

Krzysztof Lichy
Instytut Informatyki Politechniki Łódzkiej
Wólczańska 215, 90-924 Łódź
email: krzysztof.lichy@p.lodz.pl

ZASTOSOWANIE KOMPUTERA KLASY SBC JAKO RUTER PRACUJĄCY W SIECI SOHO

Streszczenie – Celem pracy jest sprawdzenie czy komputer Raspberry Pi wyposażony w odpowiednie oprogramowanie może zastąpić ruter w zastosowaniach SOHO poprzez wykonanie działającego prototypu.

Słowa kluczowe: Raspberry Pi, SOHO, ruter, Wi-Fi

1 Wstęp

Na rynku jest dostępnych wiele różnorodnych routerów dla rozwiązań klasy SOHO. Ich wspólną cechą jest fakt, że są one konstrukcją zamkniętą w której zazwyczaj poza aktualizacją lub wymianą firmware nie można dokonywać żadnych zmian. Z kolei alternatywne rozwiązanie oparte na komputerze klasy PC charakteryzuje się dużym rozmiarem, poborem mocy oraz generowaniem hałasu i ciepła. Ciekawą alternatywą wydają się mikrokomputery takie jak Raspberry Pi, czy Banana Pi. Mają one wszystkie zalety komputerów klasy PC a jednocześnie są pozbawione wspomnianych wyżej wad. W niniejszej pracy zaprezentowano prototyp routera opartego o mikrokomputer Raspberry Pi. Elementy teorii grafów

2 Dobór parametrów składowych rozwiązania

Według badań przeprowadzonych przez portal netindex.com, średnia przepustowość łącza w Polsce oscyluje (dla pobierania) w okolicach 22 Mb/s i dla wysyłania 10Mb/s [1]. Ograniczając ruter jedynie do roli bramy dostępnej najpopularniejsze obecnie standardy sieci bezprzewodowej są wystarczające w zupełności dla takich przepustowości. Należy zwrócić jednak uwagę, że w zastosowaniach SOHO bardzo często to samo urządzenie służy do łączenia ze sobą hostów w sieci co wymusza zastosowanie możliwie najwydajniejszego rozwiązania. Na dzień dzisiejszy najoptymalniejszym rozwiązaniem wydaje się być standard 802.11ac, niestety ograniczenia sprzętowe platformy Raspberry Pi praktycznie wy-

kluczają jego zastosowanie [2]. W prezentowanym rozwiązaniu zostanie wykorzystany standard 802.11n.

Jednym z aspektów który należy wziąć pod uwagę jest kwestia odporności na brak zasilania. W dużych profesjonalnych systemach odpowiadają za to skomplikowane systemy zasilania awaryjnego. W rozwiązaniach SOHO nie zawsze jest do dyspozycji UPS dlatego prezentowane rozwiązanie zostanie wyposażone w awaryjne zasilanie bateryjne/akumulatorowe uruchamiane w momencie braku zasilania prądem sieciowym. System powinien być przyjazny dla użytkownika dlatego założono że dla uproszczenia konfiguracja będzie możliwa na wyświetlaczu dotykowym za pomocą dedykowanego oprogramowania. Ze względu na specyfikę systemu Linux, na którym oparte będzie urządzenie pozostawiono możliwość konfiguracji poprzez protokół SSH. Rezygnacja z konieczności użycia przeglądarki www jest szczególnie przydatna w sieciach domowych, w których urządzenia podłączane do Internetu niejednokrotnie nie posiadają wbudowanej przeglądarki www. co jest szczególnie kłopotliwe przy pierwszym uruchomieniu

3 Platforma sprzętowa

Konstruując urządzenie zdecydowano się na wykorzystanie komputera pracującego pod kontrolą systemu Linux. Za tym wyborem przemawiają kwestie licencyjne oraz dostępność literatury. Ze względu na cenę oraz szeroko dostępną literaturę zdecydowano wykorzystać mikrokomputer Raspberry Pi. Jest to mikrokomputer wielkości karty kredytowej. Wybrano wersję B+ posiadającą procesor ARM o taktowaniu 700 MHz oraz 512 MB pamięci operacyjnej RAM. Na potrzeby projektu zdecydowano wykorzystać system Raspbian, kryterium wyboru była jego popularność oraz fakt, że jest to system bezpośrednio wywodzący się z dystrybucji Debian[12].

We wspomnianym wyżej celu uniezależnienia routera od urządzeń zewnętrznych zostanie wykorzystany dotykowy [3]. Wybrane rozwiązanie posiada przekątną 2,8 cala oraz umożliwia wyświetlanie w rozdzielczości 240x320 pikseli. Wyświetlacz ten charakteryzuje się niskim poborem prądu, niestety jego rozmiary wykluczają użycie klawiatury ekranowej. Wymusiło to konieczność zastosowania zewnętrznej klawiatury. W tej roli znakomicie sprawdziła się klawiatura bezprzewodowa [4].

System został wyposażony w dwa bezprzewodowe interfejsy sieciowe. Zdecydowano się na użycie kart TP-Link TL-WN725N [5], wybór został podyktowany faktem, że karta ta jest kompatybilna z systemem Raspbian oraz bezproblemowo działa w trybie klienta jak i punktu dostępu. Wspomniana karta działa w standardzie 802.11n. Jedną z zalet mikrokomputera Raspberry Pi jest niski pobór prądu, w celu umożliwienia

pracy w przypadku braku zasilania sieciowego układ został uzupełniony o opcjonalne zasilanie bateryjne.

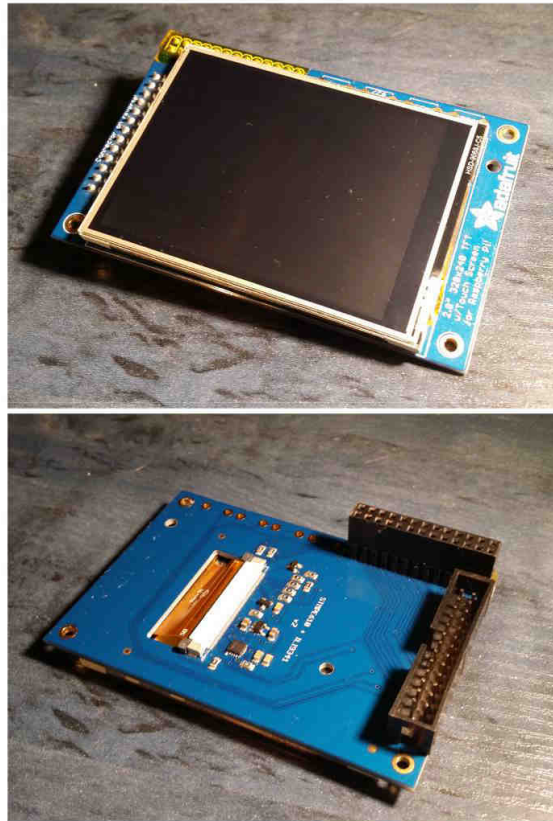


Fig. 1. Wyświetlacz dotykowy [11]

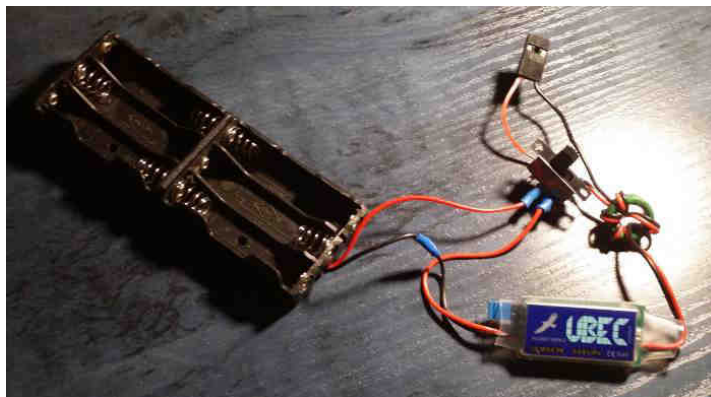


Fig. 2. Moduł zasilania bateryjnego [11]

Pewnym problemem okazało się wybranie odpowiedniej obudowy. Na rynku dostępne są różnego rodzaju obudowy uniwersalne, żadna jednak nie umożliwiła schowanie dodatkowego osprzętu (koszyk na baterie, wyświetlacz, karty sieciowe). Zgodnie z założeniami urządzenie miało stanowić zamkniętą całość w związku z tym zdecydowano się na zaprojektowanie i wykonanie własnej obudowy.

4 Oprogramowanie

Złożony mikrokomputer jest praktycznie od razu gotowy do pracy. Niestety konfiguracja systemu z poziomu dostępu terminalowego byłaby zbyt skomplikowana dla przeciętnego użytkownika. W celu ułatwienia obsługi rutera zaimplementowano graficzny interfejs użytkownika. W tym celu wykorzystano język Python [6] wraz z biblioteką graficzną Tkinter [7][13][14].

Na potrzeby projektu został wykonany indywidualny projekt obudowy. Cechuje go prostota i funkcjonalność. Priorytetem było takie ułożenie komponentów wewnątrz, aby wszystkie się bez problemu zmieściły, oraz w razie potrzeby był do nich łatwy dostęp, przy zachowaniu jak najmniejszego rozmiaru. Obudowa została podzielona na dwie części łączone ze sobą za pomocą śrub. W górnej znalazł się wyświetlacz, regulator napięcia oraz włącznik zasilania bateryjnego.

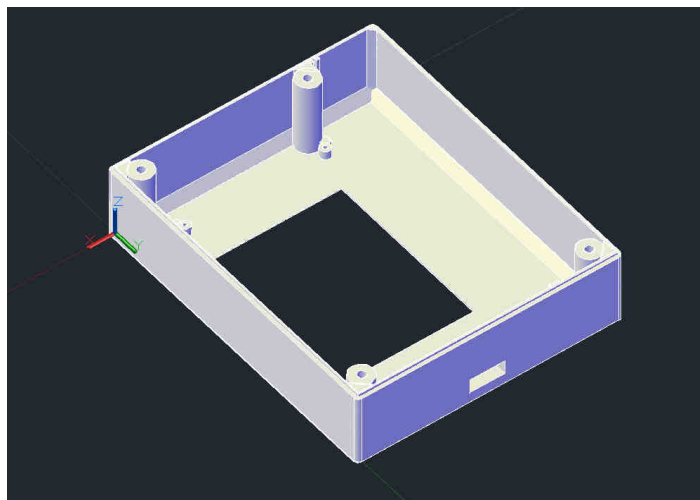


Fig. 3. Górna część obudowy [11]

W dolnej natomiast umieszczono Raspberry oraz koszyki z bateriami.

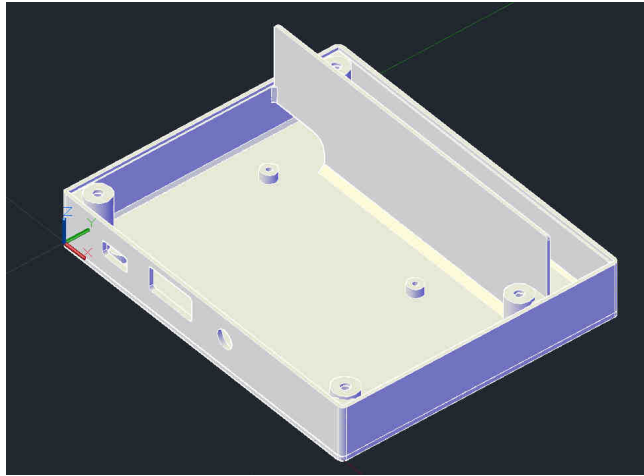


Fig. 4. Dolna część obudowy [11]

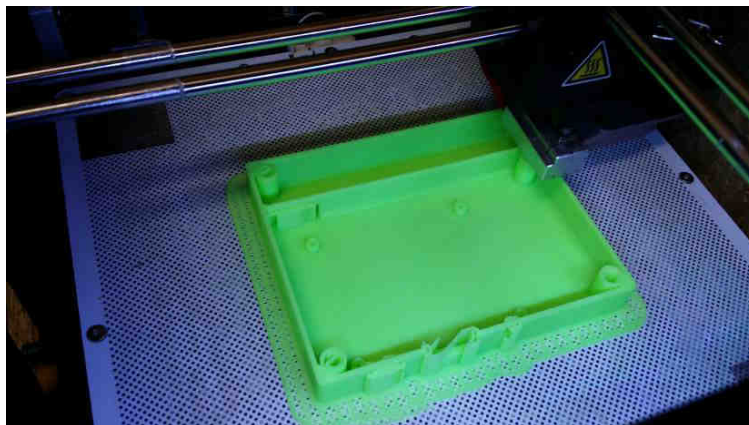


Fig. 5. Proces wydruku obudowy [11]

Komora baterii została oddzielona ścianką od komory z elektroniką, na wypadek, gdyby baterie uległy uszkodzeniu. Raspberry Pi, wyświetlacz oraz włącznik zasilania bateryjnego zostały przymocowane do obudowy za pomocą śrub, gdyż posiadają otwory technologiczne do mocowania. Regulator napięcia został przyklejony, natomiast koszyki z bateriami zostały tak dopasowane, aby ich mocowanie na stałe nie było konieczne. Ścianki obudowy mają 1,5 mm grubości a całe urządzenie ma kształt sześcianu o wymiarach 11 cm x 9 cm x 4,5 cm. W górnej części znalazł się otwór na wyświetlacz, natomiast na bocznych ścianach znajdują się otwory na włącznik zasilania, wyjście HDMI oraz wyjście audio. Obudowa została wydrukowana drukarką Zortrax M200.

Użyтым materiałem był filament Z-ABS. Jest to tworzywo sztuczne dostarczone przez producenta drukarki opracowane dla tego konkretnego modelu na bazie ABS - materiału, z którego wykonywane są obecnie większość obudów sprzętu elektronicznego, zabawek itp.

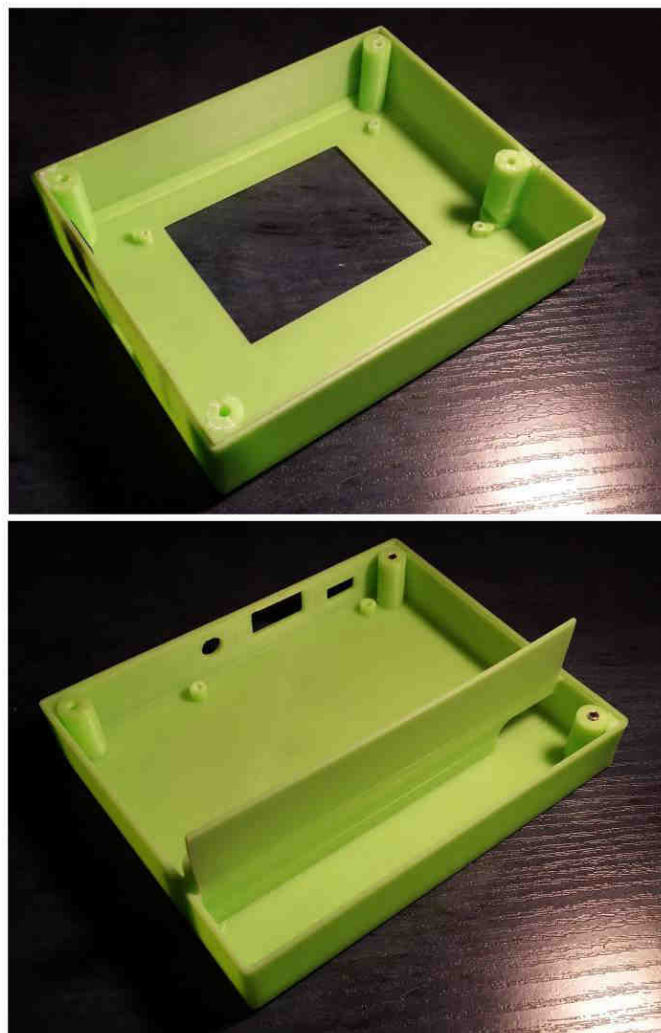


Fig. 6. Gotowa obudowa [11]

Podczas wydruku tworzone są podpory (tzw. supporty) dla elementów, które musiałyby być wydrukowane „w powietrzu”, dlatego obudowa przed złożeniem wymagała jeszcze obróbki pilnikiem i papierem ściernym. Materiał ten jest podatny na szlifowanie, więc dopasowanie obudowy nie przysporzyło większych problemów. Przy projektowaniu nale-

żało zwrócić szczególną uwagę na elementy, które powinny być ze sobą spasowane. Należało przyjąć szczelinę równą około 0,4mm pomiędzy elementami, które na siebie nachodzą, co zaoszczędziło zbędnego szlifowania. Kolejnym istotnym elementem były wzmocnienia elementów szczególnie wrażliwych, takich jak zaokrąglone łączenia ścian bocznych, tulejki na śruby łączące dwie części obudowy oraz podpierające elementy elektroniczne. Technika druku 3D nie jest doskonała i nie tworzą się jednolite połączenia pomiędzy elementami obudowy, przez co wzmocnienia są szczególnie istotne.

Wyświetlacz łączony jest z Raspberry poprzez złącze GPIO, zatem jego podłączenie nie sprawia większych problemów[11]. Producent przygotował także specjalną dystrybucję [8], w której wszystkie wymagane sterowniki i kalibracje są już zrobione i zainstalowane. Całość sprowadza się do podłączenia i instalacji Linuxa na karcie pamięci.

5 Interfejs użytkownika

Zadaniem programu jest ukrycie nieprzyjaznej użytkownikowi warstwy terminala pod postacią łatwego w obsłudze graficznego interfejsu użytkownika. Jest to nieskomplikowana aplikacja pełnoekranowa, która wywołuje polecenia terminala. Składa się ona z trzech zakładek:

- połącz
- udostępnij
- ustawienia

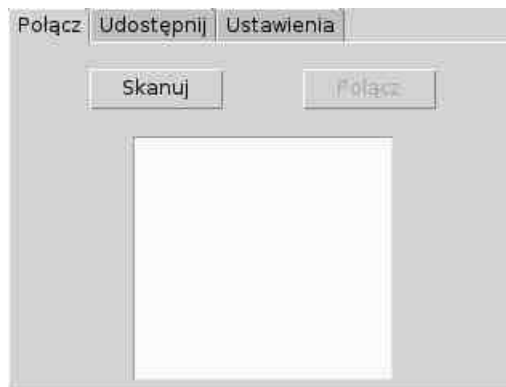


Fig. 7. Interfejs użytkownika [11]

Ostatnim etapem budowy było zamontowanie wszystkich podzespołów w obudowie. Wyświetlacz, włącznik oraz Raspberry

zostały przykręcone na śruby. Cała obudowa została również skręcona. Wszystkie elementy miały zaprojektowane swoje miejsce, więc złożenie wszystkiego było czynnością trywialną.

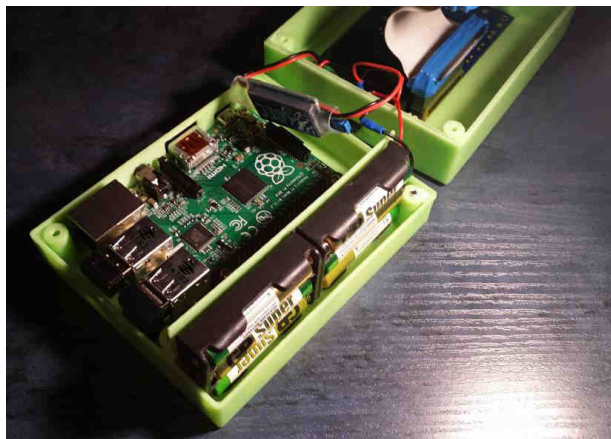


Fig. 8. Zmontowane urządzenie [11]

6 Testy

Przetestowano czy urządzenie (w sposób stabilny i nieprzerwany) jest w stanie obsługiwać 10 hostów. W tym celu podłączono hosty i wygenerowano na nich ruch sieciowy.

```
pi@raspberrypi ~ $ cat /var/lib/misc/dnsmasq.leases
1424212506 a0:a8:cd:6b:5f:30 192.168.97.29 MW7CK3OTDH8H3V 01:a0:a8
:cd:6b:5f:30
1424212356 a0:e4:53:8e:b5:87 192.168.97.28 android-294
be2d930407c08 *
1424212182 40:f3:08:c4:2b:92 192.168.97.8 android-63af4603c20c9269
01:40:f3:08:c4:2b:92
1424212056 1c:7b:21:fd:47:dc 192.168.97.3 android-8a99449fb6ef7df5
*
1424212012 a0:a8:cd:6b:5e:b3 192.168.97.23 MW7P9E99YJWI0P 01:a0:a8
:cd:6b:5e:b3
1424212009 58:1f:aa:1a:78:0f 192.168.97.46 Mati 01:58:1f:aa:1a
:78:0f
1424211529 44:d4:e0:ae:01:68 192.168.97.25 android-689673
a4e299a534 *
1424212436 60:d9:c7:32:32:30 192.168.97.39 iPhone 01:60:d9:c7
:32:32:30
1424212011 90:c1:15:1a:eb:de 192.168.97.32 androidd5d80fda7b30e70e
*
1424211753 d8:57:ef:ee:5a:f0 192.168.97.6 * 01:d8:57:ef:ee:5a:f0
pi@raspberrypi ~ $
```


Wraz podłączonymi klientami urządzenie pracowało w sposób stabilny [11]. Nie zaobserwowano przerw w połączeniach oraz jakiegokolwiek niepokojącego zachowania mikrokomputera. Kolejne testy wydajnościowy wykazały ograniczenia związane z wykorzystywaniem interfejsu USB 2.0, który okazał się wąskim gardłem systemu. Średni czas kopiowania pliku o rozmiarze 100MB między dwoma węzłami sieci wynosił około 26,27s. Co daje średnią prędkość 31Mb/s. Należy mieć jednak na uwadze, że jest to wartość przybliżona. Używany interfejs USB 2.0 jest zdecydowanie najsłabszym elementem proponowanego systemu. Ze względu na to ograniczenie, jak wspomniano wcześniej, nie ma sensu stosować interfejsy sieciowe o większej przepustowości.

O ile istnieją opisy działających access pointów zbudowanych w oparciu o Raspberry Pi [9][10] to nie udało się znaleźć opisów takich urządzeń pracujących w oparciu o zasilanie bateryjne. Proponowany system przy pełnym obciążeniu pracował stabilnie przez około 4h (przy wykorzystaniu 6 baterii AAA).

Ruter wykorzystuje WPA2-PSK z szyfrowaniem AES, nie zaobserwowano problemów wydajnościowych związanych z obsługą szyfrowania.

Ostatnim testem rozwiązania było sprawdzenie intuicyjności oraz prostoty użytkowania. Testy przeprowadzane na użytkownikach, których poproszono o przygotowanie systemu do pracy potwierdziły intuicyjność systemu. Średni czas konfiguracji oscylował w okolicy 3-4 minut [11].

7 Podsumowanie

System spełnił zakładane cele. Zauważyć jednak należy, że nie jest on w pełni konfigurowalny przez interfejs graficzny. Pewne jego funkcjonalności zostały określone na etapie tworzenia i nie mogą zostać zmienione. Oczywiście cały czas istnieje możliwość konfiguracji za pomocą protokołu SSH. W rozwiązaniu nie testowano odporności systemu dyskowego na zużycie karty sd. W kolejnych wersjach urządzenia należałoby przenieść system na pendrive lub przynajmniej jego najczęściej używane katalogi (/tmp /var/log /etc).

W kolejnych wersjach planowane jest zaimplementowanie usług katalogowej RADIUS oraz wbudowanie klienta LDAP/RADIUS

Reasumując zbudowany system wydaje się być ciekawą alternatywą dla dostępnych na rynku rozwiązań, szczególnie ze względu na modułową konstrukcję zarówno od strony soft jak i hardware.

8 Literatura

- [1] Średnia prędkość łącza w Polsce <http://www.netindex.com/download/2,2/Poland/>
- [2] Raspberry Pi model B+ <http://www.raspberrypi.org/products/model-b-plus/>
- [3] <http://www.adafruit.com/product/1601>
- [4] <http://botland.com.pl/akcesoria-usb-raspberry-pi/2381-klawiatura-bezprzewodowa-touchpad-mini-key.html>
- [5] <http://www.tp-link.com.pl/products/details/?model=TL-WN725N>
- [6] <https://docs.python.org/3/>
- [7] <https://docs.python.org/2/library/tkinter.html>
- [8] <http://www.raspberrypi.org/downloads/>
- [9] <http://osworld.pl/access-point-na-raspberry-pi/>
- [10] <https://learn.adafruit.com/downloads/pdf/setting-up-a-raspberry-pi-as-a-wifi-access-point.pdf>
- [11] Powiłań P., *Mikrokomputer Raspberry Pi użyty jako router w rozwiązaniach SOHO*, praca dyplomowa inżynierska, wydział FTIMS PŁ 2015
- [12] Brianson T., *Raspberry Pi 2: The Ultimate Raspberry Pi 2 User Guide*, Kindle Edition
- [13] Gore W., *Raspberry Pi: Guide For Simple Python & Projects Programming*, Kindle Edition
- [14] Cox T., *Raspberry Pi Cookbook for Python Programmers*, Packt Publishing 2014
- [15] Kurniawan A., *Raspberry Pi Wireless Networks*, Kindle Edition, 2015

SBC CLASS COMPUTER USED AS RUTER FOR SOHO NETWORK

Summary – The aim is to check whether the Raspberry Pi computer equipped with the appropriate software can replace the router for SOHO by making a working prototype

Keywords: Raspberry PI, SOHO, router, Wi-Fi