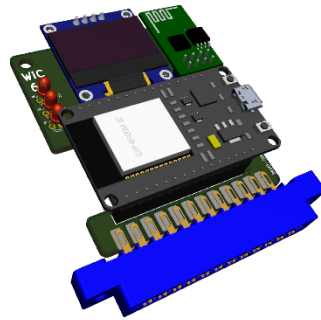


WiC-64



Wireless Interface für den Commodore 64 - VC20 - C128 / D - SX-64

Manual written by KiWi 01/2022

V1.1 – 10.01.2022

Über das WiC64:

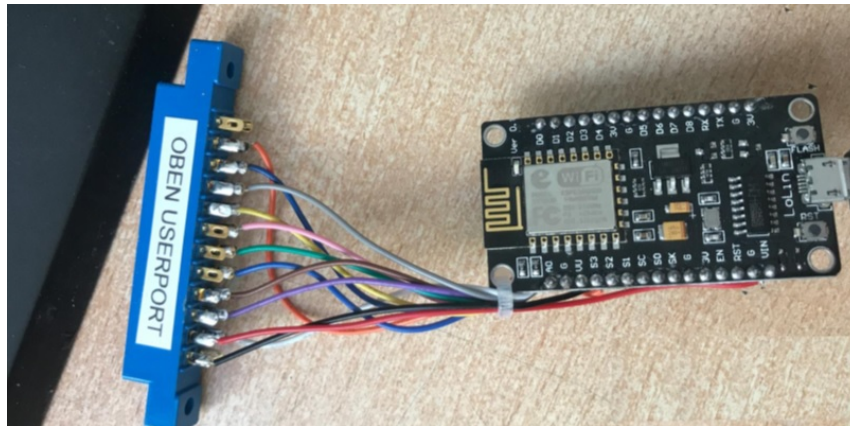
Das Wireless Interface für den Commodore 64 – aka WiC64 – Ist ein paralleles WLAN-Interface für Commodore Rechner mit Userport-Anschluss. Im Gegensatz zu den bisher bekannten „Modems“ arbeitet das WiC64 im parallelen Modus mit 8 Datenleitungen, 2 Handshakeleitungen und einer Kontrollleitung. Dadurch wird die Datenübertragung um den Faktor 20 beschleunigt: Das WiC64 kann 202 Blocks / 50kb in 2.5 Sekunden aus dem Internet laden – und zwar per HTTP:// oder TCP Verbindung. Es ermöglicht dem Rechner den vollständigen Zugang zum Internet und die Anwendungsmöglichkeiten sind nahezu unbegrenzt. Von einfachen Anwendungen wie dem Synchronisieren der Uhrzeit mit einen NTP Zeitserver über das Internet, dem Laden von Daten und Programmen von Webservern oder aus PHP-Skripten bis zu komplexem Chat und Messagesystemen ist alles möglich. Die komplette Hardware und Software ist Open Source und kann beliebig erweitert werden. Alle dazu nötigen Informationen und Downloads findet man unter

<http://www.wic64.de>

Bezugsquellen für die Platine / Einzelteile / Komplettsysteme:

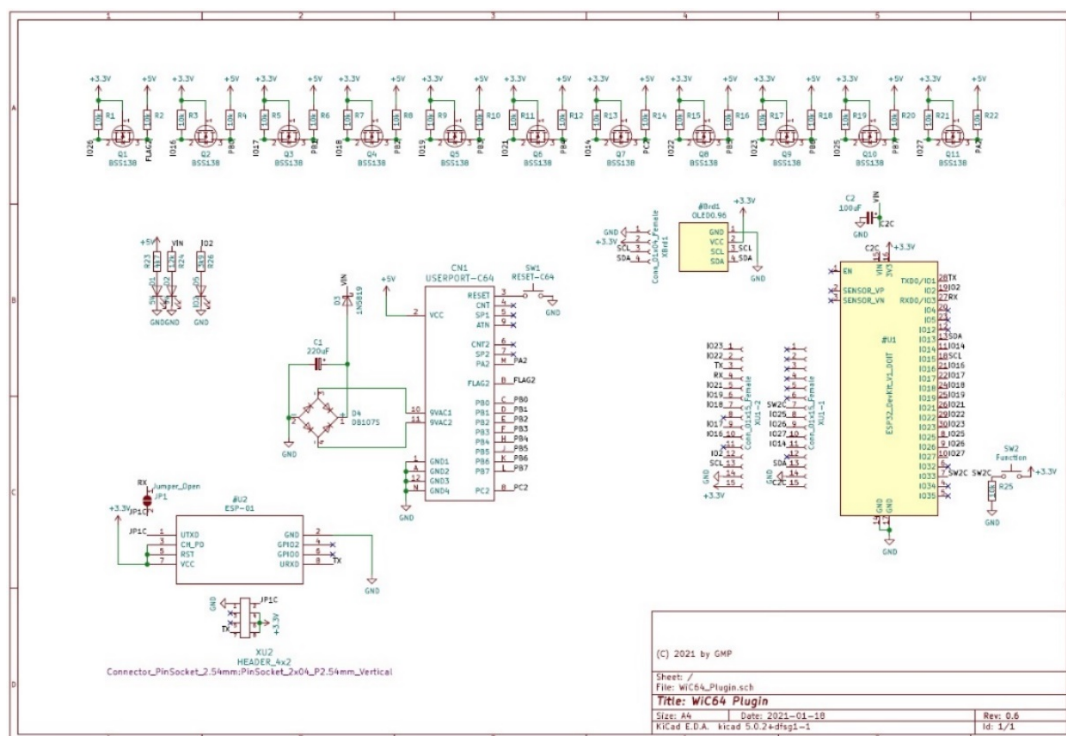
<https://restore-store.de/>

Entstehungsgeschichte:



Das Ur-WiC64 bestand aus einem ESP32 Developmentboard, einem Userportstecker und 13 Kabeln.

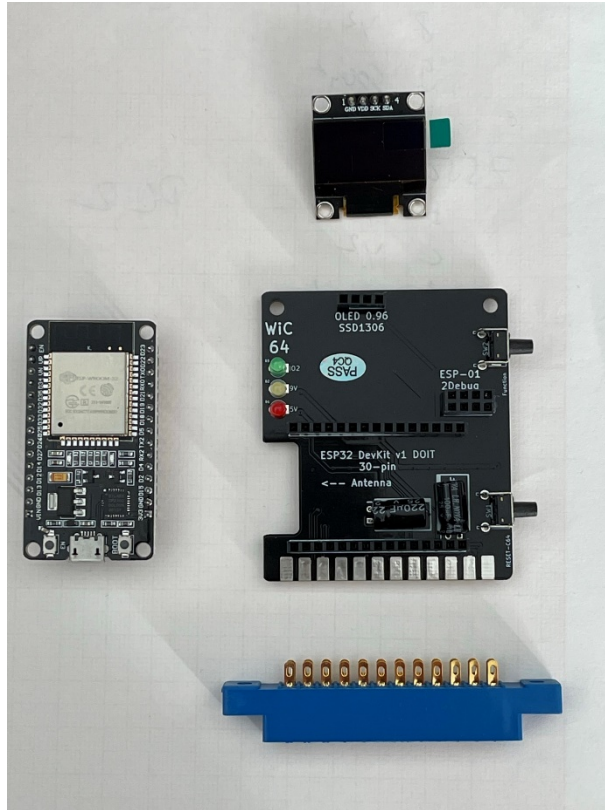
Die ESP32 Boards sind laut Hersteller +5V tolerant und können „Theoretisch“ ohne eine Angleichung an die 3.3V Levels des ESP32 direkt an den C64 Userport angeschlossen werden. Man braucht also nicht unbedingt eine Platine und ein Display, um einen WiC64 nachzubauen, es wird jedoch empfohlen. Wenn man die Levelshifter weglässt und den ESP32 direkt mit den +5V vom C64 versorgt ist ein problemloser Betrieb möglich. Laut den Commodore-Spezifikationen sollen am Userport aber nicht mehr als 100mA abgenommen werden und der ESP32 braucht beim Starten des WLANs kurzzeitig mehr als 200mA – Dies kann ein Risiko für den C64 und sein Netzteil darstellen – Daher übernehmen wir für diese Variante keine Gewähr. Bei der Entwicklung ist aber in Monaten des Betriebs absolut nichts kaputt gegangen. Für Leute die den eigenständigen Nachbau wagen wollen hier der Schaltplan:



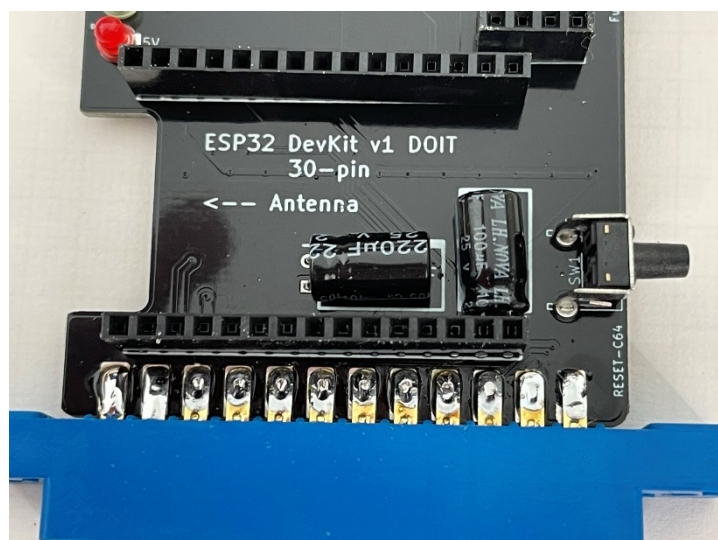
Download : <http://www.wic64.de>

Aufbauanleitung:

Der Zusammenbau ist kinderleicht. Es muss lediglich der Userport-Stecker an die Platine mit 24 Lötstellen angelötet werden. Die Platine ist schon mit allen Bauteilen fertig bestückt und verlötet – Der ESP32 und das Display müssen nur auf die Platine aufgesteckt werden.



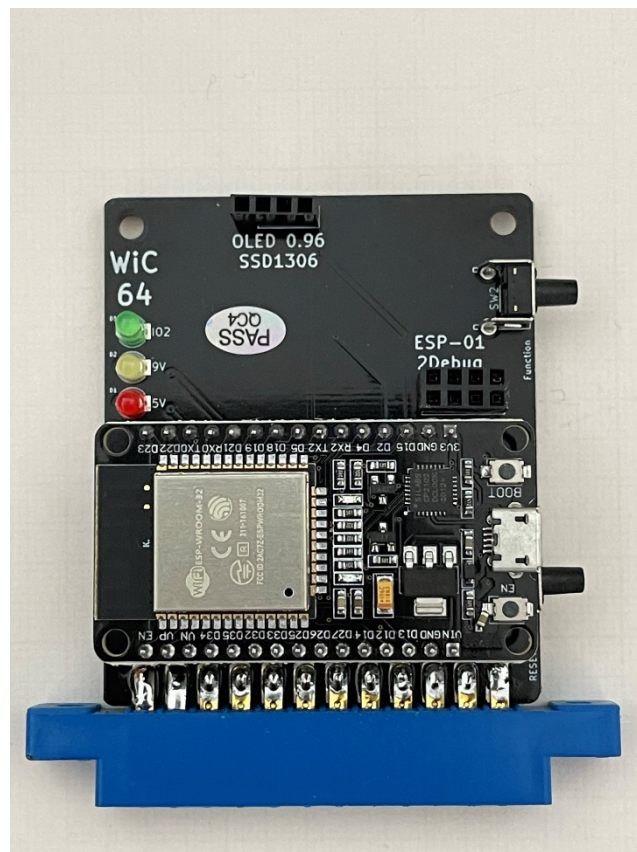
Den Userport-Stecker einfach auf die Platine stecken – Je nach Stecker muss man die Lötflächen etwas zusammen biegen damit die Kontakte enger auf der Platine anliegen. Das ist aber auch schon alles: Jetzt an 12 Stellen den Userport-Stecker an die Platine anlöten:



Dann den Stecker auch auf der Rückseite an den 12 Stellen mit der Platine verlöten:



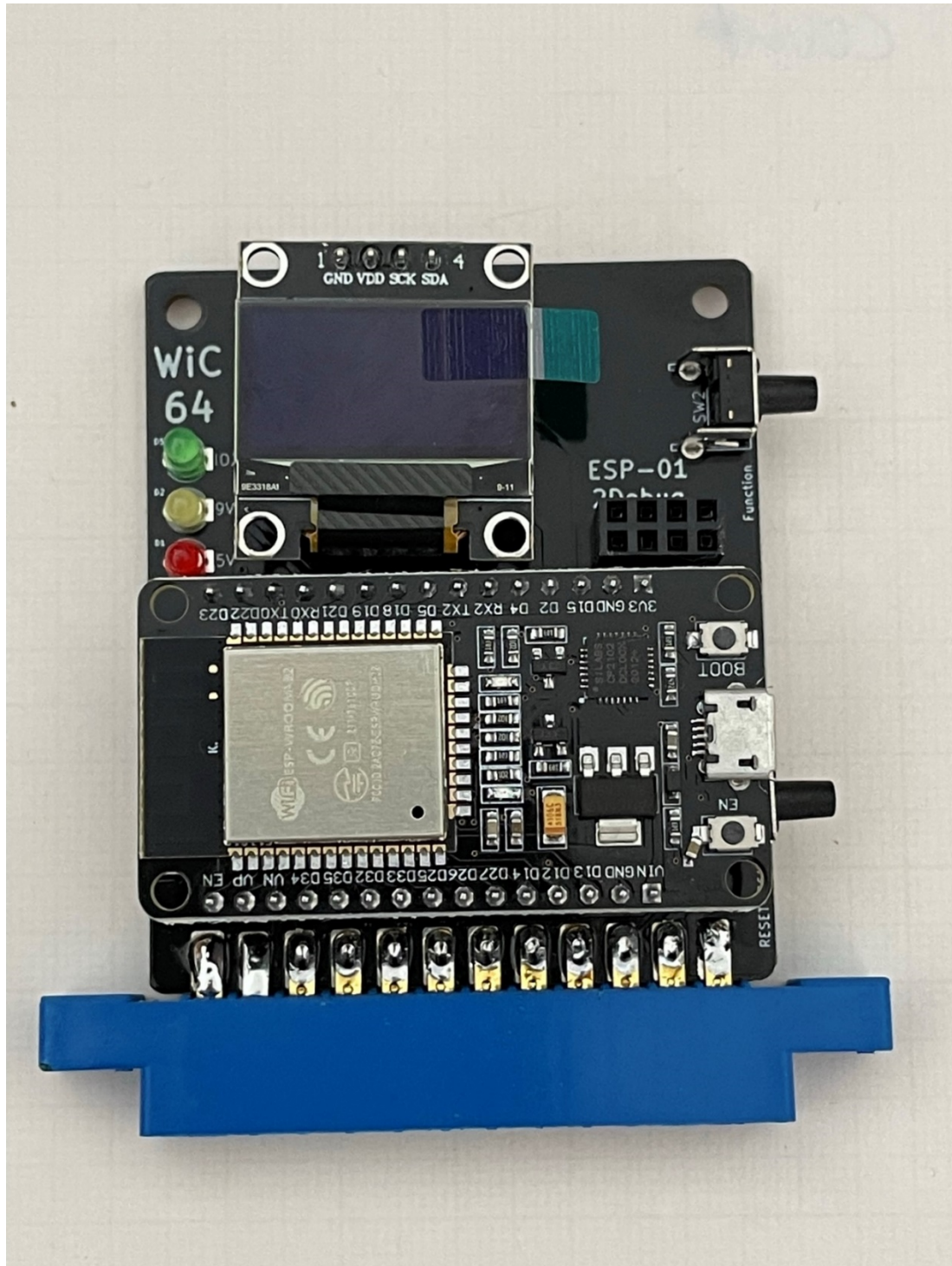
Jetzt den ESP32 auf die Platine aufstecken. Bitte achtet auf die richtige Orientierung. Die Antenne muss sich unter das Aussparung der Platine befinden – **Der Micro-USB Anschluss befindet sich auf der Seite der beiden seitlichen Taster:**



Zum Schluss wird noch das OLED Display auf die Platine gesteckt. Der Zusammenbau des WiC64 ist damit abgeschlossen.

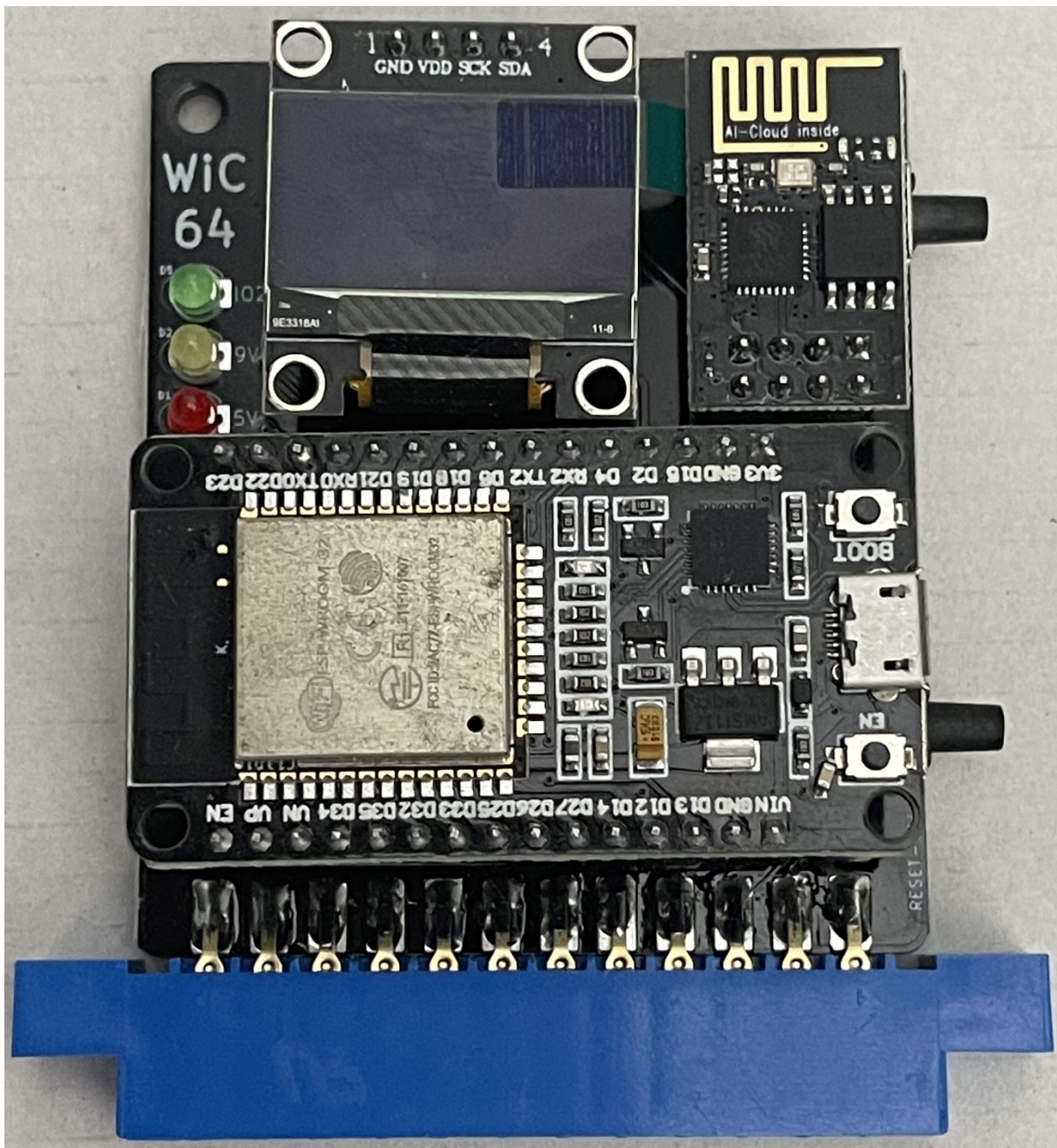
Solltet Ihr den WiC64 selbst gebaut haben, oder den ESP32 unbespielt gekauft haben, dann muss der ESP32 noch mit der WiC64 Firmware bespielt werden ! Diese findet Ihr inklusive Flasher auf

<http://www.wic64.de>



Fertig !

Optional für Entwickler: ESP01-Addon



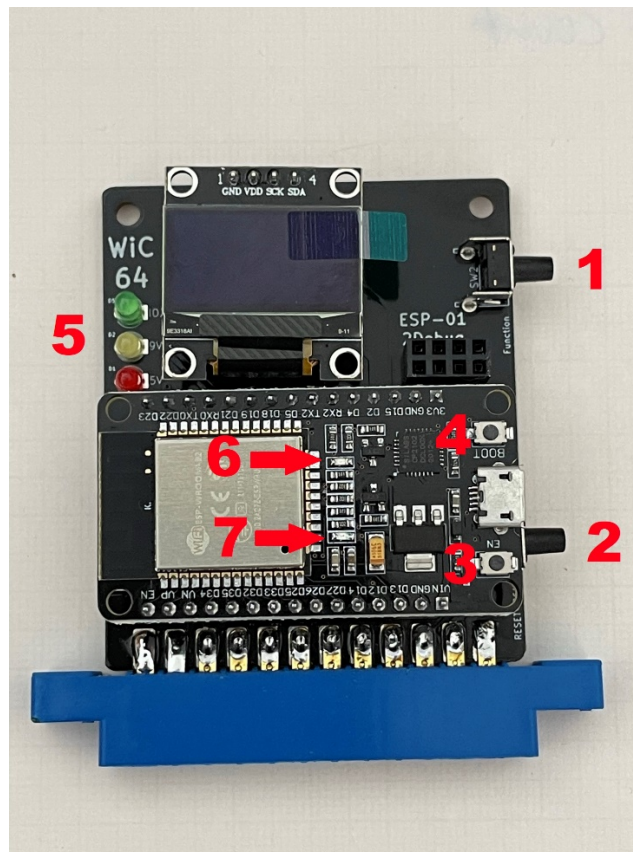
Optional kann der Wic64 noch mit einem weiteren kleinen WLAN-Chip des Typs „ESP-01“ bestückt werden. Mit diesem Chip kann man „live“ die Daten Ein- und Ausgaben zwischen dem C64 und dem ESP32 beobachten. Dazu muss der ESP-01 mit einer speziellen Software namens ESP-Link bespielt werden. Dies bietet sich aber nur für Entwickler an, die den Micro-USB Anschluss des ESP32 via Kabel nicht nutzen können oder wollen.

Für den ganz normalen Betrieb des Wic64 ist diese Erweiterung nicht erforderlich – von der Erweiterung raten wir allen normalen Benutzern dringend ab.

Eine Anleitung und die benötigte Software findet ihr im Forum64 oder auf

<http://www.wic64.de>

Übersicht:



1: Funktionstaste – Drückt man diese 5 Sekunden werden die Aktivitäts-LEDs deaktiviert. Nochmaliges drücken aktiviert sie wieder

2: RESET – Löst bei C64 einen RESET aus

3: EN (ESP32) – Löst beim ESP32 einen RESET aus

4: BOOT (ESP32) – Drückt man diese 5 Sekunden wird das WiC64 komplett deaktiviert. Nochmaliges drücken aktiviert es wieder. Benötigt, um Kompatibilität mit Speeddos und anderen Programmen her zu stellen die auch den parallelen Bus des Userport benutzen.

5: Power LEDs:

Grün = an: C64 sendet an den ESP32 Daten

Grün = aus: ESP32 sendet an den C64 Daten

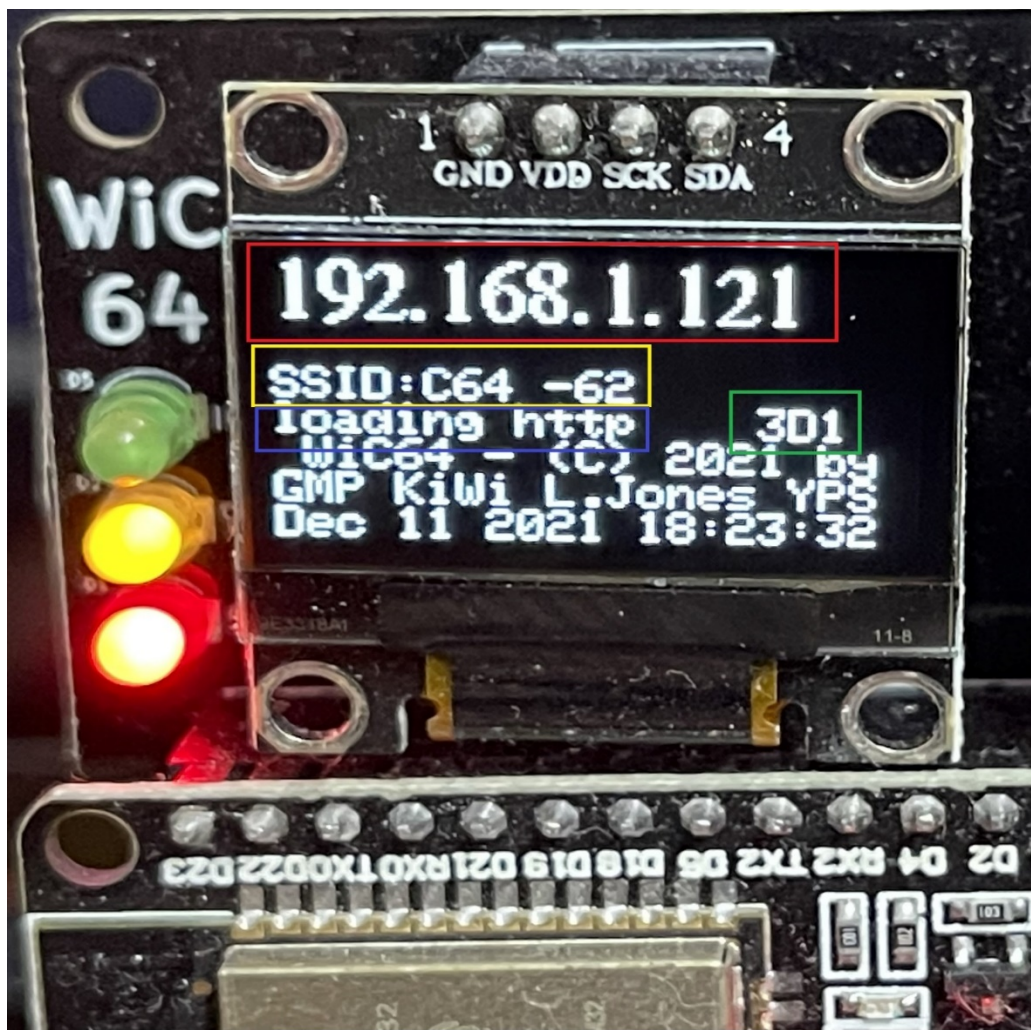
Gelb = 9V Wechselspannung liegen am Userport an – Dient zur Diagnose bei Problemen. Siehe auch: SX-64 Userport Fix

Rot = 5V Gleichspannung werden für den ESP32 generiert. Dient zur Diagnose bei Problemen.

6: Blaue LED ESP32 – Gleiche Funktion wie die grüne LED. Leuchtet sie blau sendet der C64, ist sie aus werden Daten vom ESP32 empfangen.

7. Rote Power LED ESP32 – Dient zur Diagnose bei Problemen.

Display:



Rot: Aktuelle IP-Adresse des WiC64. Sollte der WiC64 die WLAN-Verbindung im Betrieb verlieren spring diese sofort auf 0.0.0.0. Unter der angezeigten IP-Adresse kann auch das Web-Interface des WiC64 aufgerufen werden über das man zum Beispiel verschiedene Firmware-Versionen in den ESP32 laden kann.

Gelb: Name des WLANs mit dem der ESP32 gerade verbunden ist sowie der Empfangslevel in dB. Hier gilt auch: Je kleiner die Zahl ist desto besser der Empfang. Minus 30 wäre sehr gut und Minus 90 ist ein zu schwaches Signal.

Blau: Name des Kommandos was der C64 an den ESP32 geschickt hat. Primär nur für Programmierer interessant.

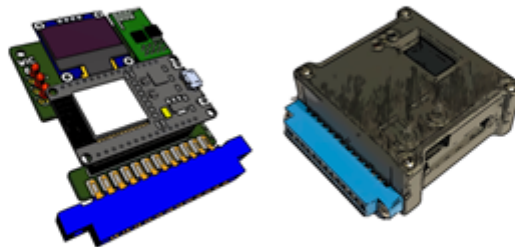
Grün: Anzahl der seit dem Systemstart empfangenen Kommandos in Hexadezimal. Interessant für Dauer bzw. Stabilitätstests. \$03D1 = 977 Durchläufe seit Reboot.

Unterste Zeile: Versionsdatum der installierten Firmware. Diese sollte mit dem Firmware-Updater-Tool möglichst immer aktuell gehalten werden.

Software downloaden:

Für den Betrieb des WiC64 benötigt man diverse Programme. Mindestanforderung ist jedoch nur der „wic64-launcher.prg“ von der WiC64 Download-Seite:

<http://www.wic64.de>



The Wireless interface for
C64 - SX64 - C128 - VIC20




> **WiC64 Portal Launcher** < > Programs & Documents <

> Contact Developer < > Order your own WiC64 <

hier bitte die aktuelle Version des Start-Programms herunterladen und auf ein Speichermedium für den C64 Übertragen (SD2IEC / U2+ usw.)



Programs & Documents

TITEL	KATEGORIEN	VERSION	UPDATED	DOWNLOAD
 WiC64 Portal Launcher 8. Januar 2022 1.30 KB 1298 Downloads	Portal, Programs	2.0	8. Januar 2022	DOWNLOAD
 WiC64 Telnet Demo 1.1 6. Januar 2022 4.15 KB 69 Downloads	Programs, Test		6. Januar 2022	DOWNLOAD
 WiC64 Koala Loader 1.5 – 500 Pictures Edition 6. Januar 2022 4.62 KB 42 Downloads	Demo		6. Januar 2022	DOWNLOAD
 ESP-01 Flasher ESP-LINK WiC64 Debugging 8. November 2021 242.89 KB 79 Downloads	Module, Programs	esp-link v2.2.3	8. November 2021	DOWNLOAD
 WiC64 Set NTP Timezone 3. November 2021 3.97 KB 79 Downloads	Module, Programs	1.0	3. November 2021	DOWNLOAD

Erstinbetriebnahme C64:

Um das WiC64 für den Erstbetrieb mit dem Internet zu Verbinden gibt es drei verschiedene Möglichkeiten das WLAN zu konfigurieren:

WLAN-Config.prg

Hier gibt man einfach seine WLAN-Zugangsdaten ein. Hat man Sonderzeichen in SSID oder Passwort, bitte den Absatz Sonderzeichen lesen.

```
----- WiC64 WLAN CONFIG 1.2 BY KiWi -----  
  
SSID: C64  
Password: 08154711  
wLAN CONFIG CHANGED  
ready.  
■
```

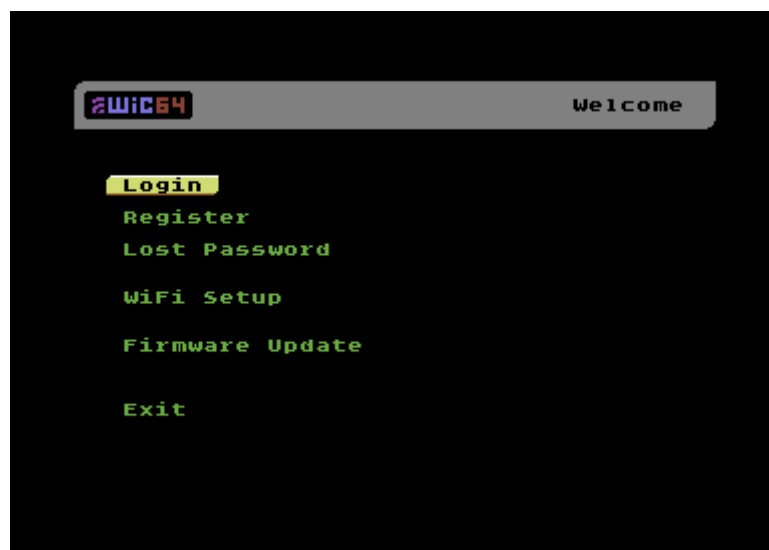
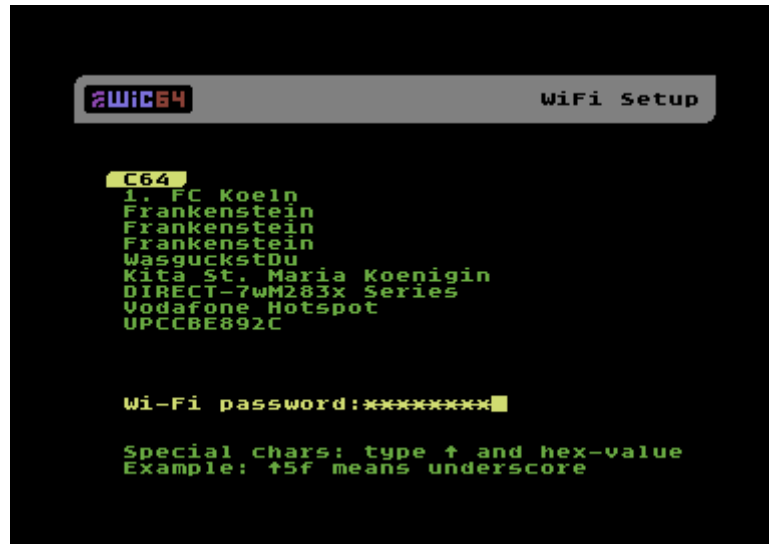
WLAN-Scanconfig.prg

Hier gibt man nur noch das Passwort ein – Alle Empfangbaren WLAN SSID's werden automatisch aufgelistet. Je besser der Empfang desto niedriger ist der Empfangswert. Hat man Sonderzeichen im Passwort, bitte ebenfalls den Absatz Sonderzeichen lesen.

```
----- WiC64 WLAN SCAN CONFIG 1.2 BY KiWi -----  
  
SCANNING NETWORKS - PLEASE WAIT ....  
0 c64 -60  
1 fRANKENSTEIN -70  
2 fRANKENSTEIN -72  
3 l. fc KOELN -72  
4 fRANKENSTEIN -92  
5 fRANKENSTEIN -93  
  
SSID NUMBER: 0  
Password: 08154711  
wLAN CONFIG CHANGED  
ready.  
■
```

wic64-launcher.prg

Die sicherlich beste und einfachste Methode das WiC64 mit dem Internet zu verbinden, um danach direkt ins WiC64 Portal einzutauchen, um mit anderen Leuten zu Chatten oder Spiele zu laden. Auch hier bei Sonderzeichen den nächsten Abschnitt beachten.



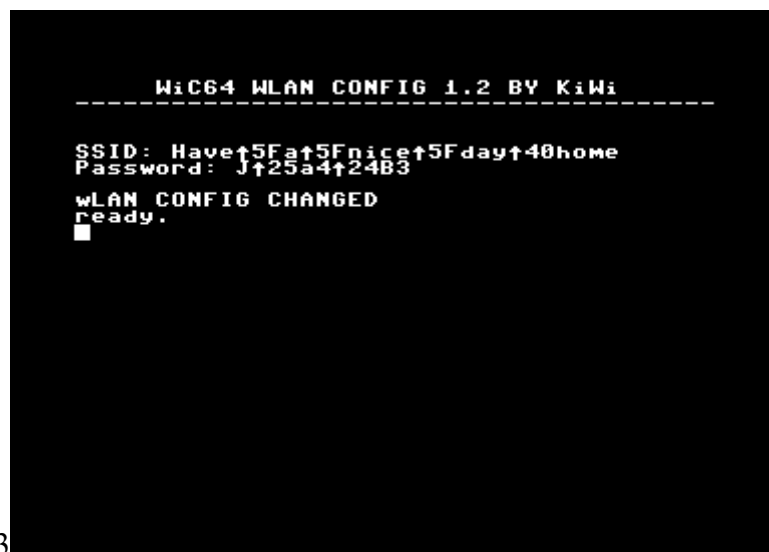
Sonderzeichen in SSID oder Passwort

Sonderzeichen werden mit dem Pfeil nach Oben (neben der Restore Taste) folgend vom hexadezimalen ASCII Wert eingegeben. Beispiel 5F = _ oder 25 = % oder 24 = \$

WLAN SSID: Have_a_nice_day@home

WLAN Passwort: J%a4\$B3

Funktioniert mit jedem der WLAN Konfigurationsprogramme. Beispiel:



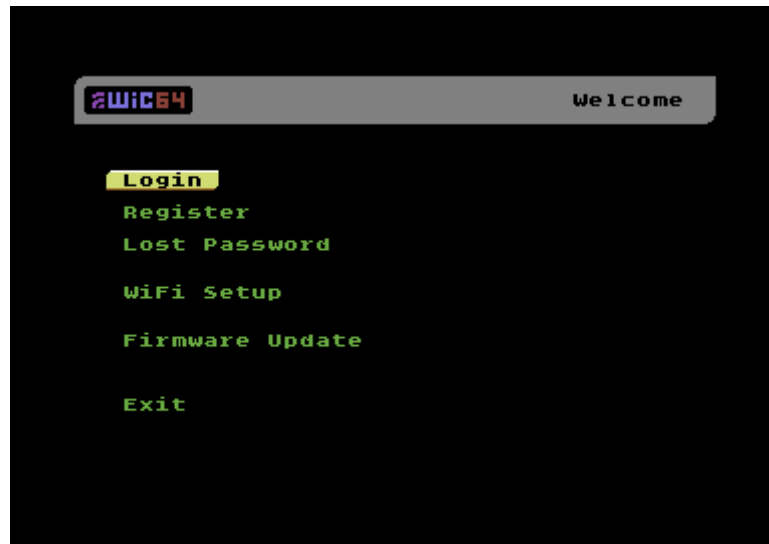
ASCII-Zeichentabelle, **hexadezimale** Nummerierung

Code	...0	...1	...2	...3	...4	...5	...6	...7	...8	...9	...A	...B	...C	...D	...E	...F
0...	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1...	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2...	SP	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
3...	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4...	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5...	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	_
6...	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7...	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

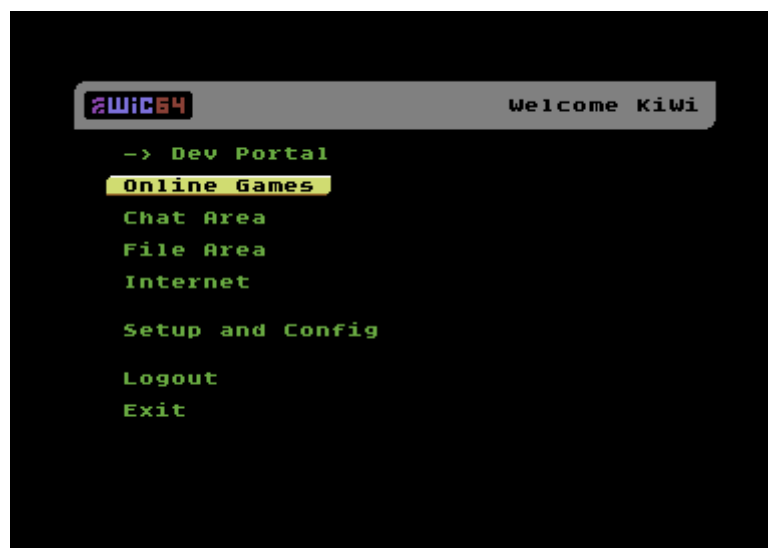
Quelle: Wikipedia

wic64-launcher.prg / WiC64 Portal

Wenn man mit dem wic64-launcher.prg sein WLAN konfiguriert hat und dies auch erfolgreich war, dann kann man von dort aus direkt auf das WiC64 Portal zugreifen. Dort kann man sich einfach und kostenlos Registrieren:



Das Portal selbst stellt viele verschiedene Funktionen zur Verfügung, darunter zum Beispiel auch eine File-Area, in der man immer die neusten Spiele herunterladen kann. Einige Funktionen sind (Stand 21.12.2021) noch nicht für die Öffentlichkeit verfügbar da das Portal sich noch in der Entwicklung befindet.



Von hier aus kann man auch den WiC64 Chat aufrufen und mit anderen Leuten auf der Welt chatten und bald auch Spiele gegeneinander spielen.

```
WIC64 Chat v1.1
initialization...

Welcome KiWi
Login into chatroom...
Login successful

User online: 2

KiWi
_____

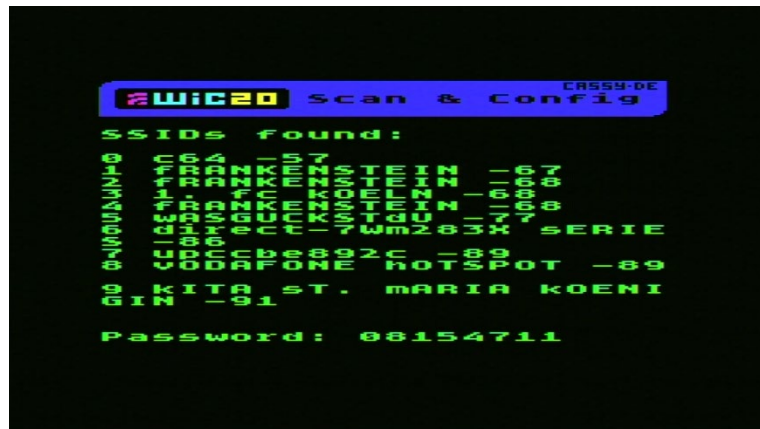
F1 = Switch Z&Y keys (beep)
F3 = Pause (border becomes grey)
F8 = Logoff and exit Chat

==> Press space to enter chatroom <==
```


Erstinbetriebnahme VC20:

Um das WiC64 für den Erstbetrieb mit dem VC20 konfigurieren muss man sich das WLAN-Konfigurationsprogramm [wic20-scanconfig.prg](https://github.com/cassy-de/WiC20) von <https://github.com/cassy-de/WiC20> laden.

Ihr findet es im Ordner /VC20PRGS/ Hier gibt man einfach die Nummer seines WLANs und das dazu gehörige Passwort ein. Hat man Sonderzeichen in SSID oder Passwort, bitte auch hier den den Absatz Sonderzeichen lesen.

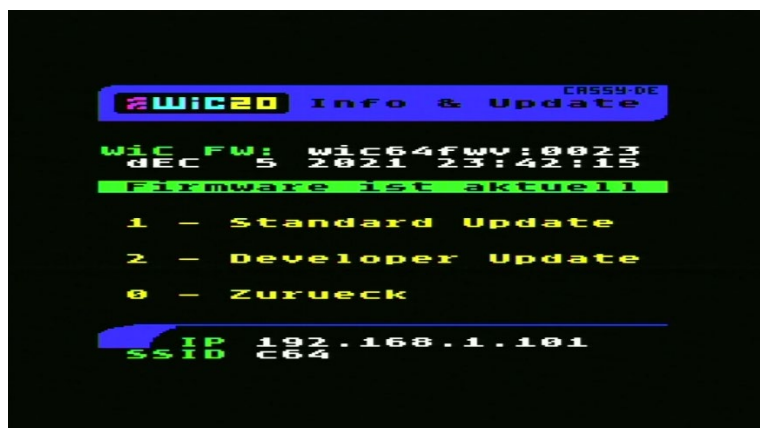


main.prg

Ebenfalls auf Cassy's Github Account findet ihr dann das main.prg für den VC20 mit dem Ihr Spiele, Demos und Tools laden könnt. Es bietet eine breite Auswahl an hochklassiger Software an.

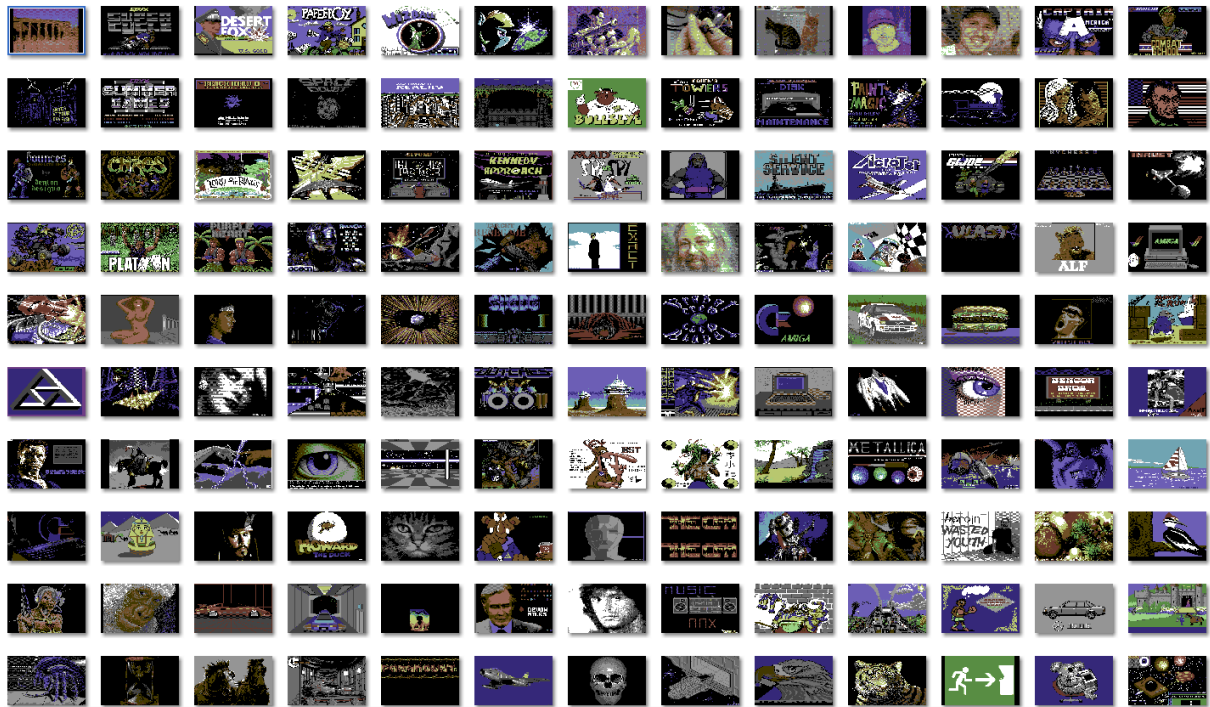


Auch ein Tool für das Updaten der WiC64 Firmware ist auf dem VC20 vorhanden:



Weitere Testprogramme:

Der Koala-Loader ist eines der ersten Programme, die für den WiC64 entwickelt wurden. Er ist ein Stabilitätstest für sowohl die Verbindung ESP32 zum C64 als auch der Internet- und WLAN Verbindung. In der aktuellen 1.5er Version lädt er 500 Koala Bilder so schnell wie es geht aus dem Internet und zeigt sie an:



Firmware Updater:

Hiermit kann die Firmware direkt über das Internet auf den aktuellen Stand gebracht werden. Der Updater lädt die ESP32 Firmware vom WiC64 Server, überprüft sie auf Richtigkeit und aktualisiert sie dann.

```
-----
WiC64 FIRMWARE UPDATER 1.3 BY KiWi
-----

1 - NORMAL FIRMWARE UPDATE
2 - DEVELOPER FIRMWARE UPDATE
0 - EXIT

Installed: DEC 11 2021 18:23:320
NO FIRMWARE UPDATE AVAILABLE

RUNNING UPDATE - SEE WIC64 DISPLAY
PLEASE WAIT 30 SECONDS FOR ESP32 REBOOT
THEN PRESS 0 TO RESTART
```

Test Tool

Vom Test-Tool ist primär der Source-Code interessant da man hier diverse Funktionen auf dem ESP32 ausprobieren kann. Einige Funktionen sind noch in der Entwicklung.

```
-----
WiC64 TEST TOOL 1.2 BY KiWi 2021
-----

1 - HTTPGET DEMO README.TXT FROM SERVER
2 - CONFIG WLAN c64 / 08154711
3 - FIRMWARE UPDATE
4 - DEVELOPER FIRMWARE UPDATE
5 - DEVELOPER DEBUG FIRMWARE UPDATE
6 - GET IP ADDRESS
7 - GET FIRMWARE VERSION
8 - SET DEFAULTSERVER WWW.WIC64.DE/PRG/
9 - REM COMMAND
A - GET UDP PACKAGE
B - SEND UDP PACKAGE
C - GET SSID LIST
D - GET SSID VIA LIST
E - SET UDP PORT 8080
F - SEND HTTP URL DECODED FOR PHP
G - GET CONNECTED WLAN NAME
H - GET RSSI WLAN LEVEL
I - GET DEFAULT SERVER
J - GET EXTERNAL IP ADDRESS
K - GET WIC64 MAC ADDRESS
L - WIC64 FACTORY RESET
```


Set Time Zone:

Dem ESP32 wird die aktuelle Zeitzone einprogrammiert. Diese wird permanent im ESP32 gespeichert und muss daher nur ein einziges Mal eingestellt werden.

```
----- WiC64 SET NTP TIME ZONE 1.1 BY KiWi -----
00 Greenwich Mean Time GMT +1:00
00 European Central Time GMT +2:00
00 Eastern European Time GMT +2:00
00 Arabian Standard Time GMT +3:00
00 Indian Standard Time GMT +5:30
00 Japan Standard Time GMT +9:00
00 Australian Central Time GMT +10:30
00 Australian Eastern Time GMT +11:00
00 New Zealand Standard Time GMT +12:00
00 Hawaii Standard Time GMT -11:00
00 Alaska Standard Time GMT -10:00
00 Pacific Standard Time GMT -8:00
00 Pacific Standard Time GMT -8:00
00 Mountain Standard Time GMT -7:00
00 Mountain Standard Time GMT -7:00
00 Central Standard Time GMT -6:00
00 Central Standard Time GMT -6:00
00 Eastern Standard Time GMT -5:00
00 Atlantic Standard Time GMT -4:00
00 Canada Newfoundland Time GMT -3:30
Time Zone (00-31):
```

Danach kann man mit Time and Date prüfen, ob die Uhrzeit via Internet richtig übermittelt wird beziehungsweise die Zeitzone richtig eingestellt wurde:

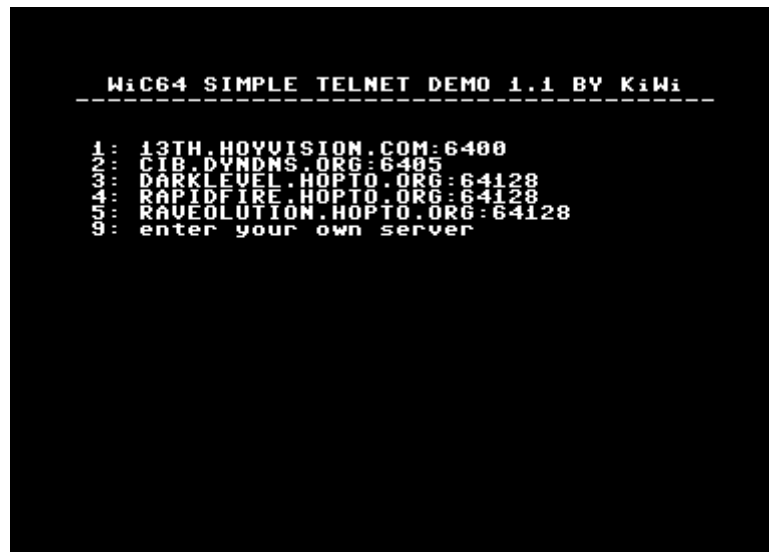
```
----- WiC64 NTP TIME AND DATE 1.0 BY KiWi -----

Time and Date: 21:01:12 21-12-2021

ready.
█
```

Telnet Demo:

Ahmt die Funktionsweise eines seriellen WLAN Modems nach. Es wird eine Telnet Verbindung zu einem Server aufgebaut und alles 1:1 auf den Bildschirm geprinted. Ist noch sehr rudimentär. Mit F1 kommt man immer zum Start zurück. Mit F3 kann man schnell längere Strings schicken, anstatt einzelne Zeichen nacheinander zu übermitteln (Mit 2x Return abschicken) und F7 schickt einen alternativen Returncode an den Server.



WiC64 Test Kernal:

Einfaches gepatchtes Kernal, um per Load Befehl Programme aus dem Internet zu laden.

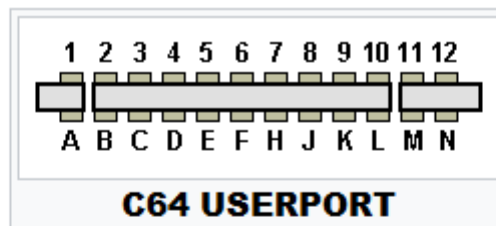
```
*** COMMODORE WIC64 BASIC V3 **  
64K RAM SYSTEM  38911 BASIC BYTES FREE  
READY.  
L"HTTP://WWW.WIC64.DE/PRG/KONG.PRG"  
LOADING  
READY.  
L\  
10 SYS18432  
READY.
```


Programmierung des WiC64:

Die Programmierung des WiC64 ist kinderleicht. Es werden einfache Befehle an das WiC64 gesendet, und das WiC64 antwortet dann passend zum Befehl.

Steuerung:

Das WiC64 ist am Userport über 8 Datenleitungen, zwei Handshake-Leitungen und eine Steuerungsleitung mit dem C64 verbunden. Die Versorgungsspannung von 5V Gleichspannung wird aus den 9V Wechselspannung generiert. **SX-64 Benutzer beachten bitte die Hinweise in dieser Anleitung wg. Kurzschlussgefahr / Fehlerbehebung SX-64 !**



Pin	Signal	Bemerkung	Pin	Signal	Bemerkung
Oberseite			Unterseite		
1	GND		A	GND	
2			B	FLAG 2	Flag 2
3			C	PB 0	Bit 0 (CIA 2)
4			D	PB 1	Bit 1
5			E	PB 2	Bit 2
6			F	PB 3	Bit 3
7			H	PB 4	Bit 4
8	PC2	Port Control	J	PB 5	Bit 5
9			K	PB 6	Bit 6
10	9 V AC	max. 100 mA	L	PB 7	Bit 7
11	9 V AC	max. 100 mA	M	PA 2	Bit 2 (CIA 2)
12	GND		N	GND	

PB0 – PB7 = 8 Datenleitung bidirektional (Lesen / Schreiben)

PC2 – Löst automatisch einen Interrupt aus (LOW/HIGH) wenn PB0-PB7 beschrieben werden

FLAG2 – Löst im CIA einen Interrupt aus wenn Daten an PB0-PB7 abgeholt werden können.

PA2 – Wenn LOW kann der ESP32 an den C64 senden. Wenn HIGH geht der ESP32 in den Empfangsmodus.

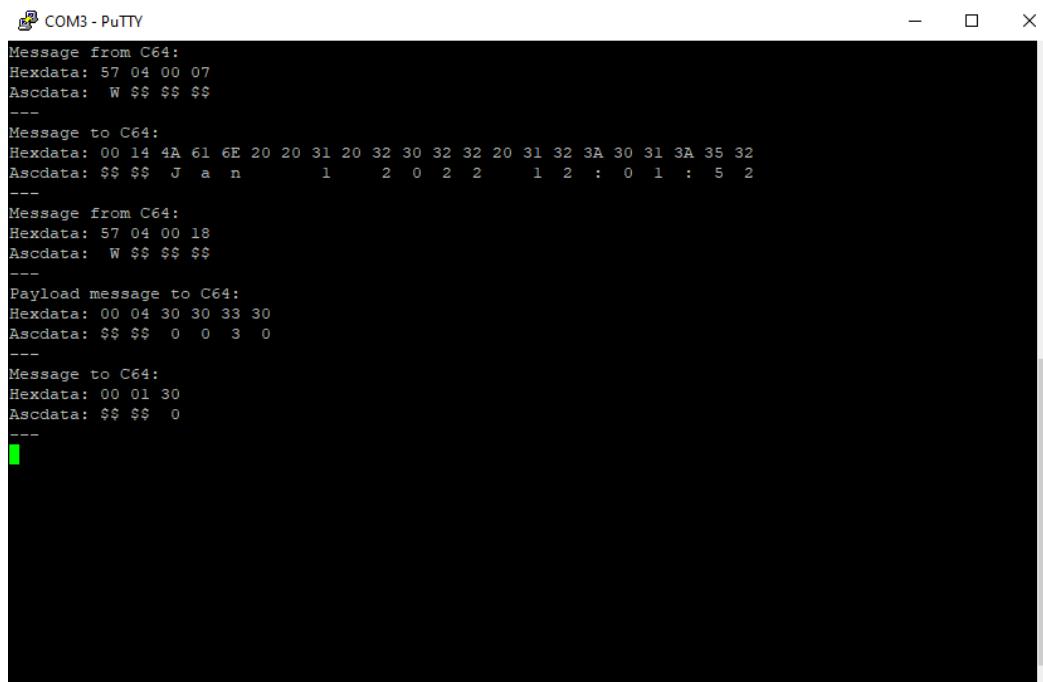
Serielle DEBUG Konsole des WiC64:

Um die Programmierung des WiC64 zu erleichtern kann man das WiC64 über den Micro-USB Anschluss an den PC anschließen und mit einem Terminal-Programm wie zum Beispiel Putty (<https://www.putty.org/>) bei 115.200 Baud die Ein und Ausgaben des WiC64 verfolgen.

Dazu muss der Serielle Treiber des ESP32 installiert sein (Findet man auf WiC64) und es muss die **DEBUG FIRMWARE 2** installiert sein. Bitte mit dem Firmware-Updater vorher prüfen!

Alternativ kann eine MEGA-DEBUG-FIRMWARE mit der Taste 9 im Firmware-Updater installiert werden. Diese gibt aber sehr viele tiefgreifende System-Informationen über die laufenden Prozesse aus, dass sich diese Art von Debugging nur für ESP32 Programmierer lohnt.

Normale Benutzer sollten IMMER die Standard-Firmware nutzen, und nur Entwickler die wirklich loggen wollen sollten auch nur dann die Developer-Firmware installieren.



```
COM3 - PuTTY
Message from C64:
Hexdata: 57 04 00 07
Ascddata: W $ $ $ $ $
---
Message to C64:
Hexdata: 00 14 4A 61 6E 20 20 31 20 32 30 32 32 20 31 32 3A 30 31 3A 35 32
Ascddata: $ $ $ $ J a n      1      2 0 2 2      1 2 : 0 1 : 5 2
---
Message from C64:
Hexdata: 57 04 00 18
Ascddata: W $ $ $ $ $
---
Payload message to C64:
Hexdata: 00 04 30 30 33 30
Ascddata: $ $ $ $ 0 0 3 0
---
Message to C64:
Hexdata: 00 01 30
Ascddata: $ $ $ $ 0
---
```

Message from C64 sind immer Bytes die der C64 an den ESP32 gesendet hat.

Message to C64 sind immer Bytes die der ESP32 an den C64 gesendet hat.

Payload message to C64 sind immer Bytes die der ESP32 vom Netzwerk geladen hat und dann an den ESP32 gesendet hat. Die Ausgabe erfolgt ausschließlich wenn die Netzwerkantwort KLEINER als 21 Zeichen ist. Dient zum Beispiel zum Prüfen von Ausgaben von einfachen PHP Scripten mit kurzen Antworten (0/1/OK/ERROR/OFFLINE usw.)

Bei den nun folgenden Beispielen wird immer zusätzlich die Ausgabe des Debug Output gezeigt um die Kommunikation einfacher zu Erklären.

Beispiel in Assembler:

Wir schalten den WiC64 in den Empfangsmodus und senden einen Befehl:

```
lda $dd02
ora #$04      Datenrichtung Port A, Leitung PA2 auf Ausgang
sta $dd02

lda $dd00
ora #$04      PA2 auf HIGH = ESP im Empfangsmodus
sta $dd00

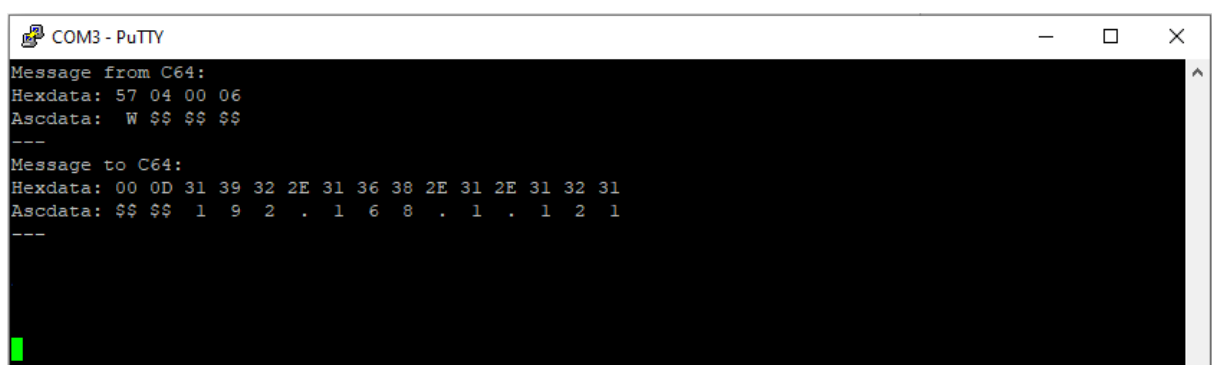
lda #$ff      Datenrichtung Port B Ausgang
sta $dd03
```

Die blaue LED auf dem ESP32 geht an, der ESP32 ist im Empfangsmodus.

Jetzt senden wir mal ein 4 Byte Kommando – ohne Schleife – an den ESP32:

```
lda #"W"      1. Zeichen ist immer ein W fuer WiC64
jsr write_byte
lda #$04      Laenge des Kommandos $0004 Byte total
jsr write_byte
lda #$00      Die Laenge high byte ($0004)
jsr write_byte
lda #$06      Kommando Nummer $06
jsr write_byte
rts

write_byte:
sta $dd01      Bit 0..7: Userport Daten schreiben
dowrite:
lda $dd0d
and #$10      Warten auf NMI FLAG2 = Byte wurde
beq dowrite    gelesen vom ESP
rts
```



```
COM3 - PuTTY
Message from C64:
Hexdata: 57 04 00 06
Ascdta:  W $ $ $ $
---
Message to C64:
Hexdata: 00 0D 31 39 32 2E 31 36 38 2E 31 2E 31 32 31
Ascdta:  $ $ $ $ 1 9 2 . 1 6 8 . 1 . 1 2 1
---
```

Man sieht in der Debug Konsole: Der C64 hat an den ESP32 das Kommando \$06 GET IP (\$57 \$04 \$00 \$06) gesendet und der ESP Antwortet mit der IP Adresse des ESP32:

192.168.1.121. Die ersten beiden Bytes sind hier das HIGH und das LOW Byte der Länge der Antwort (\$00 0D) Bytes = 13 Bytes dezimal.

Wir schalten den WiC64 in den Sendemodus und Empfangen mit dem C64 die Antwort:

```
lda #$00          Datenrichtung Port B Eingang
sta $dd03
lda $dd00
and #251          PA2 auf LOW = ESP im Sendemodus
sta $dd00
```

Jetzt holen wir mal die ersten beiden Bytes ab. Dazu müssen wir dem ESP32 mitteilen das wir Empfangen wollen. Dazu führen wir einen extra Lesezugriff aus:

```
jsr read_byte      Dummy Byte - IRQ anschieben

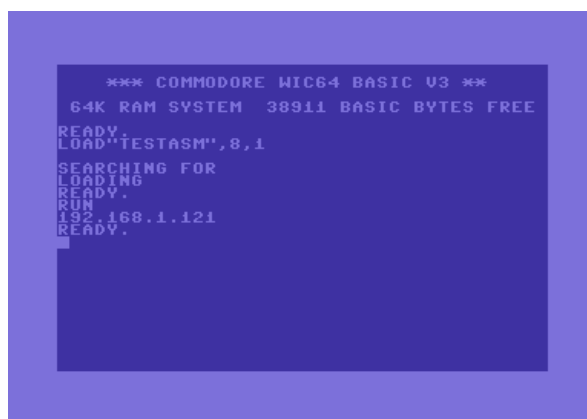
jsr read_byte      1. Byte abholen ($00)
tay
jsr read_byte      2. Byte abholen ($00)
tax

errorcheck:
cpy #$00
bne goread         Antwort groesser als $01XX bytes
cpx #$00
bne goread         Antwort ist genau null Byte 00
                     00 = Keine Antwort verfügbar.
rts

goread:
jsr read_byte
jsr $ffd2          Byte auf dem Bildschirm ausgeben
dex
bne goread
dey
cpy #$ff
bne goread
rts

read_byte:
lda $dd0d
and #$10
beq read_byte      Warten auf NMI FLAG2 =
                     ESP32 hat Byte auf Port gelegt
lda $dd01
rts
```

Wir prüfen also ob die Antwort nicht \$0000 Lang ist und geben dann alle Zeichen auf dem Bildschirm aus:



Liste der Kommandos:

Die aktuelle WiC64 Firmware kennt folgende Kommandos:

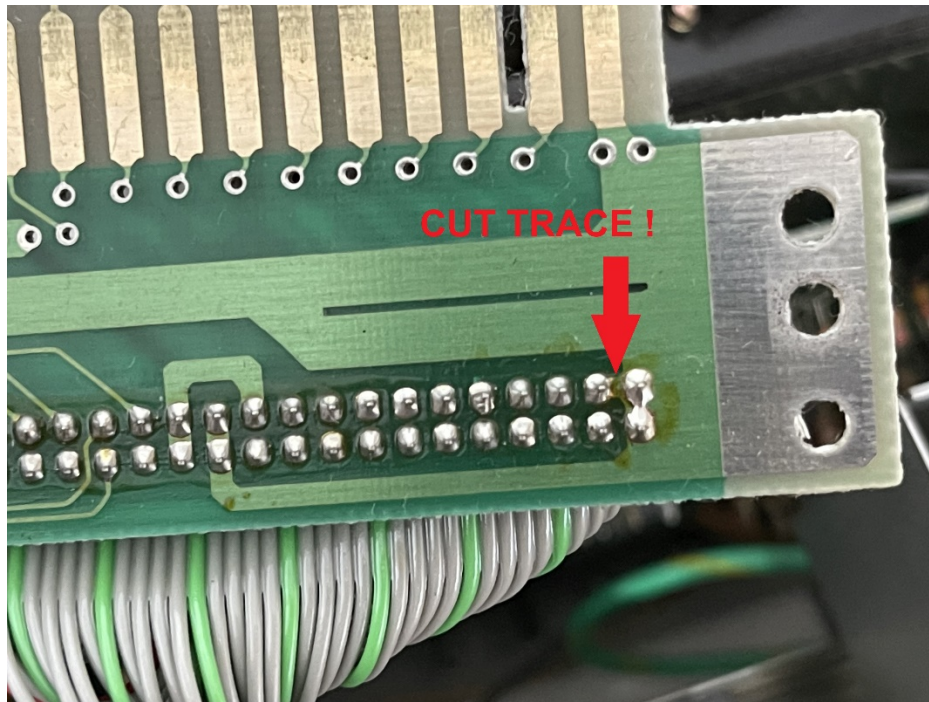
Kommando	Befehl
\$00	GET FIRMWARE VERSION
\$01	LOAD HTTP / HTTPS
\$02	CONFIG WIFI
\$03	FIRMWARE UPDATE STANDARD
\$04	FIRMWARE UPDATE DEVELOPER
\$05	FIRMWARE UPDATE ESP32 DEVELOPER
\$06	GET IP ADDRESS OF WIC64
\$07	GET FIRMWARE VERSION STATUS
\$08	SET DEFAULT SERVER
\$09	REM COMMAND FOR DEVELOPER CONSOLE
\$0A	GET UDP PACKAGE
\$0B	SEND UDP PACKAGE
\$0C	WLAN SCAN SSID
\$0D	CONFIG WLAN SCAN SSID
\$0E	CHANGE UDP PORT
\$0F	HTTP STRING CONVERSION FOR HTTP CHAT
\$10	GET ACTUAL CONNECTED SSID
\$11	GET ACTUAL WIFI SIGNAL RSSI
\$12	GET DEFAULT SERVER (SEE COMMAND \$08)
\$13	GET EXTERNAL IP OF WIC64 INTERNET CONNECTION
\$14	GET MAC ADDRESS OF WIC64
\$15	GET TIME AND DATE FROM NTP SERVER
\$16	SET TIMEZONE OF NTP SERVER
\$17	GET TIMEZONE OF NTP SERVER
\$18	CHECK IF DEVELOPER FIRMWARE UPDATE IS AVAILABLE
\$19	CHECK IF STANDARD FIRMWARE UPDATE IS AVAILABLE
\$1A	READ PREFS STRING FROM EEPROM
\$1B	SAVE PREFS STRING TO EEPROM
\$1C	TCP – UNDER DEVELOPMENT
\$1D	TCP – UNDER DEVELOPMENT
\$1E	SET TCP PORT – UNDER DEVELOPMENT
\$21	CONNECT TO SERVER:PORT VIA TCP (TELNET)
\$22	GET DATA FROM TCP CONNECTION (\$1F)
\$23	SEND DATA TO TCP CONNECTION (\$1F)
\$63	DISCONNECT FROM WLAN, RESET DEFAULT SERVER AND RESET ESP32

Wer mehr über die Programmierung wissen will, schaut in die Quellcodes des Test-Tools.
Dort werden viele Live-Beispiele erklärt:

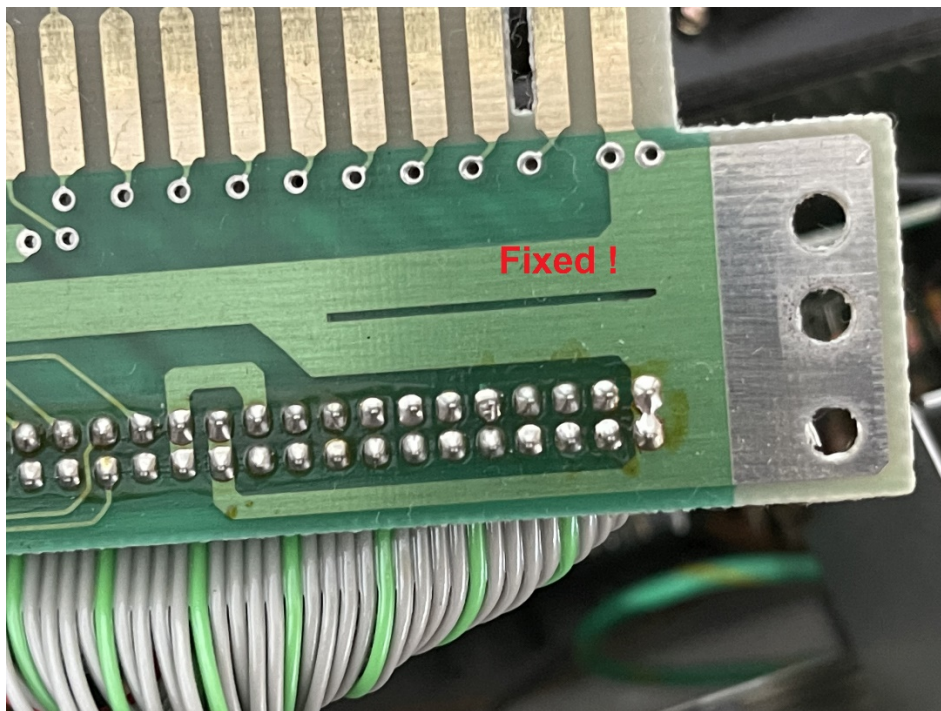
```
-----
WIC64 TEST TOOL 1.2 BY KiWi 2021
-----
1 - HTTPGET DEMO README.TXT FROM SERVER
2 - CONFIG WLAN C64 / 08154711
3 - FIRMWARE UPDATE
4 - DEVELOPER FIRMWARE UPDATE
5 - DEVELOPER DEBUG FIRMWARE UPDATE
6 - GET IP ADDRESS
7 - GET FIRMWARE VERSION
8 - SET DEFAULT SERVER WWW.WIC64.DE/PRG/
9 - EXEC COMMAND
10 - GET UDP PACKAGE
11 - SEND UDP PACKAGE
12 - GET SSID LIST
13 - SET SSID LIST
14 - GET UDP PORT 8080
15 - SEND HTTP URL DECODED FOR PHP
16 - GET CONNECTED WLAN NAME
17 - GET RSSI WLAN LEVEL
18 - GET DEFAULT SERVER
19 - GET EXTERNAL IP ADDRESS
20 - GET WIC64 MAC ADDRESS
21 - WIC64 FACTORY RESET
```

WICHTIG !! SX-64 WICHTIG !!

Es gibt beim SX-64 leider eine Inkompatibilität / Fehlbeschriftung ab Werk. Am Userport sollte eigentlich auf Pin 10 und 11 eine Wechselfspannung von 9V anliegen wie beim normalen C64 auch. Tut sie aber nicht da auf der Userport Adapterplatine im SX-64 dummerweise ein Kurzschluss auf Masse ist, der da nicht hingehört:



Nach dem Trennen dieser Leiterbahn mit einem Skalpell oder scharfen Schraubenzieher ist der Userport 100% kompatibel zum originalen C64 Userport. Eprommer oder Modems, die die 9V Wechselfspannung benötigen, können dann problemlos verwendet werden.



Quelle: <http://www.sx-64.de/sx-64-userport-fix-9v/>