

```

#include "rfid1.h"
#include <SPI.h>
#include <SD.h>
#define readers 5

File myFile;
RFID1 rfid;

uchar serNum[5];    // raw rfid ID
char cardName[30]; // User Defined Name
byte cardID[10];   // ID pulled from SD card
char pos[10];      // Location of card scan
byte cardNum[8];   // Refined value of raw ID

void setup() {
  Serial.begin(9600);    //initialize the serial
  Serial.setTimeout(100000);
  Serial.print("Initializing SD card...");
  if (!SD.begin(10)) {  //Initialize the SD reader
    Serial.println("initialization failed!");
    while (1);
  }
  Serial.println("initialization done.");
//  SD.remove("DATA.csv"); //Remove old data file
  myFile = SD.open("DATA.csv", FILE_WRITE);
  if (myFile) {          //Write from beginning of file
    Serial.print("Writing to DATA.csv...");
//    myFile.println("Card ID,Position,Card Name");
    myFile.close();
    Serial.println("done.");
  } else {
    Serial.println("error opening DATA.csv");
  }
}

void loop()
{
  for (int i=0;i<readers;i++)
  {
    rfidswap(i);        //Poll all readers
    uchar status;
    uchar str[MAX_LEN]; //Search for cards
    status = rfid.request(PICC_REQIDL, str);
    status = rfid.anticoll(str); //Returns ID to process
    if (status == MI_OK)

```

```

{
  Serial.print("Reader: ");
  Serial.print(i);          //General info ouput
  Serial.println();        //Reader # and Card ID
  Serial.println("The card's number is: ");
  memcpy(serNum, str, 5);
  rfid.showCardID(serNum); //Show the card ID
  Serial.println();
  for(int j=0,k=0;j<8;j++,k++){
    cardNum[j]=(0x0f & (serNum[k]>>4)),HEX;
    Serial.print(cardNum[j],HEX); //Convert 2bit hex ID
    j++;                          //to 1bit hex ID
    cardNum[j]=(0x0f & serNum[k]),HEX;
    Serial.print(cardNum[j],HEX);
  }
  Serial.println();
  cardlog(i); //Record card info
  dataGrab(); //Pull card info
  rfid.halt(); //Put reader in sleep mode
}
}
}

void rfidswap(int i){
  if(i==0){
    rfid.begin(8, 7, 6, 4, 5, 9);
  }//rfid.begin(IRQ_PIN,SCK_PIN,MOSI_PIN,MISO_PIN,NSS_PIN,RST_PIN)
  if(i==1){
    rfid.begin(8, 7, 6, 3, 5, 9);
  }//rfid.begin(IRQ_PIN,SCK_PIN,MOSI_PIN,MISO_PIN,NSS_PIN,RST_PIN)
  if(i==2){
    rfid.begin(8, 7, 6, 2, 5, 9);
  }//rfid.begin(IRQ_PIN,SCK_PIN,MOSI_PIN,MISO_PIN,NSS_PIN,RST_PIN)
  if(i==3){
    rfid.begin(8, 7, 6, A0, 5, 9);
  }//rfid.begin(IRQ_PIN,SCK_PIN,MOSI_PIN,MISO_PIN,NSS_PIN,RST_PIN)
  if(i==4){
    rfid.begin(8, 7, 6, A1, 5, 9);
  }//rfid.begin(IRQ_PIN,SCK_PIN,MOSI_PIN,MISO_PIN,NSS_PIN,RST_PIN)
  rfid.init(); //initialize the RFID
}

void cardlog(int i){
  myFile = SD.open("DATA.csv", FILE_WRITE);
  if (myFile){ //Open data file
    Serial.print("Card Name: End with #");

```

```

Serial.println(); //Ask for card name
Serial.write(13);
Serial.readBytesUntil('#',cardName,30);
myFile.print(serNum[0],HEX);
myFile.print(serNum[1],HEX);
myFile.print(serNum[2],HEX);
myFile.print(serNum[3],HEX);
myFile.print(serNum[4],HEX);
myFile.print(","); //Display CardID
myFile.print(i); //Display Reader
myFile.print(",");
myFile.println(cardName);
myFile.close(); //Display Card Name
Serial.print("Wrote: ");
Serial.println(cardName);
memset(cardName, 0, sizeof(cardName));
}else {
Serial.println("error writing");
myFile.close();
}
}

void dataGrab() {
myFile = SD.open("DATA.csv", FILE_READ);
while(myFile){ //Open data file
memset(cardID, 0, sizeof(cardID));
memset(pos, 0, sizeof(pos)); //Clean buffer arrays
memset(cardName, 0, sizeof(cardName));
myFile.readBytesUntil(',',cardID,30);
myFile.readBytesUntil(',',pos,10); //Data into arrays
myFile.readBytesUntil('\n',cardName,30);
for(int j=0;j<8;j++){
if('A' <= cardID[j] && cardID[j] <= 'F'){
cardID[j]=cardID[j]-55; //Ascii char value into decimal value
}else if('0' <= cardID[j] && cardID[j] <= '9'){
cardID[j]=cardID[j]-'0';
}
}
} //If cardID matches pulled ID

if((cardID[0]==cardNum[0]) && (cardID[1]==cardNum[1]) && (cardID[2]==cardNum[2]) && (cardID
[3]==cardNum[3]) && (cardID[4]==cardNum[4])) {
for(int j=0;j<8;j++){
Serial.print(cardID[j],HEX);
}
Serial.println(); //Print pulled card info
Serial.println(pos);
}
}

```

```
Serial.println(cardName);  
myFile.close(); //File close resets reading position to begining, remains open to  
scan whole file for match  
}else {  
// Serial.println("pass");  
// myFile.close();  
}  
}  
}
```