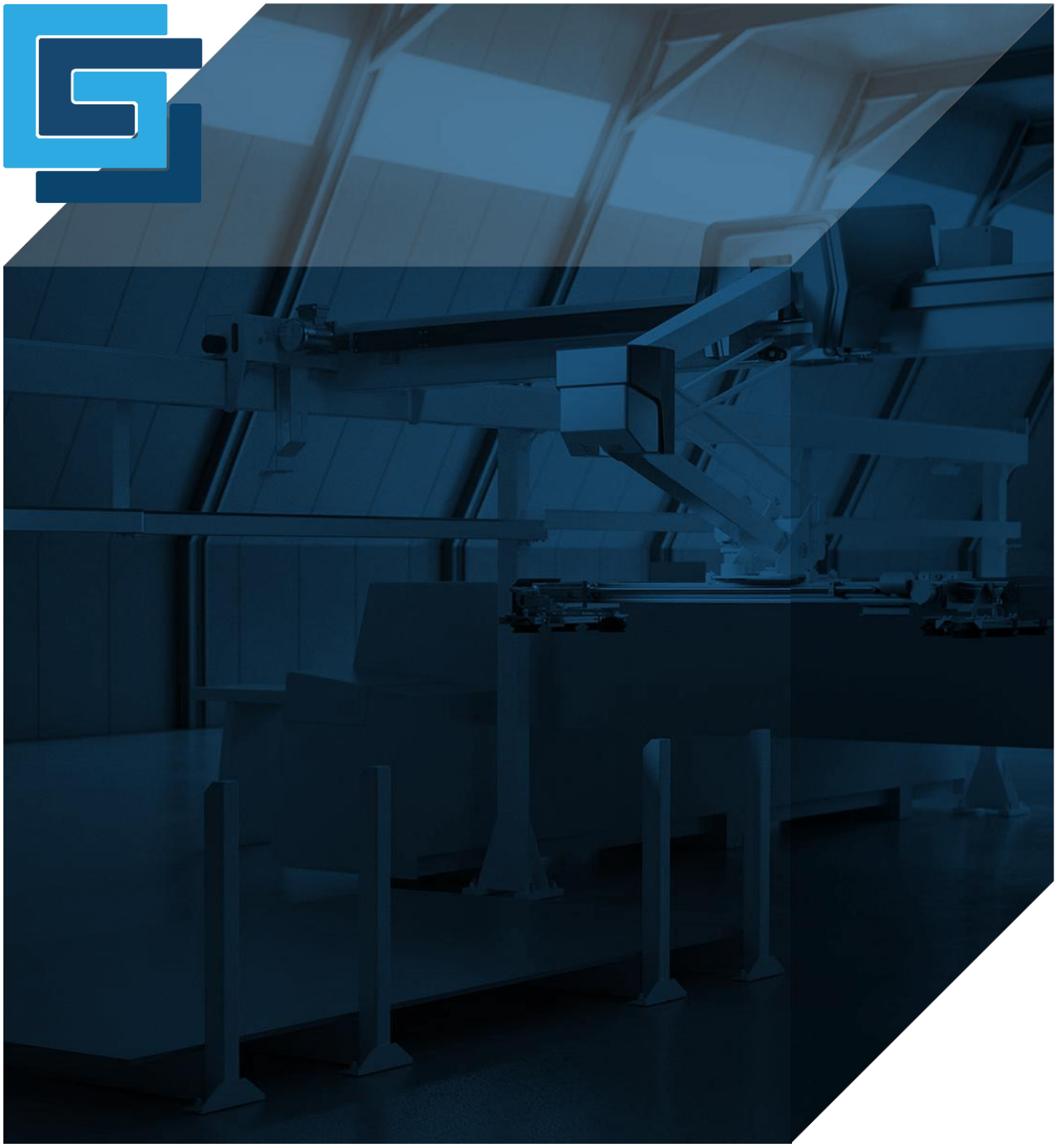


Grundner gateway communication

V3.6

Describes the basic communication possibilities of the Grundner storage system with external software systems.

04.07.2025

Index register

Changelog

Version	Date	Changed by	Description
1.4	07.04.2015	TB	Added data range for column description
			Replaced images for !t and !T and description
1.5	19.05.2015	FC	Changes at labelling data (name of CSV)
			Added command “<P” delete production list
			Added command “>N” change machine number
1.6	22.02.2016	FC	Command “<P” changed. More parameters to delete specific orders
			Added report files “successful unloading of board to machine”
1.7	08.07.2016	JG	Unload, stacking and load back to stock
	21.07.2016	DH	Added “Type” to Unload, stacking and load back to stock
1.8	19.04.2016	FC	Added report file for Unloading to output place
			Command ?M /add new value “rest piece”
1.8	04.05.2016	SH	Changed Software from Windows XP to Windows 7
2.0	08.09.2017	FC	**Updated to Lasal Standard**
			Added “Load back commissioning list.”
2.1	19.10.2017	FC	Added confirmation file for 1.5 and 1.6 “creating types”
			Added value for available stock in 1.1 “stock request”
			Added command change commissioning place number.1.13
			Added communication “stock control” 1.14
			Added report 2.4 “Report of stack at output conveyor”
2.2	10.01.2018	FC	Added command 1.15 “Get Etage info”
			Added report 2.5 “Report of deleted production order (to the saw)”
			Added report 2.6 “Report of deleted commissioning order”
			Added report 2.7 “Report of deleted input order “
			Added sample file names to each command or report
	07.08.2018	FC	Changed command 1.1 from “?M” to “?m”
2.3	19.12.2018	FC	Corrected command 1.6 ->wrong data string, removing “type” from data string
			New command 1.1 “no trigger file necessary”
			+new parameter “reserved stock” / Etage 5-6 / Customer ID
			Added parameter at command 1.11 “additional info”
2.4	28.03.2019	FC	Added command “delete rest” 1.16
	10.09.2019	FC	Added value for “stock manual rest storage” & “stock block storage” to 1.1 “stock r”
	05.11.2019	FC	Added value “density” at stock request 1.1
2.5	12.11.2019	FC	Added value “material Number” & “CustomerID” at reports 2.1, 2.2, 2.3, 2.4
			Added value “CustomerID” at 1.5 “Create a new type.”
2.6	25.02.2020	Chr. Eichinger	Added missing communication commands 1.16; 2.1-2.7
			Added gateway settings; Added strapping mode to 1.3
		FC	Command for 2.5 -2.7 & 1.15 changed
		FC	Added batch Nr. To input command 1.7
		FC	Added batch Nr. To input report 2.3
	24.11.2020	FC	Added field JobAddition, RunInfo and CustomerID to Report 2.1 /2.2 /
			Added field JobAddition and RunInfo to 1.2 /1.3
2.7	09.03.2021	JP	Added value “source manipulator” to 2.2
2.8	23.06.2021	DM	Added missing communication command 1.17 (input via transport line)
	25.06.2021	DM	Added 1.18 “Work preparing list”
2.9	07.01.2022	FH	Added 1.18.1 “AVListOrder”
	18.01.2022	FC	Correction at input command 1.7 and 1.17 / Missing parameter “Res” after “Etage”

!!!		FC	Protocol changed at command 1.17 / switched “pos number” & “number of input place”
	27.01.2022	FH	1.18.1 & 1.18.2 an Gateway.ini adjusted
3.0	15.09.2022	GM	Minor corrections
3.1	13.02.2023	GM	Added value “batch” to 2.1
3.2	20.02.2023	GM	Added report 2.8 “Report a full nestpick unstack” Added report 2.9 “Report a nestpick unstack”
3.3	08.03.2023	GM	Added field etage and CustomerID to 1.6
	02.05.2023	FC	Max.4 strapping positions (before 5 pos.) in 1.3
	19.07.2023	GM	Added priority information to 1.3
3.4	19.09.2023	FC	Changed name of report 2.8 Added command “Unload, stack and restore…” 1.11 Changed content of 1.11 “Unload, stack and restore…” destination changed to destination place / no additional functionality. Element “part category” added
	01.03.2024	GM	Added fields output date, output time, and storage date to 2.2 Modified thickness to DInt (0-65535)
3.5	06.03.2025	FC	1.11 ElementNr changed from UDINT to String
3.6	04.07.2025	KM	Changed 1.15 GetEtageNr to GetFloorNo & ConfirmEtageNr to ConfirmFloorNo Parameter tables created, new example pictures Changed cover sheet, back sheet Added JobAddition to 2.4 Changed ElementNr to Element ID Changed pieces to quantity Changed piece to part/board Changed etage to floor Added Table for Size of ASCII Parameters
	08.07.2025	MLE	Added at 1.11. a new “Part category” with value 5… outsourcing part

1. Communication with external software

1. The warehouse PC has to be linked to the network by ethernet connection (Microsoft Windows ® is installed on the warehouse PC)
2. The communication with other software is based on a files exchange, so Windows must be able to get access to shared folders from the PC which is communicating with the storage system. Standard file format is csv and data fields are semicolon (";"), comma (",") or tab (0x09) separated. So do not use these characters in text fields. A specific header in the files is never used. The content of the files is written in ASCII; character set is ANSI / Windows 1252.
3. The name of the different files and the location can be specified directly in the trigger file which is initiating the communication.
4. There are different types of communication which are possible to initiate.

Chapter	Command	Short description
1.1	?m → !c	ask for stock information
1.2	>P	order a board to the saw
1.3	!P	order a board to an output place
1.4	>R	send back a rest board
1.5	!T → !T	establish a new type of board with a fixed type number
1.6	!t → !T	establish a new type of board; type number is given automatically by the storage and will be in the range of the rest types
1.7	>E	for loading boards into the storage
1.8		Labelling machine communication
1.9	<P → !p	Delete the production list at the storage
1.10	>O	Change machine number
1.11	>S	Unload, stack and restore pieces to stock
1.12	<U → !u	Delete the commissioning list at the storage
1.13	>N	Change commissioning place number
1.14	!b	Stock control message
1.15	?F → !BestEtage	Get etage info
1.16	<R	Delete rest orders
1.17	TE	Loading boards into the storage via a transport input line
1.18	>V	Loading an order into work preparing list
2.1	!P / !Q	Report of a successful unloading of a board to a machine
2.2	!F	Report of a successful unloading of a board to an output place
2.3	!G	Report of a successful loading of a board (input)
2.4	!Y	Report of a stack which is on the conveyor line
2.5	!DelProd	Report of a deleted production order (to the saw)
2.6	!DelComm	Report of a deleted commissioning order
2.7	!DelLoad	Report of a deleted input order
2.8	!EntladenFertig	Report a full nestpick unstack
2.9	!AbstapelnTeil	Report a nestpick unstack

Strategies for communication:

Communication files with the specified filenames should only be present in the communication folder if they are fully described and not open anywhere.

For example, if a file named „order_saw.csv“ is created in the communication folder, the warehouse will immediately read this file. If the parameters are not yet entered or are incomplete, this could cause errors or incorrect commands.

For example, if a file named „order_saw.csv“ is created in the communication folder, the warehouse will immediately read this file. If the parameters are not yet entered or are incomplete, this could cause errors or incorrect commands.

To ensure trouble-free communication with the warehouse, there are two strategies:

1. **Create the file first as „.tmp“ and then change it to „.csv“ when finished**
   If you initially name the file „order_saw.tmp“ the warehouse does not recognize it as a communication file yet. This gives the operator or software time to fill in the file and enter the desired parameters. Once the file is complete, simply change the extension to „.csv“ and the warehouse will recognize it as a communication file and read it.

2. **Only insert the fully completed file into the communication folder**
   In this approach, the file be created immediately as „order saw.csv“. However, this must not happen directly in the communication folder. Instead, the operator or software has time to fill in the file with the desired parameters first. Once finished, the file can be moved into the communication folder and the warehouse can immediately work with the completed file.

Size of ASCII Parameters

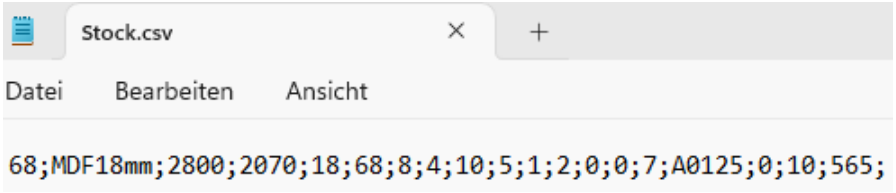
Name of parameter	Size
material name	50 characters
customer ID	14 characters
job-no	8 characters
JobAddition	4 characters
RunInfo	1 character
Element ID	14 characters

1.1 Ask for stock information from the Grundner storage system

1. Create a “request file” in the communication folder (stock_request.csv). To get stock information for all board types, simply write 0 into the file. If you need only data from specific material, add the material numbers in separate lines into this file. The file can have more than one line.
2. The end line of the data file must be “!E”.

3. As soon as the file appears in the communication folder, the storage will read this file. The storage will fill a separate file with all the available information. During filling the stock data file, the file name will be a temporary one. After all data is written, the expected filename will appear and present the available file!

The content of the output data file will look like this:



No.	Parameter	Description
1	**type**	Type number for a specific board (numeric 1- 4095)
2	**material name**	A name which fits to the type number (is determined in type data) is automatically shown. (ASCII 50 characters)
3	**length**	Panel length [mm]
4	**width**	Panel width [mm]
5	**thickness**	Panel thickness [mm]
6	**material number**	Material number for company internal administration (0 - 4294967295)
7	**stock**	Current quantity in stock
8	**stock av.**	Available stock number (Stock – production list) / no negative values
9	**stock1**	Current quantity in area1 (reserve)
10	**stock2**	Current quantity in area2 (reserve)
11	**stock3**	Current quantity in area3 (reserve)
12	**stock4**	Current quantity in area4 (reserve)
13	**stock5**	Current quantity in area5 (reserve)
14	**stock6**	Current quantity in area6 (reserve)
15	**reserved stock**	Number of boards, that has already been ordered at the storage
16	**customer ID**	A name which fits to the type number (is determined in type data) (ASCII 14 characters)
17	**stock manual rests**	Current quantity in area of manual rests (if activated)
18	**stock of block storage**	Current quantity in area of block storage (reserve)
19	**density**	Density of the material in kg/m³

All panels from the part list are noticed and added together for each level of each storage. Panels on the input place can be added, in the stock book, according to a parameter in the warehouse, when their data have already been entered in the input menu.

Sample filenames:
(Customer software) (?m)
stock_request.csv

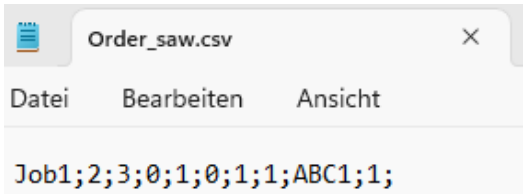
(Storage software) (!c)
Stock.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\stock_request.csv
CmdCodeX = ?m
EndungVonX = .csv
EndungZuX = .erl

[LogFile]
CmdCodeX = !c
DateinameX = C:\Grundner\temp\Stock.csv
ZugriffX = e

1.2 Order a board to the saw

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (order_saw.TMP).



No.	Parameter	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**type**	Type number for a specific board (numeric 1 – 4095)
3	**quantity**	Desired quantity (number for 1 delivery) (1 – 255)
4	**rotation**	Rotation of the panel to the saw 0…not rotated 1…90° rotated
5	**machine number**	Insert the number of the saw, to which the panel should be brought (1-4)
6	**priority**	0 = no priority 1 = priority → data insert to the first line
7	**Info**	0 – 65535 can be used for 2nd info of job-no
8	**label**	0 = the board goes directly to the machine (saw) 1 = the board must be brought to the label station first, before it comes to the machine (saw)
9	**JobAddition**	Text area can be used for additional job information (ASCII 4 characters)
10	**RunInfo**	Reserved for communication with HOMAG machines. Otherwise, free to use. (ASCII 1 character)

2. After filling the file with the necessary data, rename it to the assigned file name (order_saw.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
Order_saw.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\order_saw.csv
CmdCodeX = >P
EndungVonX = .csv
EndungZuX = .erl

1.3 Order a board to a specific output place

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (output_commissioning.TMP). For strapping you can choose up to 4 strapping positions for one package! The distance starts from the largest panel in the stack!

No.	Parameter	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**type**	Type number for a specific board (numeric 1 – 4095)
3	**quantity**	Desired quantity (1 – 255)
4	**number of output place**	Insert the number of the output place, to which the panel should be brought (0 – 255) 0…just prepare this board(s) 99…PLC calculates best output place
5	**Info**	Number for free use (0 – 65535)
6	**priority**	0 = no priority 1…100 = priority low to high (numeric 0 – 100)
7	**Strapping mode**	0 = no straps 1 = package stops in front of strapping machine for manual strapping 2 = package stops in front of strapping machine, after confirmation, the strapping starts automatically 3 = packages automatically enter the strapping machine and are strapped
8	**parameters for strapping mode**	How many straps will be applied in this string (max. 4 Parameters) 100… Determine straps based on the default value table 101-110… Use straps program 1-10
	parameter 1	Position of strap 1 (0 – 65535) [mm]
	parameter 2	Position of strap 2 (0 – 65535) [mm]
	parameter 3	Position of strap 3 (0 – 65535) [mm]
	parameter 4	Position of strap 4 (0 – 65535) [mm]
	JobAddition	Text area that can be used for additional job information (ASCII 4 characters)
	RunInfo	Reserved for communication with HOMAG machines. Otherwise, free to use. (ASCII 1 character)

The parameter No. 8 “number of parameters for strapping mode” also defines the number of parameters that exists between “number of parameters for strapping” and “JobAddition”. For example, if the number of parameters for strapping mode is 4, then “parameter 1”, “parameter 2”, “parameter 3” and “parameter 4” exist. But if “number of parameters for strapping mode” is 0, then “parameter 1” – “parameter 4” does not exist and “JobAddition” is the next parameter after “number of parameters for strapping mode”.

“Number of parameters for strapping mode” = 0

“Number of parameters for strapping mode” = 4

2. After filling the file with the necessary data, rename it to the assigned file name (output_commissioning.csv). We prefer to use the file ending **.csv**.

Remark to column “priority”:
Each output place has its own priority queue. Priority can be 0 (no priority), 1 (low priority) to maximum 100 (highest priority).
If the priority for this output place already exists, the line to be imported will be appended below the same priority. If the priority for this output place does not exist, the line to be imported is inserted above the next lower priority.

Sample filenames:
(Customer software)
Output_commissioning.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\Output_commissioning.csv
CmdCodeX = !P
EndungVonX = .csv
EndungZuX = .erl

1.4 Send back a rest board to the storage

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (NewRest.TMP).

No.	Parameter	Description
1	**type**	Type number for a specific board (1- 4095)
2	**quantity**	Quantity of rest parts (0 – 255)
3	**number of rest input place**	Insert the number of the rest place, from where the panel should be inserted (0-255)
4	**length**	Here you can enter the length [mm] of the rest part (0 – 65535)
5	**width**	Here you can enter the width [mm] of the rest part (0 – 65535) (if you have an automatic rest bring in system, the storage compares length and width with the measured values and will add them if they are not already entered)

2. After filling the file with the necessary data, rename it to the assigned file name (rest.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
NewRest.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\NewRest.csv
CmdCodeX = >R
EndungVonX = .csv
EndungZuX = .erl

1.5 Create a new type of board in the storage data base

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (GrundnerType.TMP).

No.	Parameter	Description
1	**type**	Type number for a specific board (1- 4095)
2	**material name**	Name of the created material (ASCII 50 characters)
3	**material number**	Number of the material (important for rests) (0 – 4294967295)
4	**length**	Here you can enter the length [mm] of the new board (0 – 65535)
5	**width**	Here you can enter the width [mm] of the new board (0 – 65535)
6	**thickness**	Here you can enter the thickness [mm] of the new board (0 – 65535)
7	**group**	Group number in the storage (to keep same group on specified places) (0 – 9)
8	**density**	Density of the material [kg/m³] (0 – 65535)
9	**minimum stock**	A value for a minimum stock. It's possible to print a list of missing boards according to this minimum stock (0 – 65535)
10	**floor**	Target floor/gripper for this type (0 – 255) When this type is entered in the "input menue" to load this board into the storage, this "floor" is suggested by the system!
11	**customer ID**	A name which fits to the type number. Additional key for connection to external software (ASCII 14 characters)

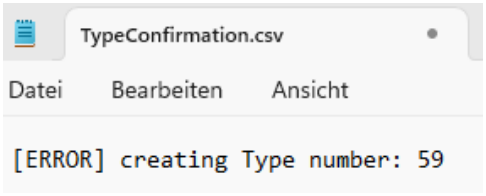
2. After filling the file with the necessary data, rename it to the assigned file name (GrundnerType.csv). We prefer to use the file ending **.csv**.

Confirmation to this request (option):

It is possible to activate a confirmation for this process in the storage system. If this option is activated, the system creates a file which contains separate information.

If the storage system has imported the data successfully, you will receive the same data back as a confirmation, like:

If the storage system is not able to import the data correctly, you will receive the information [ERROR] + additional information:



Sample filenames:
(Customer software)
GrundnerType.csv

(Storage software)
TypeConfirmation.csv

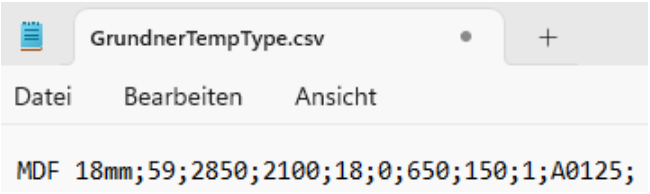
Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\GrundnerType.csv
CmdCodeX = !T
EndungVonX = .csv
EndungZuX = .erl

[LogFile]
CmdCodeX = !T
DateinameX = C:\Grundner\temp\TypeConfirmation.csv
ZugriffX = n

1.6 Create a new type of board in the storage data base and type number is given automatically by Grundner

Basically, this communication is used for creating rest boards.
If you want to create a new type of board and don't care about the type number you can use this kind of communication! The system will use the next free type number in a temporary area. After the type has a stock of 0 Boards, the type will be **deleted automatically** after an adjustable time.

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (GrundnerTempType.TMP).



No.	Parameter	Description
1	**material name**	Name of the created material (ASCII 50 characters)
2	**material number**	Number of the material (important for rests) (0 – 4294967295)
3	**length**	Here you can enter the length [mm] of the new board (0 – 65535)
4	**width**	Here you can enter the width [mm] of the new board (0 – 65535)
5	**thickness**	Here you can enter the thickness [mm] of the new board (0 – 65535)
6	**group**	Group number in the storage (to keep same group on specific places) (0 – 9)
7	**density**	Density of the material [kg/m³] (0 – 65535)
8	**minimum stock**	A value for a minimum stock. It's possible to print a list of missing boards according to this minimum stock (0 – 65535)
9	**floor**	The number of the floor, in which the boards can be loaded (0 – 255)
10	**customer ID**	Additional key for connection to external software (ASCII 14 characters)

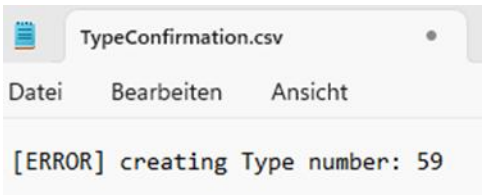
2. After filling the file with the necessary data, rename it to the assigned file name (GrundnerTempType.csv). We prefer to use the file ending **.csv**.

Confirmation to this request (option):

It is possible to activate a confirmation for this process in the storage system. If this option is activated, the system creates a file which contains separate information.

If the storage system has imported the data successfully, you will receive the same data back as a confirmation, like:

If the storage system is not able to import the data correctly, you will receive the information [ERROR] + additional information:



Sample filenames:
(Customer software)
GrundnerTempType.csv

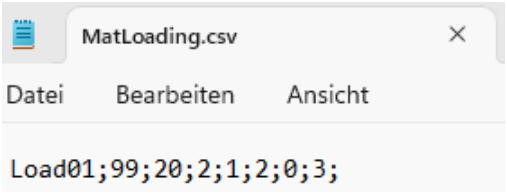
(Storage software)
TypeConfirmation.csv
This file has the same name as in command 1.5

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\GrundnerTempType.csv
CmdCodeX = !t
EndungVonX = .csv
EndungZuX = .erl

[LogFile]
CmdCodeX = !T
DateinameX = C:\Grundner\temp\TypeConfirmation.csv
ZugriffX = n

1.7 Add a line for an input command

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (MatLoading.TMP).



No.	Parameter	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**type**	Type number for a specific board (0 – 4095)
3	**quantity**	Quantity on the input place (0 – 255)
4	**number of input place**	Insert the number of the input place, from which the panel should be taken (0 – 255)
5	**InfoNo**	Number for free use (0 – 65535) (e.g. position of a parts list)
6	**floor**	Insert the number of the storage floor (level), to which the panel should be brought (0 – 255)
7	**res**	Reserve for parameter / no function (always 0)
8	**batch**	Batch number of the stack on the input station (0 – 65535)

2. After filling the file with the necessary data, rename it to the assigned file name (MatLoading.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
MatLoading.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\MatLoading.csv
CmdCodeX = >E
EndungVonX = .csv
EndungZuX = .erl

1.8 Communication to label printer

There are two possible ways to establish a communication to the label machine:

Version1:

The printer on the machine is controlled by an external software.

Following data must be provided:

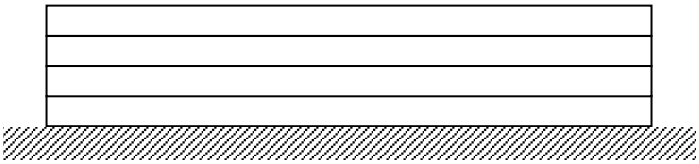
- Label positions on the board
- Order for the correct board

Label positions on the board:

For the positions of the labels on the board, a file must be created in the communication folder (e.g. “C:\Grundner\temp”). For the first board, the filename is exactly the name of the order + .CSV. Is it necessary to cut a stack of boards together, the name of the first board stays the same, but every further board has an extension of the board number added to the order name (e.g.: order-2.CSV)

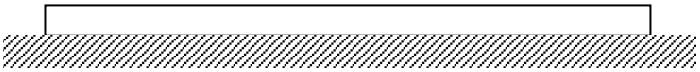
e.g.: Labelling a stack of 4 boards:

4th board...Order-4.csv
3rd board...Order-3.csv
2nd board...Order-2.csv
1st board ...Order.csv



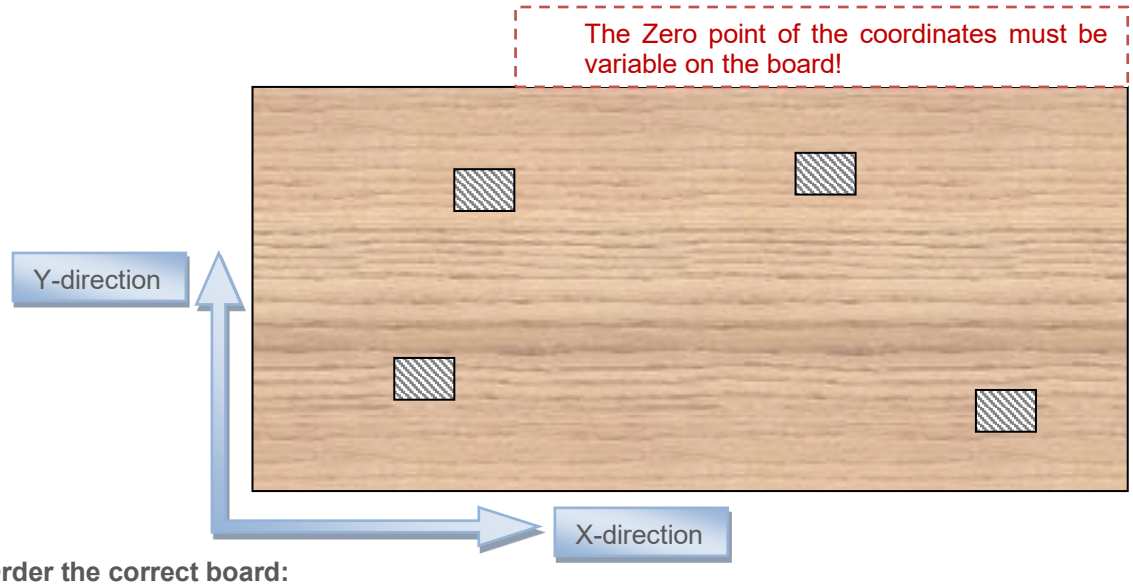
e.g.: Labeling a single board:

1st board ...Order.csv



The content of the CSV file must look like this:

No.	Parameter	Description
1	**X-pos**	X distance from reference point to the label (0 – 4294967295) [mm]
2	**Y-pos**	Y distance from reference point to the label (0 – 4294967295) [mm]
3	**res**	reserve (not in use, always 0)
4	**Element ID**	Identification of the board to find the matching label (ASCII 14 characters)



Order the correct board:

You can find the information at Order a board to the saw.
It is important to use the job number as your order name and set labelling to 1. When you send an order to the machine, you have to be sure, that you have already sent the label bitmaps and the label positions.

Workflow for Version 1:

After writing the necessary files, the panel storage is going to load the label machine with the ordered board. As soon as a board arrives on the label machine, it searches for the position file (Order.csv) and loads the positions for all labels into an internal table.

The label machine is now going to clean and measure the board. When the label machine has reached the end of the board it will send an order file to print the first label for the board! The file will always be named for example “label.txt” and the content looks like this:

No.	Parameter	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**InfoNo (cutting plan)**	Free to use as internal identifier (e.g. part lists)
3	**rotation**	rotation of the label (0 – 360) [°] (1 = 90°)
4	**Element ID**	Identification of the board to find the matching label (ASCII 14 characters)

With this information the labels can be printed. The machine will check by itself, if the labels came out fine and are positioned correctly on the stamp, before pressing the label on the board! As soon as the stamp is going down, the next print file is created in the communication folder.

Version 2:

The printer on the machine is controlled by the warehouse:

Following data must be provided:

- Label bitmaps
- Label positions on the board
- Order for the correct board

Label bitmaps:

The external software has to provide the correct bitmap files in a specific folder (e.g. “C:\Grundner\Etiketten”). The label can be created as .bmp , .jpg , .pdf . The naming of the labels is like this:
“**Order_33.bmp**”
Order Name of the order (max. 8 alpha numeric signs)
_33 Element number (necessary for label positions max. 65535)

Label positions on the board:

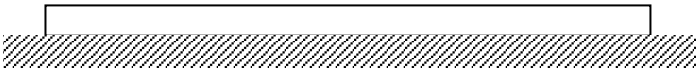
For the positions of the labels on the board, a file must be created in the communication folder (e.g. “C:\Grundner\temp”). For the first board, the filename is exactly the name of the order + .csv. Is it necessary to cut a stack of boards together, the name of the first board stays the same, but every further board has an extension of the board number added to the order name (e.g.: order-2.csv)

e.g.: Labeling a stack of 4 boards:

4th board...Order-4.csv
3rd board...Order-3.csv
2nd board...Order-2.csv
1st board ...Order.csv

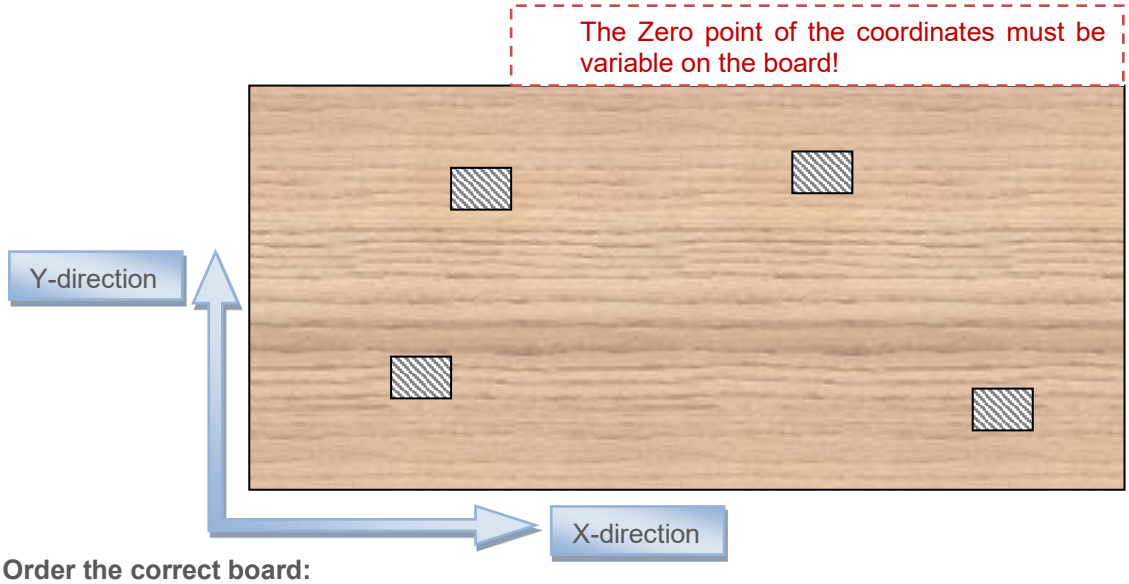
e.g.: Labeling a single board:

1st board ...Order.csv



The content of the CSV file must look like this:

No.	Parameter	Description
1	**X-pos**	X distance from reference point to the label (0 – 4294967295) [mm]
2	**Y-pos**	Y distance from reference point to the label (0 – 4294967295) [mm]
3	**res**	reserve (not in use, always 0)
4	**Element ID**	Identification of the board to find the matching label (ASCII 14 characters)



Order the correct board:

You can find the information at Order a board to the saw.
It is important to use the job number as your order name and set “label” to 1. When you send an order to the machine, you have to be sure, that you have already sent the label bitmaps and the label positions.

Workflow for Version 2:

After writing the necessary files, the panel storage is going to load the label machine with the ordered board. As soon as a board arrives on the label machine, it searches for the position file (Order.csv) and loads the positions for all labels into an internal table.
The label machine is now going to clean and measure the board. When the label machine has reached the end of the board it will immediately start to print the JPG file with the correct order name and element number (Order_12.JPG). If the correct file is not in the folder for the labels, the machine will go into error state! The machine will check by itself, if the labels came out fine and are positioned correctly on the stamp, before pressing the label on the board! As soon as the stamp is going down, the next JPG file is sent to the printer in the communication folder!

1.9 Load back (delete) the production list for the saw

First of all, it is necessary to define the name of the "data file" and the "answer file" (productionLIST_DEL.csv) in the configuration of the Grundner software. It is possible to load back the whole production list or specific orders from the storage. When the request is sent, the warehouse deletes all chosen orders from the production list and writes the deleted orders into an "answer file". If the gripper has already started to manipulate a board to the saw, this order will not be deleted in the production list of the storage and also not mentioned in the "answer file". The storage will create the "answer file" and as soon as the communication is finished. The file can be taken as soon as it exists.

1. Create a "temporary data file" (get_production.TMP) in the communication folder which contains the information of what should be done! Follow next step for the content of the file.

2. There are 3 different ways (2 parameters) to delete orders from the production list.

No.	Parameter	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**machine number**	This is the number of the machine (0 – 255)

a. **Delete the whole production list:** The file content has to be 0, which means to get the whole production list!

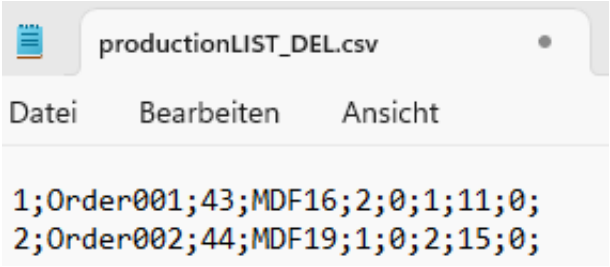
b. **Delete a specific order:** The file content has to be the name of the order which should be deleted on the first position of the csv format.

c. **Delete all orders for a specific machine:** The file content has to be the number of the machine on the second position of the csv format.

3. After creating the “temporary data file” with the correct content, rename it to the correct “data file name”.

4. The Grundner software takes this data file and creates an “answer file” according to the requested data.

5. You can take the file “answer file” (productionLIST_DEL.csv) as soon as it exists. The file will contain the deleted orders.



No.	Parameter	Description
1	**line no.**	Number of the line from the production list (1 – 255)
2	**job-no**	Text area for order designation (ASCII 8 characters)
3	**type**	Type number for a specific board (numeric 1 – 4095)
4	**material name**	A name which fits to the type number (is determined in type data)
5	**quantity**	Quantity of the boards
6	**rotation**	Rotation of the board
7	**machine**	This is the number of the machine (0 – 255)
8	**source place**	This is the number of the place, where the board is found
9	**res**	This value can be ignored

Sample filenames:
(Customer software)
get_production.csv

(Storage software)
productionLIST_DEL.csv

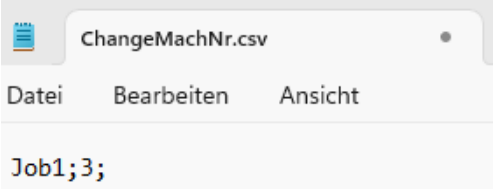
Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\get_production.csv
CmdCodeX = <P
EndungVonX = .csv
EndungZuX = .erl

[LogFile]
CmdCodeX = !p
DateinameX = C:\Grundner\temp\productionLIST_DEL.csv
ZugriffX = e

1.10 Change a machine number in the production list

This command enables to change the machine number from any order in the production list. This can be helpful to first order all the orders to machine number 0 and change the machine number later in the progress.

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (ChangeMachNr.TMP).



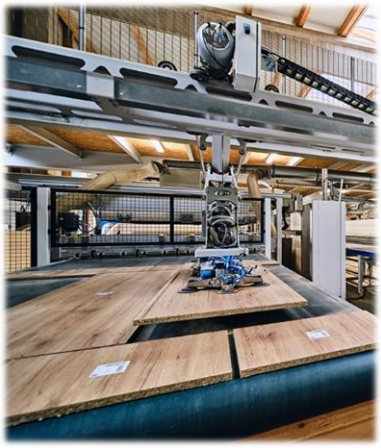
No.	**Parameter**	Description
1	**job-no**	Text area to identify the necessary order, which should be changed (ASCII 8 characters)
2	**new machine number**	Here you write the new machine number for the mentioned order (0 – 4)

2. After filling the file with the necessary data, rename it to the assigned file name (ChangeMachNr.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
ChangeMachNr.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\ChangeMachNr.csv
CmdCodeX = >O
EndungVonX = .csv
EndungZuX = .erl

1.11 Nestpick – Unload pieces from machine



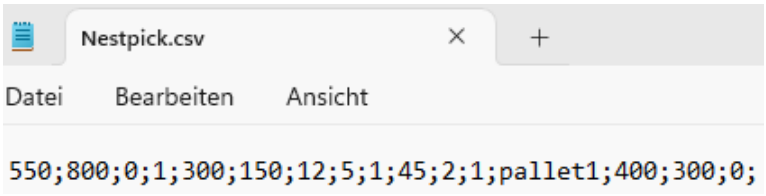
After loading a board to a cutting machine (nesting), the board is cut into small parts. To pick up these parts, some additional datasets are required. It is important to receive only the data for the nest, that is ready on the machine. The complete nest should be in one file. If data have been sent for more than one nest, the system cannot recognize, which are ready to pick and which one will be there after next cut. Each section must be sent in a separate line which contains the related data. It is possible to choose different destinations for each part.

It is possible to choose a destination place directly by submitting the correct number of the destacking place of the machine, or just keep parts together in a virtual pallet. After adding different parts together in one virtual pallet, it is necessary to connect this virtual pallet to a stacking place of the machine. This could be done automatically by adding specific rules or the user can connect them. The method for assigning destinations must be fixed in the system clarification.

This data must be provided:

- Source position of boards (offset from reference of panel)
- Source machine
- Element ID
- Size of boards
- Destination

The content of the csv file must look like this (example):



No.	Parameter	Description
1	**X-pos source**	source position of the part: X distance from reference point to the centre of the panel (0 – 4294967295) [mm]
2	**Y-pos source**	source position of the part: Y distance from reference point to the centre of the panel (0 – 4294967295) [mm]
3	**res**	reserve
4	**Element ID**	Identification of the part (ASCII 14 characters)
5	**Length**	length of the piece in x-direction (0 – 65535) [mm]
6	**Width**	width of the piece in y-direction (0 – 65535) [mm]
7	**Thickness**	height of the piece in z-direction (0 – 65535) [mm]
8	**Destination place**	number of destacking or outsourcing place (1 – 16) depends on layout of the storage and on the part category
9	**Source machine**	number of the machine (1 – 4) from which the piece has to be picked up
10	**Type**	type number from Grundner of the raw material
11	**Additional info**	Information number (0 – 65535)
12	**Part category**	1…destacking part (will be put on the destacking place) 2…Rest (goes back into the storage with new Type No.) 3…option (e.g. waste could be removed with different gripper system) 5…outsourcing part (will be put on the outsourcing place)
13	**Destination pallet name**	name of a virtual pallet, where the parts will be stacked together / 12-digit text area to identify pallet
14	**X-pos target**	target position of the part: X distance from reference point to the centre of the panel (0 – 4294967295) [mm]
15	**Y-pos target**	target position of the part: Y distance from reference point to the centre of the panel (0 – 4294967295) [mm]
16	**R-pos target**	rotation of the part on the target position 0/90/180/270 [°]

Workflow:

As soon as the file appears in the folder which is configured, it will be imported into the list for stacking parts. According to the priorities and necessary signals the job will start.

Sample filenames:
(Customer software)

Nestpick.csv

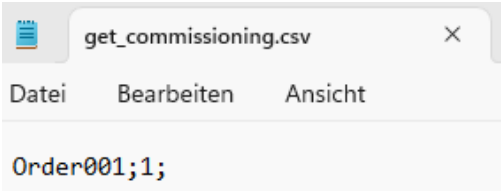
Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\Nestpick.csv
CmdCodeX = >S
EndungVonX = .csv
EndungZuX = .erl

1.12 Load back (delete) the commissioning list for an output place

First of all, it is necessary to define the name of the "data file" and the "answer file" (commissioningLIST_DEL.csv) in the configuration of the Grundner software.
It is possible to load back the whole commissioning list or specific orders from the storage. When the request is sent, the warehouse deletes all chosen orders from the production list and writes the deleted orders into an "answer file". All order with an acceptable output place number (1-16) will not be deleted in the commissioning list of the storage and also not mentioned in the "answer file".
The storage will create the "answer file" and as soon as the communication is finished. The file can be taken as soon as it exists.

1. Create a "temporary data file" (get_commissioning.TMP) in the communication folder which contains the information of what should be done! Follow next step for the content of the file.

2. There are 3 different ways (2 parameters) to delete orders from the production list.



No.	**Parameter**	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**machine number**	This is the number of the machine (0 – 255)

a. **Delete the whole commissioning list:** The file content has to be 0, which means to get the whole production list!

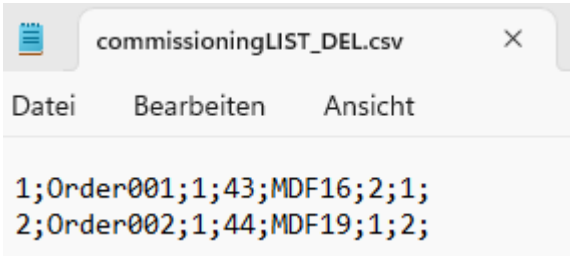
b. **Delete a specific order:** The file content has to be the name of the order which should be deleted on the first position of the csv format.

c. **Delete all orders for a specific output place:** The file content has to be the number of the output place on the second position of the csv format.

3. After creating the “temporary data file” with the correct content, rename it to the correct “data file name”.

4. The Grundner software takes this data file and creates an “answer file” according to the requested data.

5. You can take the file “answer file” (commissioningLIST_DEL.csv) as soon as it exists. The file will contain the deleted orders…



No.	Parameter	Description
1	**line no.**	Number of the line from the production list (1 – 255)
2	**job-no**	Text area for order designation (ASCII 8 characters)
3	**InfoNo**	Number for free use (0 – 65535) (e.g. position of a parts list)
4	**type**	Type number for a specific board (numeric 1 – 4095)
5	**material name**	A name which fits to the type number (is determined in type data) (ASCII 50 characters)
6	**quantity**	Quantity of the boards (0 – 255)
7	**Output place**	This is the number of the machine (0 – 255)

Sample filenames:
(Customer software)
get_commissioning.csv

(Storage software)
commissioningLIST_DEL.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\get_commissioning.csv
CmdCodeX = <U
EndungVonX = .csv
EndungZuX = .erl

[LogFile]
CmdCodeX = !u
DateinameX = C:\Grundner\temp\commissioningLIST_DEL.csv
ZugriffX = e

1.13 Change an output place number in the commissioning list

This command enables to change the number of an output place from any order in the commissioning list. This can be helpful to first order all the orders to place number 0 and change the output place number later in the progress.

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (ChangeCommNr.TMP).

No.	**Parameter**	Description
1	**job-no**	Area to identify the necessary order, which should be changed (ASCII 8 characters)
2	**new output place number**	Here you write the new output place number for the mentioned order (0 – 255)

2. After filling the file with the necessary data, rename it to the assigned file name (ChangeCommNr.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
ChangeCommNr.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\ChangeCommNr.csv
CmdCodeX = >N
EndungVonX = .csv
EndungZuX = .erl

1.14 Read list “stock control”

If this option is activated (in the Grundner machine control), the system produces a list of material which is missing or needed to reach a minimum stock level! A file will be written into a certain folder with the content defined below.

No.	Parameter	Description
1	**type**	Type number for a specific board (numeric 1 – 4095)
2	**material name**	A name which fits to the type number (is determined in type data) is automatically shown (ASCII 50 characters)
3	**length**	Panel length [mm] (0 – 65535)
4	**width**	Panel width [mm] (0 – 65535)
5	**thickness**	Panel thickness [mm] (0 – 65535)
6	**material number**	Material number for company internal administration (0 - 4294967295)
7	**stock**	Current quantity in stock (0 – 65535)
8	**minimum stock**	The minimum stock value for this type of board, which is set in the type data (0 – 65535)
9	**stock av.**	Available stock (stock – production list) / no negative stock! (0 – 65535)
10	**panel supply to warehouse**	Quantity to load into the storage, to reach necessary stock (stock – production list, compared with minimum stock) (0 – 65535)

Sample filenames:
(Storage software)
StockControl.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !b
DateinameX = C:\Grundner\temp\StockControl.csv
ZugriffX = a

1.15 Get floor number

With this command you can ask the storage for a target etage if you send a board type number. If you want the system to choose the target etage for a special type, use this command. If the storage system is not able to find a “etage number” for this type, there will be a “0” at the target etage.

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the request (GetFloorNo.TMP).

No.	Parameter	Description
1	**type**	Type number for a specific board (numeric 1 – 4095)
2	**quantity**	Desired quantity (1 – 255)

2. After filling the file with the necessary data, rename it to the assigned file name (GetFloorNo.csv). We prefer to use the file ending **.csv**.

Confirmation to this request:

The storage takes the information and is going to choose the best etage for this type of board!

No.	Parameter	Description
1	**type**	Type number for a specific board (numeric 1 – 4095)
2	**quantity**	Desired quantity (1 – 255)
3	**target floor**	The number of the floor, in which the boards can be loaded (0 – 255)

If the storage system is not able to import the data correctly, or not able to find a good etage you will receive the information “Target floor” = 0

Sample filenames:
(Customer software)
GetFloorNo.csv

(Storage software)
ConfirmFloorNo.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\GetFloorNo.csv
CmdCodeX = ?F
EndungVonX = .csv
EndungZuX = .erl

[LogFile]
CmdCodeX = !BestEtage
DateinameX = C:\Grundner\temp\ConfirmFloorNo.csv
ZugriffX = n

1.16 Delete rest orders

With this command you can delete rest orders at the storage “rest list”. It is possible to delete the complete rest orders for one machine (number of the rest input place).

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the request (DeleteRest.TMP).

No.	**Parameter**	Description
1	**no. of rest input place**	Insert the number of the rest place, for which the orders should be deleted

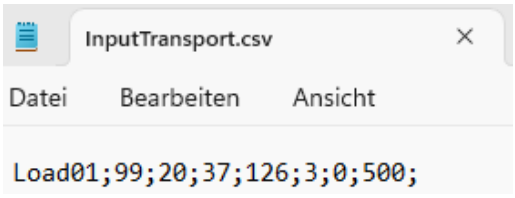
2. After filling the file with the necessary data, rename it to the assigned file name (DeleteRest.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
DeleteRest.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\DeleteRest.csv
CmdCodeX = <R
EndungVonX = .csv
EndungZuX = .erl

1.17 Add a line for an input command via transport

1. Create a temporary data file in the communication folder where the external software must have filled in the necessary data for the order (InputTransport.TMP).



No.	Parameter	Description
1	**job-no**	Text area for order designation (ASCII 8 characters)
2	**type**	Type number for a specific board (0 – 4095)
3	**quantity**	Quantity on the input place (0 – 255)
4	**number of transport place**	Number of transport place where the panel is positioned (0 – 255)
5	**InfoNo**	Number for free use (0 – 65535) (e.g. position of a parts list)
6	**floor**	Insert the number of the storage floor (level), to which the panel should be brought (0 – 255)
7	**res**	Reserve for parameter / no function (always 0)
8	**batch**	Batch number of the stack on the input station (0 – 65535)

2. After filling the file with the necessary data, rename it to the assigned file name (InputTransport.csv). We prefer to use the file ending **.csv**.

Sample filenames:
(Customer software)
InputTransport.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\temp\InputTransport.csv
CmdCodeX = TE
EndungVonX = .csv
EndungZuX = .erl

1.18 Loading an order into work preparing list

1.18.1 V1 Order to saw list

Columns and description like 1.2 Order a board to the saw

Sample filenames:
(Customer software)
PreparationOrder_Saw.csv

Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\orderAVList.csv
CmdCodeX = >V;1
EndungVonX = .csv
EndungZuX = .erl

1.18.2 V2 Order to output preparing list

Columns and description like 1.3 Order a board to a specific output place

Sample filenames:
(Customer software)
PreparationOrder_OutputPlace.csv

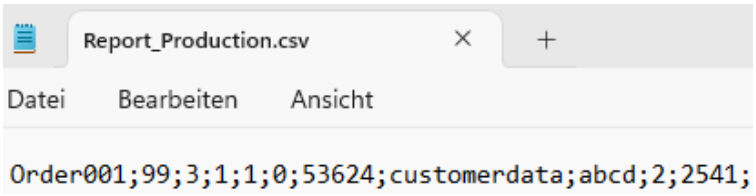
Gateway settings (Gateway.ini):
[Senden]
SendeDateinameX = C:\Grundner\Output_preparing.csv
CmdCodeX = >V;2
EndungVonX = .csv
EndungZuX = .erl

2.0 Report Files

To use these files for reading, writing or deleting, it is necessary to first rename the file and afterwards work with the renamed file. With this working procedure you can ensure not to interfere with the filling procedure from the warehouse system.

2.1 Report of a successful unloading of a board to a machine

It's possible to define a file, to which the system reports a successful delivery to a machine. Each line will contain the information for one unloading operation. (An unloading operation can contain a package of 2 or more boards) As soon as the machine releases the last board of the unloading order, the machine puts a line of data into this file. The system adds the lines at the bottom of the file. The data format looks like this:



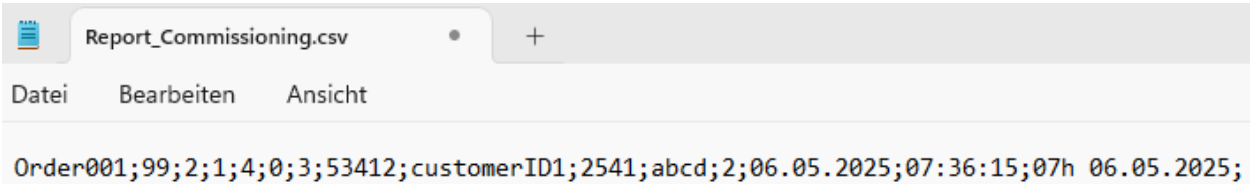
No.	Parameter	Description
1	**job-no**	Here you will receive job number of the delivered board (ASCII 8 characters)
2	**type**	Type number for the delivered board (numeric 1 – 4095)
3	**quantity**	Delivered number of boards (How many boards will be cut at the same time) (0 – 255)
4	**Info**	Here you will receive the "Info" from the unloading job, that has been given when sending jobs to the storage (0 – 65535)
5	**number of machine**	Number of the machine, to which the board is brought to (0 – 255)
6	**finished**	This value can be ignored
7	**material number**	Number of the material (important for rests) (0 – 4294967295)
8	**customer ID**	A name which fits to the type number (is determined in type data) Additional key for connection to external software (ASCII 14 characters)
9	**JobAddition**	Text area that can be used for additional job information (ASCII 4 characters)
10	**RunInfo**	Reserved for communication with HOMAG machines. Otherwise, free to use. (ASCII 1 character)
11	**batch**	Batch number of the storage on the input station (0 – 65535)

Sample filenames:
(Storage software)
Report_Production.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !P / !Q
DateinameX = C:\Grundner\temp\Report_Production.csv
ZugriffX = a

2.2 Report of a successful unloading of a board to an output place

It's possible to define a file, to which the system reports a successful delivery to an output place. Each line will contain the information for one board. As soon as the machine releases the last board of the unloading order, the machine puts a line of data into this file. The system adds the lines at the bottom of the file. The data format looks like this:



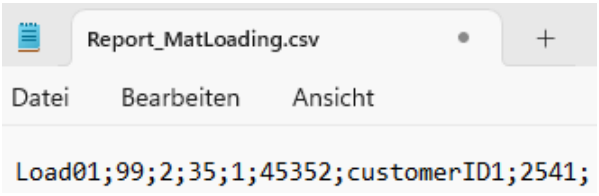
No.	Parameter	Description
1	**job-no**	Here you will receive job number of the delivered board (ASCII 8 characters)
2	**type**	Type number for the delivered board (1 – 4095)
3	**quantity**	Delivered number of boards (0 – 255)
4	**Info**	Here you will receive the "info" from the unloading job, that has been given when sending jobs to the storage (0 – 65535)
5	**number of output place**	Number of the output place, to which the board was brought to (0 – 255)
6	**finished**	0…indicates that, there are still boards to unload for this job-no. 1…indicates the last board of this job-no.
7	**source manipulator**	This number gives the manipulator which brought the panel
8	**material number**	Number of the material (important for rests) (0 – 4294967295)
9	**customer ID**	A name which fits to the type number (is determined in type data) Additional key for connection to external software (ASCII 14 characters)
10	**batch**	Batch number of the stack on the input station (0 – 65535)
11	**JobAddition**	4-digit text area can be used for additional job information
12	**RunInfo**	Reserved for communication with HOMAG machines. Otherwise, free to use. (ASCII 1 character)
13	**output date**	Date of board delivery (10 digit date, format DD.MM.YYYY)
14	**output time**	Time of board delivery (8 digit time, format hh:mm:ss)
15	**storage date**	Format xxh DD.MM.YYYY, example: 08h 30.04.2025

Sample filenames:
(Storage software)
Report_Commissioning.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !F
DateinameX = C:\Grundner\temp\Report_Commissioning.csv
ZugriffX = a

2.3 Report of a successful loading of a board (input)

It's possible to define a file, to which the system reports a successful loading of a board. Each line will contain the information for one loading operation (every board is separate line)
As soon as the machine picks up a board from the input station and has checked it successfully, the machine puts a line of data into this file. The system adds the lines at the bottom of the row. The data format looks like this example:



No.	Parameter	Description
1	**job-no**	Here you will receive job number of the loaded board (ASCII 8 characters)
2	**type**	Type number of the loaded board (1 – 4095)
3	**quantity**	Quantity of moved boards
4	**InfoNo**	Here you will receive the "InfoNo" from the loading job (0 – 65535)
5	**number of input place**	Number of loading place from the storage, from which the board has been picked up by the storage (0 – 255)
5	**finished**	0 = There are still boards left, of this order, to load into the storage 1 = The input order like it was sent, is finished. Indicates the last board of this job-no
6	**material number**	Number of the material (important for rests) (0 – 4294967295)
7	**customer ID**	A name which fits to the type number (is determined in type data) Additional key for connection to external software (ASCII 14 characters)
8	**batch**	Batch number of the stack on the input station (0 – 65535)

Sample filenames:
(Storage software)
Report_MatLoading.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !G
DateinameX = C:\Grundner\temp\Report_MatLoading.csv
ZugriffX = a

2.4 Report of a stack which is ready on the conveyor line

It's possible to define a file, to which the system reports a successful output of a stack of boards. As soon as the stack has reached the end of the conveyor transport line, the system creates a file with the content of this stack!

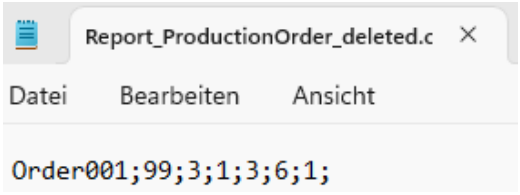
No.	Parameter	Description
1	**job-no**	Here you will receive job number of the unloaded board (ASCII 8 characters)
2	**type**	Type number of the loaded board (1 – 4095)
3	**quantity**	Quantity of moved boards (0 – 255)
4	**InfoNo**	Here you will receive the "InfoNo" from the loading job (0 – 65535)
5	**number of output station**	Number of the output place, to which the board was brought to (0 – 255)
6	**res**	reserve (0)
7	**material number**	Number of the material (important for rests) (0 – 4294967295)
8	**customer ID**	A name which fits to the type number (is determined in type data) Additional key for connection to external software (ASCII 14 characters)
9	**JobAddition**	Additional job information (ASCII 4 characters)

Sample filenames:
(Storage software)
Report_ConveyorTransport.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !Y
DateinameX = C:\Grundner\temp\Report_ConveyorTransport.csv
ZugriffX = e

2.5 Report of a deleted production order (to the saw)

It's possible to define a file, to which the system reports deleted production orders. The content of the file will look like described below. There can be more than one line in this file.
New information will just be added by the storage.



No.	Parameter	Description
1	**job-no**	Here you will receive job number of the unloaded board (ASCII 8 characters)
2	**type**	Type number of the ordered board (1 – 4095)
3	**quantity**	Ordered number of boards (0 – 255)
4	**rotation**	Rotation of the panel on the saw 0…not rotated 1…90° rotated
5	**machine number**	The number of the saw, to which the panel should be brought (1 – 4)
6	**Info**	Here you will receive the "info" from the unloading job, that has been given when sending jobs to the storage (0 – 65535)
7	**label**	0 = the board goes directly to the machine (saw) 1 = the board must be brought to the label station first, before it comes to the machine (saw)

Sample filenames:
(Storage software)
Report_ProductionOrder_deleted.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !DelProd
DateinameX = C:\Grundner\temp\Report_ProductionOrder_deleted.csv
ZugriffX = a

2.6 Report of a deleted commissioning order

It's possible to define a file, to which the system reports deleted commissioning orders. The content of the file will look like described below. There can be more than one line in this file.
New information will just be added by the storage.

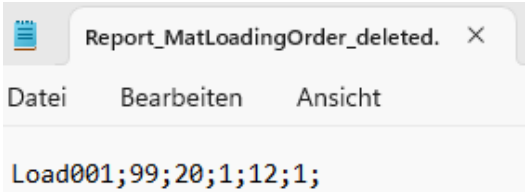
No.	Parameter	Description
1	**job-no**	Here you will receive job number of the unloaded board (ASCII 8 characters)
2	**type**	Type number of the ordered board (1 – 4095)
3	**quantity**	Ordered number of boards (0 – 255)
4	**number of output place**	Insert the number of the output place, to which the panel should be brought (0 – 255)
5	**Info**	Here you will receive the "info" from the unloading job, that has been given when sending jobs to the storage (0 – 65535)

Sample filenames:
(Storage software)
Report_CommissioningOrder_deleted.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !DelComm
DateinameX = C:\Grundner\temp\Report_CommissioningOrder_deleted.csv
ZugriffX = a

2.7 Report of a deleted input order

It's possible to define a file, to which the system reports deleted input orders (loading material into the storage). The content of the file will look like described below. There can be more than one line in this file. New information will just be added by the storage.



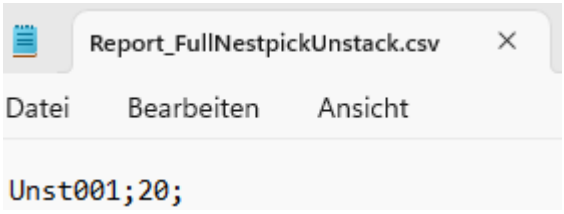
No.	Parameter	Description
1	**job-no**	Here you will receive job number of the unloaded board (ASCII 8 characters)
2	**type**	Type number of the board (1 – 4095)
3	**quantity**	Number of boards, that should be loaded (0 – 255)
4	**number of input place**	The number of the input place, from which the panel should be loaded into the storage (0 – 255)
5	**InfoNo**	Number for free use (0 – 65535) (e.g. position of a parts list)
6	**floor**	The number of the storage floor (level), to which the panel should be brought (0 – 255)

Sample filenames:
(Storage software)
Report_MatLoadingOrder_deleted.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !DelLoad
DateinameX = C:\Grundner\temp\Report_MatLoadingOrder_deleted.csv
ZugriffX = a

2.8 Report of a full nest unstack

It's possible to define a file, to which the system reports a full nest is unloaded from the machine. The content of the file will look like described below. There can be more than one line in this file.
New information will just be added by the storage.



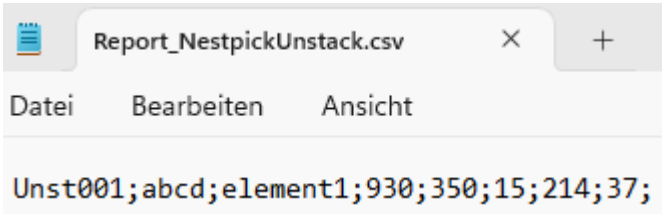
No.	**Parameter**	Description
1	**job-no**	Here you will receive job number of the unloaded board (ASCII 8 characters)
2	**number of source place**	The number of the source place (0 – 255)

Sample filenames:
(Storage software)
Report_FullNestpickUnstack.csv

Gateway settings (Gateway.ini):
[LogFile]
CmdCodeX = !EntladenFertigPlatz[X] - X = Source place
DateinameX = C:\Grundner\temp\Report_FullNestpickUnstack.csv
ZugriffX = a

2.9 Report of a nestpick unstack

It's possible to define a file, to which the system reports a nestpick unstack of a plate. The content of the file will look like described below. There can be more than one line in this file.
New information will just be added by the storage.



No.	Parameter	Description
1	**job-no**	Here you will receive job number of the unloaded board (ASCII 8 characters)
2	**JobAddition**	Additional job information (ASCII 4 characters)
3	**Element ID**	Identification of the part (ASCII 14 characters)
4	**length**	Panel length [mm]
5	**width**	Panel width [mm]
6	**thickness**	Panel thickness [mm]
7	**type**	Type number for a specific board (numeric 1 – 4095)
8	**number of dest. Place**	The number of the destination place (0 – 255)

Sample filenames:
(Storage software)
Report_NestpickUnstack.csv
or
Report_NestpickRestUnstack.csv

a) Gateway settings (Gateway.ini):

```
[LogFile]
CmdCodeX = !AbstapelnTeil
DateinameX = C:\Grundner\temp\Report_NestpickUnstack.csv
ZugriffX = a
```

b) Gateway settings (Gateway.ini) for Rest parts:

```
[LogFile]
CmdCodeX = !AbstapelnRest
DateinameX = C:\Grundner\temp\Report_NestpickRestUnstack.csv
ZugriffX = a
```

Grundner Sondermaschinen GmbH
Gewerbepark Nord 1, 4621 Sipbachzell, Austria
office@grundner.co.at +43 7240 21010
www.grundner.co.at