NIRMA UNIVERSITY

INSTITUTE OF TECHNOLOGY

NAAC ACCREDITED 'A+' GRADE

RINGTONE PLAYER USING 8051

MNM (2EC404)

21BEC116 Neel Sheth

21BEC117 Shreya Nahta

Submitted to:

Dr. Jayeshkumar Patel

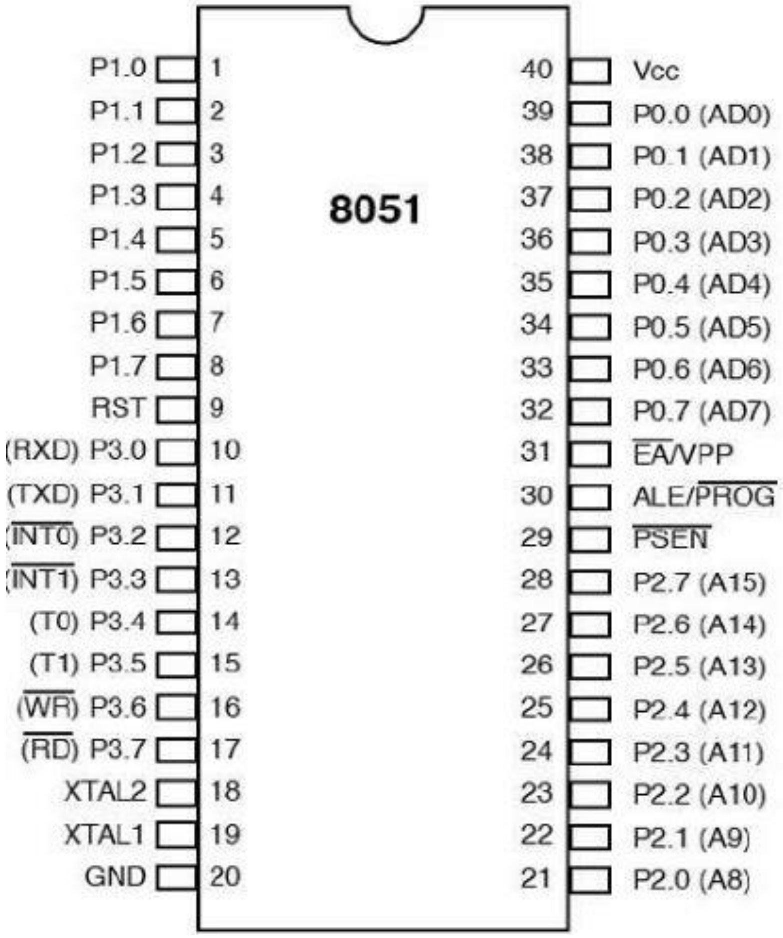
INTRODUCTION

A ringtone player using an 8051 microcontroller is a simple electronic circuit that a tone or melody, typically for use as a ringtone in a phone or other communication device. The 8051 microcontroller is a popular choice for implementing such a circuit because of its low cost, ease of programming, and availability of development tools. The basic idea behind a ringtone player using an 8051 microcontroller is to generate a square wave of a specific frequency and duration, which when played through a speaker, produces a tone. By sequencing different frequencies and durations, you can create a melody or song.

The microcontroller can be programmed in assembly or C language to generate the desired square wave using a timer and output pins. The frequency and duration of the square wave can be controlled using the timer and software, allowing for precise control over the tone and melody produced by the circuit.

Ringtone players using 8051 microcontrollers can be simple or complex, depending on the desired functionality. For example, you could create a basic ringtone player that plays a pre-programmed melody, or you could add a serial port to receive commands from an external device to play custom ringtones.

PIN DIAGRAM

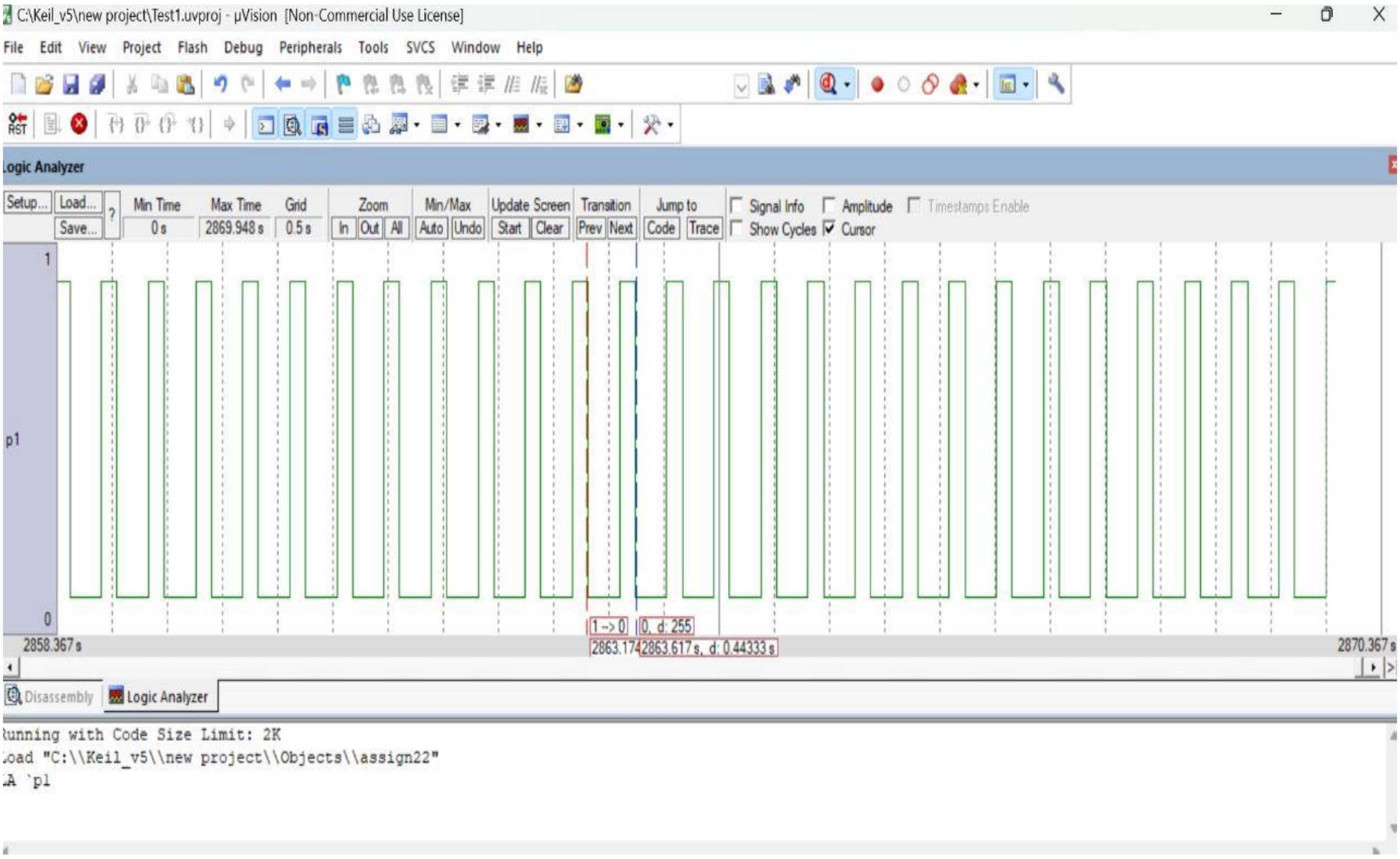


WORKING OF RINGTONE PLAYER USING 8051

The following steps are taken to implement a ringtone player using an 8051 microcontroller:

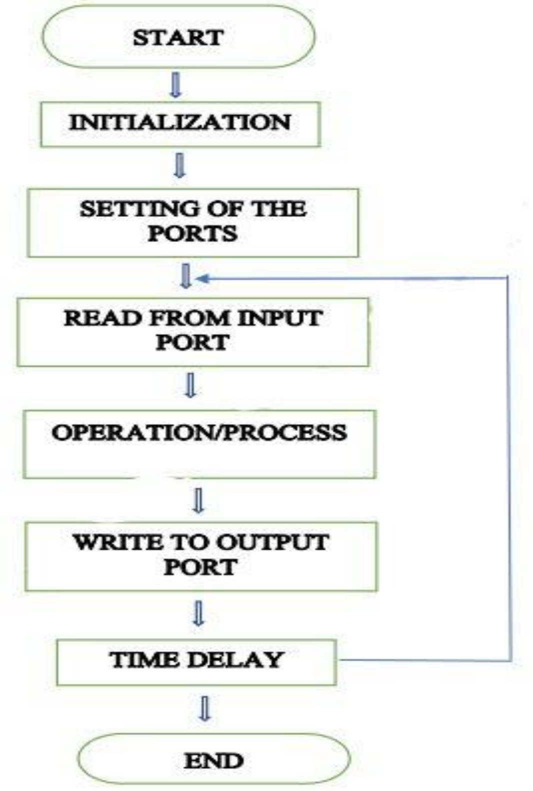
1. Choose a suitable microcontroller, such as the AT89C51 or AT89S52, which have built-in timers and programmable I/O ports.
2. Connect a crystal oscillator to the microcontroller to provide an accurate clock signal. The oscillator frequency should be set to 11.0592 MHz
3. Connect a speaker to one of the microcontroller's output pins through a resistor and capacitor network to filter out any DC component and prevent damage to the speaker.
4. Write a program in assembly or C language to generate a square wave of the desired frequency using the microcontroller's built-in timer and output pins. The frequency of the square wave will determine the pitch of the tone produced by the speaker.
5. To produce a melody, you will need to sequence different frequencies and durations of the square wave using a timer interruption. For example, you could store the frequency and duration of each note in an array and play them one by one in a loop.
6. We can also use the microcontroller's serial port to receive commands from an external device, such as a computer or smartphone, to play specific melodies or tones.

SIMULATION OF CODE



```
Running with Code Size Limit: 2K
Load "C:\\Keil_v5\\new project\\Objects\\assign22"
LA `p1
```

FLOW CHART



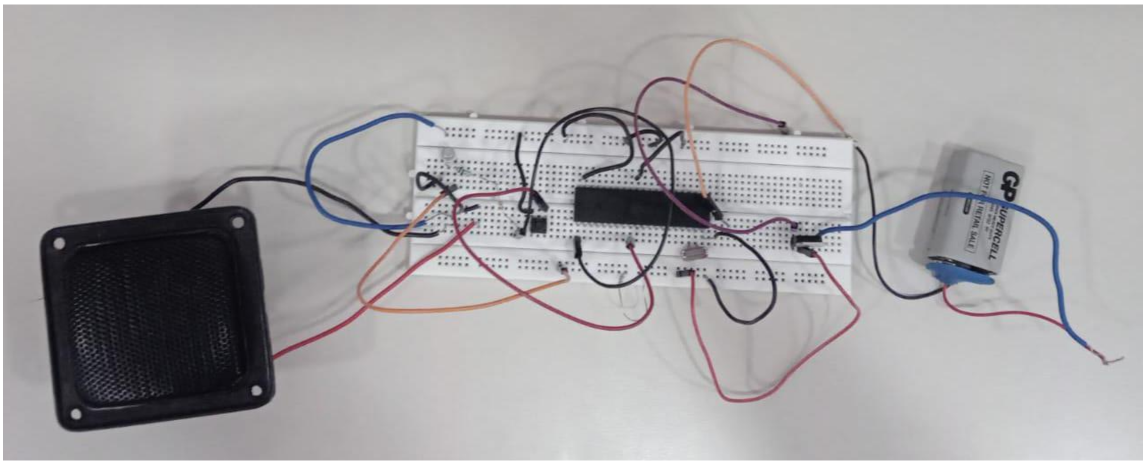
CODE

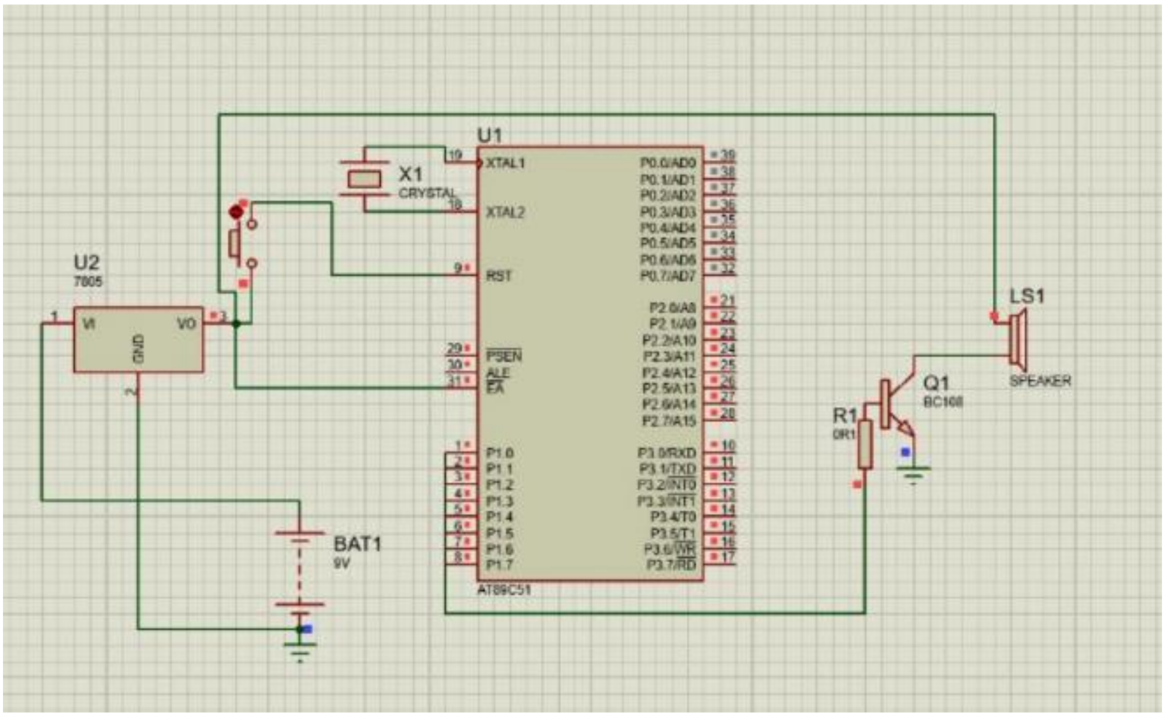
```
 ORG 0000h                        ; Set program start address
  MOV P1, #0x00                   ; Set P1 port to output
LOOP:
    ; For 2kHz
    MOV TH0, #0xFD                ; Set timer 0 high byte for 2kHz frequency
    MOV TL0, #0xFD                ; Set timer 0 low byte for 2kHz frequency
    SETB TR0                      ; Start timer 0
    SETB P1.0                     ; Turn on buzzer
    ACALL DELAY                   ; Delay for tone duration
    CLR P1.0                      ; Turn off buzzer
   ACALL DELAY                    ; Delay for half the timer period
   ACALL DELAY
    ; For 3kHz
    MOV TH0, #0xFA               ; Set timer 0 high byte for 3kHz frequency
    MOV TL0, #0xF4               ; Set timer 0 low byte for 3kHz frequency
    SETB TR0                      ; Start timer 0
    SETB P1.0                     ; Turn on buzzer
    ACALL DELAY                   ; Delay for tone duration
    CLR P1.0                      ; Turn off buzzer
    ACALL DELAY                    ; Delay for half the timer period
   ACALL DELAY

; For 4kHz
    MOV TH0, #0xF7           ; Set timer 0 high byte for 4kHz frequency
    MOV TL0, #0xE8           ; Set timer 0 low byte for 4kHz frequency
    SETB TR0                 ; Start timer 0
    SETB P1.0                ; Turn on buzzer
    ACALL DELAY              ; Delay for tone duration
    CLR P1.0                  ; Turn off buzzer
    ACALL DELAY              ; Delay for half the timer period
    ACALL DELAY

  ; For 5kHz
    MOV TH0, #0xF4           ; Set timer 0 high byte for 5kHz frequency
    MOV TL0, #0xDC           ; Set timer 0 low byte for 5kHz frequency
    SETB TR0                 ; Start timer 0
    SETB P1.0                  ; Turn on buzzer
    ACALL DELAY              ; Delay for tone duration
    CLR P1.0                   ; Turn off buzzer
    ACALL DELAY
    ACALL DELAY                ; Delay for half the timer period
 SJMP LOOP                          ; Jump back to the main loop
DELAY:
       MOV R2, #0xFF
       MOV R3, #0ffh
        MOV R4, #0ffh                        ; Initialize delay counter
 DELAY_LOOP:
          DJNZ R3, DELAY_LOOP
         DJNZ R4, DELAY_LOOP
          DJNZ R2, DELAY_LOOP       ; Decrement R2 until zero
     RET                                     ; Return from delay subroutine
      END                                     ; End of program
```

CIRCUIT DIAGRAM





Applications

An application for ringtone player is a software application designed to allow users to customize their mobile device's ringtone. It provides a collection of pre-installed ringtones and also allows users to upload their own ringtones from their device. The application provides an easy-to-use interface for users to browse through the collection of ringtones, select their desired ringtone, and set it as their default ringtone or assign it to a specific contact. In addition to the basic functionality, some ringtone player applications may include additional features such as the ability to categorize ringtones, create custom playlists, set different ringtones for different notifications, and search capabilities to help users find their desired ringtone quickly.

Ringtone player applications are available for various platforms, including Android, iOS, and web-based platforms. Users can download these applications from the appropriate platform's app store and customize their device's ringtone according to their preferences.

BILL OF MATERIALS

1. 11.0592Mhz Crystal Oscillator	-Rs 10/-
2. Push Button	-Rs 2/-
3. AT89C51 Microcontroller	-Rs 90/-
4. wires	-Rs 10/-
5. Speaker	-Rs 60/-
6. Battery(9V)	-Rs 20/-
7. BC108	-Rs 5/-
8. Resistors(10k+39k)	-Rs 4/-
Total	**-Rs 206/**

Summary

In conclusion, a ringtone player using an 8051 microcontroller is a simple yet versatile circuit that can generate a tone or melody for use in a phone or other communication device. The 8051 microcontroller is a popular choice for implementing such a circuit due to its low cost, ease of programming, and availability of development tools.

By programming the microcontroller to generate a square wave of specific frequencies and durations, you can create a wide variety of melodies and tones. The circuit can be designed to play pre-programmed melodies or to receive commands from an external device to play custom ringtones.

This provides hands-on experience in designing and programming a simple electronic circuit that can produce audible output.