

מחולל מספרים אקראיים עם ESP32

1. מבוא

מטרת מאמר זה היא להסביר כיצד ליצור מספרים אקראיים ב- ESP32 באמצעות סביבת העבודה של הארדואינו. במונחים של חומרה ל- ESP32 יש **מחולל מספרים אקראיים אמיתיים**, כלומר הערכים המתקבלים ממנו הם באמת אקראיים ולא מספרים המתבססים על סדרות מתמטיות או הרצה אקראית של מונים. המספרים האקראיים האמיתיים נוצרים בהתבסס על הרעש של תת מערכות ה- WiFi / RF Bluetooth שנמצאים בג'וק, מה שאומר שאם ה- Bluetooth וה- WiFi מושבתים, אז רק מספרים פסאודו אקראיים נוצרים. מספרים פסאודו אקראיים - **Pseudorandom numbers** - הם מספרים הנוצרים על ידי אלגוריתם במחשב המייצר סדרת מספרים או אלמנטים המדמים בקירוב תכונות של רצף אקראי. היכולת של ESP32 ליצור מספרים אקראיים אמיתיים חשובה מאוד מאחר שניתן להשתמש בהם לפעולות הצפנה. דבר מעניין שיש להזכיר בנוגע למחולל המספרים האקראיים ESP32 הוא שדגימת נתונים של 2 ג'יגה-בייט המתקבלת ממנו עם WiFi מופעל עברה את כל הבדיקות של Testsuite Number Random Dieharder שהיא חבילת בדיקות עבור מחוללי מספרים אקראיים.

על בדיקות אלו ניתן לקרוא בקישור: [Robert G. Brown's General Tools Page \(duke.edu\)](http://duke.edu/~brown/gt/)

ניתן לקרוא על יצירת מספרים האקראיים במאמר הנהדר הזה.

[ESP32 \(10\) – Random number generator – lucadentella.it](https://lucadentella.it/2018/01/10/esp32-random-number-generator/)

2. התוכנית

בתחילת התוכנית נאתחל את התקשורת הטורית עם המוניטור של הארדואינו בפונקציית ה- `setup()` כדי לראות את ההדפסות שנרצה להציג.

```
void setup()
```

```
{
```

```
    Serial.begin(115200);
```

```
}
```

לאחר מכן נקבל ונדפיס בצורה מחזורית את המספרים האקראיים בפונקציית הלולאה הראשית (`loop()`).

על מנת לקבל מספר אקראי, אנו יכולים להשתמש בפונקציית `esp_random` המוגדרת בקישור הבא:

[arduino-esp32/tools/sdk/include/esp32/esp_system.h](https://github.com/espressif/arduino-esp32/blob/master/tools/sdk/include/esp32/esp_system.h) at

[39fb8c30440a4abd5fe0e2c87609ba6798ae8013 · espressif/arduino-esp32 · GitHub](https://github.com/espressif/arduino-esp32/blob/master/tools/sdk/include/esp32/esp_system.h)

פונקציה זו איננה מקבלת ארגומנטים ומחזירה ערך אקראי בין 0 ל- `UINT32_MAX` (שהוא הערך הגדול ביותר

שיכול להכיל `int` לא מסומן של 32 ביטים שהוא 4,294,967,295). עם זאת, כדאי לשים לב שכפי שהוזכר בסעיף המבוא,

הערך יהיה אקראי באמת רק אם מערכת ה-WiFi או מערכת ה-RF Bluetooth פועלות. הפקודה שנרשום היא:

```
Serial.println(esp_random());
```

כלומר להדפיס למוניטור הטורי את הערך שהתקבל מהפונקציה `esp_random()` .

כחלופה לפונקציה הזו אנו יכולים להשתמש בפונקציה `random()` של ה `Arduino` המיושמת גם בעבודה עם סביבת העבודה של הארדואינו . הפונקציה `random()` היא **overloaded** - מספר פונקציות עם שם זהה והפונקציה שעובדת היא לפי המספר ו/או סוג הארגומנטים השונים שמעבירים אליה) . לפונקציה `random()` ניתן לקרוא על ידי העברת פרמטר קלט אחד או שניים. במקרה שאנו מעבירים רק פרמטר אחד , אנו מציינים את הגבול העליון של המספר האקראי שנוצר (לא כולל המספר עצמו!) , לכן התוצאה תהיה מספר בין 0 לבין הערך של הפרמטר פחות 1. לדוגמה : `Serial.println(random(10));` תציג במסך הטורי מספרים בין 0 ל 9 . אם נקרא לפונקציה `random` על ידי העברת שני פרמטרים אז הראשון יהיה הגבול התחתון של המספר האקראי שנוצר (כולל המספר עצמו) והשני יהיה הגבול העליון (לא כולל המספר עצמו). לדוגמה : `Serial.println(random(10,20));` תציג במסך הטורי מספרים בין 10 ל 19 .

כדאי לשים לב ששתי הגרסאות של הפונקציה `random()` קוראות לפונקציה `esp_random` במימושן, כפי שניתן לראות בקישור הבא :

[arduino-esp32/cores/esp32/WMath.cpp at 49f35ff070fcb8c82a11866ae6f81bab49dba6e3 ·](https://github.com/arduino/arduino-esp32/blob/master/cores/esp32/WMath.cpp#L49f35ff070fcb8c82a11866ae6f81bab49dba6e3)

[espressif/arduino-esp32 · GitHub](https://github.com/espressif/arduino-esp32)

לפיכך, השיקולים לגבי הצורך בחיבור WiFi או מערכת ה `RF Bluetooth` לקבלת מספרים שנוצרו באופן אקראי באמת חלים גם הם.

הערה : לא ניתן לשלוח לפונקציה `esp_random()` ערכים . אם ננסה לשלוח ערך אחד או שניים נקבל שגיאת קומפילציה :

`too many arguments to function 'uint32_t esp_random()'`

כלומר שלחנו יותר מידי ארגומנטים לפונקציה.

ניתן לראות את הקוד המלא הסופי להלן. הוספנו הדפסה נוספת לקריאות טובה יותר והשהיה של שנייה אחת בין כל איטרציה של לולאת הארדואינו.

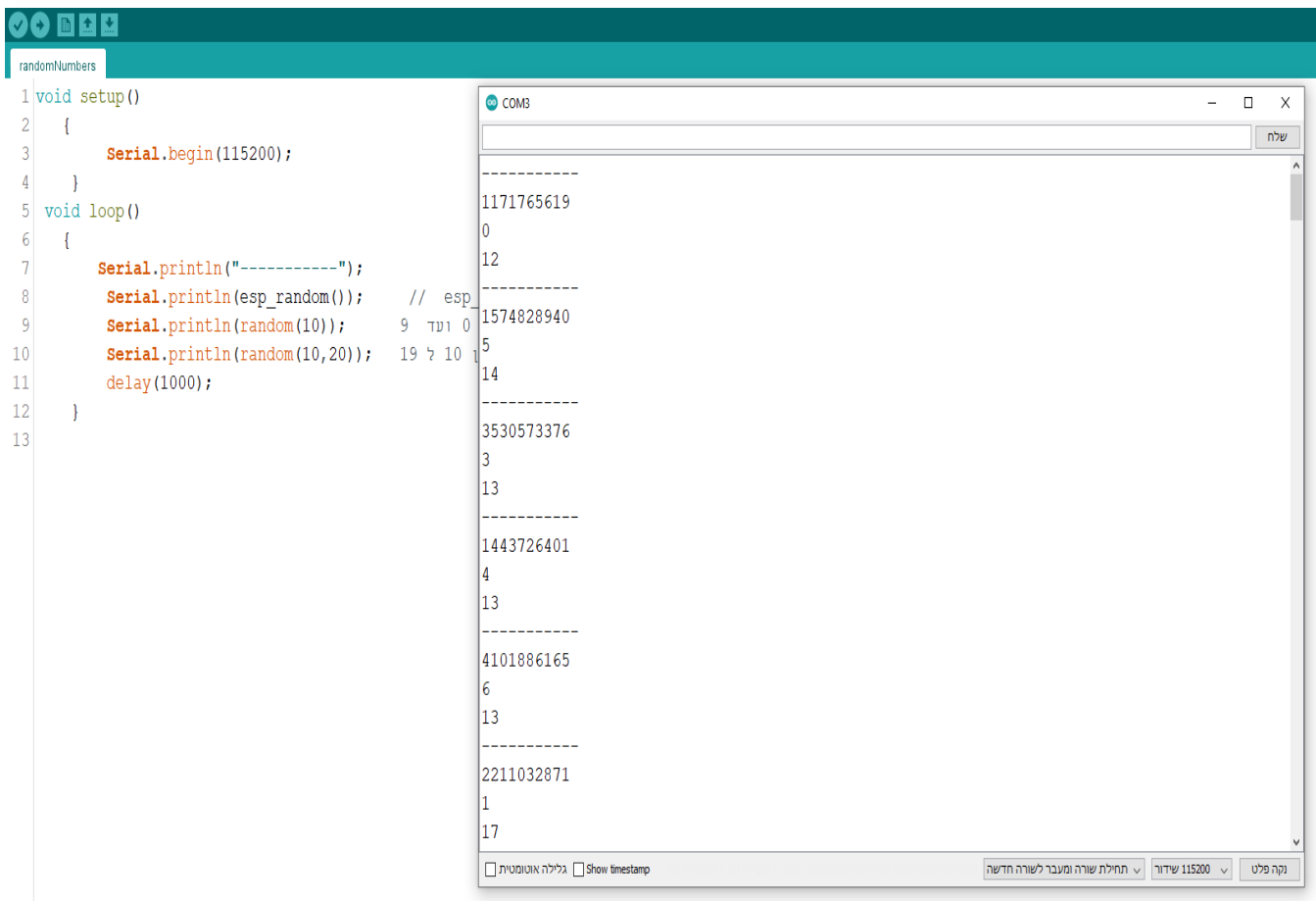
```
void setup()
{
  Serial.begin(115200);
}

void loop()
{
  Serial.println("-----");
```

```
Serial.println(esp_random()); // esp_random( ) לקבלת הערך המתקבל מהפונקציה  
Serial.println(random(10)); // לקבלת מספרים מ 0 ועד 9  
Serial.println(random(10,20)); // לקבלת מספרים בין 10 ל 19  
delay(1000);  
}
```

3. בדיקת התוכנית

לבדיקת התוכנית נבצע העלאה - upload של התוכנית לכרטיס ה esp32 שלנו. לאחר מכן נפתח את הצג הטורי של ה Arduino IDE ונבדוק את התוצאות המודפסות. הן צריכות להיות בדומה לאיור הבא המראה את המספרים האקראיים שנוצרו .



איור 1 : מחולל מספרים אקראיים המתקבלים בתצוגת המוניטור הטורי

באיור רואים מספר קבוצות של 3 שורות עם הדפסות. נסתכל על הקבוצה העליונה באיור :
ההדפסה הראשונה מתקבלת מהפונקציה `esp_random()`. בדוגמה העליונה התקבל המספר 1171765619
ההדפסה השנייה מתקבלת מהפונקציה `random(10)` שבו ביקשנו מספר בין 0 ל 9 וקיבלנו 0 .
ההדפסה השלישית הייתה עם הפונקציה `random(10,20)` כלומר מספר בין 10 ל 19 וקיבלנו 12 .

4. ביבליוגרפיה

א.

http://espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf

ב.

<https://www.arduino.cc/reference/en/language/functions/random-numbers/random/>

ג.

https://techtutorialsx.com/2017/12/22/esp32-arduino-random-number-generation/#google_vignette

ד.

https://github.com/espressif/arduino-esp32/blob/39fb8c30440a4abd5fe0e2c87609ba6798ae8013/tools/sdk/include/esp32/esp_system.h

ה.

<https://docs.espressif.com/projects/esp-idf/en/latest/esp32/api-reference/system/random.html>

www.arikporat.com