

# NC data format 4.12 for standard import

## NC data format 4.12 standard import

1.1	General .....	3
1.1.1	Conventions for names .....	3
1.1.2	Forbidden characters in the file system of Microsoft Windows operating systems .....	3
1.1.3	Conventions for dimensions .....	4
1.1.4	Conventions for prices .....	6
1.1.5	Conventions for dates .....	6
1.1.6	Conventions for orientation .....	6
1.1.6.1	Base of orientation .....	6
1.1.6.2	Orientation of the board .....	6
1.1.6.3	Orientation board to rip cut saw .....	6
1.1.6.4	Orientation of surface textures .....	6
1.1.6.5	Orientation between format and board .....	7
1.1.7	Format combining .....	8
1.1.8	Remainders .....	9
1.1.9	Filler .....	9
1.2	Data records saw area .....	10
1.2.1	General information .....	10
1.2.2	Part list data records .....	11
1.2.2.1	Internal construction of a part list file .....	11
1.2.2.2	Part-list-base-line (L 01) .....	12
1.2.2.3	Part-list-database-line (L 06) version 3.0 .....	13
1.2.2.4	Part-list-status-line (L98) .....	14
1.2.2.5	Part list statistics line (L99) .....	15
1.2.2.6	Part-list-end-line (L\$) .....	15
1.2.3	Format data records .....	16
1.2.3.1	Internal construction of a format block .....	16
1.2.3.2	Format-base-line (F 01) .....	17
1.2.3.3	Format-stacking-line (F 02) .....	22
1.2.3.4	Format-customer-text-line (F 10-29) .....	23
1.2.3.5	Format-packing-tag-base-line (F 05) .....	23
1.2.3.6	Format-packing-tag-text-line (F 30-49) .....	25
1.2.3.7	Format-label-base-line (F 06) .....	25
1.2.3.8	Format-label-text-line (F 60-79) .....	27
1.2.3.9	Format-top-surface-line (F 07) .....	28
1.2.3.10	Format-bottom-surface-line (F 08) .....	29
1.2.3.11	Format-post-processing-line (F 80) .....	30
1.2.3.12	Format strip handling line (F 81) .....	33
1.2.3.13	Format drilling program line (F 82) .....	33
1.2.3.14	Format-window-line (F 84) .....	34
1.2.3.15	Format-groove-line (F 85) .....	35
1.2.3.16	Format-milling-line (F 86) .....	37
1.2.3.17	Format-angle-line (F 87) .....	39
1.2.3.18	Format-miter-line (F 88) .....	40
1.2.3.19	Format handling line (F 90) .....	41
1.2.3.20	Format statistics line (F99) .....	42
1.2.4	Post processing data records .....	43
1.2.4.1	Internal construction of post processing files .....	43
1.2.4.2	Edge base line (K 01) .....	44

1.2.4.3	Edge-database-base-line (K 02) .....	46
1.2.4.4	Edge-end-line (K\$) .....	46
1.2.5	Board data records .....	47
1.2.5.1	Internal construction of a board file .....	47
1.2.5.2	Board-base-line (B 01) .....	48
1.2.5.3	Board-customer-text-line (B 10-19) .....	51
1.2.5.4	Board-end-line (B\$) .....	51
1.2.5.5	Board-database-base-line (B 02) .....	52
1.2.5.6	Board-statistic-line (B99) .....	52
1.3	Data Records Stacking Area .....	53
1.3.1	General .....	53
1.3.2	Format Data Records .....	54
1.3.2.1	Format stacking line (F 02) .....	54
1.3.3	Packing Specifications .....	58
1.3.3.1	Internal structure of packing-instruction files .....	58
1.3.3.2	Packing-instruction data base: .....	58
1.3.3.3	Packing-instruction base line (W 01) .....	59
1.3.3.4	Packing-instruction text line (W 10-19) .....	63
1.3.3.5	Packing-instruction data-base base line (W 02) .....	63
1.3.3.6	Packing-instruction end line (W\$) .....	63
1.3.4	Strapping instruction .....	64
1.3.4.1	Internal structure of strapping-instruction files .....	64
1.3.4.2	Strapping-instruction base line (U 01) .....	64
1.3.4.3	Strapping-instruction data-base base line (U 02) .....	67
1.3.4.4	Strapping-instruction end line (U\$) .....	67
1.3.5	Strapping rules .....	68
1.3.5.1	Internal structure of the strapping-rules files .....	68
1.3.5.2	Strapping-rule header line (U 11) .....	68
1.3.5.3	Strapping-rule base line (U 12) .....	69
1.3.5.4	Strapping-rule end line (U\$) .....	69
1.3.6	Protection-board data records .....	70
1.3.6.1	Internal structure of protection-board files .....	70
1.3.6.2	Protection-boards base line (X 01) .....	71
1.3.6.3	Protection-board data-base base line (X 02) .....	72
1.3.6.4	Protection-board end line (X\$) .....	72

This document was translated by software.

All Headlines with no content refer to information which is for internal use only.

## 1.1 General

### 1.1.1 Conventions for names

For names 20 alphanumeric characters are designated. All characters are usable, also the characters forbidden for the file system. These characters are made usable by a conversion. That is necessary, because particularly in America the characters are used [ / ] and [ ' ] for the representation by dimensions.

The input of the characters can take place in small and capital letters. Comparisons are executed however always case insensitive. ‚Lauf1 ‘ is thus equivalent to ‚lauf1 ‘ and to ‚LAUF1 ‘.

### 1.1.2 Forbidden characters in the file system of Microsoft Windows operating systems

- Forbidden characters are replaced by permitted characters. The replacement character is written after a permitted special character. As special character the character ‚is used ~ ‘, because it is used otherwise only very rarely. Thus it is comprehensible with high probability whether the characters permitted now are the result of an interchange or whether they were input by the user.
- The point character is permitted for file names. It is replaced however likewise, in order to avoid mistakes with the distinction between file name and extension. At several points in a file name Windows uses the final point for the separation of extension.
- With each operation, with which the file system is used, must take place if necessary a conversion of term. Such operations are e.g. file look up, files read, to file write etc..

Forbidden character		Replacement character	
Character	ASCII	Character	ASCII
/	047	~a	126,097
'	034	~b	126,098
:	058	~c	126,099
*	042	~d	126,100
?	063	~e	126,101
\	092	~f	126,102
<	060	~g	126,103
>	062	~h	126,104
	124	~i	126,105
.	046	~j	126,106

For the representation and the input of names it must look in such a way, as if all characters would be permitted. The forbidden characters are however kept away from the

internal file system. The user notices only, if he regards the internal file structure with the Explorer, which actually not is necessary. If the user does nevertheless, he must know that the forbidden characters were replaced. That will immediately become clear however an experienced user. An inexperienced user should not change the internal file structures.

Procedure with the import of boards into the board data base:

- With the import of boards the board type is tested for forbidden characters. If it contains forbidden characters, for which no replacement character is available, the import is rejected.
- If no forbidden characters are contained or if a replacement character is defined, the import is executed. The board type is taken over unchanged. In the file name in the board data bank the forbidden characters are replaced by the replacement characters.

Procedure during the input of board types:

- If a name is input, which contains forbidden characters without replacement characters, the name is rejected. Otherwise it is exactly the same stored, as it was input, thus if necessary with the forbidden characters.
- In the board data base the board with the settled name is looked up. The settled name is created from the stored or selected board type using the replacement characters. If the board does not exist, it can be input as beforehand described.

Procedure with the creation by NC data:

- If boards from the data base are used, the board type must be tested for the replacement characters during the NC data creation. If in the board type replacement characters are contained, they must be replaced again by the forbidden characters.

### 1.1.3 Conventions for dimensions

There is admissible three different systems of units:

- Metrically decimally (code 0): All metric mass eight-digit, integer type and in 1/1000 mm are indicated. Example: 1390mm corresponds to 01390000. When measuring system mm, resolution is 1/10 mm. The size range therefore sufficient from 0.100 mm to 99999.900 mm.
- Decimal inch (code 1) and
- Factional inch (code 2): In this case all mass integer type in 1/1024 inch are indicated. For American users an input must be in the formats fractional inch and decimal inch possible. Are stored inch the mass always in 1/1024, because as a result of it no errors arise when rounding decimal format on fraction format.

The input format for fractional inch is composed of three sections: Integer, counter and denominator. As separators can be used blanks, commas, diagonal strokes and hyphens. If the integer is equal to zero, it can be omitted. Improper fractions are shortened. The denominator of fractional inch values can take only the values 2, 4, 6, 8, 16, 32, 64 and 128.

Examples:	3 2 8	becomes	3 1/4
	1,2	becomes	1/2
	17 3/4	becomes	17 3/4
	0-5-16	becomes	5/16
	1,7/2	becomes	4 1/2



### **1.1.4 Conventions for prices**

All prices are indicated ten-digit and integer type in 1/100 currencies. That results in a range from 0,01 to 100,000,000 currencies. That should be sufficient also with currencies such as Italian Liras.

### **1.1.5 Conventions for dates**

All dates are stored with ten ASCII characters with the format ,[ jjjj-mm-tt ' ]. As internal type of variable ,SYSTEM TIME is used '.

Points in time are stored with 23 ASCII characters with the format ,[ jjjj-mm-tt-hh-mm-ss-ttt ' ]. As internal type of variable ,SYSTEM TIME is used '.

### **1.1.6 Conventions for orientation**

The designations length and width led often to confusion, mainly because of the false representation, the length are ever larger than the width. For this reason we agreed to call the two dimensions of formats and raw boards A and B. The A-edge is always that edge, which was input as dimension A into the system.

#### **1.1.6.1 Base of orientation**

The base for orientation is always the rip cut saw. The reference edge of the rip cut saw is above. The rip cut saw of a machine is always considered as unturned and has therefore orientation zero.

#### **1.1.6.2 Orientation of the board**

In material flow direction seen, the A-edge of the board is the front edge of the board.

#### **1.1.6.3 Orientation board to rip cut saw**

If a not cut board before the rip cut saw is situated, then it has orientation zero opposite the rip cut saw. The A-edge of the board is situated with the saw.

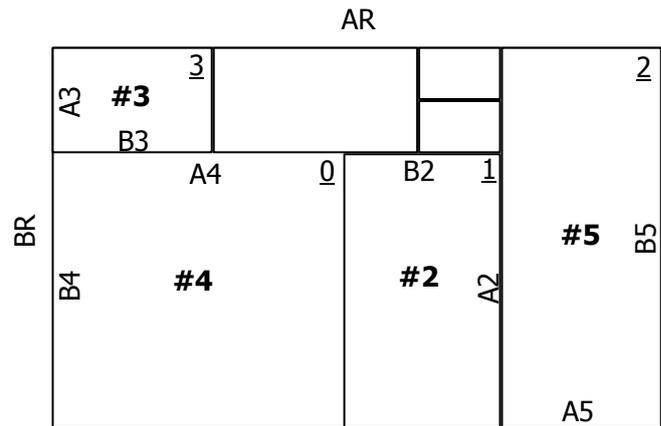
#### **1.1.6.4 Orientation of surface textures**

A structure on a format or on a raw board runs always parallel to the A-edge. Thus the dimension must be input as A-edge, which is situated parallel to the structure.

### 1.1.6.5 Orientation between format and board

The information about the orientation of two formats or boards always refers to each other to the A-edges of the two sections.

If a format is turned in relation to the raw board, then the rotation is indicated by a turning angle. The turning angle is indicated in steps by 90 degrees in the clockwise direction.



- **0=** (0 x 90 degrees = 0 degrees). The formats are situated in such a way that their A-edges are situated parallel and on the same side as the A-edge of the board (format number 4 in the example). That corresponds to the not turned status.
- **1=** (1 x 90 degrees = 90 degrees). The formats are situated in such a way that the A-edge of the format is situated parallel to the B-edge of the raw board (format number 2 in the example). That corresponds to a clockwise rotation.
- **2=** (2 x 90 degrees = 180 degrees). The A-edges of the two formats are situated again parallel, however they are not any longer on the same side as the A-edge of the board (format number 5 in the example). This case can be interesting for example if only one edge of the raw board were already processed.
- **3=** (3 x 90 degrees = 270 degrees). The formats are situated in such a way that the A-edge of the format is situated parallel to the B-edge of the raw board (format number 3 in the example). That corresponds to an anti-clockwise turn.
- **9=** unknown or freely selectable: This value is used only with the specification by set values, if no certain orientation is necessary.

### 1.1.7 Format combining

- Formats with same dimensions, edges and stacking datas can be summarized for production and therefore also for the Optimization process. Nevertheless it must be possible to manage the formats separated for example of the printout of labels and of packing tags.
- The over-delivery of a format is always assigned to the final format. The stacking software makes the decision, how the over-delivery is divided on the individual formats.
- With combined formats there can be several format references with the same schedule-referred format number. Each of these formats can belong to a different part list.

Example:

Identifier	Combined format number	Position part list	Format number part list	Quantity
F	1	2	2333	200
F	1	3	17	150

- Combined format number: If a format with others is combined, a format number for the combined format must be indicated. If the format is not combined, for the combined format number the value of the data field 'sequential position number' is entered.
- Combining the formats takes place with the creation of the Optimization order file (\*.ord). The formats of all indicated part lists for the Optimization are combined. Thus the formats are combined also for production.
- The dimensions of the formats, the board type and the board thickness of the formats must be alike.
- In the appropriate order-part-list-line combining the formats must be permitted.

Example:

Seq. NR.	Format	Komb. NR.
1	Format a	1
2	Format b	2
3	Format c	3
4	Format d	2
5	Format e	3
6	Format f	2
7	Format g	2
8	Format h	4
9	Format i	2
10	Format j	4

⇒⇒

Seq. NR.	Format	Komb. NR.
1	Format a	1
2	Format b	2
3	Format d	2
4	Format f	2
5	Format g	2
6	Format i	2
7	Format c	3
8	Format e	3
9	Format h	4
10	Format j	4

In the part lists the formats are listed according to the order of their input. According to combining they are ascending sorted according to the combined format number and the sequential numbers are accordingly modified.

At the machine the combined format number is used (format numbers in the cutting pattern). For label printouts at the machine the single format numbers must be determined by splitting up the combined format number from the schedule-referred format lines. In order to enable this splitting up easily and rapidly, all combined formats are consecutively in the NC data.

### 1.1.8 Remainders

A cutting pattern contains the formats ordered by customers. Besides a cutting pattern can contain also remainders. Remainders are actually unwanted, but sometimes unavoidable. There is sections, which were not ordered by customers, but although excessively and too unfortunate for throwing away are. The administration of the remainders takes place in two phases:

- The remainders are created either during the cutting pattern input by hand or by the computing core of the cutting pattern Optimization. At this time the remainders are treated equal as formats. They get a format number and the identifier that it concerns a remainder. They can be described therefore with the normal format data records.
- After production again raw material becomes from the remainders. They can be entered in the board data base. From this point in time on the remainders are treated equal as boards. They receive a board number and the identifier that it concerns a remainder. They can be described therefore with the normal board data records.

### 1.1.9 Filler

The fillers are input like normal formats into a part list. The fillers are planned by the Optimization only if the utilization can be improved thereby. The fillers do not have to be contained in a schedule thus necessarily.

During a follow-up optimization at the saw the same initial conditions must be as given with the original Optimization. Therefore all fillers must be contained in the NC data. Also the fillers must be contained in the NC data, which were not planned with the original Optimization.

A format becomes the filler, as its priority has a lower value than 9. The lower the priority becomes, the lower is the probability that the format is used with the Optimization.

## 1.2 Data records saw area

### 1.2.1 General information

**Data records:** Each data record (each line) is terminated with CR LF (0Dh 0Ah).

**Data fields:** Each data field has a defined size and is separated from the other data fields by a blank. A data record must contain all indicated data fields. No data fields may be omitted. Each data field therefore always starts at the same position in the ASCII string.

**Alphanumeric data fields:** These data fields must be entered left justified. Leading blanks are not permitted. If the full field width is not used and if further data fields in the data record follow, then the remainder of the data field must be filled with blanks.

**Numerical data fields:** Numerical data fields must be entered right-justified and be filled to the full field width with leading zeros.

**Attributes:** The attributes indicated in the data records are extensions intended for the variable names of variables of the respective type (e.g.. ,Any arbitrary. Attribute ').

**Variable types:** All programs are developed for 32 bit operating systems. 16 bit operating systems are not supported.

For all numerical variables the data type Integer is used. The data type Integer (int) is with 32 bit operating systems likewise 32 bits long.

For date variables SYSTEM TIME (Win32) is used.

**Line:** The designations line and data record are synonymously used. The lines in the ASCII files have a data record character in the sense of a data model. The lines normally contain no reference information, because the references are given by the position of the lines in the ASCII file.

## 1.2.2 Part list data records

The part list data records are used exclusively within the area Optimization. The part list consists in the main of the format data records stated later. Additionally there is a part-list-base-line and a part-list-end-line.

After the part-list-base-line follow the format-lines, which are combined into blocks. Within the part list the format-lines are ascending sorted according to the format number.

### 1.2.2.1 Internal construction of a part list file

The part list is divided to several files. All files have the same name, but different extensions. Example of the construction of the different part list files:

File name: BeispielStueckliste23.stk		
m_L01	Part-list-base-line (L 01)	exactly one line for each part list
m_L98	Part-list-status-line (L98)	no or exactly one line for each part list
m_L99	Part list statistics line (L99)	no or exactly one line for each part list
m_L06	<a href="#">Part-list-database-line (L 06)</a>	exactly one line for each part list
m_F01	Format-base-line (F 01)	exactly one line for each format
	To format lines see format of data records	
m_F01	Format-base-line (F 01)	exactly one line for each format
	To format lines see format of data records	
etc..		
m_L\$	Part-list-end-line (L\$)	exactly one line for each part list

File name: BeispielStueckliste23.pla
Internal construction see Internal construction of a board file

File name: BeispielStueckliste23.kan
Internal construction see Internal construction of post processing files

### 1.2.2.2 Part-list-base-line (L 01)

That is the first line of the part list. In a part list file there may be only one part-list-base-line.

⇒ [Table of contents](#)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_L01	
1	Identifier L for part lists line	1A	1		
2	Identifier 01 for base line	2N	3		
3	<a href="#">Part list name from foreign Optimization</a>	20A	6	m_NameFremd	char[ ]
4	<a href="#">Description</a>	20A	27	m_Beschreibung	char[ ]
5	<a href="#">Delivery date</a>	10N	48	m_LieferDatum	SYSTEM TIME
6	<a href="#">Creating date</a>	10N	59	m_ErstellDatum	SYSTEM TIME
7	<a href="#">System of units</a>	1N	70	m_MassSystem	int
8	<a href="#">Assembly</a>	1N	72	m_Baugruppe	int
9	<a href="#">Optimization status</a>	1N	74	m_Status	int
10	<a href="#">Original</a> ordername	20A	76	m_OriginalAuftragsnam	int
11	<a href="#">Deleting</a> flag	1N	98	m_DeleteProtected	int
	CR LF	1N	99		

3... part list name from foreign Optimization: The name of the part list corresponds to the file name. (→ conventions for names)

4... Description: Default value = empty field (spaces). This is an additional text for the part list. It is used not for the distinction by part lists and can occur therefore several times as same text. No instruction (only blank) is also possible.

5... delivery date: Default value = today's date. The delivery date is input by the user and must be same or larger as the creating date. (→ conventions for dates)

6... creating date: Default value = today's date. That is the date, on which the part list was for the first time created. (→ conventions for dates)

7... system of units: Default value = 0. (→ conventions for dimensions)

8... assembly: Default value = 0. 0= part list. 1= assembly. The record layout of the part lists and assemblies is the same. The only difference is that assemblies are not optimized directly. Assemblies must be read in into a part list. Only the part list can be optimized then. Several times used part lists can be archived as assemblies .

9... Optimization status: Default value = 0. This data field indicates whether the part list was already processed by the Optimization. Note: this data field is replaced by the suitable data field in Part-list-status-line (L98) ).

- 0= the part list was not optimized yet.
- 1= the part list was optimized.
- 2= the part list partially optimized.

10... Original ordername: Default value = empty field (spaces). Only important for re-imported schedules.

11... Deleting flag: Default value = 0. This field indicates whether the partlist may be deleted in case of automatic deleting rules. 0=yes, 1=no

### 1.2.2.3 Part-list-database-line (L 06) version 3.0

This line serves for the specification of the selection options when using data bases. There are data bases for boards, edges, groove programs and window programs. The part-list-database-line is optional. If the line is missing, then the default values are assumed.

⇒ [Table of contents](#)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_L06	
1	Identifier L for part lists line	1A	1		
2	Identifier 06 for data base line	2N	3		
3	<a href="#">Additional text</a>	20A	6	m_Zusatztext	char[ ]
4	<a href="#">Board data base use</a>	1N	27	m_PDBverwenden	int
5	<a href="#">Edge data base use</a>	1N	29	m_KDBverwenden	int
6	<a href="#">Groove data base use</a>	1N	31	m_NDBverwenden	int
7	<a href="#">Window data base use</a>	1N	33	m_FDBverwenden	int
8	<a href="#">Packing rule data base use</a>	1N	35	m_VpVDBverwenden	int
9	<a href="#">Stack cover board data base use</a>	1N	37	m_StpSpDBverwenden	int
10	<a href="#">Strapping data base use</a>	1N	39	m_UmreifungDBverwenden	int
11	<a href="#">Strapping-rules data base use</a>	1N	41	m_UmreifungsRegelnDBverwenden	int
12	<a href="#">Default value stacking parameter use</a>	1N	43	m_StandardwertStpParVerwenden	int
13	<a href="#">Stacking parameter set</a>	20A	45	m_Stapelparametersatz	char[ ]
	CR LF		65		

3... additional text: The additional text is not used at present yet.

4... board data base use: Default value = 1. This adjustment can be used, if any remainders are to be planned mandatory, although a board data base is available.

- 0= the board data base may be not used. Only boards from the part list-referred board list may be used.
- 1= the boards from the board data base are to be used with the highest priority (9).

5... edge data base use: Similar to data field 4.

- 0= the part list-referred edge files must be used with priority.
- 1= the edge data base must be used with priority. With priority the edges from the edge data base apply.

6... Groove data base use: Similar to data field 4.

7... window data base use: Similar to data field 4.

8... packing rule data base use: Packing rule data base, similar to data field 4.

9... stack cover board data base use: Stacking cover board data base, similar to data field 4.

10... Strapping data base use: Strapping data base, similar to data field 4.

11... Strapping-rules data base use: Strapping-rules data base, similar to data field 4.

12... Default value stacking parameter use: (default value = 1)

This switch specifies whether the default values of the stacking parameters may be used to determine the stacking data. The default values of the stacking parameters are only used if no other details were given.

13... Stacking parameter set: (Default value = empty field (spaces))

In this data field the name of the stacking parameter set for the parts list is saved. The data field is used to minimize the input effort in the format data. The parameter set specified here is entered by the HPO NC data creation in the corresponding format data fields, if they are still empty.

### 1.2.2.4 Part-list-status-line (L98)

This line contains different information about the status of the part list. It is used only in part list files. The line is optional. If it is missing, then for each status of the part list the default value is assumed. If a value deviating from the default value is to be stored, then the line must be inserted.

Note! Since the part lists have a n-n-relationship with the schedules, the status „Online cannot be set unique. Only if the part list were transferred directly into only one order, a unique allocation of a part list to certain schedules is possible.

Bo	Description	L/A	Pos	Attributes	Data type
X					
-	Class name			m_L98	
1	Identifier L for part lists line	1A	1		
2	Identifier 98 for status line	2N	3		
3	Status input	1N	6	m_StatusEingabe	int
4	Status Optimization	1N	8	m_StatusOptimierung	int
5	Status Online	1N	10	m_StatusOnline	int
6	Status production	1N	12	m_StatusProduktion	int
7	Status raw boards	1N	14	m_StatusRohplatten	int
8	Status remainder boards	1N	16	m_StatusRestplatten	int
	CR LF		17		

3... Status input: In this data field the status for the data input is stored. Default value = 0. This status must be changed manually.

0= no status data (default value).

1= not approved.

2= approved.

3= approved.

4... Status Optimization: In this data field the status for the Optimization is stored. Default value = 0. This status is set by the Optimization (is not implemented yet).

0= no status data (default value).

1= not optimized. No format from the part list was optimized.

2= partially optimized. At least a format from the part list was optimized. Not all formats from the part list were optimized.

3= completely optimized. All formats in the part list were optimized.

5... Status Online: In this data field the status of the machine transfer is stored. Default value = 0. This status results from the status of the schedules, which are contained in the part list (is not implemented yet).

0= no status data (default value).

1= not transmitted. None of the schedules from the part list was transferred to the machine.

2= partially transmitted. At least one of the schedules from the part list was transferred to the machine. Not all schedules from the part list were transferred.

3= completely transmitted. All schedules in the part list were transferred to the machine.

6... Status production: In this data field the status of production is stored. Default value = 0. This status results from the status of the schedules, which are contained in the part list (is not implemented yet).

0= no status data (default value).

1= not produced. No schedule in the part list was produced.

2= partially produced. At least one of the schedules in the part list was produced. Not all schedules in the part list were completely produced.

3= completely produced. All schedules in the part list were completely produced.

9= in production. One of the schedules in the part list is momentary in production.

7... Status raw boards: In this data field the status for the booking of the raw boards is stored. Default value = 0. This status results from the status of the schedules, which are contained in the part list (is not implemented yet).

0= no status data (default value).

1= not booked. For no schedule in the part list raw boards are booked out.

2= partially booked. At least for a schedule in the part list raw boards are booked out. For all schedules in the part list raw boards are not booked out.

3= completely booked. For all schedules in the part list raw boards are booked out.

4= indefinite. It cannot be determined whether the raw boards booked out are. This status can occur, if the board data base is not available.

8... Status remainder boards: In this data field the status for the booking of the remainder boards is stored. Default value = 0. This status results from the status of the schedules, which are contained in the part list (is not implemented yet).

0= no status data (default value).

1= not booked. For no schedule in the part list remainder boards are booked in.

2= partially booked. At least for a schedule in the part list remainder boards are booked in. For all schedules in the part list remainder boards are not booked in.

3= completely booked. For all schedules in the part list remainder boards are booked in.

4= indefinite. It cannot be determined whether the remainder boards booked in are. This status can occur, if the board data base is not available.

### 1.2.2.5 Part list statistics line (L99)

This line is reserved for part list-referred, calculated statistical data.

### 1.2.2.6 Part-list-end-line (L\$)

This line serves for recognition of a correct end of file. It is used only in part list files. (→ directory and file structure)

Box	Description	L/A	PoS	Attributes	Data type
1	Identifier L\$ for part-list-end-line CR LF	Å	1 3		

## 1.2.3 Format data records

The format data records are used in part lists and in the NC data. Not each area uses all data fields. Nevertheless the same data records are always used.

All format data records of a certain format are combined into a block. Within this block the format-lines are ascending sorted according to their data record identifier. Thus for each format a block of data records is formed, which is connected internally only by the fact that the data lines are consecutively in an ASCII file.

### 1.2.3.1 Internal construction of a format block

Example of a block of format lines with minimum contents:

m_F01	<a href="#">Format-base-line (F 01)</a>	exactly one line for each format
-------	---	----------------------------------

Example of a block of format lines with maximum contents:

m_F01	<a href="#">Format-base-line (F 01)</a>	exactly one line for each format, at least one line for each part list
m_F02	<a href="#">Format-base-line (F 01)</a>	0 or 1 line for each format
m_F10	<a href="#">Format-customer-text-line (F 10-29)</a>	0 or 1 line for each format
to		
m_F29	<a href="#">Format-customer-text-line (F 10-29)</a>	0 or 1 line for each format
m_F05	<a href="#">Format-packing-tag-base-line (F 05)</a>	0 or 1 line for each format
m_F30	<a href="#">Format-packing-tag-text-line (F 30-49)</a> 1	0 or 1 line for each format
to		
m_F49	<a href="#">Format-packing-tag-text-line (F 30-49)</a> 20	0 or 1 line for each format
m_F06	<a href="#">Format-label-base-line (F 06)</a>	0 or 1 line for each format
m_F60	<a href="#">Format-label-text-line (F 60-79) 1</a>	0 or 1 line for each format
to		
m_F79	<a href="#">Format-label-text-line (F 60-79) 20</a>	0 or 1 line for each format
m_F07	<a href="#">Format-top-surface-line (F 07)</a>	0 or 1 line for each format
m_F08	<a href="#">Format-bottom-surface-line (F 08)</a>	0 or 1 line for each format
m_F80	<a href="#">Format-post-processing-line (F 80)</a>	0 or 1 line for each format
m_F90	<a href="#">Format handling line (F 90)</a>	0 or 1 line for each format

### 1.2.3.2 Format-base-line (F 01)

This data record must be available for each format in an order, in a part list or in a schedule.

⇒ [Table of contents](#)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F01	
1	Identifier F for format line	1A	1		
2	Identifier 01 for format-base-line	2N	3		
3	<a href="#">Sequential number</a>	4N	6	m_LfdNr	int
4	<a href="#">Combined number</a>	4N	11	m_KombiNr	int
5	<a href="#">Category</a>	1N	16	m_Kategorie	int
6	<a href="#">Quantity of formats order</a>	6Ñ	18	m_AnzBestellt	int
7	<a href="#">Quantity of formats optimize</a>	6Ñ	25	m_AnzOptimiert	int
8	<a href="#">Board type</a>	20A	32	m_PlattenTyp	char[ ]
9	<a href="#">Board thickness</a>	8N	53	m_PlattenDicke	int
10	<a href="#">Cutting dimension A</a>	8N	62	m_MassA	int
11	<a href="#">Cutting dimension B</a>	8N	71	m_MassB	int
12	<a href="#">Finished dimension A</a>	8N	80	m_FertigmassA	int
13	<a href="#">Finished dimension B</a>	8N	89	m_FertigmassB	int
14	<a href="#">Description</a>	20A	98	m_Beschreibung	char[ ]
15	<a href="#">Format is rotatable</a>	1N	119	m_Drehbar	int
16	<a href="#">Priority of the format</a>	1N	121	m_Prioritaet	int
17	<a href="#">Quality of the format</a>	1N	123	m_Qualitaet	int
18	<a href="#">Format only turned use</a>	1N	125	m_NurGedrehtVerpl	int
19	<a href="#">Identifier group of formats</a>	2N	127	m_GruppeOpt	int
20	<a href="#">Over-delivery in piece</a>	6Ñ	130	m_UeberliefStk	int
21	<a href="#">Over-delivery in per cent</a>	3N	137	m_UeberliefProz	int
22	<a href="#">Under-delivery in piece</a>	6Ñ	141	m_UnterliefStk	int
23	<a href="#">Under-delivery in per cent</a>	3N	148	m_UnterliefProz	int
24	<a href="#">Format optimize</a>	1N	152	m_Optimieren	int
25	<a href="#">Format is sub-part</a>	1N	154	m_Kleinteil	int
26	<a href="#">Position number from external program</a>	20A	156	m_PosNrExtern	char[ ]
27	<a href="#">Line number from external program</a>	20A	177	m_ZeilenNrExtern	char[ ]
28	<a href="#">Processing note</a>	40A	198	m_BearbHinwExtern	char[ ]
29	<a href="#">Label print</a>	1N	239	m_EtikettDrucken	int
30	<a href="#">Packing tags print</a>	1N	241	m_PackzettelDrucken	int
31	<a href="#">Source assembly</a>	20A	243	m_QuellBaugruppe	char[ ]
32	<a href="#">Source part list</a>	20A	264	m_QuellStkl	char[ ]
33	<a href="#">Format number part list</a>	9N	285	m_NrInStkl	int
34	<a href="#">Delivery date</a>	10N	295	m_Lieferdatum	SYSTEM TIME
35	<a href="#">Orientation in the longitudinal strip</a>	1N	306	m_OrientierungLaengs	int
36	<a href="#">Status</a>	1N	308	m_Status	int
37	<a href="#">Internal use only</a>	<a href="#">3N</a>	310	m_AnzBestelltDNKS	int
38	<a href="#">Dimension multiplier A</a>	<a href="#">2N</a>	314	m_MassMultiplikatorA	int
39	<a href="#">Dimension multiplier B</a>	<a href="#">2N</a>	317	m_MassMultiplikatorB	int
40	<a href="#">Quantity adjustment</a>	<a href="#">7N</a>	320	m_AnzKorrektur	signed int
41	<a href="#">Part allowed in 3rd phase group</a>	<a href="#">1N</a>	328	m_DSErlaubt	int
42	<a href="#">Max. waste dimension for 3RD phase cut</a>	<a href="#">8N</a>	330	m_MaxAbfallMassImDS	int
43	<a href="#">Front pattern number</a>	<a href="#">4N</a>	339	m_PlanNrGT	int
44	<a href="#">Primary key parts list admin</a>	8N	344	m_m_PrImKeyDB	int
45	<a href="#">Edge combining</a>	1N	353	m_KanteKombinierbar	int

46	<a href="#">Nc5 board stock Id (GUID)</a>	38A	355	m_guidNc5StatusId	CSDGuid
47	<a href="#">Nc5 board storage usage Id (GUID)</a>	38A	394	m_guidNc5LagerVerwendungsId	CSDGuid
48	<a href="#">Nc5 board stock Id BoB Rest(GUID)</a>	38A	433	m_guidNc5LagerBestandsIdBoB	CSDGuid
49	<a href="#">Revolve part</a>	1N	472	m_Wenden	int
50	<a href="#">CR LF</a>				
45	<a href="#">Edge combining</a>	1N	353	m_KanteKombinierbar	int
46	<a href="#">Revolve part</a>	1N	355	m_Wenden	int
47	<a href="#">CR LF</a>		356		

3... sequential number: Sequential number within part list, order or schedule. It always starts with 1 and is continuous ascending numbered.

The sequential number may be used within the area Optimization only for part list-internal or order-internal references. The sequential position number can change, if formats are reset or added. For superordinate references the part list-referred format number must be used.

Within the area NC data can be used it also for spreading references, since no more entries can be added here or reset.

... format number combined 4: Formats with the same dimensions can be optimized combined. Thus they can be produced also combined. Nevertheless it must be possible to manage the formats separated.

(→ format combining)

Each format receives a combined format number beside the sequential number. The combined format number is assigned ascending and starts with 1.

The combined format number is only determined if an Optimization order is created. In a part list not optimized yet it is always zero.

After the order creation both format numbers are always carried. In the cutting patterns only the combined format number is used.

5... category: Default value =0. this data field assigns the format to a certain category. The category is indicated by a number from 0 to 9. At present the following categories are defined:

- 0= the format is a normal format.
- 1= the format is a remainder (→ remainders)
- 2= the format is a longitudinal strip.
- 3= the format is a intermediate format

6... quantity of formats orders: The order quantity is the desired quantity for this format. The quantity of the ordered formats is taken over from the part list.

7... quantity of formats optimizes: The optimized quantity of the formats is the quantity of the formats planned by the Optimization. Over-delivery, under-delivery and board stock are considered.

8... board type: The board type as well as the board thickness serves for the distinction of the different boards. (→ conventions for dimensions)

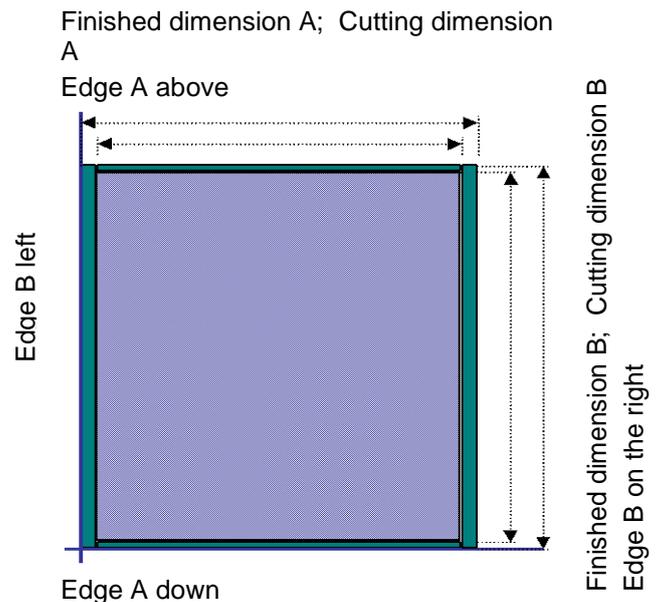
9... board thickness: The board thickness serves together with the board type for the distinction of the different boards. (→ conventions for dimensions)

10.11... cutting dimension A / B: Those are the format dimensions valid for the first phase of operation. In further phases of operation (format cutting patterns, handling of the edges etc..) the format can receive different dimensions.

(→ conventions for dimensions)

12.13... finished dimension A / B: Those are the final format dimensions. They apply to the final phase of operation. At the preceding manufacturing operations they serve only for the information of the user.

(→ conventions for dimensions)



14... description: That is additional information for the user.

(→ conventions for names)

15... format is rotatable: Default value =2. Dependent on the board type (surface texture) or other criteria it can be determined whether the format may occur turned in the cutting pattern.

0= format is not rotatable

1= format is rotatable

2= the format is not rotatable, if the used board has a structure. The format is rotatable, if the used board does not have structure. The specification of the structure of the board follows the conventions for orientation. In the case of orientation 0 is the structure parallel to the A-edge of the board.

16... priority of the format: Default value =9. The priority determines whether the format is an obligatory position or an optional position.

9= obligatory position: The format must be planned with the given quantity.

0 to 8= optional position (filler): The format does not have to be planned necessarily. The given quantity is the upper boundary. The Optimization can plan also a lower quantity. The lower the value of the priority is, the fewer the formats is taken into account.

17... quality of the format: Default value =9. Formats with the quality indicated here can be produced only from boards with same or higher quality. The quality has a range from 0 to 9.

18... format only turned use: Default value =0. This data field determines whether the format may be taken into account only turned. This data field has priority over the data field 'format is rotatable', if there are contradictions. It is important only for the transfer from CAD programs, which have a rigid allocation of length and width.

0= the format may be taken into account without limitation.

1= the format may be taken into account only turned.

19... identifier group of formats: Default value =0. That is an identifier for a certain group of formats (e.g. technology-orient). The permitted values are 0 to 99. The identifier can be used to optimize formats group-oriented. This possibility is not used however so far.

20.21.22.23... over-delivery in piece or per cent / under-delivery in piece or per cent: This data field indicates, by how much formats the planned quantity of formats may deviate from the desired order quantity. If both absolute and proportional boundaries are indicated, then the closer of the two boundaries applies. If only one of the two boundaries has the value zero, then applies only that boundary, which does not have the value zero.

24... format optimizes: Default value =1. This value determines whether the format is to be optimized. If a format is entered in a part list only because of the completeness, it can be excluded from the Optimization.

0 = the format is not optimized.

1 = the format must be optimized.

25... Format is sub-part: Default value =0. Sub-parts may not be considered by the Optimization (→machine functions, format cutting patterns)

0= the format is not sub-part.

1= the format is a sub-part.

26.27... position number from external program / line number from external program: With the receiving of data from external programs (carpenter programs, CAD programs) it must be possible to be able to refer to the original information from these programs.

28... processing note: With the receiving of data from external programs (carpenter programs, CAD programs) additional information can be transferred in this data field.

29... label print: Default value = 1.

0= no printout of labels for this format.

1= printout of labels in accordance with Format-label-base-line (F 06). If this line is missing, the standard label of the machine control system is used.

30... packing tags print: Default value = 0.

0= no printout of packing tags for this format.

1= printout of packing tags in accordance with Format-packing-tag-base-line (F 05). If this line is missing, the standard packing tag of the machine control system is used.

31... Source assembly: This data field contains the name of the assembly, if the format originates from an assembly. Thus can be determined, from which assembly this format originates.

32... source part list: This data field contains the name of the part list, from which the format originates. Together with the internal format number a unique back tracing is possible.

33... Format number part list: Each format receives a unique number, which may not be changed also with modification of the part list during entry. The number has a range from 1 to 999999999 and is rotary assigned. Thus it is guaranteed that the same number is not twice used within the part list. If the format is a remainder, the number zero is entered here.

34... delivery date: Additionally to the delivery date of the order a format specific delivery date can be indicated here. (→ conventions for dates)

35... orientation in the longitudinal strip: Default value =0. In this data field the desired orientation of the format can be indicated in its longitudinal strip. The strip orientation is the direction of the rip cut.

0= the orientation of the format can be selected freely.

1= the A-edge of the format must be situated lengthwise to the strip orientation.

2= the B-edge of the format must be situated transverse to the strip orientation.

36... status: Default value =0. In this data field it is indicated from where the format comes.

0= the format was created by the Optimization.

1= the format was input manually.

37... Digits after decimal point, Quantity of formats order:

The order quantity is the desired quantity for this format. The digits after decimal point (in 1/1000) is indicating, how much must be subtracted from the Qty ordered to reach the fractional formats in the component editor. Example: In the component editor the “Qty ordered” is displayed with 3,125. In this case the “Qty ordered” is 4 and the “Qty ordered DKS” is 875. 4 minus 0,875 give us the displayed value of 3,125.

Reason of the „digits after decimal point“:

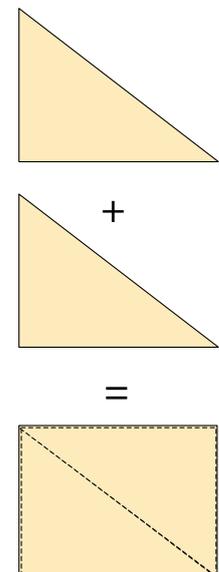
For some parts not the whole format dimension is needed, but only the half. If 2 of these parts are needed, (only) 1 complete part has to be ordered.

If for example a triangle is needed, in real this is the half of a rectangular part. If 2 of these triangles are needed, 1 (complete) parts has to be ordered, if 3 of these triangles are needed, 1,5 parts have to be ordered...

Due to the fact, that only whole parts can be produced, inside of the program must be calculated with “full numbers”.

In parts lists and components it should be possible to enter quantities with decimal (e.g. 0,5, 0,33 or 0,25).

As soon as the order will be created for the calculation kernel, all quantities will be rounded up.



38... Dimension multiplier A: Default value =0. This value determines how often the part was multiplied in the length dimension to fit into a cut-able pattern.

39... Dimension multiplier B: Default value =0. This value determines how often the part was multiplied in the width dimension to fit into a cut-able pattern.

40... Quantity adjustment: Default value =0. This value will be added to the ordered quantity of parts for the optimization. The value is signed, that means also negative values are possible (i.e. -0000005).

41... Part allowed in 3<sup>rd</sup> phase group: Default value =1. This value will allow a part to be placed into a 3<sup>rd</sup> phase group (z-cut). Possible values: 0=no, 1=yes, 2=yes, but only in dimension A direction, 2=yes, but only in dimension B direction

42...Max. waste dimension for 3<sup>rd</sup> phase cut: Default value =0. This value gives the max. allowed waste dimension for the part in the 3<sup>rd</sup> phase cut.

43...Front pattern number: Default value =0. This is the pattern number of the front group for this part.

44... Primary key parts list admin: Default value = 0. This data field contains the primary key for the parts list administration.

45... Edge combining: Default value = 0. Combining of parts with identical edges is possible. 0=yes, 1=only on rip edge, 2=only on cross edge, 3=no

46... Nc5 board stock ID: Standardwert: {00000000-0000-0000-0000-000000000000}  
This data field contains the GUID of the Nc5 board data base. With help of this ID, all data to this record set can be selected out of the data base.

47... Nc5 board storage usage ID: Default value = {00000000-0000-0000-0000-000000000000}  
This data field contains the GUID of the Nc5 board data base. With help of this ID, all data to this record set can be selected out of the data base.

48... Nc5 board stock ID BoB: Default value = {00000000-0000-0000-0000-000000000000}  
This data field contains the GUID of the Nc5 board data base. This is the remainder board, which was created from BoB.

49... Revolve part: Default value =0. This data field indicates, if the part has to be revolved or not.

- 0= part must not be revolved
- 1= part must be revolved

### 1.2.3.3 Format-stacking-line (F 02)

See data records stacking area

### 1.2.3.4 Format-customer-text-line (F 10-29)

These lines have the numbers 10-29 and are optional. In these 20 lines any customized information can be entered to the formats (e.g. article number).

Box	Description	L/A	Pos	Attributes	Data type
-	Class name		S	m_F10 to m_F29	
1	Identifier F for format line	1A	1		
2	Identifier 10-29 for format customers line	2N	3		
3	Customized text CR LF	110A	6 86	m_Text	char[ ]

### 1.2.3.5 Format-packing-tag-base-line (F 05)

This line contains packing tag information for each format. It is optional. If this line is available, it must follow after the base line of the respective format. It may exist only one format-packing-tag-base-line per processing step and format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F05	
1	Identifier F for format line	1A	1		
2	Identifier 05 for packing tag base line	2N	3		
3	<a href="#">Processing step</a>	1N	6	m_BearbeitungNr	int
4	<a href="#">Packing tag layout</a>	20A	8	m_Layout	char[ ]
5	<a href="#">Quantity of packing tags per unit</a>	2N	29	m_AnzProEinheit	Int
6	<a href="#">Quantity of units</a>	3N	32	m_AnzEinheiten	Int
7	<a href="#">Unit</a>	2N	36	m_Einheit	Int
8	<a href="#">Grafic file 1</a>	20N	39	m_Grafik1	char[ ]
9	<a href="#">Grafic file 2</a>	20N	60	m_Grafik2	char[ ]
10	<a href="#">Grafic file 3</a>	20N	81	m_Grafik3	char[ ]
11	<a href="#">Grafic file 4</a>	20N	102	m_Grafik4	char[ ]
12	<a href="#">Printer number</a> CR LF	1N	123 124	m_DruckerNr	int

3... processing step: Default value =1. This data field contains the number the processing step, to which the packing tag line is to apply. The processing step 1 is always the raw cut and also the default value for this data field. Step 2 is the intermediate processing. Step 3 is the final cut to size on the final dimension.

(→ Format-post-processing-line (F 80))

4... packing tags layout: This data field contains the name the layout, which is to be used for the packing tag. It concerns thereby a file name. If no layout name is indicated, then a default layout is used. (→ conventions for names)

5... quantity of packing tags per unit: This data field contains the quantity of the packing tags, which are to be printed per quantity of the units. If nothing is indicated, then the quantity of 1 is assumed. Examples: 2 packing tags per 100 format; 1 packing tag per stack.....

6... quantity of units: This data field contains the quantity of the units, for which the quantity of the packing tags is to be printed. If nothing is indicated, then the quantity of 1 is assumed.

7... Unit: Default value =1. The quantity of the packing tags per unit refers to the unit specified here.

1 = Per part/first part

2 = Per book

3 = Per stack

4 = Per cutting pattern

5 = Per schedule/first part

6 = Per parts in the stack

7 = Per partbook (not supported with packing tags)

8 = Per part/last part

9 = Per schedule/last part

10= Per part/first and last part

11 = Per schedule/first and last part

12 = Per stack / Quantity Row x Column

8,9,10,11... Grafic file n: Those are four names of grafic files. The grafics are printed out on the packing tags for this format. The selected grafic files must be transferred with the NC data to the machine. The file format for the grafic files must be bit-map (\*.bmp). A support for dxf and dwg is in preparation. If several grafic formats are supported, extension is not considered. If several files with the same name, but different extension are available, a file is selected automatically. The sequence of the selection is \*.bmp, \*.dxf, \*.dwg.

(→ directory structure and file structure, → conventions for names)

12... printer number: Default value =0. Here the number of the printer can be indicated, on which the packing tag is to be printed out. The allocation of the printer number to a system printer takes place in the label print program. Printer number 0 is always the system standard printer.

### 1.2.3.6 Format-packing-tag-text-line (F 30-49)

These lines contain the texts for the design of the packing tags. The texts can be printed out on the packing tags for this format. The lines are optional. If such a line is available, it must follow after the format-base-line of the respective format. Maximum 20 such lines for each format can be available.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F30 to m_F59	
1	Identifier F for format line	1A	1		
2	Identifier 30-49 for packing tag text line	2N	3		
3	Customized text	110A	6	m_Text	char[ ]
	CR LF		86		

With the parameter 001 **001 004 001 057 Use Customer text line as graphic path** these customer text lines can be used for the graphic paths.

#### PLEASE NOTE:

- Although these are packing tag texts, the paths are used for the labels!
- Only lines F30-F33 are used. All other lines can be used arbitrarily!

### 1.2.3.7 Format-label-base-line (F 06)

This line contains label information for each format. It is optional. If this line is available, it must follow after the base line of the respective format. It may be available only one format-label-base-line per processing step and format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F06	
1	Identifier F for format line	1A	1		
2	Identifier 06 for labels base line	2N	3		
3	Processing step	1N	6	m_BearbeitungNr	int
4	Labels layout	20A	8	m_Layout	char[ ]
5	Quantity of labels per unit	2N	29	m_AnzProEinheit	int
6	Quantity of units	3N	32	m_AnzEinheiten	int
7	Unit	2N	36	m_Einheit	int
8	Grafik1	20N	39	m_Grafik1	char[ ]
9	Grafik2	20N	60	m_Grafik2	char[ ]
10	Grafik3	20N	81	m_Grafik3	char[ ]
11	Grafik4	20N	102	m_Grafik4	char[ ]
12	Printer number	1N	123	m_DruckerNr	int
	CR LF		124		

[⇒ Table of contents](#)

3... Processing step: Default value =1. This data field contains the number the processing step, to which the label line is to apply. The processing step 1 is always the raw cut and also the default value for this data field. Step 2 is the intermediate processing. Step 3 is the final cut to size on the final dimension.

(→ Format-post-processing-line (F 80))

4... Labels layout: This data field contains the name the layout, which is to be used for the label. It concerns thereby a file name. If no layout name is indicated, then a default layout is used.

(→ conventions for names)

5... Quantity of labels per unit: This data field contains the quantity of the labels, which are to be printed per quantity of the units. If nothing is indicated, then the quantity of 1 is assumed. Examples: 2 labels per 100 format; 1 labels per stack.....

6... Quantity of units: This data field contains the quantity of the units, for which the quantity of the labels is to be printed. If nothing is indicated, then the quantity of 1 is assumed.

7... Unit: Default value =1. The quantity of the labels per unit refers to the unit specified here.

1 = Per part/first part

2 = Per book

3 = Per stack

4 = Per cutting pattern

5 = Per schedule/first part

6 = Per parts in the stack

7 = Per partbook (not supported with packing tags)

8 = Per part/last part

9 = Per schedule/last part

10= Per part/first and last part

11 = Per schedule/first and last part

12 = Per stack / Quantity Row x Column

8,9,10,11... Grafic n: Those are four names of grafic files. The grafics are printed out on the labels for this format. The selected grafic files must be transferred also with the NC data to the machine. The file format for the grafic files must be bit-map (\*.bmp). A support for dxf and dwg is in preparation. If several grafic formats are supported, extension is not considered. If several files with the same name, but different extension are available, a file is selected automatically. The sequence of the selection is \*.bmp, \*.dxf, \*.dwg.

(→ directory structure and file structure, → conventions for names)

12... Printer number: Default value =0. Here the number of the printer can be indicated, on which the label is to be printed out. The allocation of the printer number to a system printer takes place in the label print program. Printer number zero is always the system standard printer.

### 1.2.3.8 Format-label-text-line (F 60-79)

These lines contain the texts for the design of the labels. The texts can be printed out on the labels for this format. The lines are optional. If such a line is available, it must follow after the base line of the respective format. Maximum 20 such lines for each format can be available.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F60 to m_F79	
1	Identifier F for format line	1A	1		
2	Identifier 60-79 for labels text line	1A	3		
3	Customized text	110A	6	m_Text	char[ ]
	CR LF		86		

With the parameter **001 001 004 001 057 Use Customer text line as graphic path** these customer text lines can be used for the graphic paths. These paths are used for the PACKING TAG.

**PLEASE NOTE:**

- Although these are label texts, the paths are used for the packing tags!
- Only lines F60-F63 are used. All other lines can be used arbitrarily!

### 1.2.3.9 Format-top-surface-line (F 07)

This line contains information about the upper surface for each format. It is optional. If this line is available, it must follow after the format-base-line of the respective format. It may be available only one format-top-surface-line per format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F07	
1	Identifier F for format line	1A	1		
2	Identifier 07 for surface above line	2N	3		
3	Material	20A	6	m_Material	char[ ]
4	Thickness	8N	27	m_Dicke	int
5	Processing note	20A	36	m_Bearbeitung	char[ ]
6	Addition A	8N	57	m_ZuschlagA	int
7	Addition B	8N	66	m_ZuschlagB	int
8	Optimize	1N	75	m_Optimieren	int
9	Trim	1N	77	m_Besaeumen	int
	CR LF		78		

3... Material: This data field contains the name the surface material. The name has an only informative character. (→ conventions for names)

4... Thickness: This data field contains the thickness the surface material. The thickness has an only informative character. (→ conventions for dimensions)

5... Processing note: This data field contains a processing note for the surface the format. The note has an only informative character.

6,7... Addition A / B: The material for the surface can be larger than the dimensions of the format. This value surrounds on, like much the surface material is larger than the actual format.

(→ conventions for dimensions)

8... Optimize: Default value =0. This data field indicates whether the material for the surfaces is to be considered by the Optimization or not.

0= the surface is not optimized.

1= the surface is optimized.

9... Trim: Default value =0. This data field indicates whether the surface is to be trimmed. If the finished dimension of the format is equal as its cutting dimension and no edges are available, no more further handling takes place. With this data field an additional trim can be forced, if the surface indicates a projection.

0 = no additional trim of the surface.

1 = the surface must be trimmed in any case. All four edges are trimmed.

### 1.2.3.10 Format-bottom-surface-line (F 08)

This line contains information about the lower surface for each format. It is optional. If this line is available, it must follow after the format-base-line of the respective format. It may be available only one surface down line for each format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F08	
1	Identifier F for format line	1A	1		
2	Identifier 08 for surface down line	2N	3		
3	Material	20A	6	m_Material	char[ ]
4	Thickness	8N	27	m_Dicke	int
5	Processing note	20A	36	m_Bearbeitung	char[ ]
6	Addition A	8N	57	m_ZuschlagA	int
7	Addition B	8N	66	m_ZuschlagB	int
8	Optimize	1N	75	m_Optimieren	int
9	Trim	1N	77	m_Besaeumen	int
	CR LF		78		

3... Material: This data field contains the name the surface material. The name has an only informative character. (→ conventions for names)

4... Thickness: This data field contains the thickness the surface material. The thickness has an only informative character. (→ conventions for dimensions)

5... Processing note: This data field contains a processing note for the surface the format. The note has an only informative character.

6,7... Addition A / B: The material for the surface can be larger than the dimensions of the format. This value surrounds on, like much the surface material is larger than the actual format.

(→ conventions for dimensions)

8... Optimize: Default value =0. This data field indicates whether the material for the surfaces is to be considered by the Optimization or not.

0= the surface is not optimized.

1= the surface is optimized.

9... Trim: Default value =0. This data field indicates whether the surface is to be trimmed. If the finished dimension of the format is equal as its cutting dimension and no edges are available, no more further handling takes place. With this data field an additional trim can be forced, if the surface indicates a projection.

0 = no additional trim of the surface.

1 = the surface must be trimmed in any case. All four edges are trimmed.

### 1.2.3.11 Format-post-processing-line (F 80)

This line contains information about the post processing for a format. It is optional. If this line is available, it must follow after the base line of the respective format. It may be available only one format-post-processing-line per format.

⇒ [Table of contents](#)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F80	
1	Identifier F for format line	1A	1		
2	Identifier 80 for post processing line	2N	3		
3	<a href="#">Minus tolerance cutting dimension A</a>	8N	6	m_MinusTolA	int
4	<a href="#">Plus tolerance cutting dimension A</a>	8N	15	m_PlusTolA	int
5	<a href="#">Minus tolerance cutting dimension B</a>	8N	24	m_MinusTolB	int
6	<a href="#">Plus tolerance cutting dimension B</a>	8N	33	m_PlusTolB	int
7	<a href="#">Format cutting pattern</a>	4N	42	m_FormatPlan	int
8	<a href="#">Edge identifier A down</a> (lengthwise in front)	20A	47	m_KantCodeLV	char[ ]
9	<a href="#">Edge identifier A above</a> (lengthwise rear)	20A	68	m_KantCodeLH	char[ ]
10	<a href="#">Edge identifier B left</a> (crosswise rear)	20A	89	m_KantCodeQH	char[ ]
11	<a href="#">Edge identifier B on the right</a> (crosswise in front)	20A	110	m_KantCodeQV	char[ ]
12	<a href="#">Corner logo on the right down</a> (on the right in front)	2N	131	m_EckLogoRV	int
13	<a href="#">Corner logo left down</a> (left in front)	2N	134	m_EckLogoLV	int
14	<a href="#">Corner logo left above</a> (left rear)	2N	137	m_EckLogoLH	int
15	<a href="#">Corner logo on the top right</a> (on the right rear)	2N	140	m_EckLogoRH	int
16	<a href="#">Intermediate dimension A</a>	8N	143	m_ZwischenmassA	int
17	<a href="#">Intermediate dimension B</a>	8N	152	m_ZwischenmassB	int
18	<a href="#">Groove program</a>	20A	161	m_NutProg	char [ ]
19	<a href="#">Window program</a>	20A	182	m_FensterProg	char [ ]
	CR LF		202		

3,4,5,6... Plus/minus tolerance cutting dimension A / B: Default value = 0. In these data fields the tolerance for the respective cutting dimension can be input. The actual dimensions A and B of the format may deviate by the tolerance indicated here.

(→ conventions for dimensions)

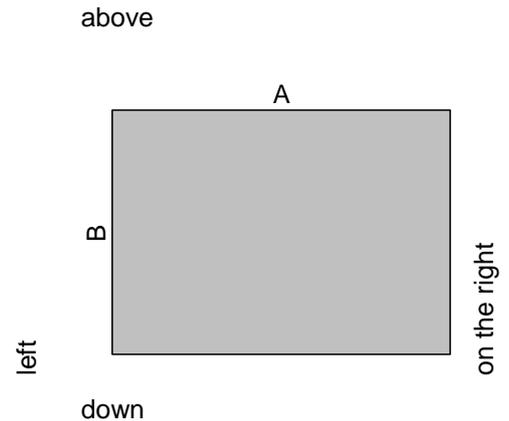
7... format cutting pattern: Default value = 0. 0= no valid format cutting pattern. A format cutting pattern is a ' cutting pattern in the cutting pattern '. It can be used to divide a format in the next processing step into further formats. A format cutting pattern is structured like a normal cutting pattern and has file extension ,ncf '. The number indicated here refers to a file with the ending ,ncf ' in the current schedule directory.

(→ directory structure and file structure)

8.9.10.11... edge identifier A down, A above, B left, B on the right:

Default value = 20 blanks. That is a reference on a schedule-referred data record with an edge definition. For each edge identifier, which is used in a format, there must be a data record with the appropriate data.

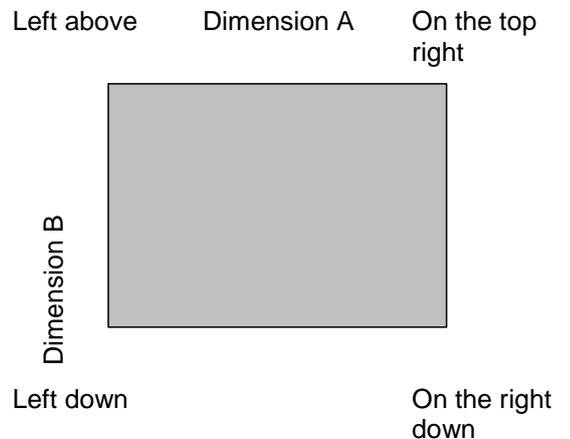
(→ conventions for names)



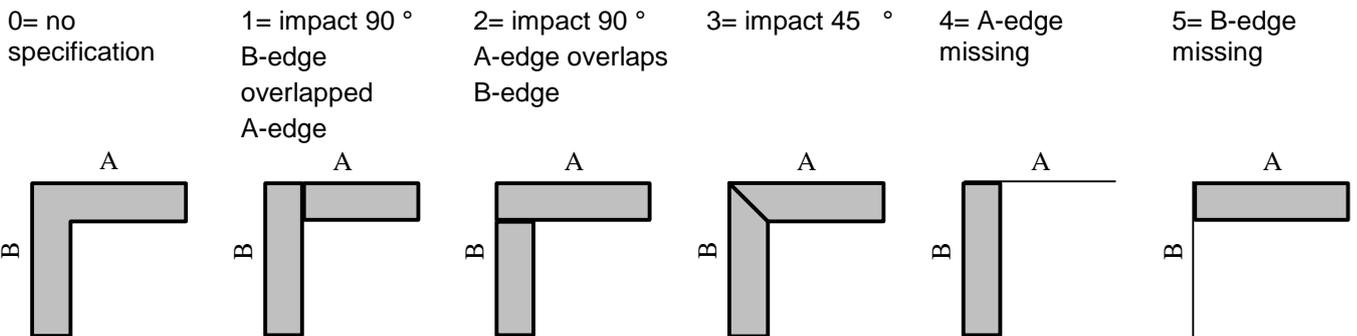
12.13.14.15... corner logo left/on the right in front/rear:

Default value = 00 (no valid corner logo).

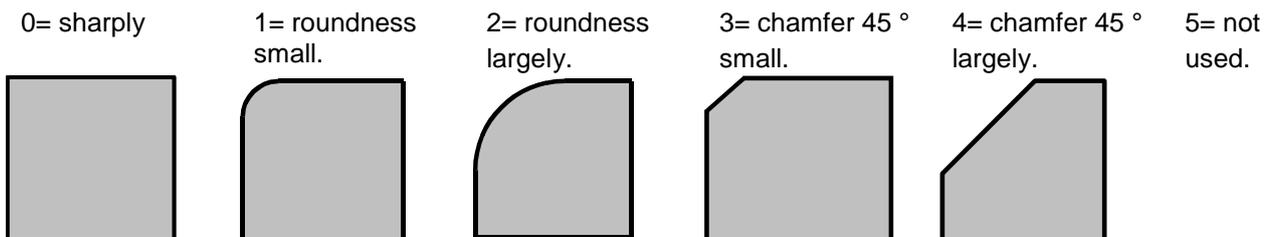
In these data fields the appropriate corners can be defined for all four corners of a format.



The code for a corner has two positions. The first code defines the type of the impact and the type of the overlap:



The second code defines the corner formation:



- Sharp corner: The corner does not receive any handling. (standard case).
- Roundness small: The corner keeps only a minimum roundness attached. (e.g. Radius=2mm).

- Chamfer small: The corner receives only a minimum chamfer (e.g. Radius=2mm).
- Roundness largely: The corner receives a roundness with radius = edge thickness / 2.
- Chamfer largely: The corner receives a chamfer with edge = edge thickness / 2.

16.17... intermediate dimension A / B:  
Default value = 0.

0= no valid intermediate dimension. This dimension is required for processing step 2, which can become necessary during the mounting of edges.

In the processing step 1 the format receives an oversize. Then it is veneered and cut to size afterwards on the intermediate dimension specified here. Afterwards the edges are glued up. Then the format is cut to size on the actual cutting dimension.

- Cutting dimension = format with oversize
- Intermediate dimension = format dimension without edges
- Finished dimension = format completely finished.

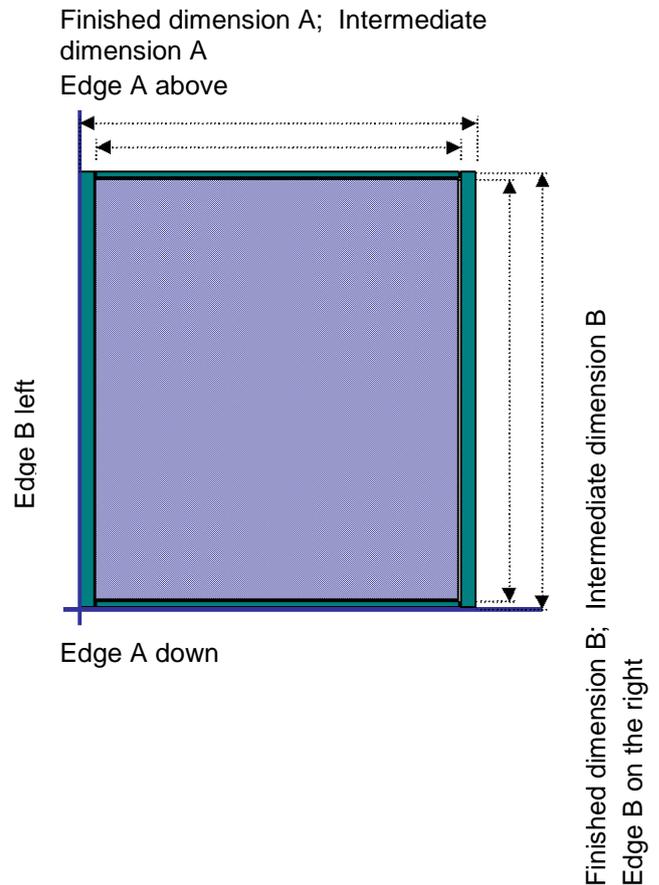
(→machine functions - process with post processing, → conventions for dimensions)

18... groove program: Default value = [ 20 blanks ].

Not in use.

19... window program: Default value = [ 20 blanks ].

Not in use.



### 1.2.3.12 Format strip handling line (F 81)

This line is not relevant for Import.

### 1.2.3.13 Format drilling program line (F 82)

The format drilling program line is optional. If the line is available, it must follow after the format-base-line of the respective format. It can be available only one format drilling program line per format.

! This data record is written only with software-Version > = 01,14.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F82	
1	Identifier F for format line	1A	1		
2	Identifier 82 for drilling program line	2N	3		
3	Name	20A	6	m_Name	char [ ]
4	Description	20A	27	m_Beschreibung	char [ ]
	CR LF		47		

3... Name of the drilling program: This data field contains the name the drilling program. It concerns thereby additional information. The drilling program does not have a meaning for the handling at the saw.

4... Description: This data field contains an additional description for the drilling program.

### 1.2.3.14 Format-window-line (F 84)

⇒ [Table of contents](#)

The Format window line is optional. If the row exists, it must follow the format of the respective base-line format

Box	Description	L/A	Pos	Attributes	Data type
-	Class-Name			m_F84	
1	Identifier F for format line	1A	1		
2	Identifier 84 for window	2N	3		
3	Index	2N	6	m_iIndex	Int
4	Start position A	8N	9	m_iStartPosA	Int
5	Start position B	8N	18	m_iStartPosB	Int
6	Dimension A	8N	27	m_iMassA	Int
7	Dimension B	8N	36	m_iMassB	Int
	CR LF		44		

#### 3... running index

4,5... Start position A / Start position B: The starting position A is measured from the left edge of the B-format. The start position B is measured from the upper (rear) A-edge. (see diagram below)

6,7... Dimension A / B: The distance A is measured parallel to the dimension A of the format. The dimension B measured parallel to dimension B of the format. (see diagram below)

S\_A= Start position A

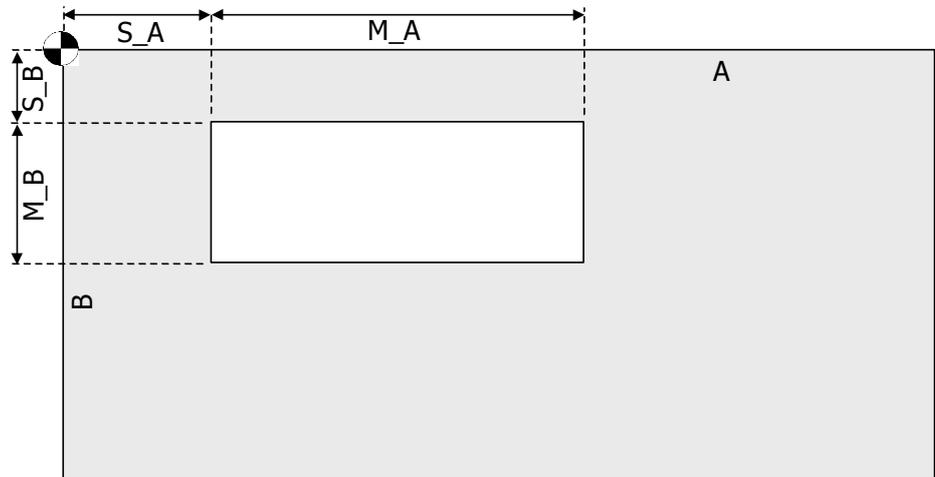
S\_B= Start position B

M\_A= Dimension A

M\_B= Dimension B

B= left B-edge

A= rear A-edge  
(= upper A-edge)



### 1.2.3.15 Format-groove-line (F 85)

⇒ [Table of contents](#)

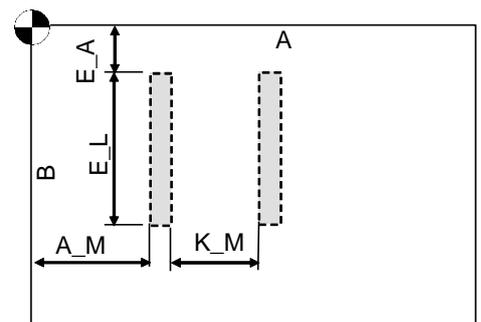
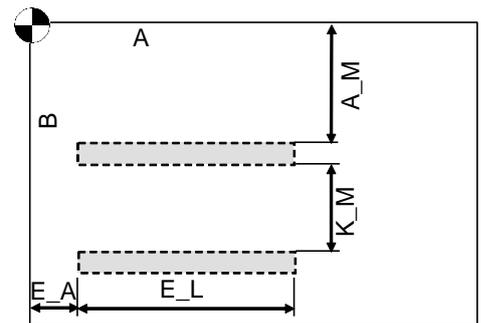
The format-grooving line is optional. If the row exists, it must follow the format of the respective base-line format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class -Name			m_F85	
1	Identifier F for format line	1A	1		
2	Identifier 85 for grooving	2N	3		
3	Index	2N	6	m_iIndex	Int
4	Reference edge	1N	9	m_iBezugskante	Int
5	Side of the groove	1N	11	m_iSeite	Int
6	Absolute dimension	8N	13	m_iAbsolutmass	Int
7	Chain dimension	8N	22	m_iKettenmass	Int
8	Number of grooves	8N	31	m_iAnzahl	Int
9	Groove width	8N	40	m_iNutbreite	Int
10	Groove depth	8N	49	m_iNuttiefe	Int
11	Groove start	8N	58	m_iEinsatzAnfang	Int
12	Groove length	8N	67	m_iEinsatzLaenge	Int
	CR LF		75		

#### 3... running index

#### 4... Reference edge:

- 0 = The groove is parallel to the A-edge of the format. Absolute dimension  $A_M$  and chain dimension  $K_M$  are measured from the upper (rear) A-edge. The start position of the groove  $E_A$  and the length of the groove  $E_L$  are measured from the left B-edge
- 1 = The groove is parallel to the B-edge of the format. Absolute dimension  $A_M$  and chain dimension  $K_M$  are measured from the left B-edge. The start position of the groove  $E_A$  and the length of the groove  $E_L$  are measured from the upper A-edge.



5... Side of the grooving: 0..Inside (bottom), 1..Outside (top)

6... Absolut dimension: The absolute dimension is the position where the first groove is located. (see diagram below)

7... chain dimension: (Default =0). See diagram below.

8... number of grooves: (Default =1). Number of consecutive grooves of the same kind.

9,10,11,12... Groove width, Groove depth, groove start, groove length: see diagram below.

Example:

A\_M=Absolut dimension

K\_M= chain dimension

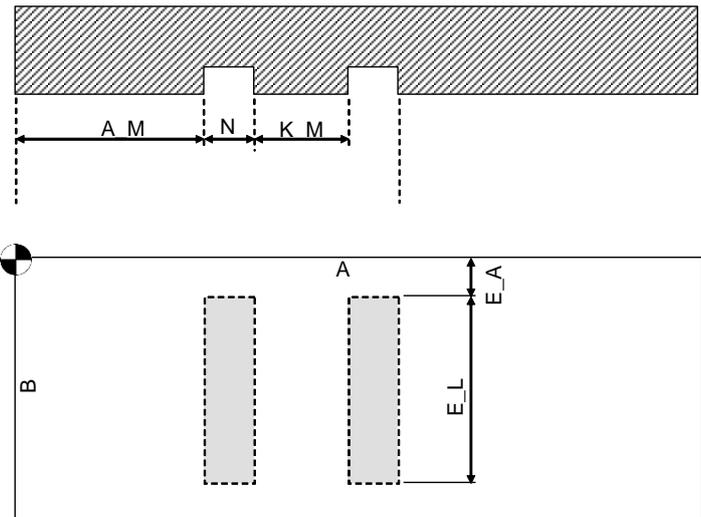
N= groove width

B= left B-edge

A= upper A-edge  
(rear A-edge)

E\_A= start position

E\_L= length



In this example, the reference edge is 1. The groove is parallel to the dimension B of the format. Absolute dimension, chain dimension and groove width are measured from the left B-edge. Start position and the length are measured from the upper (rear) A-edge.

### 1.2.3.16 Format-milling-line (F 86)

⇒ [Table of contents](#)

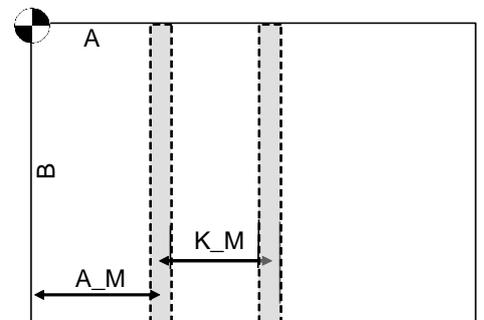
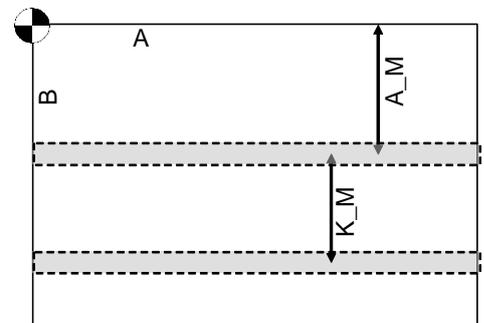
The format-milling line is optional. If the row exists, it must follow the format of the respective base-line format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class -Name			m_F86	
1	Identifier F for format line	1A	1		
2	Identifier 86 for milling	2N	3		
3	Index	2N	6	m_iIndex	Int
4	Reference edge	1N	9	m_iBezugskante	Int
5	Side of the milling	1N	11	m_iSeite	Int
6	Absolut dimension	8N	13	m_iAbsolutmass	Int
7	Chain dimension	8N	22	m_iKettenmass	Int
8	Number of millings	8N	31	m_iAnzahl	Int
9	Milling depth	8N	40	m_iFraestiefe	Int
10	CR LF		48		

#### 3... running index

#### 4... Reference edge:

- 0 = The milling is parallel to the A-edge of the format. Absolute dimension A\_M and chain dimension K\_M are measured from the upper (rear) A-edge.
- 1 = The milling is parallel to the B-edge of the format. Absolute dimension A\_M and chain dimension K\_M are measured from the left B-edge.



5... Side of the milling: 0..Inside (bottom), 1..Outside (top)

6... Absolut dimension: The absolute dimension is the position where the first milling is located. (see diagram below)

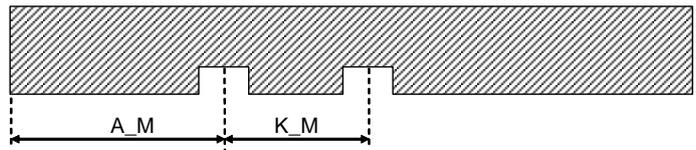
7... chain dimension: (Default =0). See diagram below.

8... number of grooves: (Default =1). Number of consecutive millings of the same kind.

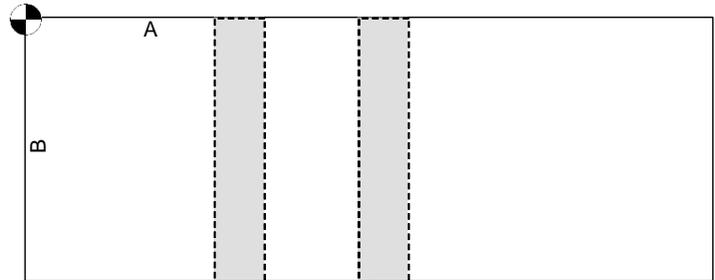
9... milling depth:

Example:

A\_M=Absolut dimension  
K\_M=Chain dimension



B= left B-Edge  
A= upper A-Edge  
(rear A-edge)



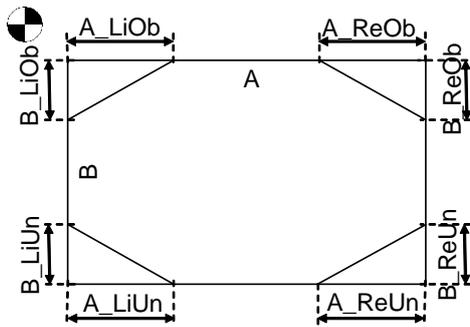
In this example, the reference edge is 1. The milling is parallel to the B-dimension of the format. Absolut dimension and chain dimension are measured from the left B-Edge.

### 1.2.3.17 Format-angle-line (F 87)

[⇒Table of contents](#)

The format-angle line is optional. If the row exists, it must follow the format of the respective base-line format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class -Name			m_F87	
1	Identifier F for format line	1A	1		
2	Identifier 87 for angle	2N	3		
3	Miter dimension A left/bottom	8N	6	m_iMassALiUn	Int
4	Miter dimension B left / bottom	8N	15	m_iMassBLiUn	Int
5	Miter dimension A left /top	8N	24	m_iMassALiOb	Int
6	Miter dimension B left / top	8N	33	m_iMassBLiOb	Int
7	Miter dimension A right/ top	8N	42	m_iMassAReOb	Int
8	Miter dimension B right / top	8N	51	m_iMassBReOb	Int
9	Miter dimension A right / bottom	8N	60	m_iMassAReUn	Int
10	Miter dimension B right / bottom	8N	69	m_iMassBReUn	Int
11	CR LF		77		



### 1.2.3.18 Format-miter-line (F 88)

[⇒Table of contents](#)

The format-miter line is optional. If the row exists, it must follow the format of the respective base-line format.

Box	Description	L/A	Pos	Attributes	Data type
-	Class -Name			m_F88	
1	Identifier F for format line	1A	1		
2	Identifier 88 for miter	2N	3		
3	Angle of the miter bottom	3N	6	m_iWinkelGehUn	Int
4	Angle of the miter top	3N	10	m_iWinkelGehOb	Int
5	Angle of the miter left	3N	14	m_iWinkelGehLi	Int
6	Angle of the miter right	3N	18	m_iWinkelGehRe	Int
7	Angle of the edge left / bottom	3N	22	m_iWinkelEckeLiUn	Int
8	Angle of the edge left / top	3N	26	m_iWinkelEckeLiOb	Int
9	Angle of the edge right / top	3N	30	m_iWinkelEckeReOb	Int
10	Angle of the edge right / bottom	3N	34	m_iWinkelEckeReUn	Int
11	CR LF		37		

1-10... Angle: -47° to +47°

### 1.2.3.19 Format handling line (F 90)

This line contains additional information about the post processing a format. It is optional. If this line is available, it must follow after the base line of the respective format. It may be available only one format handling line per format.

! This data record is written only with software-Version > = 01,14.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_F90	
1	Identifier F for format line	1A	1		
2	Identifier 90 for handling line	2N	3		
3	Raw cut active	1N	6	m_RSchnittAktiv	Int
4	Intermediate cut active	1N	8	m_ZSchnittAktiv	Int
5	Finish cut active	1N	10	m_FSchnittAktiv	Int
6	Processing step groove program	1N	12	m_BSchnittNutPrg	Int
7	Reference groove program	1N	14	m_BezugNutPrg	Int
8	Processing step window program	1N	16	m_BSchnittFensterPrg	Int
9	Reference window program	1N	18	m_BezugFensterPrg	Int
10	Processing step miter program	1N	20	m_BSchnittGehrPrg	Int
11	Reference miter program	1N	22	m_BezugGehrPrg	Int
12	Processing step drilling program	1N	24	m_BSchnittBohrPrg	Int
13	Reference drilling program	1N	26	m_BezugBohrPrg	Int
14	Processing step surface above	1N	28	m_BSchnittOberflOb	Int
15	Reference surface above	1N	30	m_BezugOberflOb	Int
16	Processing step surface down	1N	32	m_BSchnittOberflUnt	Int
17	Reference surface down	1N	34	m_BezugOberflUnt	Int
	CR LF		35		

⇒ [Table of contents](#)

3... Raw cut active: Default value = 1. This data field contains the information whether for this format a raw cut must be executed. The value for this data field is calculated from other information and cannot not be edited. From today's view the raw cut is always available. There is however scenarios conceivably, with which at the saw no raw cut, but only post processings takes place. The number of the processing step for the raw cut is always „1.“

4... Intermediate cut active: Default value = 0. This data field contains the information whether for this format an intermediate cut must be executed. The value for this data field is calculated from other information and cannot not be edited. The number of the processing step for the intermediate cut is always „2,“ even if there is no raw cut.

5... Finish cut active: Default value = 0. This data field contains the information whether for this format a finish cut must be executed. The value for this data field is calculated from other information and cannot not be edited. The number of the processing step for the raw cut is always „3,“ even if there is no intermediate cut or no raw cut.

6,8,10,12,14,16... Processing step program: Default values: 3 for groove program; 3 for window program; 3 for miter program; 3 for drilling program; 1 surface above; 1 for surface down.

In this data field one indicates, to which processing step the post processing program refers. Thus also one determines, to which dimensions the post processing program refers. The processing steps are indicated according to the possible processing steps in the saw. If the processing step enables a handling in the book, the post processing program must take place nevertheless with individual formats.

1= the post processing program is executed together with processing step 1. It refers to the dimensions of the raw cut. Even if the raw cut does not take place at the saw, the dimensions of the raw cut must admit to be.

2= the post processing program is executed together with processing step 2. It refers to the dimensions of the intermediate cut. If the format does not require the processing step 2 actually, the processing step 2 is activated by the post processing program. The post processings refer then to the dimensions of the raw cut.

3= the post processing program is executed together with processing step 3. It refers to the dimensions of the finish cut. If the format does not require the processing step 3 actually, the processing step 3 is activated by the post processing program. The post processings refer then to the dimensions of the intermediate cut or the raw cut.

7,9,11,13,15,17... Reference groove program / window program / miter program:

Default values: 1 for groove program; 1 for window program; 2 for miter program; 1 for drilling program; 2 surface above; 2 for surface down.

In this data field it is indicated whether the groove program is executed at the same time with the processing step or after the processing step.

1= the post processing program is executed at the same time with the processing step, which is „indicated “in the data field processing step.

2= the post processing program is executed in the following on the processing step, which is „indicated “ in the data field processing step.

### **1.2.3.20 Format statistics line (F99)**

This line is for format-referred, calculated statistical data reserved.

## 1.2.4 Post processing data records

The post processing data records are schedule-referred or part list-referred. Each post processing data record may appear only once in the NC data. It can be used however for several formats. For a format also any combination of post processing data records can be used.

The data records are used at three different places:

- For part lists for each type of a post processing a file is created. The file carries the names of the part list. File extension corresponds to the type of the post processing. Within these files the sequence of the data records is free. Data records with the same identifier are combined into a block.
- To data bases its own file structure applies. The internal construction of the data records and data record blocks is alike as with part lists (→ edge data base etc..)
- For schedule NC data the post processing data records of a sort are combined into blocks. Within these blocks the sequence of the data records is free. The sequence of the blocks among themselves is as follows: Edges, groove programs, window programs.

### 1.2.4.1 Internal construction of post processing files

#### Edges for part list:

File name:	BeispielStueckliste23.kan	
m_K01	<a href="#">Edge base line (K 01)</a>	at least one line for each edge file
m_K01	<a href="#">Edge base line (K 01)</a>	at least one line for each edge file
	etc..	
m_K\$	<a href="#">Edge-end-line (K\$)</a>	exactly one line for each part list edge file

#### Edge data base:

File name:	DBKanten[id,if,mm].kdb	
m_K02	Edge-database-base-line (K 02)	exactly one line for each edge data base file
m_K01	Edge base line (K 01)	exactly one line for each edge data base file
	etc..	
m_K\$	Edge-end-line (K\$)	exactly one line for each edge data base file

### 1.2.4.2 Edge base line (K 01)

Here all information is entered, which is required for the edges of a format. For each edge identifier only one edge-line may exist. The edge data records are used in part lists, NC data and in the edge data base. There are two different file identifications. (→ directory structure and file structure)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_K01	
1	Identifier K for edge-line	1A	1		
2	Identifier 01 for base line	2N	3		
3	Edge identifier	20A	6	m_Kennung	char[ ]
4	Edge material	20A	27	m_Material	char[ ]
5	Base dimension edge	8N	48	m_Rohmass	int
6	Finished dimension edge	8N	57	m_Fertigmass	int
7	Edge is covered	1N	66	m_Bedeckt	int
8	Description	20A	68	m_Beschreibung	char[ ]
9	Price of the edge per length	10N	89	m_PreisProLfm	int
10	Additive dimension	8N	100	m_Zuschlagmass	int
11	Subtractive dimension	8N	109	m_Abschlagmass	int
	CR LF		117		

3... Edge identifier: The edge identifier is the name, on which the format-post-processing-lines refer. The same edge identifier can be used for many formats. Therefore the information for each edge is only once defined. (→ conventions for names)

4... Edge material: Default value = [ 20 blanks ]. Here a name of the edge material can be indicated. The specification is purely informative. (→ conventions for names)

5... Base dimension edge: That is the edge strength before the finish cut at the saw. The base dimension for the determination of the format size before cut to size on the finished dimension required.

(→ conventions for dimensions)

6... Finished dimension edge: That is the thickness of the edge after the finish cut at the saw. If an edge is indicated, either the cutting dimension or the finished dimension must be calculated. That dimension is calculated, which is configured as the second in the input order with the format

(→ conventions for dimensions)

7... Edge is covered: Default value = 0. 0=no, 1=yes. This data field indicates whether the edge is covered by a coating (e.g. veneer).

8... Description of edge: This data field contains an additional description for the edge, e.g. for following processing steps.

9... Price of the edge per length: This data field contains the cost per unit of length for the edge material. (→ conventions for prices)

If the system of units valid for the edge is metric, then the cost per meter is counted.

If the system of units inch valid for the edge is, then the cost per foot is counted.

**10... Additive dimension:** This dimension is used for the calculation of the intermediate dimension of the format. The intermediate dimension results from cutting dimension of the format plus additive dimensions of the edges minus subtractive dimensions of the edges., (→ conventions for dimensions)

**11... Subtractive dimension:** This dimension is used for the calculation of the intermediate dimension of the format. The intermediate dimension results from cutting dimension of the format plus additive dimensions of the edges minus subtractive dimensions of the edges. (→ conventions for dimensions)

**Examples:**

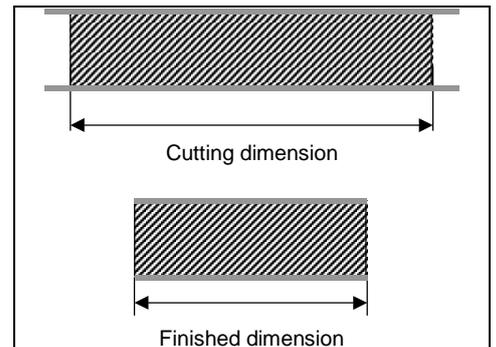
If for the edge no edge identifier is entered (format-post-processing-line)

**and**

if for the cut direction no intermediate dimension was indicated (format-post-processing-line)

**and**

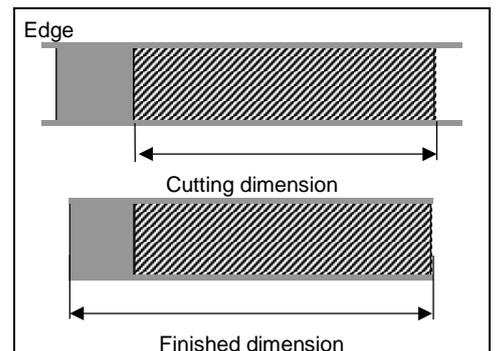
if the cutting dimension is larger than the finished dimension (format-base-line) [FormatBasisZeile](#)



**then** no edge is applied and nevertheless the format is after-sawed. A possible reason for it is glued on a in the meantime veneers, which must be trimmed. The finished dimension must be indicated directly, because it cannot be calculated.

If the edge an edge identifier covered with „the characteristic edge“(edge base line)

**then** the surface covers the edge. Therefore the format must be after-sawed.



The finished dimension is calculated from cutting dimension, base dimension of the edge and finished dimension of the edge.

If an edge for format was indicated

**and**

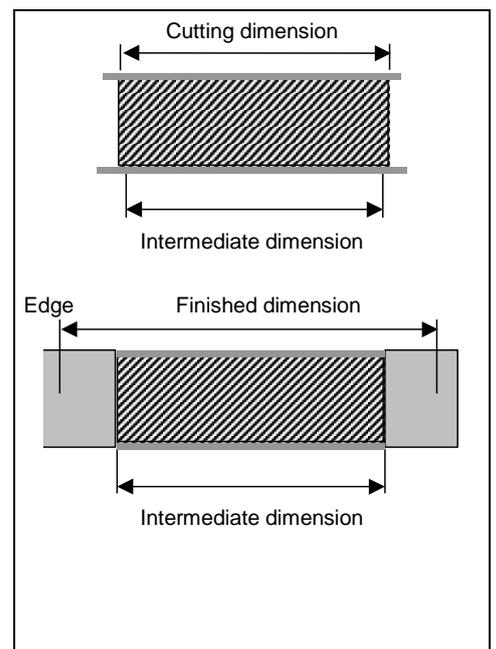
if for the cut direction an intermediate dimension was indicated

**and**

if the cutting dimension is larger than the intermediate dimension

**then**

after gluing the surfaces on the format is again trimmed. After it the edges are only glued on. The finish cut takes place last. The intermediate dimension can be indicated either directly or be calculated from cutting dimension, additive dimension and subtractive dimension. The finished dimension is calculated from intermediate dimension, base dimension of the edge and finished dimension of the edge.



### 1.2.4.3 Edge-database-base-line (K 02)

This line serves for the specification of general information for the edge lines in the edge data base. (→ edge data base)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_K02	
1	Identifier K for edge-line	1A	1		
2	Identifier 02 for data base line	2N	3		
3	System of units	1N	6	m_MassSystem	int
	CR LF		7		

3... System of units: For each system of units its own edge data base file is created. In this data field the system of units is indicated. (→ conventions for dimensions)

### 1.2.4.4 Edge-end-line (K\$)

This line serves for recognition of a correct end of file. It is used only in edge files.

(→ directory structure and file structure)

Box	Description	L/A	Pos	Attributes	Data type
1	Identifier K\$ for edge-end-line	2N	1		
	CR LF	1A	3		

## 1.2.5 Board data records

The board data records in part lists, NC data are used and in the board data base. Within each area all data fields are not used. Nevertheless the same data records are used.

There are two different file identifications. (→ directory structure and file structure)

- **\*.pla** for the board files for part lists, Optimization orders and NC data.
- **\*.pdb** for the board files of the board data base.

Board-lines, which refer to the same board number, must follow within a block directly consecutively. They must be ascending sorted according to their number.

All board-lines, which concern remainders, must be combined into a block and be directly after the final format line of the schedule (→ remainders)

### 1.2.5.1 Internal construction of a board file

Example of a part list-referred board file:

File name:	BeispielStueckliste23.pla	
m_B01	Board-base-line (B 01)	at least one line for each board file
m_B10	Board-customer-text-line (B 10-19) 1	0 or 1 line for each board file
to		
m_B10	Board-customer-text-line (B 10-19) 10	0 or 1 line for each board file
m_B01	Board-base-line (B 01)	at least one line for each board file
m_B10	Board-customer-text-line (B 10-19) 1	0 or 1 line for each board file
to		
m_B10	Board-customer-text-line (B 10-19) 10	0 or 1 line for each board file
m_B\$	Board-end-line (B\$)	

Example of board data base file:

File name:	Board type name 0-3-4".pdb	
m_B02	Board-database-base-line (B 02)	exactly one line for each file (board type/board thickness)
m_B01	Board-base-line (B 01)	at least one line for each board file
m_B10	Board-customer-text-line (B 10-19) 1	0 or 1 line for each board file
to		
m_B10	Board-customer-text-line (B 10-19) 10	0 or 1 line for each board file
m_B01	Board-base-line (B 01)	at least one line for each board file
m_B10	Board-customer-text-line (B 10-19) 1	0 or 1 line for each board file
to		
m_B10	Board-customer-text-line (B 10-19) 10	0 or 1 line for each board file
m_B\$	Board-end-line (B\$)	

### 1.2.5.2 Board-base-line (B 01)

Here all general information for a board is entered. For each board or for each remainder a board-base-line must be available.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_B01	
1	Identifier B for boards line	1A	1		
2	Identifier 01 for base line	2N	3		
3	<a href="#">Sequential position number</a>	4N	6	m_LfdPosNr	int
4	<a href="#">Category</a>	1N	11	m_Kategorie	int
5	<a href="#">Board type</a>	20A	13	m_Typ	char[ ]
6	<a href="#">Board thickness</a>	8N	34	m_Dicke	int
7	<a href="#">Board dimension A</a>	8N	43	m_MassA	int
8	<a href="#">Board dimension B</a>	8N	52	m_MassB	int
9	<a href="#">Structure</a>	1N	61	m_Struktur	int
10	<a href="#">Description</a>	20A	63	m_Beschreibung	char[ ]
11	<a href="#">Quality of the board</a>	1N	84	m_Qualitaet	int
12	<a href="#">Priority of the board</a>	1N	86	m_Prioritaet	int
13	<a href="#">Cost per unit</a>	10N	88	m_PreisProEinheit	int
14	<a href="#">Unit for price calculation</a>	1N	99	m_EinheitFuerPreis	int
15	<a href="#">Board is rotatable</a>	1N	101	m_Drehbar	int
16	<a href="#">Board only turned use</a>	1N	103	m_NurGedreht	int
17	<a href="#">Trim A down</a> (lengthwise in front)	8N	105	m_BesLV	int
18	<a href="#">Trim A above</a> (lengthwise rear)	8N	114	m_BesLH	int
19	<a href="#">Trim B on the right</a> (crosswise in front)	8N	123	m_BesQV	int
20	<a href="#">Trim B left</a> (crosswise rear)	8N	132	m_BesQH	int
21	<a href="#">Storage place for Optimization</a>	2N	141	m_LpOpti	int
22	<a href="#">Quantity of boards on stock</a>	6Ñ	144	m_AnzGelagert	int
23	<a href="#">Quantity of boards plan</a>	6Ñ	151	m_AnzVerplant	int
24	<a href="#">Specific weight</a>	8N	158	m_SpezGew	int
25	<a href="#">Boards exceeding usage</a>	6Ñ	167	m_Mehrverw	int
26	<a href="#">Release</a>	1N	174	m_Freigabe	int
27	<a href="#">Stock code</a>	20A	176	m_LagerCode	char[ ]
28	<a href="#">Parameter set Optimization</a>	20A	197	m_ParSatzOpti	char[ ]
29	<a href="#">Internal board number</a>	9N	218	m_InterneNr	int
30	<a href="#">Infeed device</a>	3Á	228	m_BeschickGer	char[ ]
31	<a href="#">Waste disposal</a>	3Á	232	m_AbfallEnts	char[ ]
32	<a href="#">Board proportion</a> P1/Pn	3N	236	m_PlattenVerhaeltnis	Int
33	<a href="#">Minimum stock quantity</a>	6N	240	m_Lagermenge	Int
34	<a href="#">Revolve board</a>	1N	247	m_Wenden	int
35	<a href="#">Nc5 board stock ID</a>	38A	248	m_guidNc5StatusId	CSDGuid
36	<a href="#">Nc5 board stock using ID</a>	38A	287	m_guidNc5LagerVerwendungId	CSDGuid
37	<a href="#">Temperature factor</a>	8N	326	m_TemperaturAusdehnungskoeff	Int
38	<a href="#">Min. Usage factor</a>	5N	336	m_MindestAusnutzung	int
39	<a href="#">Batch-ID board</a>	20A	342	m_ChargenID	char[ ]
40	<a href="#">Additional handle information</a>	2N	363	m_ZusaetzlicheInfo	int
	CR LF		365		

⇒ [Table of contents](#)

3... sequential position number: This number serves for the reference from a part list, from a schedule or from a cutting pattern (→ **Fehler! Verweisquelle konnte nicht gefunden werden.**)

4... category: Default value =0. this data field assigns the board to a certain category. The category is indicated by a number from 0 to 9. At present the following categories are defined:

- 0= the board is a normal board.
- 1= the board is a remainder. (→ remainders)
- 2= the board is a longitudinal strip.
- 4= the board is a XL remainder
- 5= the board is a block part
- 6= the board is a cover board

5... Board type: Name of the board type. Under this name are all further board specific values to find. (→ conventions for names)

6... board thickness: Together with the board type name the board thickness defines unique a certain board type. (→ conventions for dimensions)

7.8... board dimension A / board dimension B: Indicates the actual dimensions of the board. (→ conventions for dimensions)

9... structure: This data field indicates whether the surface of the board has a structure (grain) or not. If the board has a structure, formats may not be planned turned. The structure runs always parallel to the A-edge of the board. The dimensions of the raw board must be accordingly input. (→ conventions for orientation)

- 0= the surface of the board does not have structure.
- 1= the surface of the board has a structure.

10... description: Here an additional description for the board can be indicated.

11... quality of the board: Default value = 9 This data field contains a value between 0 and 9, which indicates the quality of the board. The formats of a part list can be only optimized, if the quality of the board fulfills the quality requirements of the formats. The board quality must be at least as high as the quality of the formats.

- 0= lowest quality.
- 9= highest quality (default value).

12... priority of the board: Default value =0 This data field contains a value between 0 and 9, which indicates the priority of the board. The board is the more probably planned by the Optimization, the more highly this value is.

- 0= normal priority (default value)
- 9= highest priority (mandatory use).

13... cost per unit: Default value = 1. The price indicated here refers to the unit indicated in data field 14. Since the price is used by the Optimization, it may not contain the value zero. (→ conventions for prices)

14... unit for price calculation: Default value = 0.

- 0= cost per unit of area (m<sup>2</sup> or feet<sup>2</sup>).
- 1= cost per unit of volume (m<sup>3</sup> or feet<sup>3</sup>).
- 2= cost per unit of length (m or feet).
- 3= cost per unit of weight (kg or Pound).

15... board is rotatable: Default value =1. This data field has an effect only if the parameter 'is board rotatable' in the board type on 'yes' is set.

- 1= the board is rotatable. It may be planned normally or turned.
- 0= the board may be planned only unturned.

16... board only turned use: Default value =0. So that the board can be planned only turned, the parameter must be 'board is rotatable' in the board type, on 'yes' set. Otherwise this data field does not have effect. This switch has priority over the information 'board is rotatable', if there are thereby contradictions. The switch is only of importance for the transfer from CAD programs, which have a rigid allocation of length and width.

- 0= the board may be planned in both orientations.
- 1= the board may be planned only turned.

17.18.19.20... trim A / B: Default value =0. This data field contains the size for the quality-conditioned ingate of the board. If the data field contains 0, the Optimization uses the trim specification from the board type.

(→ conventions for dimensions)

21... storage place for Optimization: Default value = 01. This data field contains an index for stocks, in which the board is on stock.

Position 1	Position 2
0... Automatic stocks	1 to 9... Stocks 1 to 9
1... Manual stocks	

22... quantity of boards on stock: This data field indicates, how many boards of this type are available in the board stock.

23... quantity of boards plans: Indicates, how many boards of this type for the current Optimization order or the current schedule are used. If the board is a remainder, then this number indicates, how are contained in the current schedule much piece of this remainder.

24... specific weight: Default value =0 The value indicated in gram per Cubic decimeter integer type as 8 positions. If as unit price / kg is used, then this value must be larger than 0.

25... boards exceeding usage: Default value =0 This data field indicates, how many boards may be used additionally for the stock. Example: If the stock is 100 and the exceeding usage is 10, then 110 boards are to the Optimization at the disposal.

26... release: Default value =0.

- 0= the boards can be used for the Optimization . They are approved.
- 1= the boards may be not used by the Optimization. They are locked.
- 2= the boards may be not used by the Optimization. They are reserved.

27... Stock code: This is the location of the raw board in text form.

28... parameter set Optimization: Default value = empty field (spaces). That is the name of the parameter set, which is to be used for the Optimization of this board.

(→ conventions for names)

29... Internal board number: Each board receives a unique number during entry. This number may not be changed also with a modification of the part list or with a modification of the data base.

30... infeed device: Default value = empty field (spaces). Here can be indicated, with which infeed device these boards to be fed in to be supposed. The specification has a sense only if several infeed devices are available. The code assembles itself as follows:

Position 1	Position 2	Position 3
------------	------------	------------

V... Vacuum loader  
 A... Infeed pusher  
 M... Nip roller infeed  
 F... Foreign infeed section  
 H... Manual infeed section

1... Device number 1  
 2... Device number 2  
 etc..

1... Orientation lengthwise  
 2... Orientation crosswise

31... waste disposal: Default value = empty field (spaces). Here can be indicated, where the release of these boards are to be disposed. The specification has a sense only if several waste disposals are available.

32... Board proportion: Default value = 0. Here you can specify the ratio between a board and another board size.

33... Min. stock quantity: Default value = 0. If the stock quantity goes below this value, a warning message appears in the optimization program.

34... Revolve Board: Default value =0. This data field indicates, if the board has to be revolved or not.

- 0= Board must not be revolved
- 1= Board must be revolved

35, 36: Default values = {00000000-0000-0000-0000-000000000000}. This is for internal use only.

37... Temperature factor: Default value =0. This data field indicates the expansion of the material in mm per m and per degree Celsius.

38... Min. usage factor: Default value =0. The board only will be used for a pattern, if this min. usage is given. Value is in 1/100 percent without comma separator. Default value =0

39... Batch-ID board: Default value = empty field (spaces). Indicates the Batch-ID of the board. This is implemented for the board categories 0, 1 and 4 (please see also field 4 Category).

40... Additional handle information: Default value = 0. A value of 1 indicates that a customer specific board barcode ID has to be checked before infeeding the board/stack into infeed area. The reference value is stored into an adjustable customer text field.

### 1.2.5.3 Board-customer-text-line (B 10-19)

These lines are optional. They serve for the transfer of additional texts. Maximum 10 lines can be input. Each line has maximum 70 characters.

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_B10 to m_B19	
1	Identifier B for boards line	1A	1		
2	Identifier 10-19 for text line xx	2N	3		
3	Text CR LF	110A	6 76	m_Text	char[ ]

### 1.2.5.4 Board-end-line (B\$)

This line serves for recognition of a correct end of file. It is used only in board files. (→ directory structure and file structure)

Box	Description	L/A	Pos	Attributes	Data type
1	Identifier B\$ for board-end-line	Å	1		

### 1.2.5.5 Board-database-base-line (B 02)

This line serves as additional information for boards within the board data base. The information in this line is redundant to the file name and applies with priority before the information of the file name. With discrepancies thus the file name must be adapted. Also the board type in the board-base-line must be adapted. (→ board data base)

Box	Description	L/A	Pos	Attributes	Data type
-	Class name			m_B02	
1	Identifier B for boards line	1A	1		
2	Identifier 02 for data base base line	2N	3		
3	Board type	20 A	6	m_Typ	char[ ]
4	Board thickness	8N	27	m_Dicke	int
5	System of units	1N	36	m_MassSystem	int
	CR LF		37		

### 1.2.5.6 Board-statistic-line (B99)

This line is reserved for calculated statistical data, which refer to the boards.

## 1.3 Data Records Stacking Area

### 1.3.1 General

**Data records:** each data record (each line) is terminated with CR LF (0Dh 0Ah).

**Data fields:** each data field is of a defined size and is separated from other data fields by a blank. A data record must contain all specified data fields. No data fields may be omitted. Therefore each data field starts always at the same position in the ASCII string.

**Alphanumerical data fields:** these data fields must be entered flush left. Leading blanks are not allowed. If the complete data field length is not used, and further data fields follow in the data record, then the remainder of the data field must be filled with blanks.

**Numerical data fields:** numerical data fields must be entered flush right and must be filled with leading zeros to the full data-field length.

**Attribute:** the attributes specified in the data records are the assigned extensions of variable names of the appropriate type (e.g. 'any\_variable\_name.attribut').

**Variable types:** all programs are developed for 32 bit operating systems. 16 bit operating systems are not supported.

All numerical variables are of integer data type. The integer (int) data type is also of 32 bit length in 32 bit operating systems. For date variables SYSTEMTIME (Win32) is used.

**Line:** the designation line and data record is used synonymously. The lines in the ASCII files have data-record character as in a data model. The lines normally don't include reference information as the references are specified by the position of the lines in the ASCII file.

## 1.3.2 Format Data Records

### 1.3.2.1 Format stacking line (F 02)

This line contains static preset-stack data of the corresponding format and is optional. If this line exists then it must be after the base line of the corresponding format. Only one format-stacking line per format is allowed. Unused integer data fields are marked with the appropriate amount of "X" characters. The designation "X" is converted internally to the value -1. Unused data fields are filled by the stacking optimizing parameter that can be changed in several steps (general, format size, board type). Unused integer data fields are marked with the appropriate amount of "X" characters. The designation "X" is converted internally to the value -1. Unused ASCII fields are marked with SPACES.

field	description	l/a	pos	attribute	data type
-	class name			m_F02	CNcF02
1	designation F for format line	1A	1		
2	designation 02 for format-stacking line	2N	3		
3	stacking height set value	8N	6	m_StHoeheSoll	int
4	unit of stacking height	1N	15	m_StHoeheEinh	int
5	minus tolerance stacking height	2N	17	m_StHoeheMinTol	
6	plus tolerance stacking height	2N	20	m_StHoehePlusTol	int
7	plus tolerance stacking height last stack	2N	23	m_StHoehePlusLetzt	int
8	minimum amount of format dimension A in stack pattern	4N	26	m_StPlanAnzAMin	int
9	maximum amount of format dimension B in stack pattern	4N	31	m_StPlanAnzAMax	int
10	minimum amount format dimension B in stack pattern	4N	36	m_StPlanAnzBMin	int
11	maximum amount format dimension B in stack pattern	4N	41	m_StPlanAnzBMax	int
12	multiple stacks allowed	1N	46	m_MehrfachStErlaubt	
13	parallel stacks allowed	1N	48	m_ParallelStErlaubt	
14	orientation in stack pattern	1N	50	m_Orientierung	int
15	stacking station	2N	52	m_StStation	int
16	stacking place number	2N	55	m_Stapelplatz	int
17	storage area	2N	58	m_Lagerplatz	int
18	stack target	3N	61	m_StZiel	int
19	packing specification	20A	65	m_VerpVorsch	char[ ]
20	strapper along format dimension A	20A	86	m_UmrEntlangA	char[ ]
21	strapper along format dimension B	20A	107	m_UmrEntlangB	char[ ]
	CR LF		128		

#### 3... Stacking height set value:

The desired stacking height is entered as per the selected unit of measurement. The stacking height does not take protection boards or pallets into consideration. „XXXXXXXX“ = stacking optimization parameter used.

4... Unit stacking height: default value =1. If a value different to 0 is entered in one of the data fields of "stacking height" then the unit code must also be specified.

0 = „X“ = stacking optimization parameter used.

1 = the set value of the stacking height is one dimension?? (→ [convention for units of measurements](#))

2 = the set value of the stacking height is a quantity (individual formats)

3 = the set value of the stacking height is a quantity of layers.

5.6... Stacking height minus/plus tolerance: default value = 50. The plus and minus tolerance of the stacking height is specified in percent. These are in reference to the set value of the stacking height. The corresponding tolerance may not be exceeded, especially when using multiple stacks.

If the plus and minus tolerance are specified as 0 percent then the exact amount of specified stacks is required. "XX" = stacking optimization parameter used.

7... Plus tolerance stacking height, last stack: default value = 50. The plus tolerance of the stacking height of the last stack is specified in percent and is referenced to the set value of the stacking height. This is to prevent a minimal stack rest at the end of the schedule. The plus tolerance may not be exceeded. "XX" = stacking optimization parameter used.

8,9,10,11... Minimum/maximum amount of format dimension A/B in stack pattern: Each a numerical value between 1 and 99. These values specify how often a format dimension A, or B, is used in the stack pattern.

- If a packing specification is specified and if therein a group name and group number is specified, then the protection board has precedence over these stack pattern specifications.
- The specified stack pattern here will be applied if the protection-board protrusion (oversize) fits this protection board with the specified group name and group number, otherwise the stack pattern will be fitted to the protection board.
- If the protection board is too small despite the stack pattern being 1x1, then an error message will be issued that this format can not be stacked on the specified protection board.
- If in the packing specification no protection board can be found for the specified group name and group number then the specified stack pattern will be applied without change and the code protection board top/bottom has full validity.
- If one of the data fields contains an "X" then the value is calculated according to the specification in the stacking-optimization parameter.
- If only the minimum, without the maximum, is specified then this value is interpreted as absolute specification.
- If only the maximum, without the minimum, is specified then the values '1 to maximum' are allowed.

12... Multiple stacks allowed: A stacking place can be utilized better by using multiple stacks. By using this data field one can force all stacks of the format to be single stacks.

1= multiple stacks allowed.

0= multiple stacks not allowed.

„X“= parallel stacks allowed or not allowed, depending on the stacking-optimization parameter.

13... Parallel stacks allowed: By using this data field parallel stacks of a format can be prohibited.

Parallel stacks means that a format can be stacked on several stacking stations and/or on several scissors lifts of a stacking station. By using parallel stacks the stacking facility can be utilized better.

1 = parallel stacks allowed.

0 = parallel stacks not allowed.

„X“= parallel stacks allowed or not allowed, depending on the stacking-optimization parameter.

14... Orientation in stack pattern: Here the desired orientation of all stacks of the format is specified.

Example:

stack specification: 2 x format dimension A, 1 x format dimension B

format orientation 0+2: : 2 columns and 1 row on scissors lift

format orientation 1+3: : 1 column and 2 rows on scissors lift

0,1,2= orientation (→ [convention for orientation](#)).

„X“ = no specification for orientation is made.

#### 15,16,17... Stacking station, stacking place, storage area:

- **Stacking station:** With this data field the use of a specific stacking station for a format can be selected. Permitted values for stacking stations are 1..99. If the data field contains "XX" then the stacking optimization selects the stacking places that are to be used. If a stacking station without turning device is selected, even if the format orientation in the cutting pattern is specified differently than in the stack pattern, then the specification of the stacking place will be ignored.
- **Stacking place :** If a specific stacking station is selected then a specific stacking place can also be selected. If this data field contains "XX" then the stacking optimization selects the stacking place that is to be used. If scissors lifts are joinable then a stacking place can consist of several scissors lifts. In this case each scissors lift can be assigned one or more stacking place numbers. A stacking place number can therefore be built up physically of several scissors lifts. These specifications can be found in the configuration of the stacking station.
- **Storage area:** In this data field a specific storage area for the format can be selected. The desired storage area is specified by a number in the range of 1-99. Currently only storage area 1 to 3 are possible. If this data field contains "XX" then the stacking optimization selects the storage area that is to be used.

18... Stack target: By entering a number of 1 to 999 a stack target can be selected in this data field. All stacks of the format are transported to this stack target by the discharge carriage. If an "X" is entered then the default value in the stacking-optimization parameter is used.

19...Packing specification: Here a file name with the packing specification for the format can be passed on. The packing specifications are stored in an ASCII file. A packing-specification file contains settings and specifications for the creation of customer-specific packing-specification files. The data of the packing specification must be converted into actual set values for a stack pattern at the creation of the be data. If the data field is filled with SPACES then the file specified in the stacking-optimization parameter is used.

(→ [directory and file structure](#))

20, 21... Strapping along the format dimension A or B: Here a file name with the strapping specification along the A or B edge can be passed on. The strapping specifications are stored in an ASCII file. A packing-specification file contains settings and specifications for the creation of customer-specific packing-specification files. The data of the strapping specification must be converted into actual set values for a stack pattern at the creation of the stack data. If the data field is filled with SPACES then the file specified in the stacking-optimization parameter is used.

(→ [directory and file structure](#))

**CAUTION:**

In each strapping specification squared timber can be specified on top and bottom. If in data field 20 and 21 strapping specifications about squared timber on top and bottom are specified then data field 20, which is strapping along format dimension A, is used and data field 21 will be ignored!

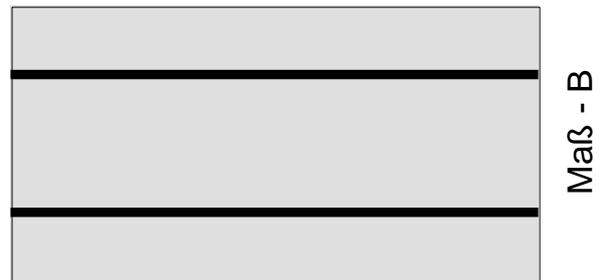
Example of strapping:

triple strapping along edge A  
of format

double strapping along edge B  
of format

**Fehler! Es ist nicht möglich, durch die Bearbeitung von Feldfunktionen Objekte zu erstellen.**

2-Umreifungen entlang der B-Kante  
des Formatmaßes  
Maß – A



### 1.3.3 Packing Specifications

The packing specifications are stored as ASCII files in NC-Daten. The extension of the file is 'vpg'. All packing specifications are in one directory.

(→ [directory and data structure](#)).

The packing specifications are used in three different places:

- For parts lists the packing specifications are stored in a file that has the name of the parts list. The extension is 'vpg'. Within these files the sequence of the data records can be chosen at will. Data records with the same designation are combined in a block. In the [format-stacking line](#) the corresponding packing specification of the format is stored.
- For data bases a unique data structure is valid. The internal structure of the data records and data record blocks is the same as for parts list-specific packing specifications. The file extension is 'vdb', for each packing specification and unit of measurement a separate file is created.
- For schedule-NC data the subsequent-work data records of one type are combined in blocks. Within these blocks the sequence of data records can be chosen at will.

When entering the format in a parts list then from a list of available packing instructions can one be selected.

#### 1.3.3.1 Internal structure of packing-instruction files

file name: ExampleVorschrift.vpg		
m_W01	<a href="#">packing-instruction base line</a>	exactly one line per packing instruction
m_W10	<a href="#">packing-instruction text line 1</a>	0 or 1 line per packing instruction
until		
m_W19	<a href="#">packing-instruction text line 10</a>	0 or 1 line per packing instruction
	etc.	
m_W\$	<a href="#">packing-instruction end line</a>	exactly one line per packing instruction

#### 1.3.3.2 Packing-instruction data base:

file name: DBVerpackungsVorschrift[id,if,mm].vdb		
m_W02	<a href="#">packing-instruction base line</a>	exactly one line per packing-instruction data-base file
m_W01	<a href="#">packing-instruction text line 1</a>	exactly one line per packing-instruction data-base file
m_W10	<a href="#">packing-instruction text line 2</a>	0 or 1 line per packing-instruction data-base file
until		
M_W19	<a href="#">packing-instruction text line 10</a>	0 or 1 line per packing-instruction data-base file
M_W\$	<a href="#">packing-instruction end line</a>	exactly one line per packing-instruction data-base file

### 1.3.3.3 Packing-instruction base line (W 01)

This is the first line in the packing instruction. In the data base of packing instructions only one packing-instruction base line may exist. In parts list-specific packing instructions any number of W 01 data lines are allowed.

The names of the data fields are not only referring to protection boards but can also be used for pallets.

field	description	l/a	pos	attribute	data type
-	class name			m_W01	CNcW01
1	designation W for packing-instruction line	1A	1		
2	designation 01 for base line	2N	3		
3	Name	20A	6	m_Name	char[ ]
4	description	20A	27	m_Beschreibung	char[ ]
5	code of protection board top	3N	48	m_SplObenCode	int
6	code of protection board bottom	3N	52	m_SplUntenCode	int
7	alignment of protection board top edge A	1N	56	m_SplObenAusrA	int
8	alignment of protection board top edge B	1N	58	m_SplObenAusrB	int
9	alignment of protection board bottom edge A	1N	60	m_SplUntenAusrA	int
10	alignment of protection board bottom edge B	1N	62	m_SplUntenAusrB	int
11	group name of protection board top	20A	64	m_SplObenGruppe	char[ ]
12	number of protection board top	3N	85	m_SplObenNr	int
13	group name of protection board bottom	20A	89	m_SplUntenGruppe	char[ ]
14	number of protection board bottom	3N	110	m_SplUntenNr	int
15	minimum protection-board protrusion A	8N	114	m_SplUeberstandA	int
16	maximum stack pattern protrusion A	8N	123	m_SplBildÜberstandA	int
17	minimum protection-board protrusion B	8N	132	m_SplUeberstandB	int
18	maximum stack pattern protrusion B	8N	141	m_SplBildÜberstandB	int
	CR LF		149		

**3... Name:** The name is the same as the file name. If importing from optimizations of other manufacturers a different name can be specified which shall be used by the optimization. The name must be unique i.e. different packing specifications must also have different names.

(→ [Name conventions](#))

**4... Description:** This is an additional text for the packing instruction. The text is not used for the distinction of packing instructions. Can be used therefore several times as the same text. No data, only spaces, is also permitted.

**5,6... Code of protection board top / code of protection board bottom:** default value = 00. These data fields are only taken into consideration when no standard protection boards are used. In that case data fields 11 to 13 must be empty. The stacking optimization then calculates the necessary protection board size. These data fields specify by which rules the protection boards are selected. For the selection of the protection-board size the following possibilities exist:

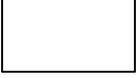
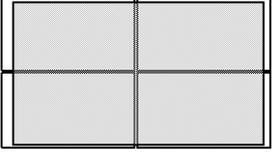
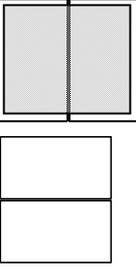
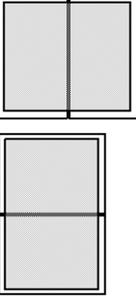
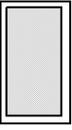
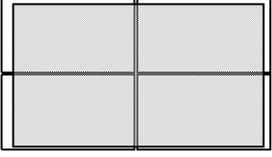
000	Only for Optimization: Code for protection board (pallet) top/bottom will be calculated according the stacking optimization parameter.
009	No protection boards for the stack of the format are to be used if the stack pattern is 1x1. If for a format the protection-board code is 009 and a stack pattern unequal to 1x1 is specified then the format cannot be stacked.

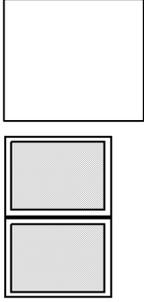
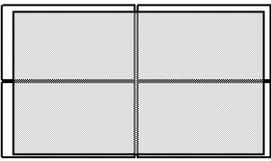
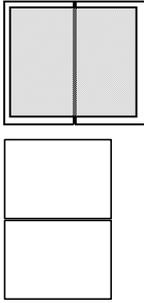
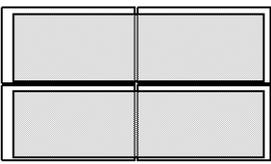
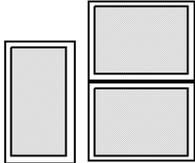
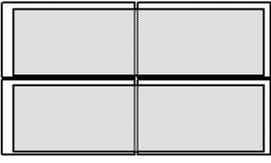


..... protection board



..... format stack

010	Exactly one protection board per each stack of the format.		
011	Exactly one protection board for each stack of the format that has a stack pattern unequal to 1x1. If the stack pattern is 1x1 then no protection board is used.		
012	Exactly one protection board for each stack of the format if the amount of columns at the scissors lift is greater than 1. The designation 'columns' does not relate to the stack pattern but to the scissors lift.		
013	As 012, in addition a protection board is used if the width of the stack on the scissors lift is bigger than its length. Also for a stack pattern with 1x1 a protection board is used.		
020	Exactly one protection board per line if the stack does not have more than one column. If the stack has more than one column a shared protection board is used.		

021	As 020, but no protection board if the stack pattern is 1x1.		
022	Exactly one protection board per line if the column amount is greater than 1 on the scissors lift. If the columns equal 1 then no protection board is used.		
023	As 022, in addition a protection board is used if the width of the stack on the scissors lift is bigger than its length. Also for a stack pattern with 1x1 a protection board will be used.		
030	A protection board per line and per column of the stack pattern. Caution: If the stacking facility cannot keep the individual columns together then several stacks are created.		
060	Concurrent cutting of the necessary protection boards top/bottom for each stack of the format (refer to the corresponding S&S optimization parameter)		

7,8,9,10... Alignment of protection board top / protection board bottom: default value = 0. Here the desired alignment of the protection board can be specified for the stacked formats.

0= no values specified.

1= flush at stop.

2= flush opposite to stop.

3= centered to stack.

Example: alignment protection board top edge A = 3. The protection board shall be stacked centered to the format in the direction of edge A but the protection board can be off center in relation to edge B of the format.

11,13... Group name of protection board top / group name protection board bottom: default value = [20 spaces]. In this data field is specified which protection-board group shall be used for this packing instruction. If a valid protection-board group is entered then a protection board from the specified protection-board group will be selected.

Refer to format-stacking line (F 02)

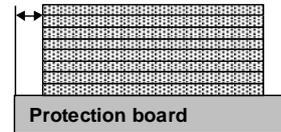
Data field 8,9,10,11...minimum / maximum amount of format dimension A/B in the stack pattern.

12,14... Number of protection board top / number of protection board bottom: default value = [0].  
 With these data fields is specified which protection board from the specified group is to be used for this packing instruction. If a valid protection-board number is entered here then the specified protection board will be used from the specified protection-board group. A protection-board group can be comprised of several protection boards. This data field is only taken into consideration if the data field 'group name protection board' contains a valid protection-board group.

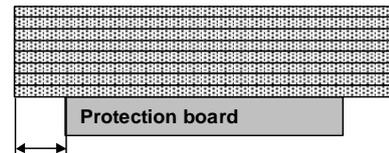
0: Each protection board from the specified protection-board group may be used.

1-999: Exactly the protection board of the specified number is to be used. If no protection board of the specified name exists then any protection board from the group may be used. If the specified protection board does not fit a fixed and predetermined stack pattern then also any protection board from the stated protection-board group may be used.

15,17... Minimum protection-board protrusion A / B: default value = 0.  
 This data field specifies how far minimally the protection board has to protrude over the stacked formats. If in this field a value different than 0 is specified then the data field 'maximum stack pattern protrusion' is ignored.



16,18... maximum stack pattern protrusion A / B: default value = 0.  
 In this field can be specified how much the stacked format may protrude maximally over the protection board.



### 1.3.3.4 Packing-instruction text line (W 10-19)

These lines contain the text for the creation of packing labels. These texts can be printed onto packing labels according to these packing instructions. The lines are optional. If such a line exists then it must follow the packing-instruction base line. As a maximum 10 such lines per each packing instruction are allowed.

field	description	l/a	pos	attribute	data type
-	class name			m_W10 until m_W19	CStrArray
1	designation W for packing-instruction line	1A	1		
2	designation 10-19 for text line	2N	3		
3	customer specific text CR LF	20A	6 26	m_Text	char[ ]

### 1.3.3.5 Packing-instruction data-base base line (W 02)

This line is used for the entry of general information of packing-instruction data lines in the packing-instruction data base.

field	description	l/a	pos	attribute	data type
-	class name			m_W02	CNcW02
1	designation W for packing-instruction line	1A	1		
2	designation 02 for data-base line	2N	3		
3	unit of measurement CR LF	1N	6 7	m_MassSystem	Int

3... Unit of measurement: For each unit of measurement its own packing-instruction data base is created. In this data field the unit of measurement is entered. (→ [convention for units of measurements](#))

### 1.3.3.6 Packing-instruction end line (W\$)

This line is used for the recognition of the correct data end and is only used for packing-instruction files. (→ [directory and file structure](#))

field	description	l/a	pos	attribute	data type
1	designation W\$ for packing-instruction end line CR LF	2A	1 3		

### 1.3.4 Strapping instruction

The strapping instructions are stored in ASCII files in NC-Daten. The extension of the file is 'umr'. All strapping instructions are in the same directory.

(→ [directory and file structure](#)).

#### 1.3.4.1 Internal structure of strapping-instruction files

file name: ExampleVorschrift.umr		
m_U01	strapping-instruction base line	exactly one line per packing instruction
m_U\$	strapping-instruction end line	exactly one line per packing instruction

#### 1.3.4.2 Strapping-instruction base line (U 01)

This is the first line of the strapping instruction. Only one strapping-instruction base line may exist in the file.

field	description	l/a	pos	attribute	data type
-	class name			m_U01	CNcU01
1	designation U for strapping-instruction line	1A	1		
2	designation 01 for base line	2N	3		
3	name	20A	6	m_Name	char[ ]
4	description	20A	27	m_Beschreibung	char[ ]
5	amount of straps	2N	48	m_AnzUmreifungen	int
6	put on edge protection	1N	51	m_KantenSchutz	int
7	thickness of squared timber	8N	53	m_KantholzDicke	int
8	groove squared timber	1N	62	m_KantholzNuten	int
9	amount of squared timber	2N	64	m_AnzKanthoelzer	int
10	bit pattern of squared timber	16A	67	m_KantholzBmp	char[ ]
11	thickness of top strip	8N	84	m_LeistenODicke	int
12	groove top strip	1N	93	m_LeistenONuten	int
13	amount of top strips	2N	95	m_AnzLeistenOben	int
14	bit pattern for top strip	16A	98	m_LeistenOBmp	char[ ]
15	bit pattern for top strips also strapped	16A	115	m_LeistenOUmrBmp	char[ ]
16	thickness of side strips	8N	132	m_LeistenSDicke	int
17	groove side strips	1N	141	m_LeistenSNuten	int
18	amount of side strips	2N	143	m_AnzLeistenS	int
19	bit pattern of side strips	16A	146	m_LeistenSBmp	char[ ]
20	amount of straps depending on strip number	1N	163	m_AnzUmrStrAbh	int
	CR LF				

3... Name: The name is the same as the file name. If importing data from external optimizations then a different file name can be specified which shall be used subsequently. The name must be unique, i.e. different strapping instructions must have different names. (→ [convention for names](#))

4... Description: This is an additional text for strapping instructions. The text is not used for the distinction of strapping instructions. Can be used therefore several times as the same text. No data, only spaces, is also permitted

5... Amount of straps: default value = 99. This is the amount of strapping in the respective direction. If the field contains the value "X" then the value from the stacking parameter is used.

6... Put on edge protection: default value =0. An edge protection is an edge piece that shall protect the stack from damage through the straps at the edges.

0= no edge protection desired.

1= edge protection desired.

7... Thickness of squared timber: Here the desired thickness of the squared timber is entered. The squared timber is put under the stack and strapped with the stack. If a stack is not stacked onto a palette then squared timber is necessary in order to transport the stack with a fork lift.

(→ [convention for units of measurement](#))

8... Groove squared timber: default value =0. Sometimes it is necessary to groove the squared timber in order to apply the strapper.

0= no groove desired

1= groove desired

9... Amount of squared timber: default value = 99. This is the amount of squared timber in the respective direction. If the data field contains the value "X" then the value from the stacking parameter is used.

10,15... Bit pattern for squared timber / top strips also strapped:

The bit pattern specifies at which strapping positions squared timber/top strips are desired. ??The amount of squared timber, amount of top strips results from the amount of high bits in the bit pattern.

Example: bit pattern for squared timber = 0000000000001010, therefore a squared timber at the 2nd and 4th strapping position is desired.

14... Bit pattern for top strips: The bit pattern specifies for which strapping top strips are to be used, the strips that will actually be strapped is specified in data field 18.

11,16... Thickness of strips on top / side: Here the desired thickness of the top and side strips is specified. The strips are put on top or onto the side of the stack. They are strapped with the stack. The strips protect the stack from the strapping.

(→ [convention for units of measurement](#))

12,17... Groove top strips / groove side strips: default value =0. Sometimes it is desired that the side strips or top strips have a groove where the strapping band can be inlaid.

0= no groove desired.

1= groove desired.

13... Amount of top strips: This data field specifies the amount of top strips in the respective direction. If the data field contains the value X then the value from the stacking parameter will be used.

18... Amount of side strips: This data field specifies the amount of top strips in the respective direction. If the data field contains the value X then the value from the stacking parameter will be used.

19... Bit pattern for side strips: The bit pattern specifies at which positions the side mounted strips are desired. The amount of strips results from the amount of high bits in the bit pattern.

Example: bit pattern for side strips = 0000000000001010 a side strip is desired at strapping position 2 and position 4.

20... Amount of strapping depending on strip number:

0= amount of strapping in data field must be complied with.

1= if the amount of strapping in data field 5 is less than the strip amount in the respective direction then each strip is strapped separately, i.e. the strapping amount will be increased. If the specified strip amount is bigger than the amount of strips then the specification of the data field remains unchanged.???

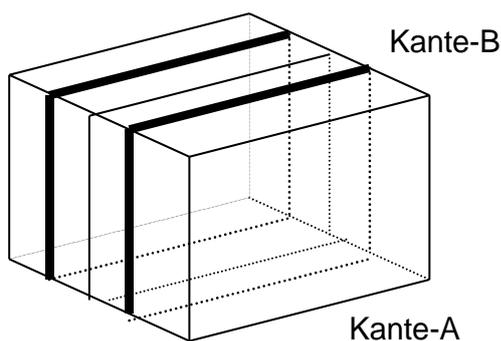
Example:

amount of desired strapping = 1

strip number along edge B = 2

Anzahl der gewünschten Umreifungen = 1

Streifenanzahl entlang der B-Kante = 2



Kante B = edge B

Kante A = edge A

### 1.3.4.3 Strapping-instruction data-base base line (U 02)

This line is used for the specification of general information for the strapping data lines in the strapping data base.

field	description	l/a	pos	attribute	data type
-	class name			m_U02	CNcU02
1	designation U for strapping line	1A	1		
2	designation 02 for data base line	2N	3		
3	unit of measurement	1N	6	m_MassSystem	Int
	CR LF		7		

### 1.3.4.4 Strapping-instruction end line (U\$)

This line is used for the recognition of a proper file end and is only used in strapping-instruction files. (→ [directory and file structure](#))

field	description	l/a	pos	attribute	data type
1	designation U\$ for strapping-instruction end line	2A	1		
	CR LF		3		

## 1.3.5 Strapping rules

### 1.3.5.1 Internal structure of the strapping-rules files

file name: example Umreifungsregel.urr		
m_U11	strapping-rule header line	strapping-rule
m_U12	strapping-rule base line	exactly one line per strapping rule
m_U12	strapping-rule base line	exactly one line per strapping rule
m_U12	strapping-rule base line	exactly one line per strapping rule
m_U12	strapping-rule base line	exactly one line per strapping rule
m_U\$	strapping-rule end line	exactly one line at the end of the file

The data records of the strapping rules are stored in an ASCII file. The extension of the file is 'urr'. The strapping-rule files are in the 'Umreifungsregeln' directory.

(→ [directory and file structure](#)). Within a protection-board data base is only one unit of measurement permitted.

### 1.3.5.2 Strapping-rule header line (U 11)

This line is used for the specification of general information for the strapping-rule line.

field	description	l/a	pos	attribute	data type
-	class name			m_X02	CNcX02
1	designation U for protection-board line	1A	1		
2	designation 11 for strapping-rule header line	2N	3		
3	unit of measurement	1N	6	m_MassSystem	int
	CR LF		7		

3... Unit of measurement: For all protection-board data-base files the same unit of measurement is valid. In this data field the unit of measurement is specified.

(→ [convention for units of measurement](#))

### 1.3.5.3 Strapping-rule base line (U 12)

These lines describe the strapping rules. The rules are applied to the formats and stack patterns at the creation of the NC files. The results of the strapping instructions are then stored in the [strapping-instruction format](#). 1 to n descriptive lines per strapping instruction are permitted.

field	description	l/a	pos	attribute	data type
-	class name			m_U03	CNcU03
1	designation U for strapping-instruction line	1A	1		
2	designation 12 for strapping rule	2N	3		
3	name of strapping rule	20A	6		char[]
4	description	20A	27	m_Beschreibung	char[ ]
5	apply rule until dimension	8N	48	m_Mass	int
6	name of strapping instruction for 1x1	20A	57	m_Umr_1x1	char[]
7	name of strapping-instruction for nxn	20A	57	m_Umr_nxn	char[]

CR LF

3... Name of strapping rule: This data field contains the name of the customer specific strapping rule.

4... Description: This is an additional text for the strapping rule. The text is not used for the distinction of strapping rules. Can be used therefore several times as the same text. No data, only spaces, is also permitted.

5... Apply rule until dimension: Until this dimension the specified strapping rule is to be applied. If several strapping rules exist for different dimensions then the strapping instruction is used that has the least difference between the dimension of the stack and the dimension specified here.

6... Strapping instruction for 1x1: This [strapping instruction](#) is used for stacks with a stack pattern of 1x1.

7... Strapping instruction for nxn: This [strapping instruction](#) is used for stacks with a stack pattern unequal to 1x1.

### 1.3.5.4 Strapping-rule end line (U\$)

This line is used to recognize the correct file end. It is only used in protection board files.

(→ [directory and file structure](#))

field	description	l/a	pos	attribute	data type
1	designation U\$ for stacking-rules end line	2N	1		
	CR LF	1A	3		

### 1.3.6 Protection-board data records

The protection board data records are stored in ASCII files in NC-Daten. The extension is 'spt'. The protection-board files are in the directory 'Schonplatten'.

(→ [directory and file structure](#)).

#### 1.3.6.1 Internal structure of protection-board files

Parts-list specific protection boards: for each parts list a separate file is created. The file has the name of the parts list and has the extension of 'spt'.

file name: Name der Stückliste.spt		
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X\$	protection-boards end line	exactly one line per packing instruction

Protection-boards data base: for each group a separate file is created. The file has the name of the group and the extension 'sdb'.

file name: Example Schonplatte.sdb		
m_X02	protection-boards data-base base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X01	protection-boards base line	exactly one line per packing instruction
m_X\$	protection-boards end line	exactly one line per packing instruction

### 1.3.6.2 Protection-boards base line (X 01)

For each protection board exactly one protection-board base line has to exist. The names of the data fields only relate to protection boards. But all data fields are valid, both for protection boards and also for palettes.

field	description	l/a	pos	attribute	data type
-	class name			m_X01	CNcX01
1	designation X for protection-board line	1A	1		
2	designation 01 for base line	2N	3		
3	group	20A	6	m_Gruppe	char[ ]
4	number	3N	27	m_Nummer	char[ ]
5	dimension A	8N	31	m_MassA	int
6	dimension B	8N	40	m_MassB	int
7	thickness	8N	49	m_Dicke	int
8	description	20A	58	m_Beschreibung	char[ ]
9	orientation	1N	79	m_Orientierung	int
10	internally used protection-board number.	4N	81	m_IntNr	int
	CR LF		85		

3... Group: default value =[20 spaces]. This data field contains the designation of the group to which the protection board belongs to. (→ [convention for names](#))

4... number: default value =1. This data field contains the number of the protection board within the group.

5,6,7... Dimension A / dimension B / thickness: This data field contains the dimension of the protection board.

(→ [convention for units of measurement](#))

8... Description: In this data field an additional description of the protection board can be entered.

9... Orientation: In data field can be specified how the protection boards is to be placed on the scissors lift.

- 0,2 protection board length parallel to scissors lift length.
- 1,3 protection board width parallel to scissors lift length.
- 9 any direction, the protection board can be placed in any direction on the scissors table.  
Refer to orientation (→ [convention for orientation](#)).

10... Internally used protection-board number: This data field contains the internal protection board number which must be unique within the schedule file \*.ncl. In the protection board pattern only one protection board number may be entered. However a standard protection board has a combination key which consists of group and number. In order to prevent information loss a unique number is required.

For the parts list import this number has no importance and can be set to 0.

### 1.3.6.3 Protection-board data-base base line (X 02)

The data records of the protection-board data base are stored in ASCII files. The extension of the file is 'sdb'. The protection board files are in the directory 'SchonplattenDB'. (→ [directory and file structure](#)). Within a protection-board data base only one unit of measurement may be used.

This line is used for the specification of general information of the protection-board line in the protection-board data base.

field	description	l/a	pos	attribute	data type
-	class name			m_X02	CNcX02
1	designation X for protection-board line	1A	1		
2	designation 02 for data-base base line	2N	3		
3	unit of measurement	1N	6	m_MassSystem	int
	CR LF		7		

3... Unit of measurement: For all protection boards in a protection-board data base file the same unit of measurement is valid. In this data field the unit of measurement is specified. (→ [convention for units of measurement](#))

### 1.3.6.4 Protection-board end line (X\$)

This line is used for the recognition of a correct file end and is only used in protection-board files.

(→ [directory and file structure](#))

field	description	l/a	pos	attribute	data type
1	designation X\$ for protection-board end line	2N	1		
	CR LF	1A	3		