TWN4

System Overview

DocRev13, May 13, 2016

Elatec GmbH

Contents

1 Introduction

TWN4 is the name of a powerful and versatile series of RFID readers and writers. Here are some of the outstanding features:

- Operation in two frequency bands: 13.56 MHz (HF) and 125 kHz / 134.2 kHz (LF)
- Modular concept consisting of reader/writer modules, carrier boards, antennas and complete devices in housing.
- Security features such as slots for secure access modules or cryptogrphic functions.
- Possibility to write programs which are running on TWN4 itself (Apps).
- Standalone or host-based operation.

There are several products available, which have in common the TWN4 technology.

2 Firmware

Please Note:

Latest version of firmware exceeds size of older models of TWN4.

The TWN4 firmware V3.00 does fit into following models of TWN4:

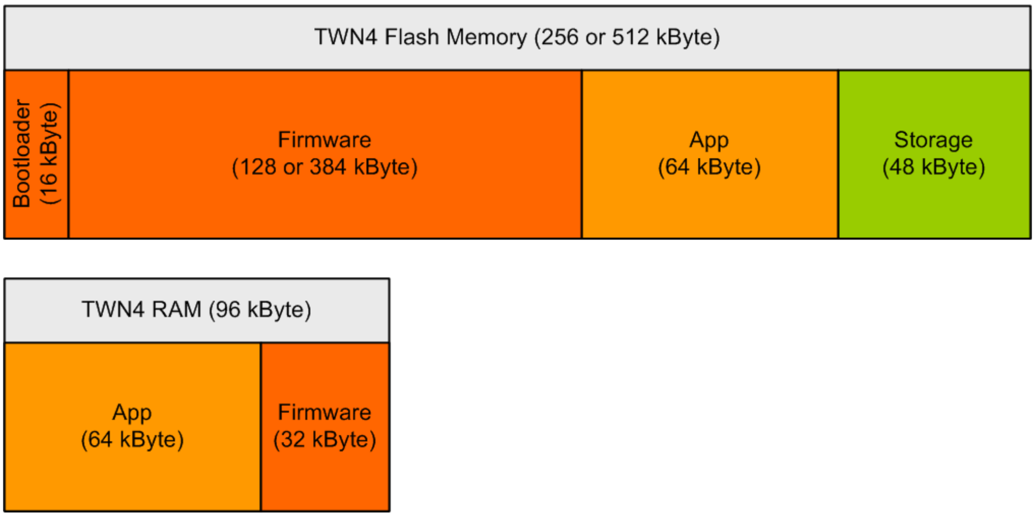
- TWN4 MultiTech Desktop or OEM Board: Revision E or later
- TWN4 MultiTech LEGIC 42 Desktop or OEM Board: Revision C or later
- TWN4 MultiTech LEGIC 45 Desktop or OEM Board: Revision D or later
- TWN4 MultiTech SmartCard or OEM Board: Revision B or later
- TWN4 MultiTech Panel OEM Board: Revision D or later
- TWN4 MultiTech Panel LEGIC 42 OEM Board: Revision C or later
- TWN4 MultiTech Core: Revision D or later
- TWN4 MultiTech Core LEGIC 42: Revision E or later
- TWN4 MultiTech Core LEGIC 45: Revision D or later
- TWN4 MultiTech Mini: All revisions
- TWN4 MultiTech Nano: All revisions
- TWN4 MultiTech Nano LEGIC 42: All revisions

TWN4 Firmware V3.00 or later does not necessarily fit into following models of TWN4:

- TWN4 MultiTech Desktop or OEM Board: Revision D or earlier
- TWN4 MultiTech LEGIC 42 Desktop or OEM Board: Revision B or earlier
- TWN4 MultiTech LEGIC 45 Desktop or OEM Board: Revision C or earlier
- TWN4 MultiTech SmartCard or OEM Board: Revision A or earlier
- TWN4 MultiTech SmartCard LEGIC 42 or OEM Board: Revision B or earlier
- TWN4 MultiTech Panel OEM Board: Revision C or earlier
- TWN4 MultiTech Panel LEGIC 42 OEM Board: Revision B or earlier
- TWN4 MultiTech Core: Revision C or earlier
- TWN4 MultiTech Core LEGIC 42: Revision D or earlier
- TWN4 MultiTech Core LEGIC 45: Revision C or earlier

2.1 Memory View

General spoken, TWN4 has internal 256kBytes or 512 kBytes of flash and 96 kBytes of RAM. The concrete size depends on the model (TWN4 Core, TWN4 Nano, TWN4 Mini). The memory is devided into several sections as shown in the following diagram:



2.1.1 Boot Loader

The boot loader is the entry point for the firmware after powering up TWN4 or after a reset.

Only the boot loader provides functions for programming new firmware or Apps. This means in order to program either a new firmware or another App, the boot loader must be entered.

2.1.2 Firmware

The firmware occupies most space in flash memory. It provides functions for accessing IO or doing RFID operations. Furthermore, the execution of an App is controlled by the firmware.

Both firmware and App cannot be read back from a TWN4.

2.1.3 App

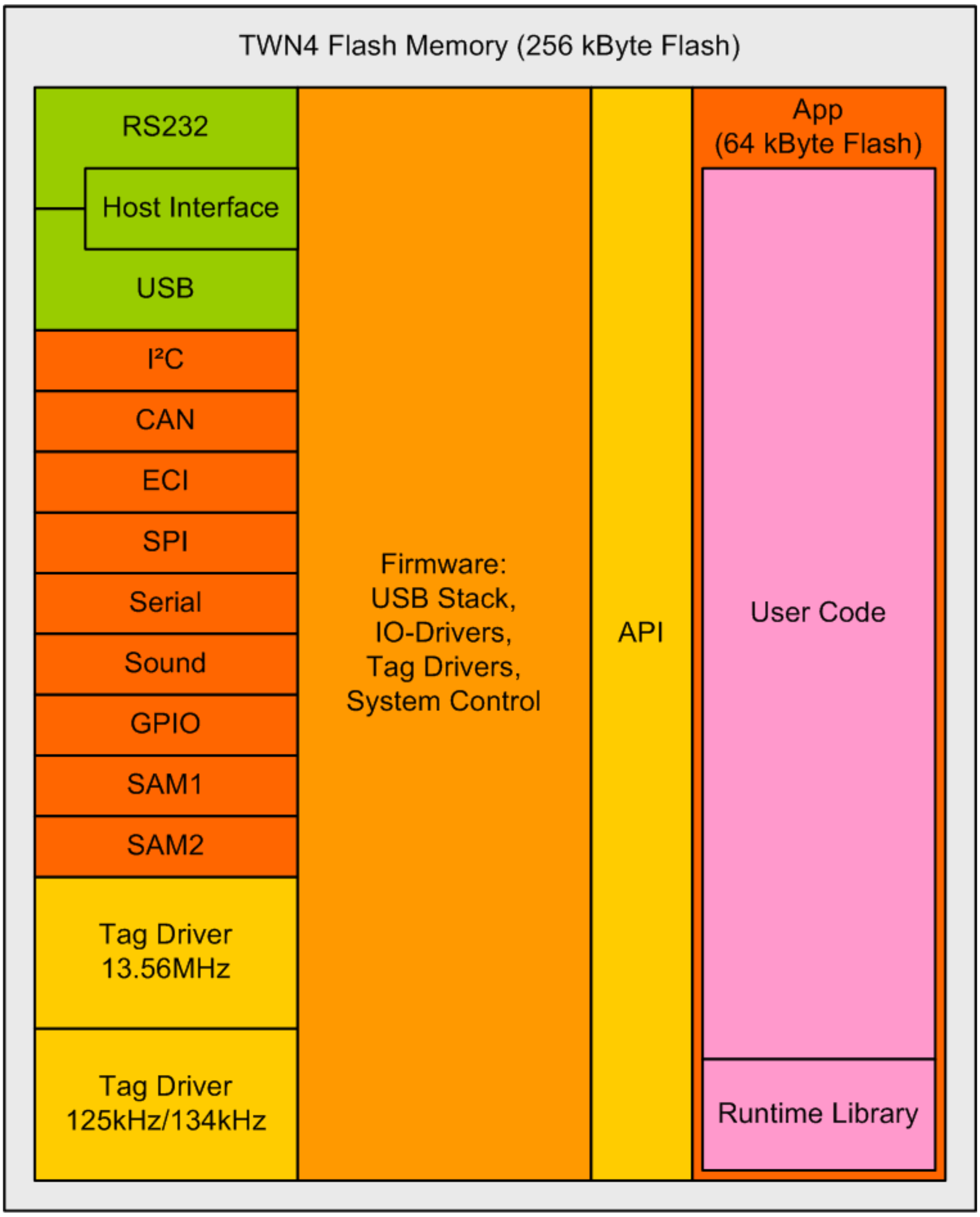
The App is the part of flash memory, which specifies the behaviour of a TWN4. Due to this, the programmer of the App has full control over the behaviour of the final application. An App can be programmed by the customer. In order to do so, an appropriate developer pack is provided.

An App cannot be read back from a TWN4. This allows to store secret keys and other cryptographic functionality as part of an App. Furthermore, the possibility to clone a device is avoided and the intellectual property is protected.

2.1.4 Storage

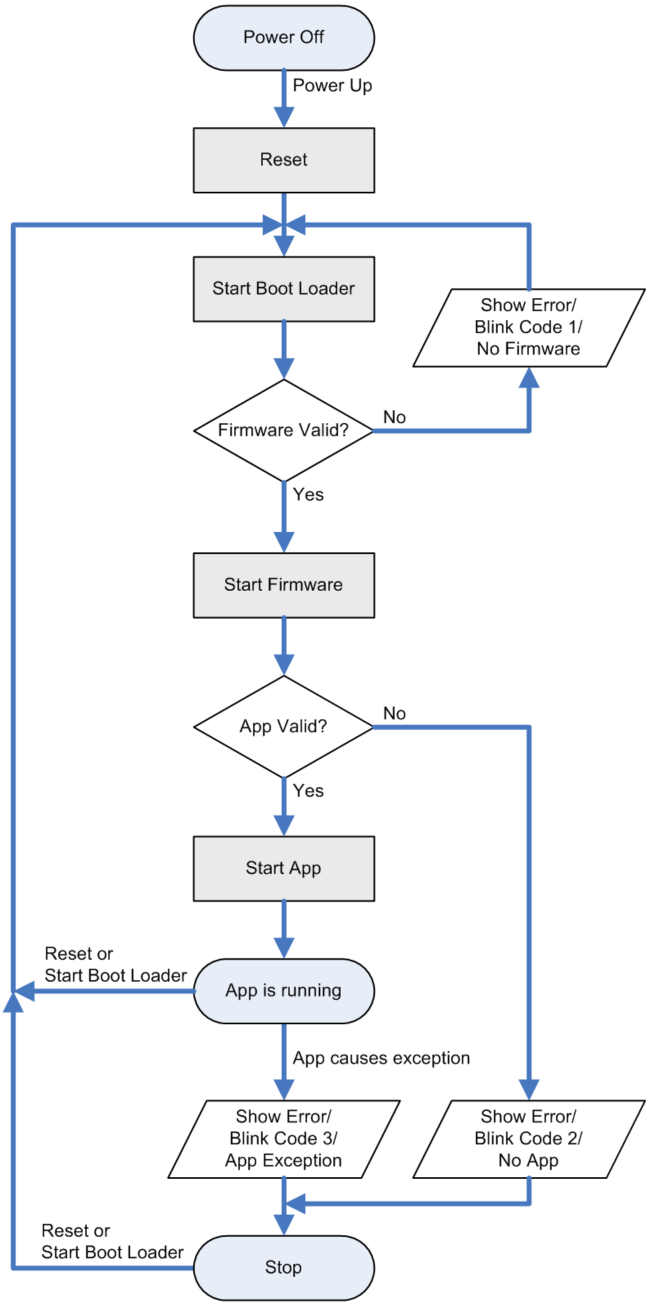
Storage is the section, where data is stored, which can be accessed via the storage functions. In other words, in this aera, the file system is located.

2.2 Functional Units



2.3 Firmware Startup Sequence

The diagram below is showing the sequence of how boot loader, firmware and App are started:



2.4 Firmware Error Conditions

There are several reasons, because the firmware may run into a unwanted condition. If this happens, the condition is shown by a on-board diagnostic red LED of the TWN4 Core Module. The LED is signalling the error code by a number of flashes separated by a pause. This signalling is called blink code. Following blink codes are defined:

- Flash 1 time: There is no valid firmware installed on TWN4. This might be caused, if programming a new firmware onto TWN4 is interrupted by a power failure. In this case, the programming must be started from the beginning.

- Flash 2 times: There is no valid App installed on TWN4. This might be caused, if programming a new App onto TWN4 is interrupted by a power failure. In this case, the programming must be started from the beginning.
- Flash 3 times: The running App caused an exception. A exception is a invalid memory access or invalid program instruction. An App is allowed to access it's own memory space only (64kByte ROM/64kByte RAM).

2.5 Backdoor for Starting the Boot Loader

During development of new Apps and under undefined circumstances, the situation might arise, the starting the boot loader is not possible anymore. In such a situation, it is useful to start the boot loader manually. This can be achieved by connecting two pins of the TWN4 core module together and do a power cycle or reset. The two pins, which have to be connected together are C25 and C28 of the TWN4 Core Module.

2.6 App & Firmware Images

Several firmware images are provided to the customer. App and firmware images can be distginguished by the extension of their filename.

2.6.1 App Images

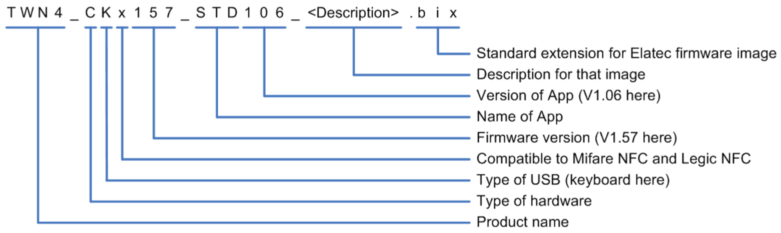
An App image has the extension .t4a. This extension replaces the older version .twn4.app, which leaded to some inconvenience when sent through the Internet. After compiling source code, the result is an App image with that extension. It can be programmed using AppBlaster into TWN4.

2.6.2 Firmware Images

A firmware image has the extension .bix. Normally, such a file contains a firmware for TWN4 and an appropriate App for the intended purpose.

2.6.2.1 Naming Scheme

There is a standard naming scheme for firmware images, which are given to the customer. This is how the name of a firmware image is constructed:


TWN4_CKx157_STD106_<Description>.bix
Standard extension for Elatec firmware image
Description for that image
Version of App (V1.06 here)
Name of App
Firmware version (V1.57 here)
Compatible to Mifare NFC and Legic NFC
Type of USB (keyboard here)
Type of hardware
Product name

Two types of hardware must be distinguished:

- ’C’, All products, which operate a TWN4 Core Module, like TWN4 Desktop, TWN4 OEM PCB, TWN4 Panel
- ’N’, Nano Module
- ’M’, Mini Reader

There are several types of USB stacks available:

- ’K’: USB HID device (keyboard)
- ’C’: USB CDC device class (virtual COM port)
- ’H’: USB HID device (reports)
- ’P’: USB CCID device (reports)

A USB stack is combined with an App. This might be either an App made by the customer or an App provided by the manufacturer. Several Apps are available:

- "STD": Standard App, TWN4 is searching for transponder and forwarding the ID to the host
- "PRS": Simple Protocol, TWN4 is running under the control by a host. TWN4 executes commands, sent by the host and returns response. In this way, nearly all TWN4 system functions can be execute remotely by the host. There is separate documentation and software available for using this mode of operation.

3 TWN4 Developer Pack

The TWN4 developer pack contains all software and documents necessary to operate, program and configure TWN4. Please see separate document for a detailed description of the program AppBlaster, which is used to prepare TWN4 for operation according to your requirements.

3.1 Installation

You received the TWN4 developer pack as zip file. In order to install the package, please follow these steps:

- Create a empty directory on your hard disk
- Unzip the entire content of the zip file into this empty directory
- You're done!

3.2 System Requirements

These are the minimum system requirements for a serious use of the TWN4 Developer Pack:

- Operating system: Microsoft Windows XP or later, 32 or 64 bit
- Microsoft .NET Framework 3.5
- Processor (CPU): 2 GHz
- Hard Disk: 200 MB
- RAM: 2 GB

4 Compatibility

The table below is giving an overview about compatibility of TWN4 to various platforms and their requirements.

System	Mode	Supported	Driver	Remark
Windows XP Windows 7 32/64 Bit Windows 8 32/64 Bit	RS232	Yes	Supported by OS	Port: e.g. COM1:
	USB HID Keyboard	Yes	Supported by OS	
	USB CDC (Virtual COM Port)	Yes	Available	Drivers are part of the developer pack
Linux 32/64 Bit (Ubuntu 10/11/12/13)	RS232	Yes	Supported by OS	Device: e.g. /dev/ttyS0
	USB HID Keyboard	Yes	Supported by OS	
	USB CDC (Virtual COM Port)	Yes	Supported by OS	Device: e.g. /dev/ttyACM0
Linux (ARM Platform)	RS232	No	N/A, platform dependent	
	USB HID Keyboard	Yes	Supported by OS	
	USB CDC (Virtual COM Port)	Yes	Supported by OS	Device e.g. /dev/ttyACM0
Windows CE	RS232	Yes	Supported by OS	Port: e.g. COM1:
	USB HID Keyboard	Yes	Supported by OS	
	USB CDC (Virtual COM Port)	No	N/A, platform dependent	

5 History of Changes

5.1 TWN4 Firmware

5.1.1 Firmware V1.23

- Initially released version.

5.1.2 Firmware V1.40

Firmware:

- LEGIC: System functions `SM4200_xxx` were renamed to `SM4X00` due to support of LEGIC chip SM4500 with identical API. Old style of functions is still available via macros.
- LEGIC: New system functions for reprogramming OS of LEGIC SM4200 or SM4500.
- USB CDC: No modprobe required anymore under Linux.
- USB CDC: TWN4 is mounted as /dev/ttyACMx instead of /dev/ttyUSBx.
- New system functions `Sleep`, `GetDeviceUID`, `SetParameters`.
- Options to keep communication port closed in order to reduce power consumption.
- Option to start an App independent of the current USB enumeration status (standalone applications).
- Support new USB class CCID in various flavours. Please contact your reseller for these features.
- New API ISO7816 for accessing two SAM slots (e.g. on the TWN4 OEM PCB).
- Random number generator, which is available via read to `CHANNEL_RNG`.
- Support for new transponders: AWID, G-Prox, Pyramid and Keri.
- Reworked crypto API with new functionality for cyphered block chaining (CBC).
- Improved (parallel) reading of LF transponders.
- Cotag: Improved reading with the additional option to turn off verify, which further speeds up recognition speed at still reliable reading performance
- Cotag: Decoded reading of ID data instead of a hash value.
- Inditag: Second read mode made available as known from TWN3 system function `IndiTagSearch2`
- Honeytag: Decoded reading of ID data instead of a hash value.
- iCLASS: Support for SIO in a SAM slot.
- iCLASS: Support for reading of PAC from appropriate transponders.
- iCLASS: Support for iCLASS type ISO15693 in addition to ISO14443B.

- MIFARE NFC ISO15693: Improved reading performance.
- New API ISO14443 with new functions for transparent communication to transponder types ISO14443A and ISO14443B.

5.1.3 Firmware V1.47

Firmware:

- Support NFC Peer-to-Peer
- Support ISO FDX-B
- Support EM4x50
- API for AT55xx
- Cotag: Improved reading including start bit
- G-Prox: Modified hash value for improved compatibility
- HITAG 2: Fix bug in Hitag2_SetPassword
- TIRIS: Relaxed read for better support of type multi page.
- App LEGIC Transparent: Support of digital I/O command
- App LEGIC Transparent: Receive-timeout of 1 second for messages
- App Tracer: Support new transponder types + receive of NDEF messages
- App Standard: Support new transponder types + receive of NDEF messages

Runtime environment:

- Optional definition of a manifest, which allows to modify parameters, before the App is started.
- Functions `memcmp`, `memcpy`, `memset` are made available to Apps.
- Definition for `NULL`
- API for using the Simple Protocol from own Apps.
- Function `SetHostChannel` for redirecting output to other than default host port.

5.1.4 Firmware V1.48

Firmware:

- Mifare NFC: Support HF frontend V2
- Mifare NFC: Reset ATS incase of newly found transponder
- Fix issue in EM4150_WritePassword which caused an exception
- Cotag: Improved reading speed
- Cotag: Option for delivering 48 bits instead of 32 bits

Runtime environment:

- More flexible API for Simple Protocol including support of optional transmission of an CRC

5.1.5 Firmware V1.49

Firmware:

- Mifare NFC: Hotfix for support of ISO15693
- Legic NFC: Hotfix for envelope command timeout issue

5.1.6 Firmware V1.57

Firmware:

- Support TWN4 Mini Reader Mifare NFC
- Support TWN4 SmartCard
- ISO7816: New system functions `ISO7816_GetSlotStatus`, `ISO7816_IccPowerOn`, `ISO7816_IccPowerOff`
- ISO14443A: New system functions `ISO14443A_GetATQA`, `ISO14443A_GetSAK`
- ISO14443-4: Fix 14443-4 timeout issue
- ISO7816: Fix T=0 speed/timeout issue
- AT55xx: Fix support of bit rates below f/64

5.1.7 Firmware V1.64

- Hitag 1/S: Increased dynamic range
- Hitag 1/S, Hitag 2: Adjustable timing via parameters
- Hitag S: Read UID of transponders, which are operating in encrypted mode
- DESfire: System function `DESfire_ChangeKey`: Bugfix regarding PICCMasterKey and Application MasterKey
- ISO14443B: Support transparent communication
- ISO14443B: New system functions `ISO14443B_GetATQB`, `ISO14443B_GetAnswerToATTRIB`
- ISO14443: Support high baudrates including parameters for setting up behaviour
- New system function `GetLastError` including error codes
- New API, which provides functions for access of internal flash storage
- MIFARE NFC: Improved reading of ISO15693
- Runtime lib: Systemfunctions `HostTestChar`, `HostWriteByte`, `HostReadByte` are now part of the runtime library
- Runtime lib: Changed naming scheme of all host communication functions by adding a leading "Host"
- Runtime lib: `HostWriteVersion` can handle version strings with up to 50 characters
- ISO7816: Rework and improved API
- LEGIC NFC: Support of system functions `ISO14443A_GetATS`, `ISO14443A_GetATQA`, `ISO14443A_GetSAK`

5.1.8 Firmware V1.68

- New API for MIFARE Plus
- USB keyboard additionaly supports TAB, Backspace, Escape, F1-F12
- LEGIC NFC: Support for MIAFRE Classic API
- New system functions for ISO14443-3
- App Simple Protocol: Support ISO14443B_GetAnswerToATTRIB
- App LEGIC transparent: Optimized timeouts

5.1.9 Firmware V1.71

- New API FELICA
- DESFire: Fixed bug in calculation of MAC in compatible authentication mode

5.1.10 Firmware V1.82

- Improved establishment of a NFC P2P connection
- Support of ICLASS SeOS PACS
- MIFARE Plus: Fixed bug in switching to higher security levels
- MIFARE Classic: Fixed bug - repeated SearchTag after Login obsolete
- Improved timing settings for FeliCa transponder search
- Improved behavior EM4x50
- Support TWN4 Mini Reader MIFARE NFC USB
- Optimized (shortened) search time for many LF transponders
- Function "Sleep": New option to put TWN4 in deeper sleep mode

5.1.11 Firmware V1.85

- Support TWN4 Nano Module.

5.1.12 Firmware V3.00

- Improve performance of Hitag 2.
- Support of contact based memory cards (SLE44xx).
- Support of transponder family NTAG.
- Support of transponder family SRX.
- Support of transponder family MIFARE Ultralight EV1.
- Authentication at MIFARE Ultralight C via SAM AV1/2 supported.

- TWN4 MultiTech Nano: Support of HF transponder types ISO14443A/B, ISO14443B', ISO15693, SRX, ICLASS.
- TWN4 MultiTech Nano: Support of contact based smartcards (ISO7816).
- TWN4 MultiTech Nano: Support of Low Power Card Detection (LPCD).
- Support new transponder type Deister
- EM4102: Determine, if f/64 or f/32 was detected
- EM4102: Setup search for f/64 and/or f/32
- EM4150: Determine, if f/64 or f/40 was detected
- EM4150: Setup search for f/64 and/or f/40
- AT55: Support start marker
- USB delivers firmware version as product string
- USB optionally can be set up to specify unique serial number
- USB keyboard repeat rate can be modified
- USB keyboard support remote wake up of host

5.2 AppBlaster

5.2.1 AppBlaster V1.03

- Initially released version.

5.2.2 AppBlaster V1.40

- New work flow based on projects and templates.
- Save and load of projects.
- Support for all new transponder types of TWN4.
- Support for iCLASS PAC.
- Possibility to generate production images.

5.2.3 AppBlaster V1.47

- Support of new types of transponders including NFC

5.2.4 AppBlaster V1.49

- Support for Cotag 48 bits

5.2.5 AppBlaster V1.57

- Include correct App name and version into firmware image
- Support TWN4 Mini Reader Mifare NFC

5.2.6 AppBlaster V1.64

- Use of C99 language standard

5.2.7 AppBlaster V1.71

- The standard project folder now by default is the folder, where the source code resides or the App-folder in case of configured Apps.
- AppBlaster now automatically fills in information regarding production image, if file name of source code follows the rules (see name of sample Apps).

5.2.8 AppBlaster V1.82

- Temporary files end with additional _temp in order to avoid collision with existing files with identical name.
- Fix minor issue in temporary file, which showed only two digits of firmware version.

5.2.9 AppBlaster V1.85

- Support TWN4 Nano Module.

5.2.10 AppBlaster V2.00

- All new version of AppBlaster:
- Cleaned up interface.
- Support multiple types of transponders with individual configuration of each type.
- Support access to data sections of various transponders.

5.3 Director

5.3.1 Director V1.00

- Initially released version.

5.3.2 Director V1.06

- Support system functions of TWN4 firmware V1.40
- Baud rate of serial communication can be adjusted
- "Simple Test": Selectable output format
- "Simple Test": Copy ID in clip board
- "Simple Test": Beep button is removed
- "Simple Test": Allow director to beep if needed
- "Simple Test": Only transponders, which are supported by TWN4 are selectable
- "Simple Test": Transponders, which are supported by TWN4 are selectable
- "Simple Test": Combo box is set up with currently configured transponders
- "Function Test": Possibility to enter a function call manually

5.3.3 Director V1.10

- Support of new transponder types and system functions according to TWN4 firmware version 1.47.
- Support bytes arrays which contain 0(!) bytes

5.3.4 Director V1.11

- Improved GUI
- Support firmware V1.48

5.3.5 Director V1.12

- Improved communication interface
- Support firmware V1.49

5.3.6 Director V1.14

- Support firmware V1.57
- Support of Simple Protocol, ASCII or binary, with or without CRC
- Support of USB HID reports
- Support of TWN4 SmartCard (slot ID-1, ID-0/SAM, slot 3 and 4)
- Support extended version string (Core Module or Mini Reader)

5.3.7 Director V1.16

- Support firmware V1.64
- Resizeable and scrollable history list
- Defined minimum size of main window
- Improved connecting mechanism and behaviour
- Colourful feedback of result of call of system functions
- Simplification of entering parameters for structure TDESFireFileSettings
- Manual input of system calls
- Many minor improvements

5.3.8 Director V1.17

- Support firmware V1.68

5.3.9 Director V1.18

- Support firmware V1.71

5.3.10 Director V1.22

- Support TWN4 Nano Module
- Minor improvements

5.3.11 Director V1.26

- Minor bug fixes
- Support TWN4 firmware V3.00

5.4 General

5.4.1 TWN4DevPack147

- Stripped down tool chain

5.4.2 TWN4DevPack148

- Link correct libgcc.a

6 Disclaimer

Elatec reserves the right to change any information or data in this document without prior notice. The distribution and the update of this document is not controlled. Elatec declines all responsibility for the use of product with any other specifications but the ones mentioned above. Any additional requirement for a specific custom application has to be validated by the customer himself at his own responsibility. Where application information is given, it is only advisory and does not form part of the specification.

All referenced brands, product names, service names and trademarks mentioned in this document are the property of their respective owners.