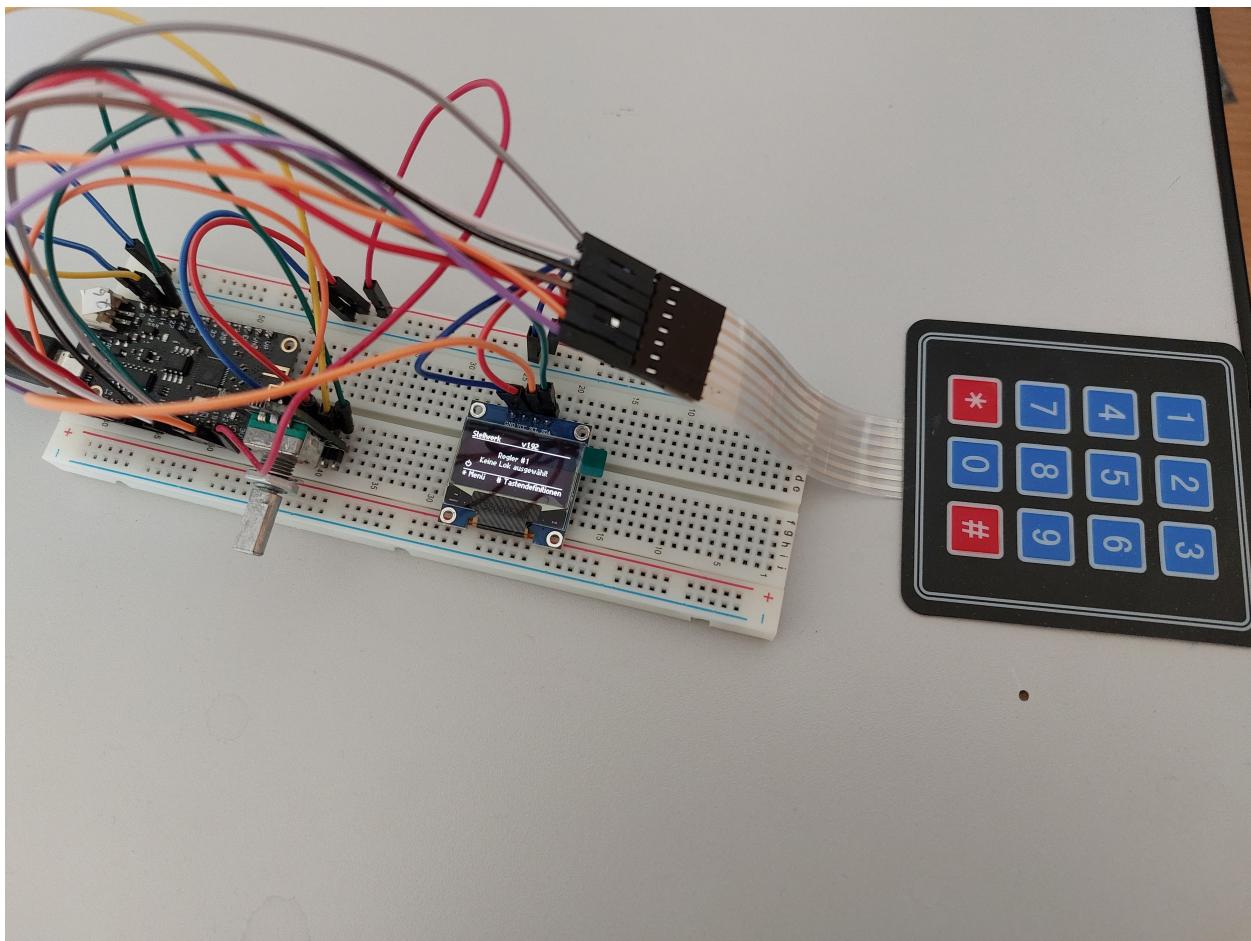


# DCC-EX v 5.4.10

09 - WiTcontroller - Basic - 4x3 - Tasten



Installation wurde unter Windows 11 sowie  
VSCode mit PlatformIO durchgeführt

Die folgenden Einstellungen und Vorgehensweise kann natürlich jeder selbst bestimmen.  
Es sind nur meine Erfahrungen.

Wichtig .....	2
Quelle .....	3
meine Versuchshardware .....	3
Verdrahtung .....	4
PIN Belegung .....	5
Programmdateien .....	6
Konfigurationsdateien .....	7
meine Anpassungen .....	8
config_buttons.h .....	8
OLED: .....	8
Folientastatur 4x3: .....	8
Übersetzungen .....	8
Benutzerdefinierter App-Name .....	8
zusätzliche / optionale Tasten .....	9
config_network.h .....	10
meine Netzwerkeinstellungen .....	10
WLAN-Ländercode .....	10
Automatische Verbindung zur ersten SSID .....	10
Automatische Verbindung zum ersten gefundenen Server .....	10

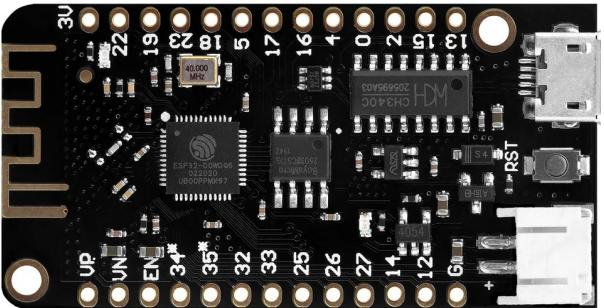
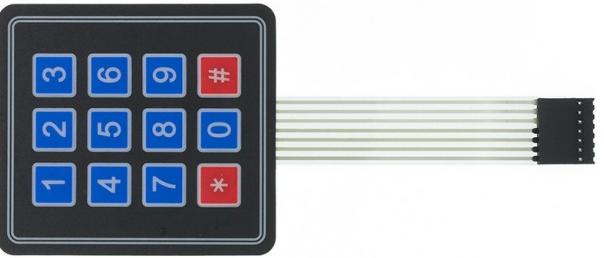
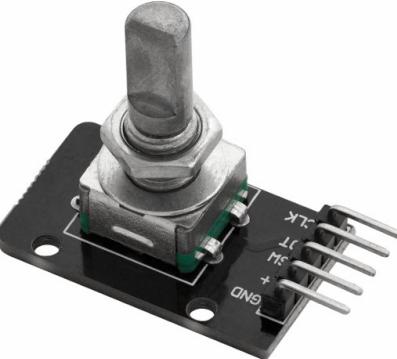
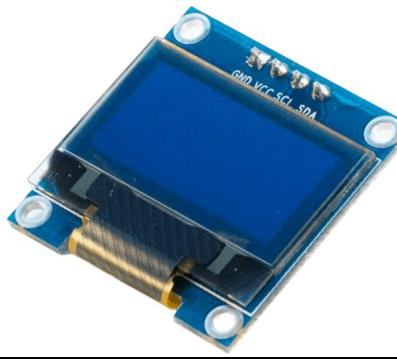
## Wichtig

- Ich übernehme keine Garantie/ Haftung auf Richtigkeit, Vollständigkeit usw. Es beruht alles auf eigener Erfahrung.
- Bei den nachfolgenden Schaltungs-/ Anschlussbeispielen ist unbedingt selbst auf die Pinbelegung zu achten, es gibt Bauelemente mit gleichen Daten aber anderer Pinbelegung.
- Auch auf die Spannungsversorgung muss unbedingt selbst geachtet werden, manche Komponenten benötigen 3,3 Volt, andere aber 5 Volt usw.

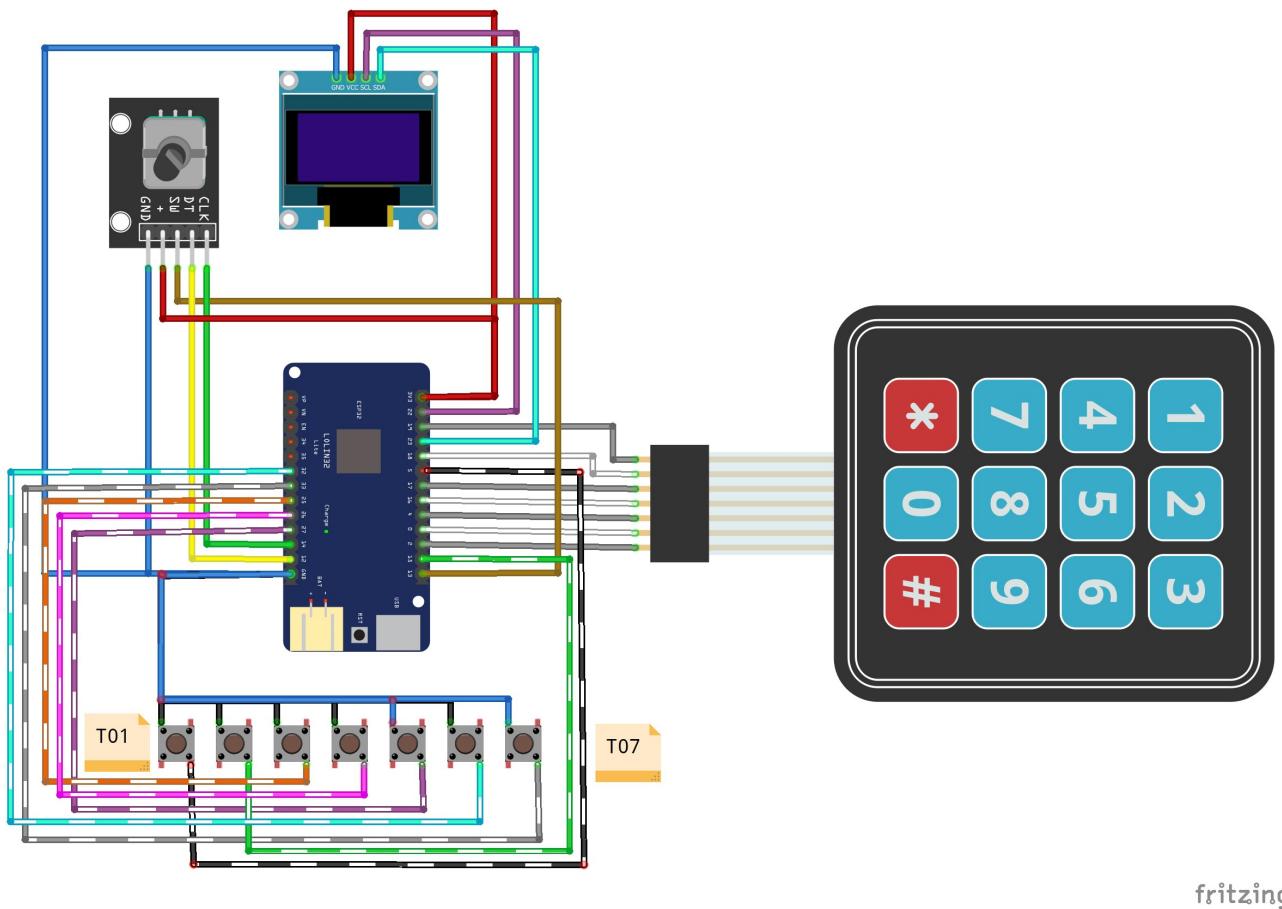
## Quelle

1. [GitHub - flash62au/WiTcontroller: WiTthrottle Protocol based WiFi model train controller for JMRI, DCC-EX etc.](#)
2. [WiTcontroller \(Physical\) — DCC-EX Model Railroading documentation](#)

## meine Versuchshardware

<b>ESP32 - Lolin32</b>	 A photograph of an ESP32-Lolin32 development board. It features an ESP32 chip, a CH340G USB-to-TTL converter, and various pins labeled with numbers 0 through 32 and letters A through D.
<b>4x3 Matrix Keypad Tastatur</b>	 A photograph of a 4x3 matrix keypad module. It has a black face with red and blue buttons labeled with digits 0-9, *, and #. A ribbon cable is attached to the back.
<b>KY-040 Drehgebermodul</b>	 A photograph of a KY-040 potentiometer module. It includes a metal potentiometer, a PCB, and a 6-pin header for connection.
<b>0,96 OLED SSD 1306</b>	 A photograph of a 0,96 inch OLED SSD 1306 display module. It has a blue screen and a gold-colored metal frame.
<b>7 x Tasten</b>	 A photograph of a 7x momentary pushbutton module. It consists of a single button with a green ring around it.

# Verdrahtung



fritzing

## PIN Belegung

PIN ESP32 Lolin32				
00	C2			
01				
02	C3			
03				
04	C1			
05				T01
06				
07				
08				
09				
10				
11				
12			DT	
13			SW	
14			CLK	
15				T02
16	R4			
17	R3			
18	R2			
19	R1			
20				
21				
22		SCL		
23		SDA		
24				
25				T03
26				T04
27				T05
28				
29				
30				
31				
32				T06
33				T07
34 (VP)				
35				
36				
37				
38				
39 (VN)				

Plus Stromversorgung z.B. USB Anschluss oder über Batterieanschluss 3,3 Volt

# Programmdateien

	[..]	<DIR>	24	
	[.pio]	<DIR>	24	
	[.vscode]	<DIR>	24	
	LICENSF		11.513	12
	Pangodream_18030_CL	cpp	0.104	15
	actions	h	1.924	19
	config_buttons	h	22.730	25
	config_buttons_example	h	23.968	19
	config_keypad/etc	h	1.989	15
	config_network	h	4.799	25
	config_network_example	h	4.266	18
	language_deutsch	h	15.943	18
	language_italiano	h	15.848	18
	Pangodream_18650_CL	h	3.400	15
	static	h	38.192	19
	WiTController	h	6.194	15
	platformio	ini	853	18
	WiTController	ino	138.270	23

- **Blaue Umrandung:** Dateien und Verzeichnisse werden von VSCode angelegt
- **Rote Umrandung:** eigentliche Programmdateien

# Konfigurationsdateien

1. Datei: „config\_buttons\_example.h“ in „config\_buttons.h“ umbenennen oder kopieren.
2. Datei: „config\_network\_example.h“ in „config\_network.h“ umbenennen oder kopieren.

# meine Anpassungen

## config\_buttons.h

### OLED:

```
#define OLED_TYPE U8G2_SSD1306_128X64_NONAME_F_HW_I2C u8g2(U8G2_R0, /*  
reset=*/ U8X8_PIN_NONE, /* clock= */ 22, /* data= */ 23);
```

```
// Dies ist eines der gängigen .9-ZOLL-OLED-Displays und standardmäßig im Lieferumfang enthalten  
#define OLED_TYPE U8G2_SSD1306_128X64_NONAME_F_HW_I2C u8g2(U8G2_R0, /* reset=*/ U8X8_PIN_NONE, /* clock= */ 22, /* data= */ 23),
```

### Folientastatur 4x3:

```
#define ROW_NUM 4  
#define COLUMN_NUM 3  
#define KEYPAD_KEYS {'1', '2', '3'}, {'4', '5', '6'}, {'7', '8', '9'}, {'*', '0', '#'}  
#define KEYPAD_ROW_PINS { 19, 18, 17, 16}  
#define KEYPAD_COLUMN_PINS { 4, 0, 2}
```

```
// Folientastatur 4x3 Standard  
#define ROW_NUM 4  
#define COLUMN_NUM 3  
#define KEYPAD_KEYS {'1', '2', '3'}, {'4', '5', '6'}, {'7', '8', '9'}, {'*', '0', '#'}  
#define KEYPAD_ROW_PINS { 19, 18, 17, 16}  
#define KEYPAD_COLUMN_PINS { 4, 0, 2}
```

### Übersetzungen

```
#include "language_deutsch.h"
```

```
// German - Deutsche  
#include "language_deutsch.h"
```

### Benutzerdefinierter App-Name

```
#define CUSTOM_APPNAME "Stellwerk"
```

```
//  
#define CUSTOM_APPNAME "Stellwerk"
```

## **zusätzliche / optionale Tasten**

Hier z.B.

- 3 Tasten T01 - T03
- **Wichtig:** Die angegebene Anzahl hier 3, muss mit der Anzahl der jeweiligen Einstellung übereinstimmen.
- belegte PIN's = 5, 15, 25
- ohne externen Pullup - Widerstand
- T01 = FUNCTION\_0 = Licht der zu steuernden Lok (siehe Datei „actions.h“)
- T02 = POWER\_ON = Anlage einschalten (siehe Datei „actions.h“)
- T03 = POWER\_OFF = Anlage ausschalten (siehe Datei „actions.h“)

```
#define USE_NEW_ADDITIONAL_BUTTONS_FORMAT true
```

```
#define USE_NEW_ADDITIONAL_BUTTONS_FORMAT true
```

```
#define NEW_MAX_ADDITIONAL_BUTTONS 3
#define NEW_ADDITIONAL_BUTTON_ACTIONS {FUNCTION_0, POWER_ON,
POWER_OFF}
#define NEW_ADDITIONAL_BUTTON_LATCHING {true, true, true}
```

```
// Muss 1 oder größer sein
#define NEW_MAX_ADDITIONAL_BUTTONS 3
#define NEW_ADDITIONAL_BUTTON_ACTIONS {FUNCTION_0, POWER_ON, POWER_OFF}
#define NEW_ADDITIONAL_BUTTON_LATCHING {true, true, true}
```

```
; {FUNCTION_0, POWER_ON, POWER_OFF}
```

siehe Datei „actions.h“

```
#define NEW_ADDITIONAL_BUTTON_PIN {5, 15, 25}
```

```
// 5, 15, 25, 26, 27, 32, 33, (34, 35, 36, 39)
#define NEW_ADDITIONAL_BUTTON_PIN {5, 15, 25}
```

```
#define NEW_ADDITIONAL_BUTTON_TYPE {INPUT_PULLUP, INPUT_PULLUP, INPUT_PULLUP}
```

```
#define NEW_ADDITIONAL_BUTTON_TYPE {INPUT_PULLUP, INPUT_PULLUP, INPUT_PULLUP}
```

## **config\_network.h**

### **meine Netzwerkeinstellungen**

```
const int maxSsids = 1;
const String ssids[maxSsids] = {"xxxxxxxxxxxxxxxxxxxxxx"};
const String passwords[maxSsids] = {"xxxxxxxxxxxxxxxxxxxxxx"};
const String routePrefixes[maxSsids] = {"IO:AUTO:"};
#define DEFAULT_IP_AND_PORT "19216817805302560"
```

```
// Geben Sie die SSIDs und Passwörter von so vielen Netzwerken ein, mit denen Sie versuchen möchten, eine Verbindung herzustellen.
const int maxSsids = 1;

// In jedem der folgenden Arrays muss die gleiche Anzahl von Entitäten (oben angegeben) vorhanden sein:

const String ssids[maxSsids] = {"xxxxxxxxxxxxxxxxxxxxxx"};
const String passwords[maxSsids] = {"xxxxxxxxxxxxxxxxxxxxxx"};

const String turnoutPrefixes[maxSsids] = {"NT"}; // erforderlich, wenn Sie Weichen verwenden möchten
// Dies ist das Präfix aller Weichensystemnamen für Ihr System, für die wiThrottle-Server in den oben genannten Netzwerken
// Hinweis: Bei DCC-EX EX-CommandStations sollte der entsprechende Wert leer sein (z.B. "")

const String routePrefixes[maxSsids] = {"IO:AUTO:"}; // erforderlich, wenn Sie Routen verwenden möchten
// Dies ist das Präfix aller Routensystemnamen für Ihr System, für die wiThrottle-Server in den oben genannten Netzwerken

// Wenn definiert, muss dies die Form 999999999999:99999 haben, z.B. "19216800400102560" entspricht 192.168.178.053:2560
// Jeder Block der IP-Adresse MUSS 3-stellig sein. Verwenden Sie bei Bedarf führende Nullen, um 3 Ziffern zu versichern.
// Der Port MUSS 5-stellig sein. Verwenden Sie bei Bedarf führende Nullen, um 5 Ziffern zu versichern.
#define DEFAULT_IP_AND_PORT "19216817805302560"
```

### **WLAN-Ländercode**

```
#define USE_WIFI_COUNTRY_CODE true
#define COUNTRY_CODE "01"
```

```
// "DE", "AT", "CH", "ES", "PT", "NL", "FR", "IT", "GR", "HU", "SI", "SK", "RO", "SE", "SI", "SK"
#define USE_WIFI_COUNTRY_CODE true
#define COUNTRY_CODE "01"
```

### **Automatische Verbindung zur ersten SSID**

```
#define AUTO_CONNECT_TO_FIRST_DEFINED_SERVER true
```

```
// Automatische Verbindung zur ersten SSID in der Liste
#define AUTO_CONNECT_TO_FIRST_DEFINED_SERVER true
```

### **Automatische Verbindung zum ersten gefundenen Server**

```
#define AUTO_CONNECT_TO_FIRST_WITHROTTLE_SERVER true
```

```
// Automatische Verbindung zum ersten gefundenen Server
#define AUTO_CONNECT_TO_FIRST_WITHROTTLE_SERVER true
```