

گزارش پروژه کاربرد کامپیوتر در کنترل

موضوع پروژه: سیستم حضور و غیاب RFID

افراد گروه: فاطمه چنگیزیان 9332682

سپیده نصراللهی 9332673

بهار 1397

مقدمه

تکنولوژی RFID یا شناسایی با امواج رادیویی، تکنولوژی ارتباط کوتاه بردی است که با آن می توان افراد یا اشیای مختلف را به وسیله کارت ها برچسب ها یا همان تگ (Tag) های RFID شناسایی کرد. این دستگاه ها امروزه کاربرد بسیار گسترده ای در زندگی ما دارند. برای مثال تمامی کارت های هوشمند مترو ، کارت های مکانیزه غذا خوری ها ، کارت های دانشجویی ، دزدگیر های فروشگاه های ، کارت های اتوبوس ، کارت های شناسایی ورود و خروج و ... همگی از تکنولوژی RFID بهره می برند.

-نحوه کار دستگاه های RFID به زبان ساده:

دستگاه های RFID می توانند از روی کارت ها و تگ ها بخوانند و بنویسند. روی کارت ها و تگ ها یک تراشه قرار دارد که بر روی آن اطلاعاتی نوشته شده است. هم دستگاه و هم کارت های RFID سیم پیچی در ساختار خود دارند. وقتی یک دستگاه کارت خوان در کنار یک کارت یا تگ قرار می گیرد، اطلاعات منحصر به فرد نوشته شده روی کارت (مخلوطی از کاراکتر و عدد) را از روی آن می خواند. این تراشه ها به واسطه القای الکترومغناطیسی روی سیم پیچ ها انرژی مورد نیاز خودشان را به دست می آورد ! در واقع سیم پیچ ها (Coil) هایی که روی هر دو قسمت دستگاه کارت خوان و کارت قرار دارد. وقتی این دو را به هم نزدیک می کنید ، سیم پیچ روی دستگاه ، جریان برقی را روی سیم پیچ کارت القا می کند و به همین سادگی کارت روشن می شود! تراشه طوری طراحی شده است که به محض روشن شدن شروع به مخابره داده های ذخیره شده در حافظه اش ، با امواج رادیویی می کند و دستگاه کارت خوان این اطلاعات را دریافت کرده و به شکل ارتباط سریال (Serial) آن را به جایی که شما می خواهید منتقل می کند. هر دستگاه RFID که خریداری بکنید ، فراغ از جنس و نام و نشانش دارای پایه های خروجی اطلاعات به شکل سریال است . ما می توانیم این اطلاعات را به وسیله آردوینو ، رزبری پای ، میکرو کنترلر های AVR، میکرو کنترلر های ARM و ... از دستگاه کارت خوان تحویل گرفته و هر کاری که در نظر داریم با اطلاعات انجام دهیم .

در این پروژه قادر خواهیم بود تا با استفاده از آردوینو و ماژول کارت خوان RFID ، یک دستگاه RFID جهت کنترل حضور و غیاب و ثبت ورود و خروج و ساعات کاری افراد بسازیم. از این دستگاه می توانید در کلاس های درس، محل کار و ... استفاده کنید. هر شاگرد یا کارمند یک تگ مخصوص دارد که برای ثبت حضور باید آن را جلوی دستگاه ریدر نگه دارد.

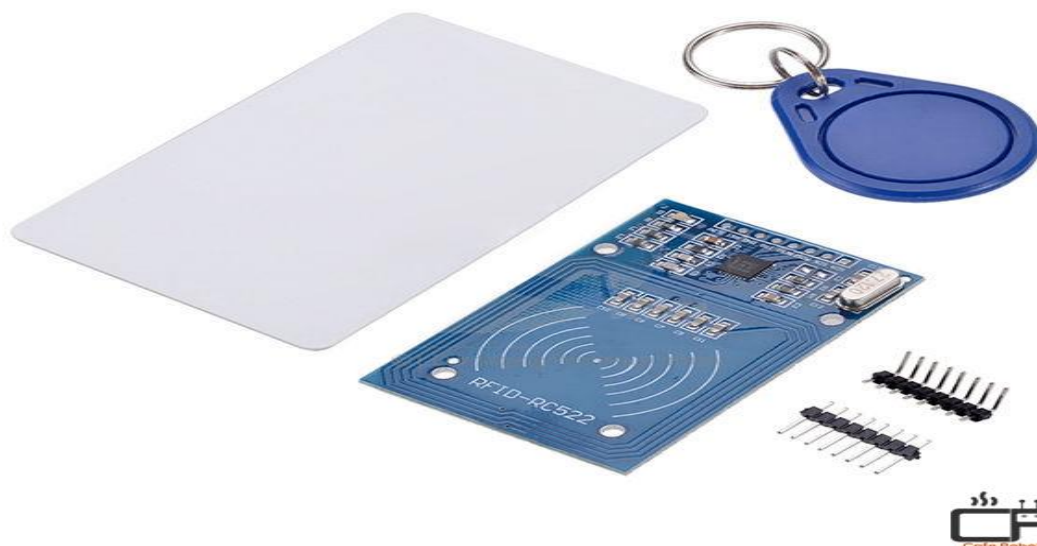
گام اول: ابزار و قطعات مورد نیاز

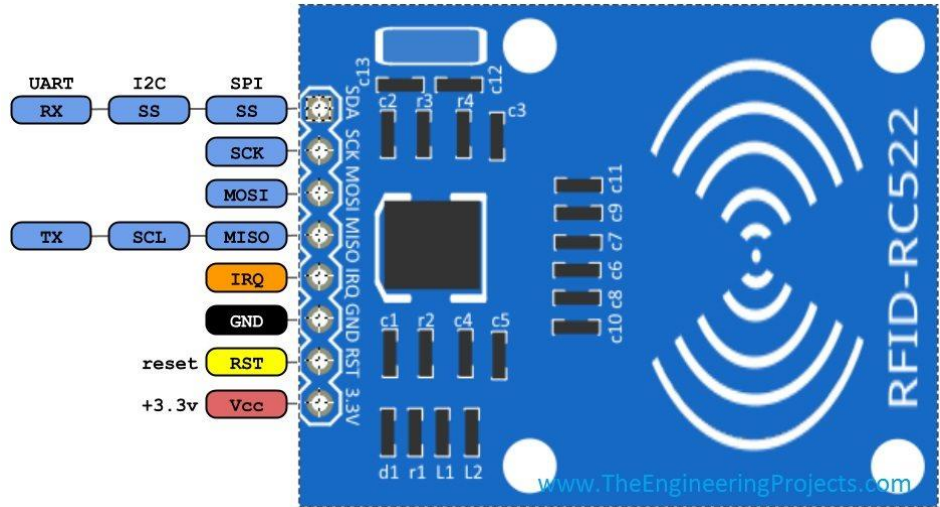
- استفاده از Arduino uno به عنوان پردازنده اصلی



- استفاده از ماژول RFID (MF-RC522): *RFID-RC522*

- استفاده از تگ ها و کارت های هم فرکانس با گیرنده RFID



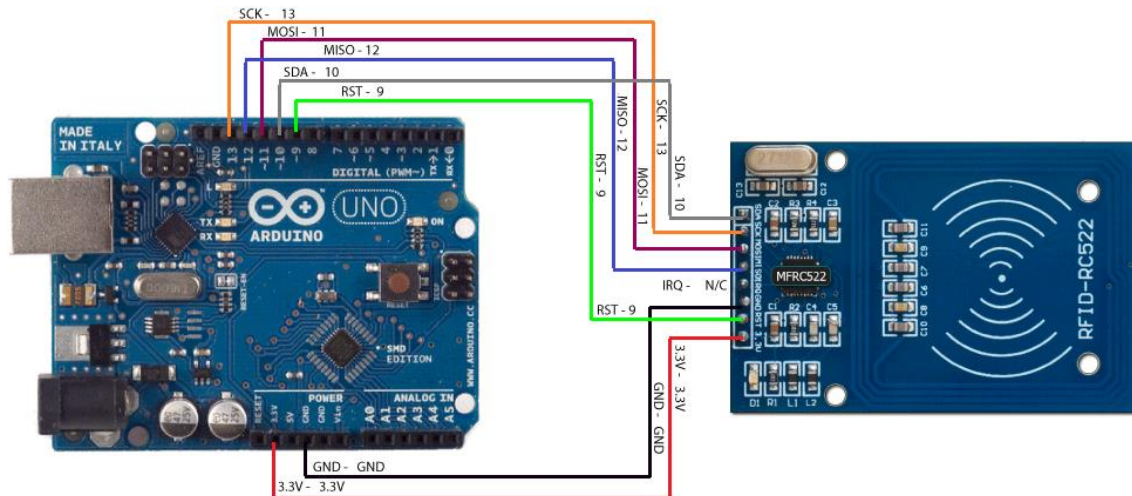


-برد برد

-سیم جامپر برد برد

گام دوم : اتصالات سخت افزار

تصویر پایین اتصالات بین ماژول ها را نشان می دهد.



Vcc همان منطق 3.3 ولت است

rst این پایه می تواند به دلخواه به هر یک از پایه های آزاد دیجیتال وصل شود و در بخش برنامه نویسی باید آن پایه define شود.

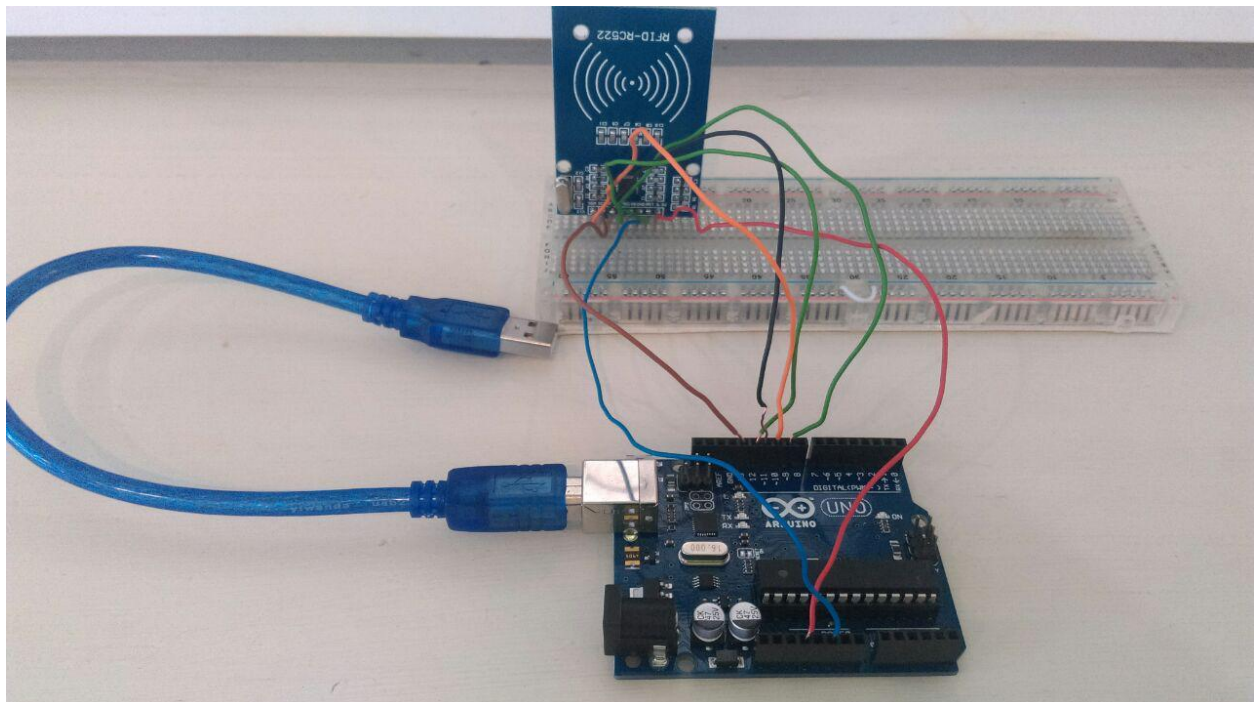
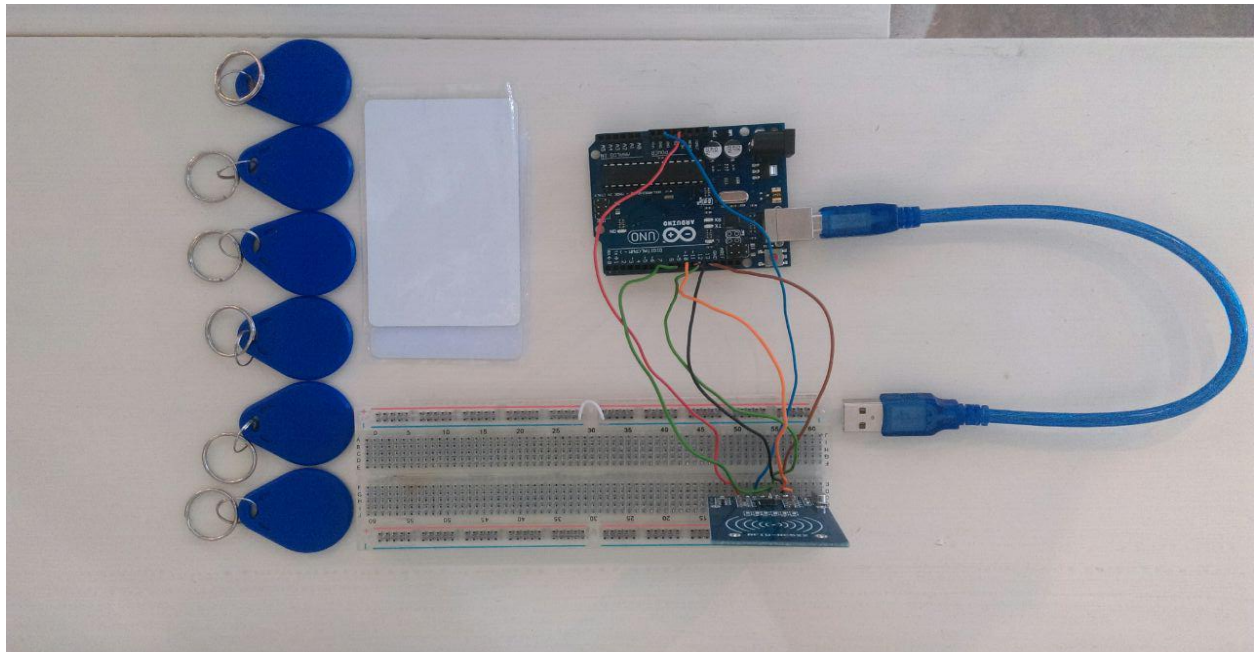
Gnd

Nss این پایه همان ss یا cs یا chip select است که این پایه هم میتواند به یک پایه دیجیتال دلخواه آزاد وصل شود و در برنامه نویسی define شود.

Irq استفاده نمی شود.

Pin	Wiring to Arduino Uno
SDA	Digital 10
SCK	Digital 13
MOSI	Digital 11
MISO	Digital 12
IRQ	unconnected
GND	GND
RST	Digital 9
3.3V	3.3V

در زیر تصویر واقعی از مدار را مشاهده میکنید



گام سوم: کد آردوینو

برای مشاهده ID تگ های RFID از کدهای زیر استفاده میکنیم:

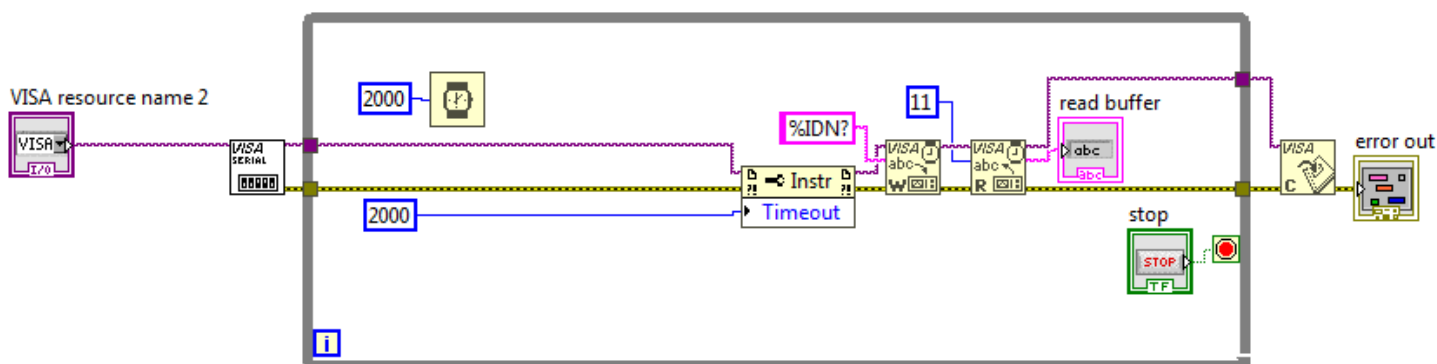
```
Arduino IDE - rfidtest | Arduino 1.8.5
File Edit Sketch Tools Help

rfidtest $
1 #include "SPI.h"
2 #include "MFRC522.h"
3
4 #define SS_PIN 10
5 #define RST_PIN 9
6 #define SP_PIN 8
7
8 MFRC522 rfid(SS_PIN, RST_PIN);
9
10 MFRC522::MIFARE_Key key;
11
12 void setup() {
13   Serial.begin(9600);
14   SPI.begin();
15   rfid.PCD_Init();
16 }
17
18 void loop() {
19   if (!rfid.PICC_IsNewCardPresent() || !rfid.PICC_ReadCardSerial())
20     return;
21
22   // Serial.print(F("PICC type: "));
23   MFRC522::PICC_Type piccType = rfid.PICC_GetType(rfid.uid.sak);
24   // Serial.println(rfid.PICC_GetTypeName(piccType));
25
26   // Check is the PICC of Classic MIFARE type
27   if (piccType != MFRC522::PICC_TYPE_MIFARE_MINI &&
28       piccType != MFRC522::PICC_TYPE_MIFARE_1K &&
29       piccType != MFRC522::PICC_TYPE_MIFARE_4K) {
30     Serial.println(F("Your tag is not of type MIFARE Classic."));
31     return;
32   }
33
34   String strID = "";
35   for (byte i = 0; i < 4; i++) {
36     strID +=
37       (rfid.uid.uidByte[i] < 0x10 ? "0" : "") +
38       String(rfid.uid.uidByte[i], HEX) +
39       (i!=3 ? " " : "");
40   }
41   strID.toUpperCase();
42
43   Serial.println(strID);
44
45   rfid.PICC_HaltA();
46   rfid.PCD_StopCrypto1();
47 }
48 }

Done uploading.
Sketch uses 6598 bytes (20%) of program storage space. Maximum is 32256 bytes.
Global variables use 293 bytes (14%) of dynamic memory, leaving 1755 bytes for local variables. Maximum is 2048 bytes.

43 Arduino/Genuino Uno on COM11
```

گام چهارم: کد لب ویو (labview)



خواندن از پورت سریال:

ابتدا اطلاعات روی پورت com که آردوینو به آن وصل است خوانده می شود.

(درواقع شماره ی تگی که به مدار RFID نزدیک شده است) و سپس در read buffer ریخته می شود.

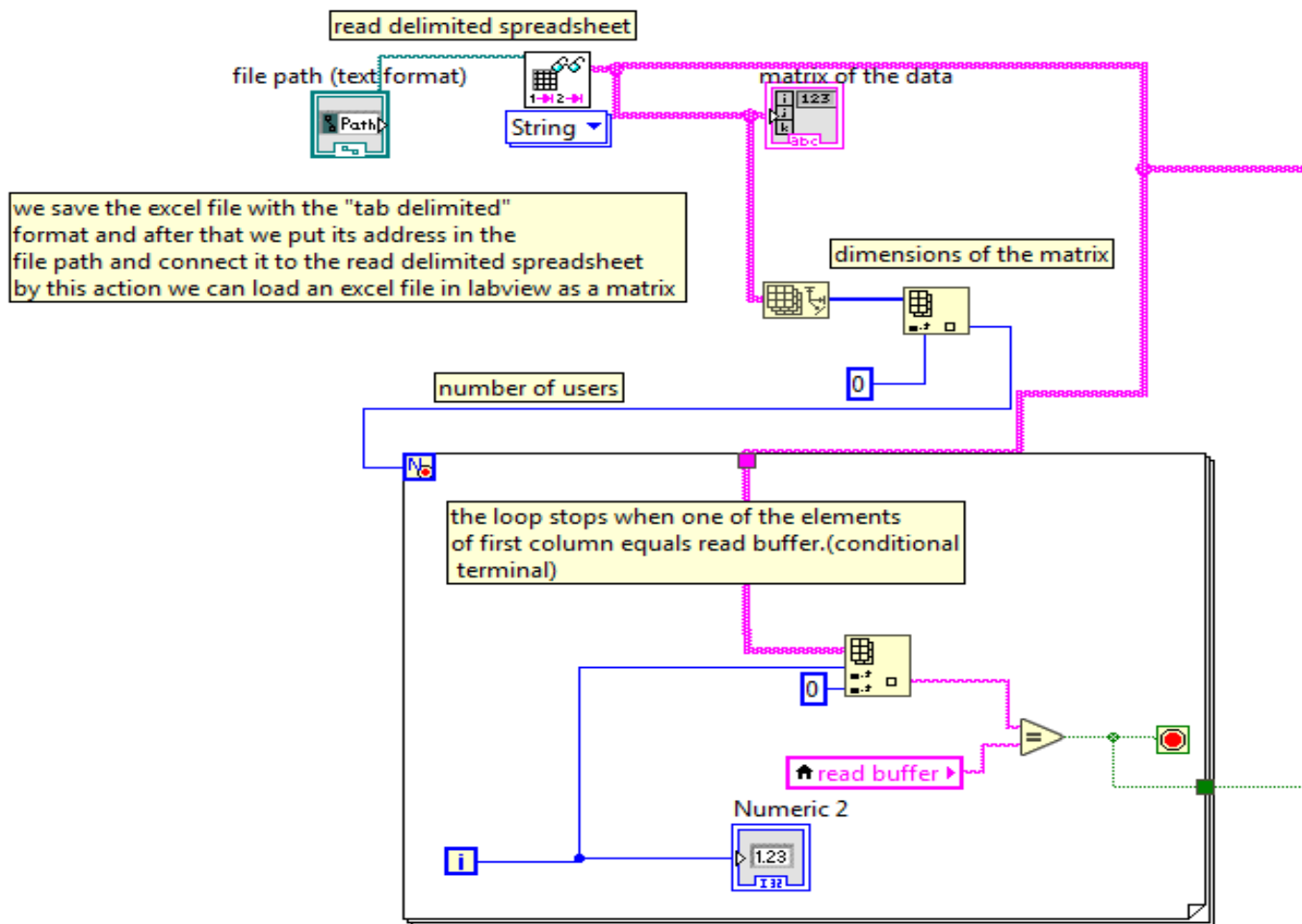
ای کار در سه مرحله انجام می شود: ابتدا لب ویو پورت مورد نظر را شناسایی میکند که چون این مرحله فقط یک بار انجام می شود بیرون از حلقه while قرار می گیرد. سپس اطلاعات از پورت خوانده شده و بعد در انتها نیز بیرون حلقه بایستی پورت را ببندیم تا دیگر توسط ما اشغال نشده باشد.

پس از این مرحله درون حلقه ی while بیرونی، ما یک عبارت شرطی داریم بدین نحو که اگر دکمه ی setting فشار داده نشود، حالت عادی برنامه اجرا می شود ینی افراد می توانند کارت خود را جلوی دستگاه قرار دهند و وارد شرکت شوند یا از آن خارج بشوند. اما اگر کاربری بخواهد عملیاتی از قبیل ثبت یک کارت مربوط به فردی جدید یا حذف کارت مربوط به یک فرد یا عوض کردن پسورد را انجام بدهد (تنظیمات را عوض کند) دکمه setting را فشار می دهد.

ورود و خروج افراد:

ما یک فایل اکسل با نام data base تعریف کرده ایم که در آن به ترتیب شماره کارت، اسم، کد ملی، سن، شماره پرسنلی، جنس، ملیت و گروه خونی کارکنان شرکت ذخیره شده است. (هر ردیف اطلاعات مربوط به یک فرد) سپس این فایل را به صورت tab delimited Text ذخیره کردیم. سپس از این فایل برای لود کردن اطلاعات در لب ویو استفاده کردیم.

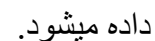
در شکل زیر، ابتدا فایل لود شده درون ماتریسی با نام matrix of data ریخته می شود، سپس ابعاد آن



گرفته شده و به تعداد سطر های این ماتریس که تعداد کارکنان است (درایه اول آرایه ای که ابعاد ماتریس را به ما می دهد) یک حلقه for تکرار میشود.

در هر بار اجرای حلقه ی بالا، خانه ی اول سطر i ام ماتریس اطلاعات کاربران، یا رشته ی عددی درون read buffer که همان شماره کارت زده شده است، مقایسه می شود. اجرای این حلقه تا زمانی ادامه پیدا

پس از آن اگر stop، مقدارش true باشد، وارد قسمت نشان داده شده در شکل زیر میشود. در اینجا سطر متناظر با کارت زده شده، آورده میشود و سپس شماره کارت و اسم فرد مورد نظر و نیز عکس وی نشان

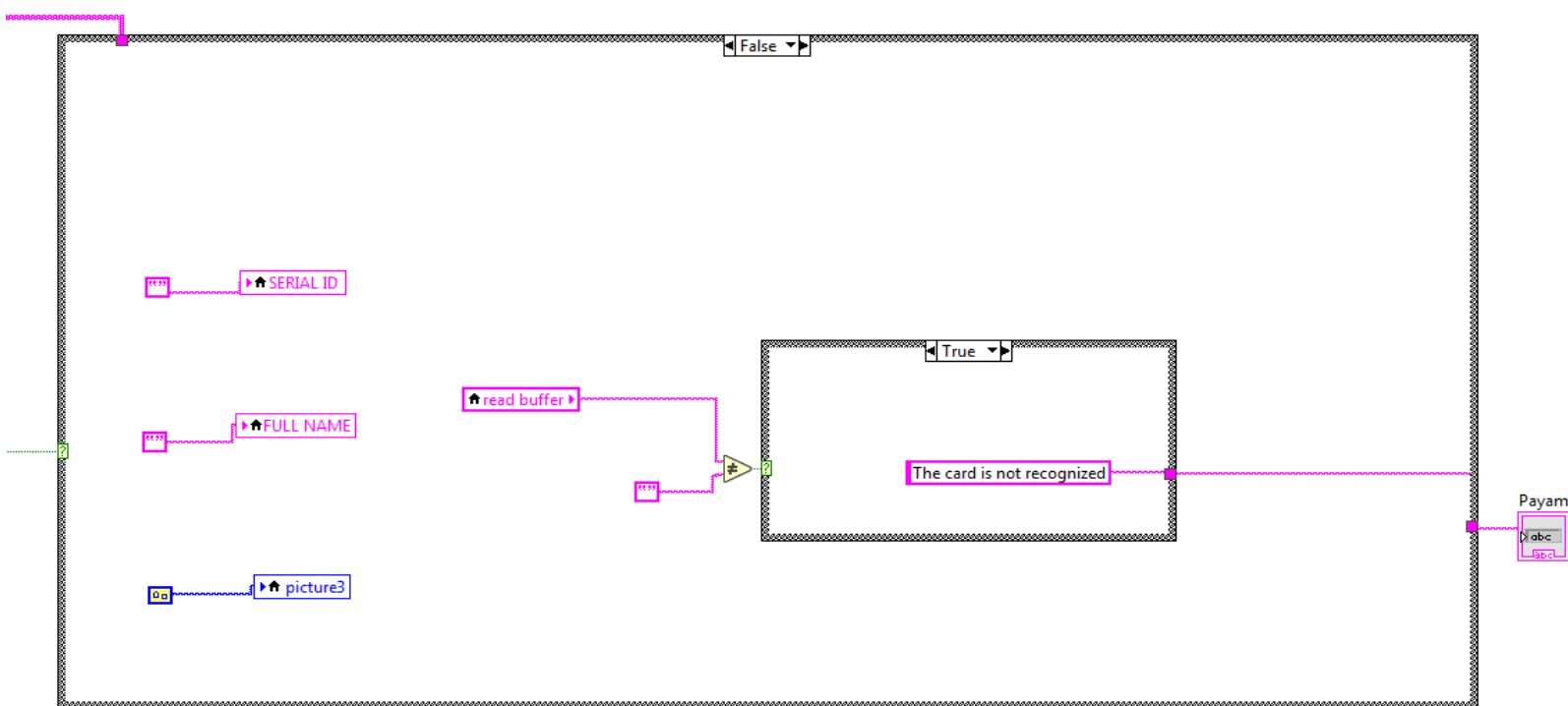


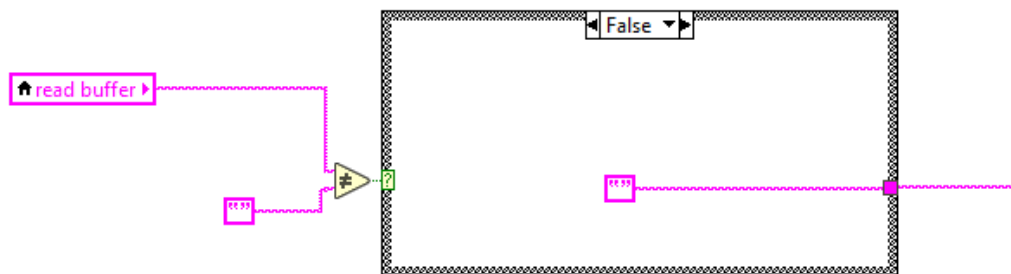
حال هدف این است که سطر مربوط به فردی که کارتش را زده که شامل اطلاعات { شماره کارت، نام، تاریخ و ساعت ورود } است به فایل تکستی با نام sabte Tarikh اضافه گردد. این کار بدین صورت انجام میگیرد که

سطر به کمک بلوک build array ساخته میشود و ماتریس لود شده در لب ویو با نام all rows افزوده میشود. همین جا نیز نمایش داده میشود. بعد همین ماتریس جدید که شامل اطلاعات به روزرسانی شده است دوباره داخل all rows ریخته می شود و در همان ادرس قبلی ینی در فایل تکست sabte Tarikh ذخیره می گردد.

در نهایت پیام "حضور موفقیت آمیز" داده می شود.

حال اگر کارت با هیچ یک از شماره های موجود در ماتریس مطابقت نداشت,وارد قسمت false از کیس می شویم که در زیر نشان داده شده است.متغیر ها صفر شده و پیام "ناشناس بودن کارت داده می شود





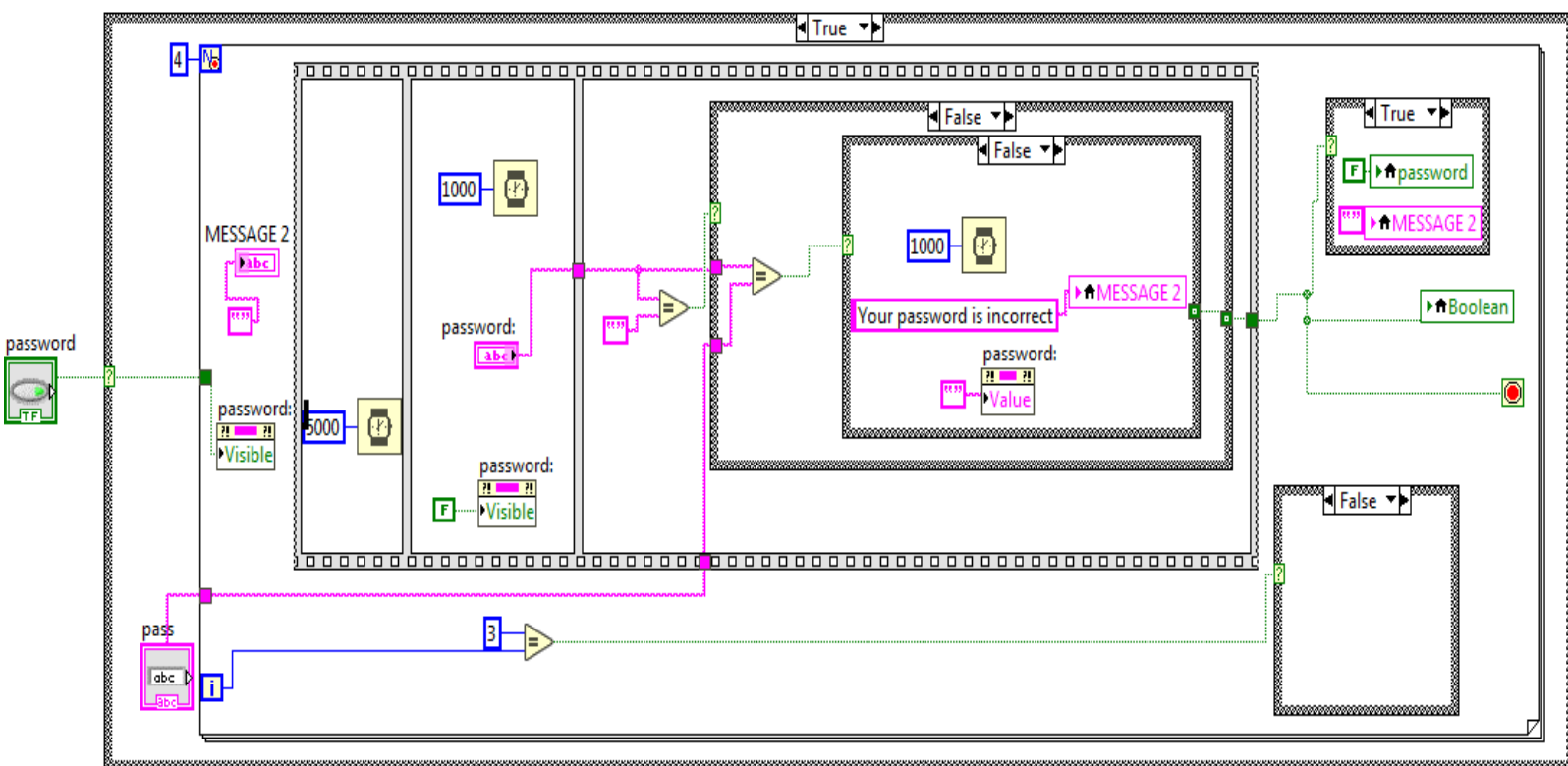
توجه کنید حالتی نیز در این قسمت case هست که هیچ کارتی زده نمیشود. در اینجا رشته ی خالی به درون پیام ریخته می شود.

در قسمت بعد مقصود این است که ساعت خروج افراد نیز ثبت و مشاهده گردد. برای این کار ابتدا بایستی زیرماتریسی از ماتریس all rows که شامل اطلاعات حضور افراد در روز کنونی می باشد و نه روز ها قبل جدا شود.

کد زیر این کار را انجام میدهد. به این صورت که درایه سوم تمام سطر ها که تاریخ است را با تاریخ کنونی مقایسه میکند و اگر درست بود شمارنده (عدد سطر نشیر آنها) را در یک آرایه ذخیره میکند. بعد ماکزیمم و مینیمم آرایه را در می آورد و زیر ماتریس را با توجه به این دو ایندکس می سازد.

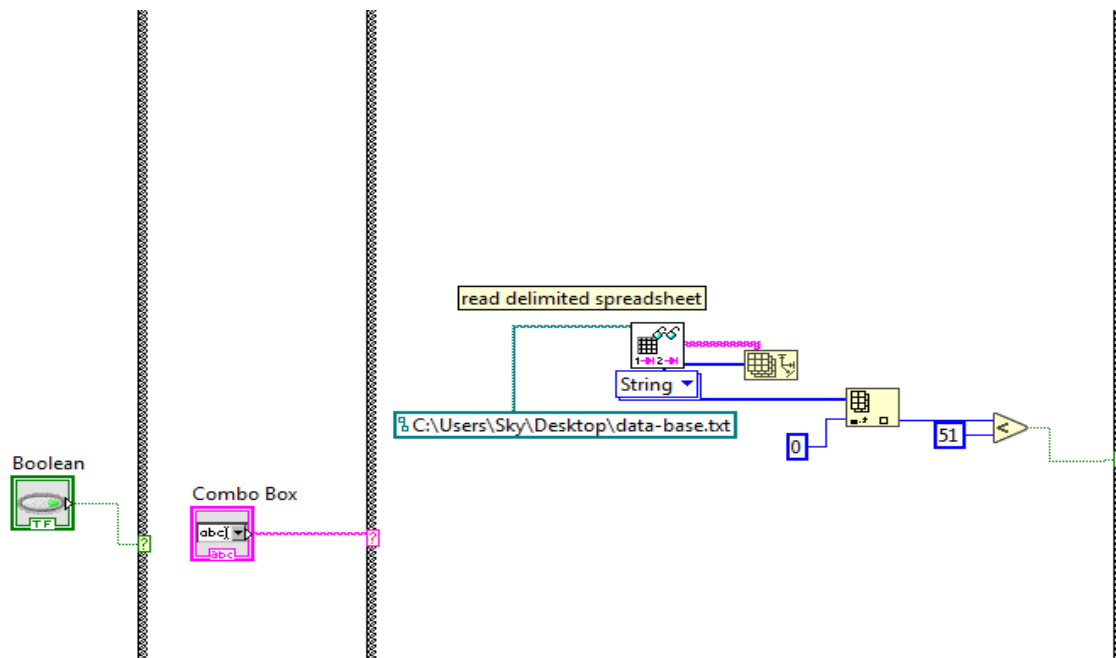
The diagram illustrates a LabVIEW program for finding the maximum and minimum values in an array. It begins with a 'home' button labeled 'all rows' that triggers a sequence of operations. The data flow starts with a '0' constant and a 'N' terminal, leading to a '2' terminal and a '1' terminal. These terminals are connected to a '2' terminal and a '1' terminal, which are then connected to a '2' terminal and a '1' terminal. The data is then processed by a '2' terminal and a '1' terminal, leading to a '2' terminal and a '1' terminal. The final output is a '2' terminal and a '1' terminal, which are connected to a '2' terminal and a '1' terminal. The diagram also includes a 'max value' and 'min value' indicator, a 'subarray' indicator, and a 'array w/ subset deleted' indicator.

در این حالت پس از ورود به حلقه for بلوک کنترلی برای ورود پسورد ظاهر میشود سپس پنج ثانیه فرصت میدهد تا پسورد را وارد کنیم پس از آن وارد یک مقایسه گر میشود تا آنچه به عنوان پسورد وارد شده بود مورد مقایسه با پسورد سیستم قرار گیرد در این حالت یا پسورد به درستی وارد شده که در این حالت بلافاصله حلقه متوقف شده و کلید پسورد را خاموش میکند تا وارد مرحله بعد شود در غیر این صورت اگر پسورد به درستی وارد نشود وارد کیس false شده و دوباره حلقه for تکرار میشود و مراحل بالا را تکرار میکند حال اگر حلقه برای بار سوم تکرار شود و همچنان پسورد ورودی اشتباه باشد در دور بعدی تکرار حلقه چون شمارنده زبانه مساوی میشود وارد کیس قسمت یابین شده و کلید ستینگ ر خاموش



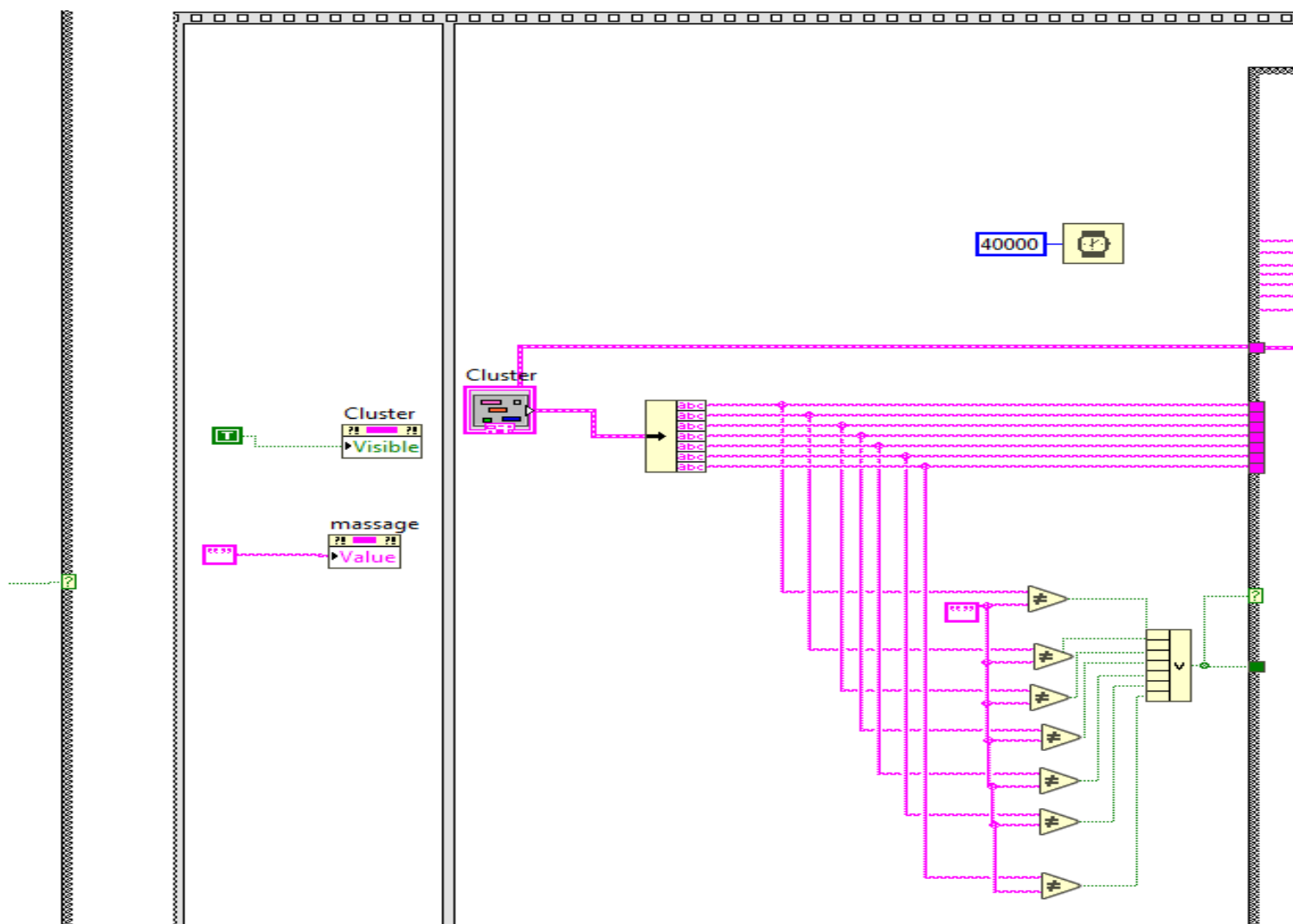
کرده و امکان ورود به قسمت ستینگ را به ایراتور نمیدهد.

پس از اینکه رمز درست وارد شد، کلید Boolean که شرط ورود کاربر به قسمتی است که میتواند تنظیمات را عوض کند، برقرار می شود. بدین معنا که اگر وی رمز را اشتبا بزند وارد این قسمت نخواهد شد. پس از آن طبق شکل زیر،

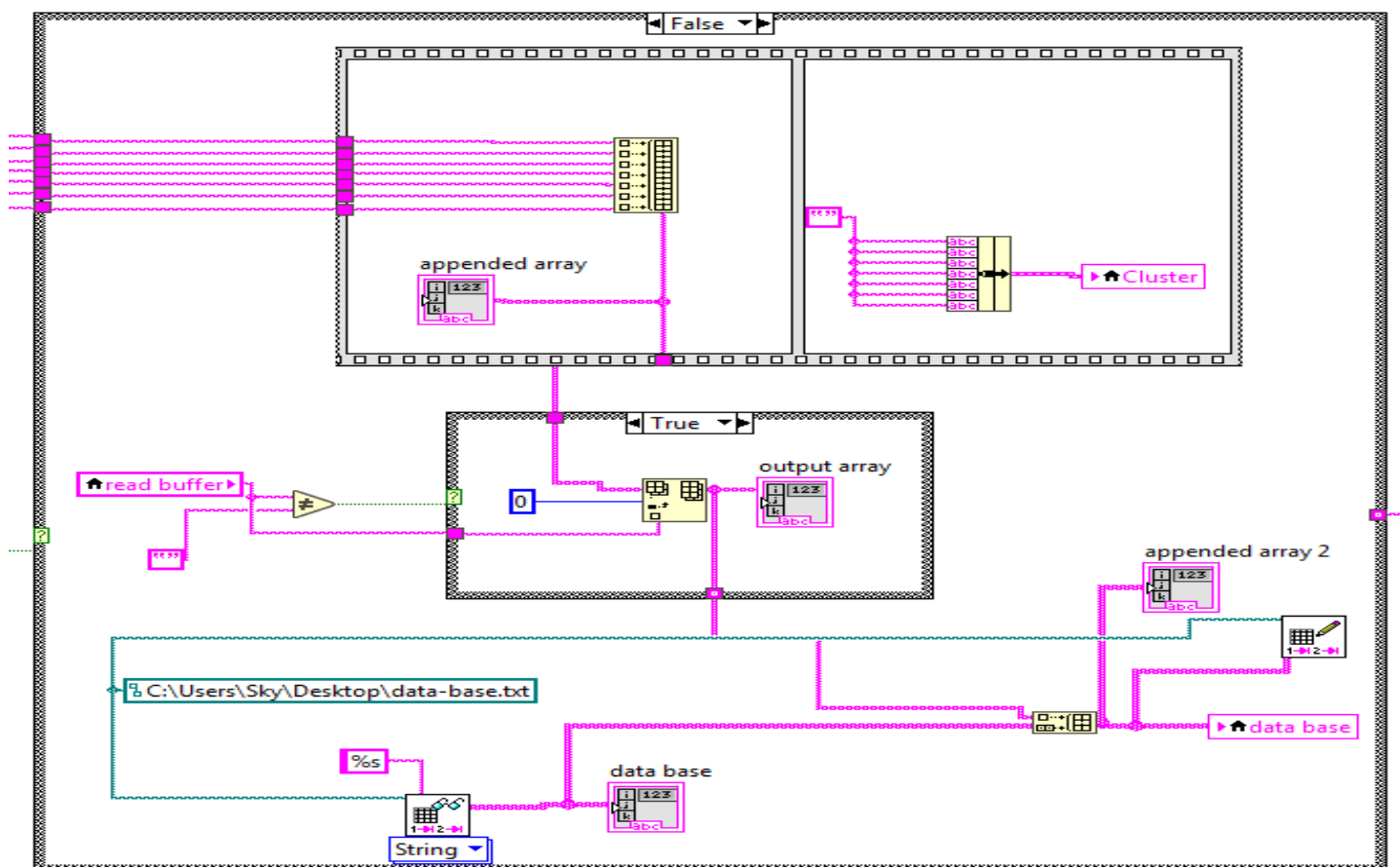


کاربر باید یکی از گزینه های combo box را انتخاب کند. با توجه به اسم مربوطه برای ثبت کارت جدی باید register انتخاب گردد. در صورت انتخاب این حالت، ابتدا فایل تکست data base فراخوانده شده و تعداد سطرهای آن که برابر با تعداد افراد ثبت شده است، محاسبه می شود. اگر این تعداد از 51 کمتر بود، ثبت نام وارد مرحله بعد میشود. در غیر این صورت پیامی تحت عنوان number of users out of range.your registration failed. ظاهر میگردد.

با فرض کمتر از 51 بودن تعداد، ابتدا cluster از حالت نامرئی در میاید و همچنین درون message رشته ی تهی ریخته میشود. پس از آن کاربر 40 ثانیه فرصت دارد تا cluster را که شامل اطلاعات

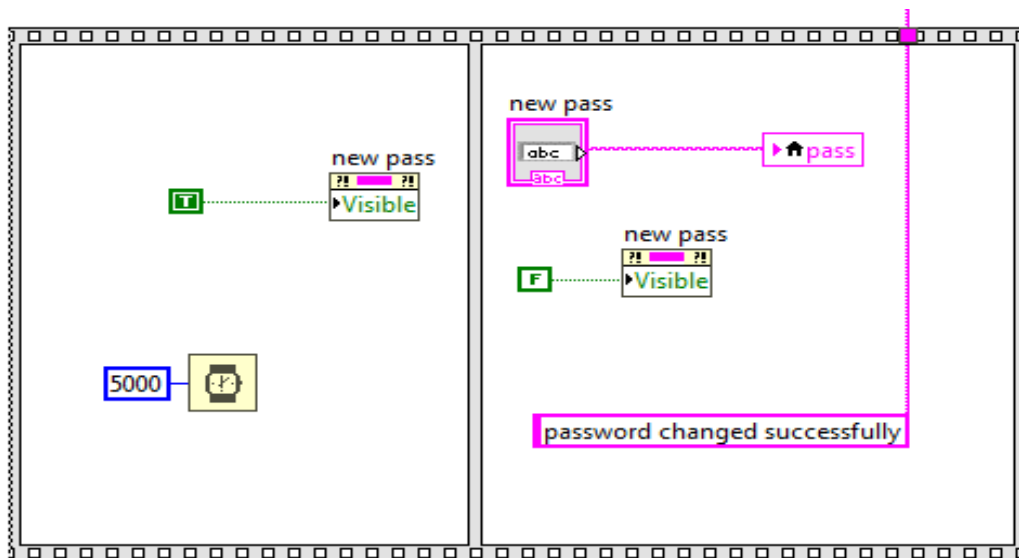


کارت جدید است را وارد کند و یک بار کارت را جلوی مدار بگیرد تا شماره آن در read buffer ذخیره گردد. بعد تک تک اجزاء کلاستر با رشته ی خالی، مقایسه می شوند. اگر جمع بولی حاصل این مقایسه گر ها یک شود (بدین معنا که حداقل کاربر یک جزء را پر کرده است) آنگاه دو حالت وجود دارد: یا کاربر همه قسمت ها را پر کرده یا حداقل یک قسمت خالی است. اگر مورد دوم باشد، با استفاده از یک حلقه ی for شرطی به اولین قسمت خالی کلاستر که رسیدیم از حلقه بیرون میاییم و سپس توسط پیامی از کاربر خواسته می شود تا تمامی قسمت های خواستع شده را پر نماید.



عوض کردن پسورد:

در صورتیکه کاربر در combo box گزینه ی change password را انتخاب نماید،وارد قسمت زیر می شود.



ابتدا new pass نمایان میشود و کاربر 5 ثانیه مهلت دارد تا پسورد جدید را وارد کند. سپس رمز جدید وارد متغیر کنترل

Pass می شود. همچنین new pass ناپدید می شود و پیام موفقیت آمیز بودن تغییر رمز داده می شود.

حذف یک کارت:

اگر کاربر در combo box گزینه ی delete را انتخاب نماید. وارد قسمت زیر می گردد:

