

mkc

MKC Michels & Kleberhoff Computer GmbH

Vohwinkeler Str. 58, D-42329 Wuppertal

Tel.: ++49 (0)202 27317 0 Fax: ++49 (0)202 27317 49

Internet: <http://www.mkc-gmbh.de>



Software Handbuch

System SD - PHD 0090

Hinweise:

Die Informationen in diesem Handbuch wurden sorgfältig zusammengestellt und überprüft. Dieses Handbuch wird stetig auf dem aktuellen Zustand gehalten. Jedoch wird von MKC keine Gewähr für fehlerhafte Informationen übernommen.

MKC behält sich das Recht vor, jederzeit ohne weitere Ankündigung technische Änderungen zur Verbesserung der Zuverlässigkeit, der Funktion oder des Designs der Produkte und Überarbeitungen des Handbuchs durchzuführen. Änderungen des Handbuchs zwischen 2 Ausgaben werden im Text nicht markiert.

Das Datum einer Ausgabe bezieht sich auf das Handbuch. Dieses muss nicht mit dem Datum der Änderung der Hardware oder Software übereinstimmen. Bei der Versionsgeschichte wird der Grund für die Handbuch Änderungen genannt.

MKC übernimmt keine Haftung für die Anwendung des hier beschriebenen Produktes. MKC übernimmt weiterhin keine Haftung für Schäden oder Folgeschäden, die durch Verwendung dieses Produktes entstehen. Diese Haftungseinschränkung bezieht sich sowohl auf jeden direkten Abnehmer sowie auf alle seine Kunden und alle Anwender des Produktes.

Es gelten ausschließlich die in diesem Dokument gemachten Zusagen über die Anwendbarkeit des hier beschriebenen Produktes.

Kommentare:

Kommentare oder Korrekturen jedweder Art sind dem Autor jederzeit willkommen. Senden Sie diese bitte an:

**MKC Michels & Kleberhoff Computer GmbH
Vohwinkeler Str. 58
42329 Wuppertal**

oder

info@mkc-gmbh.de

Handbuch Versionen

Änderungen im Handbuch werden durch eine Erhöhung der Ausgabennummer angezeigt. Handbücher, deren Ausgabe durch einen Buchstaben gekennzeichnet ist, sind vorläufige Handbücher und stimmen möglicherweise noch nicht vollständig mit dem endgültigen Produkt überein. Die erste Ausgabe, die nicht mehr als vorläufig anzusehen ist, beginnt mit der Nummerierung „1“.

Handbuch Versionen			
Ausgabe	Änderungen	Datum	
A	Vorläufige Version	15.01.2020	
1	Erste Version, 0090_PHD_05	13.02.2020	
2	Korrekturen	21.02.2020	
3	0090_PHD_06	07.07.2020	
4	0090_PHD_07	30.11.2020	
5	0090_PHD_08	10.12.2020	
6	0090_PHD_09	12.08.2021	
7	0090_PHD_10	23.08.2022	
8	0090_PHD_11	27.07.2023	

Inhaltsverzeichnis

1 EINLEITUNG.....	7
2 AUFBAU.....	9
2.1 Allgemein	9
2.2 Konfiguration.....	9
2.2.1 Netzwerk.....	9
2.2.2 Dateisysteme.....	9
2.2.3 Mkcmod Treiber.....	9
3 SKRIPTE.....	11
3.1 Programmierskripte.....	11
3.2 Testskripte.....	11
3.3 Installationsskript.....	11
3.4 Verifizierung der Installation	12
3.5 Hilfsprogramme.....	12
3.6 Hilfsskripte tap & pap.....	13

Liste der Abbildungen

Liste der Tabellen

Tabelle 1: Netzwerk Konfiguration.....	9
Tabelle 2: Programmierskripte.....	11
Tabelle 3: Testskripte.....	11
Tabelle 4: version.txt der Auslieferysteme.....	12
Tabelle 5: Hilfsprogramme.....	12
Tabelle 6: tap und pap Parameter.....	13

1 Einleitung

Die System SD - PHD 0090 ist eine SD-Karte mit einem Linux-Betriebssystem, das für das Testen und die Programmierung des eNetMaxi Moduls und der MKC1502 Trägerkarte, entwickelt wurde.

Sie ermöglicht die Installation des eNetVario-11 STD Betriebssystems auf das eMMC und beinhaltet Skripte mit denen man sowohl die Hardware als auch die Software testen kann.

Dieses Dokument beschreibt den Aufbau der System SD - PHD 0090 und gibt eine Übersicht über die Installations- und Testskripte.

2 Aufbau

2.1 Allgemein

Die System SD - PHD 0090 beinhaltet ein auf eNetMaxi basierendes Linux-Betriebssystem. Die Einzelheiten dazu können Sie dem „Software Handbuch - Linux Systeme basierend auf eNetMaxi“ entnehmen. Es baut direkt auf dem STD-System des eNetVarios auf, somit sind alle Treiber und Linux-Schnittstellen des eNetVario-11 STD Systems auch in diesem System verfügbar.

2.2 Konfiguration

2.2.1 Netzwerk

Die Ethernet-Schnittstellen werden beim Booten des Systems wie folgt konfiguriert:

Variable	eth0	eth1
MAC-Adresse	00:40:86:12:FF:FF	00:40:86:13:FF:FF
IPv4-Modus	statisch	statisch
IPv4-Adresse	192.168.15.22	192.168.16.22
IPv4-Subnetzmaske	255.255.255.0	255.255.255.0
IPv6	aus	aus

Tabelle 1: Netzwerk Konfiguration

2.2.2 Dateisysteme

Beim Start des Systems wird ein temporäres Dateisystem mit der Größe von 250MB auf dem Arbeitsspeicher des eNetMaxis angelegt und an `/tmp` eingehängt.

Dieser Bereich ist für das Logging der Test- und Installationsskripte, als auch für andere temporäre Dateien gedacht.

Dieser Speicher ist in der `/etc/fstab` Datei mit der Zeile

```
„tmpfs /tmp tmpfs defaults,size=250M 0“
```

konfiguriert.

Das Linuxdateisystem wird beim booten im read-only Modus gestartet.

Mit dem read-only Modus und dem temporären Speicher wird eine längere Lebensdauer der SD-Karte gewährleistet, da keine Daten beim Testen und Programmieren geschrieben werden.

Falls Sie den read-only Modus ausschalten möchten, müssen Sie die SD-Karte an einem PC im nicht read-only Modus einhängen und in der Datei `/etc/fstab` die „ro“ Option in der Zeile, die für das Linuxdateisystem zuständig ist, löschen.

2.2.3 Mkcmod Treiber

Mkcmod ist ein Treiber, der dem Benutzer den Zugang zu mehreren Funktionen des Moduls zur Verfügung stellt und den SPI-Speicher für die Benutzung unter Linux konfiguriert.

Der reguläre Modus des Treibers (`mode = 0`) setzt einen Schreibschutz auf den SPI-Speicher um vor ungewollten Überschreiben zu schützen.

Die System SD - PHD 0090 startet den Treiber in dem update-Modus (`mode = 4096`). In diesem Modus hat der SPI-Speicher keinen Schreibschutz gesetzt. Dies ist nötig um den Bootloader auf den SPI-Speicher zu schreiben.

Der Treiber wird mit dem `/etc/init.d/phd0090` init-Skript gestartet.

3 Skripte

3.1 Programmierskripte

Unter `/opt/mkc/scripts` finden Sie Skripte, die für die Programmierung und Konfiguration des Systems zuständig sind. Ein Skript ist dann korrekt abgelaufen, wenn es bei der Beendigung den exit-Status 0 zurückgibt.

Wenn ein Skript einen Wert zurückliefern soll, wird dieser unter `/tmp/pap_result` gespeichert.

Manche Skripte können die vorherige Ausführung anderer Skripte voraussetzen.

z.B. Skript `023_install_mlo.sh` braucht die Dateien, die von Skript `020_prepare_spi_files.sh` vorbereitet werden.

Beispiele:

Dateiname	Beschreibung
001_extract_mac0_from_spi.sh	Liest die Bootloader Umgebungsdatei vom SPI-Speicher aus und extrahiert daraus (falls vorhanden) die MAC-Adresse der ersten Ethernet-Schnittstelle
003_get_ti_mac0.sh	Liest die TI-MAC-Adresse aus
012_clean_target_device.sh	Löscht alle Daten, Partitionstabellen, etc. vom eMMC
024_install_uboot.sh	Schreibt das uBoot Image an die entsprechende Stelle des SPI-Speichers
112_mount_rootfs.sh	Hängt die zweite Partition (Linux Partition) des eMMCs an „/media/rootfs“ ein
117_unpack_rootfs_to_rootfs.sh	Entpackt die tar.gz Datei, die den Kernel, das Devicetree und die Module des Systems beinhaltet, auf die eingehängte zweite Partition des eMMCs

Tabelle 2: Programmierskripte

3.2 Testskripte

Unter `/opt/mkc-test/scripts` finden Sie die Testskripte. Diese sind für das Testen des Systems und der Hardware zuständig.

Ein Skript ist dann korrekt abgelaufen, wenn es bei der Beendigung den exit-Status 0 zurückgibt.

Wenn ein Skript einen Wert zurückliefern soll, wird dieser unter `/tmp/tap_result` gespeichert.

Beispiele:

Dateiname	Beschreibung
012_test_eth0.sh	Testet die eth0 Schnittstelle, indem es die IP-Adresse 192.168.15.24 anpingt. Setzt eine physikalische Verbindung und ein ping-fähiges Gerät an der IP-Adresse 192.168.15.24 voraus.
057_test_probe_pcal6416a.sh	Testet ob der pcal6416a Chip ordnungsgemäß an den I2C Bus angeschlossen ist.
032_test_usb_master.sh	Testet die USB Schnittstelle (USB Typ A – Stecker). Setzt ein verbundenes USB-Massenspeichergerät und initialisierte USB Treiber voraus.

Tabelle 3: Testskripte

3.3 Installationskript

Nach dem booten von dem 0090_PHD, finden Sie unter `/opt/mkc-install` das eMMC-Installations Skript „`install_to_emmc.sh`“, das für die manuelle Installation des Systems auf das eMMC benutzt werden kann. Dafür verwendet es die oben beschriebenen Programmierskripte.

Für die Installation werden die tar.gz Dateien unter `/opt/mkc-install` und die Dateien unter `/opt/mkc/files` benutzt.

Falls es auf einem Gerät ausgeführt wird, auf dem sich schon bereits ein System auf dem eMMC befindet, wird das vorhandene System gelöscht und vom neuen überschrieben! Es wird auch der Bootloader auf dem SPI-Speicher überschrieben!

3.4 Verifizierung der Installation

Ob die Installation des Systems auf das eMMC erfolgreich war, können Sie mithilfe der version.txt Datei verifizieren.

1. Booten Sie das neue System.
2. Lesen Sie die Datei version.txt unter /etc/mkc aus.
3. Ist der Inhalt dieser Datei identisch zu dem dazugehörigen Text aus der Tabelle 4, dann war die Installation erfolgreich.

PHD Version	version.txt des Ausliefersystems
0090_PHD_05	Fr 30. Aug 12:42:04 CEST 2019 SYSTEM: 1.1.4.1 (eNetVario-11 Rev1 STD) CARRIER: 1.1.2.1 (MKC1502 Rev2 STD) MODULE: 1.1.2.1 (eNetMaxi Rev2 STD) LINUX: 1.1.2.1 BUILD: 1.1.0.9
0090_PHD_06	Mi 20. Mai 10:12:18 CEST 2020 ADDON: 1.1.5.1 (mkc_linuxaddon.enetvario.std_1_1) ROOTFS: 1.1.3.1 (mkc_linuxroot.mkc1502_1_1) BOOT: 1.1.3.1 (mkc_linuxboot.enetmaxi_1_1) LINUX SDK: 1.2.1.1 (mkc_linuxsdk_1_2) BUILD: 1.1.0.10 (mkc_linuxbuild_1_1)
0090_PHD_07	Mo 23. Nov 10:33:08 CET 2020 ADDON: 1.1.6.1 (mkc_linuxaddon.enetvario.std_1_1) ROOTFS: 1.1.4.1 (mkc_linuxroot.mkc1502_1_1) BOOT: 1.1.4.1 (mkc_linuxboot.enetmaxi_1_1) LINUX SDK: 1.2.2.1 (mkc_linuxsdk_1_2) BUILD: 1.1.0.12 (mkc_linuxbuild_1_1)
0090_PHD_08	Do 10. Dez 11:35:16 CET 2020 ADDON: 1.1.7.1 (mkc_linuxaddon.enetvario.std_1_1) ROOTFS: 1.1.5.1 (mkc_linuxroot.mkc1502_1_1) BOOT: 1.1.4.1 (mkc_linuxboot.enetmaxi_1_1) LINUX SDK: 1.2.2.1 (mkc_linuxsdk_1_2) BUILD: 1.1.0.12 (mkc_linuxbuild_1_1)
0090_PHD_09	Do 12. Aug 11:13:20 CEST 2021 ADDON: 1.1.8.1 (mkc_linuxaddon.enetvario.std_1_1) ROOTFS: 1.1.6.1 (mkc_linuxroot.mkc1502_1_1) BOOT: 1.1.4.1 (mkc_linuxboot.enetmaxi_1_1) LINUX SDK: 1.2.2.1 (mkc_linuxsdk_1_2) BUILD: 1.1.0.15 (mkc_linuxbuild_1_1)
0090_PHD_10	Do 11. Aug 13:15:51 CEST 2022 ADDON: 1.1.9.1 (mkc_linuxaddon.enetvario.std_1_1) ROOTFS: 1.1.7.1 (mkc_linuxroot.mkc1502_1_1) BOOT: 1.1.5.1 (mkc_linuxboot.enetmaxi_1_1) LINUX SDK: 1.2.2.1 (mkc_linuxsdk_1_2) BUILD: 1.1.0.16 (mkc_linuxbuild_1_1)
0090_PHD_11	Mi 26. Jul 14:51:23 CEST 2023 ADDON: 1.1.10.1 (addon.enetvario_1_1) ROOTFS: 1.1.8.1 (root.mkc1502_1_1) BOOT: 1.1.5.1 (mkc_linuxboot.enetmaxi_1_1) LINUX SDK: 1.2.2.1 (mkc_linuxsdk_1_2) BUILD: 1.1.1.1 (build_1_1)

Tabelle 4: version.txt der Ausliefersysteme

3.5 Hilfsprogramme

Manche Programmier- und Testskripte benötigen zusätzliche Programme, die im normalen System nicht mitgeliefert werden oder sie benötigen eine besondere Version eines bestimmten Programms.

Diese Programme befinden sich dann unter `/opt/bin`.

Beispiele:

Programmname	Beschreibung
mkenvimage	Tool aus dem uBoot Toolkit. Konvertiert eine Bootloader-Umgebungsdatei im Text-Format zu einer Datei in binär-Format für den SPI-Speicher.
i2cdetect	Ein Programm, das den I2C Bus scannt, um alle daran verbundenen Geräte zu finden.

Tabelle 5: Hilfsprogramme

3.6 Hilfsskripte tap & pap

Unter `/opt/mkc` und `/opt/mkc-test` befinden sich die Hilfsskripte `pap.sh` und `tap.sh`.

Diese Skripte erlauben es nur mit der Skript Nummer, das richtige Test- oder Programmierskript auszuführen.

Beim Ausführen geben sie eine kurze Beschreibung des Skriptes aus und speichern das Ergebnis des Skriptes (OK/ERROR/RÜCKGABEWERT) unter `/tmp/pap_result` bzw. `/tmp/tap_result` ab.

Diese Skripte benötigen folgende Parameter:

Name	mögliche Werte	erforderlich	Beschreibung	Standardwert
func	Integer	ja	die Nummer des Skriptes	-
target	„emmc“ / „sdc“	nein	das Ziel-Gerät	„emmc“
input	String	nein	Argumente / Input für das Skript	-

Tabelle 6: tap und pap Parameter

Beispielaufruf: `./pap.sh func=0 target=sdc input=testinput`