuring chart*** and in the case of measuring processes to display measured value diagram. The operator can check the measuring and/or qualification processes controlled or taken care of her directly from the Measuring Program without selecting of the measurement and check results to display.
- It can be displayed control charts for measured or attributed samples collected not in given measuring program (for example automatically, by DataConnect).
- It can be started ***external measuring program*** and recorded collected samples in the database.
- It can be sent ***automatic alert*** by e-mail in case of different events in the course of measuring program, or in case of Out of Control on control chart following of recording the sample.

3 Syntax

During editing in ASCII file of the Measuring Program you must follow the following formal prescriptions otherwise if the program reads files containing Measuring Program, you will get error messages.

General syntactic rules:

- You can prescribe a Measuring Program in an ASCII file. The text can not contain control characters that are illegible.
- The Measuring Program can be described in several lines, the individual lines must be closed with the usual (*'Cr'=13 'Lf'=10*) characters.
- Interpretation of the Measuring Program is done item by item and the sequence of the individual items must follow the rules given in the appropriate sections.
- The items and the physical lines are not the same, because one item can be described in more than one line, it is possible to make readable the text file containing the Measuring Program and to form it. When a item is described in more lines the lines must be closed with an '\' character and directly after it with *'Cr' 'Lf'* characters,
- The interpreted data of individual items must be given between *'{'* and *'}'* brackets and within these brackets all data will be interpreted, included the space and tab characters too. The item part within the brackets will be named as *field to interpret* or in short *field* in what follows.
- In the Measuring Program each line will be considered as a item which has at least one field to interpret.
- A Measuring Program can consists of items only except the so-called note lines, that is each item must contain at least one field to interpret except the note lines.
- The Measuring Program will ignore all text and character which are out of the brackets, except the already mentioned '\' and the *'$'* section start character. Outside the brackets this character you can use only for marking the section start to execute.
- In the Measuring Program you can insert note lines in order to enhance readability of the text file. These lines must be start with an '*' character. The program will ignore these lines.
- The empty lines containing no characters will be considered as note lines so the program skips them.
- Contents of lines prescribing measurements and qualifications, meaning of identifiers and other data what you can place in individual fields, number and sequence of fields are fixed. Except from this rule are the items prescribing automatic, not depending on the operator displaying because in these items number of fields can be changed (in the interest of the retrospectively compatibility).

4 Structure of the Measuring Program file

The file containing the Measuring Program fundamentally is divided into three parts, these sections contain identifiers and data controlling execution.

The individual sections are the next.

- ***Header section***
  It contains general identification data valid for all parameters to measure as well as data used for administration of execution
- ***Describing part of the Control section***
  This section contains the number of items (commands) per types.
- ***Control section***
  This section contains the control items, which prescribe the command, must be performed.

In the file the single sections must follow each other in the above order. The *'$'* character is obligatory first character of the part describing control section because the '$' sign marks the start of the control section *(see description of item describing control section)*. This part is between the two bigger program part (section).

Based on the above structure of the file containing the Measuring Program is as follows.

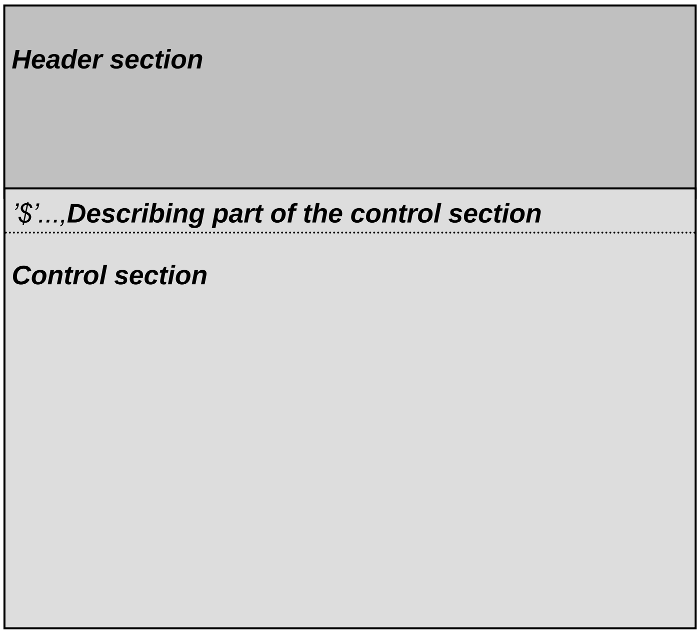

Figure 1: Personal password settings

Note

The individual sections should strictly follow the above order otherwise during decoding of the Measuring Program error messages will be displayed.

4.1 Structure of the header section

The header section of the Measuring Program contains general information related to Measuring Program.

Field number	*Line number*	*Content of the field*	*Field length [characters]*
1	1	Description of the Measuring Program	max. 50
2	2	Name of the person who prescribed the program	max. 20
3	2	ID of the person who prescribed the program	max. 9
4	3	Date of creation	16*
5	4	Date of last execution	16*
6	5	Prescribed execution frequency	max. 16
7	6	ID of the person who execute the program	max. 9 /field
8	7	Optional notes	max. 256
9	8	Execution strategy of the Measuring Program	'*A*' or '*P*' or '*K*'
10	9	Code of the product to measure and/or qualify#	max. 16
11	10	Mask may be empty code	'*EmptyMask*'
12	10	Mask filtering code	'*MaskFilter*'
13	10	Team may be empty code	'*EmptyTeam'*
14	10	Team filtering	'*TeamFilter'*
15	11	HeadTracking function	'*HeadTracking*' or '*noHeadTracking*'

**: according to format {yyyy-mm-dd hh:mm}.*
#: according to the product code defined in the database

Figure 2: Structure of the header section

Detailed description of individual fields:

1. ***Name of the Measuring Program***
   Identification of Measuring Programs and their selection for execution, modification or deletion is based on their name. This name can be max. 50 character long and it is practical to give a name that indicate contents of the Measuring Program.
2. ***Name of the person who prescribed the program***
   Name of the owner of the Measuring Program - max. 20 characters.
3. ***ID of the person who prescribed the program***
   ID of the owner of the Measuring Program - max. 9 characters
4. ***Date of creation***
   The creation date of the Measuring Program contains the fixed format date and time.
5. ***Date of last execution***
   The last execution date is stored in a fixed format field similar to the above. This field is automatically refreshed by the program at executing of the Measuring Program.
6. ***Prescribed execution frequency***
   The execution frequency of the Measuring Program. This field may contain optional text - max. 10 characters.
7. ***ID of the person who execute the program***
   Assigned persons to the Measuring Program. There is place to register persons with max. 9 characters in 125 fields (so max. 125 persons can be registered). If no identification is given then the Measuring Program can be accessible by everyone and in this case it is enough to register one empty field („{}"). More operator you can give in more independent fields (*{1}{2}{3}…*).
8. ***Comment***
   In order to help selecting the Measuring Program it is possible to include a max. 255 characters note too in the Measuring Program beside the name of the program. This note, for example message for the operator will appear in the Measuring Program selection menu.

9. Execution strategy

One character, which alludes to execution strategy. The characters representing the available strategies are as follows (their interpretation see in the chapter *Execution strategies of Measuring Program):*

- ***A***: Product strategy,
- ***P***: Parameter strategy,
- ***K***: Device strategy..

10. Product code

The code (unique identifier) of the product to measure and/or qualify. Length of the field is max. 16 characters according to the product code defined in the database. In the Measuring Program you can refer only products already stored in the database.

11. Mask may be empty code

Obligatory mask ID can switch on for the program (see ***Settings*** menu ***MPG*** menu item). The given measuring program will be exception of this by swithcing on ***Mask may be empty*** code. In this case the content of the field is *{EmptyMask}*, otherwise is empty *{}*.

12. Mask filtering code

Program displays on control chart and measurements diagram for the operator in the measuring program only samples with the mask ID which is equal to mask ID of last recorded sample by switcing on ***Mask filtering*** code. In this case the content of the field is *{MaskFilter}*, otherwise is empty *{}*.

13. Team may be empty code

Obligatory team ID can switch on for the program (see ***Settings*** menu ***MPG*** menu item). The given measuring program will be exception of this by swithcing on ***Team may be empty*** code. In this case the content of the field is *{EmptyTeam}*, otherwise is empty *{}*.

14. Team filtering code

Program displays on control chart and measurements diagram for the operator in the measuring program only samples with the team ID which is equal to team ID of last recorded sample by switcing on ***Team filtering*** code. In this case the content of the field is *{TeamFilter}*, otherwise is empty *{}*.

Note

The filtering which can give in 12th and 14th fields operates only in case of giving mask and/or team ID. If the operator does not record mask and/or team ID, program displayes all samples without filtering. Filtering by mask and team ID can be used independent or together, too.

15. HeadTracking function

By switching on the HeadTracking function program stores head / position number if user refers in *S* item to a sample stemming from a process related to head / position. By displaying the control chart (which contains samples collected by *S* item) user can see control charts for heads / positions separate. The content of this field is *{HeadTracking}* by swiching on this function, otherwise is *{noHeadTracking}*.

Example for defining the header section of the Measuring Program:

```
{Length measurement...}
{Alan Martin}{12345}
{04-04-2000 10.00}
{05-10-2000 15.30}
{every 30 minutes}
{00100}{00101}{00103}
{Take sample from positions of the machine!}
{A}
{1S-1G-1T}
{}{MaskFilter}{}{}
{noHeadTracking}
```

Figure 3: Example for defining the header section

Interpretation of the above Measuring Program header items is as follows:

1. Name of the Measuring Program: ***Length measurement...***[line 1].
2. Measuring Program has been defined by the person with identification code ***12345*** [line 2 / field 2].
3. Measuring Program has been defined by ***Alan Martin*** [line 2 / field 1].
4. Measuring Program has been defined on ***April 4, 2000 10.00 am*** [line 3].
5. Measuring Program was executed last on ***May 10, 2000 15.30 pm*** [line 4].
6. Measuring Program must be performed in ***every 30 minutes*** [line 5] .
7. This Measuring Program can execute operators with the ***00100*** [line 6 / field 1] and ***00101*** [line 6 / field 2] and ***00103*** [line 6 / field 3] personal identification codes.
8. Comment in connected to the Measuring Program: ***Take sample from positions of the machine!*** [line 7].
9. The Measuring Program follows the Product strategy ***A*** [line 8].
10. The Measuring Program applies to the product with the code ***1S-1G-1T***[line 9].
11. There will be displayed on control chart and measurements diagram for the operator only samples with the mask ID which is equal to mask ID of last recorded sample - ***MaskFilter*** [line 10 / field 2].
12. HeadTracking function is off.

Following the syntactic rules you can create the header of the Measuring Program with the same content but in a much clearly arranged and interpretable form too.

The following sample illustrates a header that was created based on the above criteria:

```
************************************************************
* HNS SPC V5.5 program  -  Measuring Program  *
************************************************************

Program description:        {Length measurement...}
        created by:         {Robert Smith}{12345}
                            {04-04-2000 10.00}
Date of last execution:     {05-10-2000 15.30}
Execution frequency:        {every 30 minutes}
Can be executed by:         {00100}{00101}{00103}

Note for the operator:      { Take sample from positions of the machine!}

Execution strategy:         {A}
Analyzed product code:      {1S-1G-1T}
Mask and party field:       {}{MaskFilter}{}{}
HeadTracking:               {noHeadTracking}
```

Figure 4: Extended example for defining header sections

Used syntactic in the above example:

- Lines beginning with the character ‘*‘ (‘*‘ character on the first position) are note lines so these will be skipped by the program.
- The empty lines will be skipped by the program during interpretation of the file containing the Measuring Program.
- In the items describing the Measuring Program you must put the individual fields between ‘{‘ and ‘}’ brackets, because the characters located outside them will be skipped too.

4.2 Structure of the control section

The control section of the Measuring Program contains the prescriptions related to the measured and/or attributed processes, to data storing, to using of external measuring program and for displaying data.

The control section can be established from fifteen different item types (command). These item types are the following based on their function:

- ***M*** type item
  This command describes the static (manual) measurement to execute. The measured value will not be stored directly after execution of the Measuring Program. The measured value can be used to calculate inherited measuring result defined in S type item.
- ***MS*** type item
  This command describes the static (manual) measurement to execute. The measured value will be stored after execution of the Measuring Program at entering data.
- ***MX*** type item
  This command describe the static (manual) measurement to execute, if samples stems from a measured process which is not included in the database. The measured value will not be stored directly after execution of the measuring program at recording data.
- ***MD*** type item
  This command can be used to define a dynamic measurement (measurement of a turning part). The measured value will not be stored directly after execution of the Measuring Program. The measured value can be used to calculate inherited measuring result defined in S type item.
- ***MDS*** type item
  This command can be used to define a dynamic measurement (measurement of a turning part). The measured value will be stored after execution of the Measuring Program at entering data.
- ***MDC*** type item
  This command must be used to define the general settings for dynamic measurements. Only one MDC type item must be defined all the time when MD or MDS type items are defined in the MPG.
- ***S type item***
  In this item you can assemble sample to store by measurement results collected by the *MS*, *M*, *MX, MS and MDS* items.
- ***MV*** type item
  Measuring process descriptor item what appears independently of executed measurements.
- ***A*** type item
  It describes the part qualification to execute. Analysis results will be stored after execution of the Measuring Program, at entering data.
- ***AS*** típusú mondat
  It describes the part qualification to execute by safety data input with OK/NOK marks. Analysis results will be stored after execution of the Measuring Program, at entering data.
- ***A1*** type item
  This command describes asking for totaled data before all other works of executed measurement. Analysis results will be stored after execution of the Measuring Program, at entering data.
- ***A2*** type item
  This command describes asking for totaled data after all other works of executed measurement. Analysis results will be stored after execution of the Measuring Program, at entering data.
- ***AV*** type item
  Attributed process descriptor item what appears independently of qualifications executed in the Measuring Program.
- ***E1*** type item
  This command describes starting of an external measuring program. This external measuring program will start before all other works.

- ***E2*** type item
  This command describes starting of an external measuring program. This external measuring program will start after all other works.

The items consisting of the control section must be placed continuously line by line. The order of item is the following:

MDC ⇨ MD ⇨ MDS ⇨MS ⇨ M ⇨ MX ⇨ S ⇨ MV ⇨ A ⇨ AS ⇨ A1 ⇨ A2 ⇨ AV ⇨ E1 ⇨ E2.

In the control section you can put limited number of items, maximum number of commands what you can use depends on the item type. When you want to prescribe more commands than these maximum, there is possible to grow them with special settings.

Attention!

The items located in the control section can always refer to one process to measure or qualify what is already stored in the database – excepting MX item.

4.2.1 The M type item

Structure of the ***M*** type item and interpretation of the individual fields.

Serial number of the field	*Content of the field*	*Length of the field*
1	*{M}, item type identifier*	1
2	Measurement serial number	max. 3
3	Parameter name	max. 20
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	Number of parts to measure (sample size) or *{?}*	max. 3
9	Gauge *{'type code'::'name'}*#	max. {7::50}
10	Interface code#	max. 6
11	Number of measuring channel# *(counted from 0)* or *{}*	max. 2
12	Serial number of conversion function (identifier)	max. 3
13	First constant of conversion function, K1	max. 8
14	Second constant of conversion function, K2	max. 8
15	Third constant of conversion function, K3	max. 8
16	Fourth constant of conversion function, K4	max. 8
17	Fifth constant of conversion function, K5	max. 8

**: according to the database definition*
#: according to the supported measuring gauge list

Figure 5: Structure of the control section's M type item

Interpretation of the single fields:

1. ***Item type identifier***
   The item type identifier must always be ***M***.
2. ***Measurement serial number***
   The items prescribing measurements can be placed in optional order. Order of measurements depends on these serial numbers and not on the simple order of single measurements. MS, M, MX, A and AS items count in the order. One sequence number can be given only once and the numbering must be continuous.
3. ***Parameter description***
   Unique parameter identifier (name) of the product to measure identified in the header section. In a Measuring Program you can refer only parameters already stored in the database – excepting *MX* item.
4. ***Workgroup code***
   Workgroup identifier (code). In this field you can give identifier of workgroups already stored in the database – excepting *MX* item.
5. ***Machine code***
   Machine identifier (code). In this field you can give identifier of machine already stored in the database – excepting *MX* item.
6. ***Serial number of head***
   The serial number of head on the given machine, or *{0}*, if data collection happens for machine. Here can be refered to a head, which is included in the database, and there is a measured process related to this head – excepting *MX* item. It there is position in an item, than the serial number of head is *{0}* compulsorily.

If there is a *{?}* in this field, the serial number of head must be given directly befor data input.

Head number can be generated automatically by the program – according to rules defined by user. In this case will be input samples other head by the several performing of the measuring program. The head number changes according to rules defined by user. Here should be given beginning value, closing value, automatic increment and actual value – in form *{?,'beginning value','closing value','automatic increment','actual value'}*. Closing value and automatic increment can be maximum equal to head number defined by given machine, beginning value must be smaller than closing value and actual value can not be larger than the head number defined by given machine. If the content of this field is for example *{?,1,4,1}*, than the order of sample is the following: 1st >> 2nd >> 3rd >> 4th >> 1st >> 2nd >> … head.

Defined closing value can be changed by the user who runs the measuring program in case of automatic head number increasing. In this case must be written a *?* before closing value – in form *{?,'beginning value','?closing value','automatic increment','actual value'}*.

7. Serial number of position

The serial number of position on the given machine, or *{0}*, if data collection happens for machine. Here can be refered to a position, which is included in the database, and there is a measured process related to this position – excepting *MX* item. It there is position in an item, than the serial number of position is *{0}* compulsorily.

Further possibilities for giving the position number is equal to writes in head number field, see in point *Serial number of head*.

8. Number of parts to measure

Number of measurements to perform relating on the given parameter.

The number of these measurements not necessarily equal to the sample size of the given parameter since these measurement are not stored directly, these you can use for the samples assembled by help of the *S* type item discussed later.

Sample size is 1, in case of individual / moving average – moving range / moving standard deviation charts.

There should be given number of parts collected to machine capability study, which can be between 1 and 999. If there is *{?}* character in sample size field, user must be given the sample size before data input in measuring program.

9. Gauge code and name

Type of gauge or interface, which will be used by measuring this parameter connected to computer, and name of gauge. It should be written a '*::*' character among type code and name. List of gauges see in *Gauges* document.

10. Interface code

In this field input the measuring gauge interface code. Description of the code you can find in the *Gauges* document.

11. Measuring channel

The third data of measuring gauge definition contains the measurement channel to use. Number them from zero. Of course you can give only that channel what you can interpret with measuring gauge defined in field 9. If the given measuring gauge has no individual channels input *{0}* in this field, or left it empty *{}*. See also *Gauges* document.

12. Conversion function

Settings relating to conversion of measured data. The first item of conversion settings is the a field what contains the code (serial number) of the conversion function to use. The conversion functions are explained in chapter *Conversion function*.

13. First constant of conversion function – K1

A conversion function can have five constants maximum depending on the function type. You can give the first constant in this field. The value can be defined according to the next example:

- *"{123}", the value can have max 7 digits,*
- *"{123.456}", the significant digits of the given number are max 7 and decimal delimiter is '.' .*

14. Second constant of conversion function – K2

Second constant of conversion function. See the writes in point *First constant of conversion function*.

15. ***Third constant of conversion function – K3***
    Third constant of conversion function. See the writes in point *First constant of conversion function*.
16. ***Fourth constant of conversion function – K4***
    Fourth constant of conversion function. See the writes in point *First constant of conversion function*.
17. ***Fifth constant of conversion function – K5***
    Fifth constant of conversion function. See the writes in point *First constant of conversion function*.

Example for the *M* type item of the control section of the Measuring Program:

```
{M}{4}{Length}{1F-1S}{1S-1G}{0}{0}{5}{HNSMUX4::caliper}{LPT1}{2}{0}{0.0}{0.0}{0.0}{0.0}{0.0}
```

Figure 6: Example for the M type item of the control section of the Measuring Program

Interpretation of the above item in field order is as follows:

1. This is an ***M*** type item, so done measurements will not be stored directly in the database.
2. This parameter must be measured after execution of the measurement and/or qualification series in the order of operations **on the fourth place**.
3. The item defines the measurement of ***Length*** named parameter.
4. This parameter must be measured on products produced by the ***1F-1S*** workgroup.
5. The parameter must be measured on the products produced by the ***1S-1G*** code machine inside the ***1F-1S*** code workgroup.
6. The given parameter does not have to be differentiated as heads ***0*** or
7. as positions ***0*** consequently even if this parameter was produced on a multihead or multiposition machine you needn't consider identification of heads and positions of the machine during sampling.
8. Number of parts to measure is ***5***.
9. The given parameter you must measure by a gauge what connects through an ***HNSMUX4*** interface unit to the ***COM1*** port of the computer. The measuring gauge must be connected to the channel ***2*** of the interface unit.
10. The measured value must be converted by the ***0*** serial number function that is there is no conversion, the values of function constants are ***0.0***.

The above item (with the same content) can also be written in a more structured way by following the general syntactic rules. You can see a such Measuring Program part in the following example:

```
{M} only measure\
{4} measured as fourth\
{Length} description of parameter to measure\
{1F-1S}{1S-1G}{0}{0} workgroup and machine code, head and position\
{5}{HNSMUX4}{COM1}{2} type of measuring gauge to use, port, channel\
{0}{0.0}{0.0}{0.0}{0.0}{0.0} the conversion function and the conversion constants
```

Figure 7: Extended example of the M type item of the control section

Used syntactic rules in the above example:

- Fields to decode must be put between brackets '{' and '}' in the items of an Measuring Program. Characters outside these brackets will be skipped by the program.
- One item can be written in more than one line, in this case the continuation must be marked with a '\' character at the end of the line.

4.2.2 The MS type item

Structure of the ***MS*** type item and the individual fields are contained in the following table.

Serial number of the field	*Content of the field*	*Length of the field*
1	*{MS}, item type identifier*	2
2	Measurement serial number	max. 3
3	Parameter name	max. 20
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	Number of parts to measure (sample size) or *{?}*	max. 3
9	Gauge *{'type code'::'name'}*#	max. {7::50}
10	Interface code#	max. 6
11	Number of measuring channel# (*counted from 0)* or *{}*	max. 2
12	Serial number of conversion function (identifier)	max. 3
13	First constant of conversion function, K1	max. 8
14	Second constant of conversion function, K2	max. 8
15	Third constant of conversion function, K3	max. 8
16	Fourth constant of conversion function, K4	max. 8
17	Fifth constant of conversion function, K5	max. 8
18	{def}, Automatic control chart displaying +	3
	or *{def, psl},* Automatic control chart displaying and Process Status Logging+	8

**: according to the database definition.*
#: according to the supported measuring gauge list.
+: not obligatory to give, when you don't want to prescribe automatic displaying, respectively process status logging leave out this field

Figure 8: Structure of the control section's MS type item

Based on the above table the ***MS*** type item is equivalent with the ***M*** type item except for the 1st type marker field.

The difference between the two items is the storing of prescribed measurements. In the case of ***M*** type items the measurement results are not stored directly, while the measurement results described in the ***MS*** type item will be stored in the database after every successful completion of the measurement.

Interpretation of the fields of the ***MS*** type item is similar to the field interpretation of the ***M*** type.

Exception to this, that can be written a '*psl*' entry next to '*def*' entry in the field 18th. '*psl*' means, that process status logging is enabled for this item. '*psl*' should not stand alone, only together with '*def*'.

Example for structure of the ***MS*** type item of the control section:

{MS}{4}{Length}{1F-1S}{1S-1G}{0}{0}{5}{HNSMUX4}{COM1}{2}{0}{0.0}{0.0}{0.0}{0.0}{0.0}

Figure 9: Example for the structure of the MS type item of control section

Interpretation of the above item is equivalent to the previously explained *M* type item, but measurement results will be stored in the database at recording data.

4.2.3 The MX type item

Structure of the ***MX*** type item and the individual fields are contained in the following table.

Serial number of the field	*Content of the field*	*Length of the field*
1	*{MX}, item type identifier*	2
2	Measurement serial number	max. 3
3	Parameter name	max. 20
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	Number of parts to measure (sample size) or *{?}*	max. 3
9	Gauge *{'type code'::'name'}*#	max. {7::50}
10	Interface code#	max. 6
11	Number of measuring channel# (*counted from 0)* or *{}*	max. 2
12	Serial number of conversion function (identifier)	max. 3
13	First constant of conversion function, K1	max. 8
14	Second constant of conversion function, K2	max. 8
15	Third constant of conversion function, K3	max. 8
16	Fourth constant of conversion function, K4	max. 8
17	Fifth constant of conversion function, K5	max. 8
18	Workgroup name*	max. 40
19	Machine name*	max. 25
20	Unit	max. 10
21	Decimals	1
22	Nominal value	No limited. It will be rounded according to number format.
23	Upper difference	
24	Lower difference	
25	Name of the file containing displayed figure	Unlimited.

**: according to the database definition.*
#: according to the supported measuring gauge list.

Figure 10: Structure of the control section's MX type item

The 2nd – 17th fields of ***MX*** item are equal to ***M*** item – according to the table above.

Reason of the difference between the two item is, that in ***M*** item are processes including in the database, but in ***MX*** item are processes not included in the database, so product, parameter, workgroup and machine information must be given in measuring program. Results will not be recorded in the database neither in case of ***MX*** item, they can be used to frame a measured sample, or do mathematic operations between measured values – in ***S*** item.

Interpretation of the single fields, which are different from ***M*** item:

18. Workgroup name
Name of the given workgroup which code is in 4th field. Program does not look for code and name of this workgroup in the database by this item.

19. Machine name
Name of the given machine which code is in 5th field. Program does not look for code and name of this machine in the database by this item.

20. Unit
Unit of the measured parameter.

21. ***Decimals***
    Decimal places for measured parameter.
22. ***Nominal value***
    Nominal value of measured parameter in case of bilateral specification, and empty {} in case of upper or lower bounded specification.
23. ***Upper difference*** / ***Upper specification limit (max)***
    Upper difference of measured parameter in case of bilateral specification, upper specification limit in case of upper bounded specification, and empty {} in case of lower bounded specification.
24. ***Lower difference*** / ***Lower specification limit (min)***
    Lower difference of measured parameter in case of bilateral specification, lower specification limit in case of lower bounded specification, and empty {} in case of upper bounded specification.
25. ***Name of the file containing displayed figure***
    Name of the file contained the figure which will be displayed by measuring of given measured parameter. This file must be placed in the directory selected in *File* menu *Select GPI directory* menu item.

Example for structure of the MX type item of the control section:

```
{MX}{2}{Length}{100-WG}{110-M}{0}{0}{5}{HNSSMUX4::micrometer}{COM1}{1}{0}{0.0}{0.0}{0.0}
{0.0}{0.0}{Workgroup #100}{Machine #110}{mm}{3}{15.30}{0.050}{-0.050}{length.bmp}
```

Figure 11: Example for the structure of the MX type item of control section

4.2.4 The MD type item

Structure of the ***MD*** type item and interpretation of the individual fields.

Serial number of the field	*Content of the field*	*Length of the field*
1	*{MD}, item type identifier*	2
2	View number (position) of measurement	max. 3
3	Parameter name*	max. 20
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	Number of parts to measure (sample size)	max. 3
9	Dynamic measuring mode *{see the code table#}*	*according to need*
10	Etalon size in mm *{see the code table#}*	*according to need*
11	Measuring channels (identifiers) *{see the code table#}*	*according to need*
12	Serial number of conversion function (identifier)	max. 3
13	First constant of conversion function, K1	max. 8
14	Second constant of conversion function, K2	max. 8
15	Third constant of conversion function, K3	max. 8
16	Fourth constant of conversion function, K4	max. 8
17	Fifth constant of conversion function, K5	max. 8

**: according to the database definition*
#: see the Dynamic measurement section!

Figure 12: Structure of the control section's MD type item

Description of fields:

1. ***Item type identifier***
   The item type identifier must always be ***MD***.
2. ***View number (position) of measurement***
   Dynamic measurements are performed by parts in a step (they are displayed in a common window) before the static measurements and qualifications, which are prescribed in the measuring program, therefore this is the first step of the measuring program's running. The number in this field means, in which position is displayed the given measured value in the measuring window - see the chapter '*Dynamic measurements*'. One serial number can be defined only once, but the numbering has not to be continuous.
3. ***Parameter name***
   Unique parameter identifier (name) of the product to measure identified in the header section. In this item you can refer only parameters already stored in the database.
4. ***Workgroup code***
   Workgroup identifier (code). In this field you can give identifier of workgroup already stored in the database.
5. ***Machine code***
   Machine identifier (code). In this field you can give identifier of machine already stored in the database.
6. ***Number of head***
   The number of head on the given machine, or *{0}*, if data collection happens for machine. Here can be referred to a head, which is included in the database, and there is a measured process related to this head – excepting *MX* item. It there is position in an item, than the serial number of head is *{0}* compulsorily.

If there is a *{?}* in this field, the serial number of head must be given directly befor data input.

Head number can be generated automatically by the program – according to rules defined by user. In this case will be input samples other head by the several performing of the measuring program. The head number changes according to rules defined by user. Here should be given beginning value, closing value, automatic increment and actual value – in form *{?,'beginning value','closing value','automatic increment','actual value'}*. Closing value and automatic increment can be maximum equal to head number defined by given machine, beginning value must be smaller than closing value and actual value can not be larger than the head number defined by given machine. If the content of this field is for example *{?,1,4,1}*, than the order of sample is the following: 1st >> 2nd >> 3rd >> 4th >> 1st >> 2nd >> … head.

Defined closing value can be changed by the user who runs the measuring program in case of automatic head number increasing. In this case must be written a *?* before closing value – in form *{?,'beginning value','?closing value','automatic increment','actual value'}*.

7. ***Number of position***
The number of position on the given machine, or *{0}*, if data collection happens for machine. Here can be referred to a position, which is included in the database, and there is a measured process related to this position – excepting *MX* item. It there is position in an item, than the serial number of position is *{0}* compulsorily.

Further possibilities for giving the position number is equal to writes in head number field, see in point *Number of head*.

8. ***Number of parts to measure***
Number of measurements to perform relating on the given parameter.

The number of these measurements not necessarily equal to the sample size of the given parameter since these measurement are not stored directly, these you can use for the samples assembled by help of the *S* type item discussed later.

Sample size is 1, in case of individual / moving average – moving range / moving standard deviation charts.
There should be given number of parts collected to machine capability study, which can be between 1 and 999. If there is *{?}* character in sample size field, user must be given the sample size before data input in measuring program.

9. ***Dynamic measuring mode***
Measuring results are calculated by the method given in this field, based on the several measuring probes used for continuous measurement (and the defined etalon size). One of the measuring mode supported by the program can be chosen, they are defined with an identifier. Possible measuring modes see in the chapter '*Dynamic measurements*'.

10. ***Etalon size***
If etalon size is required to calculate the measuring result of the given parameter, this size has to be given in this field, in mm. If etalon size is not needed, this field has to be left empty. Several measuring modes see in the chapter '*Dynamic measurements*'.

11. ***Identifiers of measuring channels***
Identifiers of measuring channels (measuring probes used for measuring the given parameter) used for calculating the given parameter have to be given in this field. Measuring channels are identified with the number of the measuring channels in order of the multiplexers. It is compulsory to write an 'P' before the number of the measuring channel, and more channels have to be separated with comma - {P1}, {P1,P2}, {P1,P2,P3,P4}, etc. The chapter '*Dynamic measurements*' contains several measuring modes and number of measuring probes needed to it.

12. ***Conversion function***
Settings relating to conversion of measured data. The first item of conversion settings is the a field what contains the code (serial number) of the conversion function to use. The conversion functions are explained in chapter *Conversion function*.

13. ***First constant of conversion function – K1***
A conversion function can have five constants maximum depending on the function type. You can give the first constant in this field. The value can be defined according to the next example:
    - *"{123}", the value can have max 7 digits,*
    - *"{123.456}", the significant digits of the given number are max 7 and decimal delimiter is '.' .*

14. ***Second constant of conversion function – K2***
    Second constant of conversion function. See the section of *First constant of conversion function*.
15. ***Third constant of conversion function – K3***
    Third constant of conversion function. See the section of *First constant of conversion function*.
16. ***Fourth constant of conversion function – K4***
    Fourth constant of conversion function. See the section of *First constant of conversion function*.
17. ***Fifth constant of conversion function – K5***
    Fifth constant of conversion function. See the section of *First constant of conversion function*.

Example for the *MD* type item of the control section of the Measuring Program:

```
{MD}{4}{Length}{1F-1S}{1S-1G}{0}{0}{5}{3}{20.000}{P5}{0}{0.0}{0.0}{0.0}{0.0}{0.0}
```

Figure 13: Example for the MD type item of the control section of the Measuring Program

Interpretation of the above item in field order is as follows:

1. This is an ***MD*** type item, so done measurements will not be stored directly in the database.
2. The measured value of this parameter must be appeared in the fourth position of the measuring screen.
3. The item defines the measurement of ***Length*** named parameter.
4. This parameter must be measured on products produced by the ***1F-1S*** workgroup.
5. The parameter must be measured on the products produced by the ***1S-1G*** code machine inside the ***1F-1S*** code workgroup.
6. The given parameter does not have to be differentiated as heads ***0*** or
7. as positions ***0*** consequently even if this parameter was produced on a multi-head or multi-position machine you needn't consider identification of heads and positions of the machine during sampling.
8. Number of parts to measure is ***5***.
9. The measured value of this parameter is the maximum of diameter. The diameter is measured using one probe.
10. The size of this diameter on the etalon part is ***20 mm***.
11. The diameter is measured by the P5 probe. It means the fifth measuring channel, which is the first channel of the first slave device when HNS INDMUX based measuring system is used.
12. The measured value must be converted by the ***0*** serial number function that is there is no conversion, the values of function constants are ***0.0***.

4.2.5 The MDS type item

Structure of the ***MDS*** type item and the individual fields are contained in the following table.

Serial number of the field	*Content of the field*	*Length of the field*
1	*{MDS}, item type identifier*	3
2	View number (position) of measurement	max. 3
3	Parameter name*	max. 20
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	Number of parts to measure (sample size)	max. 3
9	Dynamic measuring mode *{see the code table#}*	*according to need*
10	Etalon size in mm *{see the code table#}*	*according to need*
11	Measuring channels (identifiers) *{see the code table#}*	*according to need*
12	Serial number of conversion function (identifier)	max. 3
13	First constant of conversion function, K1	max. 8
14	Second constant of conversion function, K2	max. 8
15	Third constant of conversion function, K3	max. 8
16	Fourth constant of conversion function, K4	max. 8
17	Fifth constant of conversion function, K5	max. 8
18	{def}, Automatic control chart displaying +	3
	or *{def, psl},* Automatic control chart displaying and Process Status Logging+	8

**: according to the database definition*
#: see the Dynamic measurement section!
+: not obligatory to give, when you don't want to prescribe automatic displaying, respectively process status logging leave out this field

Figure 14: Structure of the control section's MDS type item

Based on the above table the ***MDS*** type item is equivalent with the ***MD*** type item except for the 1st type marker field.

The difference between the two items is the storing of prescribed measurements. In the case of ***MD*** type items the measurement results are not stored directly, while the measurement results described in the ***MDS*** type item will be stored in the database after every successful completion of the measurement.

Interpretation of the fields of the ***MDS*** type item is similar to the field interpretation of the ***MD*** type.

Exception to this, that can be written a '*psl*' entry next to '*def*' entry in the field 18th. '*psl*' means, that process status logging is enabled for this item. '*psl*' should not stand alone, only together with '*def*'.

Example for structure of the ***MDS*** type item of the control section:

{MDS}{4}{Length}{1F-1S}{1S-1G}{0}{0}{5}{3}{20.000}{P5}{0}{0.0}{0.0}{0.0}{0.0}{0.0}

Figure 15: Example for the structure of the MDS type item of control section

Interpretation of the above item is equivalent to the previously explained *MD* type item, but measurement results will be stored in the database at recording data.

4.2.6 The MDC type item

The ***MDC*** type item describes general settings (gauge interface, measuring ranges, etalon ID, etc.) for dynamic measurements, so MDC type item must be defined all the time when MD or MDS type items are defined in the MPG.

Define only one MDC type item in the MPG!

Structure of the ***MDC*** type item and interpretation of the individual fields.

Serial number of the field	*Content of the field*	*Length of the field*
1	*{MDC}, item type identifier*	3
2	Gauge code*	*according to need*
3	Interface code	max. 5
4	Measuring range of measuring channels	*according to need*
5	Calibration range of measuring channels	*according to need*
6	Digital input of foot switch#	max. 1
7	Etalon ID	max. 32
8	Period of validity of calibration [h]	max. 3
9	Name of the file containing displayed figure	max. 64
10	*{R&R},* R&R support	*3*

**: dynamic measurements can be used with HNS INDMUX-64 {MC4105}, HNS SMUX-4/8/10 or HNS USBMUX-8 only!*
#: for HNS INDMUX-64 {MC4105} only

Figure 16: Structure of the control section's MDC type item

Description of fields:

1. ***Item type identifier***
   The item type identifier must always be ***MDC.***
2. ***Gauge code***
   Dynamic measurements can be used with *HNS INDMUX-64 <MC4105>* and HNS SMUX, HNS USBMUX gauge interfaces (multiplexers), so this field must contains either of codes {MC4105}, {HNSSMUX4}, {HNSSMUX8} and {HNSSMUX10}.
3. ***Interface code***
   The *HNS INDMUX-64 <MC4105>* and the HNS USBMUX gauge multiplexers create a virtual serial port when they are connected to the PC's USB. This port appears in the list of Windows' Device Manager as a new COM port. The port can be reassigned in the Device Manager if it is needed, for example when the port assigned above the COM20. This COM must be defined in this field. Use the number of standard COM port when SMUX is connected to RS232C port of the PC. Supported ports are: COM1, COM2, COM3… COM20.
4. ***Measuring range of measuring channels (measuring range of probes)***
   The number of measuring channels is 64 when the *HNS INDMUX-64 <MC4105>* is used (multiplexer for inductive probes). Using the HNS SMUX/USBMUX, the number of measuring channels is dependent from the multiplexer type (4, 8 or 10 measuring channels for Digimatic gauges). Measuring range (possible maximum movement) of the in-built measuring probes used in the measuring system has to be given in this field. This value means a movement in both (positive and negative) directions from zero point of the measuring probe and has to be given in micrometer [µm]. Accepted deviation of several measuring probes have to be given one after the other, separated with comma. This field always has to be contained 64 values, which is the maximum number of the measuring probes in a possible measuring system, which can be used in HNS SPC program.
   If the actual value of the measuring probe is out of the acceptance range during the measuring, than the measured value is not accepted. Accuracy of the measuring channels can be extended by restricting the acceptance range, or by correct using of the multi-point linearization in case of inductive measuring probes, after the user's demand according to the requirements related to the measured part. It is recommended to set the acceptance range of the measurement on the same value as the physical measuring range of the measuring probe.

5. ***Calibration range of measuring channels***
   Dynamic measurement of certain parameters e.g. diameters requires the calibration of the measuring device. Calibration means measuring a part with known size in the measuring device. It is possible to check the settings of several measuring probes in the measuring program, during the calibration. Checking is made by analysing the displayed values of the measuring probes during the measuring of the master part. Program checks during the calibration, whether the actual values of several measuring probes are inside the acceptance range. Calibration is enabled only if all of the measuring probes are inside the specified acceptance range.
   This value means a movement in both (positive and negative) directions from zero point of the measuring probe and has to be given in micrometer [µm]. Accepted deviation of several measuring probes have to be given one after the other, separated with comma. This field always has to be contained 64 values, which is the maximum number of the measuring probes in a possible measuring system, which can be used in HNS SPC program.
   This function helps to recognize whether the setting of the measuring device or several measuring probes have been wrong, and ensures to prevent the calibration in this case. This function can be used only if zero point of several measuring probes has been set with the given master part in the course of the measuring device's setting. If zero point has been set with another master part, or you do not want to use this function, calibration range of several measuring channels should be set on the same value as the measuring acceptance range of the given measuring channels.
6. ***Digital input of foot switch***
   You can close measuring of a part using a foot switch. You can connect the foot switch to one of digital input lines of the *HNS INDMUX-64 <MC4105>* multiplexer. Define the number of digital input in this field where the foot switch is connected to (the master device has four digital inputs). The foot switch is supported directly by the HNS SMUX and HNS INDMUX. HNS SMUX and USBMUX do not require settings for foot switch.

   Comment
   The DATA buttons of Digimatic gauges are connected to HNS SMUX/USBMUX multiplexers can not be used to triggering data in the dynamic measuring mode.
7. ***Etalon ID***
   Name or code of the etalon part have to be used for gauge calibration.
   Attention, use appropriate etalon to calibrate the measuring gauge, otherwise measured values will be failed!
8. ***Period of validity of calibration [h]***
   The dynamic measuring device is ready to measure, after the calibration is done and the calibration status is valid (if an applied measuring mode require the calibration). Validity time can be assigned to the calibration. The calibration status will be changed to invalid when the validity time is expired. The program automatically asks the calibration if the current status of calibration is invalid.
9. ***Name of the file containing displayed figure***
   Name of picture file that contains the figure which will be displayed while part is under the measuring. This file must be placed in the directory selected in *File* menu *Select GPI directory* menu item.
10. ***{R&R}, R&R support***
   Logging of dynamic measurements can be turned on by setting the {R&R} code into this field. To turn off the logging leave empty this field. The program creates the log file with RnR file name extension in the MPG folder where the MPG files are located too. The RnR log file is a simple, comma separated text file, contains all results of dynamic measurements.

Example for structure of the MDC type item of the control section:

```
{MDC}{MC4105}{COM2}{1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000, ⇨
⇨1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000,1000, ⇨
⇨1000,1000}{100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100,100, ⇨
⇨100,100,100,100,100,100,100,100,100,100,100,100,100}{1}{Master part #1}{8}{Axle.bmp}{}
```

Figure 17: Example for the structure of the MDC type item of control section

4.2.7 The S type item

Before explaining the ***S*** type item let's overview systematization methods of measurements described in the Measuring Program, because role of ***S*** type item in creating samples and storing data will be more clear by understanding these things.

In the control section of the Measuring Program the measurements can be prescribed by *MS*, *M, MX, MD and MDS* type items. Each item represents one measurement unit or sample and these measurement units or samples can consists of different number of individual measurements.

The measurements collected by the Measuring Program and prescribed with *MS*, *M, MX, MD* and/or *MDS* type items can be ordered in the following table (by the above):

Order of MS, M, MX, MD and MDS items	*Measurement results (according to sample size)*									
1st item	1.	2.	3.	4.	5.					
2nd item	1.	2.	3.							
3rd item	1.	2.	3.	4.	5.	6.	7.	8.	9.	→ 50.
4th item	1.	2.	3.	4.	5.	6.	7.			
nth item	1.	2.	3.	4.	5.					

Figure 18: Tabular interpretation of measurement units of control section

The measurement results of the table can be stored in the database in two different ways:

1. Samples formed by the measurement results (in the table these are the individual lines) can be stored with use of *MS* type items (see the chapter of the *MS* type items).
2. The second option is use of the ***S*** type item.

With the ***S*** type item you can create a sample from measurement results of measurement series done by *M*, *MS, MX, MD* or *MDS* type items. This sample can be recorded to a process of the product defined in header section, or can be recorded to a process of an other product.

The ***S*** type item generates the new, directly not measured sample by using the interpreted coordinates of the above measurement results table. User can refer to a measured value with item number, and the number of measured value within the item (this value can be equal to sample size). By the above example the third, last measured result (third column of the table) of the second measured parameter (second row of the table) you can refer with the {2:3} coordinates. This two coordinates must be separated with colon, for example *{1:1}*.

Measured results in the created sample can be make by complex reference. It can be created measured samples from two measured value, by fundamental operation. In case of complex reference must be given the two measured values appointed by coordinates, and the operation which must be done between these, for example *{1:1+2:1}*, *{1:1-2:1}*, *{1:1*2:1}*, *{1:1/2:1}*.

It can be used also functions free of simple mathematics, see in chapter *Functions in the measuring program*.

Of course, the number of the item cannot be bigger, than the total number of the *M*, *MS*, *MX*, *MD* and *MDS* items, and the number of the measured value cannot be bigger, than the sample size related to the given item.

Attention!

The first number in the coordinates (the number of the item) of the measured values cannot be bigger, than the total number of the M, MS, MX, MD and MDS items, and the second number (the number of the measured value) cannot be bigger, than the sample size related to the given item. The number of the item is not determined by the executing order of the items, but according to their order in the measuring program file physically.

You can see the fields forming the ***S*** type item in the next table:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{S}, item type identifier*	1
2	<!> Product code**	max. 16
3	Parameter name*	max. 20
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}*	max. 3
7	Serial number of Position* or *{0}*	max. 3
8	Number of parts to measure (sample size)	max. 3
8+1	Reference to the 1st measurement result#	max. 11
8+2	Reference to the 2nd measurement result#	max. 11
…	…	…
8+n	Reference to the nth measurement result	max. 11
8+n+1	{def}, Automatic control chart displaying+ or *{def, psl}*, Automatic control chart displaying and Process Status Logging+	3 8

**: according to the database definition*
***: if this item does not apply to the product given in the header section, otherwise this field does not exist*
#: coordinate definition, e.g. {1:1}.
+: not obligatory to give, when you don't want to prescribe automatic displaying, respectively process status logging leave out this field

Figure 19: Structure of the 'S' type item of the control section

Example for the ***S*** type item of control section:

{S}{Length}{1F-1S}{1S-1G}{0}{0}{5}{1:1}{2:1}{3:1}{4:1}{5:1}

Figure 20: Example for the S type item of control section

Interpretation of the above item:

1. This is an ***S*** type item, so follows assembling of a sample.
2. The item assembles and records a new sample for the ***Length*** named parameter.
3. This parameter must record to data of the process related to the workgroup coded as ***1F-1S*** in the database.
4. The parameter must record to data of the process related to the machine coded as ***1S-1G*** inside of the workgroup coded as ***1F-1S***.
5. The given parameter does not have to differentiate by heads ***0*** or
6. position ***0***, so the sample must store to the given main process in the assignment of parameter and the given machine, between the data relating to the machine.
7. The sample size is ***5***, to assemble the sample five measurement result coordinates must be given.
8. The first element of the sample is the ***first measured value of the first item*** between *MS, M, MX*, *MD*, *MDS* items.
9. The second element of the sample is the ***first measured value of the second item*** between *MS, M, MX*, *MD*, *MDS* items.
10. The third element of the sample is ***the first measured value of the third item*** between *MS, M, MX*, *MD*, *MDS* items.
11. The fourth element of the sample is ***the first measured value of the fourth item*** between *MS, M, MX*, *MD*, *MDS* items.
12. The fifth element of the sample is ***the first measured value of the fifth item*** between *MS, M, MX*, *MD*, *MDS* items.

The described sample definition can be illustrated as follows:

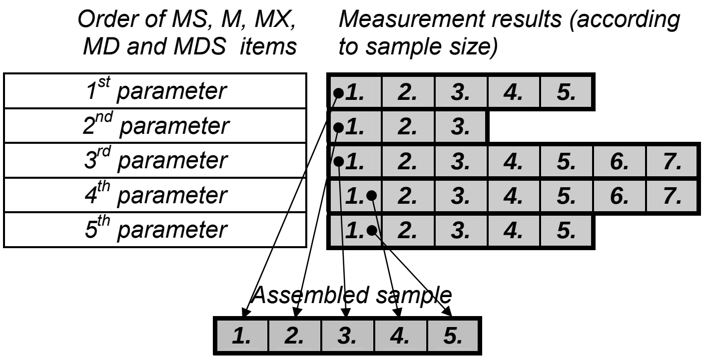

Figure 21: Example for sample definition of the S type item

The above example can be structured rationally as follows:

```
{S}\
{Length}\
{1F-1S}{1S-1G}{0}{0}\
{5}\
{0:0}{1:0}{2:0}{3:0}{4:0}
```

Figure 22: Example for dividing the S type item of the control section

The above example shows creation of samples of process capability study but of course you can use the item for machine capability studies too. The difference between the two analysis methods is only that, that in the case of process capability tests you can use samples consisting of 2-255 measured values (in the case of x_i or ${}_m x$ cards 1), while in the case of machine capability tests only 1 element samples.

4.2.7.1 Functions in the measuring program

The functions can be used in the measuring program and its interpretation are the following.

- ***SET***(x_1) ....: set value (x_1)
- ***ADD***(x_1,x_2) ....: summation (x_1+x_2)
- ***SUB***(x_1,x_2) ....: subtraction (x_1-x_2)
- ***MUL***(x_1,x_2) ....: multiplication (x_1*x_2)
- ***DIV***(x_1,x_2) ....: division (x_1/x_2)
- ***MIN***(x_1,x_2) ....: minimum (*minimum out of* x_1 *and* x_2)
- ***MAX***(x_1,x_2) ....: maximum (*maximum out of* x_1 *and* x_2)
- ***ABS***(x).....................: absolute value ($|x|$)
- ***POW***(x_1,x_2) ....: power ($x_1^{x_2}$)
- ***SQRT***(x) ....: square root ($\sqrt{x}$)
- ***EXP***(x) ....: exponential (e^x)
- ***SIN***(x) ....: sin(x)
- ***ASIN***(x)....................: arc sin(x)
- ***COS***(x).....................: cos(x)
- ***ACOS***(x) ....: arc cos(x)
- ***LOG10***(x).................: base ten logarithm (*log(10)*)
- ***LN***(x)........................: natural logarithm (*ln(x)*)
- ***TAN***(x).....................: tan(x)
- ***ATAN***(x) ....: arc tan(x)

User can refer in functions to measured values and also calculated sample parameters. References are the following.

- ***V****(x:y)* ....: reference to measured value
- ***M****(i)* ....: reference to sample median
- ***A****(i)*...........................: reference to sample mean
- ***R****(i)*...........................: reference to sample range
- ***S****(i)* ....: reference to sample standard deviation
- ***D****(i)*...........................: reference to number of rejected parts in the sample
- ***F****(i)* ....: reference to number of failures in the sample
- ***N****(i)*...........................: reference to attributed sample size

Coordinates of measured value (*x:y*) must be given according to writes in chapter *the S types item*. Serial number of sample (*i*) is the same, but here should be given only the serial number of the item (which is not the executing order of the item).

The number of a measured sample means the number of the item among the *MS*, *M*, *MX*, *MD*, *MDS* items, and the number of an attributed sample means the number of the item among the *A*, *AS* items. The attributed samples which are input in the *A1*, *A2* items, cannot be referred.

Complex functions should be given between brackets embedded interlock. A formula can be maximum 255 character.

Example for the ***S*** type item of control section:

Example for in-build parser in the ***S*** item:

```
{S}{Length}{1F}{1S}{0}{0}{1}{ SUB(MAX(V(1:1);V(2:1)),MIN(V(1:1);V(2:1)))}
```

Figure 23: Example for in-buid parser in the S item

Meaning of the example above: program find maximum value between parts measured in fore two from ***MS***, ***M***, ***MX***, ***MD***, ***MDS*** items, than get also minimum from these, than subtracts minimum from maximum. If recording of measured values measured in two points is not necessary, it is practical to use *MX* item for these.

4.2.8 The MV type item

With the ***MV*** type items you can display control charts of such measurement parameter which measurement results directly stored in the database, independently from the given Measuring Program (for example by HNS DataConnect).

You can see fields of ***MV*** type items in the next table:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{MV}, item type identifier*	2
2	Parameter description*	max. 20
3	Workgroup code*	max. 10
4	Machine code*	max. 10
5	Serial number of Head* or *{0}*	max. 3
6	Serial number of Position* or *{0}*	max. 3
7	{def}, Automatic control chart displaying +	3

**: according to the database definition.*
+: not obligatory to give, when you don't want to prescribe automatic displaying, leave out this field

Figure 24: Structure of the MV type item of the control section

By the above figure interpretation of the fields 2., 3., 4., 5., 6. and 7. of ***MV*** type items are similar to interpretation of equivalent fields of ***S*** type items.

Example for the ***MV*** type item of the control section of Measuring Programs:

```
{S}{Length}{1F-1S}{1S-1G}{0}{0}
```

Figure 25: Example for structure of the MV type items of the control section

Interpretation of the above item in order of the fields is as follows:

1. This is an ***MV*** type item, so you can make possible to display such control charts that have not direct sample recording during Measuring Programs.
2. The item displays the ***Length*** named parameter for the operator.
3. This parameter was measured on the parts produced by the workgroup coded as ***1F-1S*** in the database.
4. This parameter was measured on the parts produced on the machine coded as ***1S-1G***inside of the workgroup coded as ***1F-1S*** in the database,
5. The given parameter does not have to differentiate by heads ***0*** or
6. by positions ***0*** so control chart relating to the main process of the given machine must be displayed.

The above example can be structured reasonable as follows:

```
{MV}\
{Length}\
{1F-1S}{1S-1G}{0}{0}
```

Figure 26: Example for structure of the MV type item of the control section

4.2.9 The A type item

It is possible to asking for qualified data of parts in sample part by part. Attributed samples must be given as occurrence of a failure.

Structure of the ***A*** type item and interpretation of the different fields are in the next table:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{A}, item type identifier*	1
2	Analysis serial number	max. 3
3	Attributed parameter identifier – failure group code*	max. 10
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	{def}, Automatic control chart displaying +	3
	or *{def, psl}*, Automatic control chart displaying and Process Status Logging+	8

**: according to the database definition*
+: not obligatory to give, when you don't want to prescribe automatic displaying, respectively process status logging leave out this field

Figure 27: Structure of the A type item of the control section

Detailed explanation of the individual fields:

1. ***Item type identifier***
   The item type identifier must be ***A***.
2. ***Analysis serial number***
   In the control section you can input items describing the parameters to measure/qualify in any order. Order of measurements does not depend upon the sequence of items but on the sequence numbers given here. *MS*, *M*, *MX*, *A* and *AS* item count by defining order of the items. A sequence number can occur only once and the numbering must be continuous.
3. ***Failure group code***
   The identifier of the failure group to analyze of the product identified in the header. The identifier means the failure group code stored in the database. In a Measuring Program you can refer only failure groups already existing in the database.
4. ***Workgroup code***
   The workgroup identifier (code). You must give workgroup code already existing in the database.
5. ***Machine code***
   The machine identifier (code). In this field you can give also only machine codes already existing in the database.
6. ***Serial number of head***
   The serial number of head on the given machine, or *{0}*, if data collection happens for machine. Here can be referred to a head, which is included in the database, and there is an attributed process related to this head. It there is position in an item, than the serial number of head is *{0}* compulsorily.

   If there is a *{?}* in this field, the serial number of head must be given directly befor data input.

   Head number can be generated automatically by the program – according to rules defined by user. In this case will be input samples other head by the several performing of the measuring program. The head number changes according to rules defined by user. Here should be given beginning value, closing value and automatic increment – in form *{?,'beginning value','closing value','automatic increment'}*. Closing value and automatic increment can be maximum equal to head number defined by given machine, and beginning value must be smaller than closing value. If the content of this field is for example *{?,1,4,1}*, than the order of sample is the following: 1st >> 2nd >> 3rd >> 4th >> 1st >> 2nd >> … head.

7. ***Serial number of position***

The serial number of position on the given machine, or *{0}*, if data collection happens for machine. Here can be referred to a position, which is included in the database, and there is an attributed process related to this position. It there is position in an item, than the serial number of position is *{0}* compulsorily.

Further possibilities for giving the position number is equal to writes in head number field, see in point *Serial number of head*.

8. ***Obligatory control chart displaying***

This field works like a switch. You can switch on the control chart displaying independent of the operator with this filed. You must write *{def}* in this field when you want to display control chart automatically. It can be written a '*psl*' entry next to '*def*' entry. It means, that process status logging is enabled for this item. '*psl*' should not stand alone, only together with '*def*'. When you don't want this leave out this field.

Example for the ***A*** type item of the control section of Measuring Program:

```
{A}{5}{FG-1}{1F-1S}{1S-1G}{0}{0}
```

Figure 28: Example for structure of the A type items of the control section

Interpretation of the above item in the order of fields is as follows:

1. This is an ***A*** type item, what describes part qualification.
2. This failure group must be executed during execution of the Measuring Program as the ***fourth*** element of the measuring and qualification series.
3. This item defines analysis of the ***FG-1*** coded failure group on the parts.
4. The failure group must analyze on the parts produced on the ***1F-1S*** coded workgroup.
5. The failure group on the parts produced by the ***1S-1G*** coded machine inside the ***1F-1S*** coded workgroup,
6. The given parameter does not have to differentiate by heads ***0*** or
7. by positions ***0*** so when this parameter was produced on a multi-head or multi-position machine then during sampling you needn't take care of identification of machine heads and positions.
8. There is not obligatory control chart displaying.

You can write more structured the above item - with the same content - following the general syntactic rules. You can see a such item in the following example:

```
4: {A} parts must be qualified\
    {5} measurement must be executed as the fourth item of data entering\
    {FG-1} failure group code to analyze\
    {1F-1S}{1S-1G}{0}{0} workgroup and machine code, head and position
```

Figure 29: Extended example for structure of the A type item of the control section

Syntactic rules used in the above example:

- Fields to decode must be put between brackets '{' and '}' in the items of an Measuring Program. Characters outside these brackets will be skipped by the program.
- One item can be written in more than one line, in this case the continuation must be marked with a '\' character at the end of the line.

4.2.10 The AS type item

The ***AS*** item make for describing input of attributed samples by parts. It differs from ***A*** item, becuse in this item is safety data input (with OK/NOK marks).

Structure of the ***AS*** type item and interpretation of the different fields are in the next table:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{AS}, item type identifier*	1
2	Analysis serial number	max. 3
3	Attributed parameter identifier – failure group code*	max. 10
4	Workgroup code*	max. 10
5	Machine code*	max. 10
6	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
8	{def}, Automatic control chart displaying +	3
	or *{def, psl}*, Automatic control chart displaying and Process Status Logging+	8

**: according to the database definition*

+: not obligatory to give, when you don't want to prescribe automatic displaying, respectively process status logging leave out this field

Figure 30: Structure of the AS type item of the control section

Based on the above table the ***AS*** type item is equivalent with the ***A*** type item except for the 1st type marker field.

Difference between two item is the way of data input. In case of ***A*** item must be marked occurring failures only, but in case of ***AS*** item must be marked all failures by switching on ***OK*** (suitable) or ***NOK*** (not suitable) record.

Interpretation of the fields of the ***AS*** type item is similar to the field interpretation of the ***A*** type item.

4.2.11 The A1 and A2 type items

It is possible to asking for totalled data of sample analysis with ***A1*** and ***A2*** type items.

With the ***A1*** type item you can ask for totalled data at starting the Measuring Program, with the ***A2*** type item after executing all measurements and qualifications prescribed in the Measuring Program.

Structure of the ***A1*** and ***A2*** type items and interpretation of their fields:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{A1} or {A2},item type identifier*	2
2	Attributed parameter identifier - failure group code*	max. 10
3	Workgroup code*	max. 10
4	Machine code*	max. 10
5	Serial number of Head* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
6	Serial number of Position* or *{0}* or *{?}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}* or *{?,'beginning value','closing value','automatic increment', 'actual value'}*	max. 3 max. {?,3,3,3,3} max. {?,3,?3,3,3}
7	{def}, Automatic control chart displaying +	3
	or *{def, psl},* Automatic control chart displaying and Process Status Logging+	8

**: according to the database definition*
+: not obligatory to give, when you don't want to prescribe automatic displaying, respectively process status logging leave out this field

Figure 31: Structure of the A1 an A2 type items of the control section

Meaning of the fields:

1. The item type identifier must be ***A1*** or ***A2***,

2-7. Meaning of these fields are equivalent to the 3-8. fields of the ***A*** type item.

Example for the ***A1*** type item of the Measuring Program control section:

{A1}{FG-1}{1F-1S}{1S-1G}{0}{0}

Figure 32: Example for structure of the A1 and A2 type item of the control section

4.2.12 The AV type item

With the ***AV*** type item it is possible to display the control charts of a special type of qualification (attributed) processes. Measurement results of these processes are input independently from the given Measuring Program in the database of the program.

Structure of the ***AV*** type item and meaning of its field:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{AV},item type identifier*	2
2	Attributed parameter identifier – failure group code*	max.10
3	Workgroup code*	max.10
4	Machine code*	max.10
5	Serial number of Head* or *{0}*	max. 3
6	Serial number of Position* or *{0}*	max. 3
7	{def}, Automatic control chart displaying +	3

**: according to the database definition*

+: not obligatory to give, when you don't want to prescribe automatic displaying, leave out this field

Figure 33: Structure of the AV type item of the control section

Meaning of the fields:

1. the item type identifier must be ***AV***,

2-7 Meaning of these fields are equivalent to the 3-8. fields of the ***A*** type item.

Example for the ***AV*** type item of the Measuring Program control section:

```
{AV}{FG-1}{1F-1S}{1S-1G}{0}{0}
```

Figure 34: Example for the AV type item of the control section

4.2.13 The E1 and E2 type items

It is possible to run external measuring programs with the ***E1*** and ***E2*** type items directly during executing of an Measuring Program.

With the ***E1*** type item the program starts the external measuring program at starting the Measuring Program, with the ***E2*** type item after executing all measurements and qualifications prescribed in the Measuring Program.

Structure of the ***E1*** and ***E2*** type items and interpretation of their fields:

Serial number of the field	*Content of the field*	*Length of the field*
1	*{E1} or {E2},item type identifier*	2
2	Measuring program	max. 255
3	Parameter	max. 255
4	Pass login ID's to started external program	'*PassLoginIDs*'
5	Start DataConnect at saving of data	'*DataConnect*'

Figure 35: Structure of the E1 and E2 type items of the control section

Meaning of the fields:

1. ***Item type identifier***
   The item type identifier must be ***E1*** or ***E2***, depending on start time of the external measuring program.
2. ***Measuring program***
   The measuring program file to start. Name of the file to run (.Com, .Exe, .Bat) you can give with full path too.
3. ***Parameter***
   When it is needed to transmit parameter to the measuring program, then enter value of the parameter to transmit in this field.
4. ***Pass login ID's to started external program***
   If this field contains *{PassLoginIDs}* switch, than also sample ID's, they are given when measuring program was started will be given to the started external program. Order of ID's is following: sampling date (*yyyy.mm.dd*), sampling time (hh*:mm*), user ID, shift ID, team ID, mask ID, user name. Otherwise is this field empty *{}*.
5. ***Start HNS DataConnect at saving of data***
   External measuring programs can use the HNS DataConnect interface too to transmit measurement results, so there is possibility to start also data transmission trough DataConnect by the program at recording Measuring Program data. When the *{DataConnect}* switch is written in this field the Measuring Program starts the DataConnect at recording data. When you don't want to start the DataConnect left this field empty *{}*.

Example for the ***E1*** type item of the Measuring Program control section:

```
{E1}{Geom-2D.exe}{Test.gpr}{PassLoginIDs}{DataConnect}
```

Figure 36: Example for structure of the E1 type item of the control section

4.3 The control section describing item

At describing structure of files of Measuring Programs it was demonstrated that the describing item of the control section is located between the header and the control section.

The describing item of the control section defines the number of items to interpret in the control section in the form of the following seven values:

1. Number of ***M***, ***MS***, ***MX***, ***MD*** and ***MDS*** type items altogether
2. Number of ***S*** type items.
3. Number of ***MV*** type items.
4. Number of ***A*** and ***AS*** type items.
5. Number of ***A1*** and ***A2*** type items altogether.
6. Number of ***AV*** type items.
7. Number of ***E1*** and ***E2*** type items altogether.[*]

Attention!

*The first character of the describing item of the control section must be always **$**.*

Example for the describing item of the Measuring Program control section:

```
$ MS,M,MX,MD,MDS:{2} S:{1} MV:{1} A,AS:{1} A1,A2:{1} AV:{1} E1,E2:{1}
```

Figure 37: Example for the control section's describing item

Interpretation of the example is the next:

- The given Measuring Program contains 2 parts ***M***, ***MS***, ***MX***, ***MD*** or ***MDS*** type items, that is you must do 2 measurement on every parts.
- The Measuring Program contains 1 part ***S*** type item, that is the program assembles a third sample too by the measured values.
- The Measuring Program contains 1 part ***MV*** type item, that is there will be a fourth measuring control chart in the control chart list of the measuring program beside the three charts prescribed with the previous items.
- The Measuring Program contains 1 part ***A*** or ***AS*** type items that is you must check occurrence of faults of two failure groups on every part by part.
- The Measuring Program contains 1 part ***A1*** or ***A2*** type items, that is you need to give totaled qualification data.
- The Measuring Program contains 1 part ***AV*** type item, that is there will be only two qualification control chart prescribed with *A* type item beside the measuring control charts in the control chart list of the Measuring Program.
- Measuring program contains 1 part ***E1*** or ***E2*** type item, so it will be started an external program in measuring program.

In the control section of the Measuring Program the individual items must be arranged in the above, ***M***, ***MS***, ***MX***, ***MD***, ***MDS*** and ***S*** and ***MV*** then ***A*** and ***AS***, ***A1*** and ***A2***, ***AV*** and finally ***E1***, ***E2*** order.

5 Execution strategies of Measuring Programs

In a Measuring Program you can also define the measuring strategy beside measuring order of individual parameters to measure and checking order of attributed parameters.

In the header of the Measuring Program it is possible to select the most appropriate one to execute the Measuring Program You can select from three measurement strategy. These measurement strategies are as follows:

Code	Description
'A'	product strategy
'P'	parameter strategy
'K'	instrument strategy

Figure 38: Measurement strategies

The product strategy

When you use the product strategy the measurements must be performed in product order, that is you can start to measure and/or qualify the next product only after you have measured and/or qualified all parameters of a given product.

In the case of product strategy order of measurements and qualifications is the next:

	Measurements and qualifications									
Products	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
1st parameter	1.	6.	11.	16.	20.					
2nd parameter	2.	7.	12..							
3rd parameter	3.	8.	13.	17.	21.	24.	26.	28.	29.	30.
4th parameter	4.	9.	14.	18.	22.	25.	27.			
5th parameter	5.	10.	15.	19.	23.					

Measurement and qualification order

Figure 39: Product strategy

The parameter strategy

When you use parameter strategy the measurements and part qualifications must be made in parameter order, that is you can start to measure and/or qualify the next parameter only after you have measured and/or qualified the given parameter on all products.

In the case of parameter strategy process of measurements and qualifications is the next:

	Measurements and qualifications									
Products	1.	2.	3.	4.	5.	6.	7.	8.	9.	10.
1st parameter	1.	2.	3.	4.	5.					
2nd parameter	6.	7.	8.							
3rd parameter	9.	10.	11.	12.	13.	14.	15.	16.	17.	18.
4th parameter	19.	20.	21.	22.	23.	24.	25.			
5th parameter	26.	27.	28.	29.	30.					

Measurement and qualification order

Figure 40: Parameter strategy

The device strategy

The device strategy is a special strategy beside the above two generally applicable strategy. When you use this strategy measurements on the individual products must be done 'simultaneously' that is the program does not stops between measurements and does not waits for pushing of the data input button but continuously reads the measured values measured by the single devices.

Attention!

The device strategy is practically usable only in the case of measurements made in instruments because condition of the measurement is that the stabled values measured by the individual measuring gauges are available simultaneously. Therefore measurements done in more steps are not usable in the case of device strategy. (For example when it is needed to start extra the measurement after gripping of the sample - see description of measuring gauges.)

There is no possibility to use qualification in a Measuring Program in the case of using device strategy.

6 Conversion functions

In the Measuring Program it is possible to perform indirect measurements too. In the case of indirect measurement it is possible to define a conversion and in this case the converted measured value will be stored in the database.

The conversion is single consequently there is possibility to convert only one measurement result and result of the conversion is only one value. More measured value can not be converted into one.
The conversion can be performed with designation of one among the below explained conversion functions. The functions can have constants and these constants can be defined by the user, from the view point of the program the person prescribing the Measuring Program.
You can refer to the individual functions by their code numbers. Designation of codes will be as **f(*code number*)** in the following. Values of individual constants must be given sequentially.

f(*0*) conversion function

Default function, its meaning: ***there is no conversion***.

$$M_k = M_e$$

where

M_k: converted value,
M_e: measured value.

f(*1*) conversion function

General conversion function.

$$M_k = K_1 * M_e^{K_2} + K_3$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: exponent,
K_3: second constant of linear conversion - move -.

f(2) conversion function

General conversion function using the absolute measured value.

$$M_k = K_1 * |M_e| + K_2$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: second constant of linear conversion - move -.

f(3) conversion function

General conversion function using the logarithm of the measured value.

$$M_k = K_1 * \ln(M_e) + K_2$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: second constant of linear conversion - move -.

> ***Attention!***
> *When $M_e \leq 0$ then $M_k = M_e$*

f(4) conversion function

General conversion function using the logarithm of the measured value.

$$M_k = K_1 * \log_{10}(M_e) + K_2$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: second constant of linear conversion - move -.

> ***Attention!***
> *When $M_e \leq 0$ then $M_k = M_e$*

f(5) conversion function

General conversion function using the logarithm of the measured value.

$$M_k = K_1 * \log_{10}(M_e) + K_2$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: second constant of linear conversion - move -,

Attention!

When $M_e \leq 0$ then $M_k = K_4$,
when $M_e < K_3$ then $M_k = K_5$.

f(6) conversion function

General conversion function using exponent.

$$M_k = K_1 * e^{(K_2 * M_e)} + K_3$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: second constant of conversion,
K_3: second constant of linear conversion – set-off -.

f(7) conversion function

General conversion function using power.

$$M_k = K_1 * 10^{(K_2 * M_e)} + K_3$$

where

M_k: converted value,
M_e: measured value,
K_1: first constant of linear conversion - slope -,
K_2: second constant of conversion,
K_3: second constant of linear conversion - move -.

f(8) conversion function

Hole diameter measuring with measuring pins (Three point measuring).

$$Mk = \frac{H^2 + \left(\frac{K_3}{2}\right)^2 - \left(\frac{K_2}{2}\right)^2}{H - \frac{K_2}{2}}$$

where:

$H = H_0 + K_4 * M_e$,

$H_0 = \frac{K_1}{2} + \sqrt{\left(\frac{K_1}{2} - \frac{K_2}{2}\right)^2 - \left(\frac{K_3}{2}\right)^2}$,

M_k: hole diameter,
M_e: measured value,
K_1: standard diameter,
K_2: measuring pin diameter,
K_3: measuring pin fixed axle base,
K_4: measured value correction.

f(9) conversion function

Polynomial conversion.

$$M_k = K_1 + K_2 * M_e + K_3 * M_e^2 + K_4 * M_e^3 + K_5 * M_e^4$$

where:

M_k: converted value,
M_e: measured value,
K_1: constant of polynomial,
K_2: constant of polynomial,
K_3: constant of polynomial,
K_4: constant of polynomial,
K_5: constant of polynomial.

f(10) conversion function

For internal use it is not documented…

f(11) conversion function

For internal use it is not documented…

The conversion functions are included in the program code, consequently the function set can only be extended by changing the program.

7 Dynamic measurement

Dynamic measurements can be prescribed by *MD* and *MDS* items in the measuring program. In case of using dynamic measurement, measuring program has to contain also an *MDC* item. Structure of the items using for dynamic measurements, see in the chapters about *MD*, *MDS* and *MDC* items.

Dynamic measurements are always performed as first step of the measuring program. All of the dynamic measurements prescribed in the given measuring program are performed together (number of dynamic measurements is limited).

Performing of dynamic measurement consists of two steps:

1. measuring device calibration,
2. measuring of the part.

Attention!
Calibration and multi-point linearization of several measuring channels (the measuring channels of the multiplexer with the given measuring probe!) has to be made in the usual way, in the Gauges menu item of the HNS SPC program, apart from the measuring program - see the User manual of the program.

Calibration

Calibration is needed if there is one or more of the prescribed measurements require a known etalon size to calculate the measured value. In this case, following calibration window is displayed at starting the measuring program and at beginning of the measurement.

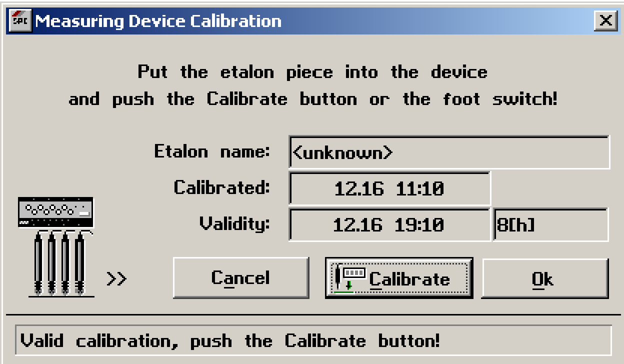

Figure 41: Measuring device calibration window in case of dynamic measurement

When calibration window is displayed, the master part (etalon) has to be placed into the measuring device and the measuring device has to be closed.

Attention!
Always the specified master part has to be used, because using of another master part results incorrect measurement. Identifier of the master part can be displayed on the screen.

To start the calibration, the ***Calibrate*** button in this window has to be pressed. Calibration can be performed by switching a foot switch or external starting button connected to the multiplexer, too.

The above calibration window contains date and validity of previous calibration and the calibration interval. If previous calibration is valid at the time of starting the measuring, data of previous calibration can be used for the measuring, it is not required to calibrate the device before each measuring. If previous calibration is valid, and the measuring device is not wanted to re-calibrate, than the ***Ok*** button of the above window has to be pressed. This possibility is recommended to use when the same measuring program is run repeatedly.

Physical position of several measuring probes are controlled during the calibration continuously; program monitors whether several probes are in the calibration range specified in the *MDC* item of the measuring program. ***Calibrate*** button can be pressed if each measuring probes used in the given measuring program displays a value inside the calibration range.

Comment
Calibration means static measurement, so it is not needed to rotate the master part.

Actual values of several probes can be displayed numerical and by bar chart, too. Actual values can be displayed by clicking the '>>' label in the window.

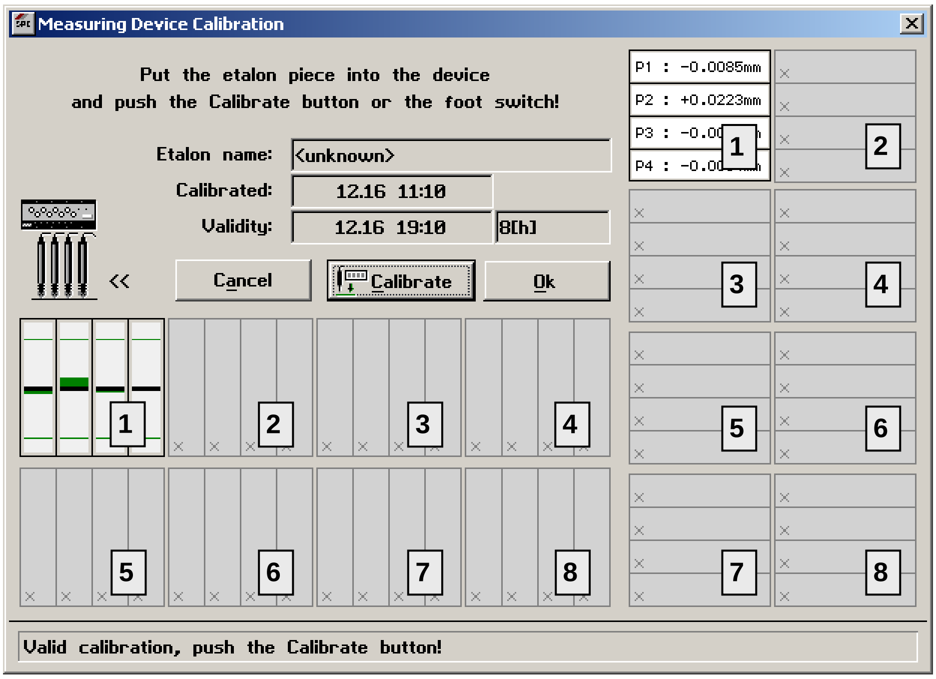

Figure 42: Measuring device calibration window with detailed data

The bar charts in the window shows the actual values of several measuring probes in the acceptance range of the calibration specified in the measuring program. If the actual position of a measuring probe is out of the acceptance range, it is marked with red-colour, and the ***Calibrate*** button is disabled in this case. In this case, it is recommended to check whether the correct master part is in the measuring device, given measuring probe is not dirty or damaged, or the fixture of given measuring probe is not damaged or is in incorrect position. Only the measuring probes used in the given measuring program are shown in this window, independently of the physical construction of the measuring device.

Meaning of the displaying window's fields

1: displayed values of measuring probes on master unit - P1, P2, P3, P4
2: displayed values of measuring probes on first slave unit - P5, P6, P7, P8
3: displayed values of measuring probes on second slave unit - P9, P10, P11, P12
…
8: displayed values of measuring probes on seventh slave unit - P29, P31, P31, P32

Attention!
Calibration data always belong to the given measuring program, therefore if more measuring programs can be run in a measuring device, calibration data and their validity are controlled by measuring programs separately. Calibration data and calibration status of several measuring programs do not affect other measuring program and calibration related to it.

Comment
The above picture is made at the first running of the measuring program, so it has not been made any calibration by this measuring program, accordingly measuring program can be continued only with calibration, and Ok button cannot be used.

Measuring

If calibration has been made or the data of previous calibration has been validated, measuring window is displayed.

As first step of the measuring, in accordance with the comment at the bottom of the window, the part to be measured has to be placed in the measuring device - has to be ready for the measuring. Measuring is started after this, by pressing the ***Measure*** button or the foot switch connected to the multiplexer.

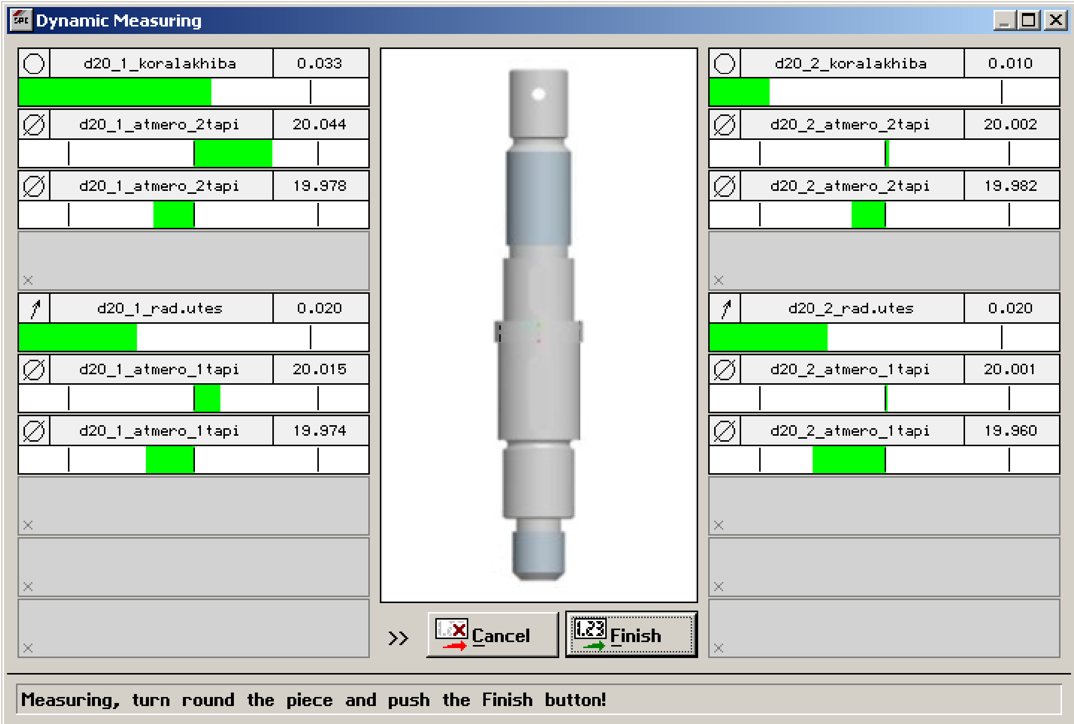

Figure 43: Measuring window in case of dynamic measurement

After starting the measurement of the given part, the specified parameters are measured continuously, and the measured and calculated values are displayed continuously, too. Several measured values are displayed by bar chart and also as numerical value. Measured values are displayed between the tolerance limits, according to the specification of the given parameter. When the above window is displayed, the part has to be rotated, and the dynamic measurement of the given part can be closed by pressing the ***Finish*** button.

Measured data, actual value of several measuring probes can be displayed during the measuring, like during the calibration.

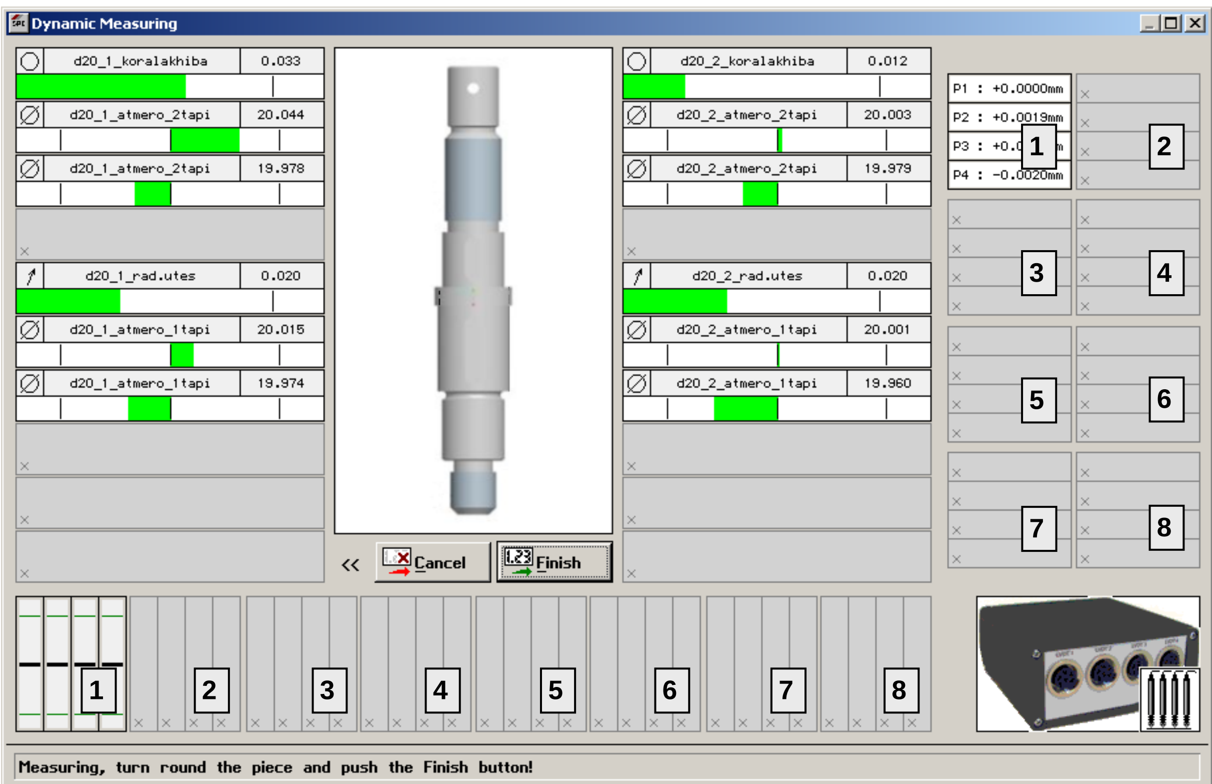

Figure 44: Measuring window with detailed data

Meaning of the displaying window's fields

1: displayed values of measuring probes on master unit - P1, P2, P3, P4
2: displayed values of measuring probes on first slave unit - P5, P6, P7, P8
3: displayed values of measuring probes on second slave unit - P9, P10, P11, P12
…
8: displayed values of measuring probes on seventh slave unit - P29, P31, P31, P32

Measured values are managed after closing the dynamic measurement of the given part in the measuring program in the usual way.

Dynamic measurement modes

	Measured parameter		***Definition***	***Number of probes***	***Position of probes***	***Other condition of the measurement***	***Calculation method***
1	**Diameter**	⌀					
				1		Master part is needed.	$D_{max} = MAX(D_e+2\Delta T)$ $D_{min} = MIN(D_e+2\Delta T)$ $D_{middle} = (D_{max}+D_{min})/2$
				1	Measuring probes fixed in clamp.	Master part is needed.	$D_{max} = MAX(D_e+2\Delta T)$ $D_{min} = MIN(D_e+2\Delta T)$ $D_{middle} = (D_{max}+D_{min})/2$
				2	Measuring probes opposite to each other.	Master part is needed.	$D_{max} = MAX(D_e+\Delta T_1+\Delta T_2)$ $D_{min} = MIN(D_e+\Delta T_1+\Delta T_2)$ $D_{middle} = (D_{max}+D_{min})/2$
2	**Circularity**	○	The tolerance zone is limited in the measuring plane perpendicular to the axis by two concentric circles a distance t apart.	2	Measuring probes opposite to each other.	Diameter has to be measured in more (2→∞) directions by rotating the part.	*Circularity* = $(D_{max}-D_{min})/2$
3	**Cylindricity**	⌭	The tolerance zone is limited by two coaxial cylinders a distance t apart.	n * 2 (n: 2→∞)	Measuring probes opposite to each other and in different sections by pairs.	Diameter has to be measured in more (2→∞) direction in each section (by rotating the part).	*Cylindricity* = $(D_{max}-D_{min})/2$
4	**Parallelism**	//					
a	Parallelism of a surface with reference to a datum line		The tolerance zone is limited by two parallel planes a distance t apart and parallel to the datum line.	2→∞	Measuring probes on the analysed surface.	Master part is needed.	*Parallelism* = $MAX(\Delta T_1,\Delta T_2,..,\Delta T_n)+MIN(\Delta T_1,\Delta T_2,..,\Delta T_n)$
b	Parallelism of a surface with reference to a datum surface		The tolerance zone is limited by two parallel planes a distance t apart and parallel to the datum surface.	n * 2 (n: 2→∞)	Measuring probes on the two end-surfaces opposite to each other by pairs.	Measuring in more direction by rotating the part. Master part is needed.	*Parallelism* = $MAX(MAX(\Delta T_1+\Delta T_2);$ $MAX(\Delta T_3+\Delta T_4);$ $..;$ $MAX(\Delta T_n+\Delta T_{n+1}))$
5	**Perpendicularity**	⊥					
	Perpendicularity of a surface with reference to a datum line		The tolerance zone is limited by two parallel planes a distance t apart and perpendicular to the datum line.	2→∞	Measuring probe on the end-surface on different radiuses.	Master part is needed.	*Perpendicularity = Total axial run-out*
6	**Concentricity and coaxiality**	◎					

a			The tolerance zone is limited by a circle of diameter t, the centre of which coincides with the datum point.	2	Measuring probes opposite to each other.	Measuring in more direction by rotating the part. Master part is needed.	*Concentricity =* $MAX(ABS(\Delta T_1-\Delta T_2))$
b			The tolerance zone is limited by a cylinder of diameter t , the axis of which coincides with the datum axis, if a ∅ sign is standing before the tolerance value.	n * 2 (n: 2→∞)	Measuring probes opposite to each other and in different sections by pairs.	Measuring in more direction by rotating the part. Master part is needed.	*Coaxiality =* $MAX(ABS(\Delta T_1-\Delta T_2);$ $ABS(\Delta T_3-\Delta T_4);$ ..; $ABS(\Delta T_n-\Delta T_{n+1}))$
7	**Radial run-out and total run-out**	↗ ⌰					
a	Radial run-out		The tolerance zone is limited in the measuring plane perpendicular to the axis by two concentric circles a distance t apart, the common centre of which lies on the datum axis.	1	Measuring probe on the cylinder-cover.	Part has to be turned about the datum axis with a whole turn.	*Run out =* $MAX(T)-MIN(T)$
b	Total radial run-out		The tolerance zone is limited by two coaxial cylinders a distance t apart the axis of which lies on the datum axis.	2→∞	Measuring probes on the cylinder-cover in more directions.	Part has to be turned about the datum axis with a whole turn.	*Run out =* $MAX(MAX(T_1)-MIN(T_1);$ $MAX(T_2)-MIN(T_2);$ ..; $MAX(T_n)-MIN(T_n))$
c	Axial run-out		The tolerance zone is limited in each radial position by two circles a distance t apart which are in a measuring cylinder the axis of which lies on the datum axis.	1	Measuring probes on the end-surface.	Part has to be turned about the datum axis with a whole turn.	*Run out =* $MAX(T)-MIN(T)$
d	Total axial run-out		The tolerance zone is limited by two parallel planes a distance t apart and perpendicular to the datum axis.	2→∞	Measuring probes on the end-surface on more radiuses.	Part has to be turned about the datum axis with a whole turn.	*Run out =* $MAX(MAX(T_1)-MIN(T_1);$ $MAX(T_2)-MIN(T_2);$ ..; $MAX(T_n)-MIN(T_n))$
8	**Distance**						
a				1			$h_{max} = MAX(h_e+\Delta T_1)$ $h_{min} = MIN(h_e+\Delta T_1)$ $h_{middle} = (h_{max}+h_{min})/2$
b				2			$h_{max} = MAX(h_e+\Delta T_1+\Delta T_2)$ $h_{min} = MIN(h_e+\Delta T_1+\Delta T_2)$ $h_{middle} = (h_{max}+h_{min})/2$

Figure 45: Dynamic measurement methods

8 Measuring program preselection

Measuring program selection menu can handles limited parts (maximum 100) measuring programs which are in a given directory (suggested to use a separate directory). This limit (100 parts) is not always enough to record in one directory all measuring programs related to a given factory, so it it necessary to change MPG directories.

1. Select MPG directory for given workplace I.

MPG directory can be selected separately, irrespectively to actual database directory. It can be selected for all workplaces a several MPG directory which contains only measuring programs used by this workplace. In all workplaces are installed HNS SPC program and are connected to a common network database. In this case can be reached several measuring programs by all workplaces, maximum 100 parts severally.

2. Select MPG directory for given workplace II.

First solution can be used also by install HNS SPC program in a common network place in a copy which can be reached by all workplaces. In this case use all workplaces a common MPG directory, as default. It is possible to use several settings (so also several MPG directories) by several workplaces. This can be set in Registry, see in document *Special settings*.

3. Measuring program preselection

First and second solutions are suitable for handling the limit related to number of measuring programs. Measuring program preselection gives a sterling solution for handling of limitless parts measuring programs theoretically. By this function can be reached all measuring programs in all workplaces, and measuring programs can be order in logical groups by several aspects (for example by product, machine, workplace etc.).

First step is creating measuring programs, than must be these order in groups by some aspect (for example product or machine). One group must be contained maximum 100 measuring programs, but limitless parts groups can be made and handled theoretically. This groups must be placed in several directories, than must be placed a navigation file in the directory wherein are placed all of the group directories. This file contains name of the groups (this name will be displayed for operator), and directories, wherein are placed measuring programs in a group.

Name of navigation file is ***$PS-NAV$.MPG***. Content and format of the file is following:

'Name of the group'='Name and path of the directory containing given group"

This file contains so much line, as much groups are created.

Attention!
Directories containing MPG groups and navigation file must be places in selected MPG directory.

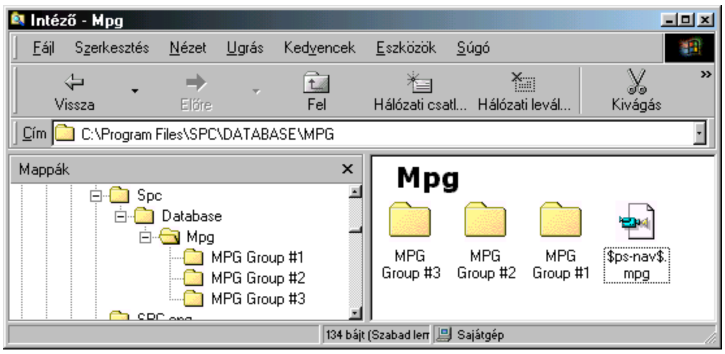

Figure 46: Parameter strategy

Content of $*PS-NAV$.MPG* file in the example above:

MPG Group #1=C:\Program Files\SPC\DATABASE\MPG\MPG Group #1
MPG Group #2=C:\Program Files\SPC\DATABASE\MPG\MPG Group #2
MPG Group #3=C:\Program Files\SPC\DATABASE\MPG\MPG Group #3

Following MPG preselection window will be displayed for operator:

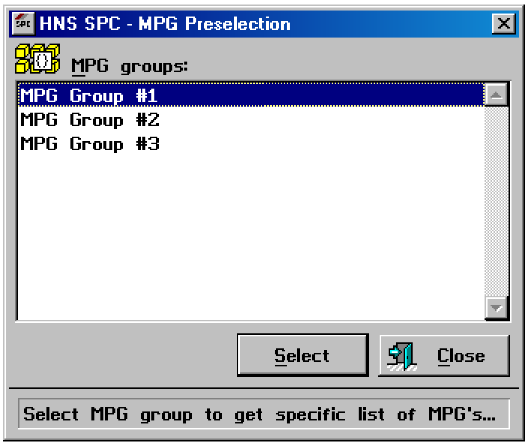

Figure 47: MPG preselection window

If selected MGP directory contains *$PS-NAV$.MPG* file, first should be select between MPG groups defined in this file. After selection a MPG group enters program into selected MPG group directory, and displays list of measuring programs found in this directory. Here can be selected the given measuring program.

4. Multilevel measuring program preselection

Measuring program preselection can consist of more - maximum three - level. In case of multilevel preselection should be placed further subdirectories and navigation file into subdirectory containing given MPG group with content presented above.

By clicking on ***Exit*** button in ***MPG preselection*** or in ***MPG selection*** windows steps the program into previous level, as default. This can be modified when program steps in main window directly. This can be set in Registry, see in document *Special settings*.

Example for multilevel MPG preselection:

Put the case that we produce two product group (*A* and *B*), within three – three products (*A1*, *A2*, *A3*, *B1*, *B2* and *B3)* in given factory. Product *A* is checked in three step (after *operation #10*, *#20* and *#30*), and product *B* is checked in two step (after *operation #10*, and *#20*). These product are produced by two – two equal machine (*11*, *12*, … *32*) in several operations. Separate measuring programs have made for checking after several operations and for several machines.

Measuring programs are following:

- *Product A1 after operation #10 by machine #11*
- *Product A2 after operation #10 by machine #11*
- *Product A3 after operation #10 by machine #11*
- *Product A1 after operation #10 by machine #12*
- *Product A2 after operation #10 by machine #12*
- *Product A3 after operation #10 by machine #12*
- *Product A1 after operation #20 by machine #21*
- *Product A2 after operation #20 by machine #21*
- *Product A3 after operation #20 by machine #21*
- *Product A1 after operation #20 by machine #22*
- *Product A2 after operation #20 by machine #22*
- *Product A3 after operation #20 by machine #22*
- *Product A1 after operation #30 by machine #31*
- *Product A2 after operation #30 by machine #31*
- *Product A3 after operation #30 by machine #31*
- *Product A1 after operation #30 by machine #32*
- *Product A2 after operation #30 by machine #32*
- *Product A3 after operation #30 by machine #32*
- *Product B1 after operation #10 by machine #11*
- *Product B2 after operation #10 by machine #11*
- *Product B3 after operation #10 by machine #11*
- *Product B1 after operation #10 by machine #12*
- *Product B2 after operation #10 by machine #12*
- *Product B3 after operation #10 by machine #12*
- *Product B1 after operation #20 by machine #21*
- *Product B2 after operation #20 by machine #21*
- *Product B3 after operation #20 by machine #21*
- *Product B1 after operation #20 by machine #22*

- *Product B2 after operation #20 by machine #22*
- *Product B3 after operation #20 by machine #22*

First level is the separation measuring programs of A and B product groups. After this measuring programs belonging to a product group are separated by operations, than by machine. In this case will be occurred measuring programs related to a given operation and given machine for all products of given product group.

Created directory structure is following:

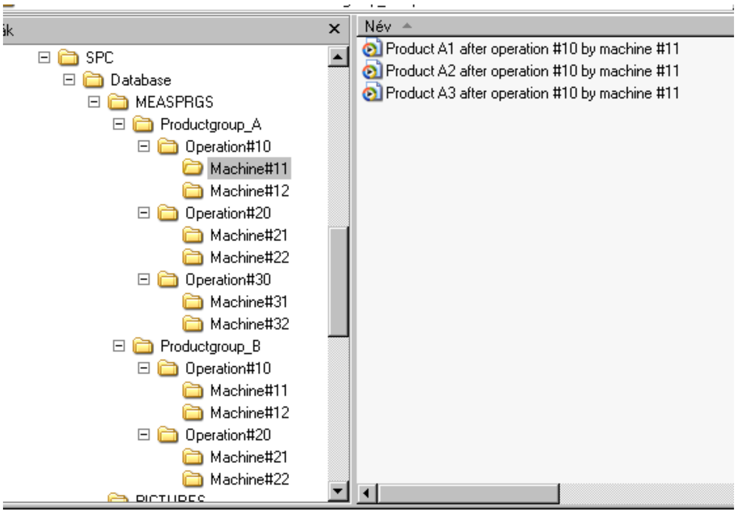

Figure 48: Example for directory structure by multilevel MPG preselection

Following figure shows the content of navigation file on first level (which is in *C:\SPC\Database\MeasPrgs* directory), and the window displayed by this.

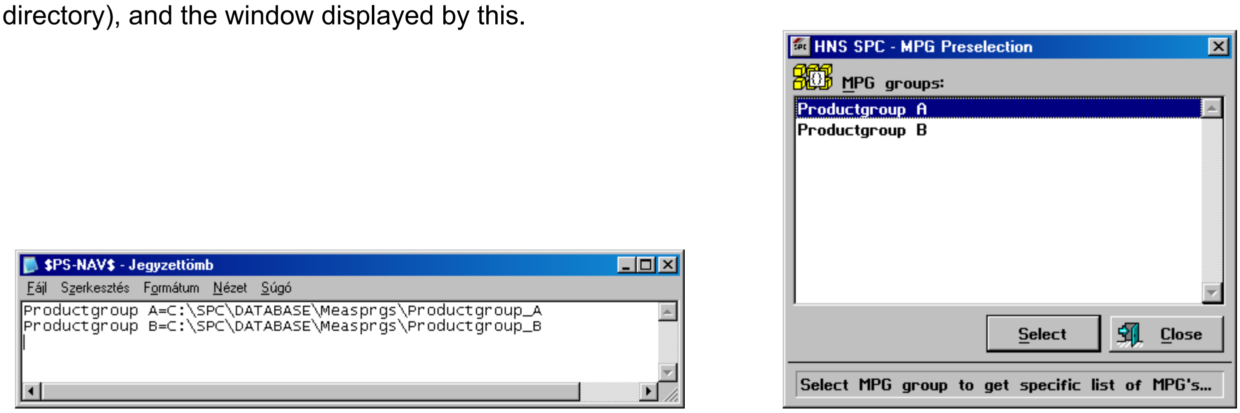

Figure 49: Example for multilevel MPG preselection window – 1st level

Following figure shows the content of navigation file on second level (which is in *C:\SPC\Database\MeasPrgs\Productgroup_A* directory), and the window displayed by this.

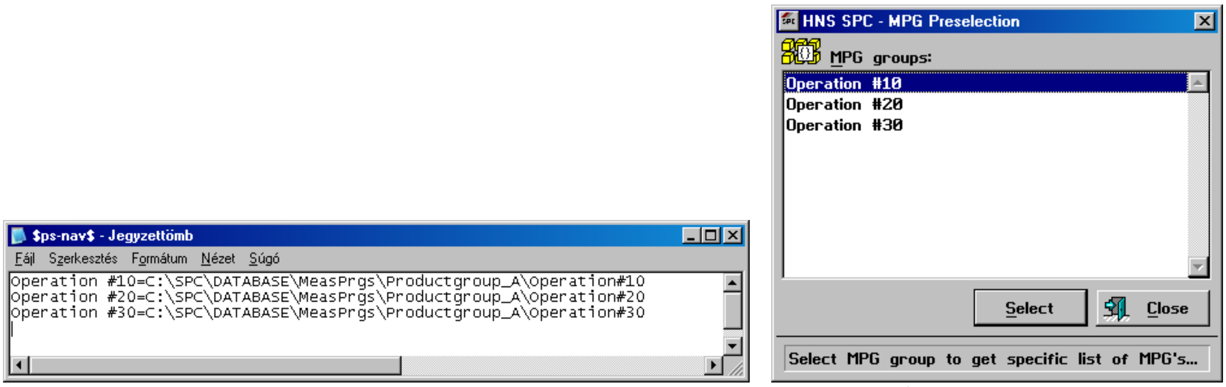

Figure 50: Example for multilevel MPG preselection window – 2nd level

Following figure shows the content of navigation file on third level (which is in *C:\SPC\Database\MeasPrgs\Productgroup_A\Operation#10* directory), and the window displayed by this.

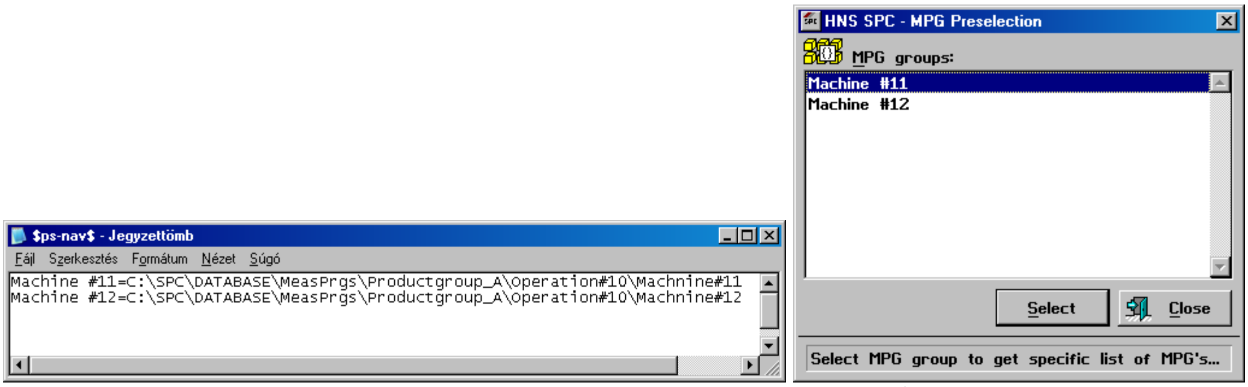

Figure 51: Example for multilevel MPG preselection window –3rd level

After selection of *Machine #11* from *Operation #10* and *Productgroup A* will be displayed following measuring program list.

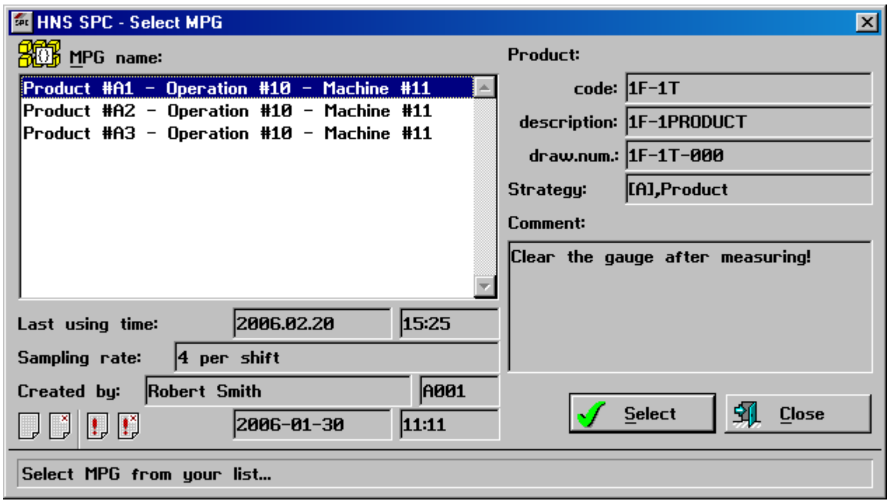

Figure 52: Mesuring program selection

9 CSV file input

Data input in MPG can be made from a CSV file.

[>>] Loading data form CSV file {A,P} option should be selected in gauge list. ***CSV >>*** button will be activate for this, and by pressing of the button ***CSV Data Source*** window will be displayed, wherein relevant settings should be given.

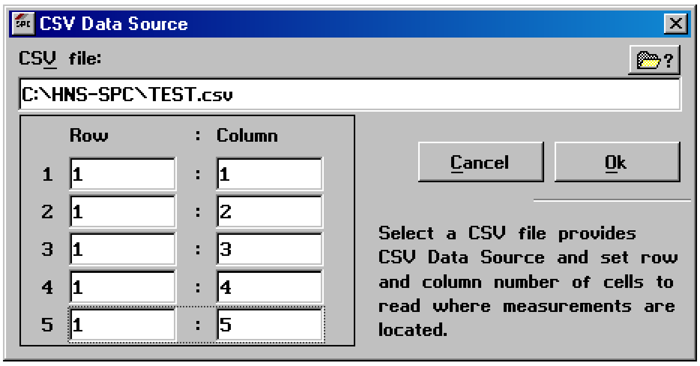

Figure 53: CSV data source settings

Name and source path of ***CSV file*** should be given in the right field. The place of sample's measured values in the CSV file should be given by giving ***row*** and ***column*** numbers. Size of this table is equal to sample size which is valid for this item.

Program checks CSV file by given file name and source path before running of MPG. If given file cannot be found, program send a error message.

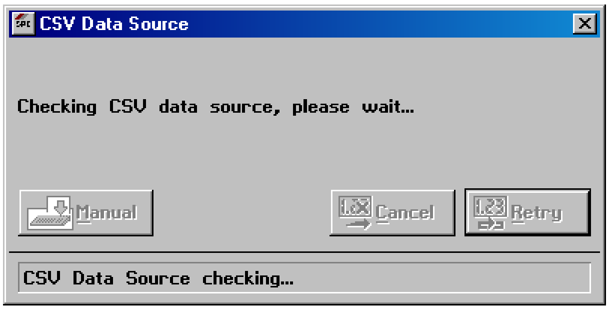

Figure 54: Ckecking CSV data source

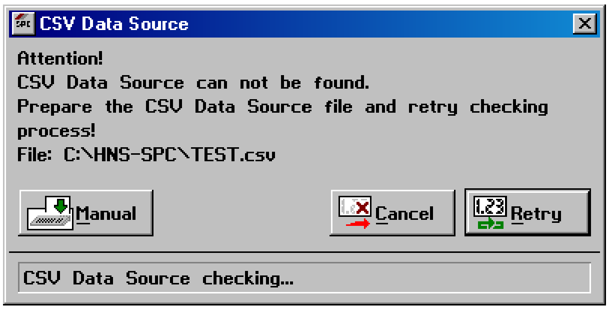

Figure 55: CSV Data Source cannot be found

Loading of measured values signs displayed ***CSV Data Source – Loading*** window.

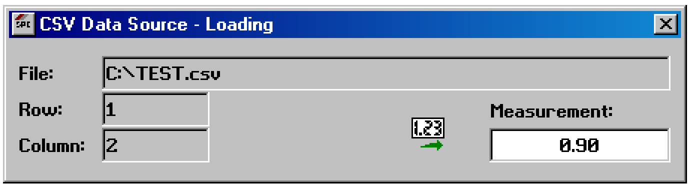

Figure 56: CSV Data Source loading

Displaying of ***CSV Data Source – Loading*** can be turn out by a *Registry*, by this data input will be faster.

Program renames the CSV file after loading in such a way that put a number to the filename and an extension '*sbk*'. The added number starts from 0, and program find lowest number which is not exist so far.

CSV data source is identified in MPG as follows:

gauge code:	*CSV*
interface code:	*file name and source path*
channel number:	place of the measured value (row, column) measured value's places are separated with comma (,)

For example: *{CSV}{C:\HNS-SPC\TEST.csv}{1:1,1:2,1:3,1:4}*

Measured values can be separated in CSV file with semicolon (;) or TAB.

Note

You can indentify the place of measure values simply if you open CSV file in an Excel table.

10 Process Status Logging (PSL file)

There will be made a new entry in this file by running a measuring program. A row of this file means an MPG item. Fields are separated with comma (,).

A row is built by measured parameter as follows.

1. '*M*' (measured parameter)
2. sampling date and time
3. computer ID, whereon MPG was run
4. workgroup code
5. machine code
6. head number (or *0*)
7. position number (or *0*)
8. product code
9. measured parameter name
10. user ID, who ran MPG
11. shift ID
12. team ID
13. mask ID
14. tool ID
15. sample size (number of measured parts)
16. number of parts out of tolerance
17. minimum measured value
18. maximum measured value
19. average of measured value
20. range of measured value (difference between minimum and maximum measured value)
21. standard deviation of measured value
22. '*OK*' / '*NOK*' decision if there is an Out of Control case on the process's control chart in reference to measured sample
23. '*OK*' / '*NOK*' decision if process distribution is equivalent to adjusted distribution after sample input
24. '*OK*' / '*NOK*' decision if process capability indices are suitable after sample input
25. or '---', if capability indices are not calculated

A row is built by attributed parameter as follows.

1. '*A*' (attributed parameter)
2. sampling date and time
3. computer ID, whereon MPG was run
4. workgroup code
5. machine code
6. head number (or *0*)
7. position number (or *0*)
8. product code
9. failure group code
10. user ID, who ran MPG
11. shift ID
12. team ID
13. mask ID
14. tool ID
15. sample size (number of measured parts)
16. faulty parts
17. '*OK*' / '*NOK*' decision if there is an Out of Control case on the process's control chart in reference to measured sample

11 Examples for measuring program

Single measuring program

```
*********************************************************************
*                    HNS SPC – MPG file                             *
*********************************************************************
{Measuring Program example}
{Robert Smith}{A0001}
{2004-04-04 16:00}
{2005-04-04 16:00}
{4/shift}
{vk}{nz}{vi}
{Clear the gauge after measuring!}
{A}
{1F-1T}
{}{}{}{}
{HeadTracking}
```

```
$ MS,M,MX,MD,MDS:{3} S:{3} MV:{1} A,AS:{1} A1,A2:{1} AV:{1} E1,E2:{0}
```

```
{MS}{1}{TEST-1}{1F-1S}{1S-1G}{0}{0}{5}{MANUAL}{}{}{0}{0.0}{0.0}{0.0}{0.0}{0.0
{MS}{2}{TEST-2}{1F-1S}{1S-1G}{0}{0}{5}{MANUAL}{}{}{0}{0.0}{0.0}{0.0}{0.0}{0.0}
{MS{3{TEST-3}{1F-1S}{1S-1G}{0}{0}{7}{MANUAL}{}{}{0}{0.0}{0.0}{0.0}{0.0}{0.0}
{S}{TEST-7}{1F-1S}{1S-1G}{0}{0}{3}{1:1}{2:1}{3:1}{def}
{S{TEST-7}{1F-1S}{1S-1G}{0}{0}{3}{1:2}{2:2}{3:2}{def}
{S}{TEST-7}{1F-1S}{1S-1G}{0}{0}{3{1:3}{2:3}{3:3}{def}
{MV}{TEST-4}{1F-1S}{1S-1G}{0}{0}
{A}{4}{FG-1}{1F-1S}{1S-1G}{0}{0}
{A1}{FG-1}{1F-1S}{1S-1G}{0}{0}
{AV}{FG-1}{1F-1S}{1S-1G}{0}{0}
```

Example for measuring program – with comments.

```
*******************************************************************
*                    HNS SPC – MPG file                           *
*******************************************************************
Name:                              { Measuring Program example }
Created by:                        {Robert Smith}{A0001}
                                   {2004-04-04 16:00}

Used:                   {2005-04-04 16:00}
Sampling rate:          {4/shift}
Dedicated to:          {vk}{nz}{vi}               << user ID's (max.125)
Comment:               {Clear the gauge after measuring!}
Strategy:              {A}
Product code:          {1F-1T}
Mask and team:     {}{}{}{}…..<< mask and team filter and and empty mask and empty filter
                                     are disabled
HeadTracking       : {HeadTracking}    << HeadTracking function is off

$ Measure/attribute: MS,M,MX,MD,MDS:{3} S:{3} MV:{1} A,AS:{1} A1,A2:{1} AV:{1} E1,E2:{0}

1: {MS} measured + recorded \
   {1} first in the order \
   {TEST-1} parameter name \
   {1F-1S}{1S-1G}{0}{0} workgroup and machine code, head and position \
   {5} sample size \
   {MANUAL}{}{} gauge, port, chanel \
   {0}{0.0}{0.0}{0.0}{0.0}{0.0} conversion function, constants of function
2: {MS}\
   {2} \
   {TEST-2} \
   {1F-1S}{1S-1G}{0}{0} \
   {5} \
   {MANUAL}{}{} \
   {0}{0.0}{0.0}{0.0}{0.0}{0.0}
3: {MS} \
   {3} \
   {TEST-3} \
   {1F-1S}{1S-1G}{0}{0} \
   {7} \
   {MANUAL}{}{} \
   {0}{0.0}{0.0}{0.0}{0.0}{0.0}
4: {S} recorded \
   {TEST-7} parameter name \
   {1F-1S}{1S-1G}{0}{0} workgroup and machine code, head and position \
   {3} sample size \
   {1:1}{2:1}{3:1} created sample \
….{def} chart automatically
5: {S} \
   {TEST-7} \
   {1F-1S}{1S-1G}{0}{0} \
   {3} \
   {1:2}{2:2}{3:2}
….{def} chart automatically
6: {S} \
   {TEST-7} \
   {1F-1S}{1S-1G}{0}{0} \
   {3} \
   {1:3}{2:3}{3:3}
….{def} chart automatically
```

12 Error messages when decoding the Measuring Program

During decoding Measuring Program error messages can occur in the program because of syntactic errors and database collision problems.

Structure of error messages is as follows:

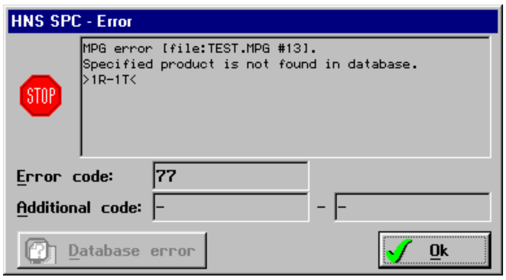

Figure 57: Error messages during Measuring Program decoding

In the first line of error messages you can see the information describing location of the error between brackets.

- Name of erroneous Measuring Program file (in the above example name of the file is *TEST.PRG*).
- Serial number of erroneous line is marked with '# (in this example the 13th line of the Measuring Program file is incorrect).

Possible error messages their meanings and correction are the next:

12.1 File errors

Measuring Program file error. The 'file-name' file is not accessible.

Decoding of Measuring Program happens in two steps – the header section and then the control section -, the error message appears when a file with correct header is not accessible during the second step.
The error can also be caused by disk error but it is more probable that the file has been deleted via the network and in this case the error can not be corrected.

File format error of file is damaged.

The defined Measuring Program file can not be opened.
The file is incorrect or may be it is illegally busy via the network so check the DOS file and the network status.

12.2 Errors in items

The command line can not be interpreted, it is too long.

The defined line of Measuring Program is too long - the 'Cr''Lf' line end code is missing - the line has more than 500 characters.
Check and correct the incorrect line

'DATA START' character is missing.

The starting '{' character of the Measuring Program field to interpret is missing. Syntactic error, or the given item has less fields to interpret than the needed.
Check and correct the appropriate line, define the missing fields.

'DATA STOP' character is missing.

The ending '}' character of Measuring Program field to interpret is missing. Syntactic error, check and correct the appropriate line.

12.3 Errors in the header

Name of measuring program is invalid. >_incorrect field_**<**

The description field is incorrect, the field is empty or the given name is too long.
Syntactic error, check and modify the incorrect description.

Operator identifier is invalid. >_incorrect field_**<**

One of the fields containing the operator codes is incorrect, it is empty or the defined code is too long.
Syntactic error, check and modify the incorrect code.

Invalid MPG comment. >_incorrect field_**<**

The field containing the Measuring Program note is incorrect, the text defined in the field is too long.
Syntactic error, check and define shorter note.

Invalid MPG strategy. >_incorrect field_**<**

Definition of execution strategy of the Measuring Program is incorrect, the given strategy code has more than one character or the given character can not be interpreted.
The correct strategy codes are as follows:

- {A},
- {P},
- {K}.

Syntactic error, check and modify the incorrect strategy code.

> *Attention!*
> *The strategy code must be given in capital letters.*

Specified product identifier is invalid. >_incorrect field_**<**

In the header section of the Measuring Program definition the code of the product to check is incorrect, the field is empty or too long.
Syntactic error, check and modify the incorrectly defined code.

Specified product is not found in database. >_incorrect field_**<**

Specified product is not found in database.
Only products stored in the database can be referred, so you must define the desired products in the database then you will be able to execute the Measuring Program.
Logical error, check the defined product code and the product information of the database.

12.4 Errors in the control section describing item

Invalid command line identifier. >_incorrect field_**<**

In the control section of the Measuring Program one of the item type identifiers can not be interpreted.
The correct item types of the control section of Measuring Programs are as follows:

- ***M***, ***MS***, ***MX***, ***MDC***, ***MD*** and ***MDS*** type items,
- ***S*** type items,
- ***MV*** type items,
- ***A*** and ***AS*** type items,
- ***A1*** and ***A2*** type items,
- ***AV*** type items,
- ***E1*** and ***E2*** type items.

Syntactic error, check and modify the incorrectly defined code.

> *Attention!*
> *The item type identifier must be given in capital letters.*

Number of measured parameters items – M/MS/MX/MD/MDS – is invalid or the measuring program is damaged.

Definition of control section item of Measuring Program is incorrect, there is a mismatch between the control section and its description item.
Logical error, check the number of ***M***, ***MS***, ***MX***, ***MD*** or ***MDS*** and ***S*** type items included in the control section and definition of the items in the describing item.

Number of measured parameter items – MV – is invalid or the measuring program file is damaged.

Definition of control section item of Measuring Program is incorrect, there is a mismatch between the control section and its description item.
Logical error, check the number of ***MV*** type items included in the control section and definition of the items in the describing item.

Number of failure group items – A/AS/A1,A2 – is invalid or the measuring program file is damaged.

Definition of control section item of Measuring Program is incorrect, there is a mismatch between the control section and its description item.
Logical error, check the number of ***A***, ***AS*** and ***A1***, ***A2*** type items included in the control section and definition of the items in the describing item.

Number of failure group items – AV – is invalid or the measuring program file is damaged.

Definition of control section item of Measuring Program is incorrect, there is a mismatch between the control section and its description item.
Logical error, check the number of ***AV*** type items included in the control section and definition of the items in the describing item.

12.5 Errors in the control section

Specified order of measuring and attributing is invalid. >*incorrect field***<**

In the control section one of the ***M*** or ***MS*** and ***MD*** or ***MDS*** type items includes incorrect sequence number. In this field the sequence number must be defined by number, for example {1} or {2} or {14} etc…
Syntactic error, check and define correctly the measurement sequence number.

Invalid parameter identifier. >*incorrect field***<**

In the Measuring Program control section one of the items contains incorrect measurement parameter identifier - description -, the field is empty or too long.
Syntactic error, check and modify the incorrectly defined description.

Specified failure group identifier is invalid. >*incorrect field***<**

In the Measuring Program control section one of the items contains incorrect failure group identifier - description -, the field is empty or too long.
Syntactic error, check and modify the incorrectly defined description.

Specified parameter is not found in database. >*incorrect field***<**

The defined parameter to measure of the defined product is not included in the database.
You can referred only parameters included in the database, so first you have to define the desired parameter in the database then you will be able to execute the Measuring Program.
Logical error, check the description of the defined parameter and the parameter information of the database.

Specified failure group is not found in database. >*incorrect field***<**

The defined failure group of the defined product is not included in the database.

You can referred only failure groups included in the database, so first you have to define the desired failure group in the database then you will be able to execute the Measuring Program.
Logical error, check the code of the defined failure group and the failure group information of the database.

Invalid workgroup identifier. >*incorrect field*<

In the Measuring Program control section one of the items contains incorrect workgroup identifier - workgroup code -, the field is empty or too long.
Syntactic error, check and modify the incorrectly defined code.

Specified workgroup is not found in database. >*incorrect field*<

The defined workgroup is not included in the database
You can refer only workgroups included in the database so first you have to define the workgroup in the database then you will be able to execute the Measuring Program.
Logical error, check the identifier of the defined workgroup and the workgroup information of the database.

Invalid machine identifier. >*incorrect field*<

In the Measuring Program control section one of the items contains incorrect machine identifier - machine code -, the field is empty or too long.
Syntactic error, check and modify the incorrectly defined code.

Specified machine is not found in the database. >*incorrect field*<

Specified machine is not found in the database.
You can refer only machines included in the database so first you have to define the machine in the database then you will be able to execute the Measuring Program.
Logical error, check the identifier of the defined machine and the machine information of the database.

Invalid head number. >*incorrect field*<

The defined head identifier can not be interpreted.
In this field the serial number of heads counted from 1 or ‘0’ must be defined. The serial number must be a number, for example {1} or {2} or {14}, etc.
Syntactic error, check the appropriate Measuring Program field and correct the number.

Specified head is not found in database. >*incorrect field*<

The defined head is not interpreted with the previously identified machine, the defined machine either has no heads or the head sequence number is bigger than number of heads of the machine.
Logical error, check the head sequence number and the machine information of the database.

Invalid position number. >*incorrect field*<

The defined position identifier can not be interpreted.
In this field the sequence number of the given position or ‘0’ must be defined similar to the head. The sequence number is defined by a number, for example {1}, or {2} or {14}, etc.
Syntactic error, check the appropriate Measuring Program field and correct the sequence number.

Specified position is not found in database. >*incorrect field*<

The defined position can not be interpreted with the previously defined machine, the defined machine either has no positions or the given position number is bigger than the number of machine positions.
Logical error, check the defined position number and the machine information of the database.

Both head and position can not be defined at same time, either of them must be zero.
Head and position number cannot be defined at the same time in an item. At least one of them must be zero.

Syntactic error, change head or positon number into zero.

***Invalid sample size.* >*incorrect field*<**

The defined sample size - number of measurements - is incorrect, the field can not be interpreted.
In this field the number of measurements must be given by numbers and this number can not be negative or ‘0’, for example {2} or {5} or {50}, etc.
Syntactic error, check the defined sample size and define the number correctly.

***Invalid gauge identifier.* >*incorrect field*<**

The defined measuring gauge code can not be interpreted. You can give only codes listed in the measuring gauges section.
Syntactic error, check the measuring gauge code to use and give it correctly.

> *Attention!*
> *The measuring gauge codes must be defined in capital letters – by the manual.*

***Invalid gauge interface port.* >*incorrect field*<**

The defined interface port of the measuring gauge can not be interpreted. Only those codes can be used which are listed in the measuring gauge section.
Syntactic error, check the interface codes and define the correct code.

> *Attention!*
> *The measuring gauge interface code must be defined in capital letters according to the manual. In the case of direct keyboard input the field must be empty “{MANUAL}{}{}”.*

Specified gauge can not be connected to specified interface port.

The measuring gauge can not be interfaced in the given way.
Logical error, since the referred interface can not be interpreted with the defined measuring gauge - for example in the case of serial line measuring gauge the defined port code is parallel. Consequently you must check the measuring gauge interface code and give the correct interface code by this.

> *Attention!*
> *In the case of keyboard input the field must be empty “{MANUAL}{}{}”.*

***Invalid gauge channel.* >*incorrect field*<**

The defined channel number is incorrect, the field can not be interpreted.
In this field the number of measuring channel must be defined by numbers (counted from 0) and this number can not be negative, but for example {0} or {1} or {2}, etc.
Syntactic error, check the defined channel number and correct the number.

> *Attention!*
> *In the case of keyboard input the field must be empty “{MANUAL}{}{}”.*

***Invalid gauge channel, this gauge has not the specified channel.* >*incorrect field*<**

The defined channel number is incorrect, because the previously identified measuring gauge does not have the defined channel. The given serial number is bigger than number of channels of measurement or interface unit.
Logical error, check the defined channel number and give it by the given device.

> *Attention!*
> *In the case of keyboard input the field must be empty “{MANUAL}{}{}”.*

***Invalid conversion function number.* >*incorrect field*<**

The defined conversion function code can not be interpreted. Code of the function to use must be defined by numbers. This code can only be one of the functions contained in the function set description. The function code can only be defined by numbers for example {0} or {1}.

Syntactic error, check the identifier field of the function content and give the correct function code.

Specified conversion function is not found in list of valid conversions. >*incorrect field***<**

The defined conversion function code does not exists, this function is not included in the function code set.
Logical error, check the defined function identifier and select a function which can be interpreted by the program.

Invalid conversion function parameter. >*incorrect field*** <**

The defined function parameter (constant) can not be interpreted, the content of the field is not a number.
Logical error, check the defined function parameter and input number satisfying the syntactic rules, for example {123} or {123.456}, etc.

Unrecognised coordinates or parser string. > *incorrect field* **<**

The 'measured value co-ordinate' definition given in the ***S*** type item of the Measuring Program is incorrect and can not be interpreted *(see the description of the S type item)*.
Syntactic error, check the defined co-ordinate and input it again according to the defined syntactic rules, for example {0:0} or {2:5}, etc.

Coordinates of measured value are invalid. >*incorrect field***<**

The 'measured value co-ordinate' defined in the ***S*** type item of the Measuring Program does not exist, the given co-ordinates do not identify any measurement defined by ***MS***, ***M***, ***MX***, ***MD*** and ***MDS*** type items.
Logical error, check the defined co-ordinate and the number of prescribed measurements.

Assembled sample items of MPG –S- must be follow the measured sample items.

In the Measuring Program the ***S*** type items can only be located after the ***MS***, ***M***, ***MX***, ***MD*** and ***MDS*** type items because the program interprets 'the measured value co-ordinates' in the ***S*** item based on these items.
Logical error, check the sequence of the ***MS***, ***M***, ***MX***, ***MD***, ***MDS*** and ***S*** type items in the control section.

Failure group items must be follow the measured parameter items.

In the measuring program the ***A***, ***AS*** and ***A1***, ***A2*** type items can only be located after the ***MS***, ***M***, ***MX***, ***MD***, ***MDS***, ***S*** és ***MV*** type items.
Logical error, check the sequence of the ***MS***, ***M***, ***MX***, ***MD***, ***MDS***, ***S***, ***MV*** and ***A***, ***AS***, ***A1***, ***A2*** type items in the control section.

12.6 Processing error

The memory is insufficient for executing the Measuring Program.

During execution of a Measuring Program the results are stored in the main memory of the computer. If there is no enough memory available then this message will appear on the screen before executing the Measuring Program.
Check the determined number of measurements in the control section and if necessary divide the given Measuring Program into more smaller Measuring Programs which can be executed individually.