RandA PhotoSharing

Hardware

RandA PhotoSharing is an application which will require the following components:

- RaspBerry Pi 2
- RaspBerry Camera
- SD Card (8GB or more)
- RandA
- Dedicated shield for RandA PhotoSharing
- HDMI monitor/TV
- USB mouse and keyboard
- 3 buttons and 2 conducting poles
- Wired or wireless Internet connection (optional)

It'll also need:

- A GMail account to send emails and save photos to cloud
- An IFTTT account to post photos on various social networks (optional)
- Social networks accounts on which the photos will be posted (optional)

Assembly instructions:

1. Connect the RaspBerry Camera to RaspBerry
2. Connect RandA to RaspBerry
3. Connect mouse, keyboard ethernet cable or wireless stick which will be needed for the configuration part
4. Connect the 3 buttons to RandA as follows:
    a. Button "Next" → pin 8
    b. Button "OK" → pin 9
    c. Button "Prev" → pin 10
5. Power the buttons by connecting them to the 5V and GND pins on RandA
6. Connect the two metal poles to the 5V and A0 pins on RandA
7. Setup your SD card (we'll explain it in the next section) and insert it in your RaspBerry, power RandA with a 5V/2A USB cable, this will also power RaspBerry

Setup

You can download a preconfigured ISO image from Elettronica In's website and skip to the sketch upload section, otherwise you can download the latest Raspbian image from RaspBerry's website and follow the step-by-step setup.
If you choose the preconfigured way, remember to edit the */etc/sstmp.conf* file as suggested into the emailing libraries setup.

Burn the RaspBian image on the SD card using Win32DiskImager which can be downloaded here: sourceforge.net/projects/win32diskimager/
Insert the SD card in your RaspBerry and power up RandA.

During the first boot you'll be prompted with the RaspBerry configuration screen, on which you'll have to accomplish the following:

1. Expand Filesystem
2. Enable Boot to Desktop/Scratch: select "Desktop Log in…"
3. Internalisation Options:
    a. Change Locale: deselect *en_GB.UTF-8 UTF-8* using SPACEBAR and select *it_IT ISO-8859-1* instead(setting it as "Default locale for the system environment")
    b. Change Timezone: select *Europe* and then *Rome*
    c. Change Keyboard Layout: select the keyboard model you're using(usually *Generic 105 keys (Intl) PC*), and then select *Other* and *Italian* as keyboard language, choose *Italian* as keyboard layout, finally confirm with *Ok* twice
4. Enable Camera: *Enable*
5. Advanced Options:
    a. SSH: *Enable*
    b. I2C: answer *Yes* to both questions

Move your cursor by pressing TAB to select <*Finish*> then confirm with *ENTER* to reboot.

The following configuration steps can be executed either via RaspBerry's GUI or via SSH using a client like Putty.

An internet connection will be necessary to proceed: if you're using an ethernet cable skip this section, otherwise plug your USB WiFi dongle in and follow these steps:

1. Open your terminal and run *dmesg | more*
2. Make sure that RaspBerry recognized the peripheral: press SPACE until you'll find a value like "Product: 802.11n WLAN Adapter ..." and then press Q to quit
3. Edit the network interfaces file typing: *sudo nano /etc/network/interfaces*
4. Insert the following lines to enable a WiFi connection:

```
auto lo
iface lo inet loopback
iface eth0 inet dhcp

allow-hotplug wlan0
auto wlan0

iface wlan0 inet dhcp
  wpa-ssid "Your WiFi network name"
  wpa-psk "Your WiFi network password"
```

5. Run the following command to reload the networking settings: *sudo service networking reload*
6. Using *ifconfig* you can check if the WiFi dongle is actually connected to the network and has been assigned an IP address.

Now that your RaspBerry has acquired internet access, run the following command to update the system and reboot:
sudo apt-get update && sudo apt-get upgrade && sudo rpi-update && sudo reboot
This will require some time and you'll have to confirm some packets' setup by pressing Y and then ENTER.

Download the latest RandA setup from https://github.com/open-electronics/RandA/releases
In the archive you'll find a readme that explains how to correctly setup RandA (InstallationREADME.TXT) and the setup files.

After having installed RandA and tested the serial communication with RaspBerry, you'll have to change Tomcat's listening port to avoid conflicts with Apache which we'll install later.
Tomcat is used to install RandA's control panel which is obsolete for our application.
On the other hand we'll use Apache to host a simple control panel for RandA PhotoSharing.
Change directory to the one containing Tomcat's configuration files:
cd /home/apache-tomcat-7.0.47/conf
Open the file server.xml: *nano server.xml* and edit the row "<*Connector port = "80" …"* replacing 80 with 8080.
Reboot RaspBerry: *sudo reboot*

Force RaspBerry to never disable its HDMI output by editing the following file:
sudo nano /etc/lightdm/lightdm.conf
Add the following row to the section [SeatDefaults]:
xserver-command=X -s 0 dpms
Close and save with CTRL+X, Y and ENTER.

Run the following commands in order to install Apache, PHP and the MySQL server we'll need to power the RandA PhotoSharing control panel.
sudo apt-get install apache2 apache2-doc apache2-utils
sudo apt-get install libapache2-mod-php5 php5 php-pear php5-xcache
sudo apt-get install php5-mysql
sudo apt-get install mysql-server mysql-client
During your setup some packets will require you to confirm their installation, press Y and then ENTER to confirm.
When you'll be prompted for a database password, type it in and remember it for later;
Install PHPMyAdmin to manage the database:
sudo apt-get install phpmyadmin
You'll be asked which web server you wish to automatically configure, select *apache2* using SPACEBAR, then confirm by pressing TAB and then ENTER.
Accept to configure PHPMyAdmin's database using dbconfig-common by answering "Yes" when you're asked.
When the setup is done, open apache2 config file:
sudo nano /etc/apache2/apache2.conf

And insert the following instruction at the bottom of the file:
Include /etc/phpmyadmin/apache.conf
Close and save by pressing CTRL+X, Y and ENTER.
Restart apache2 to load the latest changes:
sudo /etc/init.d/apache2 restart

Now you can install all the libraries necessary to the Python script by running the following commands in order:
sudo apt-get install python-dev
sudo apt-get install python-imaging-tk
sudo apt-get install python-mysqldb
sudo apt-get install imagemagick

Install the mail server and other utilities you'll need to sebd eMails using Python by running the following commands in order:
sudo apt-get install ssmtp
sudo apt-get install mailutils

Edit the ssmtp.conf file:
sudo nano /etc/ssmtp/ssmtp.conf
So that it'll look like this:

root=postmaster
mailhub=smtp.gmail.com:587
hostname=raspberrypi
AuthUser=**GmailAccount**@gmail.com
AuthPass=**GmaiPassword**
UseSTARTTLS=YES

Swap "GmailAccount" and "GmailPassword" with your email address and password.
Save and quit by pressing CTRL+X, Y and ENTER.

Edit the revaliases file:
sudo nano /etc/ssmtp/revaliases
Add the following row:
root:root@gmail:smtp.gmail.com:587
Save and quit by pressing CTRL+X, Y and ENTER.
Change permissions over the ssmtp.conf file by running:
sudo chmod 774 /etc/ssmtp/ssmtp.conf

Reboot:
sudo reboot

Now your RaspBerry is fully configured, proceed by downloading RandA PhotoSharing, move to the folder /var/www:
cd /var/www

Download RandA PhotoSharing by running:
git clone https://github.com/open-electronics/randaps.git
The downloaded folder contains:

- **admin/:** this folder contains all the web panel files
- **data/:** this folder contains all the pictures needed for the GUI (themes, overlays, logos, …)
- **photos:/** this folder contains all the pictures that will be taken
- **randaps.py:** RandA PhotoSharing python script
- **randaps_sketch.hex:** compiled RandA source code ready to be uploaded
- **randaps_sketch.ino:** non-compiled RandA source code
- **RandA-PhotoSharing.sql:** database installation
- **README.pdf:** complete guide about RandA PhotoSharing setup and utilization

Type in your web browser url-bar "*http://IP_RASPBERRY/phpmyadmin"* swapping "IP_RASPBERRY" with its effective IP address.
Login using *root* as your username and the password you chose during MySQL and PHPMyAdmin setup.
Click on the tab *Privileges*, click *Add a new user* and fill in the fields as follows:

- Username: [text field] *randaps*
- Host: [Local] *localhost*
- Password: [text field] *randaps*
- Re-type: *randaps*
- Database for user: Create a database with same name and grant all privileges
- Global privileges: Select all

Click on *Create user*.
On the right side of the screen you'll see the new database *randaps* (otherwise refresh the pace), click on it and then on the *Import* tab.
You have to import the SQL file located into the RandA PhotoSharing folder: select the file and click *Execute*.

Edit the RandA-PhotoSharing script:
sudo nano /var/www/randaps/randaps.py
In the "CUSTOMIZABLE VARIABLES" section instert your GMail password replacing "INSERT_YOUR_EMAIL_PASSWORD", without removing the quotes;
Close and save by pressing CTRL+X, Y and ENTER.

Run: *sudo visudo* and add the following line at the bottom of the files:
www-data ALL=(ALL) NOPASSWD: ALL
Close and save by pressing CTRL+X, Y and ENTER: this way we'll grant PHP the permission to act on the photos.

Upload the sketch on RandA running:
ArduLoad /var/www/randaps/randaps_sketch.hex

Now you'll have to configure all the accounts that RandA PhotoSharing will need to send the photos:

- **Gmail:** RandA PhotoSharing will use this account to send the pictures to IFTTT and to the user (if he fills in the email field)
- **IFTTT:** every time it'll receive an email from a known GMail account, it'll upload it on Google Drive (if enabled) and share it on Facebook or Twitter (if enabled)

Log into your GMail account and go to the following address: https://www.google.com/settings/security/lesssecureapps and enable access for less secure apps, this way RandA PhotoSharing will be able to send eMails on behalf of this account.

Create an IFTTT account (www.ifttt.com) and add the following recipes:

- Save photos on Google Drive: https://ifttt.com/recipes/192360-send-gmail-attachments-to-google-drive
- Upload photos on Twitter: https://ifttt.com/recipes/129743-twitter-subject-w-tw-body-tweet-attachment-photo
- Upload photos on Facebook: https://ifttt.com/recipes/13714-upload-photo-to-facebook-from-e-mail

RandA PhotoSharing usage

To execute RandA PhotoSharing you have to open the terminal via RaspBerry's GUI (not SSH) and run:
sudo python /var/www/randaps/randaps.py
To close RandA PhotoSharing press F11 (toggle fullscreen mode) and clck on the X.

The phases of the program are the following (in order of occurrence):

- **Start screen:** it shows the theme start screen or a slideshow of all the photos taken so far (depending on your settings) [Waits for OK to go on]
- **Preview:** it shows a raspberry-camera feed, using the Prev and Next buttons you can select various effects [Waits for OK to go on]
- **Measurement:** it'll ask to hold the two poles (5V e A0) and show a countdown to prepare for the photo, then it'll invite to wait while the image is processed
- **Social:** if enabled, it'll ask the user if he wants to upload the photo to all the social networks connected to IFTTT. It also shows the resulting photo [Waits for Prev or Next interpreted as Yes and No]
- **eMail:** if enabled, it'll ask the user if he wants to receive and HD version of the photo via eMail showing a textbox in which the user can insert his address. It also shows the resulting photo [Waits for a keypress to go on]
- **Photo result:** if both Social and eMail (steps 4 e 5) are disabled, this screen will show the resulting photo [Waits for OK to go on]
- **End screen:** It thanks the user with a custom image

The script will now jump back to the Start screen (punto 1) reloading all the control panel settings.

For "resulting photo" we mean the picture that RandA PhotoSharing has taken during the measurement, with the selected effect applied to it. It will also show a logo and the overlay image relative to the selected theme, with the measurement result inside.

Control panel

You can access RandA PhotoSharing control panel from every device connected to the same network by typing: *http://IP_RASPBERRY/randaps/admin* in your url bar
Use "admin" as username and "randaps" as password.

There are two main sections, the first one shows all the photos that have been taken (you can view them by clicking on the datetime), whether the user agreed to upload it to the social networks or not and eventually his eMail address; you can also download an Excel file with all the gathered data or delete all the pictures(both from the database and from the “photos” folder) using the two buttons located underneath the table.

The other section contains all the RandA PhotoSharing settings: when you edit them you have to either restart the program or reload the Start screen by doing a full run to load the new settings.
Here you can find the various setitngs:

- **Theme:** the measurement theme (maker, love cleverness, friendship, ...)
- **Add new theme:** it lets you add a new theme, the field “Value” rapresents the theme name (example: love) while the Description field can be a small description (example: Love Meter); after adding a new theme you’ll get a message which will remember you to add the theme-related image files into the *data* folder
- **Standby:** it lets you choose whether the Start Screen will show the theme image or a slideshow
- **Coordinates:** coordinates X and Y in pixels of the element which will be added to the photo (logo, overlay, result); you can modify these values hovever you want, remember that these are relative to a 1280x720 picture(unless changed in the python script) and the origin(X=0 e Y=0) is the top left corner of the photo
- **Save to SD card:** if enabled, it’ll leave the photos into the *photos* folder, otherwise it’ll delete them
- **Save to cloud:** if enabled, it’ll backup the photos on Google Drive
- **Send to social:** if enabled, it’ll ask the user if he wants to upload the photo on Facebook/Twitter (depending on the IFTTT config)
- **Send via eMail:** if enabled, it’ll ask the user if he wants to receive his picture via eMail
- **Sender eMail:** the eMail address of our Gmal account which will send the pictures to the user and to IFTTT
- **eMail object:** email subject
- **IFTTT object:** tags which will be added to the email subject for IFTTT
- **eMail body:** editor HTML which will let you customize the eMail body

- **text_start:** Start Screen text
- **text_preview:** text shown during the camera preview
- **text_measurement:** text shown during the measurement
- **text_photo:** text shown when the image is taken
- **text_wait:** text that invites the user to wait while the image is processed
- **text_social:** text that invites the user to upload the picture on the social networks
- **text_email:** text that asks the user to write his eMail addres to receive his picture
- **text_yes:** word “Yes”
- **test_no:** word “No”
- **text_end:** shown in the PhotoResult screen asking the user to press OK to complete

Customization

The images needed for RandA PhotoSharing are located into the *data* folder:

- **[tema]_screen.gif:** GIF image (1920x1080) scaled depending on the screen resolution and shown into the Start screen depending on your theme
- **[tema]_overlay.png:** PNG image(whichever dimension) it’ll represent the measurement theme
- **[tema]_end.gif:** GIF image(1920x1080) scaled depending on the screen resolution and shown at the end of the execution.
- **logo.png:** PNG image(whichever dimension) which is overlayed to the photo, it doesn’t depend on the theme

Every time you add a new theme, you’ll have to add a GIF image(1920x1080) into the *data* folder which represents the theme and will be shown into the Start screen, naming it as follows:
THEME_screen.gif (exampple: love_screen.gif)
Do the same thing with the End screen:
THEME_end.gif (example: love_end.gif)
A PNG image (whichever dimension) which server as an overlay for the photos, named as follows:
THEME_overlay.png (example: love_overlay.png)

You can get an higher level of customizability by editing the *randaps.py* file in the section “CUSTOMIZABLE VARIABLES”.

In order to add or modify a web panel user, you have to use PHPMyAdmin:
connect to *http://IP_RASPBERRY/phpmyadmin* (swapping “IP_RASPBERRY” with the actual IP address assigned to RaspBerry), select the “randaps” database and then the “users” table.
Click on the “Insert” tab and fill in the fields “user” and “password”; the “password” field is crypted using MD5, to generate an MD5 crypted string you can use one of the many online tools available..