Report
Digital Concentration Tracker
"Focus Flow"

1.Team Member:

- Alaa Khaled Taha
- Amira ElSayed El-Demerdash

2. Objective

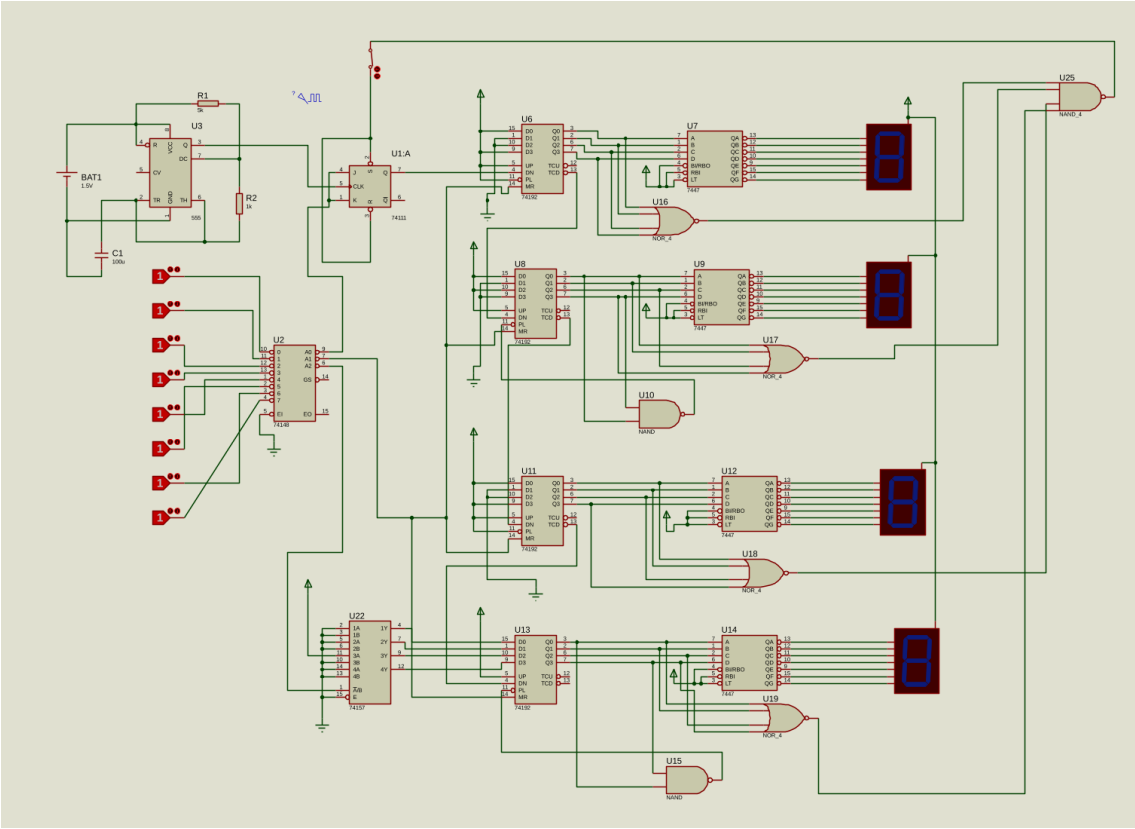
The objective of this project is to build a digital study timer using logic gates and digital components such as counters, decoders, flip-flops, and timer555.

The system allows a user to switch between two modes:50 minutes of study with 10 minutes break, then the timer starts counting down from the selected time, stops automatically at 00:00, and waits for a restart signal to begin the break timer.

3. Used Components :

555 Timer
74LS76: JK flip-flop
4 × 74LS192: BCD down counters
4 × 74LS47: BCD to 7-segment decoders
7-Segment Displays
74LS157 (MUX)
74LS148 (encoder)
74LS00 (NAND)
74LS25 (4-input NOR)

4. Circuit Diagram:



5. Project Overview:

The study timer operates using a combination of digital logic components, including BCD counters, multiplexers, decoders, flip-flops, and a 555 timer for generating clock pulses.

A 555 timer IC provides clock pulses that trigger the down-counting operation.

A JK flip-flop is used to memorizing the session mode (study session or break session), when the user pauses the session.

The counters count down in real-time from the loaded value (49:59 or 09:59) toward 00:00.

The BCD outputs from the counters are connected to BCD-to-seven-segment decoders, which drive the segment displays.

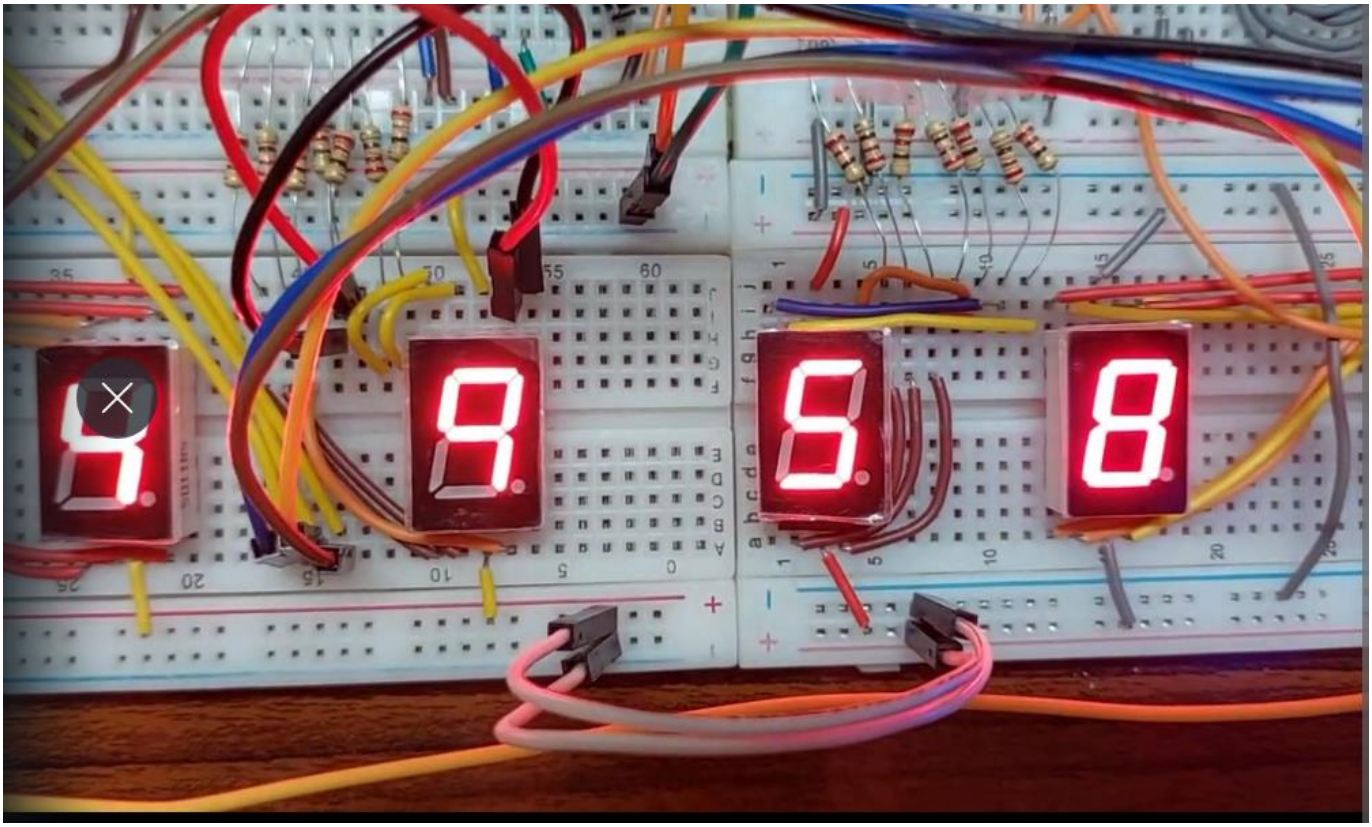
The system displays time using 4 seven-segment displays: 2 digits for minutes and 2 digits for seconds ,to allow accurate representation of the countdown from values like 49:59 or 09:59.

The user begins by selecting a mode: either 50 minutes of study or 10 minutes of break. This is done using a switch connected to a multiplexer. The selected value is loaded into the counters, The multiplexer loads a preset value into the counters based on the selected mode.

When the counter reaches zero, a logic detection circuit using NOR and NAND gates detects the state and enables the break period through flip-flop control.

To begin a new session, the user must press a reset or restart button, which reloads the preset value and resumes counting.

6. Prototype



7. Challenges and Solutions

1) If The BCD to 7-segment decoder is connected directly with the 7-segment without current limiting resistors, this may damage the display due to excessive current.

- **Solution**: We used current-limiting resistors (220Ω) between the 7447 outputs and the 7-segment segments to protect the LEDs inside the display and ensure appropriate brightness.

2)Since We used a JK flip-flop to divide the frequency by 2, we needed the 555 timer to generate a 2Hz square wave, so the output of the flip-flop would be exactly 1Hz, to match the requirement of the clock.

- **Solution:** We carefully selected the resistor and capacitor values connected to the 555 timer to produce a 2Hz output. Then, we fed this signal to the JK flip-flop, which effectively divided the frequency in half, resulting in a stable 1Hz signal **.**

3) The J-K flip-flop we used requires proper logic at the Set and Reset inputs to function correctly. If S or R are not set properly the flip-flop might stay stuck or not toggle as expected. To allow the user to start and stop the timer

- **Solution**: We ensured that the Set and Reset pins to be high so the flip-flop can operate in toggle mode using the clock and the J-K inputs. We connected them to VCC to make sure they're always active, allowing the J-K logic to function correctly when the timer is running. This was essential to get the timer to toggle only when needed.

4)The second and fourth counters in our timer should not count to 9. For example, the second digit on seconds should count only from 5 to 0 and then go back to 5. Similarly, the fourth digit of minutes might need to repeat from 4 to 0. But the 74192 counter naturally counts from 0 to 9.

- **Solution**: We used NAND gates to detect when the count reaches 6 in the second counter or 5 at the fourth counter. When that condition is met, the NAND gate sends a signal to reset the counter to 0 by triggering the Load or Clear pins appropriately. This way, we enforced the required maximum values manually using external logic.

5) After the counters reach 00:00, we needed a way to automatically stop the timer so that it doesn't continue counting or wrap around.

- **Solution**: We designed logic circuitry to detect when all counters show zero, then used (Nor and Nand logic gates) to disable the clock signal going into the counters and flip-flop. This ensures that once the countdown finishes, the system freezes at 00:00 and waits for a manual reset or restart.