

fig1: Circuit schematic for lab 11

**Theory of Operation (Hardware):**

In the circuit represented in fig1, there are various parts such as an ATmega324a microcontroller, linear arrangement of 8 switched encoded by a 74HC148 priority encoder. An Advanced DC motor board is used which is not shown in the figure. The connections to Advanced DC motor board is made via a 10 pin header (J2). The 10 pin header provides connections to a ET-MINI DC-MOTOR whose rotation speed is controlled by the ATmega324a microcontroller through creating a waveform with certain frequency at PD0. The connection of PD0 to the header J2 is shown in the figure. Serial Peripheral Interface (SPI) is used to make connections between the ATMega324A and the slaves in DC motor module. The ATmega324A contains three independent hardware timer/counters. Timer/Counter0 and Timer/Counter2 are 8-bit counters. Timer/Counter1 is 16-bit counter. Timer/Counter1 is used to create a waveform at PD0. Timer/Counter has output compare registers OCR1A and OCR1B. Values in the OCR1A or OCR1B determines how long the counter is counting before an interrupt is requested. When the counter value reaches OCR1A, a pulse of length 1ms is produced at PD0 and the counter is reset. Changing the value of the register OCR1A, changes the frequency of the waveform being generated at PD0. The keys are arranged in a linear arrangement where the key at the top is 7 and the key at the bottom is 0. Pressing key 7 halves the value of OCR1A whereas key 6 doubles the value of the OCR1A. Key 5 decreases the value of OCR1A by 50 whereas key 4 increases the value of OCR1A by 50. Key 3 decreases the value of OCR1A by 10 whereas key 2 increases the value of OCR1A by 10. Key 1 increases the frequency by 1Hz and key 0 decreases the frequency by 1Hz. 8 SIP(Single Inline Package) resistors used as pull up resistors for the switches connected to the priority encoder. Output EO of the priority encoder is connected to the INT0 pin PD2. When there’s a rising edge at PD2 the interrupt service routine corresponding to INT0 is called.

A Display On Glass (DOG) LCD alphanumeric display is used which is not shown in the figure. The connections to the LCD display is made via a 10 pin header (J2). The 10 pin header provides connections to a ST7036 Dot Matrix Controller Driver on the DOG LCD Module which also includes its own microcontroller that controls what is displayed on the LCD display. Serial Peripheral Interface (SPI) is used to make connections between the master (ATMega324A) and the slaves in the DOG LCD module. The LCD display 3 lines of display with each line capable of displaying 16 characters. The values to be displayed on the LCD are stored in the SRAM of the ATMega324A by the use of 16 byte buffers. The content of the buffers are transferred to the LCD display driver which displays the characters corresponding to the values (ASCII) stored in the buffers. Data indirect addressing technique is used to store values in the data space. Frequency and rpm are shown in the first two lines of the LCD display. These parts along with their functions in the hardware shown in fig1 will be described below.

**ATmega324A:** At the core of the circuit is the ATmega324A microcontroller (see figure 2). The ATmega324A is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega324A achieves throughputs close to 1MIPS per MHz. With 40 pins, it can be modified for alternative functions. The ATmega324A operates at a range of 2.7 – 5.5V, with two power pins (pin 10 and pin 30) and two ground pins (pin 11 and 31). The Atmega324A resets when powered on, and offers the option to have a manual external reset switch if desired. The external reset is connected to a switch as shown in fig1. The Atmega324A contains 32 general purpose registers (32 8-bit registers R0-R31). All the 32 general purpose registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in a single instruction executed in one clock cycle(*Source: Atmega324A Datasheet)*. The ATmega324A has four different ports: portA, portB, portC, portD. Each port has three registers associated with it: PORTX, DDRX, PINX; where ‘X’ simply represents A,B,C or D. In this lab, PORTA pins are configured as inputs. PD0 is configured as an output pin whereas PD2 (INT0) is configured as an input pin.



Figure 2: ATmega324a

**Resistor Network3:** The resistor network referred as R3 in the schematic is an 8 SIP(Single Inline Package) resistor pack. The resistors have a common connection which is connected to the switches in linear arrangement. The purpose of the resistors is to provide a pull up approach. When the switches are open the input to the encoder will be a ‘1’. When the switch is pressed an input of ‘0’ will be provided.

**Priority Encoder(74HC148):**

A **priority encoder** is a circuit or algorithm that compresses multiple binary inputs into a smaller number of outputs. In this case, the priority encoder is a device with internal circuitry that chooses a certain input based on the “priority”. The key-word is priority. The inputs of a 74HC148 is connected to an arrangement of switches. The switches can be arranged in various forms, for example, a matrix form or a linear form. In this lab, a linear arrangement of the switches is used. Since the switches are not directly connected to the micro-controller, this type of linear arrangement including a encoder is called **linear externally encoded keys(or switches).**



74HC148 Priority Encoder

The 74HC148 has 8 inputs and three outputs. The outputs are complimented. The linear arrangement of the switches are connected to the 8 inputs () and the outputs ) are connected to the pins . The priority encoder also has an input , which when asserted low allows the priority encoder to encode the keys(switches). The Priority Encoder also has two other outputs, . Output goes to 0 every time a switch is pressed whereas output E0 goes to 1 every time the switch is pressed. E0 is connected to a low pass filter composed of R4 and C2 whose output is connected to PD2. At PD2, an external interrupt INT0 is requested every time there’s a positive edge at PD2; if the interrupt service routine is called, then the microcontroller displays the number represented by the complement of the outputs, i.e., the outputs will be complemented because a double complement ends up giving the original value.

The 74HC148 has a wide operating voltage range of 2V to 6V. The typical propagation delay of the 74HC148 is 16 ns. The 74HC148 priority encoder encodes eight data lines to 3-Line Binary(Octal). The 3 bit output represents the position of the switch that was pressed in complemented form. For example, if the output is 000, the switch at position 111(7) was pressed. If two switches are pressed at the same time, the encoder gives priority to the higher position. That is, if switches at positions 7 and 6 were pressed at the same time, the output will be 000(complement of 7). Hence, the word, **priority.**

## FUNCTION TABLE FOR 74HC148

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  | **INPUTS** |  |  |  |  |  | **OUTPUTS** | | |  |
| **EI** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **A2** | **A1** | **A0** | **GS** | **EO** |
| H | X | X | X | X | X | X | X | X | H | H | H | H | H |
| L | H | H | H | H | H | H | H | H | H | H | H | H | L |
| L | X | X | X | X | X | X | X | L | L | L | L | L | H |
| L | X | X | X | X | X | X | L | H | L | L | H | L | H |
| L | X | X | X | X | X | L | H | H | L | H | L | L | H |
| L | X | X | X | X | L | H | H | H | L | H | H | L | H |
| L | X | X | X | L | H | H | H | H | H | L | L | L | H |
| L | X | X | L | H | H | H | H | H | H | L | H | L | H |
| L | X | L | H | H | H | H | H | H | H | H | L | L | H |
| L | L | H | H | H | H | H | H | H | H | H | H | L | H |

(*Taken from the 74HC148 datasheet)*

**DC-motor board interface(J2):** The connections to Advanced DC motor board is made via a 10 pin header (J2). The 10 pin header provides connections to a ET-MINI DC-MOTOR whose rotation speed is controlled by the ATmega324a microcontroller through creating a waveform with certain frequency at PD0. The connection of PD0 to the header J2 is shown in the figure. Serial Peripheral Interface (SPI) is used to make connections between the ATMega324A and the DC motor module.

**DOG LCD Interface(J2):** This a 10pin header that provides connections to a ST7036 Dot Matrix Controller Driver on the DOG LCD Module which also includes its own microcontroller that controls what is displayed on the LCD display. Serial Peripheral Interface (SPI) is used to make connections between the master (ATMega324A) and the slaves in the DOG LCD module. MOSI and SCK are SPI signals. RS refers to register select which selects between command register and data register in the display driver. BLC is a control signal for the backlight in the display.

**Low Pass Filter:** Low Pass Filter in the circuit is composed of the parts R4 and C2. The primary function of the circuit is to eliminate the bounces at EO. Any sharp charges will be neglected by the circuit as the voltage increases and decreases at the capacitor slowly.



**Theory Of Operation(Software):**

This system inputs a 3-bit key code from a 74HC148 each time a key is pressed. The program uses interrupt to record the positive edge at PD2(INT0), and once it finds a positive edge at PD2(where EO is connected), it calls the interrupt service subroutine(keypress\_isr). The isr keypress\_isr reads the input from PORTA and shifts the position of the bits 6, 5 and 4 to 2, 1 and 0. Checks if key 7,6,5,4,3, 2,1 or 0 was pressed. If key 7 is pressed, the frequency is doubled whereas key 6 halves the frequency. Key 5 course increments the frequency and key 4 course decrements the frequency. Key 3 fine increments the frequency and key 2 fine decrements the frequency. If key 1 is pressed, the frequency is increased by 1Hz. If key 0 is pressed, the frequency is decreased by 1 Hz. The corresponding value of OCR1A is calculated and loaded after each valid keypress.The program also uses a Timer/Counter1 interrupt.The interrupt service routine "tmr1\_comp\_match " is called every time the counter reaches the value stored at OCR1A(output compare register 1A). The counter counts the clock cycles, prescalar value of 8 is used for the clock; evrytime the counter reaches the specified value in OCR1A it clears. The isr "tmr1\_comp\_match " produces a pulse at PD0 with a width of 1ms. The delay of 1ms is provided by the use of the sybroutine "var\_delay".