

ساخت ماشین حساب با

RPi Pico

GitHub: [SAMSBSIREX/Rpi-pico-Calculator](https://github.com/SAMSBSIREX/Rpi-pico-Calculator): DLY Calculator which Rpi pico

قطعات:

کافه ربات | Pico قیمت و خرید برد رسیبری پای (رزبری پای) بیکو

I2C تک رنگ 0.91 اینچ ارتباط OLED قیمت و خرید مژول نمایشگر

کیپد | کافه ربات matrix قیمت و خرید کی پد فلت 4 در 4 ماتریسی

ابعاد 3.7 mm - 450mah - 30\*15\*6.8 V قیمت و خرید باتری لیتیوم یلیم تک سل

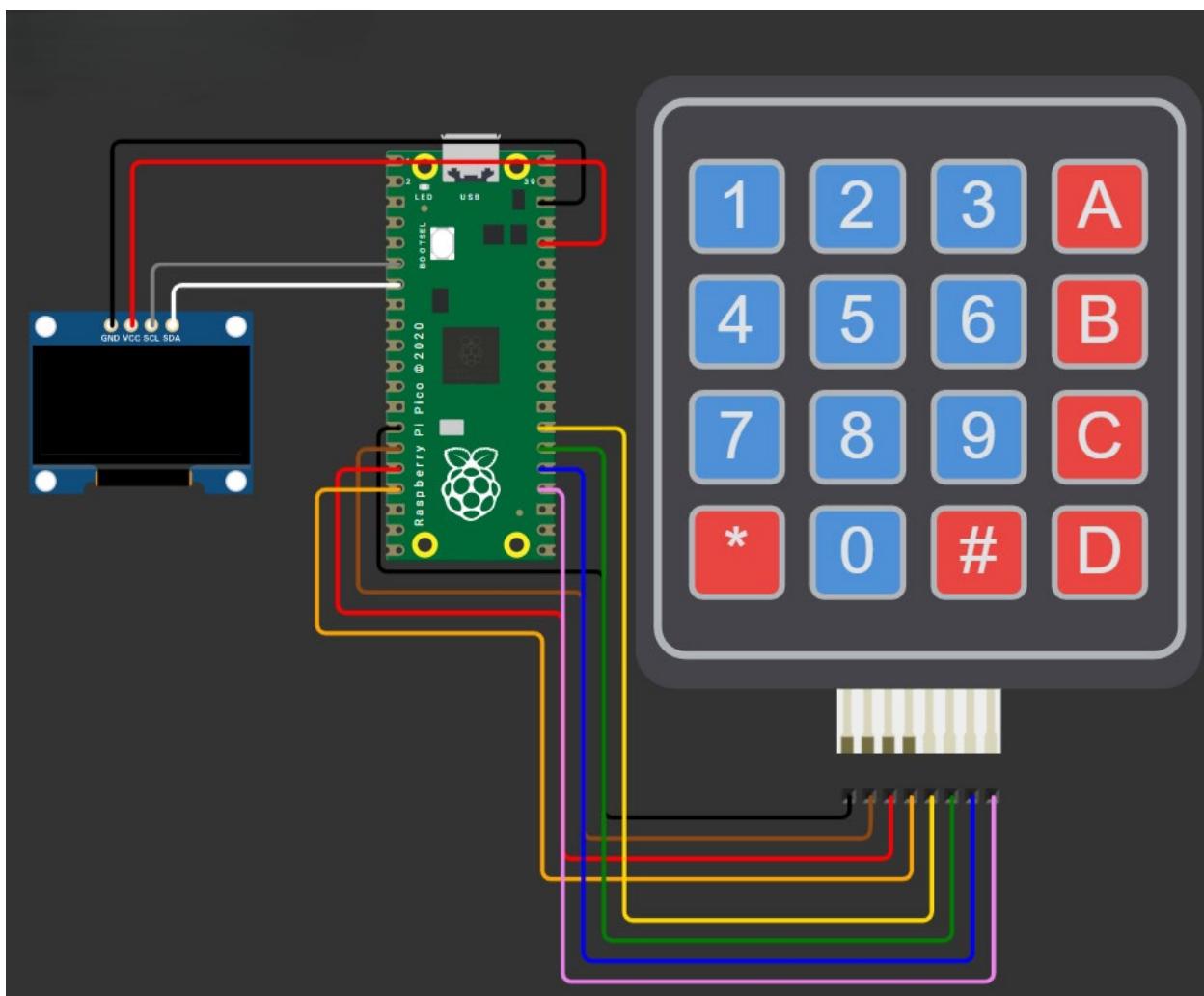
TP4056 قیمت و خرید مژول شارژر باتری لیتیومی 1 آمیر

قیمت و خرید برد سوراخ دار دولايه 3 در 7 سانتی متری | کافه ربات

مادگی - فاصله بین 2.54 میلی متر 40x40 قیمت و خرید بین هدر 1

نری - فاصله بین 2.54 میلی متر 40x40 قیمت و خرید بین هدر 1

شمایتیک اتصالات:



## کد پایتون:

```
#https://github.com/SAMSBSIREX/Rpi-pico-Calculator  
#Rpi-pico-Calculato
```

```
#Monitor >>> Rpi Pico  
#GND >>> 38 GND  
#VCC >>> 36 3V3(OUT)  
#SCK >>> 7 GP4  
#SDA >>> 6 GP5
```

```
#Keypad 4x4 Matrix >>> Rpi Pico  
#R1 >>> 14 GP10  
#R2 >>> 15 GP11  
#R3 >>> 16 GP12  
#R4 >>> 17 GP13  
#C1 >>> 27 GP21  
#C2 >>> 26 GP20  
#C3 >>> 25 GP19  
#C4 >>> 24 GP18
```

```
from machine import Pin, I2C  
import OLED  
import utime
```

```
i2c = I2C(0, scl=Pin(5), sda=Pin(4))  
oled = OLED.SSD1306_I2C(128, 32, i2c)
```

```
matrix_keys = [['1','2','3','A'],  
               ['4','5','6','B'],  
               ['7','8','9','C'],  
               ['*','0','#','D']]
```

```
rows = [10,11,12,13]  
cols = [21,20,19,18]
```

```
row_pins = [Pin(r, Pin.OUT) for r in rows]  
col_pins = [Pin(c, Pin.IN, Pin.PULL_DOWN) for c in cols]
```

```
oled.fill(0)  
oled.text("@SAMSBSIREX",1,1)  
oled.text("Press any Key",1,16)  
oled.show()  
utime.sleep(0.5)
```

```
def scankeys():  
    for i, row in enumerate(row_pins):
```

```
row.high()
for j, col in enumerate(col_pins):
    if col.value():
        key = matrix_keys[i][j]
        utime.sleep(0.3)
        row.low()
        return key
    row.low()
return None
```

```
expr = ""
```

```
while True:
    key = scankeys()
    if key:
        if key == "*":
            expr = ""
        elif key == "#":
            try:
                result = str(eval(expr))
                expr = result
            except:
                expr = "Error"
        elif key in "ABCD":
            if key == "A": expr += "+"
            if key == "B": expr += "-"
            if key == "C": expr += "*"
            if key == "D": expr += "/"
        else:
            expr += key
```

```
oled.fill(0)
oled.text(expr, 1, 16)
oled.show()
```

```
utime.sleep(0.1)
```

ویدیوی پروتایپ:

<https://drive.google.com/file/d/14QvRLHgqntjoH8li0oQVaMROSLNQgFqC/view?usp=sharing>

## اتصالات پرتوتاپ:

-دوتا پین هدر نری 20 تایی دو طرف برد پیکو لحیم کردم

-دوتا پین مادگی 20 تایی رو برد سوراخدار لحیم کردم

-یک پین هدر نری 4 تایی رو به مازول نمایش گر اولد وصل کردم

-به تمام خروجی های مازول شارژر پین هدر نری وصل کردم

-باتری لیتیوم پلیمری رو طبق پلاریته به ورودی باتری مازول شارژ وصل کردم

-یک پین هدر نری 8 تایی به برد سوراخدار وصل کردم

-برد پیکو رو به پین هدر مادگی روی برد سوراخ دار وصل کردم

-مازول نمایشگر اولد رو به برد سوراخدار وصل کردم

-مازول شارژر رو روی برد سوراخدار لحیم کردم

-کیپ 4 در 4 ماتریسی رو به پین هدر نری 8 تایی وصل کردم

-خروجی + مازول شارژر رو به کلید روش خاموش و کلید رو به وی سیس برد پیکو و

-خروجی - رو به زمین برد پیکو وصل کردم

-بقیه خروجی هارو با کابل فلت به ورودی ها و برعکس وصل کردم (طبق کامنت هایی اول کد پایتون گذاشت)

با برنامه ای ام یو ادیتور رو با نام مین روی برد اپلود کردم و همچنین کتاب خانه ای اولد رو از گیت هاب خودم

دانلود و روی برد طبق همون روش قبلی روی برد اپلود کردم

-حالانمونهی اولیه آمدهس

## نحوه کار:

این برنامه یک حلقة اصلی دارد که به صورت پیوسته منتظر فشردن کلید از کیپ ماتریسی است. برای خواندن کیپ، هر سطر به طور متناظر فعال می‌شود و ستون‌ها خوانده می‌شوند؛ اگر در تقاطع سطر فعال و یک ستون، اتصال برقرار باشد. یعنی کلیدی فشرده شده است و برنامه نماد آن کلید را از ماتریس نگاشت کلیدها می‌گیرد.

پس از تشخیص کلید، برنامه آن را به رشته‌ای که عبارت فعلی را نگه می‌دارد اضافه می‌کند یا بر اساس نوع کلید رفتار مناسب را انجام می‌دهد: کلید پاک کردن رشته را خالی می‌کند، کلید محاسبه تلاش می‌کند عبارت را محاسبه کند و نتیجه یا پیام خط را جایگزین عبارت می‌کند، و کلیدهای ویژه به نمادهای عملیاتی تبدیل می‌شوند (جمع، تفریق، ضرب، تقسیم، توان، باقی مانده).

برای جلوگیری از خواندن‌های ناخواسته هنگام فشردن، پس از شناسایی کلید یک مکث کوتاه انجام می‌شود تا از ثبت چندباره یک فشردن جلوگیری شود. پس از هر تغییر در عبارت، نمایشگر پاک می‌شود و متن عبارت یا نتیجه در موقعیت تعیین شده نمایش داده می‌شود.

محاسبه عبارت با استفاده از یک تابع ارزیابی رشته‌ای انجام می‌شود؛ اگر عبارت نادرست یا موجب خطا (مثل تقسیم بر صفر) شود، برنامه آن را گرفته و بهجای نتیجه عبارت "خطا" را نمایش می‌دهد. به همین دلیل نگارش و نحوه وارد کردن عملگرهای و اعداد باید صحیح باشد

سیم‌بندی سخت‌افزاری بهصورتی است که سطرها به پایه‌های خروجی و ستون‌ها به پایه‌های ورودی با حالت کشش-پایین متصل شده‌اند؛ وقتی سطربالا می‌رود و کلیدی فشار داده شده، پایه ستون متناظر مقدار منطقی یک می‌گیرد و این حالت توسط برنامه تشخیص داده می‌شود

در مجموع، جریان کار به این ترتیب است: اسکن کلید >>> شناسایی کلید >>> بهروزرسانی عبارت یا انجام عملیات مخصوص >>> نمایش عبارت یا نتیجه >>> تکرار

By @samsbsirex (Seyed Amir Mohammad Seyed Babaei)