

Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free, no registration required.

## How to install cross compiler (on ubuntu 12.04 LTS) for microprocessor SA1100?

0 Can someone please tell me how to install the cross compiler (programming language C) for the SA1100 microprocessor? I have ubuntu 12.04 LTS. I'm a complete noob to Linux, I just installed Ubuntu yesterday. I need a special variant of the GCC compiler that is named "arm-unknown-linux-gnu-gcc" but don know how to do it.

Can someone please help me?

1 [linux](#) [gcc](#) [ubuntu](#) [compiler](#) [arm](#)

share improve this question

asked Feb 17 at 13:14

[n00b1990](#)

33

1 6

- Did you try `apt-get install gcc-arm-linux-gnueabi` or `apt-get install gcc-4.7-arm-linux-gnueabi`? I also strongly recommend being able to compile an ordinary C program for your Linux system (i.e. learn the basics of `gcc`, `make` ... commands and how to use some editor like `emacs` or `gedit` ...) and the cross compiler you want also depends upon the system running on your SA1100 hardware board.  
– [Basile Starynkevitch](#) Feb 17 at 13:56
- I do know how to compile an ordinary C program. That was `gcc -o hello hello.c`. I already tried `apt-get install gcc-arm-linux-gnueabi` and installed but cross compiling still doesn't work. I want to cross compile the programs with (for example) `arm-unknown-linux-gnu-gcc -O2 -Wall -o hello hello.c` but it tells me the command couldn be found. Can you also tell me what `make` is for? Thank you, I appreciate your help! – [n00b1990](#) Feb 17 at 14:44
- You should actually use `gcc -Wall -g hello.c -o hello`; About `make`, read [gnu.org/software/make/manual/html\\_node/index.html](http://gnu.org/software/make/manual/html_node/index.html) and use the `arm-linux-gnueabi-gcc` program as your cross-compiler – [Basile Starynkevitch](#) Feb 17 at 14:47
- Additional hint, use `dpkg -L gcc-arm-linux-gnueabi` to learn about the files installed by package `gcc-arm-linux-gnueabi` – [Basile Starynkevitch](#) Feb 17 at 14:53
- @BasileStarynkevitch You may want to post an answer. – [Elijah Kagan](#) Feb 17 at 16:11
- show 3 more comments

### 1 Answer

active

oldest

votes

As I told in comments, try

3 `apt-get install gcc-arm-linux-gnueabi`

or

`apt-get install gcc-4.7-arm-linux-gnueabi`

I also strongly recommend being able to compile an ordinary C program for your Linux system (i.e. learn the basics of `gcc`, `make` ... commands and how to use some editor like `emacs` or `gedit` ...) and the cross compiler you want also depends upon the system running on your SA1100 hardware board. Don't forget to pass `-Wall` to any GCC compilation. You probably want to be able to debug your program (pass `-g` to GCC at compilation, and use the `gdb` debugger). When your program is running well, compile it with `-O2` to ask GCC to optimize its machine code.

Learn to use GNU `make` -e.g. to write `Makefile-s` by reading its [documentation](#) and use the `arm-linux-gnueabi-gcc` as the cross-compiler program. (You might want to use `remake` to debug your `Makefile-s` when `make` does not help enough)

You can get the list of files installed with a package with e.g. `dpkg -L gcc-arm-linux-gnueabi`

A cross compiled program executable for ARM very probably needs a Linux kernel with some `libc` (or link it statically) at least on the ARM motherboard, and you need some way to transmit the binary program from the Linux desktop to the ARM hardware.

share improve this answer

edited Feb 17 at 17:15

answered Feb 17 at 16:24

[Basile Starynkevitch](#)

45.8k

21 52

4

- +1 solely for the general advices, I can't keep repeating them enough: – [H2CO3](#) Feb 17 at 16:27
- +1 for the great answer, and the fact that I got help this quickly :) – [n00b1990](#) Feb 17 at 18:35