**SPRAWOZDANIE**

*Ćwiczenie 13*

*RaspberryPi*

**WYKONANIE:**

* Szymon Hutnik 252736
* Damian Mucha 252708

**TERMIN ZAJĘĆ:** środa TP, 11.15

**DATA:** 10.11.2021

**PROWADZĄCY:** dr inż. Tomasz Walkowiak

1. Treść zadania

Celem ćwiczenia było zapoznanie się z budową oraz RaspberryPi oraz płytki Picoboard [POTWIERDZIĆ].

Zadaniem do zrealizowania było napisanie programu w Pythonie, który pozwalałby na sterowanie układem na płytce [CZY JAK TO SIĘ FACHOWO NAZYWA]. Należało zbudować prosty układ z dostępnych elementów np. diód.

1. Opis urządzenia

Karta chipowa – nośnik danych charakteryzujący się wielokrotnością usług oraz lepszą ochroną danych w porównaniu do jego odpowiednika z paskiem magnetycznym. Najczęściej jest zbudowana z tworzywa sztucznego, w którym jest osadzony układ scalony – chip, którego styki są pokryte złotem, standard opisuje 8 rodzajów, ale producent nie musi umieszczać wszystkich. Mikroprocesor zapewnia kontrolę odczytu i zapisu danych, może kontrolować ilość nieudanych prób odblokowania, aby móc zablokować kartę po przekroczeniu limitu. Najczęściej stosowanymi mikroprocesorami są 8-bitowe moduły z pamięcią EEPROM, którą możemy podzielić na 3 obszary:

* swobodnego odczytu – zazwyczaj przechowuje powszechne informacje o karcie i/lub użytkowniku
* poufny – dostęp do niego wymaga wprowadzenia poprawnego kodu PIN
* roboczy – przechowuje dane, które są poddawane ciągłej modyfikacji

Karty chipowe można podzielić ze względu na interfejsy komunikacyjne, czyli sposób przesyłania danych:

* stykowe – zgodne ze standardem ISO-7816. W tego rodzaju kartach wyróżniamy 2 główne protokoły komunikacyjne:
  + T=0 – jednokierunkowa transmisja bajtów
  + T=1 – jednokierunkowa transmisja bloków
* bezstykowe – zgodne ze standardem ISO14443. Komunikacja z czytnikiem polega na wykorzystaniu fal elektromagnetycznych na różnych częstotliwościach

System plików kart SIM przedstawia się następująco:

Działanie czytnika kart jest proste – po umieszczeniu karty w czytniku zostaje ona zasilona przez odpowiedni styk i wysyła wiadomość ATR (Answer to Reset), która informuje czytnik o rodzaju karty, sposobie kodowania bitów i wspieranych protokołach komunikacji. Po nawiązaniu połączenia można komunikować się z kartą za pomocą prokokołu APDU.

1. Program sterujący

import RPi.GPIO as GPIO

import time

import sys

import os

GPIO.setmode(GPIO.BCM)

GPIO.setwarnings(False)

GPIO.setup(23, GPIO.OUT)

GPIO.setup(21, GPIO.OUT)

GPIO.setup(26, GPIO.IN)

GPIO.output(21, GPIO.LOW)

for q in range (5):

print("ON")

GPIO.output(23,GPIO.HIGH)

time.sleep(4)

print("OFF")

GPIO.output(23,GPIO.LOW)

time.sleep(1)

if(GPIO.input(26) == 1):

GPIO.output(21, GPIO.HIGH)

else:

GPIO.output(21, GPIO.LOW)

GPIO.cleanup

1. Przedstawienie aplikacji

Graphical user interface, text, application

Description automatically generated

1. Liteatura

* ISO/IEC 7816-4 Command Set
* Smart Card API Reference
* PC/SC wrapper classes for .NET Documentation