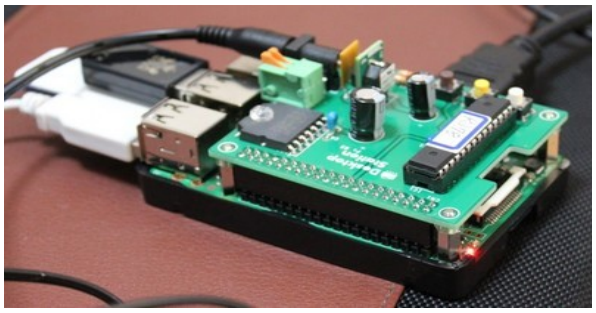
Desktop Station pi (DSpi)

DCC and Märklin Model Railway Control Expansion board for Raspberry Pi model 2B and 3B



Features

- Raspberry Pi model 2B or 3B microcomputers can be used to control DCC and Märklin locomotives and accessories *(track turnouts, switches or points)* via this DSpi plug-in daughter board.
- This board has internal 5Vdc / 1.5Amp Power Supply installed to run the Raspberry Pi motherboard. You should only connect an AC Plug Pack adaptor with a 12Vdc to 20Vdc power output, to this board. There is no need to connect a 5Vdc micro power supply to the Raspberry Pi motherboard!
- Node.js *(Javascript language)* and C ++ library and source code examples are available, free of charge, for download.
- If you use the Node.js software, you can control DCC and Märklin MM2 locomotives via your Smartphone or a Tablet using a Web Browser!
- The price for the un-assembled parts kit is 35EUR plus shipping *(3EUR SAL or 10EUR EMS)*
- You can control the digital model rail road using either DCC and/or Märklin MM2 with this small Raspberry Pi model 2B or model 3B computer in Z, N, HO, G gauge.
- Electrical current draw of up to 2 Amps can be handled by the DSpi daughter board, to operate locomotives on your railway track. Please attach a heat sink and fan if you intend to use a current draw from 2 Amps up to about 3.5 Amps.
- A single AC Plug Pack adaptor with a power output of 12Vdc ~ 24Vdc to match your model train system, is all that is required. The DSpi daughter board has a built in 5Vdc power supply providing power direct to the Raspberry Pi motherboard. There is no need to connect an AC Plug Pack adaptor with 5Vdc power, to the microUSB terminal on the Raspberry Pi model 2B or 3B motherboard.
- You can develop a high-performance DCC controller yourself.
- The system is Wireless LAN-compatible with a USB wireless LAN adapter plugged into the Raspberry Pi model 2B motherboard. The Raspberry Pi model 3B motherboard has both WiFi and Bluetooth configured on board – this allows connection to LCD monitors and various other accessories – the possibilities are limitless.
- Since the pulse generation for either DCC and/or MM2 signal/command protocols from the DSpi daughter board, out to the model trains, are carried out in the ATMEL microcomputer chip on the DSpi daughter board, there is no load on the Raspberry Pi model 2B or 3B motherboard’s main CPU.
- The component kit price is 4000 yen + mailing cost 200 yen. The kit consists of a board

and a number of loose electrical components which have to be soldered by you, onto the board. We also offer a fully assembled version of the board, at additional cost, which we test before shipment.

- Libraries in C language, and in Javascript language *(Node.js*) are distributed free of charge, and permits you to do additional software development. The Javascript language provides Web Browser capability for operating Desktop Station with iPad's and Smart phones, via a WiFi Modem Router.
- Of course, Desktop Station software is also available for the PC computer – you can also use a PC to "talk" to the Raspberry Pi motherboard, as the Raspberry Pi motherboard will be acting as a web page server. Once you have connected your PC to the wireless LAN network, you can use the PC version of Desktop Station software to operate your model trains through the Raspberry Pi and DSpi boards. The PC version of the Desktop Station software has more features built into it, compared to the current Raspbian-Jessie version.

How does it run?

Sample video on the Youtube

Purchase options - *how to get it*

1 kit of all the parts can supplied in an un-soldered condition, for 4000 yen. You will need to solder the parts onto the DSpi board. As an alternative, we can also offer a fully assembled board with all soldering completed by us, then checked for operation, before shipment to you.

Soldered kit	~~50EUR (6000 yen)~~
Unsoldered kit	~~35EUR (4000 yen)~~

The above prices are for supply ex-works Japan. Details concerning shipping costs via EMS (air freight) or SAL (sea freight) are available on our Order Form. (*As a guide, delivery by EMS to Australia takes about 3 days and to Europe and the eastern part of the USA – about 5 days – Ed.)*

Your order page form can be accessed via this link. If this does not work, please feel free to send an e-mail to the administrator.

Preparation - *necessary things*

- Either a Raspberry Pi model 2B or model 3B microcomputer. We do not recommended the use of the original Raspberry Pi 1, in all its versions, or the newer Raspberry Pi Zero, as they are just too slow.
- A CLASS10 micro SDHC card with an SD card adaptor. This will allow you to download and write the Raspbian-Jessie operating system onto the SDHC card – we would suggest a minimum microSD card capacity of 8Gb and the preferred capacity should be a 16Gb microSD card, or you may use a larger size.
- The use of a Apple Mac laptop or a Windows PC laptop, with an SD card reader/writer drive

built in, or a Windows PC tower computer with an SD card reader/writer drive fitted. Free software is available from the Raspberry Pi web site *(http://www.raspberrypi.org)* and there are instructions on how to download the Raspbian-Jessie operating system and "burn" it from an Image file, onto an SD card.

(We have found the Windows PC software recommended by the Raspberry Pi organisation to be the easiest way to "burn the image" onto the SD card, as the two pieces of software use for this purpose, have a Graphical User Interface system – Ed.)

- DSpi daughter board - pricing options for supply are shown above.
- AC Plug Pack adapter with a current output of some 1Amp to 2 Amps is highly recommended. Voltage output will depend on your model trains system - domestic Japanese model trains such as Tomix and Kato run 12Vdc, as do English Hornby model trains, and various American model trains. The older analogue system used by Märklin of Germany, used 16Vac with a pulse to 24Vac for reversing; all the new Märklin Digital systems use 16Vdc ~ 19Vdc from their AC plug pack adaptors, depending on the controller being used *(Mobile Station 2 or Central Station 3 / 3Plus, respectively)*. Please check your voltage according to your model train system.
- Wiring from the DSpi daughter board to the model train railway track. Check your model railway manufacturers requirements regarding wire size, wire length and connection methods to the rails on the track.
- If available – a good USB wireless LAN Adapter to plug into the Raspberry Pi model 2B motherboard. Check the Raspberry Pi web site for a list of suitable wireless USB LAN adaptors, which do work – be warned – many do not. The Raspberry Pi model 3B has both WiFi and Bluetooth built in, and both of these systems are easily made operational with the latest Raspbian-Jessie operating system.
- A keyboard and a mouse (both wireless) with a single "USB WiFi dongle" to plug into the Raspberry Pi 2B motherboard is the best way to go, to save on USB port usage. In the case of the Raspberry Pi 3B motherboards, a Bluetooth keyboard with touch pad mouse built in, is an excellent alternative, and it saves on having to use one of the USB ports, on the Raspberry Pi model 3B motherboard.
- A selection of nylon circuit board "stand-offs" with an M3 thread through the middle, plus a selection of M3 nylon washers, M3 steel washers and M3 steel screws. These are used to pack up the gap between the Raspberry Pi motherboard and the DSpi daughter board (HAT) once both boards are pressed together, at the GPIO pin connectors.
- Supply of an LCD computer monitor with connections for a HDMI cable, and also a HDMI cable to connect the Raspberry Pi motherboard to the LCD monitor.
- TightVNC (Viewer)

Software Download

You will need a copy of the NOOBS software, or more particularly a copy of Raspbian-Jessie Operating System dated 27-MAY-2016, from the Raspberry Pi Organization web site. Raspbian is a version of the Debian-Linux operating system, which has been compiled for use on the ARM archeticure CPU's used on the Raspberry Pi motherboard. The Raspbery Pi Organization provides detailed instructions on how to download, prepare a freshly formatted micro-SD card, and write or "burn" the Operating System image, onto the micro-SD card. You can use a Windows computer, an Apple MAC or a Linux based PC with an SD card reader/writer to format and write the operating system image to the card. *(we have*

found the Windows system to be the easiest to use – Editor).

You will also need to download the following files from our web site.

Module name	Version	Date	Notes
C++ library and ds_sample for DSpi	draft	April-2015	C language user-friendly
~~Desktop Station for RaspberryPi | and Node.js~~	~~R1~~	~~8-MAY-2015~~	~~Node.js users, fixes a bug in CV writing function~~
Desktop Station for Raspberry Pi | and Node.js	R3	9-JUNE-2016	Node.js users; the latest version of rpio, and Node.js support.

Download these ZIP files by “clicking” on each of them. The ZIP files will be saved in your Downloads folder on your Raspbian-Jessie operating system, then you will install them as we carry out the system installation, into different sub-directories (sub-folders) on the microcomputer. To extract the normal files from the compressed ZIP file format, use the File Manager programme, and once inside the Downloads folder, “left click” on the ZIP file. You will offered an option in the dialogue box, to expand the ZIP files into different locations – choose “Extract here”. The expanded files will then be saved in your folders, along with the original ZIP file.

Description of DSpi Board

The DSpi daughter board we provide, has been made to the Raspberry Pi organization size and connection specifications. This is known as a HAT specification *(Hardware Attached on Top)* and is similar in concept to the “Shield” daughter boards made for the Arduino microcomputers. The layout of the pluse generator and other components on our DSpi HAT is of our original design creation and arbitrary interpretation. Please note – if you use a Raspberry Pi motherboard with our DSpi HAT connected – then you cannot fit the twin board assembly, into a standard Raspberry Pi sealed case.

Instructions and/or commands from the software programme on the Raspberry Pi mother board, are fed through the GPIO terminal plug to the ATMEL microprocessor, on the DSpi daughter board. The ATMEL microprocessor is loaded with firmware (a software programme) which has two parts – a place where changeable variables are stored at the start of the programme, and lower half of the programme has a “continuous loop” to cycle through the instructions in the firmware programme, at very high speed *(can be upwards of 100 cycles per second)*.

The output from the firmware programme is then sent to the pluse generator on the DSpi daughter board – the special pluses, in the form of a special high frequency squared off “sine” wave format of “on” and “off” switched signals, are generated and sent out to the railway track, so that they can be received and acted upon by the decoders, in the model railway locomotives.

circuit diagram

Bill of materials

Parts No	Parts name	Specification	Notes
R2	Resistor	470 Ohm	
R3	Resistor	470 Ohm	
C1	Capacitor	0.1uF	MLC
C2	Capacitor	220uF	
C3	Capacitor	100uF	
C4	Capacitor	1uF	MLC
IC1 '	DIP socket	28pins DIP socket	
IC1	MCU	ATMEL ATMEGA328P	Main controller
IC2	Motor Driver	TOSHIBA TB6643KQ	Pulse generator for DCC and MM
IC3	DC / DC converter	MURATA OKI-78SR-5 / 1.5-W36-C	Output 5V 1.5Amax
F1	Resettable fuse	Tyco RUEF250	3.0A shut off
SW2	Tactical switch		
SW3	Tactical switch		
SW4	Tactical switch		
D2	LED	3mm, red	
D3	LED	3mm, yellow	
CN1	20 × 2 frame header		
CN2	2pins terminal		
CN3	DC input jack	MH-179P compatible	recommended high power version as 2DC0005D100
-	M3 nut		To infix TB6643KQ.
-	M3 screw		To infix TB6643KQ.
-	M3 nuts for spacer		4 pcs
-	M2.6 spacer		4 pcs
-	M2.6 screw		4 pcs

Assembly method

Please solder the parts to DSpi HAT, as per the part numbers.

It should be noted that after soldering, and inserting the DSpi board onto the Raspberry Pi motherboard, DC-in jack on the DSpi board will interfere with the RJ45 connector on the Raspberry Pi motherboard. The Kit comes with some M3 nuts, and you can use these nuts to push up underneath the board to provide clearance height. This “height/clearance” issue has been fixed on all funture boards marked with version R3a or later, and has been achieved by using a high mounted DC-in jack connector, on the DSpi daughter board .

When the two boards are pressed together at the GPIO pin socket, the gap between the printed circuit boards is 11.1mm.

One of our Australian users has come up with a way of packaging the boards together, and then mounting them on the rear of an LDC monitor. Use a selection of M3 threaded nylon “stand-offs”, 9mm in length, plus a selection thick M3 nylon washers and thin M3 steel flat washers to pack up gap, to the overall height of 11.1mm. Make sure that if you use flat steel washers – they should not make contact with the printed circuit boards or any of the electrical components.
A 20mm long M3 threaded screw then passes downwards from the top, through upper DSpi daughter board, then the “packers” and finally through the Raspberry Pi motherboard. Some additional 9mm long threaded “stand-off's” are then be screwed onto the underside of the Raspberry Pi motherboard, so that it is rasied clear of its mounting plate. Some M3 * 5mm long screws can them be fitted up through the mounting plate into the nylon threaded stand-off’s.

With this method, the holes in the DSpi board need to be hand drilled out from 2.6mm to 3mm diameter. **DO NOT USE any power tools**, to drill out the holes – you may damage the DSpi daughter board.

How to use

Connection relationship for the power supply

Please prepare the AC Plug Pack adapter, with a 12Vdc ~ 20Vdc voltage output range is preferred *(these are most commonly of a switch mode design – Ed.)*. Please ensure that you have the correct voltage for your model railway system (usually 12Vdc for most makes - Märklin is 16Vdc ~ 19Vdc).

It should be noted - you do not need to supply a 5Vdc voltage via the microUSB connector, into the Raspberry Pi motherboard. . The DSpi HAT board has a built-in 5Vdc power module with 1.5 Amp max current output supply *(a high output type of DC / DC converter)*, and has been proven to work without any problems, regardless of the load.

You should **ONLY** connect power the the female DC-in power socket on the DSpi HAT daughter board.

DO NOT connect any power cables to the microUSB female connector on the Raspberry Pi motherboard board. If you do – power to the Raspberry Pi motherboard will be doubled and damage will result.

Software preparation

Here you use the microSD card removed from the SD card adaptor, you have used to "burn" the image of the Raspbian-Jessie operating system. You will need to set up your Raspbian-Jessie operating system for your local environment *(that is – your language, time zone, computer keyboard language being used and your countries WiFi system)* so that the Raspberry Pi will be configured to your country's requirements.

You should note that Raspbian-Jessie comes "out-of-the-box" configured for use in the United Kingdom *(England, Scotland, Wales and Northern Island)* with GB English language and spelling, and also the use of a UK or GB keyboard with the GB Pound currency symbol, rather than a Dollar symbol.

The lasest Rasbian-Jessie operating system has a very good Graphical User Interface (GUI) in its Menu system to allow the changes to be easily made. Just "click" on Menu in the top left corner of the screen, then "scroll" down to "Preferences", then Scroll across to the next dropdown menu, and "click" on Raspberry Pi Configuration. This will place a dialogue box in the centre of the screen.
Near the top of the dialogue box you will see four (4) tabs :

System / **Interrfaces** / **Performance** / **Localisation**.

Under the **System** tab, "click" on "**Expand Filesystem**". This makes the Raspbian-Jessie operating system expand its file system to use up all the space on the microSD card. *(it will grow from the installed size of around 3.5Gb, up to the maximum capacity of the microSD card – Editor).*

Next "click" on the **Interfaces** tab, and at the third line down, **SPI**, "click" on the radio button to enable SPI.

Next move to **Performance** tab and choose the maximum speed if you are using a Raspberry Pi model 2B – the Raspberry Pi model 3B motherboard is already set at maximum speed.

Finally go into the **Localisation** tab to set up for your language counrty and keyboard type, time zone and country WiFi settings. When you have finished all adjustments, “click” the OK button. And you will be asked to reboot your computer to save these changes.

You should also set up a connection to the Internet through your WiFi Modem Router, and also any home network you may have, once you have re-booted the Raspberry Pi.

Model railroad side

DCC / MM2 shield has been designed to supply equivalent functions for either system. You can use the DSpi HAT daughter board, regardless of the gauge of the model railway system, as it can provide both DCC or MM2 signals for these respective protocols, out to the model railway track.

Both the DCC and MM2 protocols are a simplex transmission, out to the locomotives on the track – the commands are sent from the DSpi controller out to the decoder in the locomotives. There is no facility on the DSpi daughterboard to accept any return signals from the locomotives on the track, back to the main controller.

If you have a Märklin Digital model railway system and are using the later model designed mFX or mFX+ decoders in locomotives, then you will have to change the main address of each locomotive if you are using 2 or more locomotives at the same time, on your model track layout. Each locomotive on your track, has to have a unique address number other than the usual Märklin “24” address, which locomotives are supplied with, from the Märklin factory. This address change can be made with the aid of a Märklin Mobile Station 2 controller.

You can run a combination DCC and MM2 protocol locomotives at the same time on the one track system. For example, we have tested an American sourced MTH manufactured model locomotive, using DCC protocol, with locomotive address “3” *(their usual address set-up from the factory)* and at the same time run a Märklin locomotive with MM2 protocol, with any locomotive address other than “3” which is used by the MTH locomotive.

The MTH locomotives will need to have a centre rail power pick-up shoe, to be able to run on the Märklin 3-rail C-Track or 3-rail K-Track systems – these MTH locomotives can easily be identified by the use of a DASH 5 number at the end of the MTH model part number *(i.e. 80-3202-5 is a Challenger Steam Locomotive designed for use on Märklin 3-rail track).*

Setup procedure

It will be based on the use of the Raspbian-Jessie operating system that you installed, on the microSD card. Please verify that you are using either Raspberry Pi model 2B or 3B. We would recommend that you **DO NOT TRY** to use Raspberry Pi Zero or Raspberry Pi 1 motherboards.

With the software we provide in the form of the “ds_sample” library, for use with ther C++ language and the Javascript language (node.js) files, you have a choice of installing either.

However, we suggest that your install both types of software in the sequence shown below, as the wiringPi software allows you to confirm that your Raspberry Pi motherboard is communicating with the firmware on the DSpi HAT daughter board. The Javascript language (Node.js) will generally be preferred, as it allows you to operate in a Graphical User Interface (GUI) using Web Browser pages served from the Raspberry Pi motherboard, onto the screen of your preferred display device.

We will use the Terminal programme to set up the software and also configure the Raspberry Pi motherboard. The Terminal programme is shown as a black coloured computer monitor in the top left hand corner of the Menu bar. Just left click on it, and it will open up on your screen. The terminal programme is also referred to as a Command Line Interface (CLI), as you have to type commands then push the “Enter” on your keyboard, to get the microcomputer to do something. This technique is quite a bit different to the Graphical User Interface (GUI) that we are all used to using on a late model Windows PC or Apple MAC's.

Some important commands that you will need to know when using the CLI, are :-

cd = change directory, so that your can move through the folder system
(when you open the Terminal CLI, the directory or folder you will be starting in, is /home/pi/ so to move down the directory tree to the Downloads folder, you would type
cd /home/pi/Downloads/)

ls = list the contents of the directory that you are in
(so you can find where you are, what's available, then where to go).

Exit = to shut down and leave the Terminal CLI programme.

Raspi-config

The first series of steps involves configuring the computer so that the Raspberry Pi mother board can actually talk to the DSpi HAT daughter board.

The first instructions will be :-

```
~ $ sudo raspi-config
```

The easiest way to place this command into the Terminal CLI box, is to use a “cut-and-paste” operation – staring to the right of the “g” letter in “-config”., “click” and “hold” then drag the mouse to the left until you pass the letter “s” in the word “sudo-” , then release the left mouse button *(we want all the command in the yellow box)*. Move back into the centre of the yellow box, and “click” the right mouse button. A dialogue box will drop down. “Click” on Copy. Move over to the Terminal programme (CLI) and left click in the little box next to the “$” symbol. It will change to a solid white box, then right click on the box – a dialogue will drop down and then “click” on Paste. The command is then in the Terminal CLI box. Now just press the “Enter” key to activate the command.

The Raspberry Pi Software Configuration tool *(a rectangular box with a grey coloured background)* is now displayed in the Terminal CLI. Using your down arrow key, roll down to item “9. Advanced Options” and press “Enter”. Then roll down to “A5 SPI” and press “Enter”. Press the “enter” on your keyboard to select the highlighted **<Yes>**, then select the highlighted **<Ok>** with the “enter” key. You are then returned to the Configuration Tool menu. Use your “Tab” key to move to the **<Finish>** button and then press the “Enter”key. You are now back in the CLI.

The next instruction is :-

```
~ $ sudo nano /boot/config.txt
```

Use the same “copy-and-paste” method to open this configuration file in the Nano text editor. Once the Nano editor is open use your down arrow key to roll down to the bottom of the file. On the next blank line down, type in :-

```
dtparam=spi=on
```

Then press and hold the Control key and also press the O key (Ctrl-O) to write the new file. Then press “Enter” to save it and Control X (Ctrl-X) to exit the Nano text editor.

The next instruction is as follows :-

```
~ $ sudo nano /etc/modules
```

use the “copy-and-paste” method again, then push “enter” to open the Nano editor. Arrow down to be bottom of the file and write the line :-

```
spidev
```

Then press and hold the Control key and also press the O key (Ctrl-O) to wite the new file. Then press “Enter” to save it and Control X (Ctrl-X) to exit the Nano text editor.

Setting up Raspberry Pi for Network operation

If you carry out the following procedures, your will find that the Destop Station system will be easier and more convenient to use in a GUI, with a Web Browser over a wireless WiFi network. We are going to do the following:-

- Confirm the insertion and set up of your USB wireless LAN adapter in the case of Raspberry Pi 2B motherboards, or activate the WiFi module built onto the motherboard of the Raspberry Pi 3B – *this should have already been done by this time -Ed.*

- Change and registration of the host name (if necessary)

The importance, in the selection of your USB wireless LAN adapter, cannot be stressed enough. In Japan, you can purchase the GW-USNANO2A or the WLI-UC-GNM models for a few thousand yen, at consumer electronics retailers. These are quite old models, but they still work. The Raspberry Pi Organization has worked with major worldwide components distributors – **RS Components** and the **element 14 Group**, and they offer a plug-in "WiPi" model USB wireless LAN adapter. It has a speed of 150Mbps which is now considered slow but it works, and sells in Australia for AUD $10.00.

If you want something with more speed then have a look at the web page :-

http://elinux.org/RPi_USB_Wi-Fi_Adapter

for a listing of various brands that have been tested and work with Raspbian operating system on the Raspberry Pi 2 motherboard. It is a bit of a trial and error to find a higher speed Adapter.

The TP-Link model TL-WN823N has been tested in Australia and works with Raspbian-Jessie. It has an operating speed of up to 300Mbps and it sells for around AUD $35.00 at retail computer stores. It works to the following wireless network standards - IEEE 802.11 b/g/n.

Please Note - The latest range of AC dual band USB Wireless Adapters, which sell in Australia for around AUD $70.00 at retail computer shops, are of the latest new high speed design and can work on both the 2.4GHz and 5GHz radio frequency bands. These new models are not compatible at this time, as the software drivers have not been developed fully to run with Linux based operating systems, such as Rasbpian-Jessie OS on an ARM chip based computer system.

Perform the following steps to change the host name. Select a suitable host name – lets use a combination of DSpi and the Raspberry Pi 2B motherboard. Lets call it "DSRasPi2".

Place the following command into the Terminal box (CLI) and then press "Enter"

```
~$ sudo nano /etc/hostname
```

Then on the first line of the Nano text editor, replace the word " … raspberrypi … " by typing the word :-

DSRasPi2

Then press and hold the Control key and also press the O key (Ctrl-O) to wite the new file. Then press “Enter” to save it and Control X (Ctrl-X) to exit the Nano text editor.

Then to complete the association of the IP address and the host name, type in the following command in the Terminal CLI :-

~ $ sudo nano /etc/hosts

On the first line of the Nano text editor, opposite the I.P. address 127.0.0.1, replace the word “ ….localhost ….” with DSRasPi2 then press and hold the control key and also press the O key (Ctrl-O) to wite the new file.

System Control – Closed or Open?

It is also at this point where you must make a decision on the type of control system you wish to run. You can use the C language, but the most popular way to run the Desktop Station system is by the use of a web page, in a Graphical User Interface. The Raspberry Pi microcompouter then acts as a web page server, and the web page is shown on the LCD Monitor connected to the RaspberryPi motherboard, or to the web brower on a Smartphone or an iPad, or some other internet connectable device which can display web pages in a browser.

If you just wish to operate a very simple closed system using just a keyboard and a mouse to communicate with the DSRasPi2, and do not intend to use a Smartphone, iPad or another PC computer, then leave the IP address set at 127.0.0.1 on this first line. When you request this web address in your web browser, and the DesktopStation page will then be displayed on the Monitor.

However, if you wish to operate using an iPad or a Smartphone, via your WiFi Modem Router or any other type of WiFi device, then the Raspberry Pi will need to become a web page server on your Wireless Network. So the Raspberry PI (DSRasPi2) should be set up to become a part of the network, with its own unique static IP Address.

For home networks, these static IP addresses can be any numbers from 192.168.0.0 to 192.168.0.255, then from 192.168.1.0 up to 192.168.1.255.

Most computers and printers connect automatically to the WiFi Modem Router these days using DHCP, and are usually allocated a dynamic IP address on connection – generally in the range from 192.168.0.0 up to about 192.168.0.30 on a “first in – first served” basis.

So that you keep your “DSRasPi2” away from all this activity, we suggest to give your “DSRasPi2” a static IP address in the range from 192.168.1.150 through to 192.168.1.225. So for example, lets choose static IP address 192.168.1.175.

The while still in the Nano text editor, change the IP address on the first line, from 127.0.0.1 to your new static address, which is 192.168.1.175, so that the new static IP address can be set.

(*NOTE – once you finish this section, you will also need to update the information in your WiFi Modem Router so that it will recognise the new static IP address for the Raspberry Pi and make the connection, whenever the Raspberry Pi is powered up.)*

Then press “Enter” to save it and Control X (Ctrl-X) to exit the Nano text editor. You will now drop back out to the CLI.

Finally, in order to automatically register the DNS server at startup of the computer, install the Avahi programme, with the following command :-

```
~ $ sudo apt-get install avahi-daemon
```

For more information, install orphans Research Avahi daemon refer to the chapter.

Step out of the Terminal CLI by typing the word `exit` , then press the Enter key.

You should now shutdown the Raspberry Pi motherboard, and then re-boot the computer again.

Once the Raspbian-Jessie desktop has stablised, “click” into the Terminal programme (CLI) and when it opens up, you should now see -

```
pi@DSRasPi2: ~ $
```

So the identity of the Raspberry Pi has been changed to “DSRasPi2”.

WiringPi (for people to operate in the C language)

The Raspberry Pi GPIO *(general-purpose input output terminals)* are mounted in a two row block along one edge of the Raspberry Pi motherboard. This pin connector allows physical interface between the Raspberry Pi and the outside world. The connector pin terminals can be used to operate as part of a **UART** *(Universal Asynchronous Receiver/Transmitter)* which is the microchip with programming that controls a computer's interface to its attached serial devices. Specifically, it provides the computer with the RS-232C Data Terminal Equipment (DTE) interface so that it can "talk" to and exchange data with modems and other serial devices.

It can also functions as a **Serial Peripheral Interface** (**SPI)** bus, which is a synchronous serial communication interface specification used for short distance communication, primarily in embedded systems.

It can also function as an **I²C or I2C (Inter-Integrated Circuit**), which is a multi-master, multi -slave, single ended, serial computer bus invented by Philips Semiconductor (now NXP Semiconductors). It is typically used for attaching lower-speed peripheral ICs to processors and microcontrollers in short-distance, intra-board communication

wiringPi is a library to access GPIO and provide the UART, SPI, and I2C functions.

The first step is to ensure you have downloaded and un-ZIPPED the ds_sample.zip file in the Download folder on the Raspberry Pi. To un-ZIP the file, use your File Manager programme, and go into the Downloads folder, and "right Click" on the "ds_sample.zip" file. A drop down menu will appear, and then "click" on "Extract here". You should now have the following four (4) files, in the Downloads folder :-

DSGatewayLibPi.cpp
DSGatewayLibPi.h
ds_sample.c
Readme.TXT

Now move back to the Treminal CLI programme, and enter the following commands into the Terminal CLI, one line at a time, to permit the **wiringPi** software to be downloaded from the Internet :-

```
~ $ sudo apt-get install libi2c-dev

~ $  sudo apt-get install git-core

~ $  git clone git://git.drogon.net/wiringPi
```

Now at this point, if you open your File Manager *(the filing cabinet icon in the top lefthand side corner)*, you will see a new folder has been added, with the title "... wiringPi ...".

Left click on the Downloads folder to open it, then use the "copy and paste" method to copy these three files :-

DSGatewayLibPi.cpp
DSGatewayLibPi.h
ds_sample.c

..... and paste them into the new "wiringPi" folder. Once this is completed, close the File Manager and move back to the Terminal CLI programme.

The following commands are used to firstly build the files and then compile the files into an executable file on the Raspberry Pi computer. Place each of these commands into the CLI, one at a time and then press "Enter".

```
~ $  cd wiringPi

~/wiringPi $  ./build

~/wiringPi $  g++ ds_sample.c DSGatewayLibPi.cpp -lwiringPi -o ds_sample
```

What we have done in this last line is combine the two downloaded wiringPi files you received from our web site, and have "rolled" them together into an executable file called ds_sample.

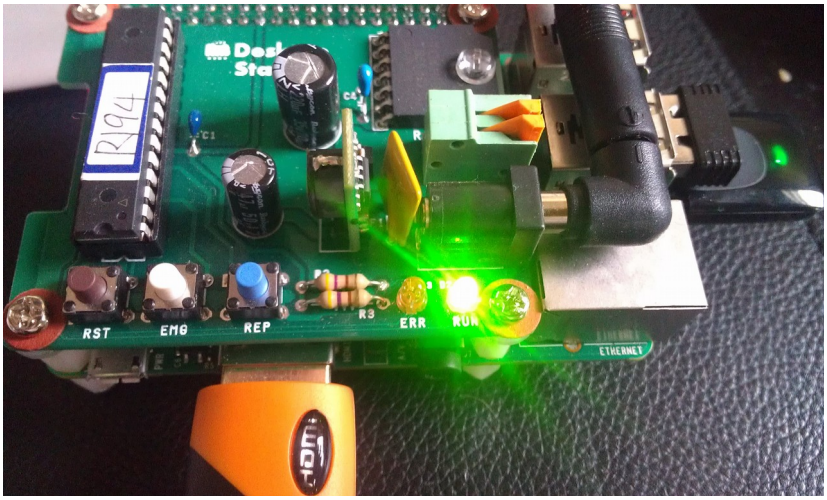
Now lets test it out, by placing the following command in the CLI, and then press "Enter".

```
~/wiringPi $ ./ds_sample
```

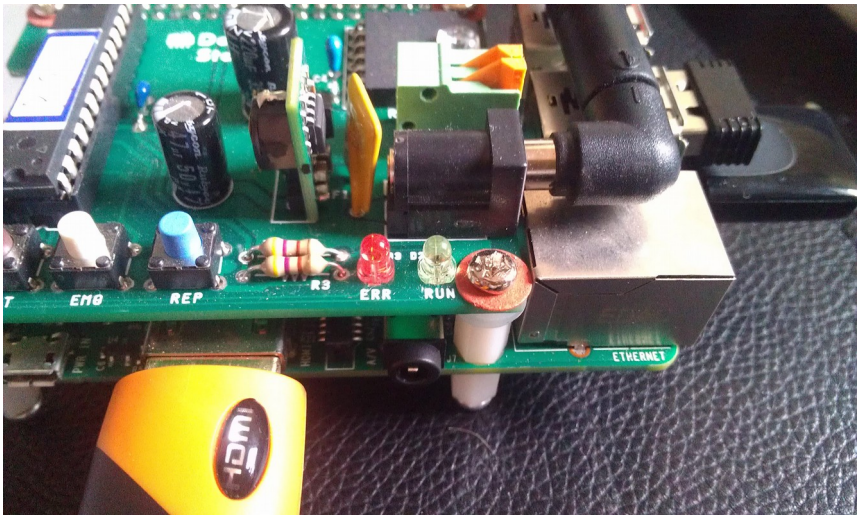
You should see the following six (6) lines show up, in the Terminal CLI box. There will be a time delay of some two(2) to five (5) seconds between each line appearing, as it is printed on the screen.

```
Power On
Run FWD
Run REV
Stop
Function on
Power off
```

We now have **wiringPi** working, if on the corner of the DSpi daughter board, you see the green “RUN” LED illuminated at Power On and remains illuminated until the Power off.



Power on



Power off

If you ever need to check it again in the future while in the CLI, just move to the wiringPi directory by tying in the command

~ $ `cd /home/pi/wiringPi` ..... and press “Enter”, then

~wiringPi $ `./ds_sample` ..... and press “Enter” again.

You should see the same six (6) lines of text appear on the screen, and the green “RUN” LED illuminates as shown above.

To finish off this section, we need to do a little house keeping. Using your File Manager GUI, go to the Download folder, and delete the four files which were un-ZIPPED from the ds_sample.zip file.

Node.js (for people who use a Web Browser GUI)

The Javascript language is a very powerfull programming lanugage used for building web servers for use across multiple platform computers, which provides us humans with a Graphical User Interface to work with, and send commands or requests through a web page, to a computer. The Node.js programme is the actual programme we use to build the server on the Raspberry Pi, and through the use of web pages, programmes such as Desktop Station can be operated easily on various terminals – that is, any device that can work with a web browser. Please see the following information on how to install the Node.js onto a Raspberry PI.

The Raspbian-Jessie OS was introduced during the 4th quarter of 2015 to replace the Raspbian-Wheezy OS. At about the same time, the Raspberry Pi Zero was introduced. A node graphical interface developed by IBM (International Business Machines) by the name of node-RED was added to the Raspbian-Jessie OS, and with it, the Node.js programme. Then in February 2016, the Raspberry Pi model 3B was introduced and in March 2016, a number further upgrades of the Raspbian-Jessie OS were carried out.

The version of Node.js installed as standard in the Raspbian-Jessie operating system, is v0.10.29, and the normally associated Node Package Manger (npm) version, was not installed. This Node.js programme is now quite old, so as part of the software installation, we will move up to Node.js v6.2.0

We have written the Desktop Station Web Server application around the use of the following Node.js components :-

software	version	notes
Raspbian OS	JESSIE, 27-Mar- 2016	
node.js	6.2.0	
npm	3.8.9	
node-rpio	0.9.11	https://github.com/jperkin/node-rpio

The first step is to check or confirm what has been actually installed in the Raspbian-Jessie operating system. Do this by typing the following command, in the Terminal CLI programme :-

~ $ `node -v` ..... and then press “Enter”

The CLI should respond with :- *v0.10.29*

Next try out npm with the following :-

~ $ `npm --version` ..... and then press “Enter”

The CLI should respond with :- *bash: npm: command not found*

So we have Node.js version v0.10.29 installed and operating, but not npm.

Now we need to upgrade to one of the latest versions of Node.js, which will be version 6.2.0, but before we do that, we have to remove the old version of Node.js, by typing the following command in the Terminal CLI :-

~ $ sudo apt-get autoremove nodejs ..... and then press “Enter”

This instruction will remove node-RED, nodejs, and nodejs-legacy, libc-ares2 and libv8-3.14.5 from the operating system.

So now lets type in the instructions to see the version number for node.js :-

~ $ node -v ..... and then press “Enter”

The Terminal CLI will respond with :-

bash: /usr/bin/node: No such file or directory.

So now nodejs v0.10.29 hase been removed from the DSRasPi2

Time to install the new version, by issuing these commands, one at a time, in the Terminal CLI programme:-

~ $ cd /usr/local/

~ local/ $ sudo wget https://nodejs.org/dist/v6.2.0/node-v6.2.0-linux-armv7l.tar.gz

~ local/ $ sudo tar --strip-components=1 -xvf node-v6.2.0-linux-armv7l.tar.gz

Now we have node.js 6.2.0 installed. Next “Exit” the Terminal CLI programme, then shut down and finally re-boot the computer. Start up the Terminal CLI, then check to see if new versions are up and running :-

~ $ node -v .... and press “Enter” *(reply is ..... v6.2.0)*

~ $ npm –version .... and press “Enter” *(reply is ..... v3.8.9)*

So we have both the latest Node.js programme installed and we also have the Node Package Manager (NPM) installed as well, which then lets us bring in other packages. Now we have to install node-rpio, by issuing the following command in the Terminal CLI :-

~ $ npm install rpio ..... and press “Enter”

(reply is … rpio@0.9.11, bindings@1.2.1, nan@2.3.5)

At this point, exit the Terminal CLI, shut down the computer, then re-boot. Once the desktop has stabilized, open the Terminal CLI box again.

One final item involves setting up rules for the rpio module. By default, the rpio module will use the /dev/gpiomem when usiung simple GPIO access. To access this device, your user needs to be a member of the gpio group, and you may ned to configure “udev” with the following rule (as root), bu issuing the following command :-

~ $ sudo nano /etc/udev/rules.d/20-gpiomem.rules .... then press “Enter”

When the Nano text editor opens up, on the first line type in the following :-

SUBSYSTEM==”bcm2835-gpiomem”, KERNEL==”gpiomem”, GROUP=”gpio”, MODE=”0660”

The press Control-O (ctrl-O) to write the file, then “Enter” to save the file, and Control-X (Ctrl-X) to exit the file.
You can then use the File manager GUI to check that the file has been inserted in the /etc/udev/rules.d directory.

Now we are going to set up the Desktop Station application for use in Node.js, by extracting files from the dsgw_R3.ZIP file in the Downloads folder. Using the File Manager programme, go into the Downloads folder, and “right click” on the dsgw_R3.ZIP file. Then select the “Extract here” instruction from the dialogue box, and press “Enter”.

A new folder or sub-directory called “dsgw_R3” will be seen. Using File Manager, “left double click “ on it to open it, and you will see the following list of files:-

dsgw.js
dsgw.sh
readme.txt.txt
template.html plus a CSS folder

The move back to the Terminal CLI box and type in the following commands :-

~ $ cd /home/pi/Downloads/dsgw_R3/

~ dsgw_R3 $ ls .... then press “Enter” *(reply is a list of 4 files, including dsgw.js)*

~ dsgw_R3 $ sudo node dsgw.js .... the press “Enter”

(relay is server started and running at 127.0.0.1:8124/)

or, if you set up the DSRasPi2 as an open system for use with a Smart phone or a Tablet

(reply is server started and running at 192.168.1.175:8124/)

The Web Server has been set up, and the application is loaded, with the computer listening for a request through Port 8124. The application has been written around the use of Port 8124, as Node.js would normally use port 80, which involves a lot of security and administrative prevleges having to be set up. By moving up to port 8124 in the

application programme, there are no security protocols involved – it is just straight communication with the computer.

Now move the Terminal CLI programme box, up towards the Left hand side top corner of the desktop, then open your preferred Web Browser.

In the command line of the Web Browser, you can type either of these commands for the Closed system

If you wish to use the Host Name (DSRasPI2), then type in the web browser bar :-

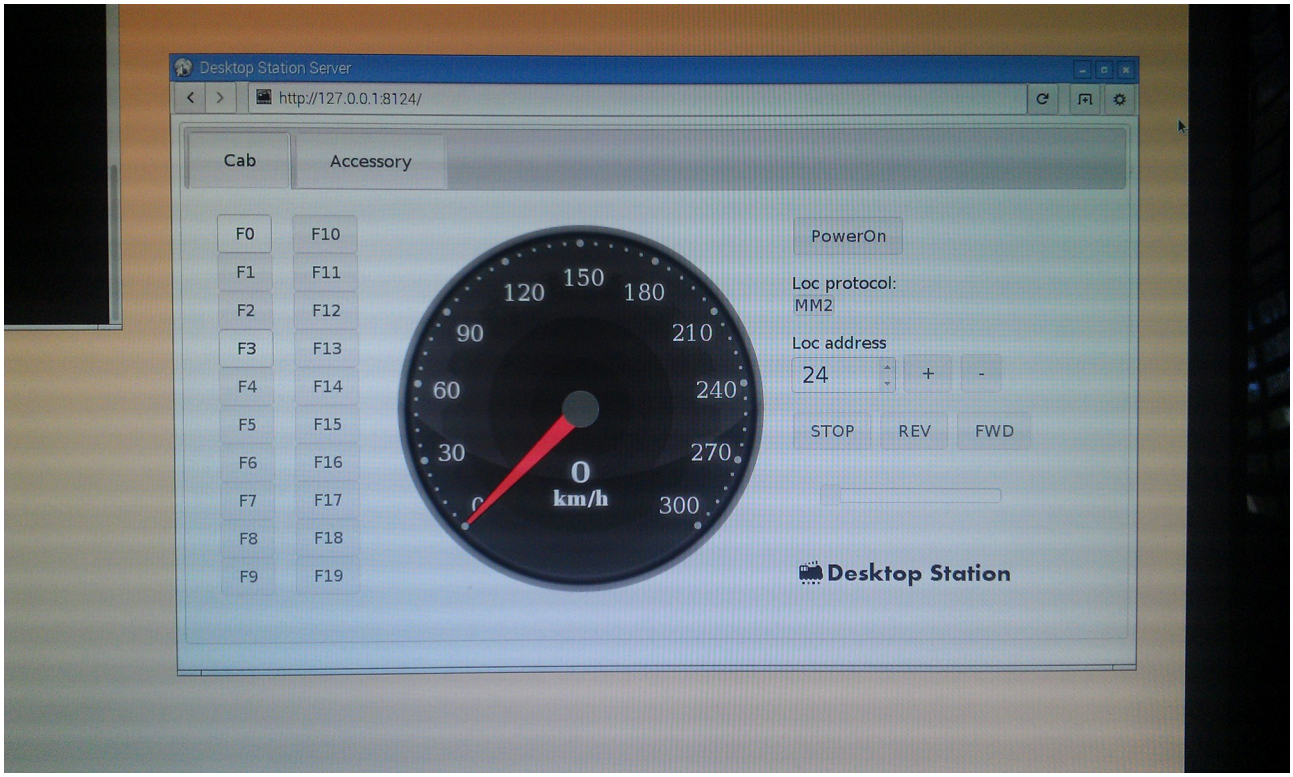
http://DSRasPI2.local:8124/

If you wish to use the IP address (127.0.0.1), then type in the web browser bar :-

http://127.0.0.1:8124/

(open system - with an iPad or Smart-phone, use http://192.168.1.175:8124/)

and if successful, the following screen will display



To operate in the web browser :-

Click on the Power On button (it will then change to Power Off)

then click on the MM2 button to change to DCC operations (click again for MM2)

move to the “Loc address” - use the “+” and “-” keys to change the Loc address.

Then click on the Forward (FWD) button on the lower right hand side. There is also a reverse (REV) button if you wish to have your locomotive move backwards.

Click on F3 button over on the LH side of the screen – this usually starts up the Locomotive engine noise.

Clock on F0 button to turn on the headlights

Finally, move Speedometer needle to say 60km/hr, with a click of the mouse near the number 60 on the face of the Speedometer, and the locomotive should start moving.

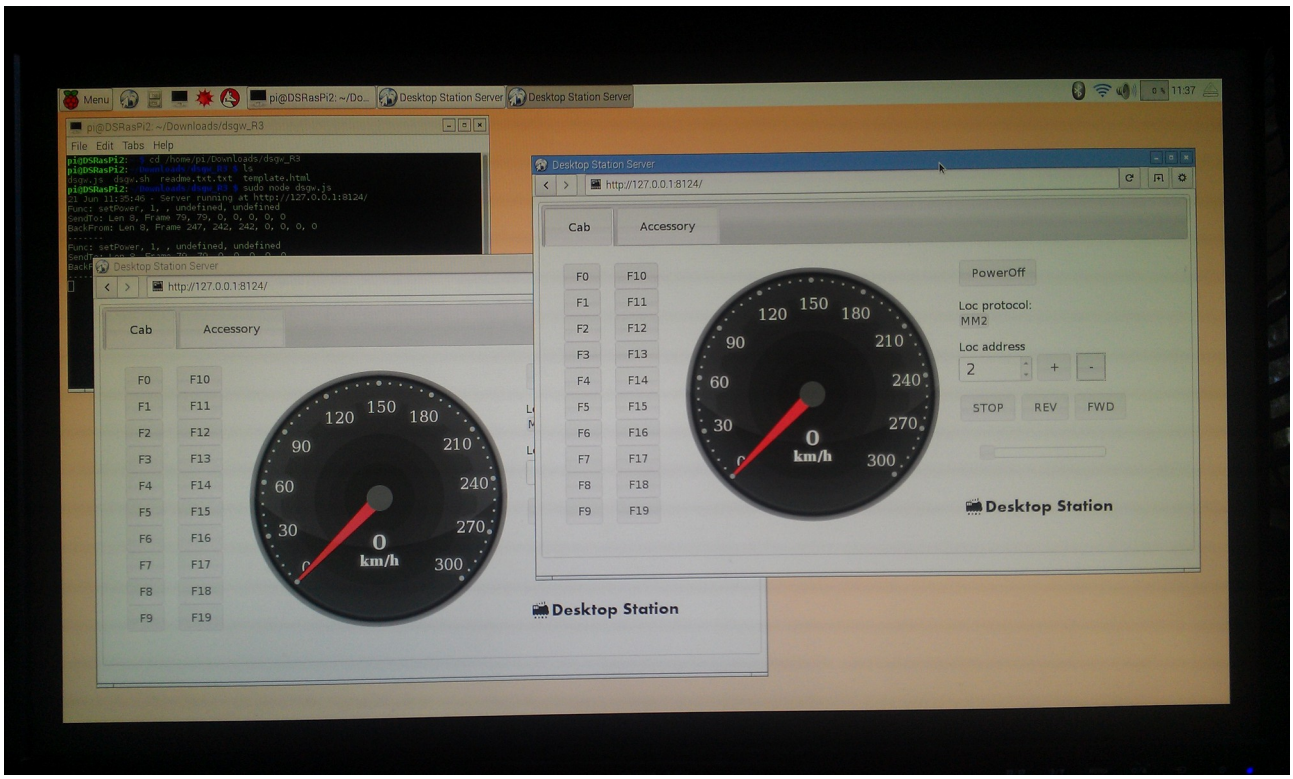
If you are going to operate 2 or more locomotives, then just open up additional web pages, and allocate a locomotive to each web page. You can then click on whichever web page for the locomotive you wish to control.

When you wish to shut down the system, bring each locomotive to a halt with the stop button – so all locomotives are stopped. Then click on the Power Off button on the upper right hand side of the web page, and it will change back to Power On, but the loco will completely shut down. You will notice the green “RUN” LED on the DSpi daughter board, also turns off. Do the same on all open pages, then click the “X” in the top right hand corner of each web page, to close it.

Finally move into the Terminal CLI programme box, and “click” the “X” in the top right hand corner of the Terminal CLI box.

Last step is to shut down the computer. You have remove the power connector from the DSpi daughter board to clear all the RAM memory and iny information inn the CPU. If you try to restart the Terminal CLI then the web page server without complete power shutdown, you will be presented with an Error page when you try to use the command “sudo node dsgw.js” in the Terninal CLI box.

This photograph below shows two (2) web browser pages open for control of two locomotives, and the Terminal CLI moved to the top left hand side corner of the desktop. The Terminal CLI allows you to see the changes being made to the variables, being sent from ther Raspberry Pi motherboard to the ATMEL microprocessor on the DSpi daughter board.



Just “click” on whichever web browser page you need to use, to make the changes to the respective locomotive.

TightVNC

TightVNC is a free remote control software package. With TightVNC, you can see the desktop of a remote machine and control it with your local mouse and keyboard, just like you would do it sitting in the front of that computer. TightVNC is:

- **free** for both personal and commercial usage, with full source code available,
- useful in administration, tech support, education, and for many other purposes,
- cross-platform, available for Windows and Unix, with Java client included,
- compatible with standard VNC software, conforming to RFB protocol specifications.

With TightVNC, you can:

- **cut your expenses** and **save your time** on traveling,
- help your friends and family to solve problems with their computers remotely,
- make sure nothing wrong is happening on your computers when you are away from your home or office.

In the Tereminal CLI programme, type the following :-

~ $ sudo apt-get install tightvncserver

```
~ $ tightvncserver
```

If you wish to start tightvncserver automatically, then rewrite the file init.d ; you will find more detailed information on the Internet via http://www.tightvnc.com

- Let's play in the Raspberry Pi - No.5: trying to VNC connection

Support

Our company, as the producer of the DSpi HAT daughter board, does not have a great deal of knowledge on Linux operating systems, Raspberry Pi microcomputers or the Node.js (Javascript) programming language.

Other than for hardware failures on the DSpi HAT board, it is up to you to be able to customise and use the systems provided, and resolve any issues, not covered in the above instructions.

In addition, we have a Desktop Station forum on our web site, which may be of assistance to you in answering questions to solved problems. There is also an opportunity to contribute to the Desktop Station community by volunteering any solutions you may have or providing any programming skills in an open source environment, to improve the Desktop Station system.

Offers of assistance, will not be refused.

Please note that the forum is conducted in the Japanese language – we would suggest you use Google Chrome or Google Chromium to view the Forum information, as this particuler web browser has a language translator built into it. Just “right click” on a page and then select “translate” from the dialogue box, and the Japanese language Kanji characters will be converted into your language. You can post messages in your language, on the Forum pages, but I would prefer you use the English language, if possible.

Thank you + Regards
Yassan.