

ETC22 – jaro 2024

RGB LAMPIČKA OVLÁDANÁ PŘES WIFI + ZVUK

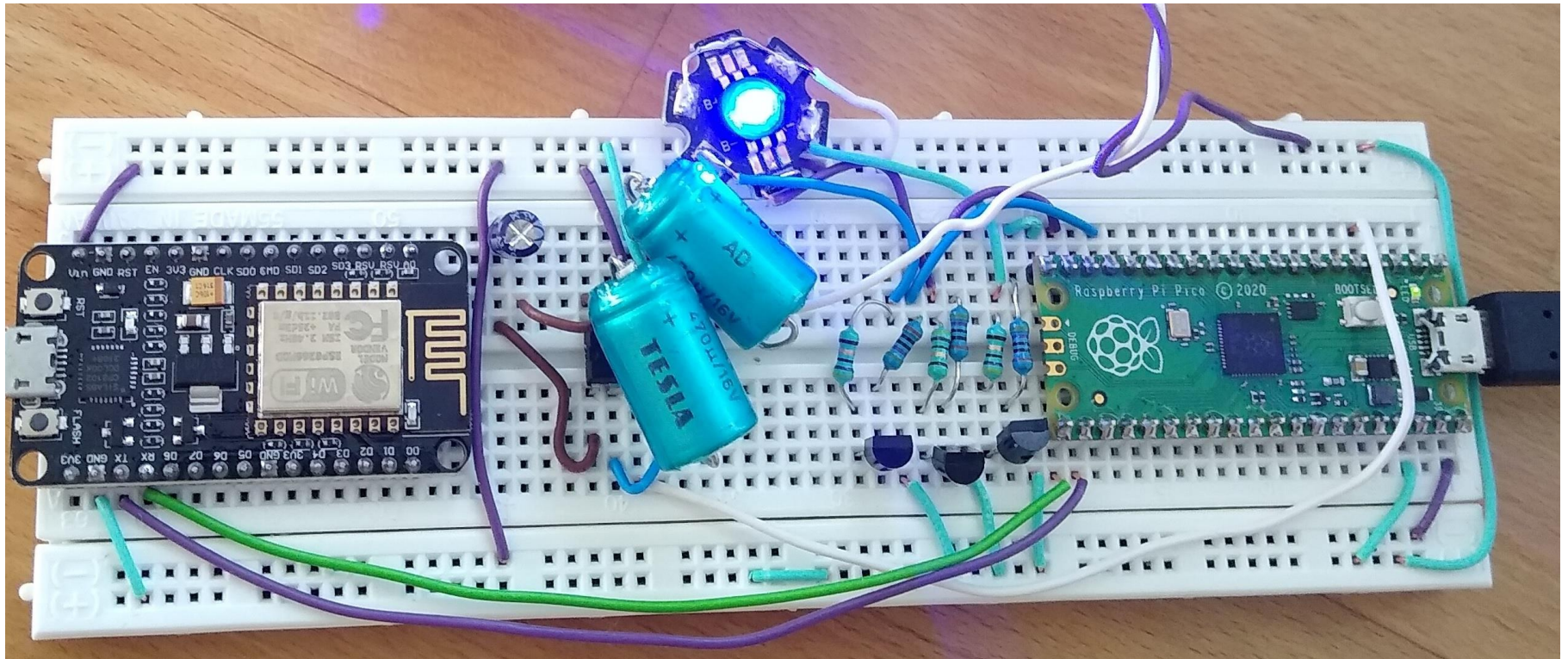
ČVUT v Praze, FEL, Katedra měření

Provedení na nepájivém poli

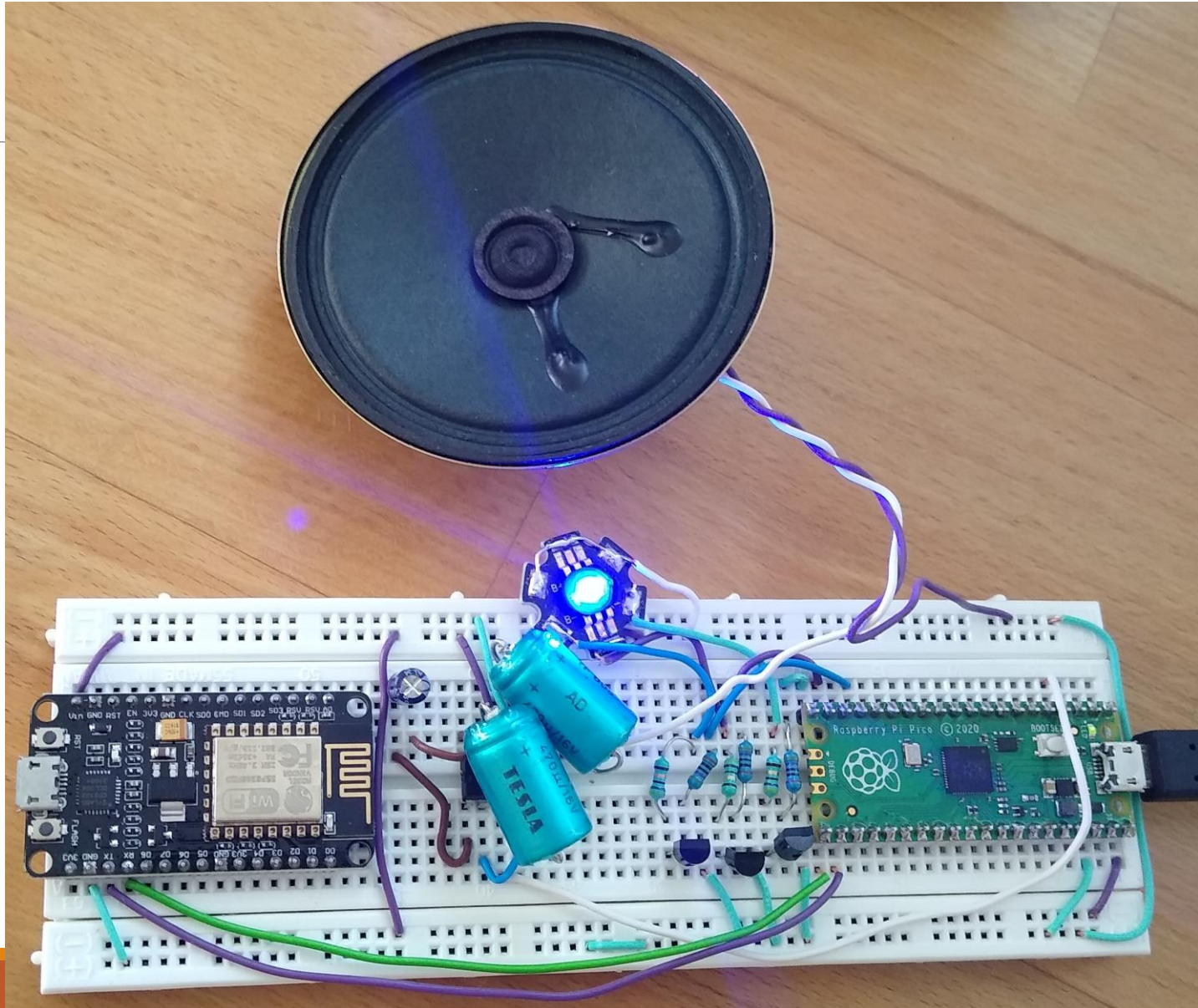
ESP8266 modul + Raspberry PICO modul

Světlo: + RGB LED + 3xNPN tranzistor

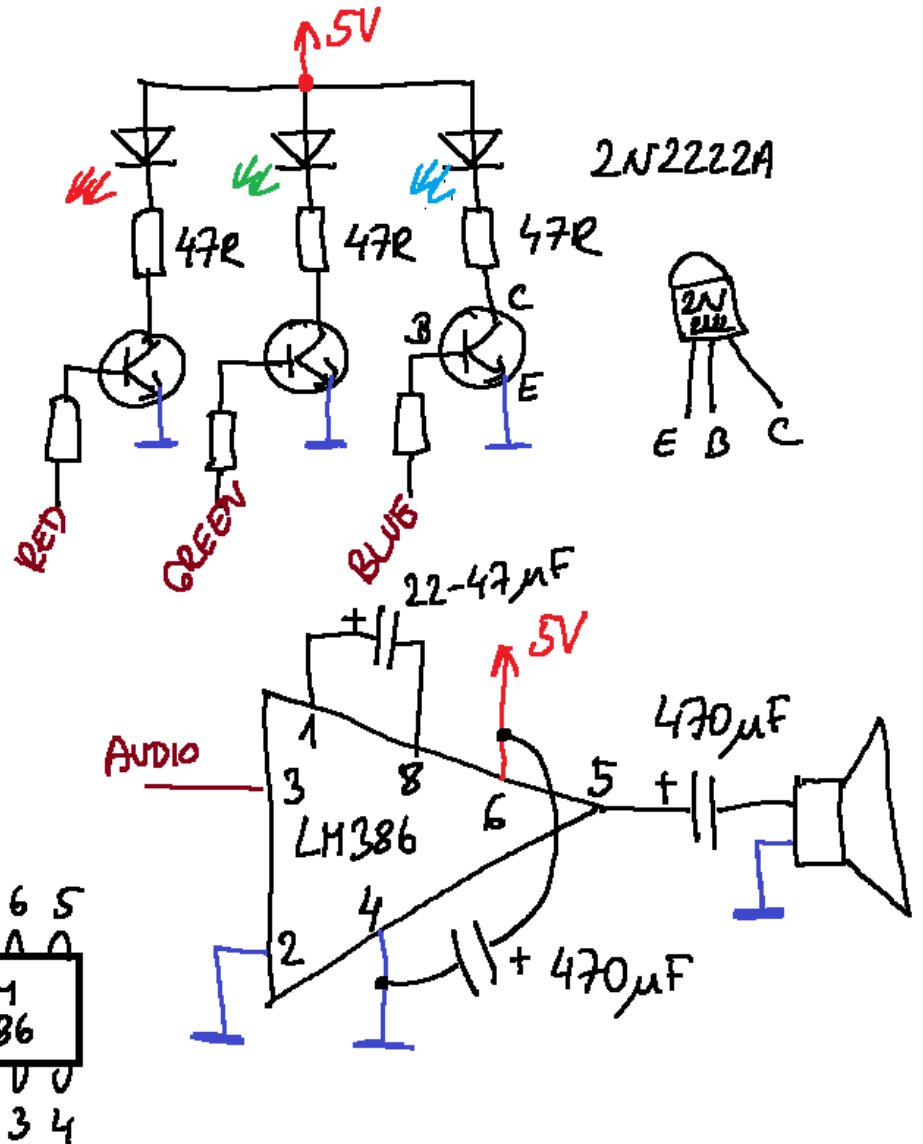
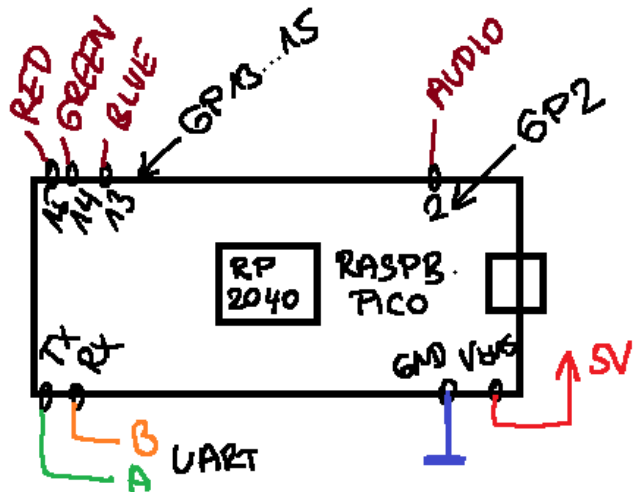
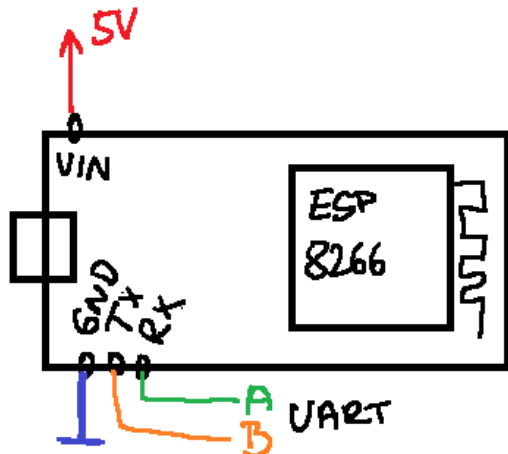
Zvuk: + LM386 zesilovač + reproduktor



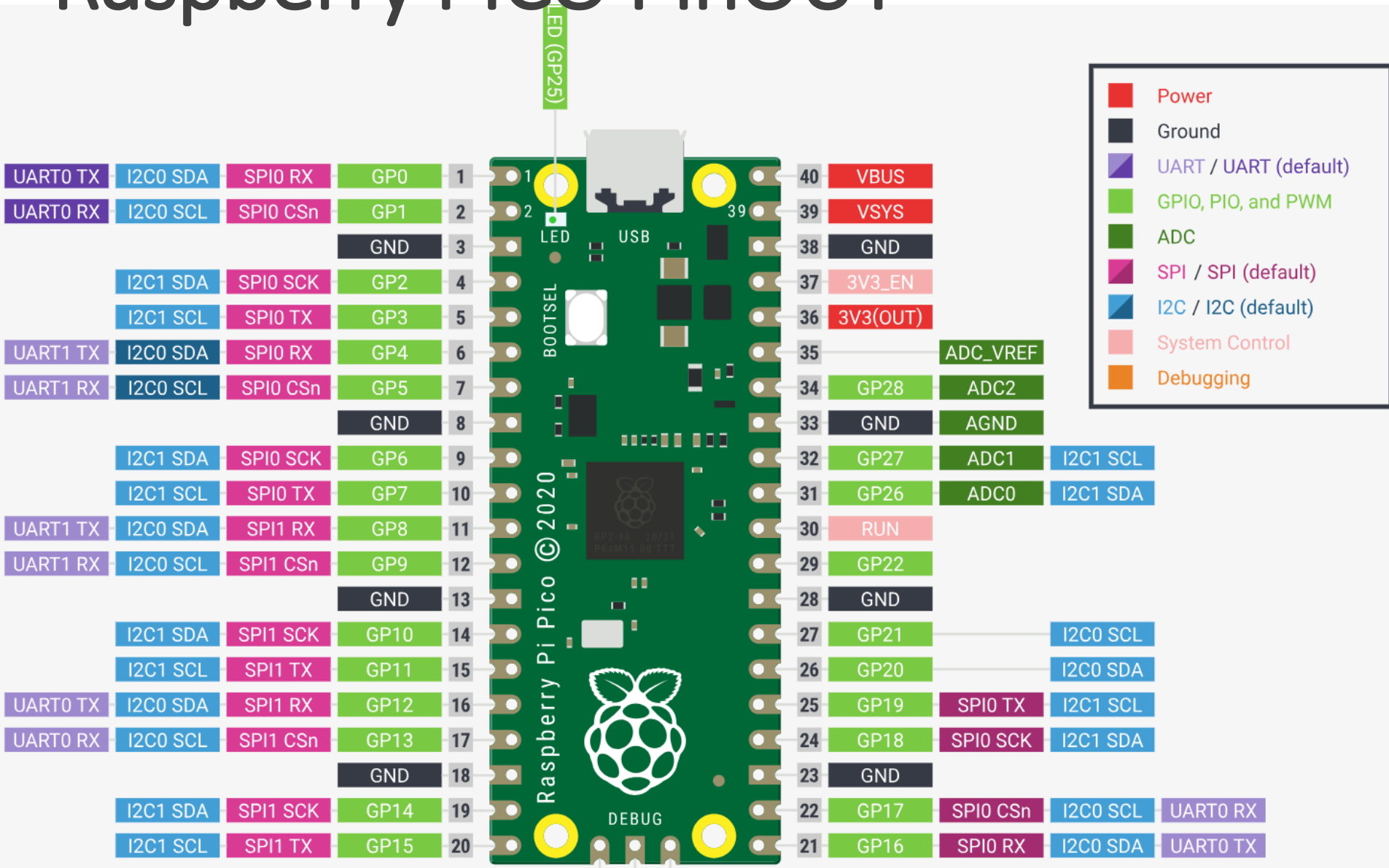
i s reproduktorem...



Zapojení

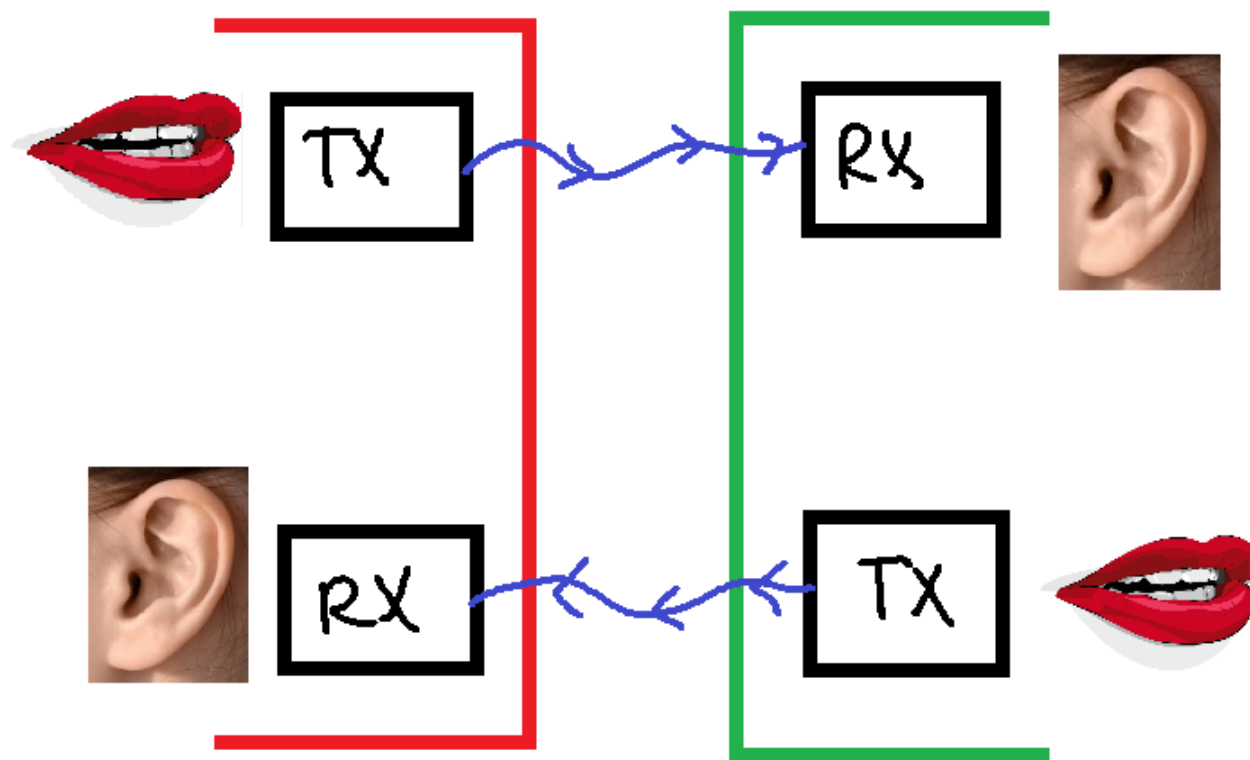


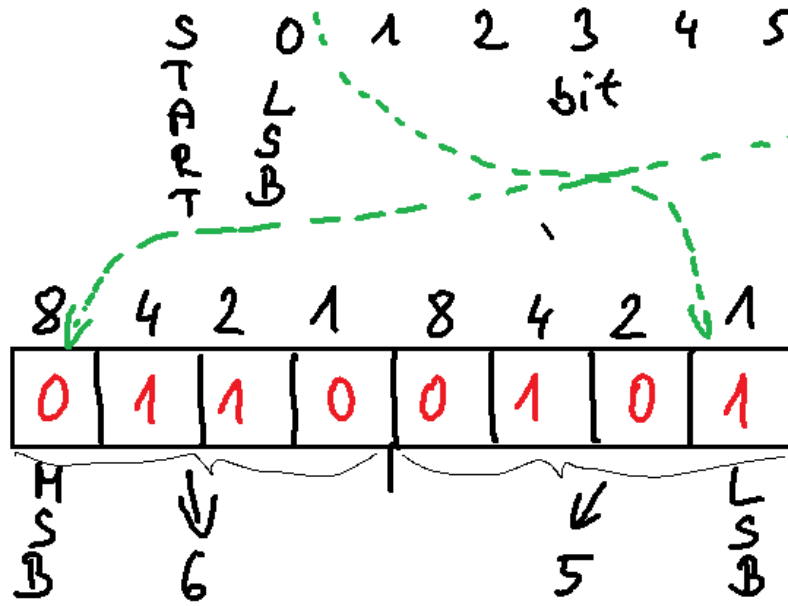
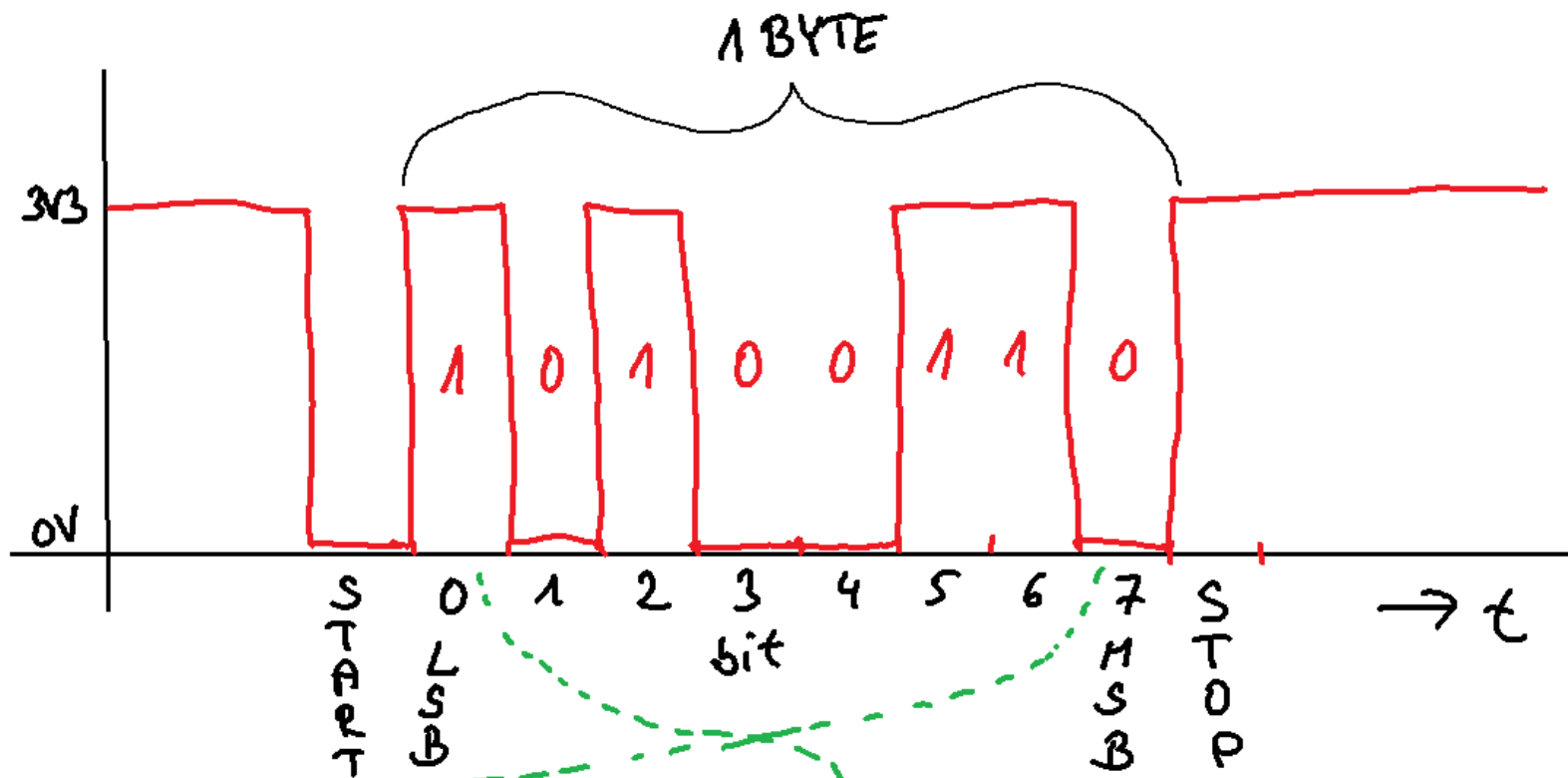
Raspberry PICO PinOUT



Jak to funguje? UART

Univerzální asynchronní přijímač a vysílač





⇒ 0x65 HEX
 101 = 6 × 16 + 5 DEC
 ⇒ znak „e“ (malé E)

Mapa znaků – 61HEX, 97DEC

Mapa znaků

Písmo:

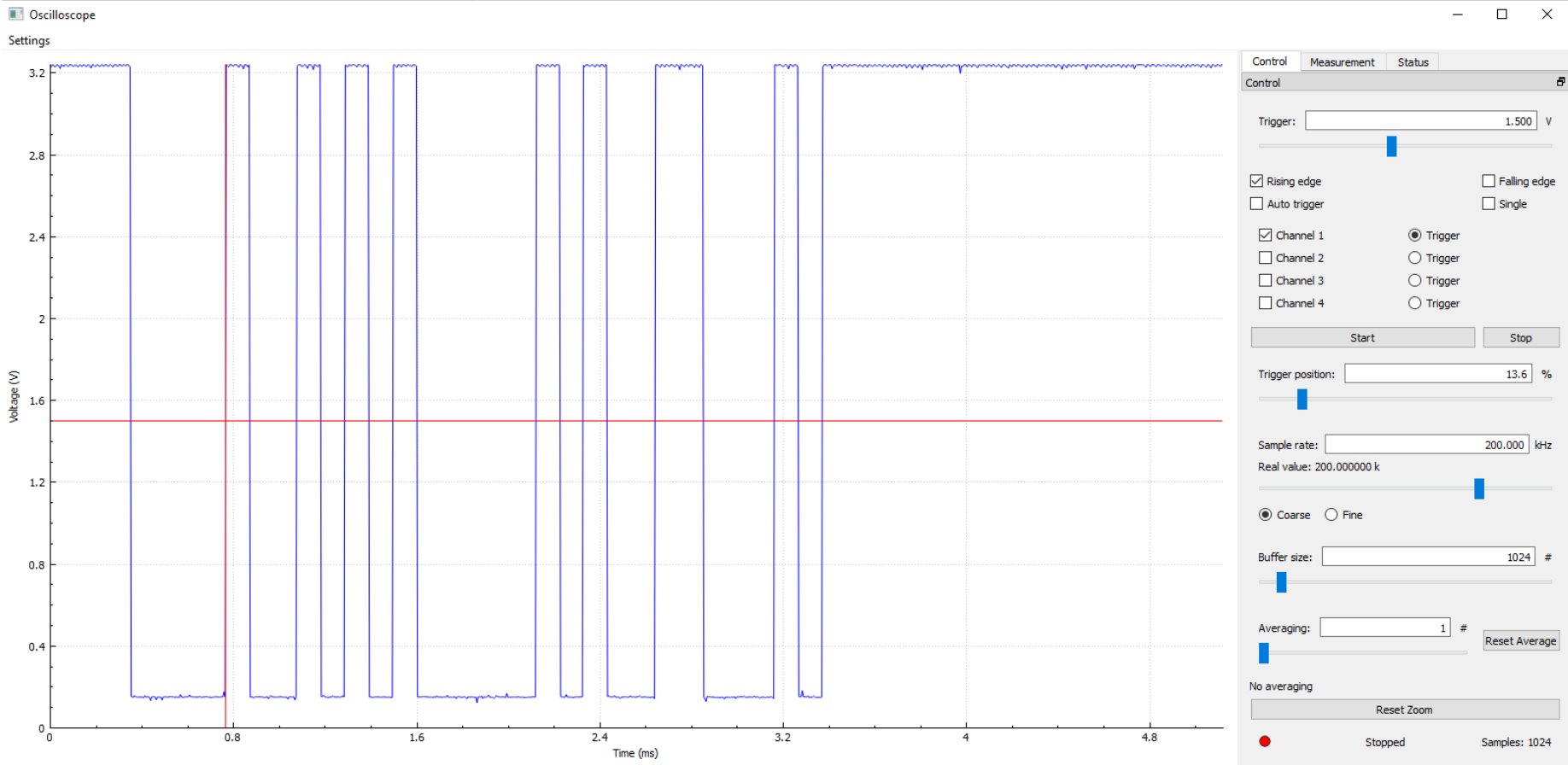
!	"	#	\$	%	&	'	()	*	+	,	-	.	/	0	1	2	3	4	^
5	6	7	8	9	:	;	<	=	>	?	@	A	B	C	D	E	F	G	H	
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	[\	
]	^	_	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	
q	r	s	t	u	v	w	x	y	z	{		}	~	ı	ç	£	¤	¥		
ı	§	¨	©	a	U+0061: Malé písmeno latinky a	´	µ	¶	·	,	´									
°	»	¼	½	¾	¿	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	
Î	Ï	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß	à	á	
â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï	ð	ñ	ò	ó	ô	õ	
ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ	Ā	ā	Ă	ă	Ą	ą	Ć	ć	Ĉ	ĉ	

Znaky ke zkopírování:

Podrobné zobrazení

U+0061: Malé písmeno latinky a

???



Ještě trocha elektroniky

pokud zapnete lampičku na mód RED, plný výkon...

- 1) jaký proud prochází LED
- 2) jaký je výkon na LED a sériovém rezistoru
- 3) kolik energie spotřebuje toto světélko za 1 rok
- 4) kolik je to Kč pokud 1 kWh stojí 8 Kč ?

na R 47R je napětí 2.99V

Přehráání zvuku pomocí Raspberry Pi PICO

- zvuk ve formátu WAV 44100 Hz sampling, mono stačí, 16-bit (lze použít například program Audacity - free)
- pomocí Thonny nahrát WAV soubory do paměti PICO (adresář sounds)
- mít na PICO tyto soubory:
wavePlayer.py, wave.py, myPWM.py, myDMA.py, chunk.py viz audio.zip

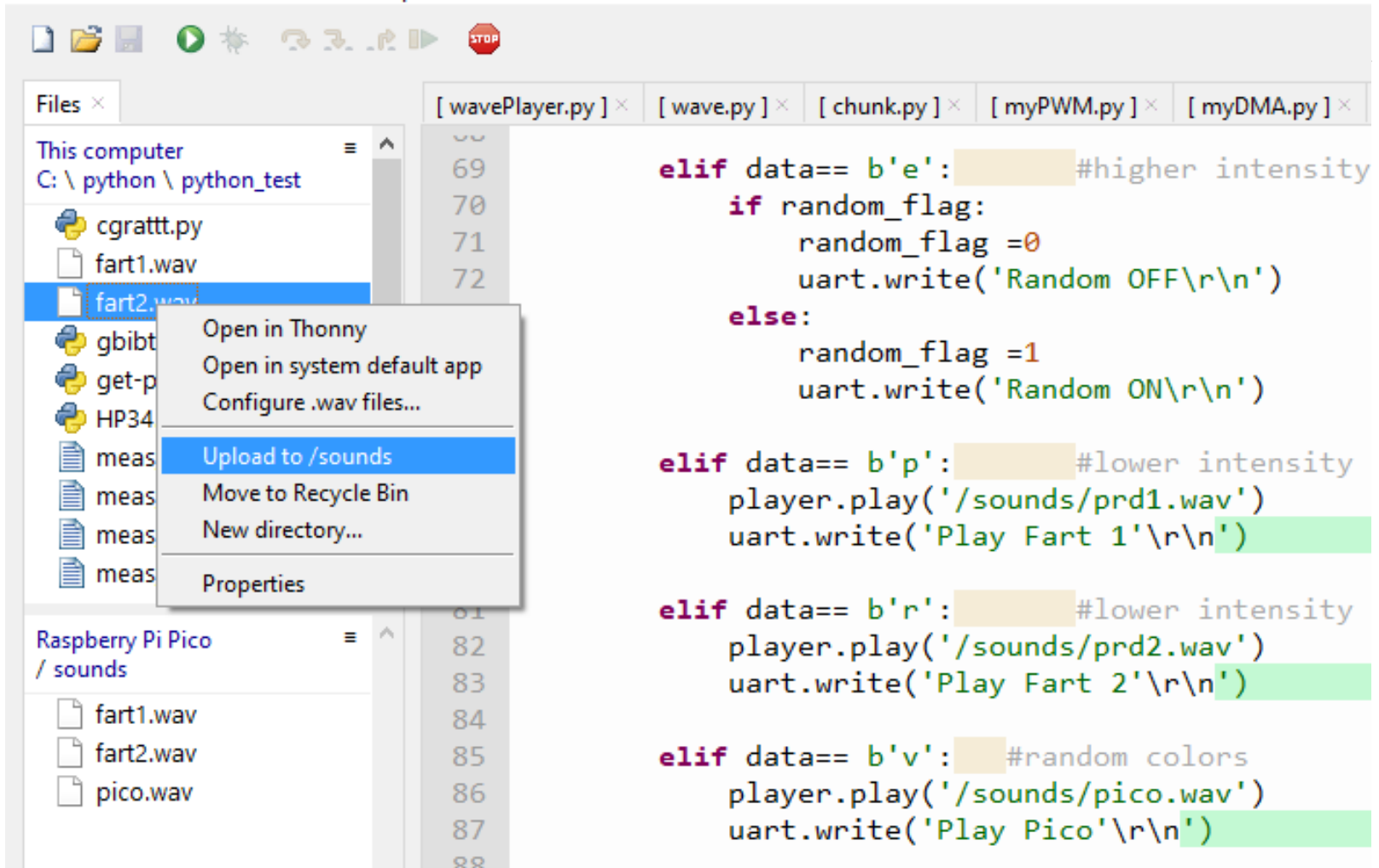
no a pak to přehrajeme takto:

```
import wave
import os as uos
from wavePlayer import wavePlayer
player = wavePlayer()
player.play('/sounds/pico.wav')
```

Thonny – nahrání souborů

Thonny - Raspberry Pi Pico :: /main_copy.py @ 10 : 1

File Edit View Run Tools Help



The screenshot shows the Thonny IDE interface. On the left, a file explorer is open, displaying the contents of the 'C:\python\python_test' directory. The files listed are 'cgrattp.py', 'fart1.wav', 'fart2.wav', 'gbibt', 'get-p', 'HP34', 'meas', 'meas', 'meas', and 'meas'. The 'fart2.wav' file is selected, and a context menu is open over it, showing options: 'Open in Thonny', 'Open in system default app', 'Configure .wav files...', 'Upload to /sounds', 'Move to Recycle Bin', 'New directory...', and 'Properties'. The 'Upload to /sounds' option is highlighted. Below the file explorer, the 'Raspberry Pi Pico / sounds' directory is visible, containing 'fart1.wav', 'fart2.wav', and 'pico.wav'. On the right, a code editor shows Python code with several lines highlighted in green. The code is as follows:

```
elif data== b'e': #higher intensity
    if random_flag:
        random_flag =0
        uart.write('Random OFF\r\n')
    else:
        random_flag =1
        uart.write('Random ON\r\n')

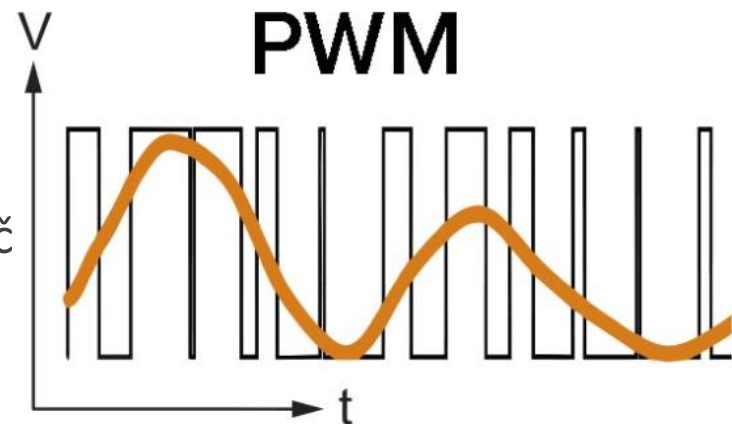
elif data== b'p': #lower intensity
    player.play('/sounds/prd1.wav')
    uart.write('Play Fart 1'\r\n')

elif data== b'r': #lower intensity
    player.play('/sounds/prd2.wav')
    uart.write('Play Fart 2'\r\n')

elif data== b'v': #random colors
    player.play('/sounds/pico.wav')
    uart.write('Play Pico'\r\n')
```

Jak to funguje?

Digitálně analogový převodník (PWM) + zesilovač

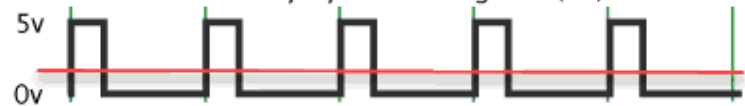


Pulse Width Modulation

0% Duty Cycle - analogWrite(0)



25% Duty Cycle - analogWrite(64)



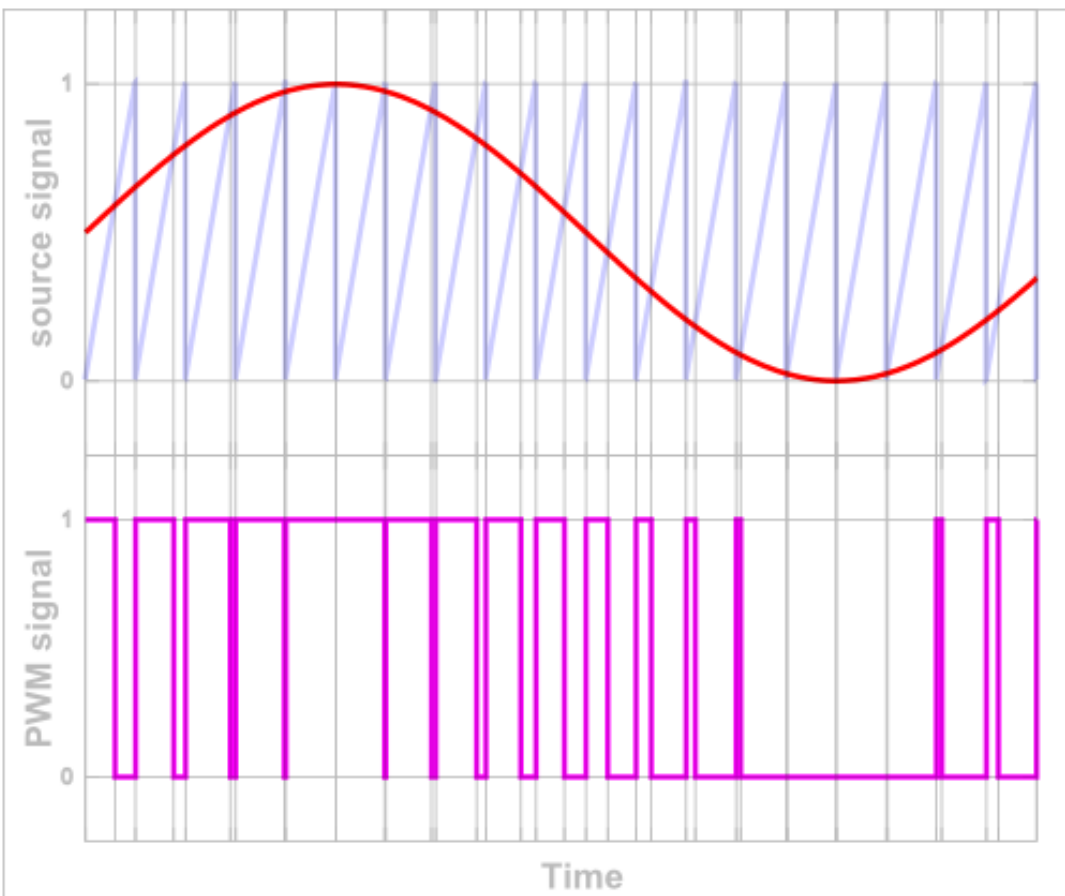
50% Duty Cycle - analogWrite(127)



75% Duty Cycle - analogWrite(191)

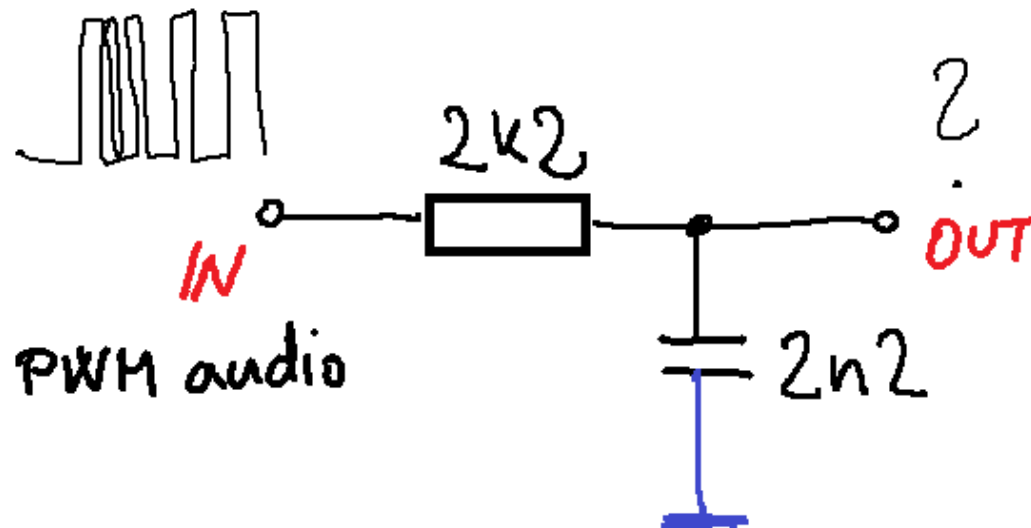


100% Duty Cycle - analogWrite(255)

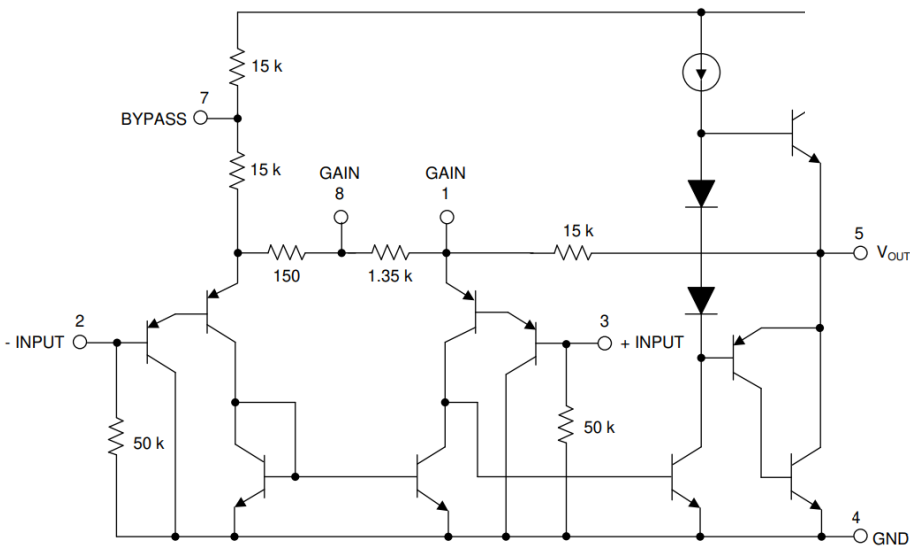
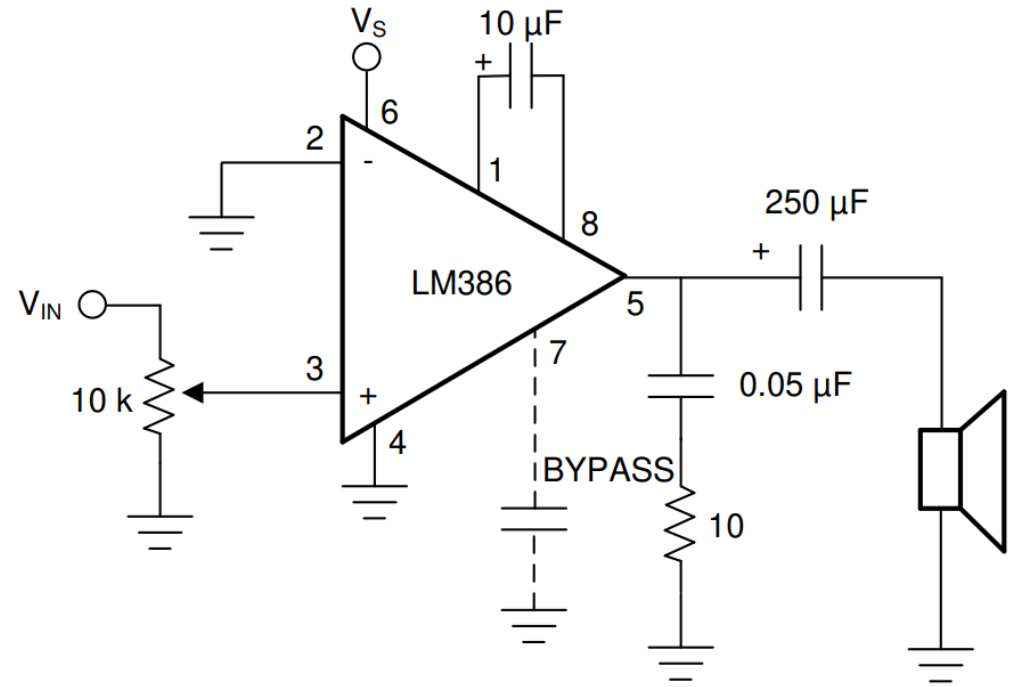


PWM na osciloskopu

- zkuste si zobrazit PWM průběh při přehrávání zvuku na výtupu „audio“
jednak přímo a také pomocí filtru typu dolní propust – RC článek 2k2 a
2n2



Zesilovač LM386



ESP8266 - ESPterm

<https://github.com/espterm/espterm-firmware>

ESPterm

```
ETC22 RGB lamp
ETC22 RGB lamp
Color 1
Color 0
Light 2
Light 1
Light 0
█
```

C+

C-

I+

I-

RND

Fullscreen Upload Config WiFi Help About

WiFi Settings

Built-in Access Point

✓ Apply!

Enabled

AP SSID RGB_LED

Password

Channel 1

Transmit power +5.00 dBm

Hide SSID

Nutno nastaviti – 115200 bd

Serial Port Parameters

✓ Apply!

This form controls the communication UART. The debug UART is fixed at 115.200 baud, one stop-bit and no parity.

Baud rate bps

Parity

Stop-bits

Nastavení tlačítek

Initial Settings

✓ Apply!

Those are the initial settings used after ESPTerm powers on, or when the screen reset command is received (`\ec`). They can be changed by the terminal application using escape sequences.

Width

26

Height

10

Header Text

ESPTerm

Show buttons



Button count

5

Button Labels

C+

C-

I+

I-

RND

Button codes
(ASCII, dec, CSV)

97

98

99

100

101

Button colors

0

0

0

0

0

Background image URL

Raspberry PICO – programování v Thonny

- použít nejnovější MicroPython

https://micropython.org/download/RPI_PICO/

.uf2 soubor nahrát do Raspberry PICO – jako na FLASHku - když se připojí do PC se stisknutým bílým tlačítkem BOOT na zeleném modulu Raspberry PICO

Program v MicroPython

```
from machine import Pin,UART, PWM

import time

from time import sleep

import random

uart = UART(1, baudrate=115200, tx=Pin(4), rx=Pin(5))

uart.init(bits=8, parity=None, stop=1)

led = Pin("LED", Pin.OUT)

ledR = machine.Pin(0)

ledG = machine.Pin(1)

ledB = machine.Pin(2)
```

.... viz externí soubor , uložit jako RGB_SOUND.py do Raspberry PICO,
pokud byste chtěli, aby se spustilo hned po zapnutí (bez PC, třeba při napájení
z powerbanky, přejmenovat soubor na main.py), pozor pak trochu problematické na změnu kódu...