

OBTENÇÃO DE CURVAS CORRENTE – VOLTAGEM USANDO UNIDADES MICROCONTROLADAS (MCUs) E A LINGUAGEM PYTHON

OBTAINING CURRENT – VOLTAGE CURVES USING MICROCONTROLLED UNITS (MCUs) AND THE PYTHON LANGUAGE

Derek Dorbação de Araujo¹, Matheus Cotta da Silva², Luis Felipe Viana Bernardes³, Taina de Azevedo Silva⁴, Miguel de Andrade Garcia⁵, Marcelo Azevedo Neves⁶

¹ UFRRJ/PET Física, derekdoracao@gmail.com

² UFRRJ/PET Física, cottafis@gmail.com

³ UFRRJ/PET Física, lfelipevbernardes.ufrj@gmail.com

⁴ UFRRJ/PET Física, taina.azevedo@ufrj.br

⁵ UFRRJ/PET Física, miguelgarcia05799@gmail.com

⁶ UFRRJ/Departamento de Física & PET Física, mneves@ufrj.br

Resumo

Este trabalho apresenta uma pesquisa em instrumentação para o ensino da Física realizada com objetivo de aplicar o uso de duas diferentes unidades microcontroladas (ou MCUs), ambas programadas usando a linguagem de alto nível Python, para obtenção de curvas corrente *versus* voltagem de dispositivos eletrônicos. Estas específicas tecnologias de informação e comunicação foram escolhidas para incluir um aspecto motivador: por serem de fácil aprendizado elas próprias podem se tornar objeto de novos estudos e aprendizagem pelos educandos. Especificamente, a pesquisa consistiu em estudar o uso de uma nova MCU, a Raspberry Pi Pico, e avaliar se seus resultados diferem dos obtidos usando uma MCU de uso já disseminado, porém de custo financeiro relativo maior, a Arduino UNO R3. Os resultados mostram que o uso da MCU Raspberry Pi Pico fornece resultados compatíveis com o uso da MCU Arduino UNO R3 e assim se propõe também o uso desta nova MCU em outros projetos de instrumentação para o ensino da Física.

Palavras-chave: curvas I-V, TIC, Arduino, Raspberry Pi Pico.

Abstract

This work presents research into instrumentation for Physics teaching carried out with the aim of applying the use of two different microcontrolled units (or MCUs), both programmed using Python, a high-level language, to obtain current versus voltage curves for electronic devices. These specific information and communication technologies were chosen to include a motivating aspect: because they are easy to learn, they themselves can become the object of new studies and learning by students. Specifically, the research consisted of studying the use of a new MCU, the

Raspberry Pi Pico, and evaluating whether its results differ from those obtained using an MCU already in widespread use, but with a higher relative financial cost, the Arduino UNO R3. The results show that the use of the Raspberry Pi Pico MCU provides results compatible with the use of the Arduino UNO R3 MCU and thus it is also proposed to use this new MCU in other instrumentation projects for teaching Physics.

Keywords: I-V curves, ICT, Arduino, Raspberry Pi Pico.

Introdução

As curvas características corrente *versus* voltagem (doravante simplesmente referidas por curvas I-V) são importantes para o estudo de diversos dispositivos eletrônicos (CAPUANO, 2009), como os resistores (isoladamente, associados em série, associados em paralelo), lâmpadas, diodos (retificadores, de efeito Zener, emissores de luz), transistores (em modo emissor comum, em modo base comum, em modo coletor comum), entre outros. Acessando os valores nas curvas I-V pode-se obter valores de grandezas físicas relevantes. A obtenção de curvas I-V pode ser feita usando tecnologias de informação e comunicação (TICs) modernas, como propomos neste trabalho, que apresenta uma pesquisa em instrumentação para o ensino da física em que duas unidades microcontroladas (ou MCUs) diferentes, porém ambas programadas usando a linguagem Python, foram usadas para obtenção de curvas I-V. Um dos objetos de estudo foi uma relativamente nova MCU, a Raspberry Pi Pico (RASPBERRY, 2024), que tem custo de cerca de 50% da MCU Arduino UNO R3 (ARDUINO, 2024) no mercado nacional. A Arduino UNO R3 já tem sido amplamente usada na pesquisa em instrumentação para ensino usando TICs (MOREIRA, 2018; MONTEIRO, 2022; OLIVEIRA, 2020), e assim buscamos construir duas plataformas para obtenção de curvas I-V: uma usando a MCU Raspberry Pi Pico e a outra usando a MCU Arduino UNO R3. Outra meta é que os resultados desta pesquisa permitam o uso das plataformas de instrumentação desenvolvidas em aulas laboratoriais de Física no Ensino Médio, no Ensino Técnico e mesmo no Ensino Superior.

Fundamentação

A MCU Arduino UNO R3 tem como componente principal o microcontrolador (μC) ATMEL ATMEGA328, um circuito integrado de 8 bits operando em 16 MHz. Esta MCU possui 28 pinos, sendo que destes efetivamente para comunicação: 6 são

para sinais analógicos de 0V a 5V, com resolução de 10 bits, 2 pinos para comunicação serial (UART), 6 para 3 protocolos de comunicação (UART, I2C, SPI). A utilização da Arduino UNO R3 usualmente requer uma linguagem de baixo nível derivada de C++.

Já a MCU Raspberry Pi Pico é uma placa microcontroladora de baixo custo e alto desempenho com interfaces digitais flexíveis, onde o μ C é o RP2040, desenvolvido pela Raspberry Pi Foundation (RASPBERRY, 2023). Esta MCU traz muitas vantagens em relação à Arduino UNO R3, entre elas 26 pinos GPIO ("entradas e saídas de uso geral") multifuncionais, com: 2 pinos de comunicação SPI, 2 pinos de comunicação I2C, 2 pinos de comunicação UART, 3 pinos de ADC ("conversão analógica-digital) de 12 bits. Seu custo é cerca de 50% de uma Arduino UNO, mas ela opera apenas com 3,3 volts de tensão de entrada e saída em seus pinos.

Uma mesma linguagem foi selecionada para programação de ambas MCUs: a linguagem Python em sua versão 3 (PYTHON, 2023), que tem sido amplamente usada em diversas aplicações, além do desenvolvimento de softwares, pode ser baixada gratuitamente (PYTHON, 2023) e possui diversos ambientes de desenvolvimento. É uma linguagem de alto nível, com comandos e sintaxe básica semelhantes a um texto em inglês e um programa escrito em Python tem alta portabilidade, i.e., pode ser facilmente adaptado para rodar em diferentes sistemas operacionais (Windows, macOS, Linux e Unix). Desta forma, integrar Python com Arduino UNO R3 e com Raspberry Pi Pico permite obter uma plataforma versátil. De fato, existem bibliotecas para Python como a PyFirmata (PYFIRMATA, 2023) e a Pyserial (PYSERIAL, 2023) que permitem estabelecer comunicação serial entre o computador (*desktop* ou *laptop*) em que roda o programa em Python e as MCUs.

Metodologia

Foram selecionados os seguintes dispositivos: dois resistores com valores nominais de 200Ω (com tolerância de 5%) a serem caracterizados individualmente, associados em série e associados em paralelo; um diodo retificador 1N4001; e um diodo Zener C2V75T.

Utilizamos um computador pessoal (PC) como motor para o processamento, separando a função de captação de dados para a MCU, e a de processamento de dados para o PC. O PC também iniciava o comando da MCU para a aquisição dos sinais desejados.

A geração do sinal de tensão elétrica a ser aplicada aos dispositivos é realizada usando um módulo conversor de sinal digital para analógico (DAC) com o chip MCP4725 da Microchip Technology [COMPONENTS101, 2023], que se comunica tanto com a MCU Raspberry Pi Pico como com a Arduino UNO R3 pelo respectivo barramento I2C [HALFACREE, 2021] de cada MCU. Para uma mesma varredura de medida efetivamente gerar uma tensão elétrica negativa, utilizamos uma pilha de 1,5 V polaridade inversa à gerada pelo módulo MCP42765. Para obter medidas do Efeito Zener foram utilizadas duas pilhas de 1,5 V em série, pois a tensão Zener nominal deste diodo é de 2,7 V, e a tensão de início do efeito Zener é 2,0 V. Como transdutor de corrente, se usou um resistor de $220\ \Omega$ (valor medido com ohmímetro) ligado em série com o dispositivo eletrônico em estudo.

A Figura 01 mostra imagens representativas das montagens realizadas com cada MCU, Arduino UNO R3 e Raspberry Pi Pico, mas com um resistor sendo o dispositivo eletrônico a ser estudado. Note-se que no caso do diodo retificador, do diodo Zener e das associações em série e em paralelo de resistores, se substitui o resistor na Figura 01 pelos respectivos arranjos de interesse. Todas as conexões foram realizadas em uma matriz de contato com fios rígidos com terminais DuPont do tipo macho.

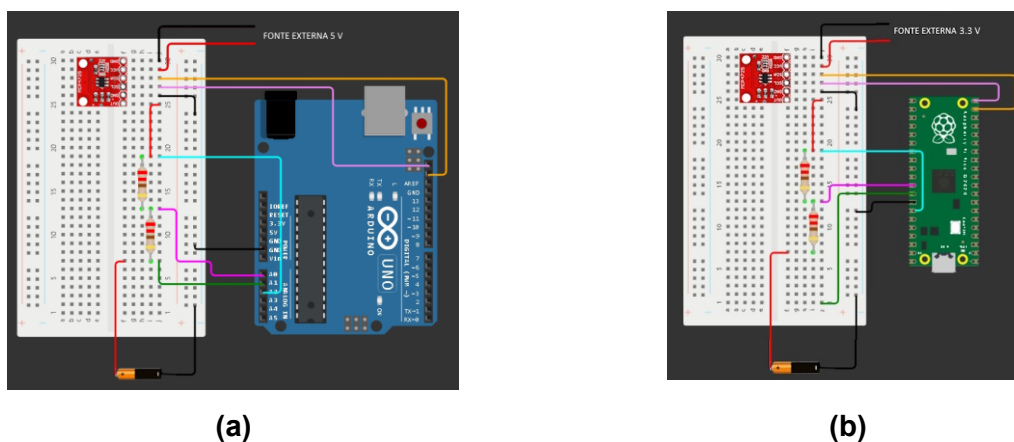


Figura 01: Esquema das montagens (a) para Arduino UNO R3, (b) para Raspberry Pi Pico.

O software desenvolvido em Python para rodar no computador pessoal (PC) utiliza a biblioteca “*pyserial*” (PYSERIAL, 2023). Para a comunicação do PC com o Arduino se escreveu um script, com a execução automatizada na linguagem Python. Uma vez coletados, os dados foram tratados com a biblioteca “*matplotlib*” do Python no programa que roda no PC para o processamento dos dados coletados, gerando gráficos adquiridos em tempo real.

No caso da curva I-V ser linear, usando a biblioteca “*stats*” se realizou ajuste linear para determinação dos coeficientes linear e angular das retas, que fornece a resistência elétrica entre os terminais. No caso dos diodos, um script traçou retas tangentes aos ramos da curva I-V antes e depois da inflexão característica da curva de um diodo. A coordenada “V” do ponto de interseção (V,I) destas retas é obtida pela equação do ponto de interseção entre duas retas e fornece a leitura da tensão notável do diodo: tensão de condução no caso do diodo retificador, e a tensão de início do efeito Zener, no caso do diodo Zener.

Resultados e Discussões

Os valores das grandezas físicas relevantes estão apresentadas nas Figuras respectivas: resistências para resistores, tensões características das inflexões nos diodos. Todos estes valores são automaticamente fornecidos ao final da medida.

A Figura 2 mostra os resultados obtidos para as curvas I-V obtidas para um resistor medido isoladamente.

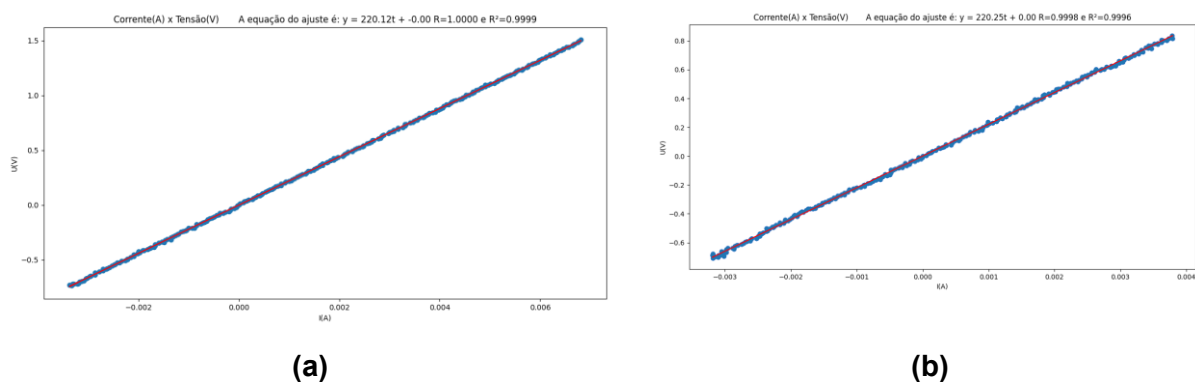


Figura 02: Curvas IV para um resistor obtidas com (a) Arduino UNO R3, (b) Raspberry Pi Pico.

A Figura 3 mostra os resultados obtidos para as curvas I-V obtidas para resistores associados em série.

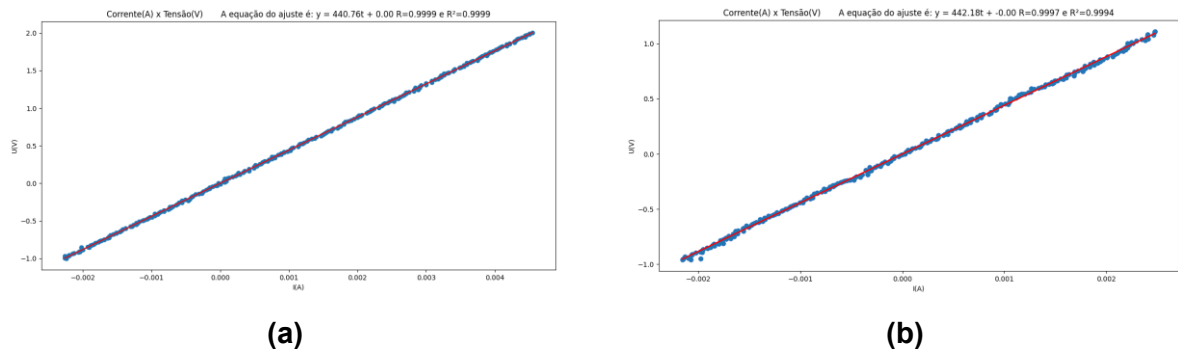


Figura 03: Curvas IV para resistores em série obtidas com (a) Arduino UNO R3, (b) Raspberry Pi Pico.

A Figura 4 mostra os resultados obtidos para as curvas I-V obtidas para resistores associados em paralelo.

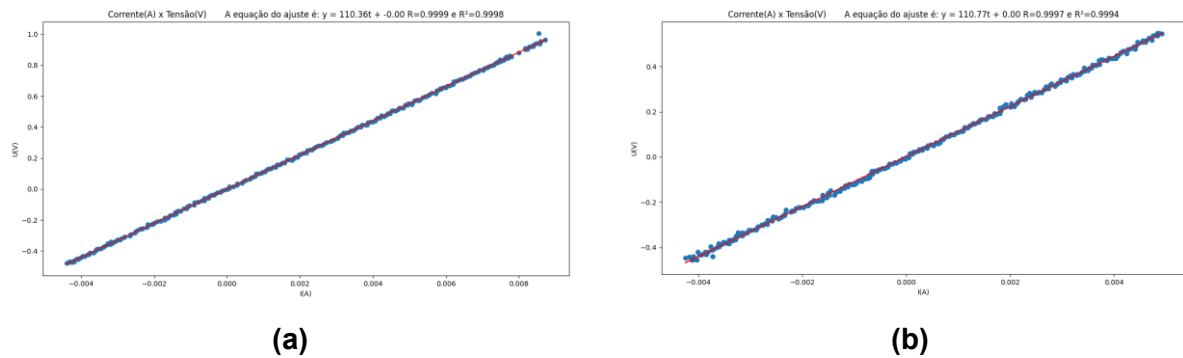


Figura 04: Curvas IV para resistores em paralelo obtidas com (a) Arduino UNO R3, (b) Raspberry Pi Pico.

A Figura 5 mostra os resultados obtidos para as curvas I-V obtidas para um diodo retificador.

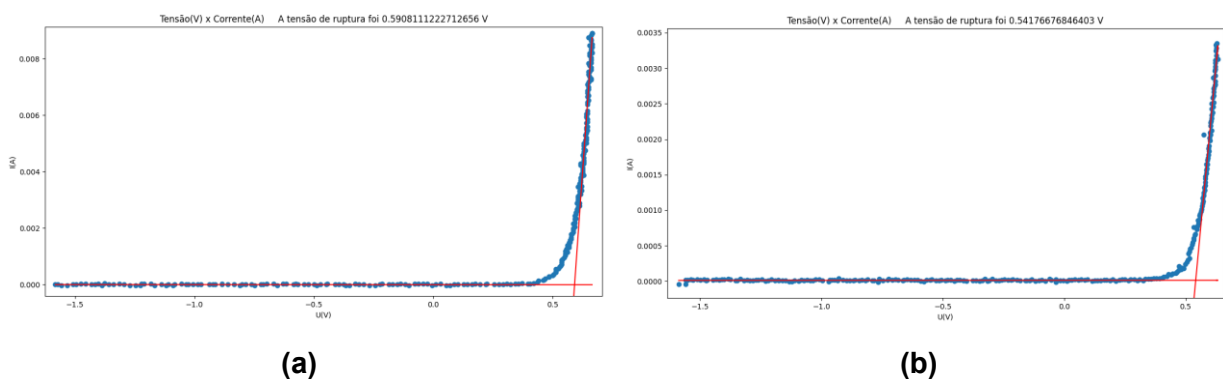


Figura 05: Curvas IV para um diodo retificador com (a) Arduino UNO R3, (b) Raspberry Pi Pico.

A Figura 6 mostra os resultados obtidos para as curvas I-V obtidas para um diodo Zener.

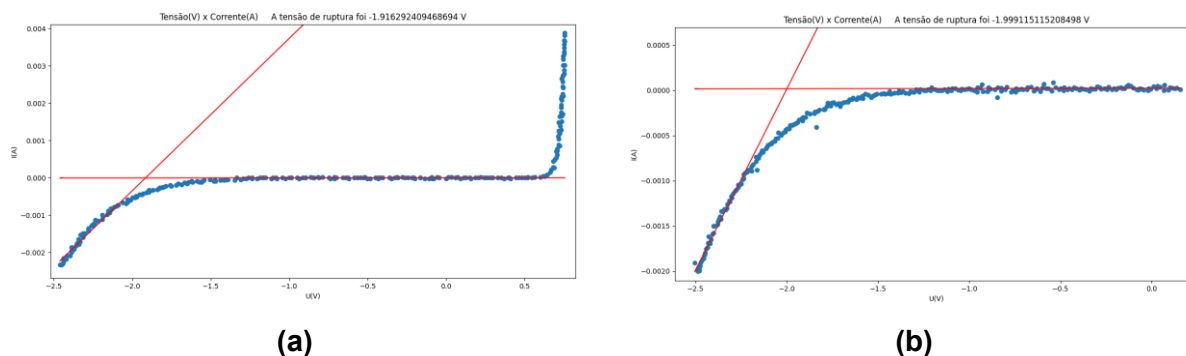


Figura 06: Curvas IV para um diodo Zener com (a) Arduino UNO R3, (b) Raspberry Pi Pico, dando destaque ao início do Efeito Zener.

A resistência individual medida com Arduino Uno R3 foi $220,12 \, \Omega$ e a medida com Raspberry Pi Pico foi $220,25 \, \Omega$ (Valor esperado: $220 \, \Omega$). A resistência equivalente da associação em série medida com Arduino Uno R3 foi $440,76 \, \Omega$ e com Raspberry Pi Pico foi $442,18 \, \Omega$ (Valor esperado: $440 \, \Omega$). A resistência equivalente da associação em paralelo medida com Arduino Uno R3 foi $110,36 \, \Omega$ e com Raspberry Pi Pico foi $110,72 \, \Omega$ (Valor esperado: $110 \, \Omega$). A tensão de condução do diodo medida com Arduino Uno R3 foi de $0,59 \, \text{V}$ e a medida com Raspberry Pi Pico foi de $0,54 \, \text{V}$ (Valor esperado: $0,6 \, \text{V}$ a $0,7 \, \text{V}$). A tensão de início do Efeito Zener medida com Arduino Uno R3 foi de $-1,916 \, \text{V}$ e a medida com Raspberry Pi Pico foi de $-1,999 \, \text{V}$ (valor esperado: $-2,0 \, \text{V}$).

Considerações Finais

Concluimos que usar MCU Raspberry Pi Pico fornece resultados compatíveis com a MCU Arduino UNO R3. Logo, pode ser considerada uma MCU integrante de sistemas de medição de baixo custo (o elemento mais caro nesta proposta é um computador com comunicação USB) para geração e visualização de curvas I-V, e a partir destas podem ser extraídos os parâmetros físicos relevantes para cada dispositivo estudado. Os resultados sugerem empregar esta plataforma Raspberry Pi Pico com Python para uso em outros sistemas de medição com aquisição de dados automatizada em laboratórios de ensino de Física. O Grupo PET-Física da UFRRJ agradece ao Fundo Nacional de Desenvolvimento da Educação (FNDE) pelo financiamento das Bolsas e do Custeio e à UFRRJ pela infraestrutura utilizada (Laboratório de Materiais e Dispositivos Supercondutores).

Referências

- (ARDUINO, 2023) ARDUINO. **Arduino Uno Rev3**. Disponível em: <<https://store.arduino.cc/products/arduino-uno-rev3>>. Acesso em 20 out. 2023.
- (CAPUANO, 2009) CAPUANO, Francisco Gabriel, MARINO, Maria Aparecida Mendes, **Laboratório de Eletricidade e Eletrônica**, São Paulo: Érica, 2009.
- (COMPONENTS101, 2023) COMPONENTS101. **MCP4725 IC Pinout, Datasheet, Equivalent, Circuit & Specs**. Disponível em: <https://components101.com/ics/mcp4725-pinout-datasheetequivalent-circuit-specs>. Acesso em: 12 jan. 2023.
- (MATPLOTLIB, 2023) THE MATPLOTLIB DEVELOPMENT TEAM. **Matplotlib: Visualization with Python**. Disponível em: <<https://matplotlib.org/>>. Acesso em 12 out. 2023.
- (MONTEIRO, 2022) J. A. Monteiro, M. F. Vilhena, F. H. S. Silva, I. C. R. de Lucena, A. L. Coutinho Júnior. **Arduino no Ensino de Física: uma Revisão Sistemática de Literatura de 2011 a 2021**. Amazônia: Revista de Educação em Ciências e Matemáticas, v. 18, n. 40, ago. 2022. Disponível em: <<https://periodicos.ufpa.br/index.php/revistaamazonia/article/view/12175>>. Acesso em: 21 out. 2023.
- (MOREIRA, 2018) M. M. P. C. Moreira, M. C. Romeu, F. R. V. Alves, F. R. O. da Silva. **Contribuições do Arduino no ensino de Física: uma revisão sistemática de publicações na área do ensino**. Caderno Brasileiro de Ensino de Física, v. 35, n. 3, p. 721-745, dez. 2018
- (OLIVEIRA, 2020). OLIVEIRA, Claudio Luis Vieira, ZANETTI, Humberto Augusto Piovesana. **Projetos com Python e Arduino: Como Desenvolver Projetos Práticos de Eletrônica**, Automação e IOT. São Paulo: Érica (2020).
- (PYFIRMATA, 2023). PYTHON SOFTWARE FOUNDATION. **pyFirmata 1.1.0**. Disponível em: <<https://pypi.org/project/pyFirmata/>>. Acesso em 20 out. 2023.
- (PYSERIAL, 2023). PYTHON SOFTWARE FOUNDATION. **pyserial 3.5**. Disponível em: <<https://pypi.org/project/pyFirmata/>>. Acesso em 12 out. 2023
- (PYTHON, 2023) PYTHON SOFTWARE FOUNDATION. **Welcome to Python**. Disponível em: <<https://www.python.org/>>. Acesso em 12 out. 2023.
- (RASPBERRY, 2023) RASPBERRY PI FOUNDATION. **Raspberry Pi Pico**. Disponível em: <https://www.raspberrypi.com/products/raspberry-pi-pico/>. Acesso em: 10 out. 2023.