

Blunk electronic Inhaber: Dipl.-Ing. Mario Blunk Buchfinkenweg 3 99097 Erfurt / Germany	cellular land line email web	+49 (0) 176 2904 5855 +49 (0) 361 6022 5184 mario.blunk@blunk-electronic.de www.blunk-electronic.de
--	---------------------------------------	--



Bedienungsanleitung und Installation der Lagerverwaltungs-Software

Stock Manager (ERP)

Doc. Version 2015-11-20-1

Überblick: *Anleitung zur Bedienung und Installation von Stock Manager unter Linux-Betriebssystemen. Zielgruppe sind EMS-Firmen, die ihre Warenwirtschaft nach Linux portieren. Herausragende Eigenschaften: Stock Manager bindet den Anwender nicht an einen bestimmten Softwarehersteller, ist OpenSource und sicherheitsrelevant programmiert in Ada.*

Schlüsselwörter: *Manufacturer, Hersteller, Distributor, part code, Herstellercode, order code, Bestellnummer, URL datasheet, Datenblatt, storage place, Lagerort, reserved, reserviert, quantity, Menge, net price, Nettopreis, BOM, EAGLE, CAE, facility, Firma, order list, Bestell-Liste, withdrawal list, Entnahme-Liste, status production, Produktionsstatus, obsolete, active, discontinued, not for new designs, checkout, ausbuchen, query, Anfrage, data base, Datenbank, csv, LibreOffice, Linux, Command Line, CLI, Kommandozeile, graphical user interface, GUI, Graphische Bedieneroberfläche, Ada, GNAT, import, compiler, plain text, Klartext, log, ssh, Server, NFS, Samba, SMB, Installations-Support, GNU, make, SQL, git*

1 Vorwort

Als ein Unternehmen, welches Elektronikbaugruppen herstellt (sg. electronic manufacturing services (EMS)) brauchen Sie ein System, daß Ihr Bauteillager verwaltet. Anderenfalls verlieren Sie sich zwischen hunderten von kleinen Kartons und Plastiktüten. Lagerverwaltungs-Software gibt es bereits, aber was diesen Produkten fehlt ist einfach umrissen:

- ➔ Portierbarkeit/Lesbarkeit der Datenbank unabhängig vom Software-Hersteller/Händler
- ➔ manuelle/menschliche Interaktion falls die Software versagt
- ➔ offener Quellcode, sg. OpenSource, um Erweiterungen, Modifikationen und Fehlerbehebung zu ermöglichen, Datenflüsse zu kontrollieren, ...
- ➔ lizensfreie und zeitlich unbegrenzte Benutzung der Software
- ➔ Linux-Betriebssystem Unterstützung

Sind diese Forderungen erfüllt, können Sie sich auf Ihre eigentliche Arbeit konzentrieren:

Elektronik herstellen !

Wir begrüßen jede Form von Kritik um dieses Produkt und dieses Dokument zu verbessern !

Inhalt

1 Vorwort.....	2
2 Schlüsselfunktionen des Stock Managers.....	5
3 Das Konzept dahinter.....	6
4 Wie wird Stock Manager bedient ?.....	7
4.1 Einen Eintrag anzeigen.....	8
4.1.1 Anzeige mit ID.....	8
4.1.2 Anzeige mit Facility Part Code.....	9
4.1.3 Anzeige mit Distributor oder Hersteller Bestellnummer.....	11
4.1.4 Anzeige aller Informationen.....	12
4.2 Einen Eintrag bearbeiten.....	13
4.2.1 Unternehmens-Daten.....	13
4.2.1.1 Menge auf Lager	13
4.2.1.2 Reservierte Menge	14
4.2.1.3 Lagerort	15
4.2.2 Hersteller-Daten.....	16
4.2.2.1 Name.....	16
4.2.2.2 Bauteil-Code.....	17
4.2.2.3 URL zum Datenblatt.....	17
4.2.3 Distributor Daten	18
4.2.3.1 Name.....	18
4.2.3.2 Bestellnummer.....	19
4.2.3.3 Mindestbestellmenge.....	20
4.2.3.4 Netto-Preis.....	21
4.3 Anlegen eines Lager-Gegenstandes.....	22
4.4 Einen Lagergegenstand löschen.....	23
4.5 Logbuch einsehen.....	24
4.6 "Rückgängig-Machen" / Rollback.....	25
4.7 Makro-Funktionen.....	26
4.7.1 Erzeugung der Facility BOM.....	27
4.7.2 Materialanfragen anhand der Facility BOM.....	28
4.7.3 Materialentnahmen anhand der Facility BOM.....	29
4.8 Hilfe bekommen.....	30
5 Facility Part Code Syntax.....	31
6 Konfigurationsdateien.....	32
7 Installation.....	33
7.1 Einzelplatz-Betrieb (Stand Alone).....	33
7.2 Server-Betrieb.....	33
8 Einschränkungen.....	34
8.1 Hersteller.....	34
8.2 Distributoren.....	34
8.3 Maximale Anzahl von Einträgen in der Lager-Datenbank.....	34
8.4 Lade- und Speicherzeiten.....	34
9 Erweiterbarkeit.....	34
10 Häufig gestellte Fragen.....	35
11 Was noch zu tun ist.....	36

12 Links.....	37
13 Haftungsausschluß.....	38

2 Schlüsselfunktionen des Stock Managers

Im Folgenden finden Sie eine Zusammenfassung dessen, was *Stock Manager* in Ihrem Unternehmen für Sie erledigen kann:

***Stock Manager* erfaßt:**

- Ein- und Ausgang von Waren (Mengen, reservierte Bestände, Verfügbarkeiten, ...)
- Lagerorte, Projektnamen, ...
- Herstellerdaten (Name, Produktionsstatus, Hersteller-Teilenummern, ...)
- Distributordaten (Name, Bestellnummern, Nettopreise, Mindestbestellmengen, ...)

***Stock Manager* generiert:**

- Materiallisten (BOM)
- Bestell-Listen (OL)
- Entnahme-Listen (WL)

3 Das Konzept dahinter

Das Produkt *Stock Manager* besteht im wesentlichen aus einer ausführbaren Linux-Datei (ELF), welche Optionen und Parameter per Kommandozeile (CLI) erhält. Leser, die vor der Kommandozeile zurückschrecken, seien hier ermutigt, weiterzulesen. Wir nehmen Sie an der Hand, wenn nötig per Telefon oder PC-Fernwartung. Mit CLI vertraute Anwender werden schnell mit *Stock Manager* zurechtkommen. Wer sich einmal an mit der Kommandozeile angefreundet hat, weiß deren Effizienz zu schätzen, versprochen !

Es gibt außerdem noch eine Konfigurationsdatei und Textdateien, die Hilfe und Beispiele enthalten. Am wichtigsten ist die *Lager-Datenbank*:

Dabei handelt es sich um eine sogenannte CSV-Datei. CSV steht für "Comma-Separated Values". Das Trennzeichen kann nicht nur ein Komma, sondern auch ein Semikolon, ein Leerzeichen, ein Doppelpunkt oder etwas anderes sein. Dieses Format hat folgende herausragende Eigenschaften:

- ➔ Es kann mit JEDER Software zur Tabellenkalkulation wie LibreOffice oder MS-Office gelesen werden.
- ➔ Sie können es mit dem einfachsten Texteditor bearbeiten.
- ➔ Sie können es einfach lesen, denn es ist in Klartext geschrieben.
- ➔ Selbst wenn das Programm *Stock Manager* versagt, wenn die Firma *Blunk electronic* nicht mehr existieren sollte, oder wenn Sie sich entscheiden nicht mehr damit zu arbeiten, Sie sind nicht verloren ! Die geleistete Arbeit, Lager-Gegenstände anzulegen und zu bearbeiten, kann mit Programmen weitergeführt werden, die csv lesen können !

**Somit geraten Sie nicht in die Fänge von Software-Herstellern,
deren Produkte auf
kryptischen binären Datenbanken basieren.**

Siehe Abschnitt 10 Häufig gestellte Fragen Seite 35 für mehr.

Stock Manager selbst meint die Kernfunktionen, die mit der Binärdatei `stock_manager` zu Verfügung gestellt werden. Es gibt keine graphische Bedieneroberfläche (GUI) mit bunten Fenstern und Eingabefeldern. Weil das Programm auf Kommandozeile basiert, ist Erweiterbarkeit in Sachen Stapelverarbeitung/Shell-Skripte und GUI impliziert. Jeder fähige Softwareentwickler kann Ihnen ein maßgeschneidertes GUI auf den *Stock Manager* setzen, daß Ihren Forderungen und finanziellen Möglichkeiten entspricht. Siehe auch Abschnitt 10 Häufig gestellte Fragen Seite 35 für mehr.

Bitte öffnen Sie an dieser Stelle den unten stehenden Link um Zugang zu den gelieferten Text-, csv- und Binärdateien zu erhalten:

https://github.com/Blunk-electronic/stock_manager/archive/v013.zip

4 Wie wird Stock Manager bedient ?

Im Folgenden gehen wir direkt in praktische Beispiele, anstatt graue Theorie zu diskutieren. Wir nehmen die Rolle der Person in Ihrer Firma an, die zum Arbeiten mit *Stock Manager* berechtigt ist. Außerdem nehmen wir an, die Installation sei vollständig und korrekt.

Die hier verwendeten Beispieldateien stehen frei zum Download zur Verfügung und können als Ausgangspunkt für Ihre eigene Materialverwaltung verwendet werden.

Das ist die wichtigste Datei: Die Materialdatenbank:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/stock_db.csv

An dieser Stelle dürfen Sie diese mit LibreOffice öffnen und einen Blick darauf werfen. Jeder Eintrag darin hat eine eindeutige Nummer (Spalte 1) und den sogenannten *facility part code* den Ihr Unternehmen verwendet (Spalte 2). Mittels dieser zwei Kennungen ist jeder Eintrag auffindbar und editierbar. So zum Beispiel gilt ID 13 einem Bauteil mit dem *facility part code* C_PAC_S_0805_VAL_100n. Dieser Code sollte einzig und ausreichend den betreffenden Lagergegenstand beschreiben. Im physischen, realen Lager des Unternehmens, wird die Tüte oder die Kiste, die das Teil enthält, mit diesem Code und der ID beschriftet. Siehe Abschnitt 5 Seite 31 für Details. So viel zu den Grundlagen. Sie können die Datei nun wieder schließen.

Es gibt ein paar Konfigurationsdateien, welche wir später in Abschnitt 6 Seite 32 besprechen.

4.1 Einen Eintrag anzeigen

Beginnen wir mit einer einfachen Abfrage der Materialdatenbank. Zum Abfragen brauchen Sie entweder die ID oder den *facility part code*.

4.1.1 Anzeige mit ID

In Ihrem Terminalfenster (der Eingabeaufforderung) geben Sie das Kommando:

`stock_manager show_by_id 13` wie unten gezeigt ein. Das Ergebnis erhalten Sie unmittelbar. Es wird der *facility part code*, das Datum der letzten Änderung, verfügbare Mengen, Lagerort und sonstige Informationen angezeigt:

```
lager@server2:~> stock_manager show_by_id 13
```

```
STOCK MANAGER version 001
```

```
=====
action          : SHOW_BY_ID
part id         : 13
```

```
-----
part id         : 13
part code       : C_PAC_S_0805_VAL_100n
date edited    : 2015-03-18 18:51:12
qty on stock   : 279
qty reserved   : 0
qty available  : 279
storage place  : n/a
project        : n/a
remarks        : vmax=50V
-----
```

```
lager@server2:~>
```


4.1.2 Anzeige mit Facility Part Code

Wenn der vollständige oder ein Fragment des *facility part codes* bekannt ist, kann die Anfrage wie unten gezeigt gestartet werden. Im Falle mehrerer Übereinstimmungen, werden alle angezeigt:

Beispiel #1:

```
lager@server2:~> stock_manager show_by_fac_code C_PAC_S_0805_VAL_100n
```

```
STOCK MANAGER version 001
```

```
=====
```

```
action          : SHOW_BY_FAC_CODE  
part code       : C_PAC_S_0805_VAL_100n
```

```
-----  
part id         : 13  
part code       : C_PAC_S_0805_VAL_100n  
date edited     : 2015-03-18 18:51:12  
qty on stock    : 279  
qty reserved    : 0  
qty available   : 279  
storage place   : n/a  
project         : n/a  
remarks         : vmax=50V  
-----
```

```
-----  
part id         : 14  
part code       : C_PAC_S_0805_VAL_100n_Vmax_100V  
date edited     : 2015-03-13 17:10:34  
qty on stock    : 0  
qty reserved    : 0  
qty available   : 0  
storage place   : n/a  
project         : n/a  
remarks         : n/a  
-----
```

```
Number of parts found : 2  
lager@server2:~>
```

Beispiel #2:

Nehmen wir an, es soll nach einem Widerstand mit dem Wert 422 Ohm gesucht werden:

```
lager@server2:~> stock_manager show_by_fac_code _VAL_422
```

```
STOCK MANAGER version 001
```

```
=====
action          : SHOW_BY_FAC_CODE
part code       : _VAL_422
```

```
-----
part id         : 161
part code       : R_PAC_S_0805_VAL_422
date edited    : 2015-03-18 18:59:01
qty on stock   : 8
qty reserved   : 0
qty available  : 8
storage place  : n/a
project        : n/a
remarks        : n/a
-----
```

```
lager@server2:~>
```

Korrekterweise sollten Widerstandswerte unter 1 Kiloohm mit R enden. In diesem Fall wäre das VAL_422R. Sollte es auch einen Kondensator mit 422 Nanofarad geben, würde auch dieser hier angezeigt werden.

Es darf auch mit Platzhaltern gearbeitet werden. So würde der Befehl

```
stock_manager show_by_fac_code *PAC_S_0805*
```

alle Bauteile mit dem SMD-Gehäuse Typ 0805 anzeigen.

4.1.3 Anzeige mit Distributor oder Hersteller Bestellnummer

Wenn die vollständige oder ein Fragment der Bestellnummer eines Herstellers oder Distributors bekannt ist, kann die Anfrage wie unten gezeigt gestartet werden. Im Falle mehrerer Übereinstimmungen, werden alle angezeigt. Die Verwendung der Option *full* ist hier angebracht (siehe auch Abschnitt 4.1.4 Seite 12):

Beispiel #1:

```
lager@server2:~> stock_manager full show_by_order_code 732736

STOCK MANAGER V009
=====
action          : SHOW_BY_ORDER_CODE
order code      : 732736

-----
part id         : 66
part code       : IC_PAC_S_SO6_VAL_74LVC1G18DCK
date edited    : 2015-05-15 10:34:57
qty on stock   : 6
qty reserved   : 0
qty available  : 6
storage place  : S3/8
project        : n/a
remarks        : n/a

-----
manufacturer 1 : TI
date edited    : 2015-07-28 08:15:00
part code      : SN74LVC1G18DCKR
url datasheet 1 : http://www.farnell.com/datasheets/1737053.pdf
url datasheet 2 : n/a
status prod.   : ACTIVE

-----
distributor 2  : AX
order code     : 732736
date edited    : 2015-07-28 08:16:08
qty min.       : 1
price net      : 0.11

-----
distributor 3  : FARNELL
order code     : 2334935
date edited    : 2015-04-07 11:15:02
qty min.       : 1
price net      : 0.19

-----
lager@server2:~>
```

Soll nach einem Hersteller-Code gesucht werden, ist die Anfrage z.B. so zu stellen:

```
lager@server2:~> stock_manager full show_by_manu_code LM324*
```

4.1.4 Anzeige aller Informationen

Wird die Option *full* angewendet, kommen deutlich mehr Informationen zum Lagergut zur Anzeige. Für EMS wichtige Informationen betreffen Hersteller, Distributoren, Preise und Produktionsstatus. Lassen Sie uns die Bauteilsuche noch einmal starten, mit der Option *full*:

```
lager@server2:~> stock_manager full show_by_id 13
```

```
STOCK MANAGER version 001
```

```
=====
action          : SHOW_BY_ID
part id         : 13

-----
part id         : 13
part code       : C_PAC_S_0805_VAL_100n
date edited    : 2015-03-18 18:51:12
qty on stock    : 279
qty reserved    : 0
qty available   : 279
storage place   : n/a
project        : n/a
remarks        : vmax=50V

-----
manufacturer 1  : AVX
date edited    : 2015-03-17 14:32:09
part code      : 08055C104JAT2A
url datasheet 1 : http://www.avx.com/docs/Catalogs/cx7r.pdf
url datasheet 2 : n/a
status prod.   : ACTIVE

-----
distributor 1   : REICHELT
order code      : X7R-G0805 100N
date edited    : 2015-03-17 14:43:21
qty min.       : 1
price net      : 0.04

-----
distributor 3   : FARNELL
order code      : 1740673
date edited    : 2015-03-17 14:45:24
qty min.       : 10
price net      : 0.01

-----
lager@server2:~>
```

An dieser Stelle fragen Sie vielleicht wieviele Hersteller oder Distributoren erlaubt sind. Sie möchten eventuell noch mehr Eigenschaften wie. *RoHS*- oder *UL*-Konformität unterbringen. Dazu siehe Abschnitt 8 und 9 für Limitierungen und Erweiterungen.

4.2 Einen Eintrag bearbeiten

4.2.1 Unternehmens-Daten

4.2.1.1 Menge auf Lager

Meistens werden Teile eingelagert oder dem Lager entnommen. Am einfachsten geht das für einzelne Teile wie unten gezeigt. Soll mit einem einzigen Vorgang eine große Menge von Teilen dem Lager entnommen werden, lesen Sie bitte in Abschnitt 4.7.3 auf Seite 29 nach. Mengenangaben werden hier in Form einer Differenz angegeben. Das heißt, wenn Sie ein Teil einlagern, geben Sie 1 ein. Nehmen Sie ein Teil aus dem Lager, geben Sie -1 ein. :

```
lager@server2:~> stock_manager edit 13 qty_delta_stock 1
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: QTY_DELTA_STOCK
part code BEL   : C_PAC_S_0805_VAL_100n
storage place   : S1/1
qty reserved    : 0
qty on stock old: 279
qty stock delta : +1
qty on stock new: 280

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

ACHTUNG: Das Bearbeiten (Editieren) eines Eintrages fordert vom Bediener immer eine Bestätigung !

Wird die Option *no_confirmation* verwendet, erfolgt die Einforderung dieser Bestätigung nicht. **Verwenden Sie diese Option mit Vorsicht !** Diese Option ist sinnvoll, wenn *stock_manager* aus einem Shellskript heraus gestartet wird:

```
lager@server2:~> stock_manager no_confirmation edit 13 qty_delta_stock 1
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: QTY_DELTA_STOCK
part code BEL   : C_PAC_S_0805_VAL_100n
storage place   : S1/1
qty reserved    : 0
qty on stock old: 279
qty stock delta : +1
qty on stock new: 280

lager@server2:~>
```

4.2.1.2 Reservierte Menge

Manchmal muß eine bestimmte Menge von Teilen für spezielle Projekte reserviert werden. In solch einem Fall, geben Sie die Differenz von neu zu reservierenden oder freizugebenden Teilen an. Mit dem Ausführen des Kommandos, ändert sich automatisch die Menge verfügbarer Teile:

```
lager@server2:~> stock_manager edit 13 qty_delta_reserved 70
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: QTY_DELTA_RESERVED
part code BEL   : C_PAC_S_0805_VAL_100n
storage place   : S1/1
qty on stock    : 279
qty reserved old: 0
qty rsvd. delta : +70
qty reserved new: 70

ARE YOU SURE ? (y/n) :y
```

```
lager@server2:~> stock_manager show_by_id 13
```

```
STOCK MANAGER version 001
=====
action          : SHOW_BY_ID
part id         : 13

-----
part id         : 13
part code       : C_PAC_S_0805_VAL_100n
storage place   : S1/1
date edited     : 2015-03-19 17:20:24
qty on stock    : 279
qty reserved    : 70
qty available   : 209
storage place   : S1/1
project         : n/a
remarks         : vmax=50V
-----

lager@server2:~>
```

ACHTUNG: Reservierte Teile können nicht aus dem Lager entnommen werden !

4.2.1.3 Lagerort

Der Lagerort ist wie unten gezeigt zuzuweisen. Beachten Sie: Werden Leerzeichen im Text verwendet, muß dieser in Hochkommas eingeschlossen werden:

```
lager@server2:~> stock_manager edit 13 storage_place "Shelf 4, Box 23"
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: STORAGE_PLACE
part code BEL   : C_PAC_S_0805_VAL_100n
place old       : S1/1
place new       : Shelf 4, Box 23

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

Siehe Abschnitt 11 Was noch zu tun ist für mehr.

4.2.2 Hersteller-Daten

4.2.2.1 Name

Um den Namen des Herstellers (eng. Manufacturer) zuzuweisen, finden Sie das folgende Beispiel. Beachten Sie: Werden Leerzeichen im Text verwendet, muß dieser in Hochkommas eingeschlossen werden:

```
lager@server2:~> stock_manager edit 13 manufacturer_2_name multicom
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: MANUFACTURER_2_NAME
part code BEL   : C_PAC_S_0805_VAL_100n
manuf. name old : n/a
manuf. name new : multicom

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

Um einen Hersteller von einem Eintrag zu **entfernen**, weisen Sie ihm einfach den Namen "n/a" (nicht zugewiesen) zu:

```
lager@server2:~> stock_manager edit 13 manufacturer_2_name n/a
```

Daraufhin wird dieser Hersteller nicht mehr für dieses Teil angezeigt.

4.2.2.2 Bauteil-Code

Jeder Hersteller hat eine eigene Nomenklatur für seine Produkte. Achtung: Der Hersteller-Code ist nicht zu verwechseln mit dem *facility part code* ! Beachten Sie: Werden Leerzeichen im Text verwendet, muß dieser in Hochkommas eingeschlossen werden:

Der Bauteil-Code des Herstellers 1 wird geändert wie hier gezeigt:

```
lager@server2:~> stock_manager edit 13 manufacturer_1_part_code 08055C104JAT2A
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: MANUFACTURER_1_PART_CODE
part code BEL   : C_PAC_S_0805_VAL_100n
manuf. name     : AVX
part code old   : NCC1701-E
part code new   : 08055C104JAT2A

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

4.2.2.3 URL zum Datenblatt

Der Link zum Datenblatt wird zugewiesen wie in diesem Beispiel:

```
stock_manager edit 250 manufacturer_1_datasheet_1 http://www.ti.com/BCW60.pdf
```

Beachten Sie: Werden Leerzeichen im Text verwendet, muß dieser in Hochkommas eingeschlossen werden. Siehe Abschnitt 8 Einschränkungen für maximale Anzahl von Links zum Datenblatt.

4.2.3 Distributor Daten

4.2.3.1 Name

Um einem Bauteil einen Distributor zuzuweisen oder ihn zu ändern, siehe das Beispiel unten. Achtung: Werden Leerzeichen verwendet, muß der Text in Hochkommas eingeschlossen werden:

```
lager@server2:~> stock_manager edit 13 distributor_2_name "AX ELECTRONIC"
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: DISTRIBUTOR_2_NAME
part code BEL   : C_PAC_S_0805_VAL_100n
dist. name old  : n/a
dist. name new  : AX ELECTRONIC

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

Wie in Abschnitt 4.2.2.1 Seite 16 ausgeführt, wird ein Distributor gelöscht, in dem sein Name nach "n/a" geändert wird. Daraufhin erscheint der betreffende Distributor nicht mehr.

4.2.3.2 Bestellnummer

Distributoren verwenden Bestellnummern für die Bauteile, die sie vertreiben. Folglich muß ein Bestellecode einem Lager-Gegenstand zugewiesen werden. Der Bestellcode kann eine Nummer, ein Text oder eine Mischung daraus sein. Achtung: Enthält der Bestellcode Leerzeichen, muß er in Hochkommas eingeschlossen werden:

```
lager@server2:~> stock_manager edit 13 distributor_2_order_code 445723
```

```
STOCK MANAGER version 001
```

```
=====
```

```
action          : EDIT
part id         : 13
property to edit: DISTRIBUTOR_2_ORDER_CODE
part code BEL   : C_PAC_S_0805_VAL_100n
distributor name: AX ELECTRONIC
order code old  : n/a
order code new   : 445723
```

```
ARE YOU SURE ? (y/n) :y
```

```
lager@server2:~>
```

4.2.3.3 Mindestbestellmenge

Häufig muß eine Mindestbestellmenge eingehalten werden. Um diese zu dem betreffenden Bauteil zu vermerken, siehe das folgende Beispiel:

```
lager@server2:~> stock_manager edit 13 distributor_2_qty_min 10
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: DISTRIBUTOR_2_QTY_MIN
part code BEL   : C_PAC_S_0805_VAL_100n
distributor name: AX ELECTRONIC
dist. order code: 445723
qty min old     : 1
qty min new     : 10

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

Der Initialwert (Default) für die Bestellmenge ist 1.

4.2.3.4 Netto-Preis

Der Netto-Preis (also ohne MwSt.) für EIN Bauteil wird wie im nächsten Beispiel gezeigt, eingepflegt. Später, wenn die Facility BOM generiert wird (siehe Abschnitt 4.7.1 Erzeugung der Facility BOM Seite 27), werden diese Zahlen zur Kalkulation des Materialpreises einer Baugruppe hinzugezogen:

```
lager@server2:~> stock_manager edit 13 distributor_2_price_net 0.05
```

```
STOCK MANAGER version 001
=====
action          : EDIT
part id         : 13
property to edit: DISTRIBUTOR_2_PRICE_NET
part code BEL   : C_PAC_S_0805_VAL_100n
distributor name: AX ELECTRONIC
price net old   : 0.00
price net new   : 0.05

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

An dieser Stelle erfolgt **keine** Betonung der verwendeten Währung. Wenn Sie in einem Land arbeiten, daß mit Euro rechnet, betrachten Sie diese Zahlen als EURO. Arbeiten Sie mit Russischem Rubel, betrachten Sie die Zahlen einfach als Rubel.

Siehe Abschnitt 11 Was noch zu tun ist für mehr Informationen.

4.3 Anlegen eines Lager-Gegenstandes

Bisher haben wir nur mit der bestehenden Lager-Datenbank `stock_data_base` gearbeitet:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/stock_db.csv

Aber Gegenstände müssen auch neu angelegt werden. Das Neu-Anlegen eines Gegenstandes wird anhand des Facility Part Codes vorgenommen. Da es sich hier um eine Version des *Stock Managers* für EMS handelt, muß die Syntax des Facility Part Codes bestimmten Mindestregeln gehorchen. Siehe Abschnitt 5 Facility Part Code Syntax Seite 31. Die Syntax kann auch an andere Objekte angepaßt werden. Siehe Abschnitt 11 Was noch zu tun ist für mehr.

Was auch immer Sie einlagern wollen, es bedarf des Facility Part Codes, der zu Ihrer Firma gehört. Anderenfalls, gehen Sie irgendwann verloren zwischen willkürlich gewählten Bezeichnungen.

Nun ein Beispiel:

```
lager@server2:~> stock_manager add IC_PAC_S_SO8_VAL_AT24C08

STOCK MANAGER version 001
=====
action          : ADD
part code       : IC_PAC_S_SO8_VAL_AT24C08

ARE YOU SURE ? (y/n) :y

Part has been added to data base with part_id 274

lager@server2:~>
```

Nach dem Anlegen eines neuen Lagergegenstandes, wird diesem automatisch eine ID zugewiesen. Nun dürfen Sie mit der Edit-Funktion Mengen, Hersteller- und Distributor-Daten zuweisen.

4.4 Einen Lagergegenstand löschen

Soll ein Gegenstand nicht mehr länger gelagert werden, sollte dieser gelöscht werden. Als Sicherheitsmaßnahme, kann das nur anhand des exakten Facility Part Codes erfolgen.

VORSICHT: Mit dem Löschen, gehen alle früher vermerkten Informationen wie Mengen, Hersteller- und Distributor-Daten verloren !

Hier das Beispiel:

```
lager@server2:~> stock_manager delete IC_PAC_S_SO8_VAL_AT24C08
```

```
STOCK MANAGER version 001
=====
action          : DELETE
part_code_fac   : IC_PAC_S_SO8_VAL_AT24C08

WARNING : DELETE PART WITH
        ID       : 274
        PART_CODE : IC_PAC_S_SO8_VAL_AT24C08

ARE YOU SURE ? (y/n) :y

lager@server2:~>
```

4.5 Logbuch einsehen

Alle Edit-Operationen, also Änderungen, werden aufgezeichnet. Somit ist es möglich, frühere Vorgänge einzusehen. Siehe das Beispiel unten:

```
lager@server2:~> stock_manager log

STOCK MANAGER version 001
=====
action          : LOG
stock data base : /home/luno/verwaltung/lager/stock_db.csv

stock manager activities log
-----
DATE
YYYY:MM:DD HH:MM:SS   | action
-----
.
.
.
2015-03-20 08:19:27   | edit  13  distributor_2_name  AX ELECTRONIC
2015-03-20 08:23:20   | edit  13  distributor_2_order_code  445723
2015-03-20 08:26:30   | edit  13  distributor_2_qty_min  10
2015-03-20 08:30:51   | edit  13  distributor_2_price_net  0.05
2015-03-20 08:38:51   | edit  13  distributor_2_name  n/a
2015-03-20 08:48:47   | add  IC_PAC_S_SO8_VAL_AT24C08
2015-03-20 08:54:54   | delete  IC_PAC_S_SO8_VAL_AT24C08
lager@server2:~>
```


4.6 “Rückgängig-Machen” / *Rollback*

Nach jeder Edit-Operation wird eine Sicherheitskopie der ursprünglichen Lager-Datenbank im Hintergrund erstellt. Jede Edit-Operation kann rückgängig gemacht werden mit einem sogenannten *Rollback*. Dieses Rollback stellt einfach die Lager-Datenbank vom jüngsten Backup wieder her:

```
lager@server2:~> stock_manager roll_back
```

```
STOCK MANAGER version 001
```

```
=====
```

```
action          : ROLL_BACK
```

```
stock data base : /home/luno/lager/stock_db.csv
```

```
restore from    : /home/luno/lager/bak/BAK_2015-03-20_09-05-39__stock_db.csv
```

```
ARE YOU SURE ? (y/n) :y
```

```
lager@server2:~>
```

Hinweis: Backup-Dateien werden nie vom *Stock Manager* gelöscht.

4.7 Makro-Funktionen

Bis jetzt, haben wir uns mit individuellen Einträgen bestimmter Lager-Gegenstände beschäftigt. Im bestimmten Fällen, wenn es um sehr viele Edit-Operation geht, kann das aber sehr zeitintensiv und fehleranfällig sein. Situationen, die hohe Effizienz fordern, sind zum Beispiel:

1. Datenbank-Abfragen über die Verfügbarkeit von Bauteilen anhand der Facility BOM
2. Materialentnahmen anhand der Facility BOM
3. Erzeugung von Material-Entnahmelisten
4. Erzeugung von von Bestell-Listen für Distributoren

Der Ausgangspunkt für diese Vorgänge ist die sogenannte *Facility BOM*.

4.7.1 Erzeugung der Facility BOM

Die Facility Material-Liste ist eine csv-formatierte Datei, die die zu bestückenden Bauteile einer Leiterplatte enthält. Statt vieler Worte finden Sie ein Beispiel hier:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/facility_bom.csv

Hierin sehen Sie Mengenangaben, Part IDs, Part Codes und statistische Zahlen für EINE zu bestückende Leiterplatte. Aber wie kommt man zu dieser Datei ?

Die Antworten:

1. durch schreiben von Hand (nicht ratsam)
2. mit Hilfe des *Stock Managers*

Punkt 2 ist das was wir bevorzugen. Jedes CAE-Werkzeug (wie z.B. *CadSoft EAGLE*) kann eine roh-BOM mit grundlegenden Informationen über Bauteilname, Value und einiger anderer Eigenschaften ausgeben.

Das jeweilige CAE-Werkzeug muß so angepaßt werden, daß es Spalten und Kopfzeilen erzeugt, wie in diesem Beispiel gezeigt:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/bom_created_by_eagle_ulp.csv

Diese Datei wird nun mit *Stock Manager* importiert:

```
lager@server2:~> stock_manager make_bom bom_created_by_eagle_ulp.csv facility_bom.csv
```

```
STOCK MANAGER version 001
=====
action          : MAKE_BOM
facility name    : BEL
input eagle bom : bom_created_by_eagle_ulp.csv
output bom      : facility_bom.csv

BOM ready

lager@server2:~>
```

Das wars. Siehe das Ergebnis hier:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/facility_bom.csv

Der Facility BOM-Generator prüft, ob alle nötigen Teile im Lager angelegt wurden. Ist ein Teil nicht als Lager-Gegenstand vorhanden, wird der Vorgang abgebrochen und über fehlende Einträge informiert.

Nun können Sie Makro-Funktionen ausführen.

4.7.2 Materialanfragen anhand der Facility BOM

Bevor eine Bestückung beginnen kann, muß sichergestellt sein, daß alle nötigen Teile physisch auf Lager sind. Fehlen Teile, wird eine Bestell-Liste erzeugt. Wenn Sie also vorhaben, eine Leiterplatte zu bestücken, gehen Sie so vor:

Beispiel #1:

```
lager@server2:~> stock_manager query_bom facility_bom.csv 1
```

```
STOCK MANAGER version 001
```

```
=====
action          : QUERY_BOM
input fac. bom  : facility_bom.csv
order list      : facility_bom_order.csv
qty of units    : 1
```

All positions of given BOM are available on stock.

No order list created.

```
lager@server2:~>
```

Alles ist prima. Alle Teile sind vorhanden. Nehmen wir nun an, es müssen 4 Platinen bestückt werden:

Beispiel #2:

```
lager@server2:~> stock_manager query_bom facility_bom.csv 4
```

```
STOCK MANAGER version 001
```

```
=====
action          : QUERY_BOM
input fac. bom  : facility_bom.csv
order list      : facility_bom_order.csv
qty of units    : 4
```

Parts need to be ordered. Please read file 'facility_bom_order.csv' !

```
lager@server2:~>
```

Eine Bestell-Liste wurde erzeugt. Mit dieser können Sie nun Bestellungen bei Distributoren platzieren:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/order_list.csv

4.7.3 Materialentnahmen anhand der Facility BOM

Im Unterschied zu einer Materialanfrage (*query_bom*), nimmt die Operation *checkout_bom* viele Materialentnahmen mit einem Mal vor. Natürlich müssen die Teile danach auch physisch aus dem Lager entnommen werden. *Stock Manager* wird dazu die Facility BOM und die Anzahl zu bestückender Platinen übergeben.

Im ersten Beispiel nehmen wir an, daß nicht alle nötigen Teile auf Lager sind:

Example #1:

```
lager@server2:~> stock_manager checkout_bom 4
```

```
STOCK MANAGER version 001
=====
action          : CHECKOUT_BOM
input fac. bom  : facility_bom.csv
order list      : facility_bom_order.csv
withdrawal list : facility_bom_withdrawal.csv
qty of units    : 4

Parts need to be ordered. Please read file 'facility_bom_order.csv' !

Withdrawal list not created.

No parts checked out from stock.

lager@server2:~>
```

So lange Teile fehlen, werden keinerlei Materialentnahmen verbucht. Eine Bestell-Liste wird statt dessen erzeugt. Folglich kann die Bestückung nicht beginnen. Sie sind nun gezwungen, die nötigen Teile zu beschaffen und die Mengen im Lager zu aktualisieren (Siehe Abschnitt 4.2.1.1 Menge auf Lager Seite 13) und schließlich das Ganze zu wiederholen.

Nun nehmen wir an, es soll nur EINE Leiterplatte bestückt werden:

Beispiel #2:

```
lager@server2:~> stock_manager checkout_bom facility_bom.csv 1
```

```
STOCK MANAGER version 013
```

```
=====
action          : CHECKOUT_BOM
input fac. bom   : facility_bom.csv
order list       : facility_bom_order.csv
withdrawal list  : facility_bom_withdrawal.csv
qty of units     : 1
```

```
All positions of given BOM are available on stock.
```

```
No order list created.
```

```
WARNING: All positions of given BOM will be checked out from stock !!!
```

```
WARNING: After checkout you must take all items from stock as listed in file
'facility_bom_withdrawal.csv' !
```

```
ARE YOU SURE ? (y/n) :y
```

```
Now withdraw items from stock as listed in file 'facility_bom_withdrawal.csv' !
```

```
lager@server2:~>
```

Das Ergebnis ist offensichtlich. Es brauchen keine Teile beschafft werden, eine Entnahme-Liste wurde generiert und die Lagermengen wurden aktualisiert. Entsprechend dieser Liste

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/withdrawal_list.csv

sind Sie nun gefordert, Bauteile umgehend aus dem Lager zu entnehmen.

4.8 Hilfe bekommen

Wann immer Sie Hilfe oder Beispiele brauchen, geben Sie einfach,

```
stock_manager help
```

ein, was Ihnen eine Liste von Beispielkommandos mit kurzen Erläuterungen zurück gibt.

Sie können diese Hilfe-Dateien auch hier finden:

https://github.com/Blunk-electronic/stock_manager/blob/master/conf/.stock_manager/help_english.txt

https://github.com/Blunk-electronic/stock_manager/blob/master/conf/.stock_manager/help_german.txt

Wenn Sie gar nicht weiterkommen, kontaktieren Sie uns bitte per Email oder Telefon. Siehe Kontaktdaten auf dem Deckblatt dieser Anleitung.

5 Facility Part Code Syntax

Die gegenwärtige Ausführung des *Stock Managers* ist auf EMS (Electronic Manufacturing Services) zugeschnitten. Er kann auch auf andere Anwendungen zugeschnitten werden. Dazu gehören z.B. medizinische Daten, Güter/Personenverkehr, Nahrungsmittel, Kundendatenbanken, ...)

Elektronische Komponenten im EMS-Bereich fordern eine bestimmte Nomenklatur. In der Vergangenheit hat sich eine Nomenklatur gefunden, die hier detaillierter beschrieben werden soll:

Nehmen wir zum Beispiel das Bauteil im Lager mit der ID 13 (Wir hatten damit in früheren Kapiteln zu tun.) Sein Facility Part Code lautet:

C_PAC_S_0805_VAL_100n

Für einen Elektroniker gibt es hier nicht viel zu erklären:

1. Der Facility Part Code beginnt immer mit dem Prefix C, R, L, Q, IC, LED, DIS, X, J, S, K und so weiter. Das Prefix sagt aus, zu welcher Kategorie das Teil gehört (Kondensator, Widerstand, Induktivität, Quartz, ...)
2. Das nächste Feld ist ein reserviertes Schlüsselwort PAC_S oder PAC_T welches das folgende Feld, die Gehäusevariante, ankündigt. In diesem Fall handelt es sich um das SMD 0805 Gehäuse.
3. Das Schlüsselwort VAL zeigt an, daß das nachfolgende Feld den eigentlichen Kapazitätswert (100 Nanofarad) beinhaltet.

Soviel zu den minimal nötigen Parametern, die den Kondensator beschreiben. Sind mehrere Eigenschaften zu nennen, könnte das Bauteil mit ID 40 als Beispiel dienen:

Der Facility Part Code dieses Bauteiles lautet:

C_PAC_T_RADIAL_7mm_GRID_2.54mm_VAL_100u_Vmax_35V

1. Es ist ein Kondensator.
2. Sein Gehäuse ist 7mm im Durchmesser, bedrahtet (THT), radial mit einem Raster von 2.54mm.
3. Kapazität 100 Mikrofarad (Statt "µ" muß "u" verwendet werden.)
4. Maximale Spannung 35V

Sie können nach Belieben weitere Eigenschaften wie Maximaltemperatur, Temperaturkoeffizient, Verluste u.s.w. angeben. Wichtig ist dabei das die Eigenschaft ankündigende Schlüsselwort. Nehmen Sie sich die Zeit und studieren Sie die Facility Part Codes anderer Bauteile in der Beispieldatenbank:

https://github.com/Blunk-electronic/stock_manager/blob/master/example_database/stock_db.csv

Innerhalb des CAE-Werkzeuges ist der Facility Part Code den Bauteilen zuzuweisen. Er funktioniert als Primärschlüssel vom Schaltplan/Layout bis zur Bestell- oder Materialentnahme-Liste.

6 Konfigurationsdateien

Damit der *Stock Manager* korrekt funktioniert, ist etwas Konfiguration entsprechend der ihn verwendenden Fabrik oder Firma nötig.

Ein verstecktes Verzeichnis *.stock_manager/*, welches sich im Home-Verzeichnis des Bedieners befindet, enthält die Hilfe-Dateien (siehe Abschnitt 4.8 Hilfe bekommen Seite 30) und am wichtigsten: die Konfigurationsdatei *stock_manager.conf*. Ein Beispiel finden Sie hier:

https://github.com/Blunk-electronic/stock_manager/blob/master/conf/.stock_manager/stock_manager.conf

Diese Datei ist einfach zu verstehen:

```
language english
currency EUR
# The first facility name will be used as primary.
# Others serve as alternatives in the order they appear here.
# If primary namee not available in eagle bom file, alternatives will be checked.
facility_name BEL JEN

#directories
directory_of_stock_data_base  /home/luno/controlling/stock
directory_of_log               /home/luno/controlling/stock/log
directory_of_backup            /home/luno/controlling/stock/bak

#customer specific prefixes NOTE: Affected parts require a function assigned to !
customer_prefixes             XS XP XD
```

Hier stellen Sie die Sprache der Hilfe-Funktion, die Währung und den Firmennahmen ein. Beschreiben Sie die Firma oder Fabrik mit maximal drei Großbuchstaben, ähnlich wie es mit Flughäfen gemacht wird (FRA or SXF). Zum Beispiel:

"*Blunk electronic*" kann mit BEL abgekürzt werden.

Die letzten drei Zeilen bestimmen, wo die Lager-Datenbank gespeichert ist, wo das Logbuch und die Sicherheitskopien sich befinden. Die dort angegebenen Verzeichnisse (oder Ordner) können lokal oder im Netzwerk liegen. Unter UNIX und Linux sind das z.B. NFS oder Samba-Freigaben. Es ist **äußerst ratsam**, diese in die reguläre Datensicherung einzubinden.

Eine Verzeichnisangabe mittels Umgebungsvariablen wie z.B. \$HOME/controlling/stock wird zur Zeit noch nicht unterstützt.

7 Installation

Für UNIX- oder Linux-Administratoren, sollte das Kompilieren und Installieren keine ernsthaften Schwierigkeiten mit sich bringen. Es sind keine Bibliotheken nötig. Am einfachsten ist es, einfach die Binärdatei *stock_manager* nach `$HOME/bin` zu kopieren und die Umgebungsvariable `$PATH` entsprechend anzupassen.

Wie auch immer, zögern Sie nicht, uns zu kontaktieren wie auf dem Deckblatt dieser Anleitung angegeben.

Siehe auch Abschnitt 11 Was noch zu tun ist Seite 36 für noch nicht abgeschlossene Baustellen.

Die Entwicklung, Releases und Quellcodes sind hier mit zu verfolgen oder per Download verfügbar:

https://github.com/Blunk-electronic/stock_manager

7.1 Einzelplatz-Betrieb (Stand Alone)

Wenn nur ein Bediener vorgesehen ist, mit *Stock Manager* zu arbeiten, sind keine speziellen Sicherheitsmaßnahmen nötig.

7.2 Server-Betrieb

Wenn eine Gruppe von Leuten mit *Stock Manager* arbeiten darf und soll, sind folgende Punkte **höchst ratsam**:

1. Richten Sie einen Nutzer am Server ein, der Lese- und Schreibrecht auf die Lager-Datenbank hat.
2. Stellen Sie sicher, daß dieser Nutzer sich nur EINMAL am Server anmelden kann.
3. Stellen Sie sicher, daß nur dieser Nutzer die Binärdatei *Stock Manager* ausführen darf.

8 Einschränkungen

8.1 Hersteller

Gegenwärtig sind maximal zwei Hersteller für einen Lagergegenstand möglich.

8.2 Distributoren

Gegenwärtig sind maximal sechs Distributoren für einen Lagergegenstand möglich.

8.3 Maximale Anzahl von Einträgen in der Lager-Datenbank

Die Obergrenze für die Menge von Einträgen ist durch die Größe des Arbeitsspeichers (RAM) und der Größe der Swap-Partition (Auslagerungsdatei) bestimmt.

8.4 Lade- und Speicherzeiten

Da es sich bei *Stock Manager* im Wesentlichen um eine einzige ausführbare Linux-Datei handelt, wird die Lager-Datenbank bei jeder Ausführung von *Stock Manager* geladen und gespeichert. In langsamen Netzwerken kann das zu Datenstau und Verzögerungen führen.

9 Erweiterbarkeit

Wie in Abschnitt 3 Das Konzept dahinter auf Seite 6 ausgeführt, kann das Programm mit externer Software zusammenarbeiten. Erweiterungen dieser Art sind möglich:

1. Stapelverarbeitung (Shell-Skripte)
2. Aufruf aus externer Software heraus
3. Graphische Benutzeroberfläche (GUI)
4. Schnittstelle zu SQL-Datenbanken

10 Häufig gestellte Fragen

Leser, die an MS-Excel gewöhnt sind, fragen: "Warum verwenden Sie keine xls-Dateien ?".

Antworten:

- ➔ Xls ist ein verschlossenes, Microsoft-eigenes Format.
- ➔ Es ist binär kodiert.
- ➔ Es kann nur durch MS-Software erzeugt werden.
- ➔ Es kann nach 10 Jahren nicht mehr korrekt von neueren MS-Excel-Versionen gelesen werden.
- ➔ MS ist nicht für sicherheitsrelevante Anwendungen geeignet.

Warum wird eine Lager-Verwaltung sicherheitsrelevant in Ada programmiert ?

- ➔ Die Bestückung von Leiterplatten (PCBs) mit elektronischen Komponenten ist sicherheitsrelevant.
- ➔ Fehlbstücke Leiterplatten verursachen extensiven Verlust an Zeit, Geld und Material.
- ➔ Elektronische Komponenten für Luftfahrt, Eisenbahnen und medizinische Anwendungen sind sicherheitsrelevant.

11 Was noch zu tun ist

1. eine Funktion, die eine Statistik über die Lager-Datenbank erzeugt (Menge aller Teile, finanzieller Wert der gelagerten Gegenstände, ...)
2. Berücksichtigung und Ausgabe der eingestellten Währung nach Preisen
3. Übersetzung der Hilfe-Dateien in andere Sprachen
4. Implementierung von Datei-Lock-Mechanismen
5. Fehlerfreiheit sicherstellen, wenn der erste Lagergegenstand in der Lager-Datenbank angelegt werden soll und diese noch nicht existiert.
6. Erzeugung der Verzeichnisse für Logbuch und Sicherheitskopien, falls diese noch nicht vorhanden sind.
7. Erstellung einer Linux Man-Page
8. Unterstützung von Zeitzonen
9. Verbesserung der "no_confirmation" - Funktion
10. Implementierung einer show_by_storage_place-Funktion
11. Sicherstellen, daß ein Lagerplatz nur einmal vergeben werden kann. Eventuell ist das keine gute Idee, weil damit eine durchaus beabsichtigte Mehrfachbelegung eines Lagerortes ausgeschlossen würde.
12. Verbesserung des Facility Part Code – Checks.
13. Loggen der Rollback-Operationen
14. Installations-Skript

12 Links

- (1) *CadSoft EAGLE Training und Beratung* – ein vernünftiger Weg zu vernünftiger Arbeit auf <http://www.blunk-electronic.de>

- (2) Die Office-Alternative : *LibreOffice* auf <http://www.libreoffice.org>



- (3) *EAGLE* – ein effizientes Schaltplan und Layout-Werkzeug auf <http://www.cadsoftusa.com>



- (4) Suchen Sie nach einem schlanken **Boundary Scan Test System** ? Sehen Sie **hier**!

13 Haftungsausschluß

Diese Anleitung wurde mit großer Sorgfalt erstellt. Ich übernehme keine Verantwortung für Irrtümer und Fehler, die in diesem Dokument vorkommen. Ich behalte mir das Recht vor, Änderungen jederzeit und ohne Ankündigung vorzunehmen. Ich verpflichte mich nicht, Aktualisierungen hierin vorzunehmen.

Mario Blunk