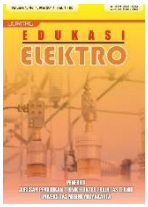
Volume 07, No. 01, May 2023, pages 44 - 52

JEE

Jurnal Edukasi Elektro

https://journal.uny.ac.id/index.php/jee



Quiz Bell Design Using NodeMCU ESP8266 for Quiz Contest at IAIN Pontianak

Havizul[1](✉), Zulkarnain[1]

[1] Institut Agama Islam Negeri Pontianak

havizul@gmail.com

Abstract—Quiz Buzzer is one of important elements in a quizz contest. In Tadris Matematika Study Program's Birthday event, one of contest held is quiz contest for High School Student in Kalbar Province. Hereby, author is going to design the buzzer circuit which is able to be used in that quiz contest. This circuit was build by some components, namely push button, buzzer, indicating light, transistor switching circuit for driving the buzzer and indicating light, and a NodeMCU ESP8266 as the system controller. Design of hardware was conducted by referring to literature studies, basic theory, and datasheet. Testing of developed codes was conducted over the Quiz Buzzer Protoyping that is assembled on the breadboard. Testing results showed that the quiz buzzer circuit working properly as needed and the design of 3 teams quiz buzzer.

Keywords: Quiz Bell, Smart Quiz Buttons, Arduino Quiz Bell, ESP8266

Article submitted March 15th, 2023. Resubmitted May 8th, 2023. Final acceptance May 31st, 2023.

1 Introduction

A Quiz Contest (Elementary School / Junior High School) or Smart & Accurate Contest (Senior High School) program in the 1980s was divided into three rounds, namely a round of questions for each team, a round of throwing questions from teams that could not answer to other teams, and a round of scramble. In that quiz contest, a special button is needed to ring the bell and light the indicator lamp on precisely and accurately to determine which team had pressed the button first at almost the same time. The problem was found in this quiz contest is that the committee needs a wireless-based special quiz-contest button that can ring the buzzer and light the indicator lamp on precisely and accurately. As stated by Ref [1].

The study which was conducted by Ref [2] aims to design and build Arduino-based and computer interfaces-based quiz-contest bells for operated by 5 teams. At a time only one button can be activated. For a display device, we use an LCD monitor and an quiz-contest codes that was developed using Visual Basic. This quiz-contest bell was developed to assist the quiz-contest jury in determining which team had pressed the quiz-contes button first.

As stated in Ref [3], the quiz technique is a reward group-based cooperative learning model, by dividing students in the class into several small groups, and giving rewards to students who successfully answer questions. The implementation of learning using the quiz technique has problems when the jury will determine which team presses the button first compared to the other teams in the scramble questions session, so a quiz bell with a high degree of accuracy is needed. The Quiz bell is designed using an AC relay component, a 220Vac indicator light, and a 220Vac buzzer.

The Quiz-Contest Bell or the Quiz-Contest Button, with this device, jury can determine which participant or team is entitled to answer questions during the scramble round. In the scramble round, when one of the teams or competitors has pressed the quiz-contest button, the other participant's buttons will not function until the system is reset again by the jury. In addition, the bell or buzzer will also ring and the indicator lamp of related team will light up. How important this device is in the quiz contest event, because it has a very decisive function.

At IAIN Pontianak, the Tadris Mathematics Study Program is carrying out a quiz competition between schools in West Kalimantan in order to celebrate the 2nd anniversary in 2022. The author took the initiative to design a quiz bell circuit that use NodeMCU ESP8266, with 3 teams buttons, 1 jury button, 3 indicator lamps, and a buzzer.

2 Methods

This research was conducted through 4 phases, namely needs identification, design, development, and testing & revision. Date and location of the research was conducted in December 2022 at IAIN Pontianak. In the design phase, a blue-print of the quiz-contest bell circuit scheme was developed. In development phase, the Arduino code was developed using the Arduino IDE 1.8.13 up to the compiling step to get a compiled (*.hex) file. The testing phase is carried out by using a prototype of NodeMCU-based quiz bell which is assembled on a breadboard, if deficiencies or errors are found during this testing phase or the device that has been developed is not as needed, then a revision is made. These steps can be seen in Figure 1.

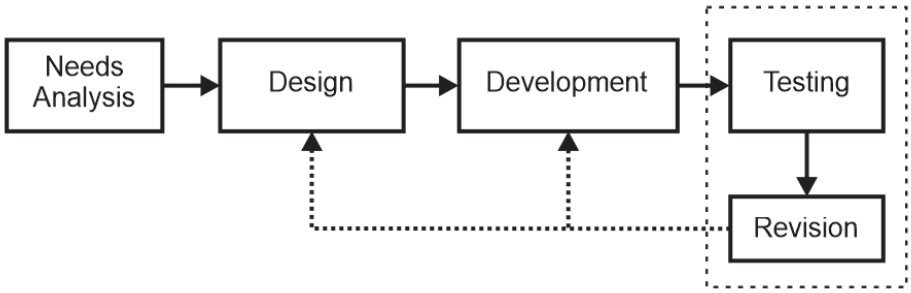


Fig. 1. Quiz bell device development model

3 Results and Discussions

The working principle of the quiz bell is that if one of the teams / participants presses the button first, the other team / participant's quiz button will not work, the indicator lamp will light up, and the bell / buzzer will be ringing. Furthermore, the jury has its own button (reset button), where when it was pressed, all participant buttons will not function, the indicator lamp will light off, and the buzzer will be muted / inactive. This working principle can also be expressed in the truth table of table 1, where PBx = Push Button (LCC), Lx = Indicator Lamp, and BZ = Buzzer. In the table, 1 means true, while 0 means false.

Table 1. The truth table of quiz bell working principle

	PB1	PB2	PB3	PBR	L1	L2	L3	BZ
Team A	1	0	0	0	1	0	0	1
Team B	0	1	0	0	0	1	0	1
Team C	0	0	1	0	0	0	1	1
Jury	0	0	0	1	0	0	0	0

In Figure 2 you can see the full scheme of the Arduino-based Quiz Bell Circuit. It shows that the buttons for participants use the notations of PB1, PB2, and PB3. While the jury button (reset button) uses the notation of PR1.

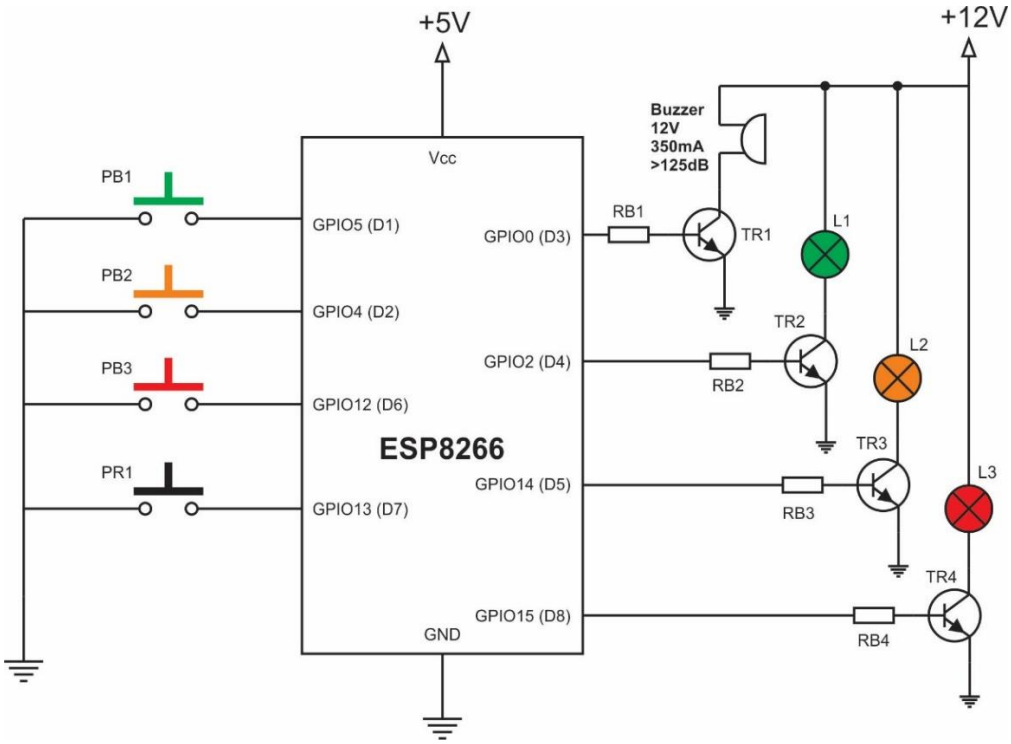


Fig. 2. Arduino-based quiz bell circuit

The circuit in this push button element (participant and jury buttons) uses the internal pull-up resistors which is available in the ESP8266 nodeMCU. While the output is configured as active-high to activate the base of the NPN transistor which will drive L1, L2, L3, and the buzzer. Refer to [4], the ESP8266 has 17 GPIO pins which can be used for various functions by programming the appropriate registers. On the other hand, as explained by [5] that the GPIO0-GPIO15 pins when configured as inputs can activate internal pull-ups, whereas specifically for pin GPIO16 (XPD_DCDC) they can only activate internal pull-downs. In general, these GPIO pins can be configured as input or output. Then, another literature source quoted from [6] confirms that some of the GPIO pins of the ESP8266 can be used as input or output (I/O) safely, some others are not recommended for use as I/O, while the rest can be used for special functions. GPIO pins that are safe to use as inputs with active internal pull-up resistors are GPIO4, GPIO5, GPIO12, GPIO13, and GPIO14. While the GPIO pins that are safe to use as outputs with both active-high and active-low configurations are GPIO0, GPIO2, GPIO4, GPIO5, GPIO12, GPIO13, and GPIO14. As for several other GPIO pins, some conditions must be considered as described in the datasheet when they will be used as an input or output. Furthermore, other literature sourced from [7] explains that the ESP8266 has 17 GPIO pins, but only 11 pins can be used, because 6 pins (GPIO6-GPIO11) are used to connect to flash memory chips. If we try to use these 6 pins as I/O, there will be a risk of a crash in the program. Based on those literatures, in the design of this quiz bell pins GPIO4, GPIO5, GPIO12, and GPIO13 are used as push button inputs with the built-in pull-up resistor configuration activated. As for the pins that function as outputs (to light the indicator lamp up and ringing the buzzer), GPIO0, GPIO2, GPIO14, and GPIO15 pins are used.

On the output side, there are two types of loads, they are buzzer and the indicator light. These two types of loads are driven by the transistor circuit (TR1-TR4) which are known as transistor switching circuit. The first load (buzzer) requires a current of 350 mA, so the TR1 specification must

have an Ic capability of ≥ 350 mA. In the design, so that the transistor is not burdened with the maximum working current, the value of Ic ≥ 2x350 mA is applied. Based on the electrical characteristics table in [5] and an explanation of Digital I/O voltage and current restrictions by [7], that each ESP8266 GPIO pin is capable of carrying a maximum current of 12 mA, with a minimum voltage of 2.5 Volts, typical 3.3 Volts and a maximum of 3.6 Volts. Therefore, in this design so that the ESP8266 is not burdened with maximum working current, the GPIO0 (D3) current will be limited to ≤ 0.5x12 mA. So here the βdc value of TR1 can be calculated through the following formula:

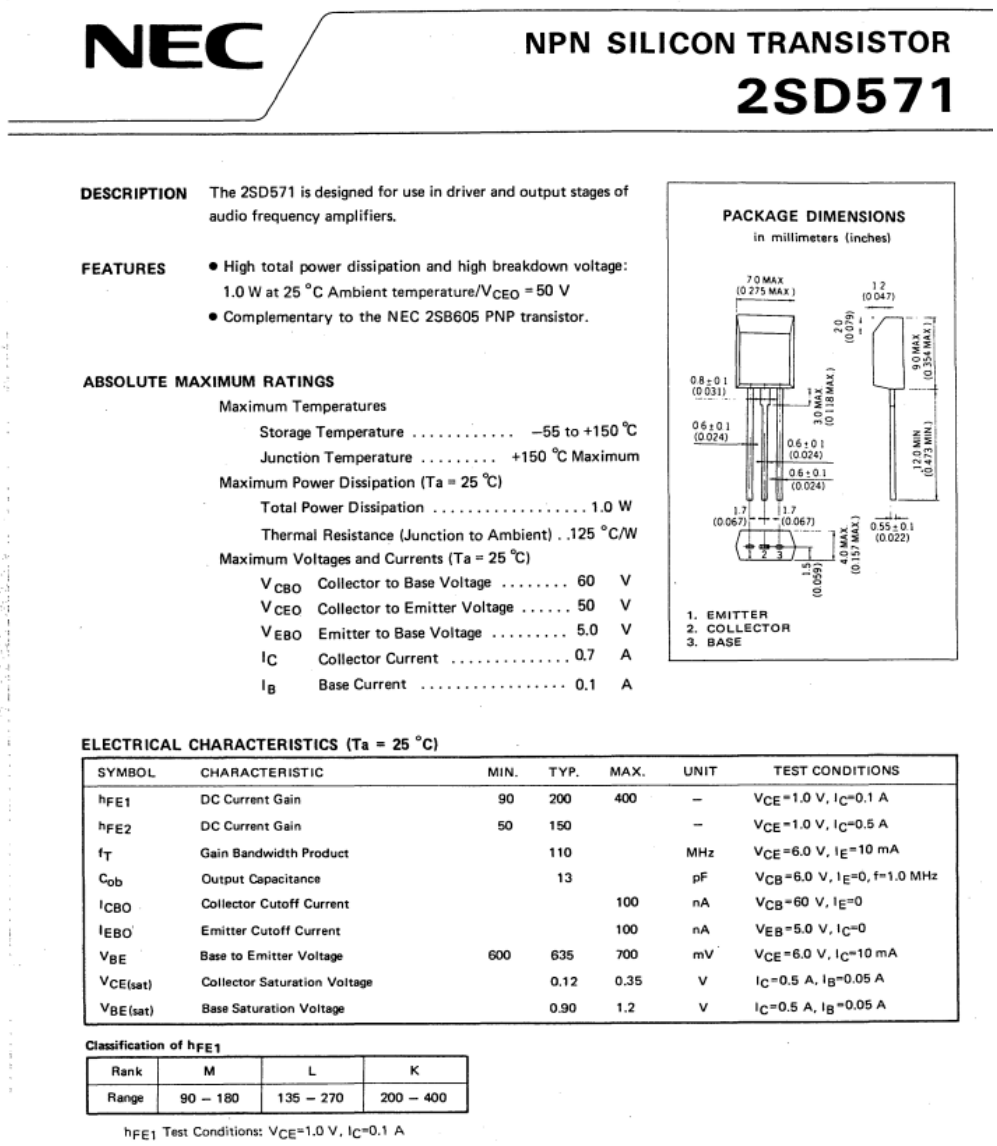
$$\beta dc \geq \frac{I_c}{I_b} \geq \frac{700\ mA}{6\ mA} \geq 116 \quad (1)$$

Based on the above parameters, there are many types of transistors that can be used, one of them is the 2SD571 NPN transistor. This transistor has a typical βdc value of 150 mA at Ic = 500 mA, as shown in the datasheet (figure 3). The maximum Ic that can be handled is 700 mA, it fits to the needs of this design. Furthermore, the value of RB1 can be calculated with the following formula:

$$R_b = \frac{(V_{cc} - V_{be})}{I_b} = \frac{(12V - 0{,}7V)}{6mA} = 1{,}88\ Kohm \quad (2)$$

So, the resistance value available for Rb which is close to 1800 Ohm (1K8). The values above, especially Ib, are in accordance with the parameters listed in the datasheet. On the other hand, as explained by [8] that in the design line, to ensure the transistor that functions as a switch works properly at the saturation point, Ib is made 10 times greater than Ib(sat). If you want to follow this design line, then the base current that must be flowed by the GPIO0 pin should be 60 mA, then this will cause the ESP8266 to be damage. As a solution, if you still want to follow that design line, you can change the type of transistor that has a βdc value ≥ 1160 or by using a Darlington transistor circuit.

Furthermore, to determine the type of transistor TR2-TR4 and its base resistance (Rb), the method used is the same as for determining the value of the parameters in TR1 above, depending on the load specifications (L1, L2, and L3). In this design, the indicator lamp requires a current of +- 75 mA, still low enough while compared to the Ic(max) range of the 2SD571 transistor which is 0.7 A or 700 mA. Therefore, the value and circuit configuration on TR1 also applicable for used by TR2-TR4.



NEC

NPN SILICON TRANSISTOR

2SD571

DESCRIPTION The 2SD571 is designed for use in driver and output stages of audio frequency amplifiers.

FEATURES

- High total power dissipation and high breakdown voltage: 1.0 W at 25 °C Ambient temperature/V_{CEO} = 50 V
- Complementary to the NEC 2SB605 PNP transistor.

ABSOLUTE MAXIMUM RATINGS

Maximum Temperatures

Storage Temperature ..... −55 to +150 °C

Junction Temperature ..... +150 °C Maximum

Maximum Power Dissipation (Ta = 25 °C)

Total Power Dissipation ..... 1.0 W

Thermal Resistance (Junction to Ambient) . .125 °C/W

Maximum Voltages and Currents (Ta = 25 °C)

V_{CBO} Collector to Base Voltage ..... 60 V

V_{CEO} Collector to Emitter Voltage ..... 50 V

V_{EBO} Emitter to Base Voltage ..... 5.0 V

I_C Collector Current ..... 0.7 A

I_B Base Current ..... 0.1 A

PACKAGE DIMENSIONS

in millimeters (inches)

1. EMITTER
2. COLLECTOR
3. BASE

ELECTRICAL CHARACTERISTICS (Ta = 25 °C)

SYMBOL	CHARACTERISTIC	MIN.	TYP.	MAX.	UNIT	TEST CONDITIONS
h_{FE1}	DC Current Gain	90	200	400	–	V_{CE}=1.0 V, I_C=0.1 A
h_{FE2}	DC Current Gain	50	150		–	V_{CE}=1.0 V, I_C=0.5 A
f_T	Gain Bandwidth Product		110		MHz	V_{CE}=6.0 V, I_E=10 mA
C_{ob}	Output Capacitance		13		pF	V_{CB}=6.0 V, I_E=0, f=1.0 MHz
I_{CBO}	Collector Cutoff Current			100	nA	V_{CB}=60 V, I_E=0
I_{EBO}	Emitter Cutoff Current			100	nA	V_{EB}=5.0 V, I_C=0
V_{BE}	Base to Emitter Voltage	600	635	700	mV	V_{CE}=6.0 V, I_C=10 mA
$V_{CE(sat)}$	Collector Saturation Voltage		0.12	0.35	V	I_C=0.5 A, I_B=0.05 A
$V_{BE(sat)}$	Base Saturation Voltage		0.90	1.2	V	I_C=0.5 A, I_B=0.05 A

Classification of h_{FE1}

Rank	M	L	K
Range	90 – 180	135 – 270	200 – 400

h_{FE1} Test Conditions: V_{CE}=1.0 V, I_C=0.1 A

Fig. 3. Datasheet of transistor 2SD571

After the design of the hardware side is complete, then proceed with the design on the software side. At this stage, 2 outputs are generated, namely the algorithm of the program in visual model (flowchart) and program source code. Figure 4 shows the Quiz Bel source code algorithm in visual model (flowchart).

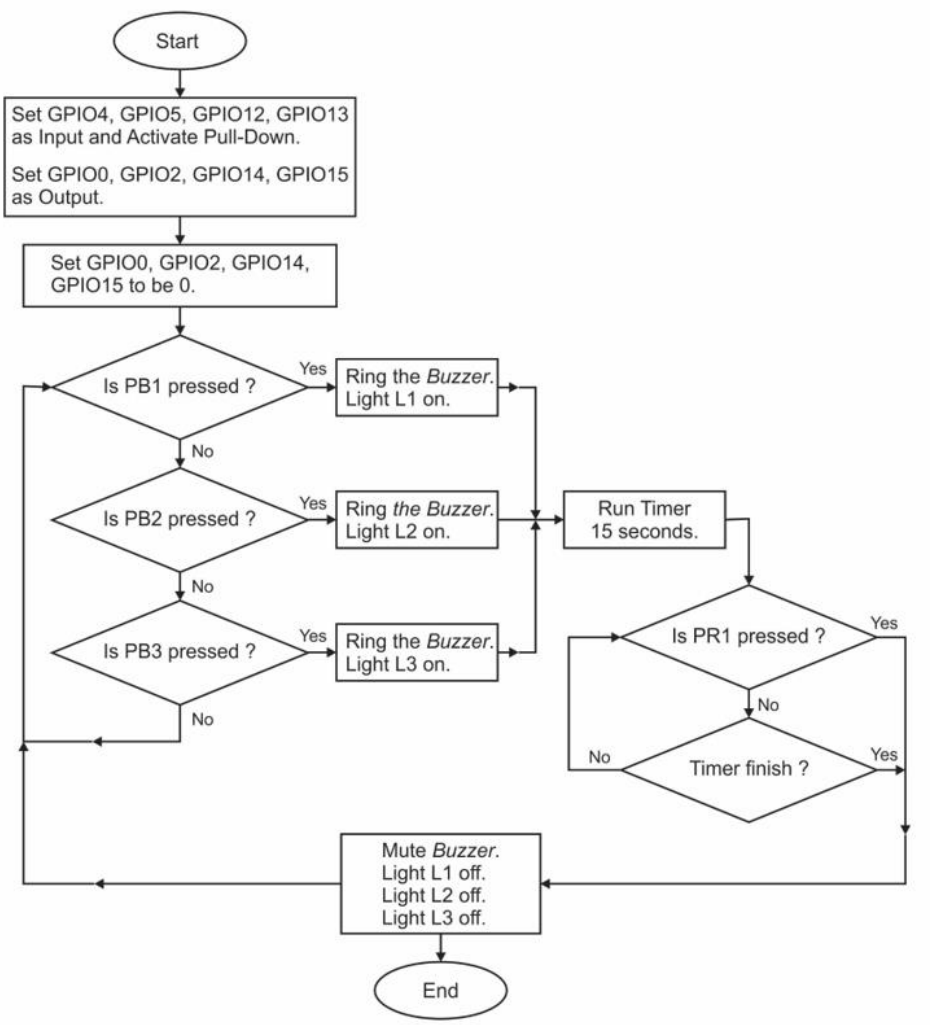


Fig. 4. Quiz bel source code algorithm

This algorithm is then translated into the Arduino programming language, so it will produce the following program code.

```
//constants won't change. They're used
//here to set pin numbers:
const int PB1 = 5;     //GPIO5
const int PB2 = 4;     //GPIO4
const int PB3 = 12;    //GPIO12
const int PR1 = 13;    //GPIO13

const int Buzzer = 0; //GPIO0
const int L1 = 2;      //GPIO2
const int L2 = 14;     //GPIO14
const int L3 = 15;     //GPIO15
//variables will change:
int pbState = 1;   // variable for reading the pushbutton status
int prState = 1;
void setup() {
```

```
  //initialize PB & PR as an input:
  pinMode(PB1, INPUT_PULLUP);
  pinMode(PB2, INPUT_PULLUP);
  pinMode(PB3, INPUT_PULLUP);
  pinMode(PR1, INPUT_PULLUP);
  //initialize Buzzer & L1/L2/L3 pin
  //as an output:
  pinMode(Buzzer, OUTPUT);
  pinMode(L1, OUTPUT);
  pinMode(L2, OUTPUT);
  pinMode(L3, OUTPUT);
}
void loop()
{
  if (digitalRead(PB1) == 0)
  {
    digitalWrite(Buzzer, 1);
    digitalWrite(L1, 1);
    waitPR();
  }
  else if (digitalRead(PB2) == 0)
  {
    digitalWrite(Buzzer, 1);
    digitalWrite(L2, 1);
    waitPR();
  }
  else if (digitalRead(PB3) == 0)
  {
    digitalWrite(Buzzer, 1);
    digitalWrite(L3, 1);
    waitPR();
  }
}
void waitPR()
{
  //Buzzer ringing time duration is 15 seconds
  unsigned long maxDurasiBuzzerOn =
  millis() + 15000;
  while (millis() < maxDurasiBuzzerOn)
  {
    if (digitalRead(PR1) == 0)
    {
       break;
    }
    //delay 1/10 seconds so that
    //looping while will be executed properly
    delay(100);
  }
  digitalWrite(Buzzer, 0);
  digitalWrite(L1, 0);
  digitalWrite(L2, 0);
  digitalWrite(L3, 0);
}
```

The last phase is to test the code that has been written by uploading it to the Quiz Bel prototype device. From the results of the tests that have been conducted, the Quiz Bel prototype device can work properly according to its function as designed in Table 1 or Figure 4. The testing process that has been carried out can be seen in Figure 5.

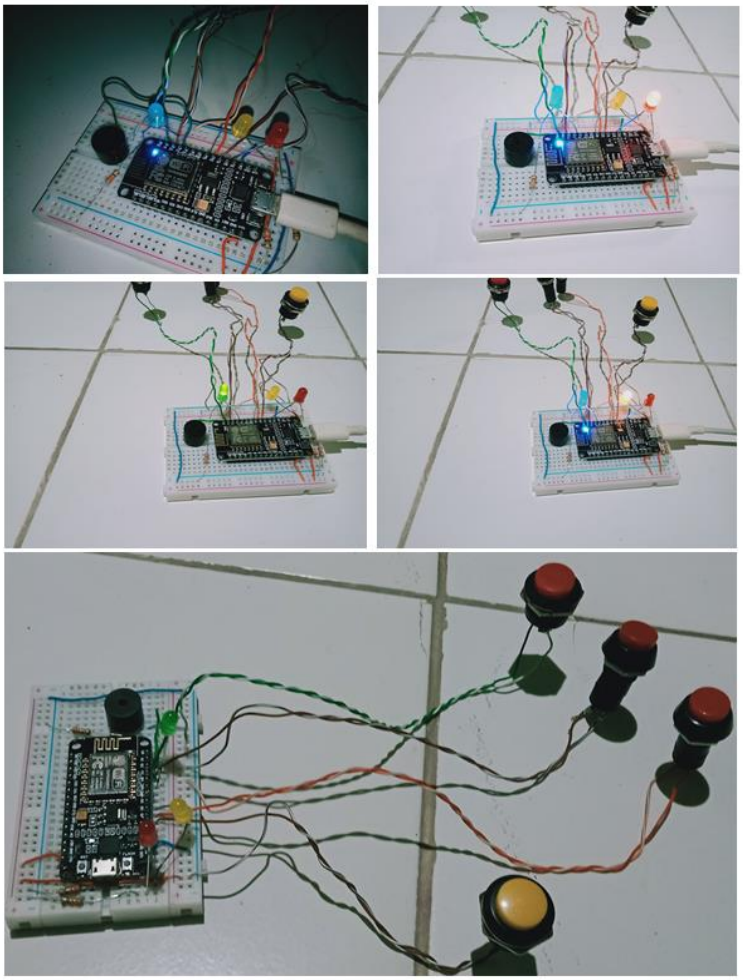


Fig. 5. Program code testing through quiz bell prototype

4 Conclusion

The design of the Quiz Bell using NodeMCU ESP8266 in this paper has been successfully tested. On the hardware side, the design is carried out by referring to the results of literature studies, basic theory, and datasheets in order to produce a Quiz Bell circuit design that meets the standards. On the software side, Quiz Bell software development was carried out, starting from developing program algorithms, developing program code, compiling program code, untill uploading program code to ESP8266.

The test results show that the Quiz Bell prototype can function properly according to the design that was made at the initial phase, namely in accordance with the functional rules of the Quiz Bell in table 1, as well as the program flow diagram (figure 4) which is a representation of the working principle hardware specifically.

This NodeMCU ESP8266 based Quiz Bell has a WiFi feature which is useful for further development of the device. For example, Design of Quiz Bell that are based on Wireless. On a wireless-

based, more than three participant buttons (quiz buttons) can be expanded, this is because as many quiz buttons can be made according to the quality of the Access Point and the WiFi signal covered, there is no longer the problem of GPIO limitations.

5 Acknowledgment

On December 27th, 2022, the Tadris Mathematics Study Program at IAIN Pontianak held several competitions to celebrate the 2nd Anniversary. Held at Abdul Rani Hall starting at 08.00 WIB until finished. One of the competitions held was the Quiz Contest for high school students at the same level in West Kalimantan. The Quiz Contest was attended by SMAN 1 Pontianak, SMAN 2 Pontianak, SMAN 8 Pontianak, SMAN Kesehatan Pontianak, Madrasah Aliyah Imaduddin Pontianak, and Madrasah Aliyah Al-Hikmah Sambas. From this activity the author came up with an idea to design an Arduino-based Quiz Bell, which is also the outcome of a needs analysis phase which is the initial phase of a research.

The author would like to express his deepest gratitude to all fellow lecturers who have participated in this event, so that both directly and indirectly have helped the author to participate and contribute through this simple writing. First of all, the author would like to thank Mr. Zulkarnain, S.Sc., M.Pd as Head of Tadris Matematika Study Program, Mrs. Desty Septianawati, M.Pd as Secretary of Tadris Mathematika Study Program, Mrs. Yumi Sarassanti, M.Pd as Mathematics Education Lecturer, Mrs. Hidayu Sulisti, S.Sc, M.Pd as a Mathematics Education Lecturer, Ms. Komala-sari, M.Pd as an Information Technology Lecturer, and to all Lecturers in Tadris Matematika Study Program which the author cannot mention one by one, as well as to all students who have helped to carry out this event, which directly or indirectly contributed to the writing of this paper.

6 Daftar Pustaka

[1] G. Y. Kurniawan, "Tombol Cerdas Cermat Wireless Berbasis Arduino Nano," Prodi DIII Teknik Elektronika Jurusan Teknologi dan Kejuruan Universitas Pendidikan Ganesha, Singaraja, 2020.
[2] F. Wahyudi and S. Winardi, "Rancang Bangun Bel Cerdas Cermat Berbasis Arduino dan Interfacing Komputer," Prodi Sistem Komputer Fakultasi Ilmu Komputer Universitas Narotama, Surabaya, 2016.
[3] A. B. Sudibya, "Pengembangan Media Pembelajaran Buzzer Quiz Sebagai Alat Bantu Pada Teknik Cerdas Cermat," Fakultas Keguruan dan Ilmu Pendidikan Universitas Sebelas Maret, Surakarta, 2016.
[4] Espressif System, ESP8266 Technical Reference, China: www.espressif.com, 2017.
[5] Espressif Systems, ESP8266EX Datasheet, China: www.espressif.com, 2022.
[6] Random Nerd Tutorials, "ESP8266 Pinout Reference: Which GPIO pins should you use?," 6 May 2019. [Online]. Available: https://randomnerdtutorials.com/esp8266-pinout-reference-gpios/.
[7] Pieter, "A Beginner's Guide to the ESP8266," 2017.
[8] A. P. Malvino, Prinsip-Prinsip Elektronik, 2 ed., Jakarta: Erlangga, 1981.
[9] Components101, "NodeMCU ESP8266," 22 April 2020. [Online]. Available: https://components101.com/development-boards/nodemcu-esp8266-pinout-features-and-datasheet.
[10] NEC, "Transistor NPN Vceo:50V Ic:700mA Pc:1W 110MHz TO-92," 2023. [Online]. Available: https://www.pilianidis.gr/images/sync_products/01-2SD571/2SD571.pdf. [Accessed 16 Jan 2023].

7 Authors

Havizul, S.T., M.Pd is a Lecturer at the Institut Agama Islam Negeri Pontianak, in the Mathematics Tadris Study Program, Faculty of Tarbiyah and Teacher Training, whose address is at Jl. Letjend Suprapto No. 14, Continent of Malay Land, Kec. Pontianak Sel., Pontianak City, West Kalimantan 78243.

Zulkarnain, S.Sc., M.Pd is a Lecturer at the Institut Agama Islam Negeri Pontianak, in the Mathematics Tadris Study Program, Faculty of Tarbiyah and Teaching Science.