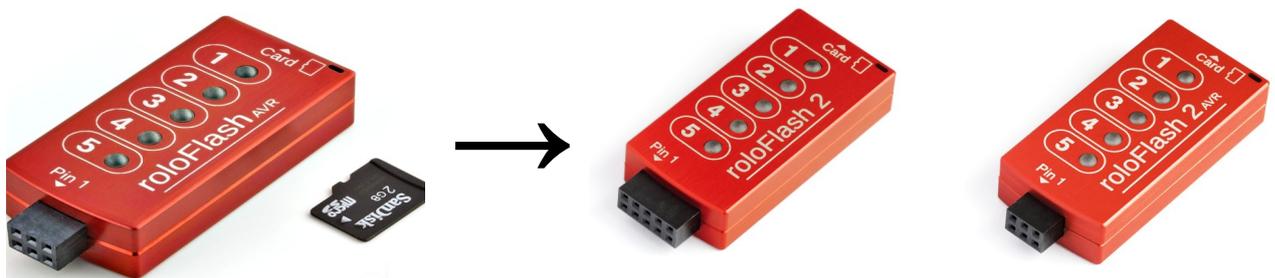


halec
Herrnröther Str. 54
63303 Dreieich
Germany

www.halec.de



Migrations-Leitfaden roloBasic-Skripte für roloFlash AVR und roloFlash-2-Familie



Anleitung zur Migration von roloBasic-Skripten von roloFlash AVR auf die
roloFlash-2-Familie (roloFlash 2 und roloFlash 2 AVR)

Dokumentenversion 1.7.1 vom 2025-04-27
(Stand der Software: 07.AA)

Copyright © 2009-2025 halec. Alle Marken, Logos und Bilder sind Eigentum der jeweiligen Hersteller bzw. Urheber. Änderungen und Irrtümer vorbehalten.

Inhaltsverzeichnis

I	roloFlash: API-Versionen.....	.iii
II	Konzeptionelle Unterschiede.....	.iv
III	Einzelne Funktionen.....	.vi
IV	Verwendete Dateien.....	.viii

I roloFlash: API-Versionen

Folgende roloFlash-API-Versionen sind zur Zeit im Umlauf:

- Für roloFlash AVR:
 - (nur eine Version im Umlauf)
- Für die roloFlash-2-Familie:
 - 02
 - 04
 - 05
 - 06
 - 07

Sie erkennen die Version an der Major-Version der Software der roloFlash-2-Familie.

Beispiel: „07.AB“: API-Version 7 (Major-Version = „07“, Minor-Version = „AB“)

Diese Anleitung zeigt Ihnen, wie Sie Ihre roloBasic-Skripte von roloFlash AVR auf die roloFlash-2-Familie in der API-Version 7 anpassen.

Hinweis:

Auf der zum roloFlash gelieferten microSD-Karte befinden sich im Verzeichnis „scripts“ zahlreiche Beispiel-Skripte für verschiedene Abläufe und verschiedene Mikrocontroller, die evtl. ein besserer Anlaufpunkt zur Konvertierung Ihrer Skripte sein könnten als diesem Migrations-Leitfaden zu folgen.

Hinweis:

Wenn Sie Hilfe benötigen bei der Anpassung Ihrer roloBasic-Skripte

- von roloFlash AVR auf die roloFlash-2-Familie oder
- von einer roloFlash-2-API-Version auf eine andere,

kontaktieren Sie uns bitte per E-Mail an [<rh@halec.de>](mailto:rh@halec.de).

II Konzeptionelle Unterschiede

roloFlash AVR	roloFlash-2-Familie
<p>Ein roloBasic-Skript kann ein Magic-Cookie in der ersten Zeile haben. Dieses muss dann als Kommentar kodiert sein:</p> <pre>!roloFlash AVR</pre>	<p>Ein roloBasic-Skript kann ein Magic-Cookie in der ersten Zeile haben. Dieses muß mit einem „#“ beginnen und lautet zum Beispiel:</p> <pre>#roloFlash 2, v07.AB</pre> <ul style="list-style-type: none"> • Bitte schreiben Sie auch bei roloFlash 2 AVR „#roloFlash 2“, nicht „roloFlash 2 AVR“ • Dabei ist V07 die API-Version, die der Major-Version der roloFlash-Software entspricht • Dabei ist das „AB“ die Minor-Version der roloFlash-Software und dient nur zur Dokumentation. Sie können hier auch ein „*“ als Wildcard angeben: <pre>#roloFlash 2, v07.*</pre> <p>Sie können auch mit einem „+“ anzeigen, dass dieses Skript mit dieser oder neuerer Firmwareversion ausgeführt werden kann. Das ist der empfohlene Standard:</p> <pre>#roloFlash 2, v07+</pre>
<p>Unterstützt nur eine Familie an Targets, die immer über ISP angeschlossen sind.</p> <p>Auswirkung auf das API: - Sofortige Kommunikation mit dem Target möglich (z.B. getSignature)</p>	<p>Unterstützt mehrere Familien an Targets, die über unterschiedliche Busse angeschlossen sind. Bei manchen Bussen können auch mehrere Targets angeschlossen sein (JTAG für STM32).</p> <p>Auswirkung auf das API: - Zuerst muss ein Bus geöffnet werden (z.B. busHandle = bus_open(ISP, 100000), man erhält ein Bus-Handle. - Dann muss das Target unter Angabe des Bus-Handles geöffnet werden, z.B. mit targetHandle = target_open(busHandle, 0) - Erst jetzt kann mit dem Target unter Angabe des Target-Handles kommuniziert werden, z.B.</p>

	target_getDeviceId(targetHandle)
getSignature bestimmt implizit auch Parameter wie Flashsize und FlashPagesize	<p>target_getDeviceId hat keine Seiteneffekte.</p> <p>Die nötigen Parameter können aus einer internen Datenbank ausgelesen werden und müssen für das Flashen explizit gesetzt werden:</p> <pre>dbHandle = db_getHandle(<targetName>) Beispiel: dbHandle = db_getHandle(Atmega128) flashSize = db_get(dbHandle, DB_FLASHSIZE) pageSize = db_get(dbHandle, DB_FLASHPAGESIZE) target_setMemoryMap targetHandle, FLASH, MEM_SIZE, flashSize target_setMemoryMap targetHandle, FLASH, MEM_PAGESIZE, pageSize</pre>

III Einzelne Funktionen

roloFlash AVR	roloFlash-2-Familie
targetPresent	target_getPresent(<targetHandle>)
programTarget	target_setMode targetHandle, PROGRAMMODE
runTarget	target_setMode targetHandle, RUNMODE
restartTarget	target_restart <targetHandle>
SetProgrammingSpeed <speed>	bus_open(ISP, <index>, <speed>) oder bus_setSpeed <busHandle>, <speed>
getTargetVoltage	getTargetBoardVoltage
readBits(<index>)	target_readBits(<targetHandle>, <index>)
writeBits index, values	target_writeBits <targetHandle>, <index>, <values>
getSignature	target_getDeviceID(<targetHandle>)
getFlashLayout	target_getMemoryMap(<targetHandle>, FLASH, MEM_SIZE) und target_getMemoryMap(<targetHandle>, FLASH, MEM_PAGESIZE)
setFlashLayout(<size>, <pagesize>)	target_setMemoryMap <targetHandle>, FLASH, MEM_SIZE, <size>) und target_setMemoryMap <targetHandle>, FLASH, MEM_PAGESIZE, <pagesize>)
getEepromLayout	target_getMemoryMap(<targetHandle>, EEPROM, MEM_SIZE) und target_getMemoryMap(<targetHandle>, EEPROM, MEM_PAGESIZE)
setEepromLayout(<size>, <pagesize>)	target_setMemoryMap <targetHandle>, EEPROM, MEM_SIZE, <size>) und target_setMemoryMap <targetHandle>,

	EEPROM, MEM_PAGESIZE, <pagesize>)
setExtendedAddressMode <value>	target_setExtendedAddressMode <targetHandle>, <value>
clearMemoryLayout	target_clearMemoryLayout <targetHandle>
eraseFlash	target_erase <targetHandle>
writeFileToFlash 0, <filename>	target_writeFromFile <targetHandle>, 0, <filename>, HEX, FLASH, WRITEONLY
writeVerifyFileToFlash 0, <filename>	target_writeFromFile <targetHandle>, 0, <filename>, HEX, FLASH, WRITEVERIFY
verifyFileToFlash 0, <filename>	target_writeFromFile <targetHandle>, 0, <filename>, HEX, FLASH, VERIFYONLY
writeFileToEeprom 0, <filename>	target_writeFromFile <targetHandle>, 0, <filename>, HEX, EEPROM, WRITEONLY
writeVerifyFileToEeprom 0, <filename>	target_writeFromFile <targetHandle>, 0, <filename>, HEX, EEPROM, WRITEVERIFY
verifyFileToEeprom 0, <filename>	target_writeFromFile <targetHandle>, 0, <filename>, HEX, EEPROM, VERIFYONLY
Alle Dateisystem-Funktionen, z. B.:	
fsOpen	fs_open
...	...
Alle LED-Funktionen, z. B.:	
ledOn	led_on
...	...

IV Verwendete Dateien

Ab der Major-Version 05 wurden die verwendeten Dateien umbenannt:

roloFlash-2-Familie mit Major-Version höchstens 04 oder roloFlash AVR	roloFlash-2-Familie mit Major-Version mindestens 05 hier für V07:
<code>run.bas</code>	<code>run_v07.bas</code>
<code>RUN.BIN</code>	<code>RUN_V07.BIN</code>
<code>rbc.exe</code>	<code>rbc_v07.exe</code>
<code>compile.bat</code>	<code>compile_v07.bat</code>

Damit ergeben sich folgende Möglichkeiten:

- Wenn Sie einen Mischbetrieb von roloFlash AVR und roloFlash-2 haben, dann können Sie SD-Karten so vorbereiten, daß beide Versionen vorhanden sind. Eine solche SD-Karte kann dann sowohl in roloFlash AVR wie auch in roloFlash-2 verwendet werden. Auf der SD-Karte sind dann die Dateien RUN.BIN für roloFlash und RUN_V07.BIN für roloFlash 2 mit Major-Version 07 vorhanden.
- Bei zukünftigen Versionen (z.B. Major-Version 07) für roloFlash 2 können Sie die Skripte für mehrere Major-Versionen (z.B. RUN_V07.BIN und RUN_V07.BIN) auf die SD-Karte kopieren. Damit ist ein Mischbetrieb mit verschiedenen roloFlash 2 Firmwareversion möglich.
- Hinweis: Änderungen, die nur die Minor-Version betreffen (z.B. von V07.AA nach V07.AB), haben keine Einfluß auf die Skripte.
- Hinweis: Sie können jederzeit einen roloFlash 2 auch wieder auf eine ältere Firmware zurückbringen. Daher können Sie eine neue Version testen und anschließend entscheiden, ob Sie bei der Version bleiben oder auf eine andere Version wechseln.