Handbuch Administration / Administration Manual

Administrationsanleitung für ACD Geräte mit Android™*/
Administration Instructions for ACD Devices with Android™*

Version: 1.22

© Alle Rechte vorbehalten – ACD Elektronik GmbH
Dieses Dokument darf ohne Zustimmung weder vervielfältigt noch Dritten zugänglich gemacht werden.
Bitte beachten, dass in der Dokumentation ggfs. bereits abgekündigte Produkte enthalten sind.

© Copyright – ACD Elektronik GmbH
This document may not be reproduced or made available to third parties without permission.
Please note that the documentation may contain products that have already been discontinued.

---

* Eingetragenes Warenzeichen – Android™ – Android ist eine Marke von Google LLC
* Registered trademark – Android™ – Android is a trademark of Google LLC

Deutsch

1 Übersicht 4
1.1 Ziel und Umfang dieses Handbuchs 4
1.2 Abkürzungsverzeichnis 4
1.3 Betriebssystem Android 4
1.3.1 ACD SystemApp 4
1.3.2 ACD TerminalSettings 4
1.4 Gerätetypen – Funktionen und Unterschiede 6
1.4.1 Liste der ACD Geräte mit Android 6
1.4.2 Ein- und Ausschalten 6
1.5 Startprozess 8
2 Gerätefunktionen der Mobilen Handheld Computer 8
2.1 Akku 8
2.1.1 Akkuwechsel 8
2.1.2 Optimiertes Laden 8
2.2 Tastatur 9
2.2.1 Benutzerdefinierte Tastaturlayouts 9
2.3 Scannereinheit 11
2.3.1 Scanner- und Imagertypen 11
2.3.2 Betriebsvarianten der Scaneinheit 12
2.3.3 Scannerkonfiguration 12
2.4 Ethernet 12
2.5 Wireless LAN (WLAN) 12
2.6 BT-kompatibler Nahbereichsfunk 13
2.7 Mobile Datenverbindung 13
2.8 NFC/RFID 13
3 Gerätefunktionen der Staplerterminals 13
3.1 Ethernet 13
3.2 Wireless LAN (WLAN) 13
3.3 BT-kompatibler Nahbereichsfunk 14
4 Zugang über ADB 14
4.1 ADB über Netzwerk permanent erlauben 15
4.2 Platform Tools 15
4.3 Android USB Gerätetreiber für Windows 15
4.4 Android MTP Gerätetreiber für Windows 16
4.5 Zugriff über ADB beim M2Smart®SE 16
4.6 Zugriff über ADB beim MFT10/MFT12 16
5 Fernwartung mit SCRCPY bei Mobilen Handheld Computern 17
5.1 Fernwartung mit SCRCPY über Netzwerk 17
6 Betriebssystemversionen 17
7 NTP Server bei Mobilen Handheld Computern 18
8 ACD KioskMode 18
9 Nachinstallationen bei Mobilen Handheld Computern 18
10 Over the Air (OTA) und Firmware Update 18
10.1 Android Update Paket 18
10.2 Firmware Update 19
11 Abbildungsverzeichnis 19
12 Tabellenverzeichnis 19

English

13 Overview 20
13.1 Goal and scope of this manual 20
13.2 List of abbreviations 20
13.3 Android operating system 20
13.3.1 ACD SystemApp 20
13.3.2 ACD TerminalSettings 20
13.4 Device types – functions and differences 22
13.4.1 List of the ACD devices with Android 22
13.4.2 Switching on and off 22
13.5 Start process 24
14 Device functions of the Mobile Handheld Computer 24
14.1 Rechargeable battery 24
14.1.1 Rechargeable battery change 24
14.1.2 Optimized charging 24
14.2 Keyboard 25
14.2.1 User-defined keyboard layout 25
14.3 Scanner unit 27
14.3.1 Scanner and Imager types 27
14.3.2 Operating variants of the scan unit 28
14.3.3 Scanner configuration 28
14.4 Ethernet 28
14.5 Wireless LAN (WiFi) 28
14.6 BT-compatible close-range radio 29
14.7 Mobile data connection 29
14.8 NFC/RFID 29
15 Device functions of the vehicle terminals 29
15.1 Ethernet 29
15.2 Wireless LAN (WiFi) 29
15.3 BT-compatible close-range radio 29
16 Access via ADB for the mobile devices 30
16.1 Permanently allowing ADB via network 30
16.2 Platform tools 31
16.3 Android USB device driver for Windows 31
16.4 Android MTP device driver for Windows 31
16.5 Access via ADB for M2Smart®SE 32
16.6 Access via ADB for MFT10/MFT12 32
17 Remote maintenance with SCRCPY for the Mobile Handheld Computer 33
17.1 Remote maintenance with SCRCPY via network 33
18 Operating system versions 33
19 NTP Server for the Mobile Handheld Computer 34
20 ACD KioskMode 34
21 After-the-fact installations for the Mobile Handheld Computer 34
22 Over the Air (OTA) and Firmware Update 34
22.1 Android Update Package 35
22.2 Firmware Update 35
23 List of figures 35
24 List of tables 35

1 Übersicht

1.1 Ziel und Umfang dieses Handbuchs

Dieses Dokument versteht sich als Betriebsanleitung für die Administration der Grundfunktionalität von ACD Geräten mit Android. Es werden Themen behandelt, die ein grundlegendes Verständnis für die Funktionsweise von Android vermitteln sollen, sowie praktische Anweisungen für immer wiederkehrende administrative Aufgaben geben.

Ziel ist es, dem Anwender das Wissen an die Hand zu geben, ACD Geräte mit Android grundlegend auf den Einsatz einer kundenspezifischen Applikation vorzubereiten. Applikationsspezifische Einstellungen sind nicht Thema dieser Abhandlung; hierfür wird auf die jeweilige gesonderte Dokumentation verwiesen.

Beschreibungen, die auf Programme oder betriebssystemeigene Einstellungsdialoge verweisen, beziehen sich auf deutsche Desktop-Spracheinstellungen.

1.2 Abkürzungsverzeichnis

Abkürzung	Beschreibung
OS	Betriebssystem/Operating System
ADB	Android Debug Bridge
SCRCPY	Screen Copy
MTP	Media Transfer Protocol

Tabelle 1: Abkürzungsverzeichnis

1.3 Betriebssystem Android

Neben anderen Betriebssystemen kommt bei ACD Geräten das Betriebssystem Android (durch ACD modifiziert) zum Einsatz.

Android enthält einen Linux-Kernel, ist jedoch keine klassische Linux-Distribution, wie für Desktop und Server bekannt. Teilweise sind Grundeigenschaften, wie sie bei einem System der UNIX Familie erwartet würden, stark verändert. Trotzdem sind viele Eigenschaften vorhanden, die mit anderen zahlreichen Embedded-Linux-Distributionen vergleichbar sind.

1.3.1 ACD SystemApp

Die ACD SystemApp ist bereits auf den Mobilen Handheld Computern vorinstalliert. Über diese können verschiedene Informationen des Geräts mit Android angezeigt und konfiguriert werden. Im Menü Systemupdate können Updates installiert werden. Für die installierten ACD Apps lassen sich die Lizenzschlüssel im Menü Lizenzen aktivieren.

1.3.2 ACD TerminalSettings

Über die App ACD TerminalSettings, welche auf den Staplerterminals zum Einsatz kommt, können verschiedene Informationen über das Gerät ausgelesen, sowie Einstellungen getätigt werden.

1.3.2.1 System
Im Bereich System werden Informationen zur aktuellen Spannung und Helligkeit angezeigt. Die Einstellung Antennenumschaltung hat derzeit noch keine Funktion.

Kalibrierung
Der Touchscreen wird bereits vorkalibriert ausgeliefert. Sollten dennoch Probleme bei der Eingabe auftreten, kann über die Schaltfläche Kalibrierung eine erneute Einstellung des Touchscreens vorgenommen werden. Folgend ist der Ablauf beschrieben:

- Auf die Schaltfläche Kalibrierung tippen
- Einige Sekunden warten bis der Vorgang beginnt
- Die vier Zielscheiben nacheinander mittig mit einem Stylus antippen
- Nach der vierten Zielscheibe einige Sekunden warten, bis die Konfiguration übernommen wurde. Sobald der Bildschirm kurz blinkt, ist die Konfiguration gespeichert. Bitte vorher keine Eingaben tätigen!

Erfolgt beim Kalibrieren 15 Sekunden lang keine Eingabe bricht der Vorgang automatisch ab. Bei einer Fehleingabe – wenn eine Zielscheibe nicht genau getroffen wurde – kann der Vorgang durch Warten bewusst ohne Speicherung beendet werden.

1.3.2.2 Informationen
Im Bereich Informationen werden die Hardwarerevision, aktuelle Leistungsaufnahme, Firmwareversion und die Seriennummer des Geräts angezeigt.

1.3.2.3 Batterie
Im Bereich Batterie werden die Batterieladung sowie deren Batterielaufzeit angezeigt. Diese Anzeige gibt einen ungefähren Zustand wieder und bezieht sich auf eine Batterie mit maximaler Ladekapazität.

1.3.2.4 Tastenfunktionen
Im Bereich Tastenfunktionen können die fünf Hardwaretasten der Staplerterminals F1, -, OK, + und F2 mit Aktionen belegt werden.

Ferner können die Tasten bzw. Tastenkombinationen zum Einschalten und Herunterfahren des Staplerterminals sowie das Verhalten des Staplerterminals bei Zündung (Einschalten, Ausschalten) konfiguriert werden.

1.3.2.5 Über
Im Bereich Über wird die Versionsnummer der App ACD TerminalSettings ausgegeben.

1.4 Gerätetypen – Funktionen und Unterschiede

1.4.1 Liste der ACD Geräte mit Android

Gerät	Produkt-version … und höher	Betriebssystem	Prozessor	Taktrate	RAM-Speicher	Flash-Speicher abzüglich OS
M2Smart® HMI10TE Wired LE	3.2.0	Industrial+ für Android	Qualcomm®	2x Core 2,1 GHz und 4x Core 1,8 GHz	4 GB	64 GB
M210CS[1]	2.6.1	Industrial+ für Android	NXP i.MX8M Mini	1,8 GHz Quad Core	4 GB	32 GB
M260TE	1.8	Industrial+ für Android	NXP i.MX6	1 GHz Dual Lite	1 GB	8 GB
M266SE	1.7	Industrial+ für Android	NXP i.MX6	1 GHz Dual Lite	1 GB	8 GB
M270	1.10	Industrial+ für Android	NXP i.MX6	1 GHz Dual Lite	1 GB	8 GB
M270SE	1.13	Industrial+ für Android	NXP i.MX8M Mini	1,8 GHz Quad Core	2 GB	16 GB
M2Smart®	1.6	Industrial+ für Android	Qualcomm® Snapdragon 410	1,2 GHz Quad Core	1 GB	8 GB
M2Smart®SE	1.17	Industrial+ für Android	NXP i.MX8M Mini	1,8 GHz Quad Core	2 GB	16 GB
M2Smart®SE 10 Zoll	1.11	Industrial+ für Android	NXP i.MX8M Mini	1,8 GHz Quad Core	4 GB	16 GB
MFT10/ MFT12	1.3.0	Android 6.0.1	Intel® Celeron® N2930	1,83 GHz Quad Core	4 GB	64 GB

Tabelle 2: ACD Geräte mit Android

Details zu den Varianten der ACD Geräte mit unterschiedlichen Speichergrößen sind aus den technischen Daten des jeweiligen Geräts zu entnehmen.

1.4.2 Ein- und Ausschalten

Die ACD Geräte lassen sich über die Tastatur Ein- und Ausschalten. Dabei wird zwischen folgenden Modi unterschieden:

Einschalten
Das Gerät war zuvor vollständig ausgeschaltet.
Das Betriebssystem fährt beim Einschalten komplett neu hoch

Geordnetes Herunterfahren
Das Betriebssystem wird ordnungsgemäß heruntergefahren und das Gerät danach stromlos geschaltet.

Idle Mode
Das Gerät wird in einen Stromsparmodus versetzt. Einige notwendige Systemkomponenten werden am Leben erhalten, ein Wiederaufwecken des Geräts erfolgt zügig, da das Betriebssystem nicht erneut gebootet werden muss. Allerdings ist das Gerät in diesem Fall nicht stromlos und somit kann dieser Modus nur eine gewisse Zeit gehalten werden (abhängig von der Akkukapazität des Geräts).
Das Gerät kann mit der gleichen Tastenkombination, wie in nachfolgender Tabelle beschrieben, vom Idle Mode wiederaufgeweckt werden.

[1] Kundenspezifisches Produkt

Hard Reset
Bei einem Hard Reset wird das Gerät stromlos geschaltet. Das Betriebssystem versucht geordnet herunterzufahren, sollte dies nicht möglich sein wird nach einer definierten Zeit das Gerät trotzdem stromlos geschaltet.

Gerät	Einschalten	Geordnetes Herunterfahren	Idle	Hard Reset
M210CS[1]	**C** und **Enter** für zwei Sekunden gedrückt halten.	**C** und **Enter** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	**C** und **Enter** für 300 ms gedrückt halten und wieder loslassen.	**C** und **Enter** für zehn Sekunden gedrückt halten.
M260TE	**C** und **Enter** für zwei Sekunden gedrückt halten.	**C** und **Enter** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	Wird aktuell nicht unterstützt.	**C** und **Enter** für acht Sekunden gedrückt halten.
M266SE	**C** und **Enter** für zwei Sekunden gedrückt halten.	**C** und **Enter** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	Wird aktuell nicht unterstützt.	**C** und **Enter** für acht Sekunden gedrückt halten.
M270	**C** und **Enter** für zwei Sekunden gedrückt halten.	**C** und **Enter** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	Wird aktuell nicht unterstützt.	**C** und **Enter** für acht Sekunden gedrückt halten.
M270SE	**C** und **Enter** für zwei Sekunden gedrückt halten.	**C** und **Enter** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	**C** und **Enter** für 300 ms gedrückt halten und wieder loslassen.	**C** und **Enter** für zehn Sekunden gedrückt halten.
M2Smart®	**On/Off** Taste für zwei Sekunden gedrückt halten.	**On/Off** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	**On/Off** für 300 ms gedrückt halten und wieder loslassen.	**On/Off** Taste für sieben Sekunden gedrückt halten.
M2Smart®SE	**On/Off** Taste für zwei Sekunden gedrückt halten.	**On/Off** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	**On/Off** für 300 ms gedrückt halten und wieder loslassen.	**On/Off** Taste für zehn Sekunden gedrückt halten.
M2Smart®SE 10 Zoll	**On/Off** Taste für zwei Sekunden gedrückt halten.	**On/Off** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	**On/Off** für 300 ms gedrückt halten und wieder loslassen.	**On/Off** Taste für zehn Sekunden gedrückt halten.
M2Smart® HMI10TE Wired LE	**On/Off** Taste für drei Sekunden gedrückt halten.	**On/Off** gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	**On/Off** für 300 ms gedrückt halten und wieder loslassen.	**On/Off** Taste für zwanzig Sekunden gedrückt halten.
MFT10/ MFT12	**OK** Taste für zwei Sekunden gedrückt halten.	**F1** Taste gedrückt halten, bis Ausschaltmenü erscheint. Über Ausschaltmenü herunterfahren.	Wird aktuell nicht unterstützt.	**Reset-Taster** für mindestens vier Sekunden gedrückt halten.

Tabelle 3: Ein-/Ausschalten

1.5 Startprozess

Der Startprozess bei Android besteht aus drei wesentlichen Phasen.

Bootloader
Der Bootloader ist das initiale Programm, das direkt nach dem Einschalten des Gerätes gestartet wird. Dabei werden erste notwendige Hardwareinitialisierungen vorgenommen. Darüber hinaus ist der Bootloader für die Übergabe der Kernel-Parameter und das Starten des Linux-Kernels verantwortlich.

Kernel
Der Kernel ist die Schnittstelle zwischen Hardware und Android. Er prüft, welche Hardware zur Verfügung steht und sucht passend dazu die vorhandenen Gerätetreiber.

Android
Sobald das animierte ACD-Logo beim Startprozess zu sehen ist, ist der Kernel vollständig geladen und es wird Android (User-Space) geladen. Hierbei werden Android spezifische Dienste gestartet. Nach dem vollständigen Startprozess ist der Homescreen von Android sichtbar.

2 Gerätefunktionen der Mobilen Handheld Computer

2.1 Akku

Die Stromversorgung der ACD Geräte wird über einen Akku sichergestellt. Applikationen können vom Betriebssystem alle notwendigen Informationen zum Akku erhalten.

2.1.1 Akkuwechsel

Backupfunktionen für die Uhr und eine Versorgungsüberbrückung während des Akkuwechsels werden von großen Kondensatorkapazitäten übernommen, die binnen einiger Minuten vom Akku nachgeladen werden. Aus diesem Grund können nicht beliebig viele Akkuwechsel während dem laufenden Betrieb unmittelbar hintereinander durchgeführt werden.

Die Uhr führt auch ohne Akku die Zeitzählung einige Tage fort. Der Akkuwechsel, bei geladenen Stützkondensatoren, ist innerhalb von ca. fünf Sekunden zu vollziehen.
In dieser Zeit wird das Display aus Stromspargründen deaktiviert. Alle übrigen Gerätefunktionen, insbesondere Prozessor, Speicher und WLAN bleiben funktionstüchtig, was das Prinzip des Akku-Schnellwechsels im laufenden Betrieb unterstreicht.

Es ist zu beachten, dass beim M266SE kein Schnellwechselakku vorhanden ist und somit der Akkuwechsel im laufenden Betrieb nicht möglich ist.

2.1.2 Optimiertes Laden

Hinweis: Diese Möglichkeit steht ausschließlich auf dem Gerät **M2Smart®SE** zur Verfügung.

In der ACD SystemApp unter dem Reiter Akku-Optimierung ist der Ladeoptimierungs-Schalter zu finden, über welchen ein optimiertes Ladeverfahren aktiviert werden kann.
Dabei werden die Ladegrenzwerte automatisch auf einen oberen Wert von 80 % und einen unteren Wert von 60 % gesetzt.
Der integrierte Akku wird somit beim Ladevorgang bis zum oberen Grenzwert geladen und anschließend nicht weiter nachgeladen. Ein erneuter Ladevorgang beginnt erst, wenn der Ladezustand des Akkus unter 60 % fällt bzw. 59 % erreicht.

Dieses Verfahren verringert die Belastung des Akkus durch dauerhaft hohe Ladezustände und unterstützt so eine längere Akkulebensdauer.

2.2 Tastatur

ACD Geräte werden über eine Kombination mehrerer Tasten aktiviert bzw. ein-/ausgeschaltet (siehe Kapitel 1.4.2).

ACD Geräte verfügen über auf den jeweiligen spezifizierten Einsatzzweck optimierte Tastaturen.
Manche sind zweistufig aufgebaut. Es existiert eine zweite Tastaturebene, die mit einer Funktionstaste aktiviert werden kann. Diese muss zusammen mit der gewünschten Taste ausgelöst werden.

Gerät	Taste für zweite Tastaturebene
M210CS[1]	C-Taste
M260TE	C-Taste
M266SE	C-Taste
M270	C-Taste
M270SE	C-Taste

Tabelle 4: Standardbelegung zweite Tastaturebene

Sowohl die erste Ebene (ohne Umschaltung) als auch die zweite Ebene kann in ihrer Funktion pro Taste für das Gerät frei definiert werden. Darunter wird jedoch nicht nur die Hinterlegung eines anderen Zeichens verstanden, welches nach Druck der Taste ausgelöst ("eingeschleift") wird. Vielmehr stehen folgende Möglichkeiten zur Verfügung:

- Einschleifung von frei wählbaren ASCII Zeichen anstatt der Standardbelegung
- Einschleifung von steuernden Zeichen (ESC, TAB,...)
- Auslösen gerätespezifischer Funktionen (z. B. Scanner)

2.2.1 Benutzerdefinierte Tastaturlayouts

Benutzerdefinierte Tastaturlayouts stehen momentan nur für die M2Smart®, M2Smart®SE, M260TE, M270 und M270SE Geräte zur Verfügung.

Zur Konfiguration der Tastatur beim M2Smart®SE und M270SE wird auf das separate Dokument „Tastaturkonfiguration-unter-Android_Keyboard-Configuration-under-Android_de_en“ verwiesen.

Die Konfiguration der M260TE und M270 Geräte wird nachfolgend beschrieben.
Die Hinterlegung der Tastaturdefinition erfolgt in der Datei *keyb_config.conf* die sich im Ordner */sdcard/ACD/keyboard/* befindet. Diese Datei kann auf dem Gerät oder remote (z. B. mit SCRCPY) mit der Totalcommander App bearbeitet werden. Auch ein Austausch der Datei durch eine angepasste Datei ist möglich.

Die Umdefinition einer Taste erfolgt nach dem Schema
Taste = Funktion

Für jede Zuordnung wird eine eigene Zeile verwendet, der Bezeichner ist immer klein geschrieben und Funktionsnamen müssen hochgestellt angegeben werden. ASCII Zeichen werden mit Hochkommas, Funktionen zusätzlich zum Hochkomma mit Kleiner-/Größerzeichen umgeben. Bei Abweichungen von diesen Vorgaben wird die Zuordnung nicht erkannt und die Standardbelegung verwendet.

Schlüssel (mit Beispiel)	Erklärung	Wirkung
k_1 = "a	Fehlerhafte Definition	Standardbelegung wird verwendet
k_2 = "<NOP>"	Taste 2 wird umdefiniert	Keine Reaktion
k_f4 = "A"	Taste F4 wird umdefiniert	„A“ anstatt „F4“
k_scr = "<RSCANNER>"	Rechte Scannertaste	Aktiviert den Scanner
s_3 =	Leere Definition	Standardbelegung wird verwendet
k_s3 = "<BACK>"	Taste S3 wird umdefiniert	Android Backbutton anstatt Standard

Tabelle 5: Tastaturkonfigurationsdatei

Die Bezeichner für die vorhandenen Tasten des Gerätes sind über die Geräte vereinheitlicht. Nachfolgende Tabelle gibt eine beispielhafte Übersicht.

Bezeichner	Taste
k_x	Eine Taste x der ersten Tastaturebene (ohne Shift)
s_x	Eine geshiftete Taste x (zweite Tastaturebene)
k_sx	Sondertaste x
s_sx	Geshiftete Sondertaste x
k_1	Taste Eins
s_2	Taste Shift-Zwei
k_f1	Taste F1
k_s1	Taste Pfeil auf
k_s2	Taste S
k_s3	Taste Pfeil ab
k_scl	Scannertaste links
s_sch	Scannertaste Griff mit Shift (zweite Tastaturebene)

Tabelle 6: Tastaturkonfigurationsdatei Tastenbezeichner

Eine Liste der verfügbaren Funktionsbezeichner, welche den oben aufgeführten Tasten zugeordnet werden können, gibt die nachfolgende Tabelle. Diese sind in folgender Weise für die Verwendung in der **keyb_config.conf** zu formatieren:

Tastenbezeichner = "<FUNKTIONSNAME>"
Beispiele:

- k_f1 = "<F1>"
- k_scl = "<LSCANNER>"
- s_s3 = "<DEGREE>"

Funktionsname	Funktion	Funktionsname	Funktion
NOP	Deaktiviert die Taste	**UP**	Nach oben
LSCANNER **RSCANNER** **HSCANNER** **SSCANNER**	Aktiviert den Scanner, der erste Buchstabe kennzeichnet die Scannertaste (Links, Rechts, Griff, S-Taste)	**DOWN**	Nach unten
		RIGHT	Nach rechts
		LEFT	Nach links
		END	Zum Ende
VOLUMEDOWN	Lautstärke verringern	**MOVEHOME**	Zu Position 1
VOLUMEUP	Lautstärke erhöhen	**PAGEUP**	Seite nach oben
MUTE	Gerät stummschalten	**PAGEDOWN**	Seite nach unten
CAMERA	Kamera auslösen (nur in Kameraapp möglich)	**PLUS**	Plus Zeichen
POWER	Ausschaltdialog öffnen	**COMMA**	Komma Zeichen
RECENTAPPS	Android App Übersicht	**MINUS**	Minus Zeichen
HOME	Android Home Button	**DOT**	Punkt Zeichen
BACK	Android Zurück Button	**BACKSLASH**	Backslash Zeichen
FORWARD	Vorwärts (Browser)	**SEMICOLON**	Semikolon Zeichen
NEXT	Nächster Titel (Medien)	**EQUAL**	Gleich Zeichen
BRIGHTNESS DOWN	Bildschirmhelligkeit senken	**SLASH**	Slash Zeichen
BRIGHTNESSUP	Bildschirmhelligkeit eröhen	**F1**	Funktionstaste 1
STOP	Stop (Medien)	**F2**	Funktionstaste 2
PLAY	Abspielen (Medien)	**…**	…
PAUSE	Pause (Medien)	**F12**	Funktionstaste 12
MENU	Menu öffnen (in einer App)	**GAUML**	Taste Ä
SELECT	Auswählen	**AUML**	Taste ä
CLEAR	Feld löschen (Textfeld)	**GOUML**	Taste Ö
INSERT	Einfügentaste	**OUML**	Taste ö
DELETE	Löschentaste	**GUUML**	Taste Ü
BACKSPACE	Rücktaste	**UUML**	Taste ü
CTRL	Controltaste	**ESZETT**	Taste ß
ALT	Alttaste	**PARAGRAPH**	Taste §

Funktionsname	Funktion	Funktionsname	Funktion
ENTER	Entertaste	**MIKRO**	Taste µ
TABBACK	Tab zurück Taste	**CHARTABSET**	Character Tab Set Zeichen
CAPSLOCK	Feststelltaste	**DEGREE**	Grad ° Zeichen
SHIFT	Shifttaste	**SUP2**	„Hoch 2“ Zeichen
ESC	Escapetaste	**SUP3**	„Hoch 3“ Zeichen
TAB	Tabulatortaste	**EURO**	€ Zeichen

Tabelle 7: Tastaturkonfigurationsdatei Funktionen

Einige Funktionstasten können nur mit bestimmten Apps verwendet werden (z. B. Kamera).
Um die Konfiguration auf Standard zurückzusetzen muss die keyb_config.conf gelöscht und das Gerät neugestartet werden. Ein Werksreset erzielt dasselbe Ergebnis.

Es werden nur Standard ASCII Zeichen ausgewertet, Unicode Zeichen (z. B. das € Zeichen) müssen über die Funktionstabelle angegeben werden.

ASCII Tabelle mit unterstützten Zeichen:
https://de.wikipedia.org/wiki/American_Standard_Code_for_Information_Interchange#ASCII-Tabelle

2.3 Scannereinheit

In den ACD Geräten ist in der Regel ein Scanner integriert. Auf den Geräten sind aufgrund der verschiedenen Tastaturvarianten unterschiedliche Tasten als Standardbelegung für die Auslösung des Scanners konfiguriert:

Gerät	Scannertaste
M210CS[1]	Tasten an der linken und rechten Gehäuseseite. S-Taste auf dem Tastaturfeld (Tastaturkonfiguration beachten).
M260TE	Tasten an der linken und rechten Gehäuseseite. S-Taste auf dem Tastaturfeld (Tastaturkonfiguration beachten).
M266SE	Weiße und Orange Funktionstaste(n) auf dem Tastaturfeld.
M270	Tasten an der linken und rechten Gehäuseseite. S-Taste auf dem Tastaturfeld (Tastaturkonfiguration beachten).
M270SE	Tasten an der linken und rechten Gehäuseseite. S-Taste auf dem Tastaturfeld (Tastaturkonfiguration beachten).
M2Smart®	Tasten an der linken und rechten Gehäuseseite.
M2Smart®SE	Tasten an der linken und rechten Gehäuseseite.

Tabelle 8: Standardbelegung Scannertasten

2.3.1 Scanner- und Imagertypen

Je nach Mobilem Handheld Computer und Einsatzzweck können unterschiedliche Typen von Scanmodulen in einem ACD Gerät integriert werden:

Scanmodul	Beschreibung
SE 965	1D-Scanner
SE 1524	1D-Longrange-Scanner
SE 4500	2D Imager
SE 4710	2D Imager
SE 4750SR	2D-Shortrange Imager
SE 4750MR	2D-Midrange Imager
SE 4770	2D-Shortrange Imager
SE 4850	2D-Longrange Imager
SE 5500	2D-Advanced-Range Imager
EX 25	2D-Longrange Imager

Tabelle 9: Scanner- und Imagertypen

Details zu den Varianten der ACD Geräte mit unterschiedlichen Scanmodulen sind aus den technischen Daten des jeweiligen Geräts zu entnehmen.

2.3.2 Betriebsvarianten der Scaneinheit

Das im ACD Gerät integrierte Scanmodul kann auf mehrere Arten genutzt werden.

Nicht geräteangepasste Applikationen verwenden eine Scandateneinschleifung in den Tastaturpuffer, um Scandaten direkt an der Cursorposition zu platzieren. Dies wird vom Systemdienst ACD ScanService übernommen.

Applikationsentwickler können die ACD Scanner-API verwenden, welche mehr Kontrolle über den integrierten Scanner ermöglicht. Details befinden sich in der Dokumentation „Android_Handbuch_Programmierung_Programming_Manual“. Da das Scanmodul nicht auf beide Arten gleichzeitig bedient werden kann, ist eine gleichzeitige Nutzung beider Betriebsarten nicht möglich.

2.3.3 Scannerkonfiguration

Die ACD ScanConfig App ist auf den ACD Geräten vorinstalliert und wird dazu verwendet Scannereinstellungen vorzunehmen. Unter anderem können über die App folgende Einstellungen vorgenommen werden:

- Konfiguration von Scannerparameter (erlaubte Barcodes, …)
- Konfiguration der Scannerdaten (anhängen von Prefix, Suffix, …)
- Benachrichtigungston beim Scan eines Barcodes
- Etc.

Die Konfigurationsdaten der ACD ScanConfig werden auf dem Gerät unter */sdcard/ACD/scanner* abgelegt.

Die jeweilige Beschreibung der Scannerparameter kann über das Datenblatt des Herstellers ermittelt werden.

Scanmodul	Beschreibung
SE 965	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se96x.html
SE 1524	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se1524er.html
SE 4500	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4500.html
SE 4710	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4710.html
SE 4750SR	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4750.html
SE 4750MR	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4750.html
SE 4770	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4720-se4770.html
SE 4850	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4850.html
SE 5500	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se55.html
EX 25	https://www.honeywellaidc.com/products/oem-scan-engines/2d-imagers/ex25

Tabelle 10: Datenblätter Scannerparameter

2.4 Ethernet

ACD Geräte, die über einen Ethernet-Anschluss verfügen, können in ein kabelgebundenes Netzwerk integriert werden. Standardmäßig bezieht das Gerät seine Netzwerkkonfiguration per DHCP.

2.5 Wireless LAN (WLAN)

Sofern das ACD Gerät über eine WLAN Funkkarte verfügt kann es in ein existierendes WLAN eingebunden werden. Die Einstellungen können, wie von Android gewohnt, über die Systemeinstellungen des Betriebssystems vorgenommen werden.

Details zu den unterstützen WLAN Standards sind aus technischen Daten des jeweiligen Geräts zu entnehmen.

ACD bietet die Möglichkeit weitere WLAN Parameter, außerhalb der von Android standardmäßig angebotenen Werte, einzustellen. Die Parameter können in /sdcard/ACD/wifi/wifi_config.conf auf dem Gerät editiert werden. Dort sind die Werte auch ausführlich beschrieben.

Damit die Einstellungen übernommen werden, muss der Wartungscode „77696“ in der ACD SystemApp eingegeben werden.

2.6 BT-kompatibler Nahbereichsfunk

Sofern das Gerät über ein Bluetooth-Modul verfügt, kann es mit anderen Bluetooth-Geräten gekoppelt werden. Die Einstellungen können, wie von Android gewohnt, über die Systemeinstellungen des Betriebssystems vorgenommen werden.

Details zu den unterstützen Bluetooth Versionen sind aus den technischen Daten des jeweiligen Geräts zu entnehmen.

2.7 Mobile Datenverbindung

Sofern das Gerät über ein integriertes LTE-Modem verfügt, kann die mobile Datenverbindung zum Zugriff auf das Internet genutzt werden. Vorbedingung ist die Integration einer SIM- oder Mikro-SIM-Karte im Gerät. Bei jedem Booten erfolgt die Abfrage der Pin der SIM-Karte.

Einstellungen können, wie von Android gewohnt, über die Systemeinstellungen des Betriebssystems vorgenommen werden.
Über die Entwickler-Optionen ist es dabei möglich, die Pin-Abfrage zu unterbinden und die mobile Datenverbindung auf Dauer zu aktivieren.

2.8 NFC/RFID

Sofern das Gerät über ein NFC/RFID Modul verfügt kann es NFC Tags und kompatible RFID Tags auslesen oder beschreiben und verschieden Aktionen ausführen wie z. B. die Kopplung mit Bluetoothgeräten. NFC kann in den Einstellungen unter „Verbundene Geräte“ ein- oder ausgeschaltet werden.

Unterstütze NFC/RFID Tags		
MIFARE DESFire EV1	NTAGI2C	NTAG413 DNA
MIFARE DESFire EV2	NTAGI2C Plus	NTAG424 DNA
MIFARE DESFire Light	MIFARE Classic	NTAG424 DNA Tag Tamper
MIFARE Identity	NTAG203F	ICODE SLI
MIFARE Plus	NTAG210	ICODE SLI-L
MIFARE Plus SE	NTAG213	ICODE SLI-S
MIFARE Plus EV1	NTAG215	ICODE SLIX
MIFARE Ultralight	NTAG216	ICODE SLIX-L
MIFARE Ultralight C	NTAG213F	ICODE SLIX-S
MIFARE Ultralight EV1	NTAG216F	ICODE SLIX2
MIFARE Ultralight Nano	NTAG210u	ICODE DNA
NTAG203	NTAG213 Tag Tamper	

Tabelle 11: Unterstütze NFC/RFID Tags

3 Gerätefunktionen der Staplerterminals

3.1 Ethernet

ACD Geräte, die über einen Ethernet-Anschluss verfügen, können in ein kabelgebundenes Netzwerk integriert werden. Standardmäßig bezieht das Gerät seine Netzwerkkonfiguration per DHCP.

3.2 Wireless LAN (WLAN)

Sofern das ACD Gerät über eine WLAN Funkkarte verfügt, kann es in ein existierendes WLAN eingebunden werden. Die Einstellungen können, wie von Android gewohnt, über die Systemeinstellungen des Betriebssystems vorgenommen werden.

Details zu den unterstützen WLAN Standards sind aus technischen Daten des jeweiligen Geräts zu entnehmen.

3.3 BT-kompatibler Nahbereichsfunk

Sofern das Gerät über ein Bluetooth-Modul verfügt, kann es mit anderen Bluetooth-Geräten gekoppelt werden. Die Einstellungen können, wie von Android gewohnt, über die Systemeinstellungen des Betriebssystems vorgenommen werden.

Details zu den unterstützen Bluetooth Versionen sind aus den technischen Daten des jeweiligen Geräts zu entnehmen.

4 Zugang über ADB

Die Android Debug Bridge (ADB) ermöglicht den Zugriff auf ein Android Gerät über die Shell.
Um diese zu verwenden, muss das Android SDK installiert sein und die Kommandozeile den Pfad kennen, in dem adb.exe liegt.

Bei einem MFT10/MFT12 kann die Verbindung erst nach Ablage eines öffentlichen ADB Schlüssels vorgenommen werden. Dieser Vorgang ist in Kapitel 4.6 Zugriff über ADB beim MFT10/MFT12 erklärt.

ADB bietet einige Möglichkeiten, um auf ein Gerät zuzugreifen. Die wichtigsten davon sind:

- **adb devices**
  Zeigt die am PC per ADB angeschlossenen Geräte an.
- **adb connect <IP>**
  Als Alternative zu USB, kann eine ADB Verbindung zu einem Android Gerät auch über eine Netzwerkverbindung (LAN, WLAN) geöffnet werden.
- **adb shell**
  Öffnet die Shell zu dem Gerät. Der Android User in der Shell ist typischerweise „shell“.
- **adb push <lokaler Pfad> <remote Pfad>**
  Kopiert eine Datei vom lokalen Rechner auf das Gerät.
- **adb pull <lokaler Pfad> [<remote Pfad>]**
  Kopiert eine Datei vom Gerät auf den lokalen Rechner.
- **adb install <lokaler Pfad>**
  Installiert eine Android App/ein Package auf dem Gerät.
- **adb uninstall <package>**
  Deinstalliert eine Android App/ein Package vom Gerät.
- **adb forward <lokaler Port> <remote Port>**
  Leitet einen Port vom lokalen Rechner über USB an das Gerät weiter. Dadurch kann etwa der VNC Viewer angesprochen werden, auch wenn keine Netzwerkverbindung zum Gerät vorliegt. Durch den Befehl „adb forward tcp:5900 tcp:5900“ wird auf dem lokalen Rechner Port 5900 abgehört und über die adb-Verbindung auf Port 5900 auf dem Gerät weitergeleitet. Der VNC Viewer kann nun mit dem PC als Ziel (localhost) geöffnet werden und zeigt durch das Forwarding letztendlich auf das Gerät.
- **adb tcpip <port>**
  Um diesen Befehl ausführen zu können muss das Gerät per USB bereits verbunden sein. Über dieses Remote-Kommando wird der ADB Server auf dem Zielgerät im Netzwerkmodus gestartet.

  Ist kein adb über USB möglich, um ADB über das Netzwerk zu erlauben, kann der ADB Server auf dem Zielgerät alternativ auch über den Wartungscode „555500“ in der ACD SystemApp im TCP Modus neu gestartet werden.

4.1 ADB über Netzwerk permanent erlauben

Standardmäßig ist ADB über Netzwerk auf den Android Geräten deaktiviert, um einen Fernzugriff ohne physischen Zugang zum Gerät zu unterbinden und somit Sicherheitsrisiken zu minimieren.

Allerdings kann diese Einstellungen auf dem Gerät überschrieben werden:

- **setproppersist.adb.tcp.port <port>**
  Diese Einstellung wird erst durch einen Werksreset wieder zurückgenommen.

  Ist kein ADB über USB möglich, kann der ADB Server alternativ auch über den Wartungscode „559955“ in der ACD SystemApp dauerhaft auf TCP/IP umgestellt werden.

4.2 Platform Tools

Die Android Platform Tools beinhalten notwendige Programme um auf das Android Gerät, unter anderem über USB, zugreifen zu können. Diese Tools werden von Google zur Verfügung gestellt:
https://developer.android.com/studio/releases/platform-tools

4.3 Android USB Gerätetreiber für Windows

Sollten die Android USB Gerätetreiber unter Windows nicht automatisch installiert werden, können diese für folgende Geräte über das ACD Kundenportal heruntergeladen und manuell installiert werden.

Gerät	USB Treiber im ACD Kundenportal
M210CS[1]	Bitte kontaktieren Sie unseren Support
M260TE	Android-USB-Treiber-i.MX6.zip
M266SE	Android-USB-Treiber-i.MX6.zip
M270	Android-USB-Treiber-i.MX6.zip
M270SE	Android-USB-Treiber-i.MX8MM.zip
M2Smart®	Android-USB-Treiber-410e.zip
M2Smart®SE	Android-USB-Treiber-i.MX8MM.zip
M2Smart®SE 10 Zoll	Android-USB-Treiber-i.MX8MM.zip
M2Smart® HMI10TE Wired LE	Android-USB-Treiber-QCM5430.zip

Tabelle 12: USB Treiber im ACD Kundenportal

4.4 Android MTP Gerätetreiber für Windows

Sollten die Android MTP Treiber unter Windows nicht automatisch installiert werden, können diese manuell nachinstalliert werden.

Folgende Schritte müssen für die manuelle Installation durchgeführt werden:

- ACD-Gerät mit Computer verbinden
- Geräte-Manager starten und nach dem MTP Gerät suchen
- Rechtsklick auf das mit Ausrufezeichen markierte MTP Gerät –> Treiber aktualisieren
- Auf dem Computer nach Treibersoftware suchen
- Unten anklicken –> Aus einer Liste verfügbarer Treiber auf meinem Computer auswählen
- Standardmäßiges MTP Gerät –> MTP Gerät auswählen und mit Weiter fortfahren (siehe Bild unten)

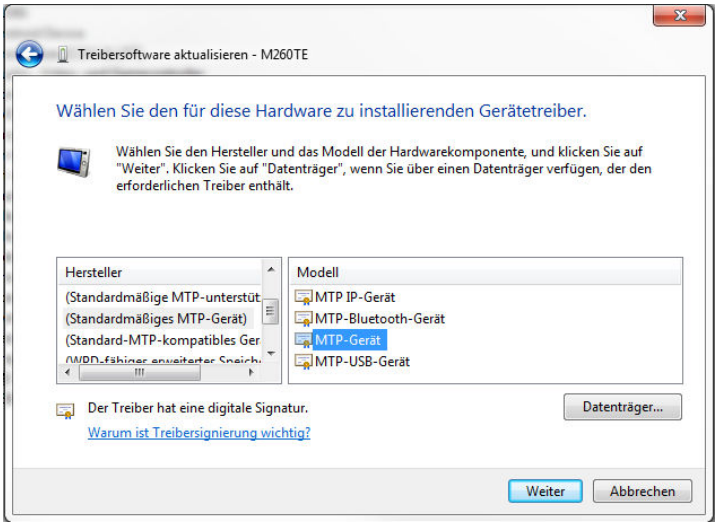

Abbildung 1: Manuelle Gerätetreiber-Installation über Geräte-Manager

4.5 Zugriff über ADB beim M2Smart®SE

Beim M2Smart®SE kann die untere USB-Schnittstelle für mehrere Zwecke (Host oder Slave) verwendet werden.

- Host, um untere Module zu verwenden, welche mittels USB angebunden werden.
- Slave, um eine ADB-Verbindung über USB zu einem PC herstellen zu können.

Mithilfe der vorinstallierten **ACD USB-OTG App** kann der Betriebsmodus umgestellt werden. Für den Zugriff über USB mit ADB ist die Variante DS2Smart® Developerbox der Dockingstationen notwendig.

4.6 Zugriff über ADB beim MFT10/MFT12

Der Zugriff über ADB erfolgt beim MFT10/MFT12 ausschließlich über eine Netzwerkverbindung. Bei der Installation des Android SDK wird ein Schlüsselpaar (öffentlicher und privater Schlüssel) erstellt – diese werden im Normalfall im Benutzerverzeichnis im Unterordner .android abgelegt.

Beispiel öffentlicher Schlüssel unter Windows:
C:\Users\<Benutzername>\.android\adbkey.pub

Beispiel öffentlicher Schlüssel unter Linux:
/home/<Benutzername>/.android/adbkey.pub

Dieser öffentliche Schlüssel muss auf dem MFT10/MFT12 hinterlegt werden. Der dafür vorgesehene Pfad lautet:
/data/misc/adb/adb_keys

Hierbei muss beachtet werden, dass adb_keys der Name der Zieldatei ist. Gegebenenfalls muss adbkey.pub auf dem Zielgerät in adb_keys umbenannt werden.

Anschließend muss die Datei adb_keys auf dem Zielgerät mit folgenden Dateiberechtigungen versehen werden: -rw-rw-rw-.
Dies kann mit dem vorinstallierten TotalCommander durchgeführt werden.

Um Verbindungen von mehreren Quellen zuzulassen, kann die adb_keys auf dem Zielgerät auch mehrere öffentliche Schlüssel beinhalten. Ein weiterer öffentlicher Schlüssel wird an das Ende der Datei in einer neuen Zeile angehängt.

5 Fernwartung mit SCRCPY bei Mobilen Handheld Computern

Mit dem Fernwartungstool Screen Copy kann das Gerät über USB oder Netzwerk aus der Ferne gesteuert werden. SCRCPY steht unter folgendem Link zum Download bereit:
https://github.com/Genymobile/scrcpy#get-the-app

Hinweis:
SCRCPY verwendet intern ADB, um die Fernwartung auszuführen. SCRCPY wird somit inklusive einer lokalen Version der adb.exe ausgeliefert. Dies kann zu Überschneidungen führen, wenn bereits eine ADB-Version installiert ist, die auf dem Host System Systemweit verfügbar ist.
Sollte eine ADB-Version bereits auf dem Host System installiert/konfiguriert sein, bietet es sich an, die **adb.exe** die mit SCRCPY ausgeliefert wird, zu löschen/umzubenennen, um sie somit inaktiv zu schalten.

5.1 Fernwartung mit SCRCPY über Netzwerk

Da SCRCPY mittels einer ADB-Verbindung die Fernwartung zur Verfügung stellt und ADB über das Netzwerk aus Sicherheitsgründen auf dem Gerät zunächst freigeschaltet werden muss, muss vorab ADB über Netzwerk erlaubt werden.

Details dazu finden Sie in Kapitel 4 „Zugang über ADB“ und Kapitel 4.1 „ADB über Netzwerk permanent erlauben“.

Kurzanleitung:
1. Host PC: „adb connect <IP Adresse des Android Geräts>“
2. Host PC: scrcpy.exe starten

6 Betriebssystemversionen

Die Betriebssystemversion von Android kann in den Systemeinstellungen ausgelesen werden.
Desktop -> App Übersicht -> Einstellungen -> Über das Tablet

Hier sind drei relevante Versionsinformationen zu finden:

Android-Version
Enthält die allgemeine Android Betriebssystemversion. Bietet damit Rückschlüsse auf grundlegende Funktionen von Android und dessen API-Level.

Kernel-Version
Die Versionsinformation gibt Rückschlüsse auf den verwendeten Standard-Linux Kernel aufgrund dessen noch weitere Anpassungen durch ACD vorgenommen wurden. Das Datum, das angezeigt wird, gibt das Erstellungsdatum des Kernels durch ACD an.

Build-Nummer
Beinhaltet die ACD Produktversion und eine interne eindeutige ID des Softwarestandes.

7 NTP Server bei Mobilen Handheld Computern

Android bietet offiziell keine Möglichkeit einen spezifischen NTP Server zu setzen. Damit ein Gerät mit Android die Zeit synchronisieren kann benötigt es einen Internetzugriff. Standardmäßig gleicht Android seine Uhrzeit mit dem Zeitserver **2.android.pool.ntp.org** ab.
Auf ACD Geräten wurde diese fehlende Funktionalität nachgerüstet und kann unter **Desktop -> App Übersicht -> Einstellungen -> Datum & Uhrzeit -> NTP Server** angepasst werden.

Hinweis: Beim M270, M260TE, M266SE und MFT1x wird diese Einstellung erst nach einem Neustart vollständig übernommen.

8 ACD KioskMode

ACD Geräte für Android bieten zusätzliche Mechanismen, um das Betriebssystem vor ungewollten Eingriffen der Bediener zu schützen. Ein selbst entwickelter Launcher kann vom Kunden konfiguriert werden, sodass der Endbenutzer nur die gewünschten Apps zur Auswahl hat. Der Launcher kann außerdem weitere Verweise in das System sperren, wie etwa das Herunterziehen der Statusleiste.

Weitere Details können dem „ACD_KioskMode_KioskBrowser_Handbuch_Manual_Android" entnommen werden.

9 Nachinstallationen bei Mobilen Handheld Computern

ACD bietet auf dem Gerät, unter dem Verzeichnis /sdcard/ACD/Apps/, Standardapplikationen an, die vom Kunden nachinstalliert werden können.

Dazu muss auf dem Gerät der Datei Explorer geöffnet und zu dem genannten Verzeichnis gewechselt werden. Wird die App ausgewählt, öffnet Android einen Installationsdialog, der mit „Installieren" bestätigt werden muss.

Hinweis: Je nach Einstellung kann es sein, dass Android untersagt die App über diesen Weg zu installieren („Installation gesperrt. Aus Sicherheitsgründen kann die App nicht installiert werden"). Um diese Meldung zu umgehen, muss in den Einstellungen die Installationsmethode über diesen Weg zugelassen werden. Die Einstellung dazu befindet sich unter **Desktop -> App Übersicht -> Einstellungen -> Sicherheit -> Unbekannte Herkunft (aktivieren)**.

10 Over the Air (OTA) und Firmware Update

Android lässt sich mit einem OTA Paket, das von ACD zur Verfügung gestellt wird, aktualisieren. Bei der Aktualisierung gehen keine Daten oder Einstellungen verloren. Programme wie z. B. der ACD Mobile Device Manager können OTA und Firmware Updates ohne Interaktion am Gerät automatisch mithilfe eines Dienstes einspielen. Eine manuelle Installation ist aber ebenfalls möglich:
Das erhaltene Paket (im .zip Format) muss dazu z. B. mit einem USB Stick auf das Gerät kopiert werden. Das Paket darf nicht entpackt oder verändert werden. In der ACD SystemApp im Untermenü „Systemupdate" lässt sich dieses Paket dann auswählen und installieren. Die Installation erfolgt dann mithilfe eines Dienstes im Hintergrund. Nach einem Neustart ist das Update installiert und das Gerät ist einsatzbereit.

10.1 Android Update Paket

Ein Update Paket kann neben dem normalen Android Update weitere Komponenten enthalten, die dann, während dem Update ebenfalls installiert werden. Ausnahme sind die MFT1x Staplerterminals, diese unterstützen nur normale Android Update Pakete. Der Name des Update Paketes gibt an, welche Komponenten enthalten sind:

```
ota_<gerät>_v<version>_abp.zip -> ota_m2smartse_v1.8_abp.zip
```

Im Namen ist angegeben:

- Das Gerät, für das das Paket erstellt wurde
- Die Android Image Version, die mit dem Paket installieren wird
- Die Angabe, welche Komponenten enthalten sind (das Update-Paket kann eine beliebige Kombinationen der folgenden Komponenten enthalten):
  - „a“: Das Android Image ist im Paket enthalten
  - „b“: Der Bootloader ist im Paket enthalten
  - „p“: Die Gerätefirmware ist im Paket enthalten
  - „o“: Die OEM Daten sind im Paket enthalten

10.2 Firmware Update

Die Gerätefirmware und die Modulfirmware eines aufgesteckten Moduls kann bei M2Smart®(SE) Geräten auch unabhängig von Android aktualisiert werden. Dazu wird, ähnlich wie oben beschrieben, das Update Paket (in diesem Fall eine .bin Datei) auf das Gerät kopiert und in der ACD SystemApp im Untermenü „Systemupdate“ ausgewählt und installiert. Bei einem Update der Gerätefirmware wird das Gerät neu gestartet.

11 Abbildungsverzeichnis

Abbildung 1: Manuelle Gerätetreiber-Installation über Geräte-Manager 16

12 Tabellenverzeichnis

Tabelle 1: Abkürzungsverzeichnis 4
Tabelle 2: ACD Geräte mit Android 6
Tabelle 3: Ein-/Ausschalten 7
Tabelle 4: Standardbelegung zweite Tastaturebene 9
Tabelle 5: Tastaturkonfigurationsdatei 9
Tabelle 6: Tastaturkonfigurationsdatei Tastenbezeichner 10
Tabelle 7: Tastaturkonfigurationsdatei Funktionen 11
Tabelle 8: Standardbelegung Scannertasten 11
Tabelle 9: Scanner- und Imagertypen 11
Tabelle 10: Datenblätter Scannerparameter 12
Tabelle 11: Unterstütze NFC/RFID Tags 13
Tabelle 12: USB Treiber im ACD Kundenportal 15

13 Overview

13.1 Goal and scope of this manual

This document provides operating instructions for the administration of the basic functionality of ACD devices with Android. Topics will be treated that provide a basic understanding of how Android works, as well as practical instructions for repeating administrative tasks.

The goal is to give the user the knowledge to prepare ACD devices with Android for the use of a customer-specific application. Application-specific settings are not the topic of this document; for this, reference will be made to the respective separate documentation.

Descriptions that refer to programs or operating system settings dialogs refer to English desktop language settings.

13.2 List of abbreviations

Abbreviation	Description
OS	Operating System
ADB	Android Debug Bridge
SCRCPY	Screen Copy
MTP	Media Transfer Protocol

Table 1: List of abbreviations

13.3 Android operating system

In addition to other operating systems, the Android operating system (modified by ACD) is used for ACD devices.

Android includes a Linux kernel, however it is not a classic Linux distribution like the ones for desktop and server. Sometimes, basic properties, as might be expected of a system in the UNIX family, are changed a lot. Nevertheless, many properties are present that are comparable to other numerous embedded Linux distributions.

13.3.1 ACD SystemApp

The ACD SystemApp is already pre-installed on the Mobile Handheld Computer. You can use it to display and configure various device information with Android. Updates can be installed in the system Update menu. For the installed ACD Apps, the license keys can be activated in the Licenses menu.

13.3.2 ACD TerminalSettings

The ACD TerminalSettings app, for the Vehicle Mount Terminals MFT10/MFT12 can be used to read various items of information and make settings via the device.

13.3.2.1 System

The System area shows information about the current voltage and brightness. The Antenna switching setting has no function at present.

Calibration

The touchscreen is already pre-calibrated when delivered. If problems occur when making entries, however, the touchscreen can be reset with the Calibration button.
The process is as follows:

- Tap the Calibration button
- Wait a few seconds until the process begins
- Tap the four targets with a stylus one after the other
- After the fourth target, wait a few seconds until the configuration has been applied. As soon as the screen flashes briefly, the configuration is saved. Please do not make any entries before then!

If no input is received for 15 seconds during calibration, the process is automatically canceled. If an incorrect entry is made, for example if a target is not tapped accurately, the process can be deliberately terminated without saving by waiting.

13.3.2.2 Information

The Information area shows the hardware version, current power consumption, firmware version and serial number of the device.

13.3.2.3 Battery

The Battery area shows the battery charge and the runtime of the battery. This display represents an approximate status and refers to a battery with maximum charge capacity.

13.3.2.4 Key Functions

Actions can be assigned to the five hardware keys F1, -, OK, + and F2 in the Key Functions area.

The keys or combinations of keys can also be configured to turn on or shut down the terminal and to define the behavior of the terminal when ignition occurs (turn on, turn off).

13.3.2.5 About

The About area shows the version number of the ACD TerminalSettings app.
Device types - functions and differences

13.4 Device types – functions and differences

13.4.1 List of the ACD devices with Android

Device	Product version ... and higher	Operating system	Processor	Clock speed	RAM (Random Access Memory)	Flash memory less OS
M2Smart® HMI10TE Wired LE	3.2.0	Industrial+ für Android	Qualcomm®	2x Core 2.1 GHz and 4x Core 1.8 GHz	4 GB	64 GB
M210CS[2]	2.6.1	Industrial+ for Android	NXP i.MX8M Mini	1,8 GHz Quad Core	4 GB	32 GB
M260TE	1.8	Industrial+ for Android	NXP i.MX6	1 GHz Dual Lite	1 GB	8 GB
M266SE	1.7	Industrial+ for Android	NXP i.MX6	1 GHz Dual Lite	1GB	8 GB
M270	1.10	Industrial+ for Android	NXP i.MX6	1 GHz Dual Lite	1 GB	8 GB
M270SE	1.13	Industrial+ for Android	NXP i.MX8M Mini	1.8 GHz Quad Core	2 GB	16 GB
M2Smart®	1.6	Industrial+ for Android	Qualcomm® Snapdragon 410	1.2 GHz Quad Core	1 GB	8 GB
M2Smart®SE	1.17	Industrial+ for Android	NXP i.MX8M Mini	1.8 GHz Quad Core	2 GB	16 GB
M2Smart®SE 10 inch	1.11	Industrial+ for Android	NXP i.MX8M Mini	1.8 GHz Quad Core	4 GB	16 GB
MFT10/ MFT12	1.3.0	Android 6.0.1	Intel® Celeron® N2930	1.83 GHz Quad Core	4 GB	64 GB

Table 2: ACD devices with Android

Details about the variants of the ACD devices with different memory sizes can be found in the technical data of the respective device.

13.4.2 Switching on and off

The ACD devices can be switched on and off via the keyboard. Here, there is a distinction between the following modes:

Power up
The device was previously completely switched off.
The operating system boots up completely again on power up.

Proper shutdown
The operating system is shut down properly and then the device is de-energized.

Idle (power save mode)
The device is put into power save mode. Some essential system components remain active, and a re-awakening of the device occurs quickly since the operating system does not have to be re-booted. However, the device is not de-energized in this case and thus this mode can only be maintained for a certain time (depending on the rechargeable battery capacity of the device).
The device can be re-awakened from idle with the same key combination listed in the table below.

[2] Customized product

Hard reset
With a hard reset, the device is de-energized. The operating system tries to shut down properly; if this is not possible, then the device is de-energized anyhow after a defined time.

Device	Switching on and off	Proper shutdown	Idle (power save mode)	Hard reset
M210CS[2]	Hold down **C** and **Enter** for two seconds.	Hold down **C** and **Enter** until Shutdown appears on screen. Shutdown via the Shutdown menu.	Hold down **C** and **Enter** for 300 ms and release.	Hold down **C** and **Enter** for ten seconds.
M260TE	Hold down **C** and **Enter** for two seconds.	Hold down **C** and **Enter** until Shutdown appears on screen. Shutdown via the Shutdown menu.	Currently not supported.	Hold down **C** and **Enter** for eight seconds.
M266SE	Hold down **C** and **Enter** for two seconds.	Hold down **C** and **Enter** until Shutdown appears on screen. Shutdown via the Shutdown menu.	Currently not supported.	Hold down **C** and **Enter** for eight seconds.
M270	Hold down **C** and **Enter** for two seconds.	Hold down **C** and **Enter** until Shutdown appears on screen. Shutdown via the Shutdown menu.	Currently not supported.	Hold down **C** and **Enter** for eight seconds.
M270SE	Hold down **C** and **Enter** for two seconds.	Hold down **C** and **Enter** until Shutdown appears on screen. Shutdown via the Shutdown menu.	Hold down **C** and **Enter** for 300 ms and release.	Hold down **C** and **Enter** for ten seconds.
M2Smart®	Hold down **On/Off** for two seconds.	Hold down On/Off until Shutdown appears on screen. Shutdown via the Shutdown menu.	Hold down **On/Off** for 300 ms and release.	Hold down **On/Off** for seven seconds.
M2Smart®SE	Hold down **On/Off** for two seconds.	Hold down On/Off until Shutdown appears on screen. Shutdown via the Shutdown menu.	Hold down **On/Off** for 300 ms and release.	Hold down **On/Off** for ten seconds.
M2Smart®SE 10 inch	Hold down **On/Off** for two seconds.	Hold down On/Off until Shutdown appears on screen. Shutdown via the Shutdown menu.	Hold down **On/Off** for 300 ms and release.	Hold down **On/Off** for ten seconds.
M2Smart® HMI10TE Wired LE	Hold down **On/Off** for three seconds.	Hold down On/Off until Shutdown appears on screen. Shutdown via the Shutdown menu.	Hold down **On/Off** for 300 ms and release.	Hold down **On/Off** for twenty seconds.
MFT10/ MFT12	Hold down **C** and **Enter** for two seconds.	Hold down **F1** until Shutdown appears on screen. Shutdown via the Shutdown menu.	Currently not supported.	Hold down **Reset-Button** for at least four seconds.

Table 3: Switching on/off

13.5 Start process

The start process for Android consists of three essential phases.

Bootloader
The bootloader is the initial program that is started directly after the device is switched on. In the process, the first hardware initializations are done. In addition, the bootloader is responsible for transferring the kernel parameters and starting the Linux kernel.

Kernel
The kernel is the interface between hardware and Android. It checks which hardware is available and seeks the appropriate device drivers.

Android
As soon as the animated ACD logo is visible during the start process, the kernel is completely loaded and Android (User-Space) is loaded. In the process, Android-specific services are started. After the complete start process, the Android home screen is visible.

14 Device functions of the Mobile Handheld Computer

14.1 Rechargeable battery

The power supply of the ACD devices is ensured by a rechargeable battery. Applications can receive all necessary information about the rechargeable battery from the operating system.

14.1.1 Rechargeable battery change

Back-up functions for the clock and a power bridging during rechargeable battery change are handled by large capacitor capabilities that are loaded by the rechargeable battery within a few minutes.
For this reason, it is not possible to change the rechargeable battery as many times as desired, one time after another during ongoing operation.

The clock continues counting time for several days even without rechargeable battery. The rechargeable battery change, with loaded supporting capacitors, must be done within approx. five seconds.
In this time, the display is deactivated in order to save power. All other device functions, in particular processor, memory, and WiFi, remain functional, which underscores the principle of the rechargeable battery change during ongoing operation.

14.1.2 Optimized charging

Note: This option is available exclusively on the **M2Smart®SE** device.

In the ACD SystemApp, under the “Battery Optimization” tab, you will find the charging optimization switch, which can be used to activate an optimized charging process.

This automatically sets the charge limits to an upper limit of 80 % and a lower limit of 60 %.
The built-in battery is thus charged to the upper limit during the charging process and is not recharged beyond that point. A new charging cycle begins only when the battery’s charge level falls below 60 % or reaches 59 %.

This process reduces the strain on the battery caused by consistently high charge levels, thereby helping to extend the battery’s service life.

14.2 Keyboard

ACD devices are activated or switched on/off with a combination of several keys (see Chapter 13.4.2).

ACD devices have keyboards optimized for their respective specific purpose.
Some are structured in two levels. There is a second keyboard level which can be activated with a function key. This must be enabled together with the desired key".

Device	Desired key for second keyboard level
M210CS[2]	C Key
M260TE	C Key
M266SE	C Key
M270	C Key
M270SE	C Key

Table 4: Default occupancy for second keyboard level

The first level (without switching) as well as the second level can be freely defined for function by using a key for the device. However, this does not mean simply saving another character that will be generated after the key is pressed ("looped in"). It means rather that the following options are available:

- Looping in freely selectable ASCII characters instead of the default assignment
- Looping in characters with a control function (ESC, TAB, etc.)
- Triggering device-specific functions (scanner, etc.)

14.2.1 User-defined keyboard layout

User-defined keyboard layouts are currently available for M2Smart®, M2Smart®SE, M260TE, M270 and M270SE devices only.

For the configuration of the keyboard on the M2Smart®SE and the M270SE please refer to the separate document "Tastaturkonfiguration-unter-Android_Keyboard-Configuration-under-Android_de_en".

The configuration of M260TE and M270 devices is described below.
The keyboard definition is saved in file *keyb_config.conf* which resides in the */sdcard/ACD/keyboard/* folder. This file can be edited on the device or remotely (for example with SCRCPY) with the Totalcommander app. It is also possible to replace the file with an adapted file.

A key is redefined according to the following pattern
Key = function

A separate line is used for each assignment. The designator is always written in lowercase and function names must be superscript. ASCII characters are enclosed in single quotes, functions with less than/greater than signs in addition to the single quotes. If there are deviations from these specifications the assignment is not recognized and the default assignment will be used.

Identifier (with example)	Explanation	Effect
k_1 = "a	Faulty definition	Default assignment will be used
k_2 = "<NOP>"	2 key is redefined	No response
k_f4 = "A"	F4 key is redefined	"A" instead of "F4"
k_scr = "<RSCANNER>"	Right scanner button	Activates the scanner
s_3 =	Empty definition	Default assignment will be used
k_s3 = "<BACK>"	S3 key is redefined	Android Backbutton instead of default

Table 5: Keyboard configuration file

The designators for the available keys of the devices are uniform for all devices.
The table below gives a sample overview.

Identifier	Key
k_x	A key x on the first keyboard level (not shifted)
s_x	A shifted key x (second keyboard level)
k_sx	Special key x
s_sx	Shifted special key x
k_1	Key One
s_2	Key Shift-Two
k_f1	Key F1
k_s1	Key arrow down
k_s2	Key S
k_s3	Key arrow up
k_scl	Left Scan key
s_sch	Scanner key handle with shift (scanner button keyboard level)

Table 6: Keyboard configuration file for keyboard designator

A list of available function designators that can be assigned to the keys listed above appears in the table below. They should be formatted as follows for use in **keyb_config.conf**:

Key designator = "<FUNCTION NAME>"
Examples:

- k_f1 = "<F1>"
- k_scl = "<LSCANNER>"
- s_s3 = "<DEGREE>"

Function name	Function	Function name	Function
NOP	Disables the key	**UP**	Go up
LSCANNER **RSCANNER** **HSCANNER** **SSCANNER**	Activates the scanner. The first letter identifies the scanner button (left, right, handle, S-key)	**DOWN**	Go down
		RIGHT	Go right
		LEFT	Go left
		END	Move to the end
VOLUMEDOWN	Volume down	**MOVEHOME**	Move to position 1
VOLUMEUP	Volume up	**PAGEUP**	Page up
MUTE	Mute the device	**PAGEDOWN**	Page down
CAMERA	Trigger camera (only possible in camera app)	**PLUS**	Plus character
POWER	Open poweroff menu	**COMMA**	Comma character
RECENTAPPS	Show recent android apps	**MINUS**	Minus character
HOME	Android home button	**DOT**	Dot character
BACK	Android back button	**BACKSLASH**	Backslash character
FORWARD	Forward (Browser)	**SEMICOLON**	Semicolon character
NEXT	Next title (Media)	**EQUAL**	Equal character
BRIGHTNESS DOWN	Turn display brightness down	**SLASH**	Slash character
BRIGHTNESSUP	Turn display brightness up	**F1**	Function key 1
STOP	Stop (Media)	**F2**	Function key 2
PLAY	Play (Media)	**…**	…
PAUSE	Pause (Media)	**F12**	Function key 12
MENU	Open menu(in app)	**GAUML**	Key Ä
SELECT	Select	**AUML**	Key ä
CLEAR	Clear textfield	**GOUML**	Key Ö
INSERT	Insert key	**OUML**	Key ö
DELETE	Delete key	**GUUML**	Key Ü
BACKSPACE	Backspace	**UUML**	Key ü
CTRL	Control	**ESZETT**	Key ß
ALT	Alt key	**PARAGRAPH**	Key §

Function name	Function	Function name	Function
ENTER	Enter key	**MIKRO**	Key µ
TABBACK	Tab back key	**CHARTABSET**	Character Tab Set
CAPSLOCK	Capslock key	**DEGREE**	Degree ° character
SHIFT	Shift key	**SUP2**	„Superscript 2“ character
ESC	Escape key	**SUP3**	„Superscript 3“ character
TAB	Tabulator key	**EURO**	€ character

Table 7: Keyboard configuration file functions

Some function keys can only be used with certain apps (for example camera).

To reset the configuration to the default, delete keyb_config.conf and restart the device. A factory reset achieves the same purpose.

Only standard ASCII characters are evaluated. Unicode characters (for example the € sign) must be generated by means of the function table.

ASCII table with supported characters:
https://en.wikipedia.org/wiki/ASCII#Printable_characters

14.3 Scanner unit

A laser scanner module is usually integrated in the mobile devices. Due to the different keyboard variants, different keys are configured on the devices as the default assignment for enabling the scanner:

Device	Scanner key
M210CS[2]	Keys on the left and right side of the housing. S Key on the keyboard. (Note keyboard configuration).
M260TE	Keys on the left and right side of the housing. S Key on the keyboard. (Note keyboard configuration).
M266SE	White and orange function key(s) on the keyboard.
M270	Keys on the left and right side of the housing. S Key on the keyboard. (Note keyboard configuration).
M270SE	Keys on the left and right side of the housing. S Key on the keyboard. (Note keyboard configuration).
M2Smart®	Keys on the left and right side of the housing.
M2Smart®SE	Keys on the left and right side of the housing.

Table 8: Default occupancy of the scanner keys

14.3.1 Scanner and Imager types

Depending on the usage, different types of scan modules can be integrated into an ACD device:

Scan module	Description
SE 965	1D Scanner
SE 1524	1D longrange Scanner
SE 4500	2D Imager
SE 4710	2D Imager
SE 4750SR	2D shortrange Imager
SE 4750MR	2D midrange Imager
SE 4770	2D shortrange Imager
SE 4850	2D longrange Imager
SE 5500	2D advanced range Imager
EX 25	2D longrange Imager

Table 9: Scanner and Imager types

For details about the variants of the ACD devices with different scan modules, see the technical data for the device in question.

14.3.2 Operating variants of the scan unit

The scan module integrated into the ACD device can be used in several ways.

Non device-adapted applications use a scan data looping in in the keyboard buffer in order to place scan data directly at the cursor position. This is taken over by the ACD ScanService system service.

Application developers can use the ACD scanner API, which enables more control of the integrated scanner. For details see the " Android_Handbuch_Programmierung_Programming_Manual" documentation. Because the scan module cannot be used in both ways at the same time, a simultaneous use of both operating modes is not possible.

14.3.3 Scanner configuration

The ACD ScanConfig app is pre-installed on the ACD devices and will make scanner settings for this. Among other things, the app can be used to make the following settings:

- Configuration of scanner parameters (permitted bar codes, etc.)
- Configuration of the scanner data (attachment of prefix, suffix, etc.)
- Notification tone when scanning a bar code
- Etc.

The configuration data of the ACD ScanConfig will be stored on the device under */sdcard/ACD/scanner*.

Scan module	Description
SE 965	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se96x.html
SE 1524	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se1524er.html
SE 4500	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4500.html
SE 4710	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4710.html
SE 4750SR	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4750.html
SE 4750MR	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4750.html
SE 4770	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4720-se4770.html
SE 4850	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se4850.html
SE 5500	https://www.zebra.com/us/en/support-downloads/oem/oem-engines/se55.html
EX 25	https://www.honeywellaidc.com/products/oem-scan-engines/2d-imagers/ex25

Table 10: Datasheets scanner parameter

14.4 Ethernet

ACD devices that have an Ethernet connection can be integrated into a cabled network. By default, the device draws its network configuration via DHCP.

14.5 Wireless LAN (WiFi)

Insofar as the ACD device has a WiFi radio card, it can be incorporated into an existing WiFi. The settings can, as is usual with Android, be made via the system settings of the operating system.

For details about the supported WiFi standards, see the technical data for the device in question.

ACD offers the option of adjusting other WiFi parameters beyond the values offered as standard by Android. The parameters can be edited in /sdcard/ACD/wifi/wifi_config.conf on the device. The values are also described there in detail.

To ensure that the settings are applied, maintenance code "77696" must be entered in the ACD System app.

14.6 BT-compatible close-range radio

If the device has a Bluetooth module, it can be paired with other Bluetooth devices. The settings can, as is usual with Android, be made via the system settings of the operating system.

For details about the supported Bluetooth versions, see the technical data for the device in question.

14.7 Mobile data connection

If the device has an integrated LTE modem, the mobile data connection can be used to access the Internet. Precondition is integration of a SIM or micro SIM card into the device.
The user is prompted for the SIM card PIN every time the system boots.

As is usual with Android, settings can be made via the system settings of the operating system.
Then the developer options can be used to suppress the PIN prompt and activate the mobile data connection long-term.

14.8 NFC/RFID

If the device has an NFC/RFID module, it can read or write NFC tags and compatible RFID tags and perform various actions such as pairing with Bluetooth devices. NFC can be turned on or off in the settings under "Connected devices."

Supported NFC/RFID tags		
MIFARE DESFire EV1	NTAGI2C	NTAG413 DNA
MIFARE DESFire EV2	NTAGI2C Plus	NTAG424 DNA
MIFARE DESFire Light	MIFARE Classic	NTAG424 DNA Tag Tamper
MIFARE Identity	NTAG203F	ICODE SLI
MIFARE Plus	NTAG210	ICODE SLI-L
MIFARE Plus SE	NTAG213	ICODE SLI-S
MIFARE Plus EV1	NTAG215	ICODE SLIX
MIFARE Ultralight	NTAG216	ICODE SLIX-L
MIFARE Ultralight C	NTAG213F	ICODE SLIX-S
MIFARE Ultralight EV1	NTAG216F	ICODE SLIX2
MIFARE Ultralight Nano	NTAG210u	ICODE DNA
NTAG203	NTAG213 Tag Tamper	

Table 11: Supported NFC/RFID tags

15 Device functions of the vehicle terminals

15.1 Ethernet

ACD devices that have an Ethernet connection can be integrated into a cabled network. By default, the device draws its network configuration via DHCP.

15.2 Wireless LAN (WiFi)

Insofar as the ACD device has a WiFi radio card, it can be incorporated into an existing WiFi. The settings can, as is usual with Android, be made via the system settings of the operating system.

For details about the supported WiFi standards, see the technical data for the device in question.

15.3 BT-compatible close-range radio

If the device has a Bluetooth module, it can be paired with other Bluetooth devices. The settings can, as is usual with Android, be made via the system settings of the operating system.

For details about the supported Bluetooth versions, see the technical data for the device in question.

16 Access via ADB for the mobile devices

The Android Debug Bridge (ADB) enables access to an Android device via the shell.
In order to use this, the Android SDK must be installed and the command line must know the path where adb.exe is located.

For an MFT10/MFT12, the connection cannot be made until after a public ADB key is saved. This process for MFT10/MFT12 is explained in section 16.6 Access via ADB for MFT10/MFT12. ADB offers some possibilities for accessing a device. The most important of these are:

- **adb devices**
  Indicates the devices connected to the PC via ADB.
- **adb connect <IP>**
  As an alternative to USB, an ADB connection to an Android device can also be opened via a network connection (LAN, WiFi).
- **adb shell**
  Opens the shell to the device. The Android user in the shell is typically “shell”.
- **adb push <local path> <remote path>**
  Copies a file from the local computer to the device.
- **adb pull <local path> [<remote path>]**
  Copies a file from the device to the local computer.
- **adb install <local path>**
  Installs an Android app/a package on the device.
- **adb uninstall <package>**
  Uninstalls an Android app/a package from the device.
- **adb forward <local port> <remote port>**
  Forwards a port from the local computer via USB to the device. Thus, for example, the VMC viewer can be addressed, even if there is no network connection to the device. With the command “adb forward tcp:5900 tcp:5900” port 5900 on the local computer is tapped and forwarded via the adv connection to port 5900 on the device. The VNC viewer can now be opened with the PC as target (localhost) and points through the forwarding to the device.
- **adb tcpip <port>**
  To be able to run this command, the device must already be connected via USB. This remote command is used to start the ADB server on the target device in network mode.

  If adb is not possible via USB to allow ADB via the network, it is also possible to restart the ADB server on the target device with maintenance code "555500" in the ACD SystemApp in TCP mode.

16.1 Permanently allowing ADB via network

By default, ADB is disabled via network on the Android devices to prevent remote access without physical access to the device, thus minimising security risks.

However, these settings can be overridden on the device:

- **setprop persist.adb.tcp.port <port>**
  These settings are only cancelled by a factory reset.

  If no ADB via USB is possible, the ADB server can alternatively be permanently switched to TCP/IP via the maintenance code "559955" in the ACD SystemApp.

16.2 Platform tools

The Android Platform Tools include necessary programs for accessing the Android device, among other things via USB. These tools are made available by Google:
https://developer.android.com/studio/releases/platform-tools

16.3 Android USB device driver for Windows

If the Android USB device driver should not be installed automatically under Windows, it can be downloaded from the ACD customer portal and installed manually.

Device	USB driver on the ACD customer portal
M210CS[2]	Please contact our support
M260TE	Android-USB-Treiber-i.MX6.zip
M266SE	Android-USB-Treiber-i.MX6.zip
M270	Android-USB-Treiber-i.MX6.zip
M270SE	Android-USB-Treiber-i.MX8MM.zip
M2Smart®	Android-USB-Treiber-410e.zip
M2Smart®SE	Android-USB-Treiber-i.MX8MM.zip
M2Smart®SE 10 inch	Android-USB-Treiber-i.MX8MM.zip
M2Smart® HMI10TE Wired LE	Android-USB-Treiber-QCM5430.zip

Table 12: Android USB device driver for Windows

16.4 Android MTP device driver for Windows

If the Android MTP drivers should not be installed automatically under Windows, they can be installed manually after the fact.

The following steps must be executed for the manual installation:

- Connect ACD device to computer
- Start the Device Manager and search for the MTP Device
- Right-click on the MTP device -> Update driver marked with an exclamation point
- Search on the computer for driver software
- Click at the bottom -> Select driver on my computer from a list of available drivers
- Standard MTP device –> Select MTP device and continue with Next (see image below)

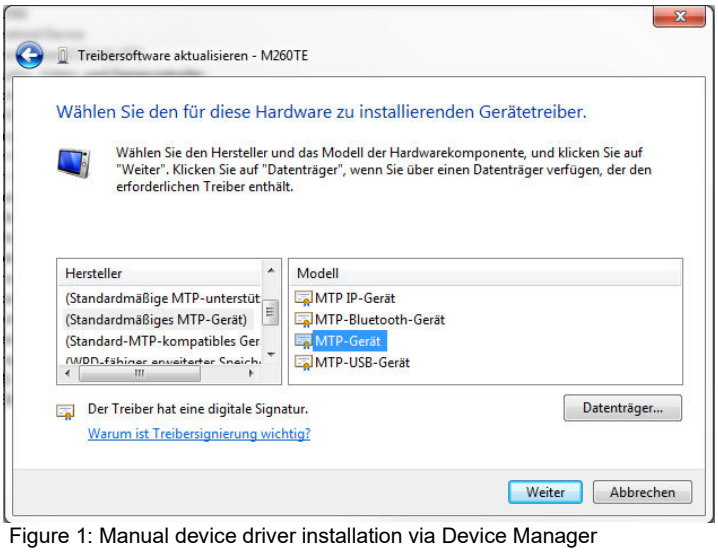

Figure 1: Manual device driver installation via Device Manager

16.5 Access via ADB for M2Smart®SE

With the M2Smart®SE, the lower USB interface can be used for multiple purposes (host or slave).

- Host for using lower modules, which are connected via USB.
- Slave to be able to set up an ADB connection to a PC via USB.

The operating mode can be changed using the pre-installed **ACD USB-OTG app**. For access via USB with ADB, the DS2Smart® Developer box variant of the docking stations is required.

16.6 Access via ADB for MFT10/MFT12

Access via ADB for MFT10/MFT12 is exclusively via a network connection. During the installation of Android SDK, a key pair is created (public and private key). Normally these keys then reside in the user directory in subfolder .android.

Example of a public key under Windows:
C:\Users\<user name>\.android\adbkey.pub

Example of a public key under Linux:
/home/<user name>/.android/adbkey.pub

This public key must be saved in the MFT10/MFT12. The path provided for this is:
/data/misc/adb/adb_keys

Note that adb_keys is the name of the target file. It may be necessary to rename adbkey.pub on the target device to adb_keys.

Then the following file authorizations must be given to file adb_keys on the target device: -rw-rw-rw-.
This can be done with the pre-installed TotalCommander.

To allow connections from multiple sources, adb_keys on the target device can also contain multiple public keys. Another public key is appended to the end of the file in a new line.

17 Remote maintenance with SCRCPY for the Mobile Handheld Computer

With the remote maintenance tool Screen Copy, the device can be controlled remotely via USB or network. SCRCPY is available for download using the following link:
https://github.com/Genymobile/scrcpy#get-the-app

Note:
SCRCPY uses ADB internally to carry out the remote maintenance. SCRCPY is thus delivered including a local version of adb.exe. This can cause overlaps if an ADB version is already installed that is available system-wide on the host system.
If an ADB version should already be installed/configured on the host system, it is a good idea to delete/rename the **adb.exe** that is delivered with SCRCRP in order to render it inactive.

17.1 Remote maintenance with SCRCPY via network

Since SCRCPY provides remote maintenance by means of an ADB connection and ADB via the network must first be enabled on the device for security reasons, ADB via network must be allowed in advance.

Details can be found in chapter 16 “Access via ADB for the mobile devices" and chapter 16.1 “Permanently allowing ADB via network“.

Quick guide:

1. Host PC: "adb connect <IP address of Android device>"
2. Host PC: start scrcpy.exe

18 Operating system versions

The Android operating system version can be read out in the system settings.
Desktop -> App overview -> Settings -> About the tablet

There are three relevant pieces of version information here:

Android version
Contains the general Android operating system version. Thus offers conclusions about the basic Android functions and their API level.

Kernel version
The version information allows you to draw conclusions about the standard Linux kernel used on the basis of which additional adjustments were made by ACD. The date that is displayed indicates the date the kernel was created by ACD.

Build number
Includes the ACD product version and an internal unique ID of the software state.

19 NTP Server for the Mobile Handheld Computer

Officially, Android offers no opportunity to set a specific NTP server. So that a device can synchronize the time with Android, it requires Internet access. By default, Android compares its time with the time server **2.android.pool.ntp.org**.
On ACD devices, this missing functionality was retrofitted and it can be adjusted under **Desktop -> App overview -> Settings -> Date & Time -> NTP Server**.

Note: On the M270, M260TE, M266SE and MFT1x this setting is only fully accepted after a restart.

20 ACD KioskMode

ACD devices for Android offer additional mechanisms for protecting the operating system against undesired interventions by the operator. A self-developed launcher can be configured by the customer so that the end user has only the desired apps from which to select. The launcher can also lock additional references in the system, such as the pulling down of the status bar.

For additional details, see the “ACD_KioskMode_KioskBrowser_Handbuch_Manual_Android”.

21 After-the-fact installations for the Mobile Handheld Computer

On the device, under the directory /sdcard/ACD/Apps, ACD offers standard applications that can be installed by the customer after the fact.

For this, you must open File Explorer on the device and change to the named directory. If the app is selected, Android opens an installation dialog with which “Install” must be confirmed.

Note: Depending on the setting, it can be that Android forbids the app to install this way (“Installation blocked. For security reasons, the app cannot be installed”). In order to work around this message, in the setting, this installation method must be permitted. The setting for this is under **Desktop -> App overview -> Settings -> Security -> Unknown origin (activate)**.

22 Over the Air (OTA) and Firmware Update

Android can be updated with an OTA package provided by ACD. No data or settings are lost during the update. Programs such as the ACD Mobile Device Manager can automatically apply OTA and firmware updates without interaction on the device by using a service. However, a manual installation is also possible:
To do this, the package received (in .zip format) must be copied to the device, for example with a USB stick. The package must not be unpacked or modified. This package can then be selected and installed in the ACD SystemApp in the "System Update" submenu. The installation then takes place with the help of a service in the background. After a restart, the update is installed and the device is ready for use.

22.1 Android Update Package

An update package can contain other components besides the normal Android update, which are then also installed during the update. Exceptions are the MFT1x forklift terminals, which only support normal Android update packages. The name of the update package indicates which components are included:

```
ota_<gerät>_v<version>_abp.zip -> ota_m2smartse_v1.8_abp.zip
```

The name indicates:

- The device for which the package was created
- The Android image version that is installed with the package
- An indication of which components are included (an update package can contain any combination of the following components):
  - "a": The Android image is included in the package
  - "b": The Bootloader is included in the package
  - "p": The device firmware is included in the package
  - "o": OEM data is included in the package

22.2 Firmware Update

The device firmware and the module firmware of an attached module can also be updated independently of Android for M2Smart®(SE) devices. To do this, the update package (in this case a .bin file) is copied to the device, selected in the "System update" submenu of the ACD SystemApp and installed, similar to the procedure described above. If the device firmware is updated, the device will be restarted.

23 List of figures

Figure 1: Manual device driver installation via Device Manager 31

24 List of tables

Table 1: List of abbreviations 20
Table 2: ACD devices with Android 22
Table 3: Switching on/off 23
Table 4: Default occupancy for second keyboard level 25
Table 5: Keyboard configuration file 25
Table 6: Keyboard configuration file for keyboard designator 26
Table 7: Keyboard configuration file functions 27
Table 8: Default occupancy of the scanner keys 27
Table 9: Scanner and Imager types 27
Table 10: Datasheets scanner parameter 28
Table 11: Supported NFC/RFID tags 29
Table 12: Android USB device driver for Windows 31