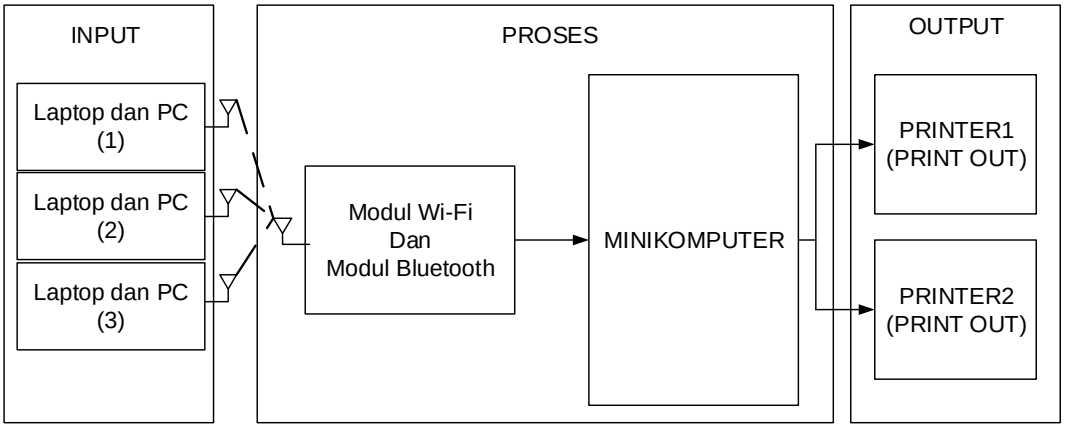
BAB III

PERANCANGAN ALAT

Perancangan merupakan bagian yang penting dari pembuatan alat yang akan dirancang dalam tugas akhir ini. Perancangan sistem dirancang agar alat yang dibuat lebih terkonsep berdasarkan teori dan referensi yang yang berkaitan dengan alat yang akan dirancang, sehingga akan mempermudah dalam pengerjaan dan analisanya.

3.1. Blok Diagram Sistem

Dalam perancangan sistem *printer* nirkabel secara umum terdapat tiga bagian yaitu masukan (*input*), proses (*process*) dan keluaran (*output*). Berikut ini Gambar 3.1 merupakan blok diagram sistem yang akan dirancang.



Gambar 3. 1 Blok Diagram Sistem

3.1.1. Fungsi Masing-masing Blok

Berikut uraian fungsi dari masing-masing bagian utama blok diagram rancang bangun sistem *printer* nirkabel.

1. Masukan (*Input*)

Untuk bagian masukan (*input*) terdapat dua macam perangkat yang berfungsi untuk memberikan masukan untuk SBC (Single Board Computer). Sebelum terkoneksi dengan sistem, pengguna harus masuk ke jaringan *Wi-Fi* untuk koneksi Wi-Fi atau *pairing* dengan *Bluetooth* untuk memilih komunikasi menggunakan *Bluetooth*. Setelah terkoneksi, pengguna memilih *printer* yang telah disediakan sistem dan kemudian pengguna sudah dapat melakukan *print out data* yang akan dicetak. *Input* yang diterima berupa data-data dokumen, gambar dan yang sejenisnya.

a. Modul *Bluetooth* digunakan sebagai alat komunikasi antara perangkat pengirim dengan sistem perangkat *printer* nirkabel. *Bluetooth* akan dipasangkan pada *SBC (Single Board Computer)* melalui piranti antar muka USB dan dijadikan sebagai akses poin. Komunikasi *Bluetooth* dapat dilakukan dengan memasukan kode pengaman pada saat *pairing* pada Laptop atau PC (*Personal Computer*), setelah *pair* maka pengguna laptop atau PC (*Personal Computer*) sudah terkoneksi dengan sistem *printer* nirkabel.
b. Modul *Wi-Fi* juga digunakan sebagai alat komunikasi selain *Bluetooth* yang fungsinya juga untuk menerima data dokumen yang akan dikirimkan dari Laptop atau PC (*Personal Computer*). Bentuk komunikasi Wi-Fi tersebut menggunakan metode *client server*.

2. Proses (*Process*)

Pada bagian proses, *SBC (Single Board Computer)* digunakan sebagai perangkat penyimpan *driver printer* dan kontrol pada sistem. *SBC (Single Board*

Computer) akan memproses data dari input dan memberikan perintah *print* kepada perangkat *output* (*printer*) secara otomatis.

3. Keluaran (*Output*)

Bagian keluaran/output adalah bagian yang merupakan hasil ekseskusi perangkat. Output yang digunakan pada rancang sistem *printer* nirkabel adalah pengeksekusian *print* yang dikirimkan ke *printer* melalui piranti antar muka USB.

3.1.2. Cara Kerja Sistem

Sistem *printer* nirkabel akan bekerja apabila *driver printer* telah terinstal di dalam sistem. Setiap tipe *printer* memiliki *driver* yang berbeda, agar *printer* dapat mencetak perlu diinstalkan *driver* pada sistem. Sistem menerima data dokumen yang akan dicetak melalui komunikasi *Bluetooth* ataupun *Wi-Fi*.

Untuk komunikasi *Bluetooth*, sistem akan menjadikan *Bluetooth* tersebut untuk dijadikan sebagai akses poin. Sistem secara otomatis meminta perangkat pengguna untuk melakukan *pairing* apabila pengguna ingin terkoneksi melalui *Bluetooth*. Setelah *pair* sistem akan langsung terkoneksi dengan perangkat pengirim (Laptop/PC (*Personal Computer*). *User* yang dapat mengakses *printer* pada komunikasi *bluetooth* hanya satu user dikarenakan komunikasi pada *bluetooth poin to poin*. Kemudian sistem sudah dapat menerima data yang dikirimkan oleh pengguna. Indikator *Bluetooth* dijelaskan pada tabel di bawah ini.

Tabel 3. 1 Indikator *Bluetooth*

Kondisi	**Indikator**
Bluetooth Standby	*Blinking*
Pairing	*Blinking Fast*
Connected	*Blinking Fast*
Disconnected All Client	*Blinking*

Untuk komunikasi *Wi-Fi*, modul *Wi-Fi* juga sama seperti *Bluetooth* yaitu sistem menjadikan modul *Wi-Fi* sebagai akses poin. Sistem menyediakan jaringan akses dengan IP *address* yang sudah ditentukan oleh sistem.

Sistem akan menerima semua data yang masuk, namun pada saat *print out* data yang pertama yang akan dieksekusi terlebih dahulu, sedangkan data yang kedua dan seterusnya akan disimpan dahulu sampai data yang pertama selesai di *print out*. Kemudian data selanjutnya yang akan dieksekusi dan seterusnya. Setelah sistem selesai melakukan *print out*, sistem akan *standby* kembali untuk menerima data yang dikirimkan pengguna laptop atau PC (*Personal Computer*) untuk dicetak.

3.2. Pemilihan Komponen

Pemilihan jenis komponen dalam perancangan suatu perangkat elektronik dilakukan bertujuan untuk menentukan kualitas bahan, desain alat dan biaya yang akan digunakan karena berdampak langsung pada tingkat efisiensi dan efektifitas perangkat yang akan dibuat.

Dalam perancangan dan pembuatan rancang bangun sistem *printer* nirkabel menggunakan *SBC (Single Board Computer)* dan media *wireless* menggunakan *Wi-Fi* dan *Bluetooth*, proses pemilihan komponen yang akan digunakan harus diperhatikan agar biaya yang digunakan seminimal mungkin dengan hasil yang maksimal.

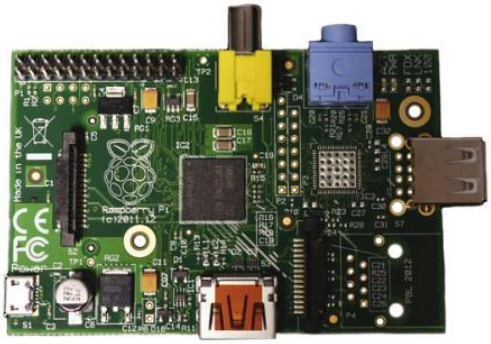
3.2.1. Pemilihan Jenis *SBC (Single Board Computer)*

Jenis *SBC (Single Board Computer)* yang digunakan pada perancangan sistem *printer* nirkabel harus memiliki kriteria berdasarkan sistem yang akan dibuat, sebagai berikut.

- Mampu menyimpan *driver printer* dan program yang akan dirancang.
- *Support* dengan *printer* yang akan dipasangkan pada sistem yang akan dirancang.

Tabel 3.2 Uraian Jenis *SBC (Single Board Computer)*

Spesifikasi	**Jenis *SBC (Single Board Computer)***		
	Raspberry Rev.A	**Raspberry Rev.B**	**Cubieboard 1G**
Prossesor	ARMv7 700 Mhz	ARMv7 700 Mhz	ARM cortex-A8 1 GHz
RAM	256 MB	512 MB	1 GB
GPIO	*Yes*	*Yes*	*No*
SDCARD	*Up to* 32 GB	*Up to*32 GB	*Up to* 64 GB
Power Source	5 VDC	5 VDC	5 VDC
Harga	Rp.456.000	Rp.540.000	Rp.750.000



Gambar 3. 2 Raspberry Pi Rev.B

Berdasarkan Tabel 3.2 dapat ditarik kesimpulan bahwa jenis *SBC (Single Board Computer)* yang sesuai dengan porsinya adalah Raspberry Rev.B. Prosesor dan RAM berpengaruh terhadap kecepatan ekseskusi perintah, Prosesor sebagai otak pada *SBC (Single Board Computer)* yang akan mengeksekusi perintah, besarnya frekuensi *clock* pada prosesor berpengaruh terhadap kecepatan eksekusi perintah. Pada saat sistem *standby* maka membutuhkan RAM sebagai memori penunjang agar prosesor tidak bekerja keras dalam mengeksekusi perintah. Besarnya RAM berpengaruh terhadap kinerja *SBC (Single Board Computer)*,

karena apabila eksekusi perintah terlalu banyak dan memiliki RAM yang kecil, *SBC (Single Board Computer)* akan mengalami *hang* dan tidak dapat mengeksekusi perintah. Tetapi, dengan hanya memiliki prosesor 700 MHz dan RAM sebesar 512 MB sudah dapat mengeksekusi perintah *print out* kepada *printer* yang digunakan.

3.2.2. Pemilihan Modul *Bluetooth*

Bluetooth merupakan salah satu komponen yang penting pada perancangan sistem ini, untuk itu *Bluetooth* harus memenuhi kriteria sebagai berikut.

- Memiliki kestabilan konektivitas
- Mampu sebagai akses poin dalam sebuah konektivitas

Tabel 3. 3 Uraian Perbandingan Jenis *Bluetooth*

Spesifikasi	**Jenis Komponen**		
	BT Cliptec RZB828	**BT v2.0 ES-388**	**BT Mini Cliptec V.2.1 RZB737**
Sensitivitas	-80dBm	-80dBm	-80dBm
Effective Range	*Up to* 10 m	0 ~ 10 m	0 ~ 10 m
Connection Mode	*Master and Slave*	*Master and Slave*	*Master and Slave*
Bit Error Rate	0,1%	0,1%	0,1%
Data Transfer	2 Mbps	2 Mbps	2-3 Mbps
Dimensi	20.4x10.1x2.8 mm	51.1x15.4x7.2 mm	20.4 x10.1x 2.8 mm
Harga	Rp. 83.000,-	Rp.28.000,-	Rp. 50.000,-



Gambar 3. 3 Modul *Bluetooth* BT v2.0 ES-388

Pada Tabel 3.3 perbandingan jenis *Bluetooth* secara umum memiliki keunggulan yang sama pada spesifikasinya. Perbedaannya terlihat pada dimensi dan harga yang ditawarkan. *Bluetooth* v2.0 ES-388 sudah dapat dijadikan sebagai media penghubung karena dengan memiliki *data rate* sebesar 2 Mbps sudah mampu untuk menerima file tanpa mengalami gangguan saat proses penerimaan file dan sistem sudah dapat berfungsi sesuai dengan rancangan yang akan dibuat.

3.2.3. Pemilihan Modul Wi-Fi

Modul *Wi-Fi* berfungsi sebagai media nirkabel selain *Bluetooth* untuk menerima dokumen yang akan dicetak. Untuk itu modul yang digunakan harus memenuhi ktriteria sebagai berikut.

- Memiliki konektivitas yang stabil.
- Mampu sebagai akses poin dalam sebuah jaringan.
- *Support* dengan *SBC (Single Board Computer)* yang digunakan.

Tabel 3. 4 Uraian Perbandingan Jenis Modul *Wi-Fi*

Spesifikasi	**Jenis Modul Wi-Fi**		
	Ralink RT5370	**TP-LINK TL-WN725N**	**Edimax EW-7811Un**
Wireless Standard	IEEE 802.11b/g & 802.11n	IEEE 802.11b, IEEE 802.11g, IEEE 802.11n	IEEE 802.11b/g/n
Operating Frequency	2.412~2.4835 GHz	2.400~2.4835GHz	2.400~2.4835GHz
Data Rate	11b: 11Mbps 11g: 54Mbps	11b: Up to 11Mbps 11g: Up to 54Mbps 11n: Up to 150Mbps	11b: Up to 11Mbps 11g: Up to 54Mbps 11n: Up to 72Mbps 11n: Up to 150Mbps
Dimensi	16.4x14.7x58mm	18.6x15x7.1mm	18.6x14.9x7.1mm
Security	WEP 64/128bit , WPA, WPA2, 802.1x, and 802.11i	64/128 WEP, WPA/WPA2, WPA-PSK/WPA2-PSK	64/128 WEP, WPA/WPA2 and 802.1x
Harga	**Rp. 100.000,-**	**Rp. 110.000,-**	**Rp. 135.000,-**



Gambar 3. 4 Modul Wi-Fi Ralink RT5370

Pada Tabel 3.4 perbedaan antara ketiganya terdapat pada *data rate*, TPLINK TL-WN725N dan Edimax EW-7811Un memiliki *data rate* hingga 150Mbps, sedangkan Ralink RT5370 hanya 10 Mbps. Jaringan Wi-Fi sifatnya *point to multipoint* dan modul *Wi-Fi* ini harus memiliki kestabilan konektivitaas yang sangat baik agar data yang diterima tidak *corrupt* atau mengalami gangguan. Modul *Wi-Fi* ini dijadikan sebagai akses poin, tetapi dengan *data rate* minimal 10 Mbps sudah dapat dijadikan akses poin dan dapat bekerja sesuai dengan fungsinya. Maka modul yang digunakan ialah Ralink RT5370.

3.3. Perancangan Perangkat Lunak

Perancangan perangkat lunak pada perangkat sistem *printer* nirkabel dirancang agar sistem yang dirancang dapat mengolah dan mengekseskusi perintah sesuai dengan yang harapan. Rancangan yang dilakukan dengan memasukkan beberapa konfigurasi modul yang terpasang pada SBC (*Single Board Computer*), menginstalkan aplikasi dan konfigurasi jaringan pada sistem SBC (*Single Board Computer*).

3.3.1. Konfigurasi *Wi-Fi*

Modul *Wi-Fi* yang dipasang pada SBC (*Single Board Computer*) belum terkonfigurasi dan belum bisa digunakan. Agar modul *Wi-Fi* dapat diaktifkan dan dapat berfungsi sesuai fungsinya. Pada saat system baru dinyalakan modul-modul yang terpasang pada SBC (*Single Board Computer*) perlu diregistrasi ulang, untuk itu *sistem* perlu ditambahkan script sebagai berikut.

allow-hotplug wlan0

iface wlan0 inet static

Script di atas berfungsi untuk mendeteksi perangkat tambahan yaitu modul *Wi-Fi* yang dipasangkan kepada sistem yang sudah berjalan. Sehingga setelah perangkat tersebut diregistrasi pada sistem, maka sistem dapat mengenali perangkat tambahan modul *Wi-Fi* yang dipasangkan tersebut. Sedangkan iface wlan0 inet static untuk meng-*assign* IP *address* dari perangkat tersebut dengan IP *static*. Selanjutnya untuk mengaktifkan *Wi-Fi* sebagai akses poin dengan menambahkan *script* sebagai berikut.

allow-hotplug mon.wlan0

iface mon.wlan0 inet static

Script di atas berfungsi untuk menjadikan perangkat modul *Wi-Fi* yang terpasang pada sistem untuk dijadikan sebagai akses poin, sehingga *client* dapat mengakses jaringan tersebut. Untuk akses poin tersebut juga meng-*assign* IP *static* dengan menambahkan iface mon.wlan0 inet static.

3.3.2. Konfigurasi Akses Poin *Wi-Fi*

Modul *Wi-Fi* yang dipasang pada SBC (*Single Board Computer*) belum terkonfigurasi sebagai akses poin (AP). Modul *Wi-Fi* dapat berfungsi sesuai fungsinya yaitu sebagai akses poin dan dapat diakses *client* dengan penambahan *script* pada konfigurasi hostapd sebagai berikut.

 # SSID to be used in IEEE 802.11 management frames

ssid=Printer_Server

SSID (*Service Set Identifier*) merupakan penamaan pada akses poin sistem *printer* tanpa kabel ini. *Client* dapat terkoneksi dengan sistem ini dengan cara mengkoneksikan ke dalam jaringan *Wi-Fi* dengan ssid *Printer*_Server.

Jaringan pada sistem ini dapat diproteksi menggunakan *password* dengan mengkonfigurasikan sebagai berikut.

wpa=1

wpa_passphrase=printer01

WPA (*Wi-Fi Protected Access*) adalah suatu sistem yang juga dapat diterapkan untuk mengamankan jaringan nirkabel. WPA = 1 (*Broadcast of ESSID*) dimaksudkan untuk men-*share* jaringan *Wi-Fi* dengan penambahan pengaman yang berupa *password* teks maupun kombinasi dengan angka. Pengaman dapat dinonaktifkan atau dihilangkan (*unsecured*) pada jaringan *Wi-Fi* tersebut dengan cara menambahkan karakter # pada awal setiap *script* tersebut, sehingga *script* tersebut hanya menjadi *comment*.

3.3.3. *Script Install* CUPS dan Konfigurasi Samba

CUPS singkatan dari *Common Unix Printing System*, sebuah sistem *printing* bermodul yang menjadikan SBC (*Single Board Computer*) sebagai *Server Printer*. Berikut ini cara menerapkan CUPS pada sistem *printer* tanpa kabel.

sudo apt-get install cups

sudo usermod –aG lpadmin pi

sudo cupsctl –share-printers –remote-printers –remote admin

sudo / etc/init.d/cups start

Awal penerapan CUPS (*Common Unix Printing System*) pada sistem ini dengan cara menginstalkan CUPS (*Common Unix Printing System*) tersebut kemudian mengkonfigurasikan untuk melakukan otorisasi manajemen *printer*. Setelah terkonfigurasi selanjutnya mengaktifkan *sharing printer* dan kemudian menjalankan CUPS tersebut.

Sedangkan Samba adalah implementasi dari SMB protokol yaitu protokol yang digunakan oleh sistem operasi MS dan Linux seperti linux pada LX *Terminal*-nya yang berfungsi untuk men-*sharing file* dan *printer*. Dengan SMB, *client* dapat mengakses *printer* yang di-*share* oleh sistem dengan mengkonfigurasi sebagai berikut.

[homes]

comment = Home Directories

browseable = no

[pi]

comment = Home Directories of pi

browseable = no

[printers]

comment = All Printers

browseable = no

path = /var/spool/samba

printable = yes

read only = yes

[print$]

comment = Printer Drivers

path = /var/lib/samba/printers

browseable = yes

read only = yes

guest ok = no

[Driver Printer]

comment = Data share

path = /home/pi/Driver Printer

browseable = yes

read only = yes

Samba akan men-*share* semua direktori pada SBC (*Single Board Computer*), untuk i*tu browseable* pada direktori *home* dan pi sama dengan *no* (*disable*) agar hanya *printer* dan *driver printer* yang di *share*.

Script di atas berfungsi untuk men-*share printer* dan *file driver printer* yang terdapat dalam sistem, sehingga *client* dapat mengakses *printer* yang akan digunakan pada sistem dan *driver printer* digunakan apabila *client* membutuhkan *driver* pada saat menambahkan *printer* yang akan digunakan. Konfigurasi *Bluetooth*

Pada sistem *printer* tanpa kabel, *Bluetooth* dikonfigurasikan untuk dijadikan sebagai akses poin. *Bluetooth* dapat dijadikan sebagai akses poin dengan menambahkan aplikasi menggunakan pemograman phyton berikut ini pada sistem.

```
import os
import sys
import dbus
import dbus.service
import dbus.mainloop.glib
import gobject
from optparse import OptionParser


pin_code = "0000"


class Agent(dbus.service.Object):
```

```
@dbus.service.method("org.bluez.Agent", in_signature="o",
out_signature="s")

def RequestPinCode(self, device_path):

device = dbus.Interface(bus.get_object("org.bluez", device_path),
"org.bluez.Device")

properties = device.GetProperties()

print "Pairing and trusting device %s [%s]" % (properties["Alias"],
properties["Address"])

device.SetProperty("Trusted", dbus.Boolean(True))

return pin_code


if __name__ == '__main__':


if (os.getuid() != 0):

print "You must have root privileges to run this agent. Try 'sudo pinaple-
agent [--pin <PIN>]'"

raise SystemExit


parser = OptionParser()

parser.add_option("-p", "--pin", action="store", dest="pin_code",
help="PIN code to pair with", metavar="PIN")

 (options, args) = parser.parse_args()


# use the pin code if provided

if (options.pin_code):
```

```
pin_code = options.pin_code


# get the dbus system bus
dbus.mainloop.glib.DBusGMainLoop(set_as_default=True)
bus = dbus.SystemBus()
# get the Bluez manager and default Bluetooth adapter
manager = dbus.Interface(bus.get_object("org.bluez", "/"),
"org.bluez.Manager")
adapter_path = manager.DefaultAdapter()
adapter = dbus.Interface(bus.get_object("org.bluez", adapter_path),
"org.bluez.Adapter")
```

Script aplikasi dibawah ini berfungsi untuk mem-*broadcast bluetooth* agar dapat dikenali dan diakses oleh *client*. Secara otomatis *Bluetooth* akan *standby* dan apabila *client* melakukan koneksi melalui *Bluetooth* maka sistem akan meminta untuk melakukan *pairing*.

```
# set the adapter to discoverable
print "Making Bluetooth adapter discoverable"
adapter.SetProperty("Discoverable", dbus.Boolean(True))


# register the pinaple agent
agent = Agent(bus, agent_path)
adapter.RegisterAgent(agent_path, "KeyboardDisplay")
print "Waiting to pair devices...."
```

mainloop = gobject.MainLoop()

try:

mainloop.run()

except KeyboardInterrupt:

print "\nMaking Bluetooth adapter undiscoverable"

adapter.SetProperty("Discoverable", dbus.Boolean(False, variant_level=1))

adapter.UnregisterAgent(agent_path)

raise SystemExit

Aplikasi diatas berfungsi untuk mengkonfigurasikan modul *bluetooth* sebagai akses poin. Konfigurasi ini menentukan *discoverable Bluetooth* dan *pairing*. Seperti pada *Wi-Fi, Bluetooth* juga akan *standby* dan menunggu *client* yang akan melakukan koneksi melalui *bluetooth* dan kemudian melakukan *pairing* secara otomatis.

3.3.4. Konfigurasi IP *Address*

IP *address* merupakan alamat yang terdapat pada sebuah jaringan. Jaringan sistem *printer* nirkabel mempunyai IP *address* yang berfungsi sebagai alamat sebuah *server* yang bertujuan agar *client* dapat terkoneksi pada jaringan sistem *printer* nirkabel. Berikut ini konfigurasi IP *address Wi-Fi, tethering* dan *Bluetooth* dengan menambahkan *script* pada konfigurasi *interfaces*.

#Konfigurasi IP address Wi-Fi

allow-hotplug wlan0

iface wlan0 inet static

address 192.168.2.1

netmask 255.255.255.0

gateway 192.168.2.1

#Konfigurasi IP address akses poin Wi-Fi

allow-hotplug mon.wlan0

iface mon.wlan0 inet static

address 192.168.2.24

netmask 255.255.255.0

gateway 192.168.2.1

#Konfigurasi IP address NAP (Network Access Protocol) Bluetooth

allow-hotplug bnep0

iface bnep0 inet static

address 192.168.3.24

netmask 255.255.255.0

gateway 192.168.3.1

iface default inet dhcp

Script di atas adalah konfigurasi IP *address* sebagai alamat modul *Wi-Fi* dan modul *Bluetooth* sebagai akses poin. IP *address* tersebut adalah IP *address static*, IP *address* dijadikan *static* agar IP *address printer* yang di *share* tidak berubah ketika *client* memutuskan koneksi atau sistem di-*restart* sehingga *client*

tidak perlu menambahkan ulang *printer* apabila koneksi terputus atau *reconnet* pada jaringan sistem *printer* tanpa kabel.

3.3.5. Konfigurasi IP *Address Client* DHCP

Konfigurasi IP *address client* DHCP (*Dynamic Configuration Protocol*) berfungsi agar pada sisi *client* tidak perlu menginput IP *address* secara manual. Konfigurasi dilakukan dengan cara menambahkan *script* di bawah ini.

interface=wlan0, bnep0

dhcp-range=192.168.2.30, 192.168.2.130, 255.255.255.0

dhcp-range=192.168.3.30, 192.168.3.31, 255.255.255.0

Script di atas ditambahkan pada /etc/dnsmasq.conf. IP *address* tersebut memiliki *range user* ID pada setiap masing-masing media komunikasi. *Range* IP *address* dapat diatur sesuai keinginan dengan cara mengubah *range* pada *dhcp-range* nya.

3.3.6. *Startup Script*

Startup script berfungsi untuk meng-*assign environment* sistem linux ketika sistem pertama kali dinyalakan, maka sifatnya paten sampai sistem tersebut dimatikan. Jadi ketika sistem dinyalakan kembali sistem akan meng-assign SBC (*Single Board Computer*) sebagai *printer server*.

 # reconfigure wireless access point for wi-fi dongle & Bluetooth dongle

 # setup wi-fi ip address on 192.168.2.24 gateway 192.168.2.1

 # setup Bluetooth NAP ip address on 192.168.3.24 gateway 192.168.3.1

printf "\r\nReconfigure networking parameters ...\n"

```
printf "Setting-up wi-fi dongle on 192.168.2.1\n"

_setup_wifi=`sudo ifconfig wlan0 192.168.2.1`

_setup_wifi=`sudo ifup wlan0`

printf "$_setup_wifi\n"

printf "Update networking parameters ...\n"

_reconfig=`sudo service networking restart`

printf "$_reconfig\n"

printf "Setup Wi-fi access-point address on 192.168.2.24\n"

_setup_wifi_ap=`sudo ifconfig mon.wlan0 192.168.2.24`

_setup_wifi_ap=`sudo ifup mon.wlan0`

printf "$_setup_wifi_ap\n"

_ifconfig_all=`ifconfig -a`

printf "$_ifconfig_all\n"

printf "System is ready ....\n\r"

_start_Bluetooth=`sudo hciconfig hci0 name Printer_Server`

printf "$_start_Bluetooth"

printf "Make Bluetooth discoverable and pairable\n"

printf "Pair the bluetooh using '1234' on your device.\n\r"

printf "\r\n"

#_start_Bluetooth=`sudo pinaple-agent -p 1234`

#printf "$_start_Bluetooth\n"


exit 0
```

Script yang ditambahkan ini pada /etc/rc.local berfungsi untuk mengkonfigurasikan dua buah *network* yang kerja secara bersamaan tapi berada pada dua buah IP yang berbeda, konfigurasi ini dibuat karena pada umumnya kernel hanya mengkonfigurasikan untuk satu buah *network* yang bekerja dalam satu waktu. Agar kedua *network* tersebut berjalan sesuai dengan fungsinya, maka pada bagian *network interfaces* yang sebelumnya dikonfigurasikan sendiri bentuk konfigurasi *startup*-nya adalah seperti pada *script* di atas.

3.3.7. *Login Script*

Login script digunakan untuk meng-*assign environment* sistem ketika membuka suatu *environment shell* (contohnya: lx-terminal adalah aplikasi shell_gui). Untuk setiap kali membuka aplikasi lx-terminal maka ".bashrc" akan selalu dijalankan terlebih dahulu sehingga *environment* sistem siap untuk selanjutnya untuk digunakan.

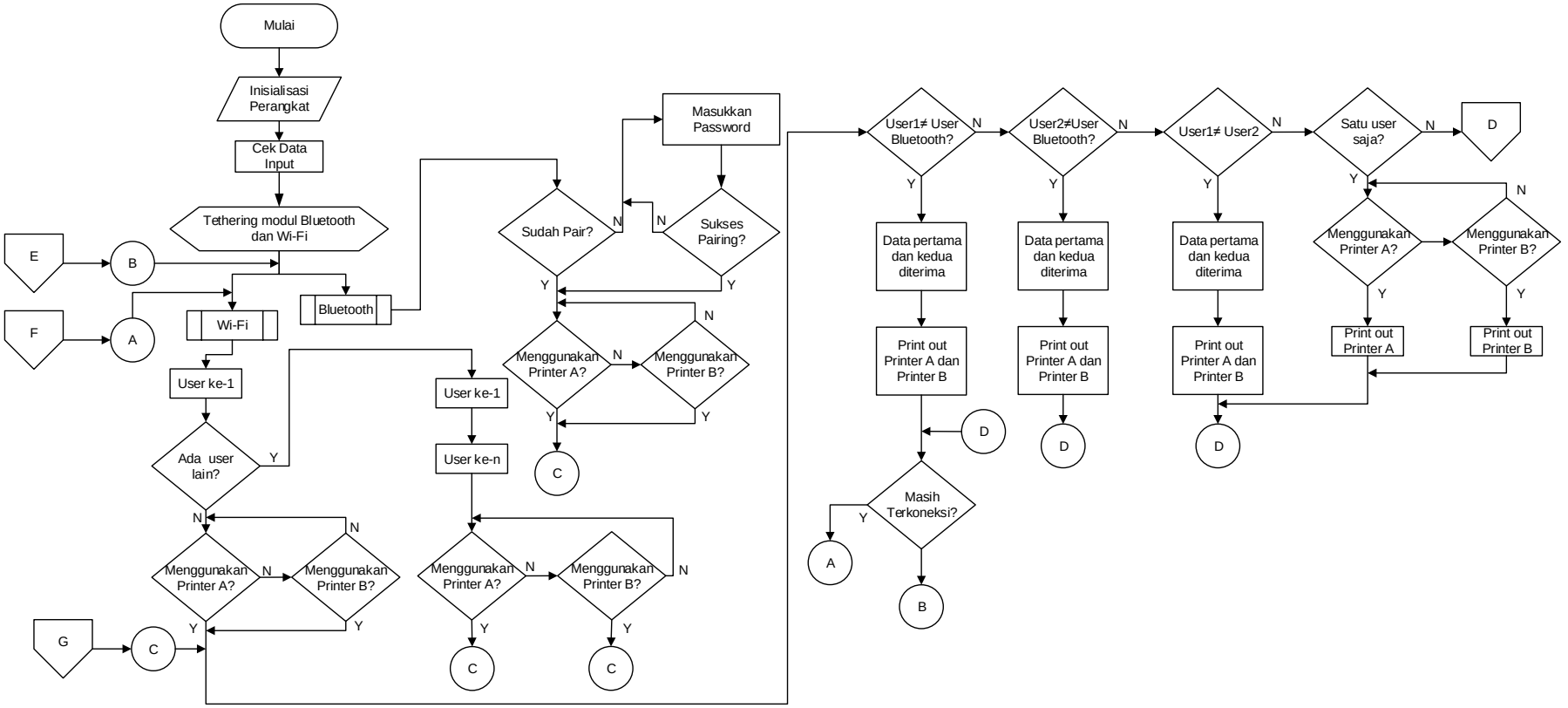
```
# add to make Bluetooth discoverable and pairable for other users
#printf "\r\n\r"
#sudo service samba restart
printf "\r\n\r"
sudo pinaple-agent -p 1234
```

Pada .bashrc ditambahkan *script* di atas untuk memanggil aplikasi phyton yang fungsi aplikasi tersebut untuk mengkonfigurasikan modul *Bluetooth* sebagai akses poin pada saat sistem baru dinyalakan. Sehingga pada saat sistem dinyalakan secara otomatis sistem linux akan membaca .bashrc yang kemudian modul

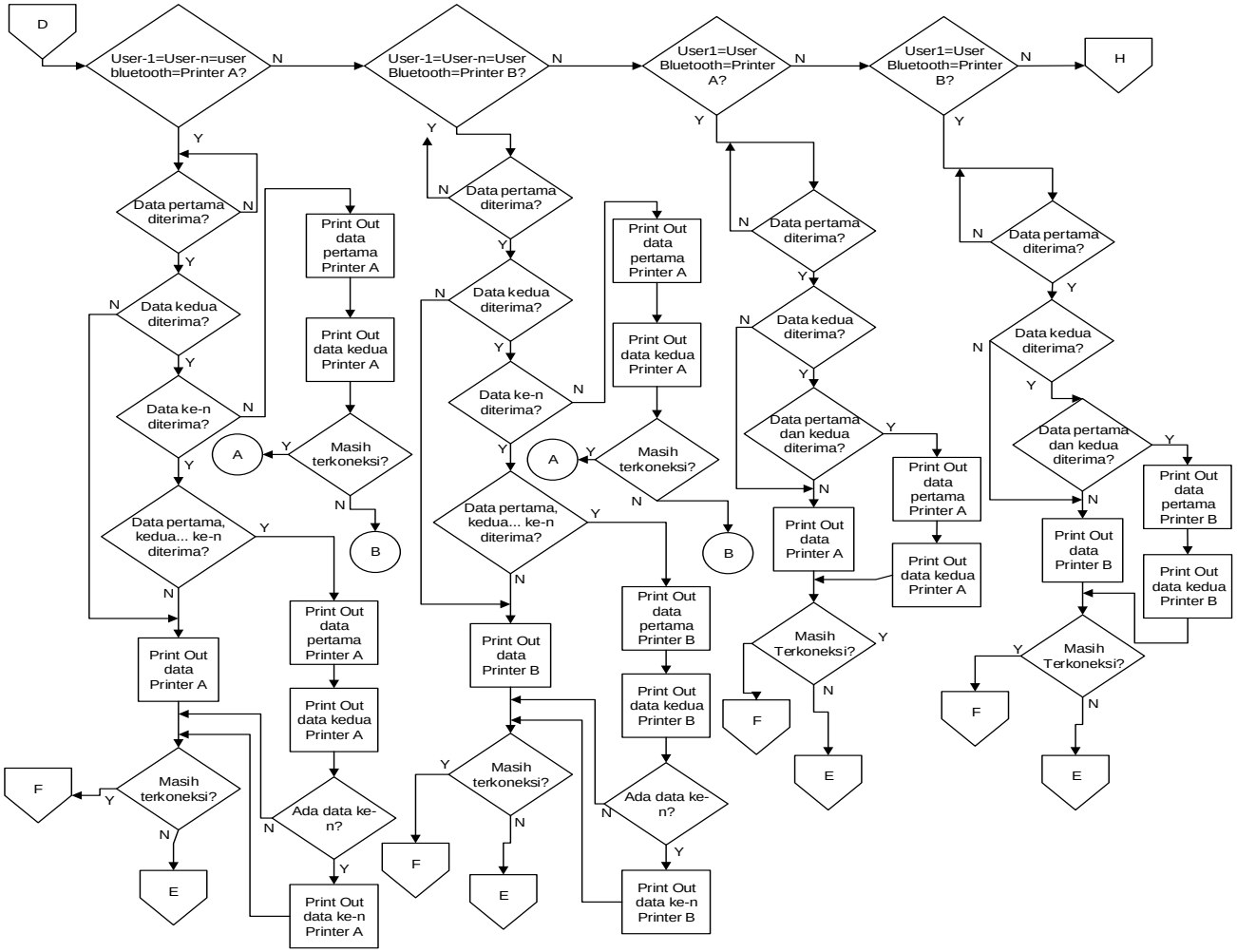
*Bluetooth t*ersebut dijadikan akses poin dan secara otomatis akan meminta *pairing* apabila *client* ingin mengakses *printer* yang di *share* melalui *Bluetooth*.

3.4. Perancangan Sistem

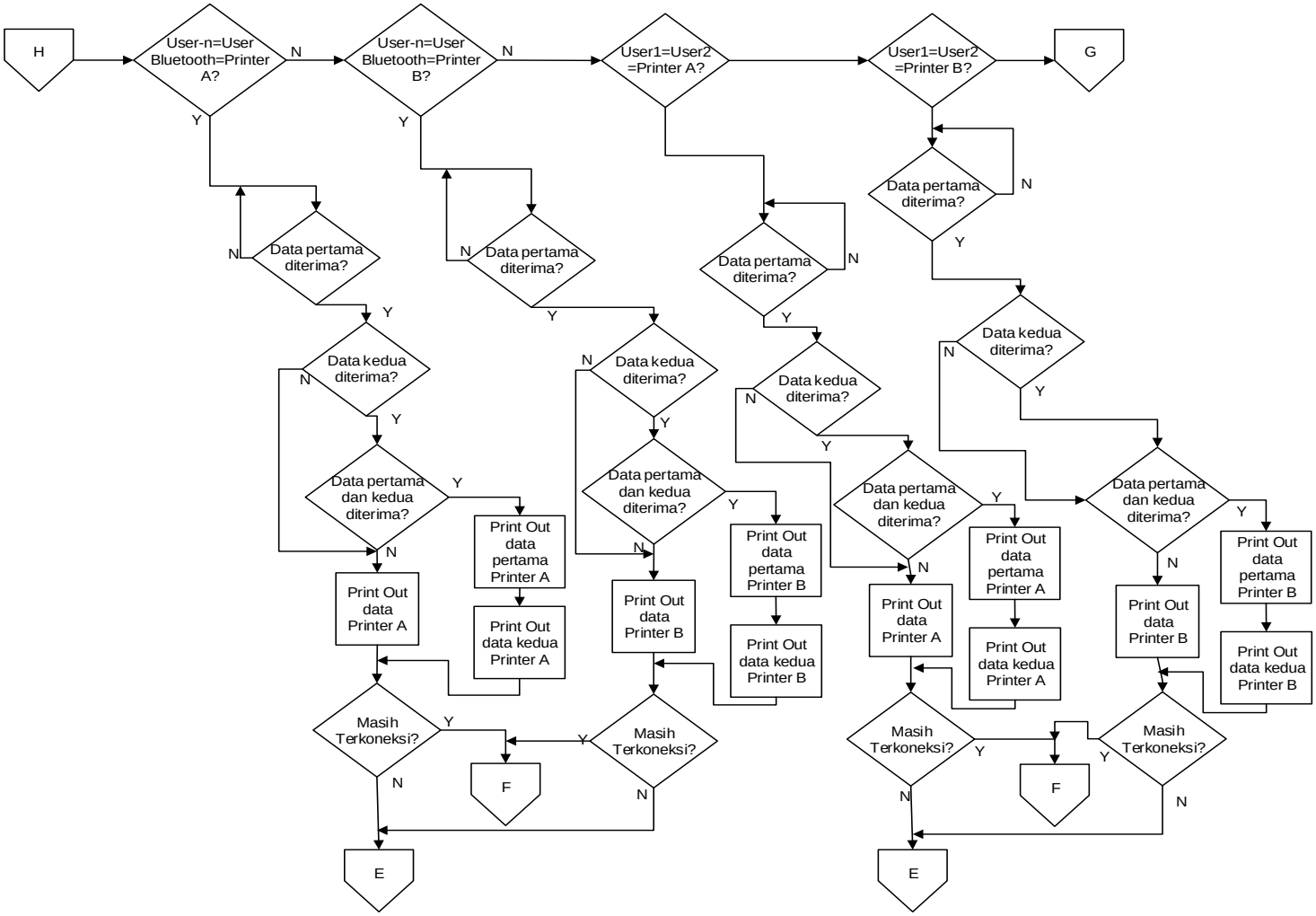
Perancangan perangkat lunak dibahas dengan menggunakan diagram alir (*flowchart*). Setiap masukan (*input*) yang diterima akan diatur pada perangkat yang kemudian akan diproses untuk menentukan pada bagian keluaran (*output*). Berikut ini alur sistem yang akan dirancang.



Gambar 3.5 Diagram Alir Sistem Rancang Bangun *Printer* Nirkabel (Bagian 1)



Gambar 3. 6 Diagram Alir Sistem Rancang Bangun *Printer* Nirkabel (Bagian 2)



Gambar 3. 7 Diagram Alir Sistem Rancang Bangun *Printer* Nirkabel (Bagian 3)

Pada Gambar 3.5 merupakan diagram alir rancang bangun sistem *printer* nirkabel. Berikut penjelasan dari diagram alir tersebut.

Pada awal diagram alir, SBC (*Single Board Computer*) membaca *hardware* yang terpasang, contohnya modul *Bluetooth*, modul *Wi-Fi* dan *printer* yang terpasang pada sistem. Kemudian *SBC (Single Board Computer)* membaca *software* yang sudah dipasang didalam *SBC (Single Board Computer)* termasuk membaca *driver printer*.

Setelah kondisi sistem siap *SBC (Single Board Computer)* secara otomatis akan menjadi *printer server*, modul *Wi-Fi* yang dijadikan akses poin oleh sistem akan *standby* untuk menerima *client* yang akan masuk pada jaringan yang disediakan oleh sistem *printer* nirkabel.

Pada rancang bangun sistem *printer* tanpa kabel ini, modul *Bluetooth* juga dijadikan akses poin oleh sistem, walaupun dijadikan akses poin sistem hanya dapat menerima satu user saja dikarenakan komunikasi *bluetooth point to point*. Kemudian sistem secara otomatis akan meminta *pair* perangkat pengguna laptop/PC (*Personal Computer*) apabila perangkat pengguna mengakses *Bluetooth*. Setelah *pair* maka sistem secara otomatis terkoneksi dengan perangkat pengguna. Kemudian sistem dapat menerima data yang dikirimkan pengguna melalui *Bluetooth*.

Sistem menerima semua data yang dikirimkan melalui *Bluetooth* ataupun jaringan *Wi-Fi*, kemudian mengeksekusi data tersebut untuk di *print out*. Sistem akan menerima semua data yang masuk dan melakukan *print out* secara otomatis, namun pada saat *print out* data yang pertama yang akan dieksekusi terlebih dahulu

sedangkan data yang kedua dan seterusnya akan disimpan dahulu sampai data yang pertama selesai di *print out*. Kemudian data selanjutnya yang akan dieksekusi dan seterusnya. Setelah sistem selesai melakukan *print out*, sistem akan *standby* kembali untuk menerima data yang dikirimkan pengguna laptop atau PC (*Personal Computer*) untuk dicetak. Sedangkan untuk melakukan *print out* secara bersamaan dengan dua buah *printer* yang berbeda, maka sistem akan melakukan *print out* secara bersamaan pada kedua *printer* tersebut.

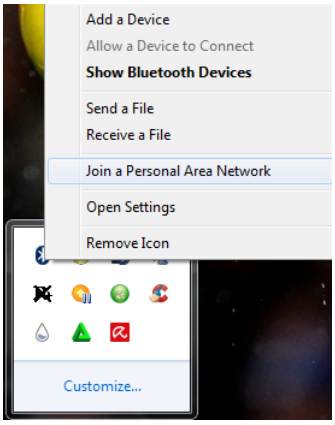
3.5. Tata Cara Penggunaan Perangkat

Dalam penggunaan alat perlu dijelaskan tahap-tahap cara penggunaan alat tersebut agar *user*/*client* dapat menggunakan perangkat tersebut sesuai dengan fungsinya. Dibawah ini akan dijelaskan cara penggunaan alat melalui media komunikasi *bluetooth* dan *Wi-Fi*.

3.5.1. Penggunaan Melalui Media Komunikasi Bluetooth

Komunikasi *bluetooth ready*/*standby* saat indikator *bluetooth blinking* dan *user* sudah dapat melakukan koneksi melalui media komunikasi *bluetooth* dengan cara sebagai berikut.

a. *User* mengklik *icon bluetooth* dan memilih *Join a Personal Area Network*.



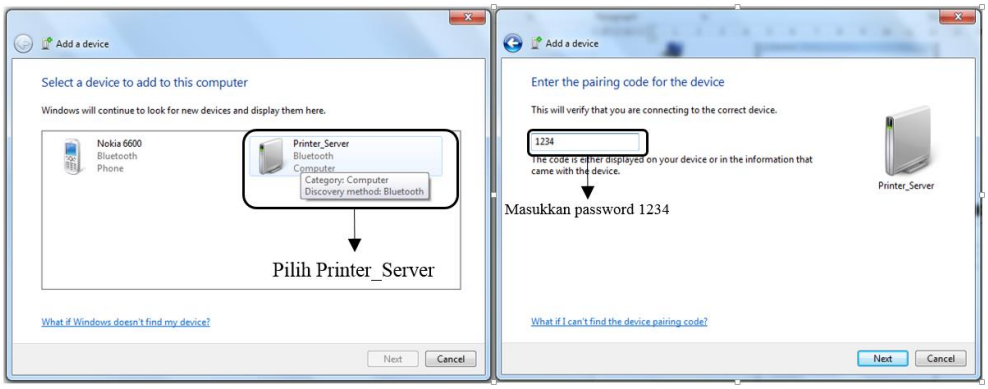
Gambar 3.8 *Join a Personal Area Network*

b. Kemudian *user* mencari perangkat *Printer*_Server dengan cara mengklik *add a device* pada *Bluetooth Personal Area Network*.



Gambar 3. 9 *Add a device*

c. Pilih Perangkat *Printer*_Server dan melakukan *pairing* dengan memasukkan *password* 1234.

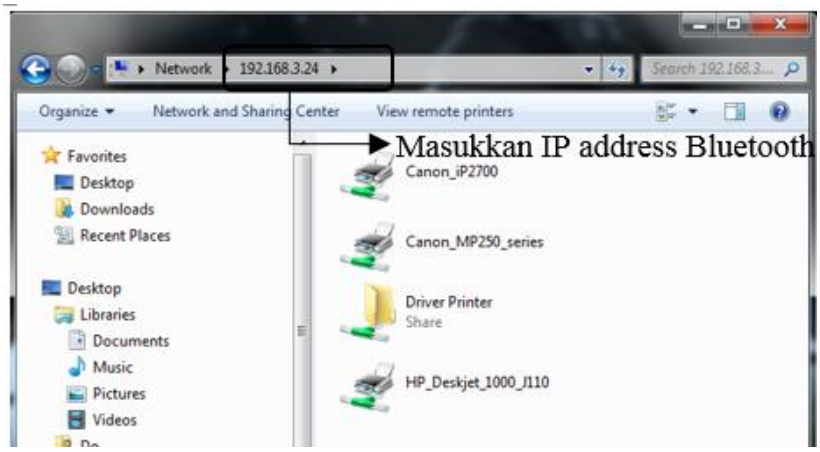


Gambar 3. 10 *Pairing* Perangkat *Printer*_Server

d. Pilih *Connect using* dan pilih *access point* sehingga *client* sudah dapat terkoneksi dengan perangkat

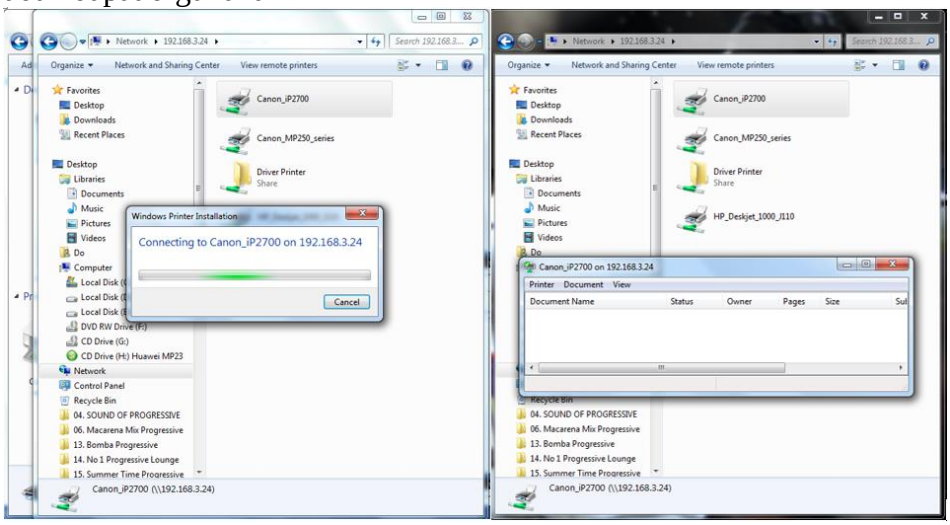


e. Kemudian masuk Windows Explorer dan masuk ke *network* untuk memilih *printer* yang di-*share* melalui komunikasi *bluetooth*. Pada *address bar*, masukkan IP *address bluetooth* agar tidak terjadi *cross* dengan komunikasi *Wi-Fi*. Apabila *user* tidak memiliki *driver printer*, *user* dapat mengambil *driver* dengan cara *browse* pada perangkat *Printer*_Server.



Gambar 3. 11 Pemilihan *Printer* dan *Driver Printer*

Printer sudah dapat digunakan

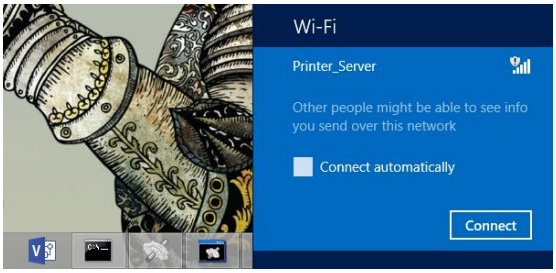


Gambar 3. 12 *Success Add Printer*

3.5.2. Penggunaan Melalui Media Komunikasi Wi-Fi

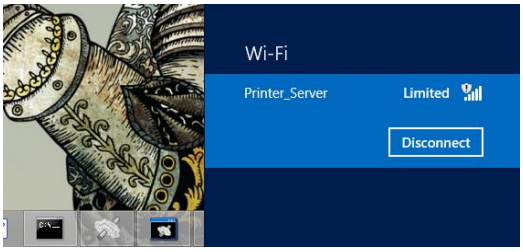
Komunikasi *Wi-Fi ready*/*standby* saat ssid Wi-Fi sudah terlihat pada laptop/PC (Personal Computer) user dan *user* dapat melakukan koneksi melalui media komunikasi *Wi-Fi* dengan cara sebagai berikut.

a. User mengklik ssid *Printer*_Server, lalu connect.



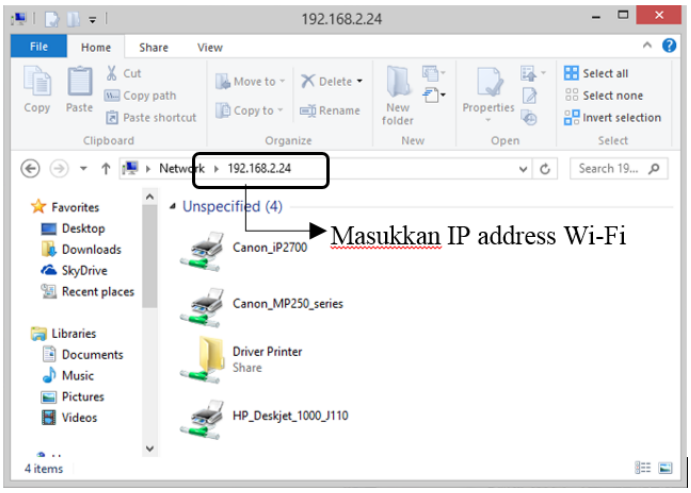
Gambar 3. 13 SSID *Printer Server*

b. Setelah berhasil melakukan koneksi maka akan tampil seperti gambar di bawah ini.



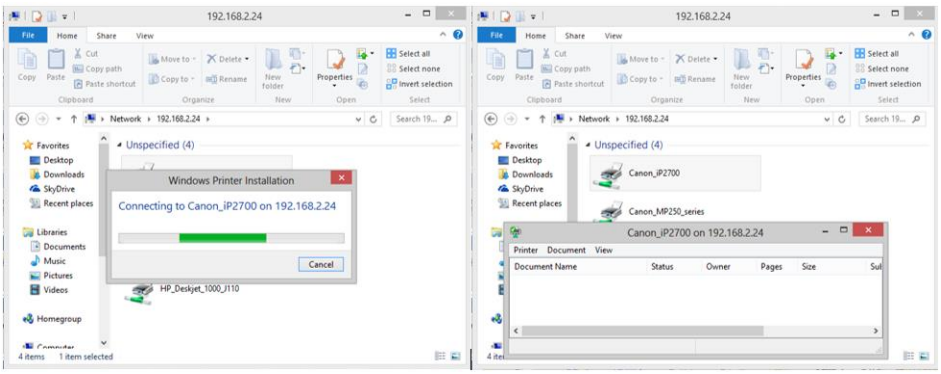
Gambar 3. 14 *Printer*_Server *Connect*

c. Kemudian masuk Windows Explorer dan masuk ke *network* untuk memilih *printer* yang di-*share* melalui komunikasi *Wi-Fi*. Pada *address bar*, masukkan IP *address Wi-Fi* agar tidak terjadi *cross* dengan komunikasi *Bluetooth*.



Gambar 3. 15 *Add Printer* Melalui Media Komunikasi *Wi-Fi*

d. Setelah berhasil *add printer* maka *printer* dapat digunakan.



Gambar 3. 16 *Success Add Printer*