VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Electronics Engineering (SENSE)

PROJECT REPORT			
COURSE CODE / TITLE	BECE204L– Microprocessors and Microcontrollers		
PROGRAM / YEAR/ SEM	B.Tech (ECE)/II Year/ WINTER 2023-2024		
DATE OF SUBMISSION	03.05.2024		
TEAM MEMBERS DETAILS	**REGISTER NO.**	**NAME**	
	22BEC1001	LAKSHITA KJ	
	22BEC1051	RADHIKA P	
	22BEC1232	SWAPNA S	
	22BEC1302	S LAKSHITHA	
PROJECT TITLE	**DIGITAL DICE USING 8051 MICROCONTROLLER**		
COURSE HANDLER'S NAME	**DR.N.SUBHASHINI**	**REMARKS**	
COURSE HANDLER'S SIGN			

ABSTRACT:

This project presents the design and implementation of a digital dice made using an 8051 microcontroller kit. The digital dice is a popular electronic gadget used in various games and simulations, offering a random number generation mechanism like traditional dice. The project aims to create a compact, efficient, and user-friendly digital dice system with the flexibility to customize the number of sides on the dice. The hardware architecture of the digital dice roller comprises an 8051 microcontroller, an LCD unit, push-button switches for user input, and supporting components. The microcontroller is programmed to generate random numbers within the desired range, corresponding to the sides of the dice. The user interface allows players to roll the digital dice by pressing a button, and the generated number is displayed on the seven-segment display. This project tackles the design and construction of a digital dice using an 8051 microcontroller kit. Imagine a familiar six-sided die, but with the functionality condensed into a user-friendly electronic gadget. This digital version offers the same thrill of random number generation as its traditional counterpart, making it ideal for various games and simulations. The key to this project lies in creating a system that's compact, efficient, and adaptable.

KEYWORDS:

Microcontroller - A microcontroller, or microcontroller unit, is a small computer on a single integrated circuit. A microcontroller contains one or more CPUs along with memory and programmable input/output peripherals.

Random number generation - Random number generation is a process by which, often by means of a random number generator, a sequence of numbers or symbols that cannot be reasonably predicted better than by random chance is generated.

External Interrupts - Microcontroller 8051 consists of two external hardware interrupts: INT0 and INT1 as discussed above. These interrupts are enabled at pin 3.2 and pin 3.3. It can be level-triggered or edge-triggered.

LCD Interface - The LCD display controller provides an interface between the multimedia processor and a flat-panel display module. The controller can be integrated as a part of system on chip or can be discrete. The image rendered by the application is displayed on the screen for the user by the LCD controller.

INTRODUCTION:

The digital dice project involves the use of an 8051 microcontroller kit and the algorithm of random number generation. The digital dice use the generation of a random number, equivalent to that of rolling a dice. The regular dice used in several popular board games like Ludo, Snakes and Ladder, etc. have proven to be biased during the manufacture of the dice or their usage. Since the centre of gravity of a regular dice affects its rolling and thus has a very high possibility of being biased or becoming biased with regular usage, a digital dice would ensure fairness and an unbiased game. In contrast, the fairness of a digital dice is independent of its mass and centre of gravity, making it more reliable.

NOVELTY:

This project simulates rolling a die using assembly language and an interrupt. Imagine a register in the processor like a constantly spinning wheel with six sides labelled 1 to 6. An assembly language program keeps incrementing a value in this register, making it cycle through 1 to 6 repeatedly. This creates a pseudo-random effect, mimicking the unpredictable nature of a die roll.

An interrupt acts like someone pressing a button to roll the die. When the interrupt occurs, the program stops whatever it's doing and checks the current value in the register. This value represents the "rolled" number, which is then displayed on an LCD screen, just like the result of a real die roll.

In essence, the project uses a clever trick with a continuously changing register to generate a sequence that appears random, and the interrupt acts as the trigger to capture a specific value from this sequence and display it, simulating the act of rolling a die.

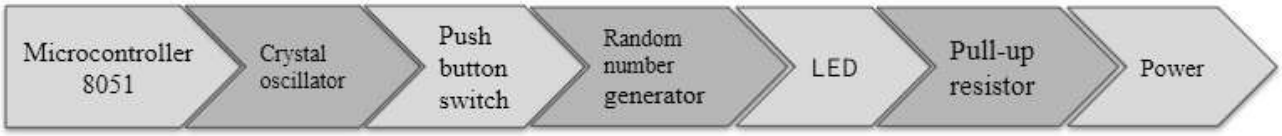
The already existing circuits for digital dice using arduino UNO is not compact and can be biased. Whereas as the logic we use for random number generation makes sure that the results is unbiased.

COMPONENTS/ SOFTWARE REQUIRED:

Hardware Required: Microcontroller 8051 development board, LCD board.

Software Required: Keil U Vision5

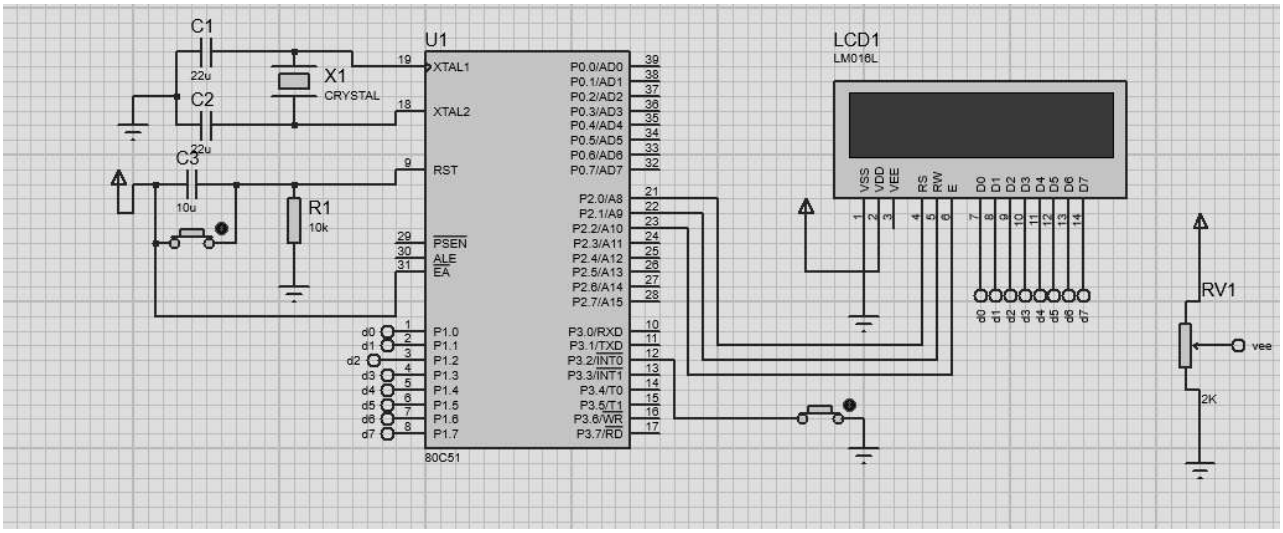
BLOCK DIAGRAM:



The push button switch is connected to one of the external interrupt pins of the 8051 microcontroller.

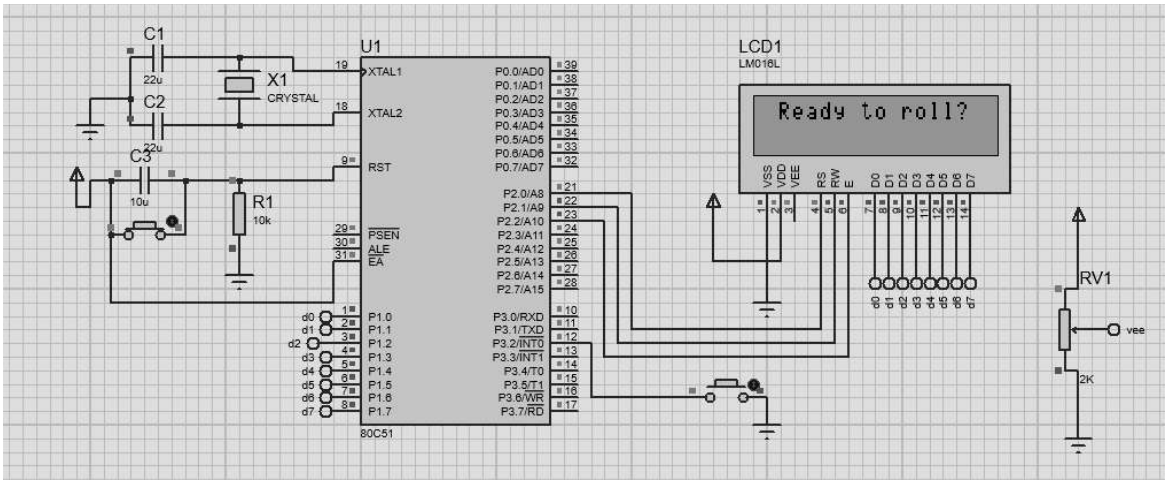
- Hence when the button is pressed, an interrupt is generated in the microcontroller and the corresponding interrupt service routine is invoked.
- The ISR in turn does the process of checking for the random number that is in storage in the register at that point of time and moves the same into the accumulator, hence to the port of the 8051 that is connected to the LCD.
- This value is then displayed to the users on the LCD.

SCHEMATIC USED IN PROTEUS SIMULATION:

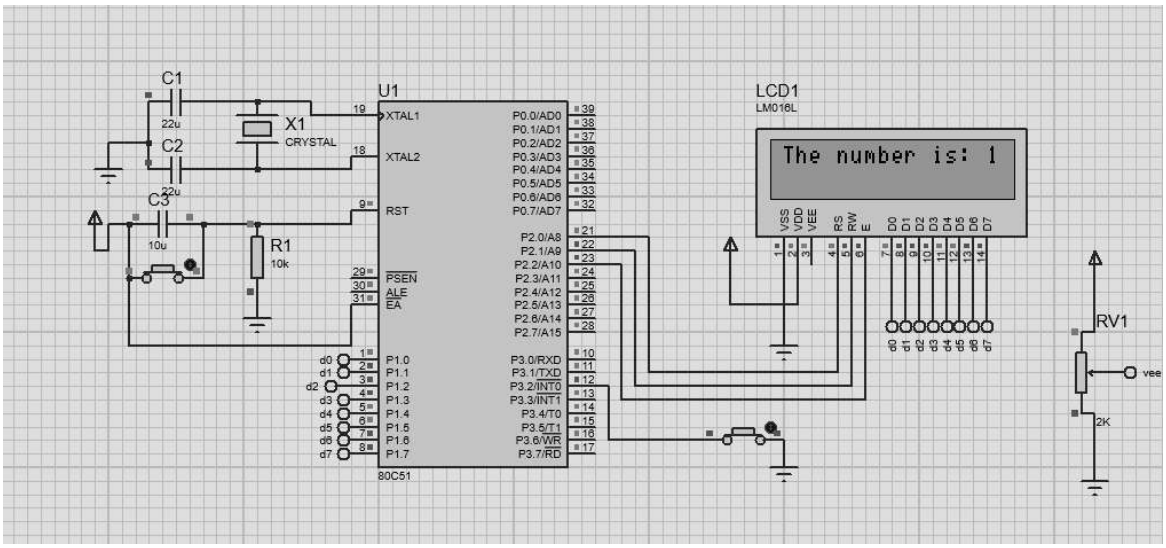


Output:

When the INT0 is pressed to roll the dice:

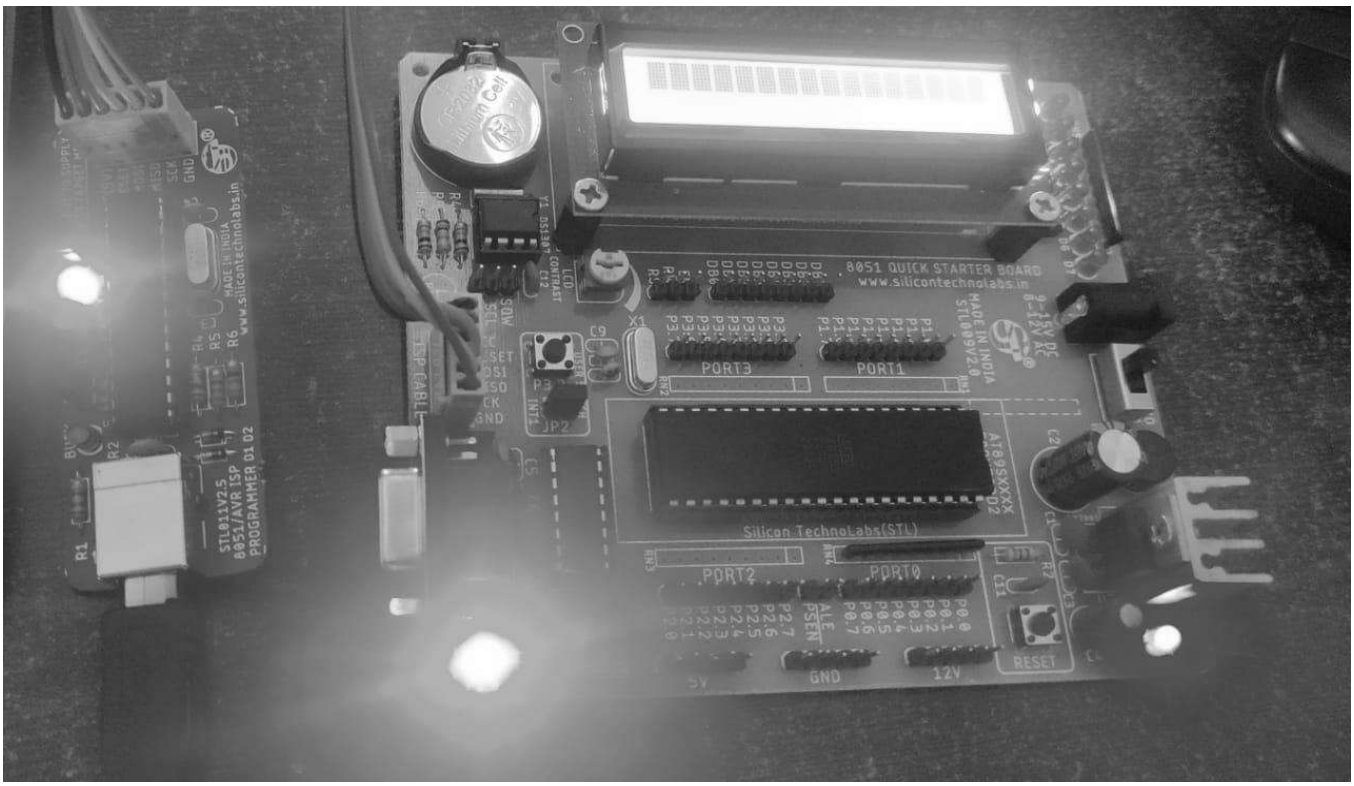


The number obtained after rolling the die appears on the LCD:



HARDWARE IMPLEMENTATION:

We use a random number generating code to generate numbers from 1 to 6, like a dice. We later use an interrupt function to move the number into the LCD to display it. When the interrupt push button is pressed, the interrupt service routine is called. The number saved in the register is moved to the accumulator and then to the LCD. We create a delay program to create a delay between the display of numbers to make it easier for the players to view the displayed number. We use DPTR to display an introduction.



ASSEMBLY LANGUAGE PROGRAM:

```
ORG 0000H

LJMP MAIN

ORG 0003H ; External Interrupt INT0 is used

LJMP THROW

ORG 0030H

MAIN: MOV P1,#00H

MOV IE,#81H
```

```
ACALL LCD_INITIALISE
MOV DPTR, #CODE1
INTRO: CLR A
MOVC A, @A+DPTR
MOV P1,A
ACALL DATAWRT
ACALL DELAY
INC DPTR
JZ NEXT1
SJMP INTRO
NEXT1: NOP
MOV R1,#01H
RANDOM:
CJNE R1,#06H,NEXT
MOV R1,#01H
NEXT: INC R1
SJMP RANDOM
LCD_INITIALISE: MOV A,#38H; the code to initialise the LCD
ACALL COMNWRT
ACALL DELAY
MOV A,#0EH
ACALL COMNWRT
ACALL DELAY
MOV A,#01H
ACALL COMNWRT
ACALL DELAY
```

```
MOV A, #06H
ACALL COMNWRT
ACALL DELAY
MOV A,#81H
ACALL COMNWRT
ACALL DELAY
RET
CODE1: DB 'Ready to roll?', 0
ORG 0300H
THROW: CJNE R1,#01H,CHECK2
MOV DPTR, #CODE2
LOOP1: CLR A
MOVC A, @A+DPTR
MOV P1,A
ACALL DATAWRT
ACALL DELAY
INC DPTR
JZ NEXTT1
SJMP LOOP1
NEXTT1:MOV P1,#'1'
ACALL DATAWRT
ACALL DELAY1
MOV A, #01H
ACALL COMNWRT
ACALL DELAY
LJMP EXIT
```

```
CHECK2: CJNE R1,#02H, CHECK3
MOV DPTR, #CODE2
LOOP2: CLR A
MOVC A, @A+DPTR
MOV P1,A
ACALL DATAWRT
ACALL DELAY
INC DPTR
JZ NEXT2
SJMP LOOP2
NEXT2: MOV P1,#'2'
ACALL DATAWRT
ACALL DELAY1
MOV A, #01H
ACALL COMNWRT
ACALL DELAY
LJMP EXIT
CHECK3: CJNE R1,#03H, CHECK4
MOV DPTR, #CODE2
LOOP3: CLR A
MOVC A, @A+DPTR
MOV P1,A
ACALL DATAWRT
ACALL DELAY
INC DPTR
JZ NEXT3
```

```
LJMP LOOP3

NEXT3: MOV P1,#'3'

ACALL DATAWRT

ACALL DELAY1

MOV A, #01H

ACALL COMNWRT

ACALL DELAY

LJMP EXIT

CHECK4: CJNE R1, #04H, CHECK5

MOV DPTR, #CODE2

LOOP4: CLR A

MOVC A, @A+DPTR

MOV P1,A

ACALL DATAWRT

ACALL DELAY

INC DPTR

JZ NEXT4

LJMP LOOP4

NEXT4: MOV P1, #'4’

ACALL DATAWRT

ACALL DELAY1

MOV A, #01H

ACALL COMNWRT

ACALL DELAY

LJMP EXIT

CHECK5: CJNE R1,#05H, CHECK6
```

```
MOV DPTR, #CODE2
LOOP5: CLR A
MOVC A, @A+DPTR
MOV P1,A
ACALL DATAWRT
ACALL DELAY
INC DPTR
JZ NEXT5
LJMP LOOP5
NEXT5: MOV P1, #'5’
ACALL DATAWRT
ACALL DELAY1
MOV A, #01H
ACALL COMNWRT
ACALL DELAY
LJMP EXIT
CHECK6: MOV DPTR, #CODE2
LOOP6: CLR A
MOVC A, @A+DPTR
MOV P1,A
ACALL DATAWRT
ACALL DELAY
INC DPTR
JZ NEXT6
LJMP LOOP6
NEXT6: MOV P1, #'6’
```

```
ACALL DATAWRT

ACALL DELAY1

MOV A, #01H

ACALL COMNWRT

ACALL DELAY

EXIT: NOP

RETI

CODE2: DB 'The number is:', 0

;We shall assume P1 is the port connected to the LCD.

;P2.0 is Reset, P2.1 is R/W, P2.2 is Enable

COMNWRT: MOV P1, A

CLR P2.0

CLR P2.1

SETB P2.2

ACALL DELAY

CLR P2.2

RET

DATAWRT: SETB P2.0

CLR P2.1

SETB P2.2

ACALL DELAY

CLR P2.2

RET

DELAY: MOV R3, #50

HERE1: MOV R4, #255

HERE2: DJNZ R4, HERE2
```

```
DJNZ R3, HERE1

RET

DELAY1:MOV R0,#100

MOV TMOD, #01H

AGAIN: MOV TL0, #00H

MOV TH0, #00H

SETB TR0

HERE: JNB TF0, HERE

CLR TR0

CLR TF0

DJNZ R0, AGAIN

RET

END
```

APPLICATIONS:

This project demonstrates concepts like random number generation, LED interfacing, and user input handling. It has the potential to replace physical dice in board games with a compact and portable solution. This can be particularly useful for games requiring multiple dice or where lost dice are a concern. We can utilize the digital dice as a random number generator for various purposes. This could involve educational simulations, simple decision-making in games, or even creating a basic random number lottery system. The project's flexibility allows for creating dice with more than six sides, catering to niche games or simulations requiring a wider range of random numbers. The project ensures the fairness of the dice.

CONCLUSION:

In summary, the project develops an unbiased digital dice by using a simple yet novel way of generating a random number and displays it upon calling of the interrupt, imitating the rolling of a dice, to then display that number through an LCD to the user.

The technique used in this project is a success as the number generation is based only upon the calling of an interrupt which depends on an external user input which is random. Thus the die this symbolises is purely unbiased and applicable for a variety of online games and other uses.

Scope for future work lies in developing a random number generator that does not work in the cycling fashion of moving from 1 to 6

This can be a predictable process and thus lead to manipulation of the game. An alternative process would generate the number without following such an order.

REFERENCES:

- **https://www.researchgate.net/publication/375094945_Random_Number_Generators_Principles_and_Applications**
- **https://www.researchgate.net/publication/348578518_Digital_Dice_using_555_Timer**