:

: R4136

1. Python.

```
import numpy as np
import matplotlib.pyplot as plt
import os
import random
import scipy
import torch

torch.cuda.synchronize()
torch.cuda.empty_cache()

cuda = torch.device('cuda')
print(torch.cuda.get_device_properties(cuda))
```

```
_CudaDeviceProperties(name='NVIDIA GeForce RTX 3080 Laptop
```

2. .

```
name = random.choice(os.listdir("dataset"))

# name = 'testLab1Var7.csv'

print(f"Dataset: {name}")

dataset = np.genfromtxt(f"dataset/{name}", delimiter=',')

dataset = [dataset[:, i] for i in range(dataset.shape[1])]
title = ["time", "current", "voltage"]

dataset_dict = dict(zip(title, dataset))
```

```
Dataset: testLab1Var11.csv
```

3. .

,

, .

```
"""

"""
time_period = 0.1

time_interval = random.random() * (dataset_dict["time"][-1]
time_interval = (time_interval, time_interval + time_perio

print(f"              {time_interval}")
```

```
            (28.65413982391623, 28.75413982391623)
```

```
plt.plot(dataset_dict["time"], dataset_dict["current"])
plt.xlim(time_interval)
plt.xlabel('   ,  ')
plt.ylabel('      ,  ')
plt.legend(["            "])
```

4. L R.

.

:

$$\begin{cases} u = e + R \times i + L \times \dfrac{di}{t} \\ M - M_C = J\dfrac{d\omega}{t} \\ M = C_M \times \Phi \times i \\ e = C_\omega \times \Phi \times \omega \end{cases}$$

u - ,

e - ( ) ,

i - ,

Φ - , ,

M - ,

5

Y $X \times K$,

$$e = Y - X \times K$$

:

$$S(K) = \sum e_i^2 = e^T \times e = (Y - X \times K)^T \times (Y - X \times K)$$

:

$$\sigma_Y = \sqrt{\frac{S(K)}{n}}$$

```
e2_Y = torch.mm(Y_tensor.T - torch.mm(X_tensor, K_approx).T
sigma2_Y = torch.divide(e2_Y, Y_tensor.shape[0])

sigma_Y = torch.sqrt(sigma2_Y)
sigma_Y = sigma_Y.cpu().numpy()[0][0]
```