**Aim:**

**Transistor Identification and SOS Signal Transmission: Building Efficient Circuits with Raspberry Pi**

**Objectives:**

1. **Understanding Basics of Electronics:**

Understanding basics of electronics and its components/items such as Breadboard, Resistors, Diodes, Transistors etc.

1. **Identification and verification of Transistors.**

Explore the fundamental properties of transistors, examining their identification markers and verifying their functionality through practical experimentation.

1. **Flashing SOS Signal in Morse Code using External LED:**

Develop an electronic circuit wherein an external LED transmits the SOS distress signal in Morse code — comprising **three dots, three dashes, and three dots** — continuously using Raspberry Pi Pico. The LED illuminates for **0.25 seconds for a dot and 1 second for a dash, with a 0.5-second interval (GAP time) between dots and dashes**. This pattern repeats continuously, with a 2-second delay between repetitions.

1. **Building an Efficient External Electronic Circuit:**

Design a more streamlined electronic circuit employing an external LED and a buzzer controlled by Raspberry Pi Pico through a **transistor**. This circuit efficiently transmits the SOS signal in Morse code, ensuring effective communication in emergency scenarios.

**Summary of Experiment - 2 Goals and Outcomes**

By the end of this experiment, students will have gained a comprehensive understanding of transistors and their applications in electronic circuits. Through hands-on activities, they will identify and verify transistors, enhancing their practical knowledge of semiconductor devices.

Moreover, students will acquire the skills to design and implement efficient electronic circuits using Raspberry Pi Pico. They will master the art of transmitting the SOS distress signal in Morse code through an external LED and a buzzer, demonstrating their proficiency in coding and circuitry.

**Pre-Lab Questionnaire:**

1. What is Biasing?
2. Differentiate between forward and reverse biasing.
3. In terms of application, give a comparison between the Diode and Transistor.
4. Is Morse code still used in 2023?
5. What is the primary function of a transistor in an electronic circuit?
6. Explain the significance of the SOS signal in Morse code.
7. What is the purpose of using a transistor in this experiment's circuit?
8. What is the dot time, dash time, and gap time in Morse code, and how are they represented in this experiment's circuit?
9. How is the SOS distress signal represented in Morse code (dots and dashes)?
10. Explain the purpose of the dot time, dash time, and gap time in Morse code transmission.

**Answers to Pre-Lab Questions**

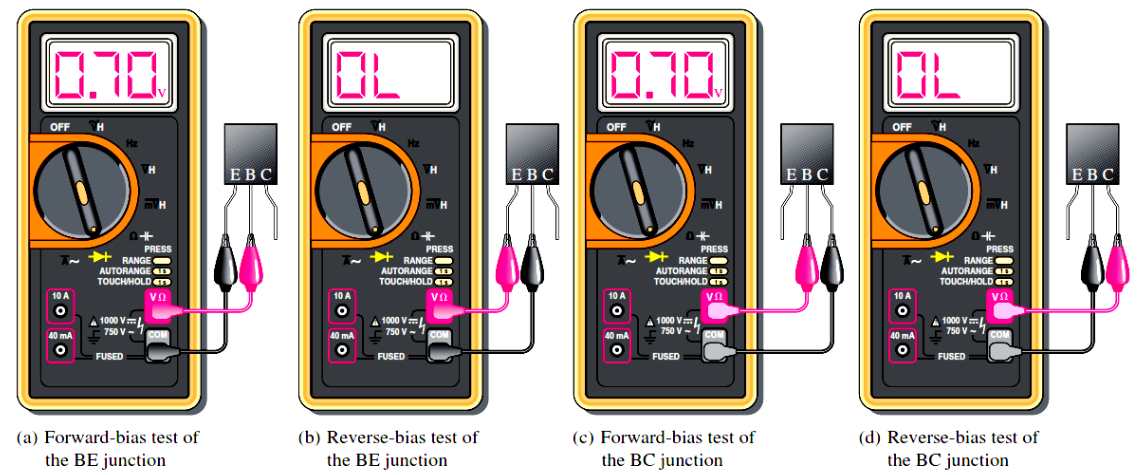
**Components/Equipment Required:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sl. No.** | **Name of the**  **Component / Equipment** | **Specification** | **Quantity** |
| **1** | Raspberry Pi Pico | RP2040 microcontroller chip, 125MHz | 1 |
| **2** | Raspberry Pi Pico cable | USB Type A to Micro-B | 1 |
| **3** | Resistors (carbon type) | ¼ watt (330Ω, 10k) | 1 each |
| **4** | LED | 3mm, Red, Blue, Green | 1 each |
| **5** | Diode (Si diode) | 1N4148 | 1 |
| **6** | Transistor | 2N2222, BC 108, BC 548 | 1 each |
| **7** | Buzzer | Active | 1 |
| **8** | Breadboard | 840 Tie points | 1 |
| **9** | Digital Multimeter | --------------------------- | 1 |
| **10** | Jumper Wire | --------------------------- | As per requirement |

**Objective 2**

**Identification and verification of Transistors.**

**Circuit / Schematic Diagram**



P/N

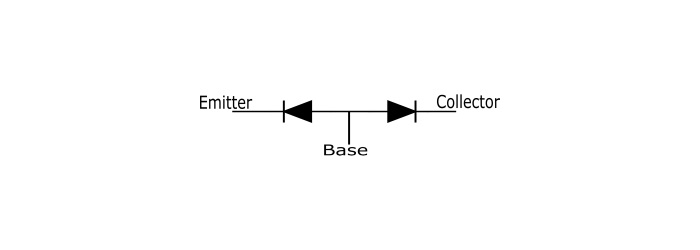
N/P

P/N

**1**

**2**

**3**



***Transistor Schematic Diagram***

(Figure 1 : Schematic diagram of a Transistor Identification and verification)

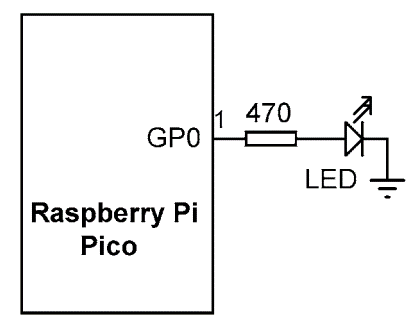
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Unknown  Transistors | Biasing | Voltage between Terminal 1 & 2 (V1) | Voltage between Terminal 2 & 3 (V2) | Type of Transistor  (NPN or PNP) | V1 > V2 | V2 > V1 | Remark |
| Unknown  Transistor  1 | **Forward Bias** |  |  |  | **1 : \_\_\_\_\_**  **2: \_\_\_\_\_**  **3: \_\_\_\_\_** | **1 : \_\_\_\_\_**  **2: \_\_\_\_\_**  **3: \_\_\_\_\_** |  |
| **Reverse Bias** |  |  |  |  |
| Unknown  Transistor  2 | **Forward Bias** |  |  |  | **1 : \_\_\_\_\_**  **2: \_\_\_\_\_**  **3: \_\_\_\_\_** | **1 : \_\_\_\_\_**  **2: \_\_\_\_\_**  **3: \_\_\_\_\_** |  |
| **Reverse Bias** |  |  |  |  |

**Objective 3**

**Flashing SOS Signal in Morse Code using External LED:**

Develop an electronic circuit wherein an external LED transmits the SOS distress signal in Morse code — comprising **three dots, three dashes, and three dots** — continuously using Raspberry Pi Pico. The LED illuminates for **0.25 seconds for a dot and 1 second for a dash, with a 0.5-second interval (GAP time) between dots and dashes**. This pattern repeats continuously, with a 2-second delay between repetitions.

**Circuit / Schematic Diagram**



(Figure 2 : Circuit diagram for an external LED flashes the SOS signal in Morse code)

**Code**

**write a program that turns an external LED to flash the SOS signal in Morse code (three dots, followed by three dashes, followed by three dots) continuously.**

**Observation**

**Figure 3: (Simulation-based electronic circuit in which an external LED flashes the SOS signal in Morse code.)**

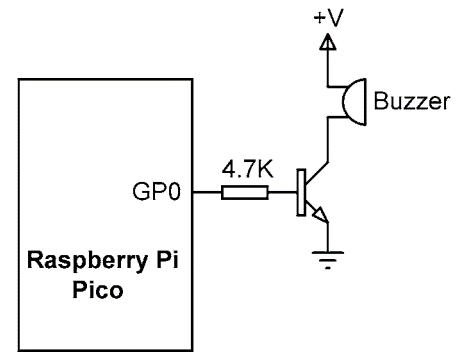
**Figure 4: (Hardware implementation-based electronic circuit in which an external LED flashes the SOS signal in Morse code.)**

**Objective 4**

**Building an Efficient External Electronic Circuit:**

Design a more streamlined electronic circuit employing an external LED and a buzzer controlled by Raspberry Pi Pico through a **transistor**. This circuit efficiently transmits the SOS signal in Morse code, ensuring effective communication in emergency scenarios.

**Circuit / Schematic Diagram**



**(Figure 5 : Circuit diagram to blink an external LED with buzzer sends the SOS signal in Morse code using Raspberry Pi Pico and a transistor.)**

**Code**

**Implementation of a more efficient electronic circuit in which an external LED with a buzzer sends the SOS signal in Morse code using Raspberry Pi Pico and a transistor.**

**Observation**

**Figure 6: (Hardware Implementation of an external LED with buzzer sending the SOS signal in Morse code using Raspberry Pi Pico and a transistor.)**

**Conclusion:**

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

**Precautions:**

|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

|  |
| --- |
|  |

**Post Experiment Questionnaire:**

1. Describe the role of the transistor in the circuit.
2. Explain the significance of the gap time in Morse code transmission.
3. Discuss the challenges faced during the implementation of the circuit and how they were resolved.
4. How could the Morse code transmission circuit be extended for more complex messages or signals?
5. Discuss the real-world applications of circuits transmitting Morse code signals.
6. Explain the role of the gap time in Morse code transmission and why it is necessary.
7. How does the transistor amplify the signal in the circuit?
8. Discuss any challenges faced during the experiment and how they were overcome.
9. In what real-world scenarios could a circuit transmitting Morse code signals be employed?

**Answers to Post-Lab Questions**

|  |  |  |  |
| --- | --- | --- | --- |
| **(Signature of the Faculty)** | |  | **(Signature of the Student)** |
|  | | **Name:** |  |
| **Date:** |  | **Registration No.:** |  |
|  |  | **Branch:** |  |
|  | | **Section** |  |