Example 12.1. Design an interface 32 Ω speaker and use it to generate a soft 1 kHz sound.

Solution: To make sound we need to create an oscillating wave. In this example, the wave will be a simple square wave. At 3.3V, a 32 Ω speaker will require a current of about 100 mA. The maximum the TM4C123 can produce on an output pin is 8 mA. If we place a resistor in series with the headphones, then the current will only be 3.3V/(1500+32Ω) = 2.2mA. To generate the 1 kHz sound we need a 1 kHz square wave. There are many good methods to generate square waves. In this example we will implement one of the simplest methods. We will activate a periodic interrupt and toggle an output pin in the ISR. To generate a 1 kHz wave we will toggle the PA5 pin every 500 µs. We will assume the PLL is active and the system is running at 80 MHz. We wish to initialize the SysTick to interrupt with a period of 500 µs. The correct value for reload is 39999 ((500µs/12.5ns)-1). If the bus frequency were to be 16 MHz, we would set the reload value to be 7999 ((500µs/62.5ns)-1). Since this sound wave output is a real-time signal, we set its priority to highest level, which is 0. See Program 12.6.

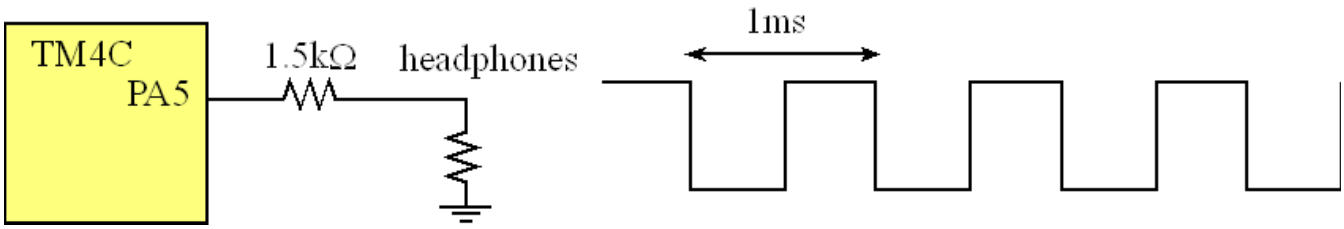

Figure 12.6. A squarewave output connected to a speaker will generate sound.

```
void Sound_Init(void){ unsigned long volatile delay;
  SYSCTL_RCGC2_R |= 0x00000001; // activate port A
  delay = SYSCTL_RCGC2_R;
  GPIO_PORTA_AMSEL_R &= ~0x20;      // no analog
  GPIO_PORTA_PCTL_R &= ~0x00F00000; // regular function
  GPIO_PORTA_DIR_R |= 0x20;     // make PA5 out
  GPIO_PORTA_DR8R_R |= 0x20;    // can drive up to 8mA out
  GPIO_PORTA_AFSEL_R &= ~0x20;  // disable alt funct on PA5
  GPIO_PORTA_DEN_R |= 0x20;     // enable digital I/O on PA5
  NVIC_ST_CTRL_R = 0;           // disable SysTick during setup
  NVIC_ST_RELOAD_R = 39999;     // reload value for 500us (assuming 80MHz)
  NVIC_ST_CURRENT_R = 0;        // any write to current clears it
```

```
  NVIC_SYS_PRI3_R = NVIC_SYS_PRI3_R&0x00FFFFFF; // priority 0
  NVIC_ST_CTRL_R = 0x00000007;  // enable with core clock and interrupts
  EnableInterrupts();
}

void SysTick_Handler(void){
  GPIO_PORTA_DATA_R ^= 0x20;     // toggle PA5
}
```

Program 12.6. Sound output using a periodic interrupt (C12_SoftSound).

Observation: To make a quieter sound, we could use a larger resistor between the PA5 output and the speaker.