

---

## SoM-9260 Kernel Compiling

### Download the SDK

Create a working directory in your home directory called `som9260`.

Download the SoM-9260 SDK and extract it to your working directory. The current version can be found here:

```
ftp://ftp.emacinc.com/Controllers/SoM/SoM-9260M/Tools
```

The current version will have a name similar to `SOM9260-OE_SDK-<version>.eabi.tgz`. Using the most recent, highest version is recommended for new projects unless you have been directed otherwise by the EMAC-OE Development Team. New versions typically will have the same libraries with newer examples.

### Download the Kernel

This requires a Linux distribution with Internet access, Bash shell, and `wget` package.

Save the following files in the `som9260` directory:

```
ftp://ftp.emacinc.com/Controllers/SoM/SoM-9260M/Software/Linux-Kernel/scripts/kernel-build-cross.sh  
ftp://ftp.emacinc.com/Controllers/SoM/SoM-9260M/Software/Linux-Kernel/scripts/mkimage  
ftp://ftp.emacinc.com/Controllers/SoM/SoM-9260M/Software/Linux-Kernel/scripts/makeimage\_auto.sh  
ftp://ftp.emacinc.com/Controllers/SoM/SoM-9260M/Software/Linux-Kernel/scripts/patch\_at91\_e1.8.sh
```

These can be downloaded using `wget` from the command line:

```
wget <URL>
```

From a shell in the `som9260` directory execute:

```
chmod +x patch_at91_e1.8.sh  
./patch_at91_e1.8.sh
```

This will download the vanilla kernel, patches, and configuration file from EMAC's FTP server and assembles them into a SoM-9260 patched kernel tree directory. Processing time will vary with the speed of your Internet connection.

### Prepare the Cross-Compiler

Edit the script `kernel-build-cross.sh` and modify the `CROSS_COMPILE` variable to point to your working SoM-9260 SDK directory.

Change `<SDK_release>` to the release directory you are using on the following line:

```
CROSS_COMPILE=$HOME/som9260/<SDK_release>/gcc-4.1.1-arm-linux-gnueabi/bin/arm-linux-gnueabi-
```

Once the `CROSS_COMPILE` variable is set, make the `kernel-build-cross.sh` script executable.

```
chmod +x kernel-build-cross.sh
```

This example assumes that the user wants to create a build tree with the suffix of the current date.

### Build the Kernel

Run the kernel configuration option to generate the build directory.

```
./kernel-build-cross.sh linux-2.6.25.at91 config 20090305r0
```

If you do not want to make changes to the kernel, select Exit from the menu and save changes when prompted.

This will create a build directory called `build-2.6.25-20090305r0`. This command can be rerun as many times as needed to get the proper `.config` for the Linux kernel.

```
./kernel-build-cross.sh linux-2.6.25.at91 build 20090305r0
```

Compiling with the default options will take approximately 10 minutes on a modern multi-core CPU, however for development and debugging, multi-core/multi-threading has been turned off (`make -j` option). Time will vary depending on the modules required. After a successful compile, a gzipped tar file (also known as a tarball) called `kernel-2.6.25.tar.gz` is created in the build directory as well as a `zImage` file containing the kernel binary.

## Generate the uImage Boot Kernel

The `zImage` must be converted to a U-Boot `uImage` in order to be usage by the SoM-9260 bootloader.

To convert the kernel `zImage` to a `uImage` file, use the `makeimage_auto.sh` script. First make the script executable:

```
chmod +x ./makeimage_auto.sh
```

To generate the `uImage` file you will need the build directory without the `build-` prefix as the argument.

```
./makeimage_auto.sh 2.6.25-20090305r0
```

This will generate the file `uImage-2.6.25-20090305r0` which is now usable by U-Boot and the SoM-9260.

## Technical Support

Technical support is provided by EMAC as a service from one-year of date of purchase. If you have any questions regarding cross-compiling the kernel or the SoM-9260, please our EMAC Technical Support Inquiry web page. Technical Support requires a valid serial number of the product in question.

<http://www.emacinc.com/support/>