Implementation of the below circuit using avr-gcc

M Sai Sarath Chandra
chandu.4567890@gmail.com
FWC22117
IIT Hyderabad-Future Wireless Communication Assignment

March 2023

Contents

1 Problem

GATE EC-2019
Q.25. In the circuit shown,the clock frequency, i.e.,the frequency of the clock signal ,is 12 KHz.The frequency of the signal at Q2 is ..... KHz.

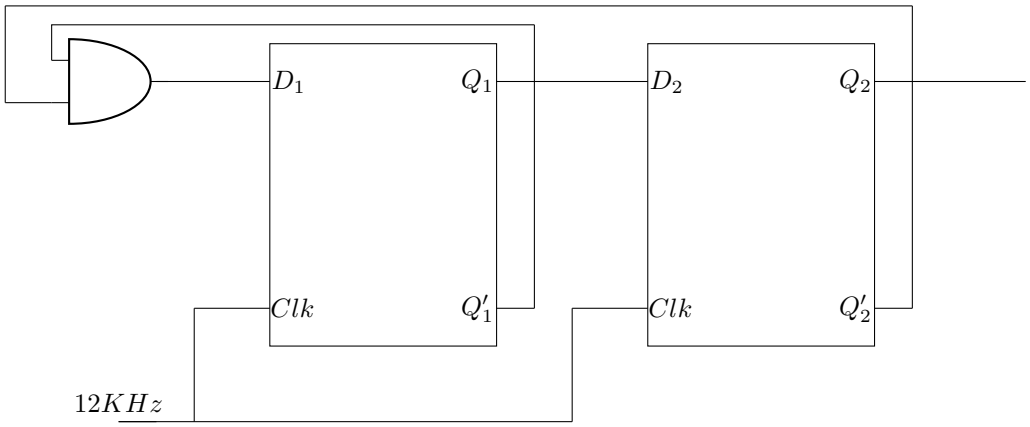

Figure 1: circuit

2 Introduction

The aim is to implement the above sequential circuit using D flip-flops (IC 7474) and to find out the frequency of the signal at Q2(it is given that the frequency of the clock signal is 12KHz).IC 7474 is a dual positive edge triggered D type flip flop,which means it has two separate flip-flop that are triggered by the rising edge of a clock signal.

In the above circuit Q_1,Q_2 are inputs and D_1,D_2 are outputs.So,from the circuit the expressions of D_1 and D_2 are:

$D_1 = Q_1'Q_2'$.

$D_2 = Q_1$.

Below is the transition table of the above circuit which is as follows:

INPUT		OUTPUT	
Q_1	Q_2	D_1	D_2
0	0	1	0
1	0	0	1
0	1	0	0

Table 1: Transition table

3 Components

COMPONENTS		
Component	Value	Quantity
Resistor	=220 Ohm	1
Arduino	UNO	1
Seven Segent Display	Common Anode	1
Decoder	7447	1
Flip Flop	7474	1
Jumper Wires		20
Breadboard		1

Table 2: Components

4 Hardware

IC 7474 is a D flip-flop integrated circuit that is commonly used in digital electronics applications.It is a dual positive edge-triggered by the rising edge of a clock signal.Below is the pin diagram of IC 7474:

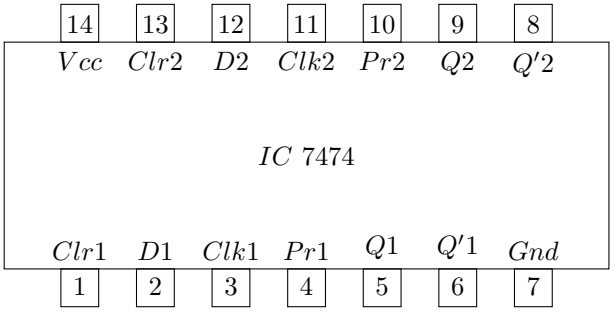

Figure 2: 7474

The connections between the arduino and IC 7474 is as follows:

	INPUT		OUTPUT		CLOCK		VCC			
ARDUINO	D2	D3	D5	D6	D13		5V			
7474	5	9	2	12	3	11	1	4	10	13
7447			1	7				16		

Table 3: connections

5 Software

The code to implement the above circuit is :

```
#include<avr/io.h>
#include<util/delay.h>
#include<stdbool.h>

void loop()
{

    generateTone(12000, 2000);
    _delay_ms(2000);
}

void generateTone(uint16_t frequency, uint16_t duration)
{
    uint16_t period = 1000000 / frequency;
    uint16_t pulseWidth = period / 2;

    uint16_t numCycles = frequency * duration / 1000;

    for (uint16_t i = 0; i < numCycles; i++)
    {
        PORTB |= (1 << PB5);
        _delay_us(42);
        PORTB &= ~(1 << PB5);
        _delay_us(42);
    }
}
int main(void)
{

bool B,A;
bool Q1=0,Q2=0;

DDRD = 0b01100000;
DDRB = 0b00100000;
PORTD = 0b11000000;
PORTB = 0b00000011;

while(1)
```

```
{
    generateTone(12000, 2000);

        PORTB = ((1 << PB5));

        _delay_ms(200L);

B=((!Q1) && (!Q2));
A=((Q1) && (!Q2));

PORTD = (A << 5);
PORTD = (B << 6);

Q1 = (PIND & (1 << PIND2)) == (1 << PIND2);
Q2 = (PIND & (1 << PIND3)) == (1 << PIND3);

        PORTB = ((0 << PB5));

        _delay_ms(200L);
}

        return 0;
}
```