How to Install sccKit 1.4.0
revision 1.5

Date	Revision	Description
2/24/11	1.0	First draft.
3/14/11	1.1	Added step to edit named.conf.options.
3/15/11	1.2	Caution that bitstream for CD is unavailable; added information about new sccPowercycle
3/19/11	1.3	Typos
3/23/11	1.4	Added the need to run install.csh after Step 12; put the use of sccPowercycle –r as the preferred replacement for rebooting MCPC.
3/28/11	1.5	Addeddescription of ssh_config. Edited description of in.rck.zone and ex.rck.zone. Described where to obtain the downloads.

Table of Contents

List of Figures

Introduction

sccKit 1.4.0 uses an EMAC interface in addition to the PCIe interface. The `crbif` driver now has two components:

The PCIe interface to the sccKit software components (for example, the sccGui and sccKit commands)

The Ethernet device `crb0` that communicates with the device `pc0` on the actual SCC core (Ethernet over PCIe or Ethernet over Emac).

The first component is still active when you configure the new sccKit. The driver `crbif` will still be loaded.

The second component will only become active, when `sccMacEnable` is not set in `systemSettings.ini`. When `sccMacEnable` is set, “Ethernet over PCIe” is disabled and “Ethernet via EMAC” is enabled. “Ethernet over PCIe” and “Ethernet via EMAC” are mutually exclusive.

Figure 1 illustrates how the SCC and the MCPC are connected when running sccKit 1.4.0. There are two Ethernet cables coming from the SCC chassis. One is referred to as PortX (where X is A, B, C, or D), and the other as the BMC cable. The MCPC also has two Ethernet cables and two NICs. The `eth0` cable connects to the Internet, most likely through your own router and firewall. The `eth1` cable connects to the BMC. Typically, users configure `eth1` to have a virtual ethernet connection `eth1:1` as well. This allows the BMC and the MCPC to be on the same subnet so that you can telnet into the BMC from the MCPC.

Figure 2 is a photo of the back of the SCC chassis.

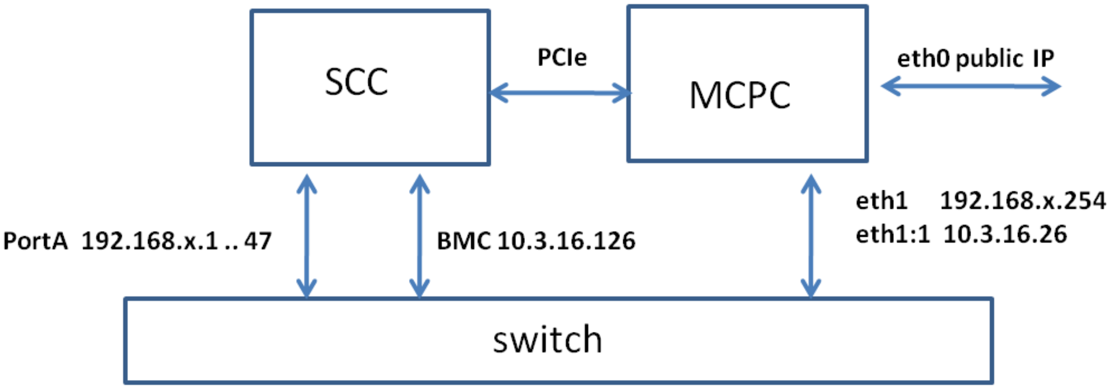


Figure 1: An SCC/RockyLake System Showing the Connection for the Ethernet Cables and the PCIe cable.

To begin the installation, you need the file `sccKit_1.4.0.tar.bz2`. This file is downloadable from http://marcbug.scc-dc.com/svn/repository/tarballs. The source code for sccKit 1.4.0 is also available. It is tagged as `sccKit_V1.4.0` in http://marcbug.scc-dc.com/svn/repository/tags.

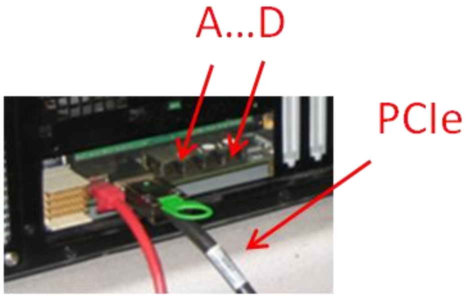


Figure 2: Photo of the Back of the SCC Chassis, Showing the BMC Ethernet Cable, the PCIe cable and the Port ABCD Sockets.

Step 1

Become root on your MCPC. You need the compressed tar file that contains sccKit 1.4.0. It's called something like sccKit_1.4.0.tar.bz2. Copy it to `/opt/sccKit` and untar it.

```
tar xjvf sccKit_1.4.0.tar.bz2
```

Step 2

Edit `/etc/exports` to export 192.168.*x*.0/24 where *x* is greater than or equal to 2. It's a different value for each MCPC in the Data Center (for example, *x* is 3 on marc006). If you have a standalone MCPC/SCC system, you can choose *x* to be anything you want, as long as it's not 0 or 1. Examples in these instructions are taken from marc006. Here is the line to add to `/etc/exports`.

```
/shared 192.168.3.0/24(rw,insecure,no_root_squash,async)
```

`eth1` is on the subnet 192.168.x. `eth1` and the cores must be on the same subnet. `eth1` and the cores for each MCPC all plug into the same switch, and we don't want conflicts.

Step 3

Define a virtual IP for the eth1 NIC. Put this virtual IP on the same subnet as the BMC. If you are upgrading from sccKit 1.3.0, you already know what the IP of your BMC is. If you are bringing up a brand new MCPC/SCC system, refer to the file How to Set the IP Address of the BMC.

Your `/etc/network/interfaces` file should have entries for `eth0`, `eth1`, and `eth1:1`. This example shows eth0 as static; marc006 is actually DHCP. If you are using DHCP for your MCPC, refer to the file How to Configure the MCPC to use DHCP .

```
auto eth0
iface eth0 inet static
address 204.253.140.98
netmask 255.255.255.0
gateway 204.253.140.1

auto eth1
iface eth1 inet static
address 192.168.3.254
netmask 255.255.255.0
```

```
auto eth1:1
iface eth1:1 inet static
address 10.3.16.26
netmask 255.255.255.0
```

Step 4

Connect an Ethernet cable from Port A of the SCC to the Ethernet switch; this must be a Gigabit switch. Some systems do not have a useable Port A. To see what ports you have working, telnet to the BMC and look for the following line in the sign-on message.

```
Usable GB ETH 1111
```

The 1111 here indicates that all for EMAC ports are working. They read ABCD left to right. You need to see at least one 1. The instructions here assume that Port A, the leftmost port, is working.

Step 5

Edit `/opt/sccKit/systemSettings.ini`. Replace *x* with 2, 3, 4, etc.

```
[General]
CRBServer=204.253.140.66:5010
memorySize=8
platform=RockyLake
maxTransId=64
sccFirstMac=00:45:4D:41:44:31 ← anything as long as the first field is 00.
sccHostIp=192.168.x.254
sccFirstIp=192.168.x.1
sccMacEnable=a ← if you are using the leftmost port. Otherwise choose b, c, or d.
```

You can enable both Ports A and B or both Ports C and D. To do that you would specify `sccMacEnable=ab` or `sccMacEnable=cd`. When you specify two ports as in `sccMacEnable=ab`, the first 24 cores use Port A and the next 24 cores use Port B.

Step 6

Enter the directory `/etc/bind`. Copy the file `1.168.192.zone` and name the copy `x.168.192.zone`. Replace *x* with 2, 3, 4, etc. Remove but save the files `0.168.192.zone` and `1.168.192.zone`.

Be sure to update the serial number. A typical serial number looks like 0911161516. Increment the least significant digit. Here is a quote from the site, http://www.tech-recipes.com/rx/305/dnsbind-create-a-basic-zone-file/

> To make things fast and efficient, BIND processes zone files into another format. When BIND loads a zone file at startup, it checks the serial number and only processes the zone file if the serial number is bigger than its previously processed version. So, if you change the zone file but not the serial number, BIND will ignore the changes.

Step 7

Edit the file `ex.rck.zone`. Change 192.168.1.1..46 to 102.168.*x*.1..48. Also change 192.168.1.254 to 192.168.*x*.254.

Edit the file `in.rck.zone` in `/etc/bind`. Change the lines with 192.168.0.1..48 to 192.168.*x*.1..48. Also edit the `rckhost` line, changing 192.168.1.254 to 192.168.x.254.

Be sure to update the serial number in each file.

Step 8

Edit the file `ssh_config` in `/etc/ssh`. Remove the entries for Host 192.168.0.* and Host 192.168.1.*. Add an entry for Host 192.168.2.*. Use the same values for this entry.

Step 9

Edit the file `/etc/bind/named.conf..local`. Add the following lines to the end of the file.

```
zone "x.168.192.in-addr.arpa" IN {
    type master;
    file "/etc/bind/x.168.192.zone";
    allow-update { none; };
};
```

where *x* is the first digit of your new zone file.

Step 10

Enter the directory `/etc/ssh`. Edit the file `ssh_config`. Add the following lines at the end of the file. Replace *x* with 2, 3, 4, etc.

```
Host 192.168.x.*
IdentityFile /opt/sccKit/.ssh2/openssh_id_rsa
ForwardX11 no
StrictHostKeyChecking no
```

Step 11

Restart bind.

```
/etc/init.d/bind9 restart
```

Step 12

Enter the directory `/opt/sccKit`. Break the link current ➔ 1.3.0 and make the link current ➔1.4.0.

```
rm current
ln -s 1.4.0 current
```

Step 13

The sccKit tar file comes with a bitstream, which will be loaded into the FPGA. This bitstream is located in the directory `/opt/sccKit/1.4.0/firmware/RockyLake/update`. A typical name is `rl_20110110_ab.bit`.

Note the `_ab` suffix in the filename. This indicates that the bitstream is for eMAC ports A and B. It's not possible to have a single bitstream for all four eMAC ports because of FPGA size limitations. If you want to change from using eMAC ports A and B to C and D, you must change the FPGA bitstream.

To change the current bitstream, refer to How to Update the Bitstream.

Note the `_ab` suffix in the filename. This indicates that the bitstream is for eMAC ports A and B. It's not possible to have a single bitstream for all four eMAC ports because of FPGA size limitations. sccKit 1.4.0 comes with two bitstreams: an `_ab` bitstream and a `_cd` bitstream.

Enter the directory `/opt/sccKit/current/firmwa`re and execute `install.csh`.

Step 14

Either reboot the MCPC or issue the `sccPowercycle -r` command. For the `sccPowercycle` command to run successfully the following conditions must be met.

- The MCPC must be running Ubuntu Linux.
- The PCI bus ID of the Hib card (the PCIex card) needs to be smaller than the PCI bus ID of the graphics card; that is, the Hib card needs to use a PCIe card slot with a smaller ID.
- Both PCIe slots should be 16x.

To find out the BUS-ID, you can issue `lspci` on the MCPC. The ID (first number of each line) of the device that's called "Device c048" (for Copperridge) or "Device c148" (for Rocky Lake) needs to be smaller than the ID of the VGA controller. For example,

```
username@yourcomputer:/opt/sccKit/current/firmware$ lspci |grep VGA
04:00.0 VGA compatible controller: Matrox Graphics, Inc. MGA G200e [Pilot]
ServerEngines (SEP1) (rev 02)
username@yourcomputer:/opt/sccKit/current/firmware$ lspci |grep c148
01:00.0 Memory controller: Intel Corporation Device c148
username@yourcomputer:/opt/sccKit/current/firmware$
```

If the above conditions are met, issue the command

```
sccPowercycle -r
```

Step 15

Skip this step if you were able to run `sccPowercycle`.

Remove the `crbif` driver if it is loaded. You can tell if it's loaded with

```
lsmod |grep crbif
```

Remove it with

```
rmmod crbif
```

Attempting to remove `crbif` when it is not loaded is a benign operation.

Step 16

Skip this step if you were able to run `sccPowercycle`.

Telnet to the BMC. You can get the BMC IP address from `/opt/sccKit/systemSettings.ini`.

```
telnet BMC_IP_address 5010
```

Step 17

Skip this step if you were able to run `sccPowercycle`.

Issue the command

```
power off
```

Then wait about 60 seconds before issuing the commands

```
power on
exit
```

Step 18

Skip this step if you were able to run `sccPowercycle`

Shutdown the MCPC. Wait about 60 seconds. Bring the MCPC back up.

Step 19

The `crbif` driver should be loaded. Check that it is with `lsmod`. If it is not loaded, often a complete and careful power cycle will solve the problem. Please refer to the file How to 100% Power Cycle the MCPC/RockyLake System . If you still do not see the `crbif` driver loaded, please post your problem in the SCC Forum.

Step 20

Train the system interface. Reset the cores. Boot Linux on all the cores.

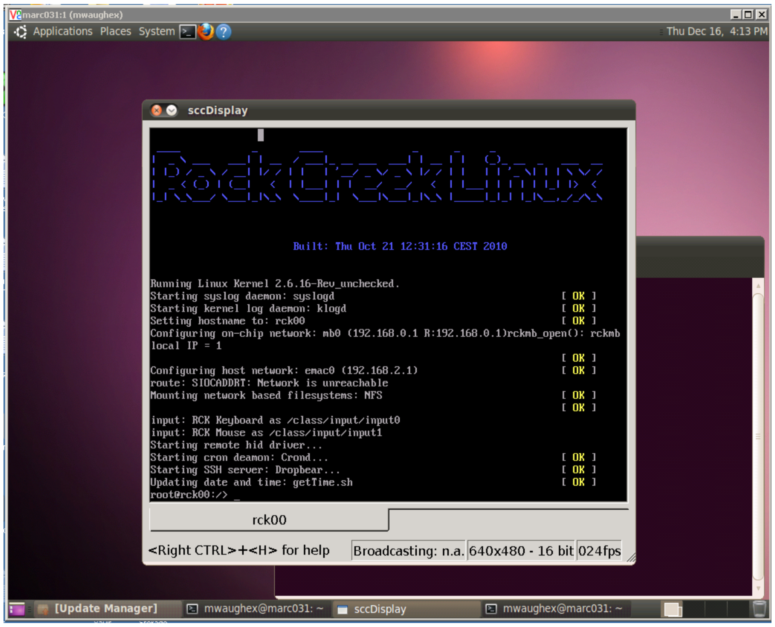
```
sccBmc -i
sccReset -g
sccBoot -l
```

Step 21

Try out the new sccKit. To use the GUI, you must have a desktop. If you are remote, you can use VNC. Please refer to the file How to VNC into the SCC DC from Windows .

Start up sccDisplay on one core.

```
sccDisplay 0
```

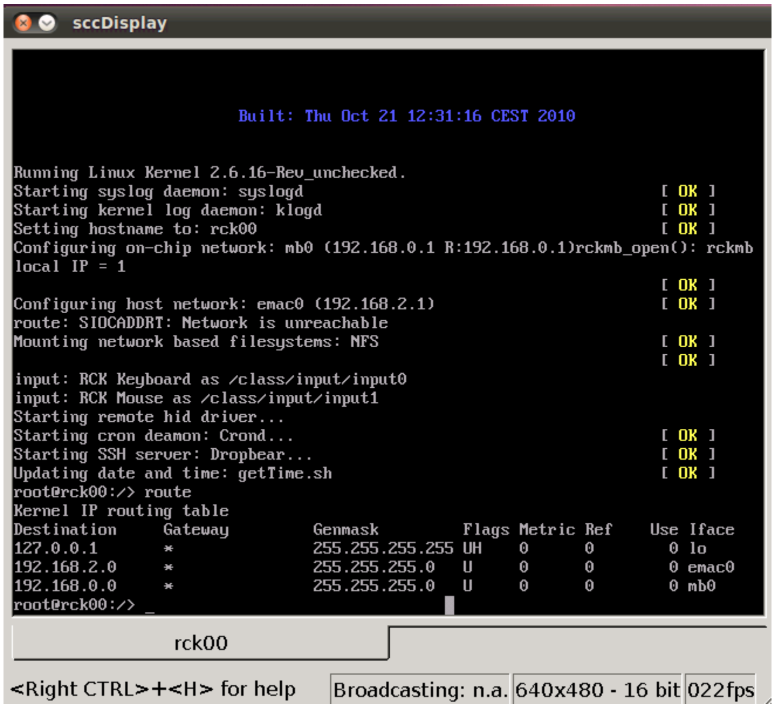


```
                Built: Thu Oct 21 12:31:16 CEST 2010


Running Linux Kernel 2.6.16-Rev_unchecked.
Starting syslog daemon: syslogd                              [ OK ]
Starting kernel log daemon: klogd                            [ OK ]
Setting hostname to: rck00                                   [ OK ]
Configuring on-chip network: mb0 (192.168.0.1 R:192.168.0.1)rckmb_open(): rckmb
local IP = 1
                                                             [ OK ]
Configuring host network: emac0 (192.168.2.1)                [ OK ]
route: SIOCADDRT: Network is unreachable
Mounting network based filesystems: NFS                      [ OK ]
                                                             [ OK ]
input: RCK Keyboard as /class/input/input0
input: RCK Mouse as /class/input/input1
Starting remote hid driver...
Starting cron deamon: Crond...                               [ OK ]
Starting SSH server: Dropbear...                             [ OK ]
Updating date and time: getTime.sh                           [ OK ]
root@rck00:/> _
```

Step 22

Invoke the `route` command. Note the emac0 interface.



```
                Built: Thu Oct 21 12:31:16 CEST 2010


Running Linux Kernel 2.6.16-Rev_unchecked.
Starting syslog daemon: syslogd                              [ OK ]
Starting kernel log daemon: klogd                            [ OK ]
Setting hostname to: rck00                                   [ OK ]
Configuring on-chip network: mb0 (192.168.0.1 R:192.168.0.1)rckmb_open(): rckmb
local IP = 1
                                                             [ OK ]
Configuring host network: emac0 (192.168.2.1)                [ OK ]
route: SIOCADDRT: Network is unreachable
Mounting network based filesystems: NFS                      [ OK ]
                                                             [ OK ]
input: RCK Keyboard as /class/input/input0
input: RCK Mouse as /class/input/input1
Starting remote hid driver...
Starting cron deamon: Crond...                               [ OK ]
Starting SSH server: Dropbear...                             [ OK ]
Updating date and time: getTime.sh                           [ OK ]
root@rck00:/> route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
127.0.0.1       *               255.255.255.255 UH    0      0        0 lo
192.168.2.0     *               255.255.255.0   U     0      0        0 emac0
192.168.0.0     *               255.255.255.0   U     0      0        0 mb0
root@rck00:/> _
```

How to Update the Bitstream

To update the bitstream, place the new bitstream file in `/opt/sccKit/1.4.0/firmware/RockyLake/update.`

- Edit `update.sh`. In that file, replace the name of the old bitstream with the new one
- Edit `update.txt`. Increment the number stored in that file.
- Ensure that the bitstream file has read access for everyone.
- Enter the directory `/opt/sccKit/1.4.0/firmware`.
- Issue the command `install.csh`. The `install.csh` script removes the `crbif` driver (the PCIe driver) if it is present. Then, it installs the driver package with `dkms`. The Ubuntu Linux on your MCPC must have `dkms` installed. Then, the script logs onto the BMC as root and updates the bitstream. When it completes, the SCC chip is powered off.
- If you see errors in the output from `install.csh`, issue the command `apt-get remove crbif-dkms` and rerun `install.csh`.
- If you are running sccKit 1.4.0, issue the command `sccPowercycle -r`. However, using `sccPowercycle -r` in place of rebooting the MCPC is only possible under certain conditions (which are usually met). Refer to Step 14. Otherwise remove `crbif`, tenet to the BMC, manually cycle the power, and reboot the MCPC. Refer to Step 15 through Step 18.