

Arm DS 安装激活手册

目录

安装要求.....	2
软件下载.....	2
ARM DS 软件下载.....	2
单机版.....	3
一、 下载 Arm DS license 文件.....	3
1.1 注册ARM 帐户.....	3
1.2 登录ARM 帐户, 下载 License 文件.....	5
二、 软件安装.....	8
2.1 Windows 版安装.....	8
2.2 Linux 版安装.....	8
三、 激活软件.....	11
3.1 Windows 版.....	11
3.2 Linux 版.....	15
网络版.....	18
一、 下载 ARM DS license 文件.....	18
1.1 注册ARM 帐户.....	18
1.2 登录ARM 帐户, 下载 License 文件.....	20
二、 license 合并.....	24
三、 软件安装.....	24
3.1 Windows 版安装.....	24
3.2 Linux 版安装.....	25
四、 服务器端管理工具安装.....	28
Windows 版.....	28
五、 激活软件.....	30
5.1 Windows 版.....	30
5.2 Linux 版.....	34

安装要求

操作系统			硬件要求
Windows 10	红帽	Ubuntu	双核2GHZ及以上
	企业版 Linux 7工作站	桌面版18.04 LTS	4GB RAM及以上
		桌面版 20.04 LTS	4GB以上可用 硬盘空间

软件下载

ARM DS 软件下载

ARM DS 软件下载

<https://developer.arm.com/tools-and-software/embedded/arm-development-studio/downloads>

Arm compiler 软件下载

<https://developer.arm.com/tools-and-software/embedded/arm-compiler/downloads/version-6>

FlexNet 软件下载

<https://developer.arm.com/downloads/-/downloads>

单机版

一、 下载 Arm DS license文件

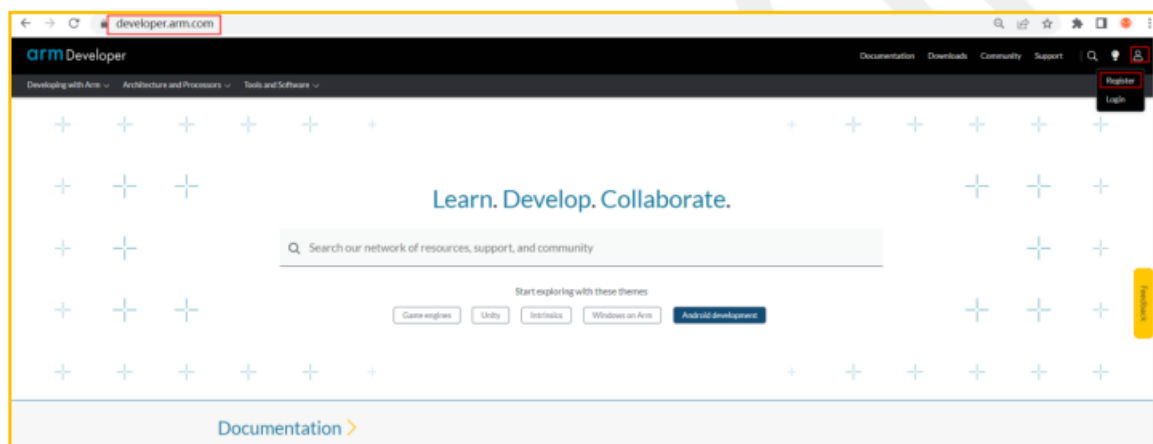
1.1 注册 ARM 帐户

如果您已经有 ARM 帐户了，请跳过”注册账号”这一节，直接看第二节”登录 ARM

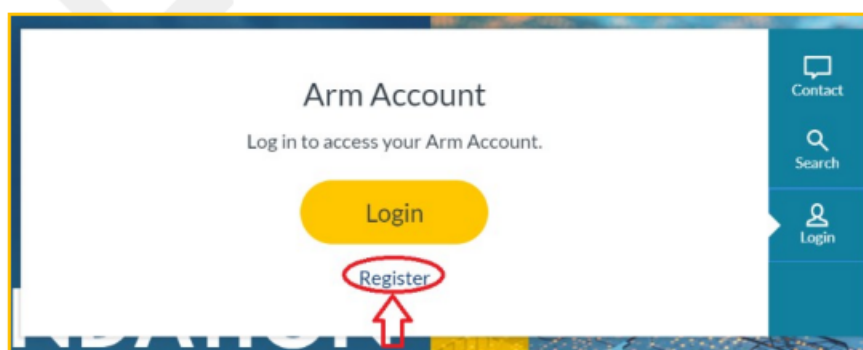
帐户，下载 License 文件”

如果打开 ARM 官网太慢，建议用笔记本电脑连接手机热点来访问 ARM 官网

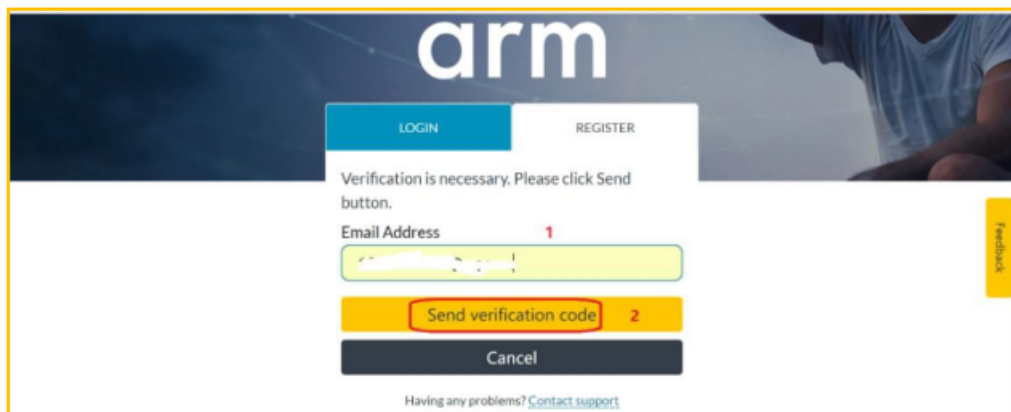
- 1) 打开浏览器在地址栏中输入<https://developer.arm.com/> 进入 ARM 官网



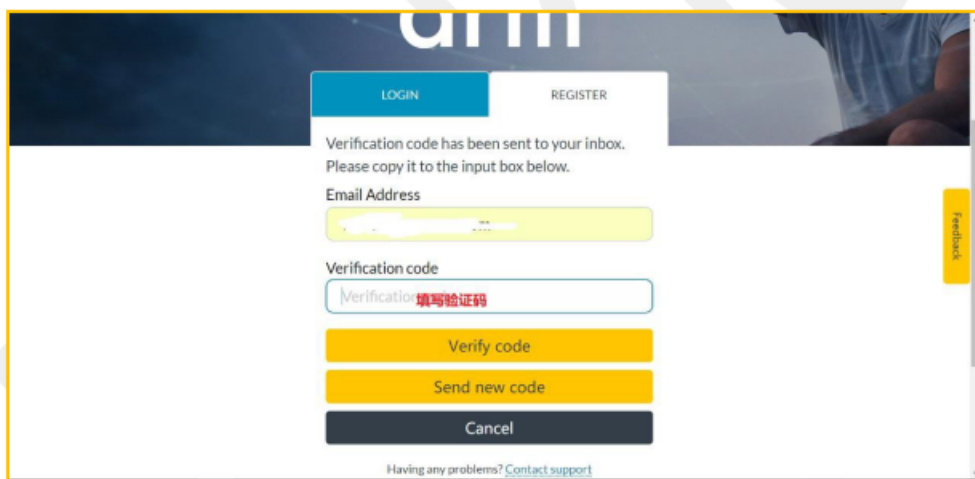
- 2) 点击右上角  人物图标, 选择 Register 进入注册页面



- 3) 在 Email address 栏处输入需要注册的邮箱地址，然后点击 Send verification code 按钮，向填写的邮箱发送一份带有验证码的邮件。



- 4) 收取邮件并将验证码填写到 Verification code 栏处，然后点击 Verify code 按钮。



注意：如果您没有收到验证码可以点击 **Send new code** 按钮重新向刚才填写的邮箱中发送验证码;如果还是没有收到邮件，则有可以是邮箱过滤了一些陌生邮件，请到垃圾邮件中查看。但垃圾邮件中还没找到时，则可能是邮箱被设置了拦截功能请联系 IT 帮忙处理。

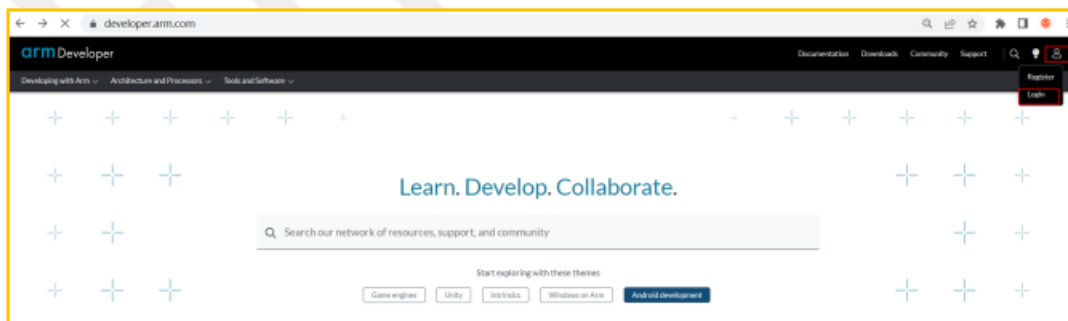
- 5) 填写相关信息，然后点击 Create 按钮创建帐户



The image shows the ARM Developer registration form. It has a header with the ARM logo and tabs for 'LOGIN' and 'REGISTER'. The 'REGISTER' tab is active. The form asks for 'Email Address', 'First name', 'Last name', 'Country' (a dropdown menu), 'Password', and 'Confirm password'. Below these fields, there is a checkbox for 'I would like to receive marketing communications from Arm.' and a link to 'Arm's privacy policy'. At the bottom, there are two buttons: 'Create' (highlighted with a red circle) and 'Cancel'. A footer link says 'Having any problems? Contact support'.

1.2 登录ARM 帐户，下载 License 文件

- 1) 在浏览器的地址栏中输入<https://developer.arm.com/> 进入 ARM 官网，
右上角  人物图标点击**Login in** 按钮。



LOGIN

REGISTER

Email Address

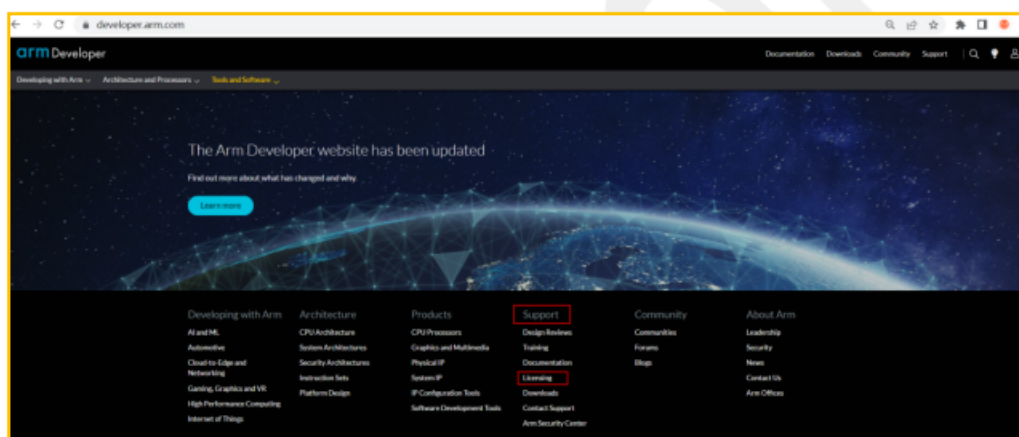
peter.liu@emdoor.com

Password

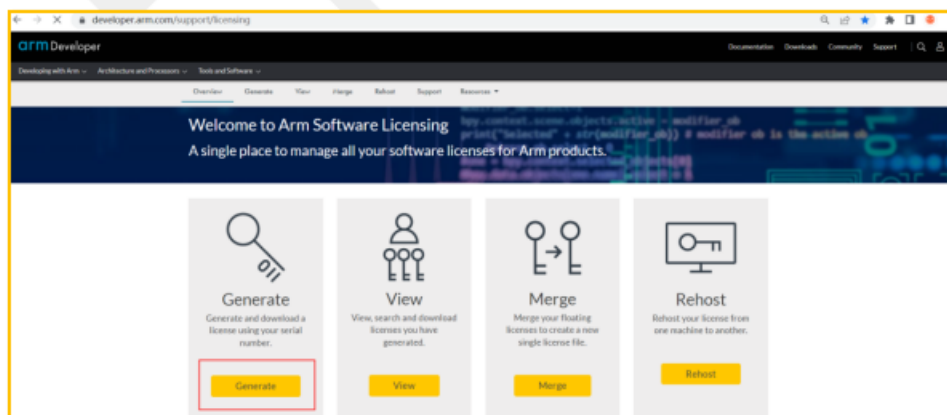
Log in

Forgot your password?

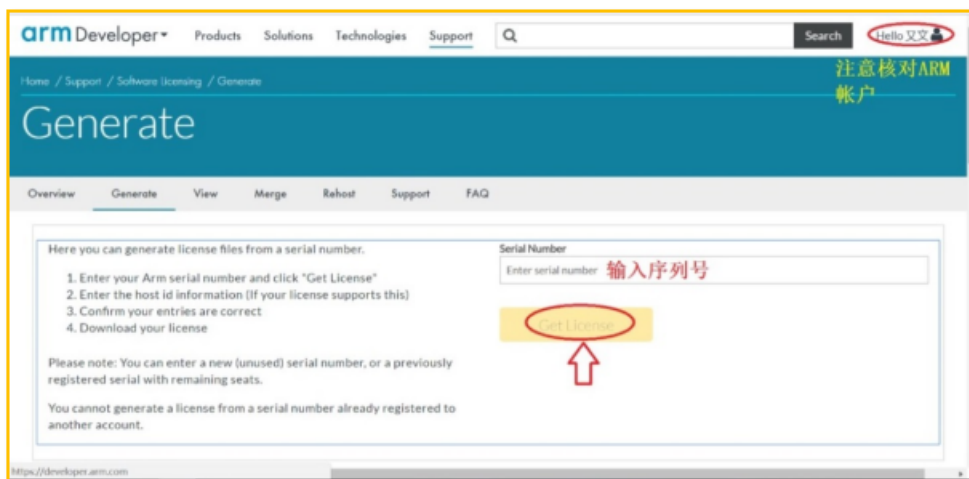
2) 向下拖动垂直滑动条，找到 Support-->License按钮。



3) 点击 licensing 按钮，找到 Generate 按钮。



4) 输入序列号，然后点击 Get License 按钮生成



- 5) 选择 HostID Type 处选择类型为 ETHERNET，在 HostID 处填写服务器的 MAC 地址，然后点击 Generate 按钮进入确认页面。核对下 MAC 地址确认无误后点击 Confirm 按钮。

- 6) 点击 Download License 按钮下载 license 文件，如果没有出现下载窗口请设置浏览器允许弹窗。

- 7) 在打开的页面中点击 Save 按钮，下载 License 文件。


```
peter@ubuntu: ~/ArmDs
peter@ubuntu:~/ArmDs$ ls
DS000-BN-00001-r21p2-00rel0.tar
peter@ubuntu:~/ArmDs$ tar xzvf DS000-BN-00001-r21p2-00rel0.tar
DS000-BN-00001-r21p2-00rel0/
DS000-BN-00001-r21p2-00rel0/license_terms/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/models/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/models/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/dstream-st/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/dstream-st/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/streamline/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/streamline/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/arm_compiler_for_embedded_g/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/arm_compiler_for_embedded_g/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/graphics_analyzer/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/graphics_analyzer/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/examples/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/examples/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/debugger/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/debugger/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/redistributables.txt
DS000-BN-00001-r21p2-00rel0/license_terms/license_agreement.txt
DS000-BN-00001-r21p2-00rel0/license_terms/supplementary_terms.txt
DS000-BN-00001-r21p2-00rel0/readme.html
DS000-BN-00001-r21p2-00rel0/armds-2021.2.sh
peter@ubuntu:~/ArmDs$
```

3) 执行安装，sudo ./armds-2021.2.sh

```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0
peter@ubuntu:~/ArmDs/DS000-BN-00001-r21p2-00rel0$ ls
armds-2021.2.sh  license_terms  readme.html
peter@ubuntu:~/ArmDs/DS000-BN-00001-r21p2-00rel0$ sudo ./armds-2021.2.sh
```

```
END USER LICENCE AGREEMENT FOR THE ARM(r) DS-5 DEVELOPMENT STUDIO ("DS-5")

THIS END USER LICENCE AGREEMENT ("LICENCE") IS A LEGAL AGREEMENT BETWEEN YOU (EITHER A SINGLE INDIVIDUAL, OR SINGLE LEGAL ENTITY) AND ARM LIMITED ("ARM") FOR THE USE OF THE SOFTWARE ACCOMPANYING THIS LICENCE. ARM IS ONLY WILLING TO LICENSE THE SOFTWARE TO YOU ON CONDITION THAT YOU ACCEPT ALL OF THE TERMS IN THIS LICENCE. BY CLICKING "I AGREE" OR BY INSTALLING OR OTHERWISE USING OR COPYING THE SOFTWARE YOU INDICATE THAT YOU AGREE TO BE BOUND BY ALL OF THE TERMS OF THIS LICENCE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENCE, ARM IS UNWILLING TO LICENSE THE SOFTWARE TO YOU AND YOU MAY NOT INSTALL, USE OR COPY THE SOFTWARE, AND YOU SHOULD PROMPTLY RETURN THE SOFTWARE TO YOUR SUPPLIER AND ASK FOR A REFUND OF ANY LICENCE FEE PAID (IF ANY).

"Clang" means a compiler front end developed by http://clang.llvm.org.

"Licence Key" means an electronic licence key issued to you by ARM to enable the use of the Software.

"LLVM" means a code generation and optimization framework developed by http://llvm.org.

"Seat" means a written or electronic authorisation from ARM to run one copy of each of the licence managed components (as defined in the Software package) concurrently and only in the quantities stated in the sales order and delivery confirmation email, for the time period enabled by your Licence Key, and where applicable, only with the target devices specified in the sales order and/or licence file.

"Separate Files" means the separate files identified in Section 2 of the Schedule.

Please answer with one of: 'yes' or 'no/quit'
```

按“q”退出阅读 License 信息，输入“Yes”直接回车，开始安装 ARM DS。

```
Please answer with one of: 'yes' or 'no/quit'  
Do you agree to the above terms and conditions? yes
```

4) 确认安装平台信息，直接回车

```
Please answer with one of: 'yes/y' or 'no/n'  
Run installation platform requirement checks? [default: yes]
```

5) 选择安装 ARM DS 的路径：

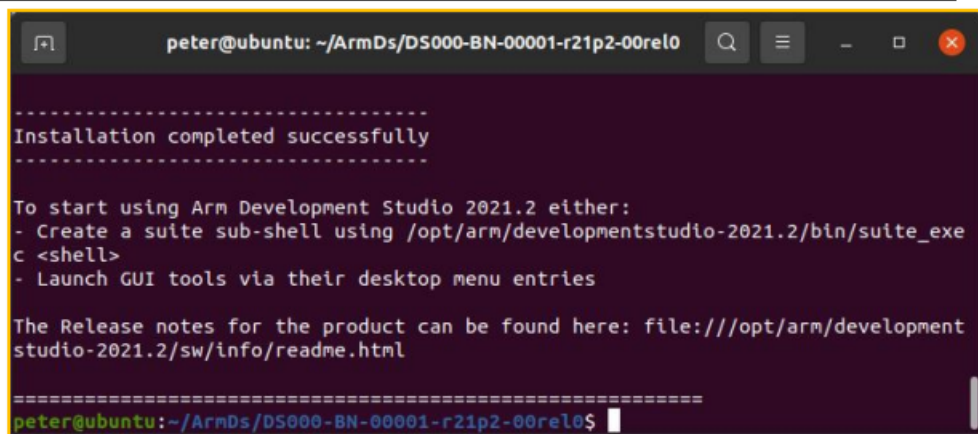
```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0  
*** WARNING ***  
One or more of the Arm Development Studio system dependencies are not installed,  
see below for a list of the missing libraries.  
  
libz.so.1 (32-bit)  
One of:  
libwebkit-1.0.so.2 (64-bit)  
libwebkitgtk-1.0.so.0 (64-bit)  
libwebkitgtk-3.0.so.0 (64-bit)  
  
We recommend you install the missing dependencies, otherwise the programs will n  
ot run correctly. For further information see the Arm Development Studio readme.  
  
Where would you like to install to? [default: /opt/arm/developmentstudio-2021.2]  
|
```

是否创建安装目录，默认直接回车开始安装

```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0  
libwebkit-1.0.so.2 (64-bit)  
libwebkitgtk-1.0.so.0 (64-bit)  
libwebkitgtk-3.0.so.0 (64-bit)  
  
We recommend you install the missing dependencies, otherwise the programs will n  
ot run correctly. For further information see the Arm Development Studio readme.  
  
Where would you like to install to? [default: /opt/arm/developmentstudio-2021.2]  
  
Please answer with one of: 'yes/y' or 'no/n'  
'/opt/arm/developmentstudio-2021.2' does not exist, create? [default: yes]  
  
--- Installing to '/opt/arm/developmentstudio-2021.2' (This may take a while...)  
|
```

根据不同电脑配置安装时间一般从几分钟
到半个小时不等；

6) 安装完成



```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0
-----
Installation completed successfully
-----

To start using Arm Development Studio 2021.2 either:
- Create a suite sub-shell using /opt/arm/developmentstudio-2021.2/bin/suite_exe
  c <shell>
- Launch GUI tools via their desktop menu entries

The Release notes for the product can be found here: file:///opt/arm/development
studio-2021.2/sw/info/readme.html

=====
peter@ubuntu:~/ArmDs/DS000-BN-00001-r21p2-00rel0$
```

➤ 单独编译器安装

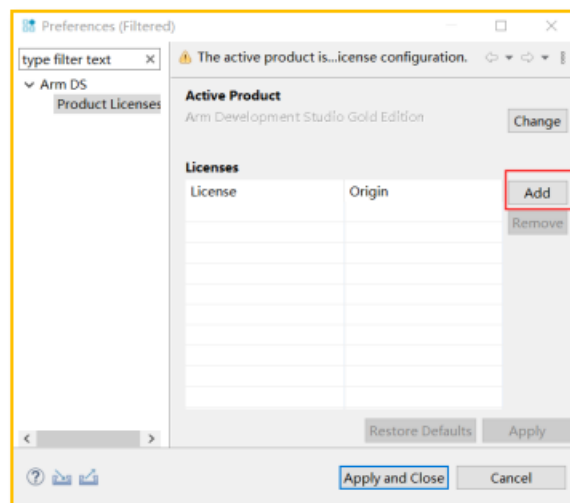
解压并切换到 **Arm compiler** 解压的目录中，然后在命令行终端中输入 **./setup.sh** 就行

三、 激活软件

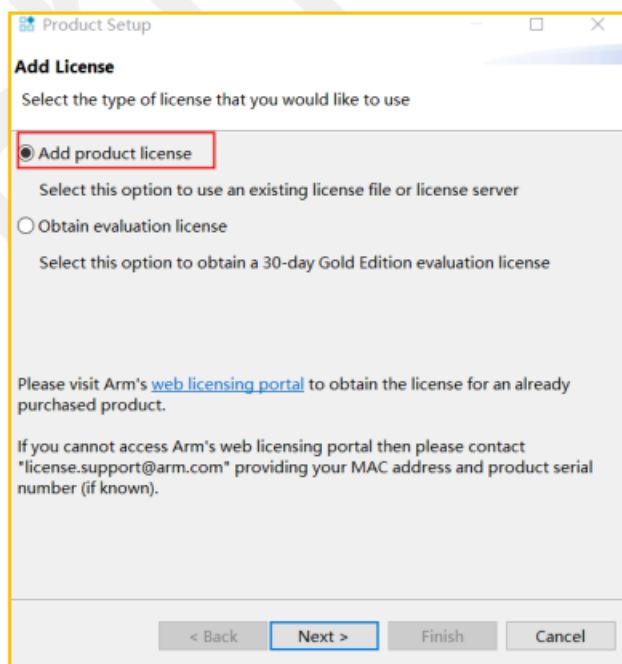
3.1 Windows 版

➤ 使用 ARM DS 软件

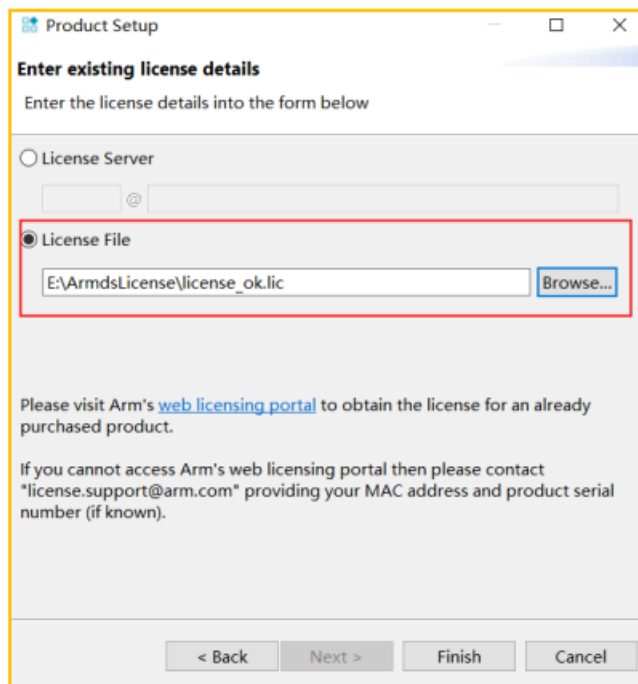
- 1) 打开Eclipse for ARM DS。
- 2) 选择“ Help> ARM License Manager”，再点击“Add”。



- 3) 选择第一个选项，再点击“Next”。



4) 选择第二个选项“License file”，添加好License文件，再点击“Finish”。



➤ 单独使用编译器

注：只能是Arm Development Studio Silver, Gold or Platinum 版本的 license 才兼容以前的旧版本编译器，Arm Development Studio Evaluation or Bronze 版本不支持。

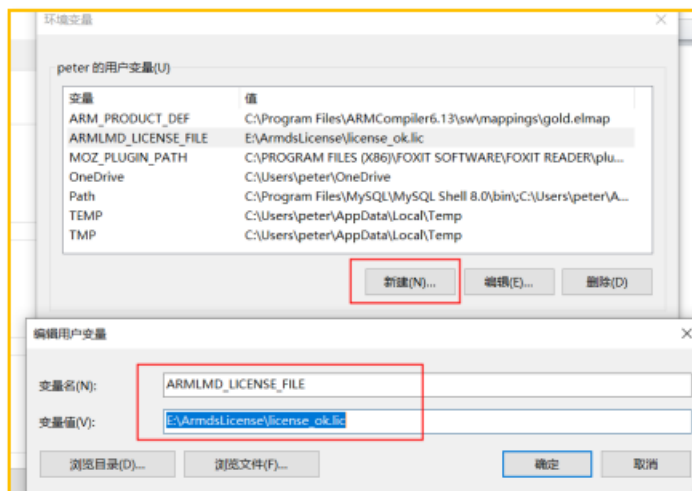
1) 产品类型说明文件所在路径如下，也可以把对应的产品类型说明文件复制到 armclang 的路径下。



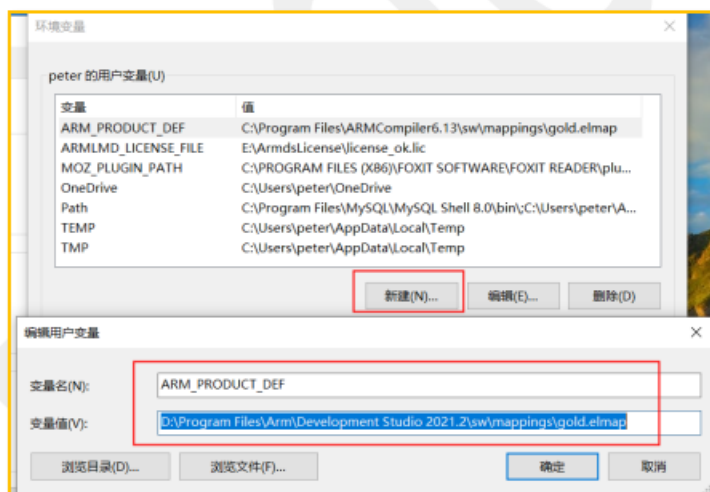
注意：对于 arm 原厂给的 1 年(给 FAE 的)都是 eval.elmap，正常购买的才是按照 arm ds

的版本类型来选择合适的文件。如果不确定用哪个，可以逐个尝试。

- 2) 设置 License 环境变量 ARMLMD_LICENSE_FILE,
ARMLMD_LICENSE_FILE=license 文件路径, 如:



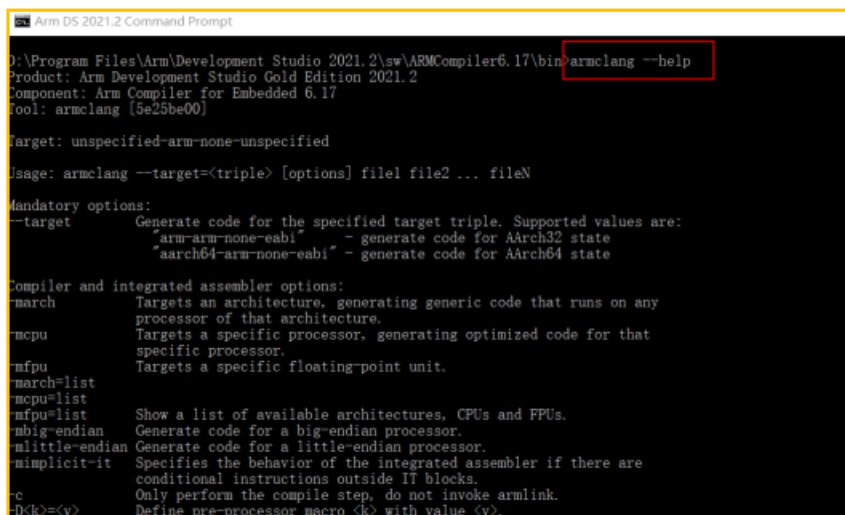
- 3) 创建 ARM_PRODUCT_DEF 环境变量
ARM_PRODUCT_DEF=elmap 文件的绝对路径设置步骤和
ARMLMD_LICENSE_FILE 类似



4) 验证:

在 cmd 窗口中切换到arm 编译器路径, 输入编译器名称并回车。比如: `armclang`

`--help`



```
Arm DS 2021.2 Command Prompt
D:\Program Files\Arm\Development Studio 2021.2\sw\ARMCompiler6.17\bin>armclang --help
Product: Arm Development Studio Gold Edition 2021.2
Component: Arm Compiler for Embedded 6.17
Tool: armclang [5e25be00]

target: unspecified-arm-none-unspecified
Usage: armclang --target=<triple> [options] file1 file2 ... fileN

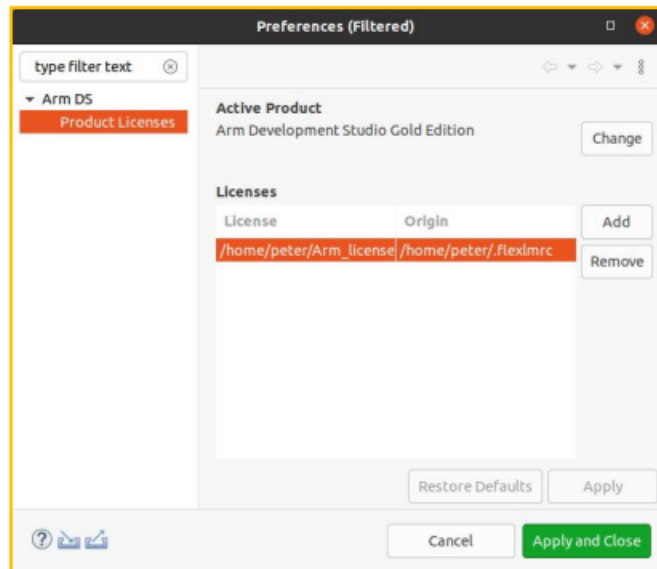
Mandatory options:
--target          Generate code for the specified target triple. Supported values are:
                  "arm-arm-none-eabi" - generate code for AArch32 state
                  "aarch64-arm-none-eabi" - generate code for AArch64 state

Compiler and integrated assembler options:
-march            Targets an architecture, generating generic code that runs on any
                  processor of that architecture.
-mcpu            Targets a specific processor, generating optimized code for that
                  specific processor.
-mfpu            Targets a specific floating-point unit.
-march=list       Show a list of available architectures, CPUs and FPU's.
-mcpu=list        Generate code for a big-endian processor.
-mfpu=list        Generate code for a little-endian processor.
-mbig-endian      Specifies the behavior of the integrated assembler if there are
-mlittle-endian   conditional instructions outside IT blocks.
-mimplicit-it     Only perform the compile step, do not invoke armlink.
-c               Define pre-processor macro <k> with value <v>.
```

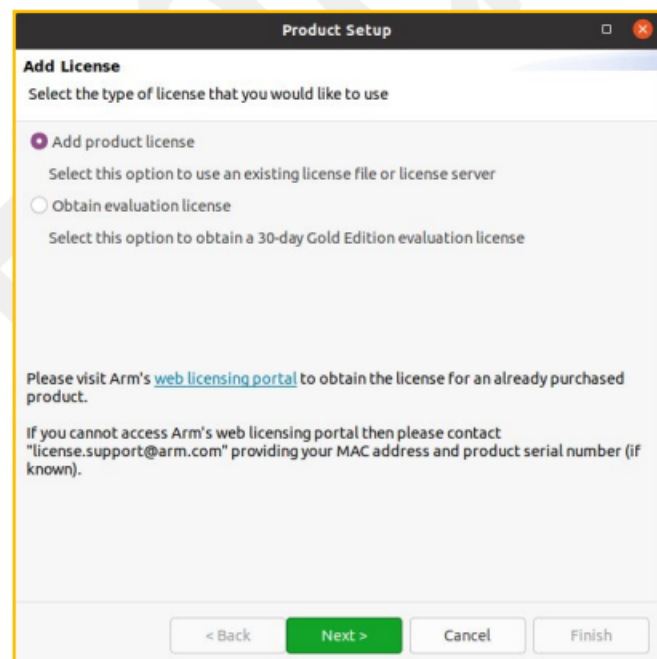
3.2 Linux 版

➤ 使用 ARM DS 软件

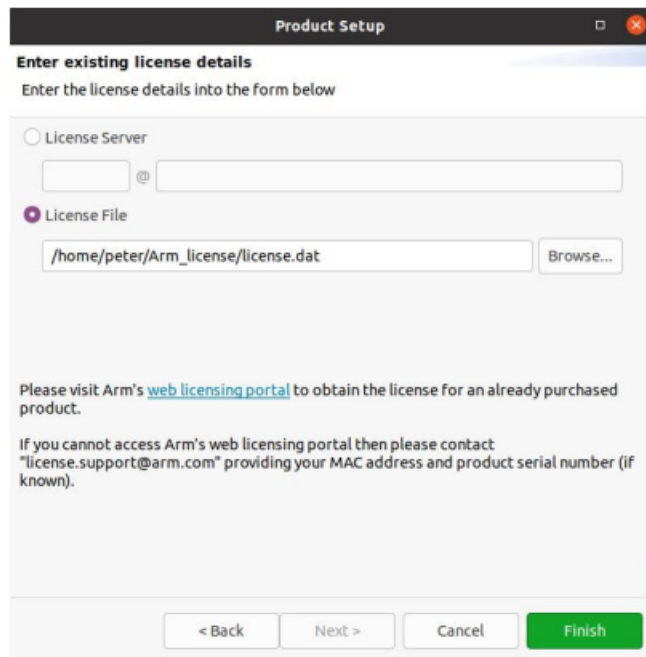
- 1) 打开Eclipse for ARM DS。
- 2) 选择“ Help> ARM License Manager”，再点击“Add”。



- 3) 选择第一个选项，再点击“Next”。



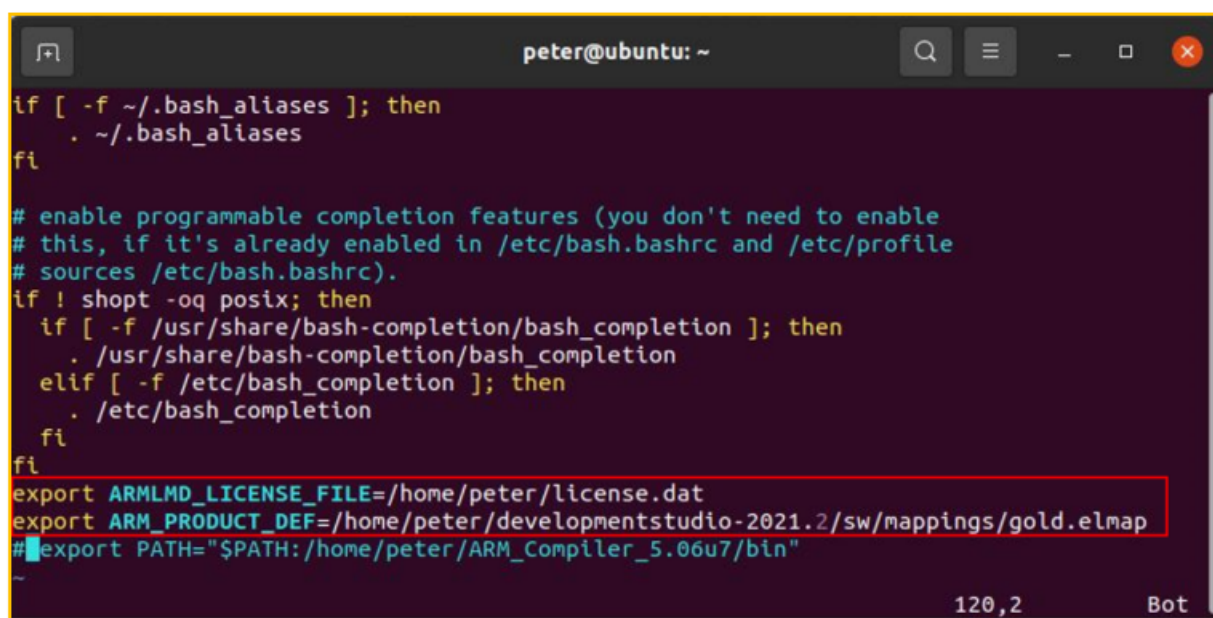
4) 选择第二个选项“License file”，添加好License文件，再点击“Finish”。



➤ 单独使用 arm compiler

注：只能是Arm Development Studio Silver, Gold or Platinum版本的license才兼容以前的旧版本编译器，Arm Development Studio Evaluation or Bronze版本不支持。

- 1) 在用户目录下的“.bashrc”文件中添加“ARMLMD_LICENSE_FILE”和“ARM_PRODUCT_DEF”环境变量的相关内容，如下图所示：



```
peter@ubuntu: ~  
if [ -f ~/.bash_aliases ]; then  
    . ~/.bash_aliases  
fi  
  
# enable programmable completion features (you don't need to enable  
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile  
# sources /etc/bash.bashrc).  
if ! shopt -oq posix; then  
    if [ -f /usr/share/bash-completion/bash_completion ]; then  
        . /usr/share/bash-completion/bash_completion  
    elif [ -f /etc/bash_completion ]; then  
        . /etc/bash_completion  
    fi  
fi  
export ARMLMD_LICENSE_FILE=/home/peter/license.dat  
export ARM_PRODUCT_DEF=/home/peter/developmentstudio-2021.2/sw/mappings/gold.elmap  
# export PATH="$PATH:/home/peter/ARM_Compiler_5.06u7/bin"
```

ARMLMD_LICENSE_FILE=license 文件的绝对路径

ARM_PRODUCT_DEF=elmap 文件的绝对路径

对于arm原厂给的1年(给FAE的)都是eval.elmap，正常购买的才是按照arm ds的版本类型来选择不同的文件。如果不确定用哪个，可以逐个尝试。

2) 验证:

在 cmd 窗口中切换到arm 编译器路径,输入编译器名称并回车。比如: armclang
--help

```
peter@ubuntu: /opt/arm/developmentstudio-2021.2/sw/AR...
peter@ubuntu:/opt/arm/developmentstudio-2021.2/sw/ARMCompiler6.17/bin$ ./armclang --help
Product: Arm Development Studio Gold Edition 2021.2
Component: Arm Compiler for Embedded 6.17
Tool: armclang [5e25bf00]

Target: unspecified-arm-none-unspecified

Usage: armclang --target=<triple> [options] file1 file2 ... fileN

Mandatory options:
--target          Generate code for the specified target triple. Supported values
are:
                  "arm-arm-none-eabi"      - generate code for AArch32 state
                  "aarch64-arm-none-eabi" - generate code for AArch64 state

Compiler and integrated assembler options:
-march            Targets an architecture, generating generic code that runs on an
y
                  processor of that architecture.
-mcpu            Targets a specific processor, generating optimized code for that
specific processor
```

网络版

一、 下载 ARM DS license文件

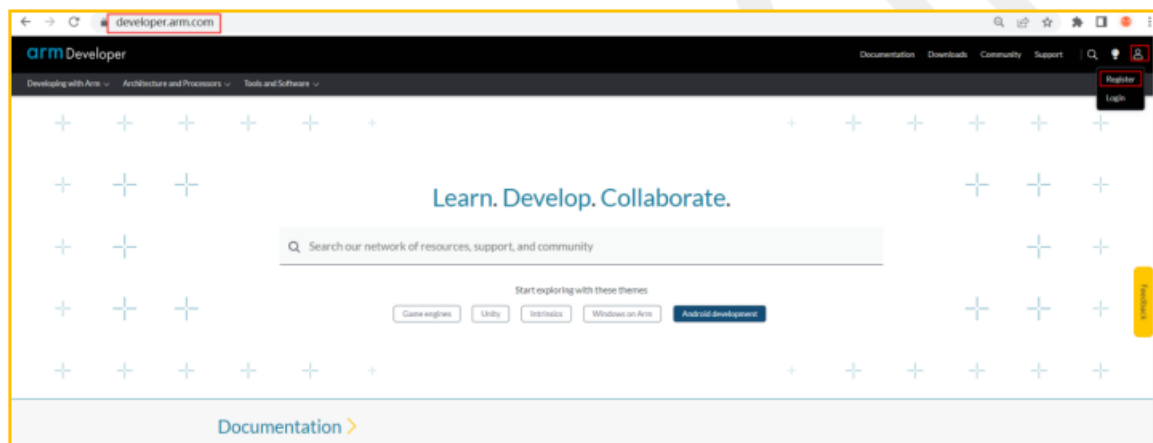
1.1 注册ARM 帐户

如果您已经有 ARM 帐户了，请跳过”注册账号”这一节，直接看第二节”
登录 ARM

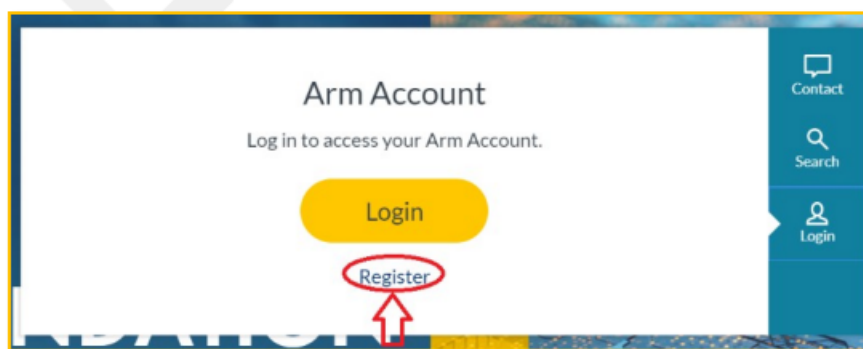
帐户，下载 License 文件”

如果打开 ARM 官网太慢，建议用笔记本电脑连接手机热点来访问 ARM 官网

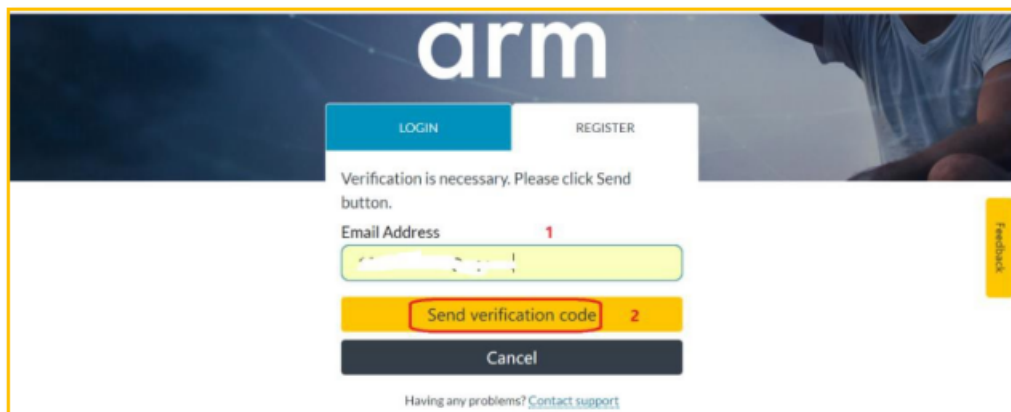
- 1) 打开浏览器在地址栏中输入<https://developer.arm.com/> 进入 ARM 官网



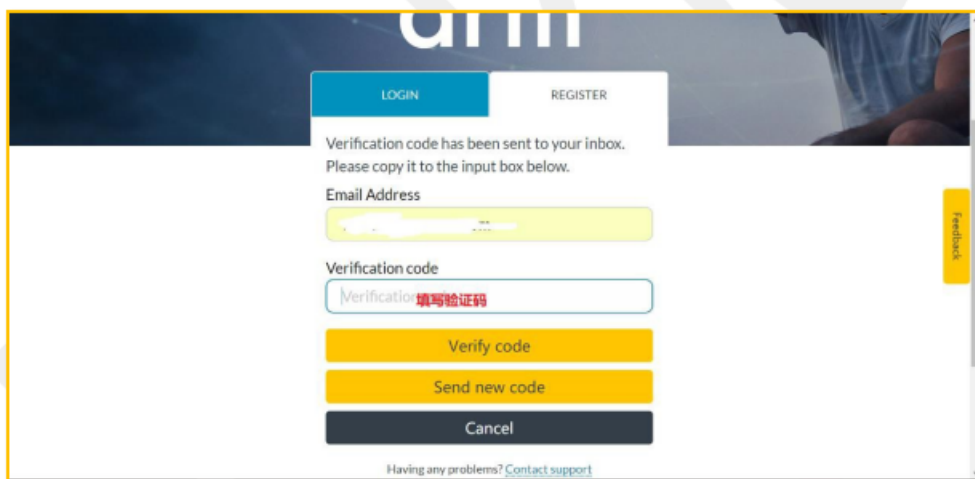
- 2) 点击右上角  人物图标, 选择 Register 进入注册页面



- 3) 在 Email address 栏处输入需要注册的邮箱地址，然后点击 Send verification code 按钮，向填写的邮箱发送一份带有验证码的邮件。



- 4) 收取邮件并将验证码填写到 Verification code 栏处，然后点击 Verify code 按钮。



注意：如果您没有收到验证码可以点击 **Send new code** 按钮重新向刚才填写的邮箱中发送验证码;如果还是没有收到邮件，则有可以是邮箱过滤了一些陌生邮件，请到垃圾邮件中查看。但垃圾邮件中还没找到时，则可能是邮箱被设置了拦截功能请联系 IT 帮忙处理。

- 5) 填写相关信息，然后点击 Create 按钮创建帐户



The image shows the ARM Developer registration form. It has a header with the ARM logo and tabs for LOGIN and REGISTER. The REGISTER tab is active. The form asks for the following details:

- Email Address (with a placeholder email address)
- First name
- Last name
- Country (a dropdown menu)
- Password (with a placeholder '填写密码')
- Confirm password (with a placeholder '再次填写密码')

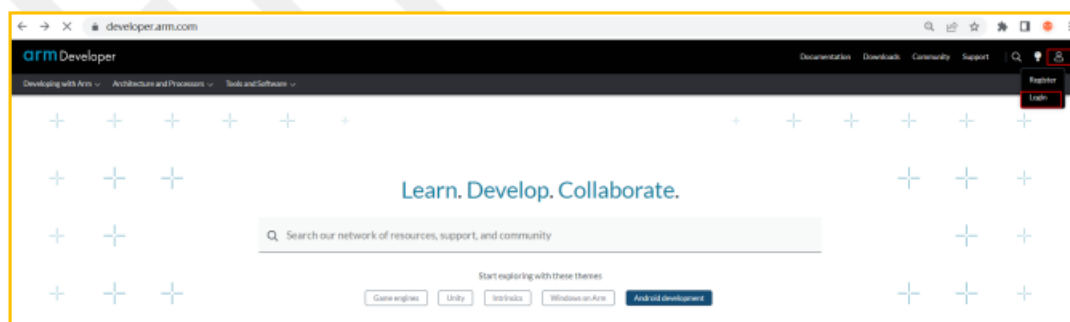
Below the password fields, there is a statement: "Arm will process your information in accordance with [Arm's privacy policy](#)." and a checkbox for "I would like to receive marketing communications from Arm." which is currently unchecked.

At the bottom, there are two buttons: "Create" (highlighted with a red circle and labeled "点击创建") and "Cancel".

At the very bottom, there is a link: "Having any problems? [Contact support](#)."

1.2 登录ARM 帐户，下载 License 文件

- 1) 在浏览器的地址栏中输入<https://developer.arm.com/> 进入 ARM 官网，
右上角人物图标点击**Login in** 按钮。



LOGIN

REGISTER

Email Address

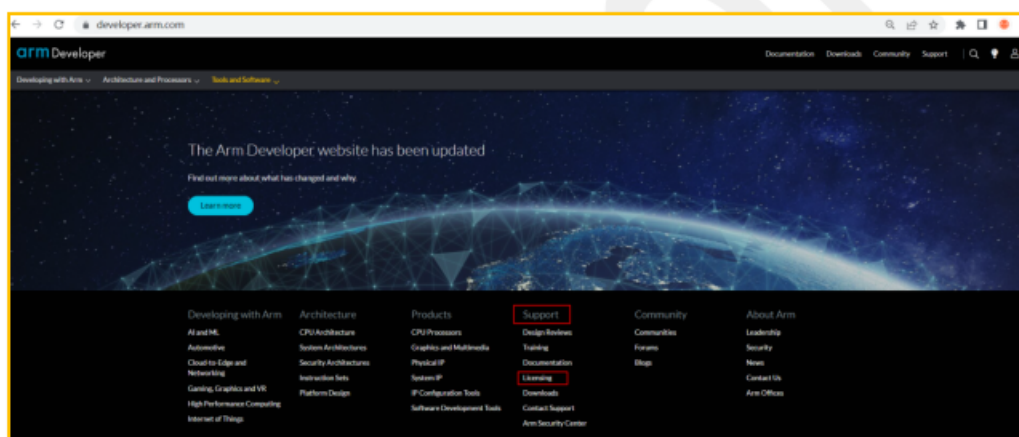
peter.liu@emdoor.com

Password

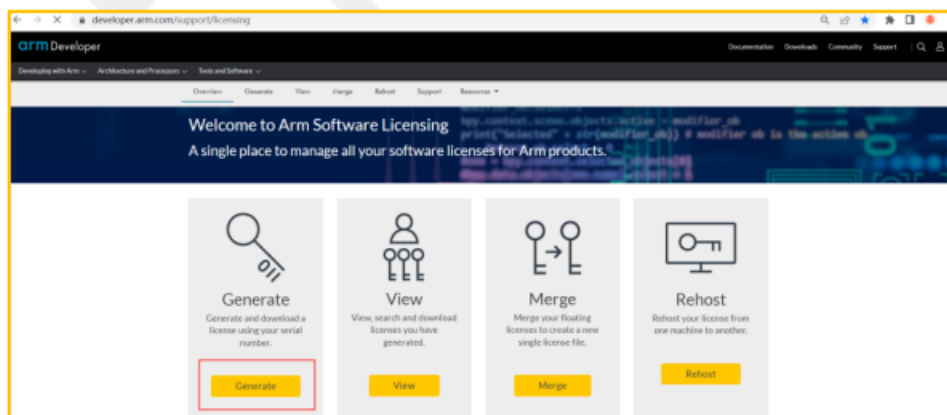
Log in

Forgot your password?

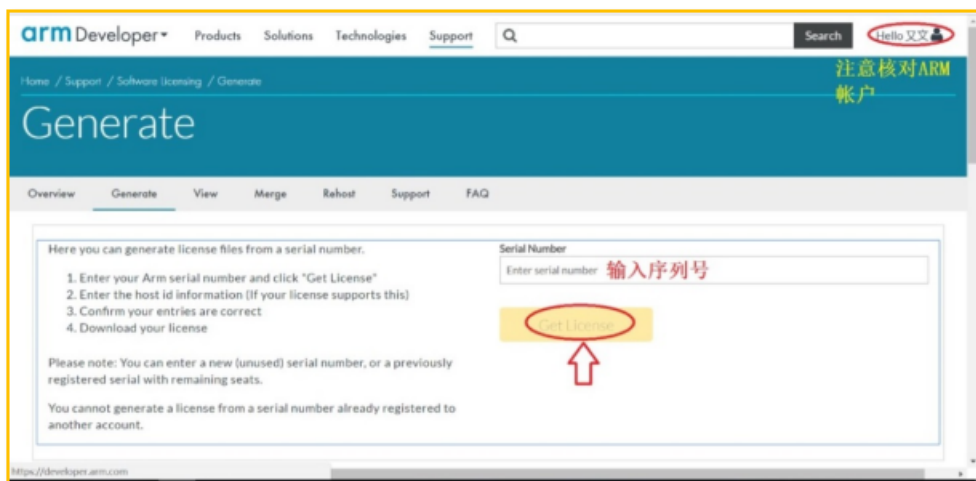
2) 向下拖动垂直滑动条，找到 Support-->License按钮。



3) 点击 licensing 按钮，找到 Generate 按钮。



4) 输入序列号，然后点击 Get License 按钮生成



- 5) 选择 HostID Type 处选择类型为 ETHERNET，在 HostID 处填写服务器的 MAC 地址，然后点击 Generate 按钮进入确认页面。核对下 MAC 地址确认无误后点击 Confirm 按钮。

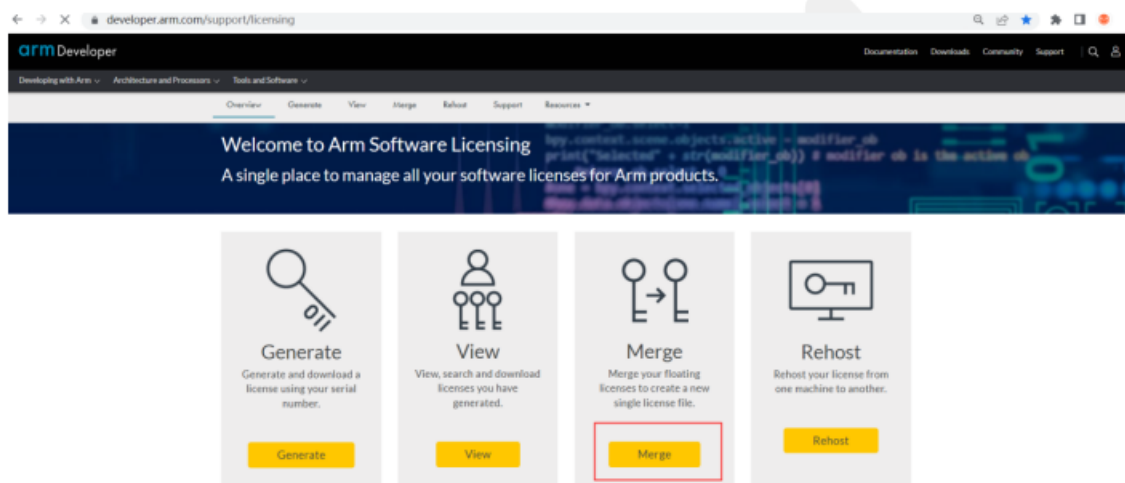
- 6) 点击 Download License 按钮下载 license 文件，如果没有出现下载窗口请设置浏览器允许弹窗。

二、 license合并

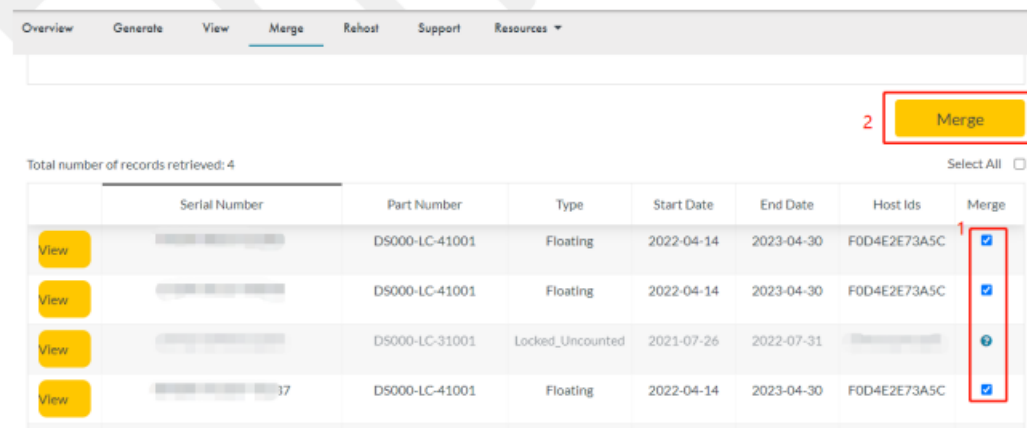
注意：如果您购买的 arm 软件有多个序列号且都需要部署在同一台服务器上就需要合并这些 license 文件。如果不是这种情况请忽略 "

License 合并" 这一节

- 1) 在浏览器的地址栏中输入<https://developer.arm.com/> 进入 ARM 官网, 登录 arm 账号, 进入 “<https://developer.arm.com/support/licensing>” 页面, 点击“Merge”



- 2) 勾选需要合并的 license, 注意：只有 MAC 地址一样的才能合并。点击 Merge 按钮按照向导完成合并



三、 软件安装

3.1 Windows 版安装

➤ ARM DS 软件

双击安装包，下一步下一步即可(无需另行安装插件)。

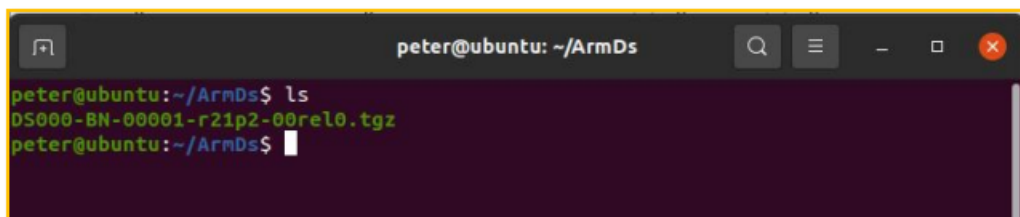
➤ 单独编译器安装

解压 ARM Compiler 文件到自定义目录，双击运行 `setup.exe` 进行 ARM Compiler 程序的安装。

3.2 Linux 版安装

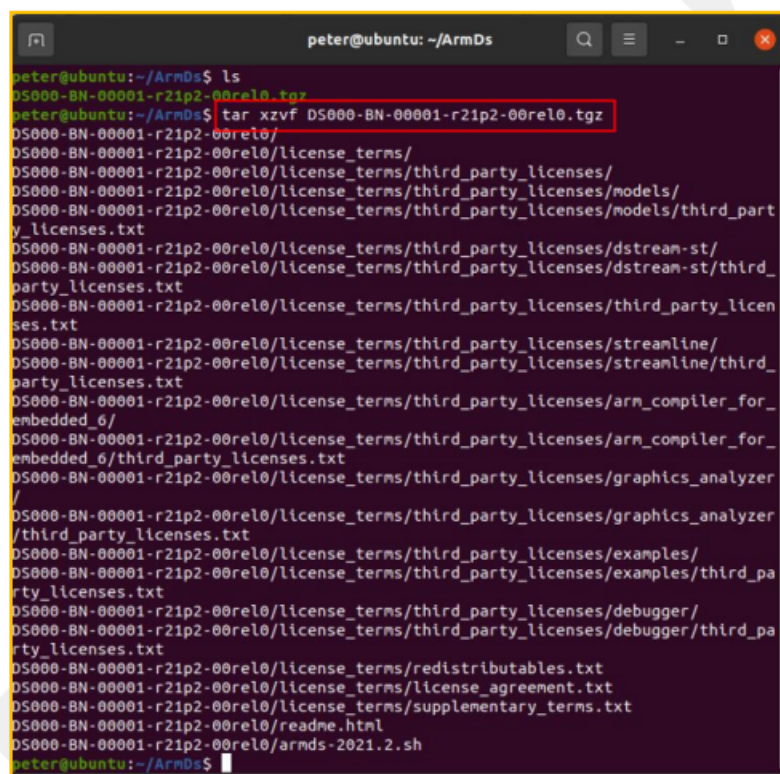
➤ ARM DS 软件

- 1) 把 ARM DS 的安装包拷贝到有读写权限的目录;



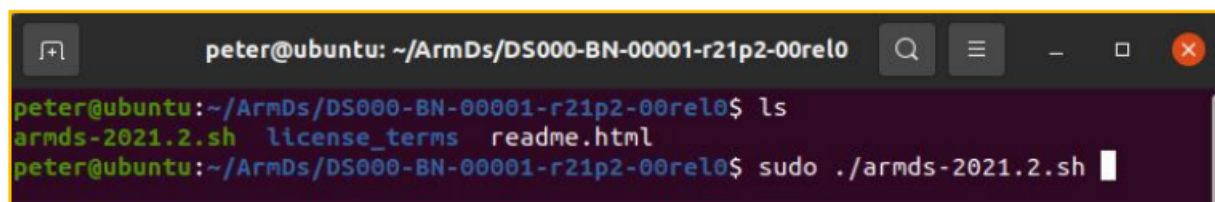
```
peter@ubuntu: ~/ArmDs
peter@ubuntu:~/ArmDs$ ls
DS000-BN-00001-r21p2-00rel0.tgz
peter@ubuntu:~/ArmDs$
```

- 2) 解压缩安装包, `tar xzvf DS000-BN-00001-r21p2-00rel0.tgz`



```
peter@ubuntu: ~/ArmDs
peter@ubuntu:~/ArmDs$ ls
DS000-BN-00001-r21p2-00rel0.tgz
peter@ubuntu:~/ArmDs$ tar xzvf DS000-BN-00001-r21p2-00rel0.tgz
DS000-BN-00001-r21p2-00rel0/
DS000-BN-00001-r21p2-00rel0/license_terms/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/models/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/models/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/dstream-st/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/dstream-st/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/streamline/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/streamline/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/arm_compiler_for_embedded_6/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/arm_compiler_for_embedded_6/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/graphics_analyzer/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/graphics_analyzer/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/examples/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/examples/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/debugger/
DS000-BN-00001-r21p2-00rel0/license_terms/third_party_licenses/debugger/third_party_licenses.txt
DS000-BN-00001-r21p2-00rel0/license_terms/redistributables.txt
DS000-BN-00001-r21p2-00rel0/license_terms/license_agreement.txt
DS000-BN-00001-r21p2-00rel0/license_terms/supplementary_terms.txt
DS000-BN-00001-r21p2-00rel0/readme.html
DS000-BN-00001-r21p2-00rel0/armds-2021.2.sh
peter@ubuntu:~/ArmDs$
```

- 3) 执行安装, `sudo ./armds-2021.2.sh`



```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0
peter@ubuntu:~/ArmDs/DS000-BN-00001-r21p2-00rel0$ ls
armds-2021.2.sh  license_terms  readme.html
peter@ubuntu:~/ArmDs/DS000-BN-00001-r21p2-00rel0$ sudo ./armds-2021.2.sh
```

```
END USER LICENCE AGREEMENT FOR THE ARM(r) DS-5 DEVELOPMENT STUDIO ("DS-5")

THIS END USER LICENCE AGREEMENT ("LICENCE") IS A LEGAL AGREEMENT BETWEEN YOU (EITHER A SINGLE INDIVIDUAL, OR SINGLE LEGAL ENTITY) AND ARM LIMITED ("ARM") FOR THE USE OF THE SOFTWARE ACCOMPANYING THIS LICENCE. ARM IS ONLY WILLING TO LICENSE THE SOFTWARE TO YOU ON CONDITION THAT YOU ACCEPT ALL OF THE TERMS IN THIS LICENCE. BY CLICKING "I AGREE" OR BY INSTALLING OR OTHERWISE USING OR COPYING THE SOFTWARE YOU INDICATE THAT YOU AGREE TO BE BOUND BY ALL OF THE TERMS OF THIS LICENCE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENCE, ARM IS UNWILLING TO LICENSE THE SOFTWARE TO YOU AND YOU MAY NOT INSTALL, USE OR COPY THE SOFTWARE, AND YOU SHOULD PROMPTLY RETURN THE SOFTWARE TO YOUR SUPPLIER AND ASK FOR A REFUND OF ANY LICENCE FEE PAID (IF ANY).

"Clang" means a compiler front end developed by http://clang.llvm.org.

"License Key" means an electronic licence key issued to you by ARM to enable the use of the Software.

"LLVM" means a code generation and optimization framework developed by http://llvm.org.

"Seat" means a written or electronic authorisation from ARM to run one copy of each of the licence managed components (as defined in the Software package) concurrently and only in the quantities stated in the sales order and delivery confirmation email, for the time period enabled by your Licence Key, and where applicable, only with the target devices specified in the sales order and/or licence file.

"Separate Files" means the separate files identified in Section 2 of the Schedule.

Please answer with one of: 'yes' or 'no/quit'
```

按“q”退出阅读 License 信息，输入“Yes”直接回车，开始安装 ARM DS。

```
Please answer with one of: 'yes' or 'no/quit'
Do you agree to the above terms and conditions? yes
```

4) 确认安装平台信息，直接回车

```
Please answer with one of: 'yes/y' or 'no/n'
Run installation platform requirement checks? [default: yes]
```

5) 选择安装 ARM DS 的路径:

```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0
*** WARNING ***
One or more of the Arm Development Studio system dependencies are not installed,
see below for a list of the missing libraries.

libz.so.1 (32-bit)
One of:
libwebkit-1.0.so.2 (64-bit)
libwebkitgtk-1.0.so.0 (64-bit)
libwebkitgtk-3.0.so.0 (64-bit)

We recommend you install the missing dependencies, otherwise the programs will not run correctly. For further information see the Arm Development Studio readme.

Where would you like to install to? [default: /opt/arm/developmentstudio-2021.2]
```

6) 是否创建安装目录，默认直接回车开始安装



```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0
libwebkit-1.0.so.2 (64-bit)
libwebkitgtk-1.0.so.0 (64-bit)
libwebkitgtk-3.0.so.0 (64-bit)

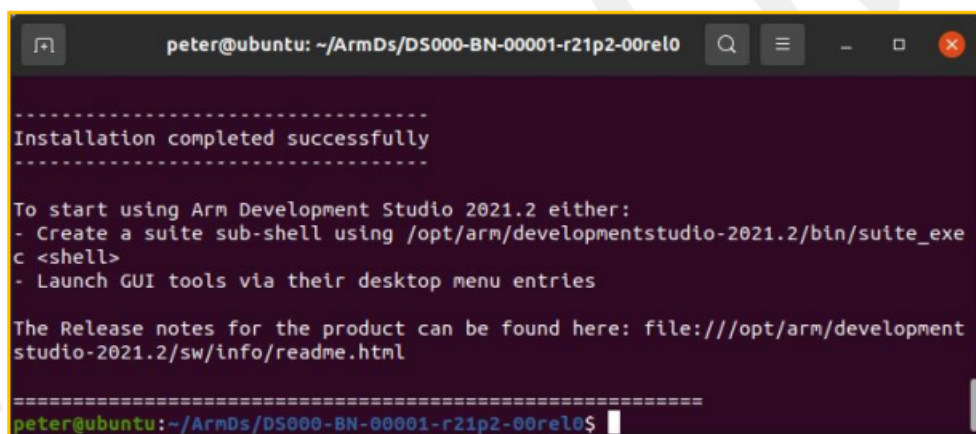
We recommend you install the missing dependencies, otherwise the programs will not run correctly. For further information see the Arm Development Studio readme.

Where would you like to install to? [default: /opt/arm/developmentstudio-2021.2]

Please answer with one of: 'yes/y' or 'no/n'
'/opt/arm/developmentstudio-2021.2' does not exist, create? [default: yes]
--- Installing to '/opt/arm/developmentstudio-2021.2' (This may take a while...)
```

根据不同电脑配置安装时间一般从几分钟到半个小时不等；

7) 安装完成



```
peter@ubuntu: ~/ArmDs/DS000-BN-00001-r21p2-00rel0

-----
Installation completed successfully
-----

To start using Arm Development Studio 2021.2 either:
- Create a suite sub-shell using /opt/arm/developmentstudio-2021.2/bin/suite_exec <shell>
- Launch GUI tools via their desktop menu entries

The Release notes for the product can be found here: file:///opt/arm/developmentstudio-2021.2/sw/info/readme.html

=====
peter@ubuntu:~/ArmDs/DS000-BN-00001-r21p2-00rel0$
```

➤ 单独编译器安装

解压并切换到 **Arm compiler** 解压的目录中，然后在命令行终端中输入 **./setup.sh** 就行

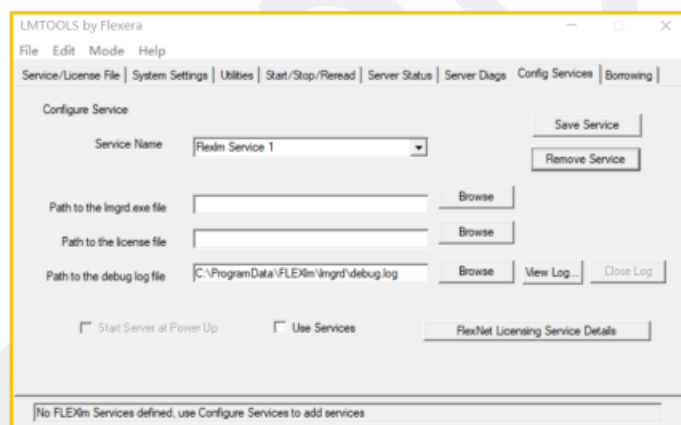
四、 服务器端管理工具安装

Windows 版

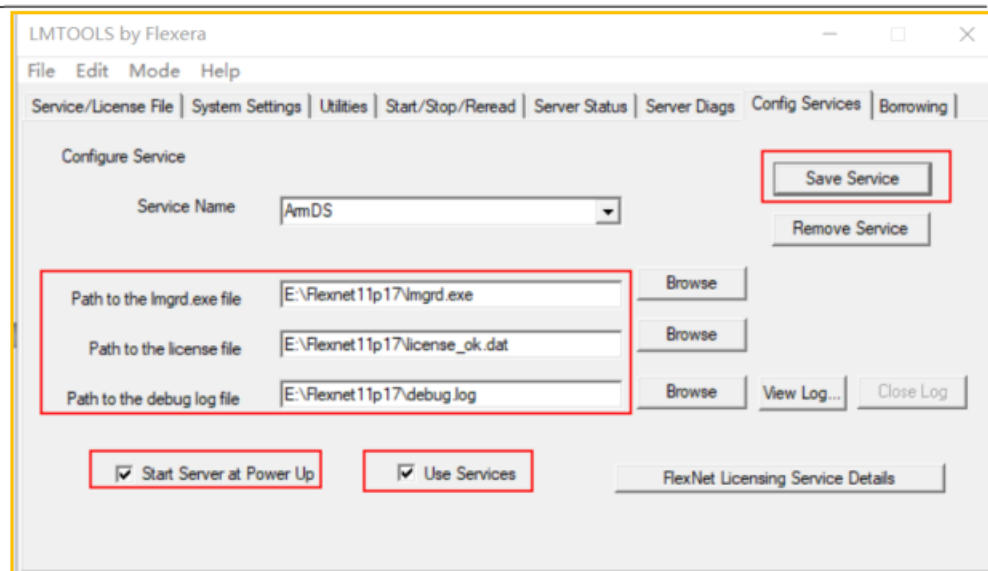
1) 我们以 X86-64 位的Flexnet工具软件为例，下载Flexnet工具后，需要解压，解压后打开文件夹。如下图：



2) 双击 Imtools.exe，出现以下对话框，点击 Config Services。



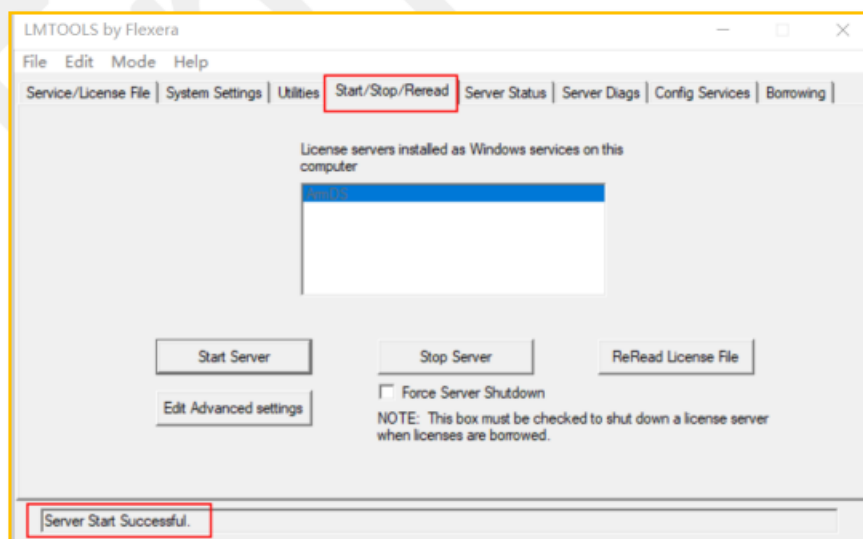
3) 点击 Config Services，开始配置服务。



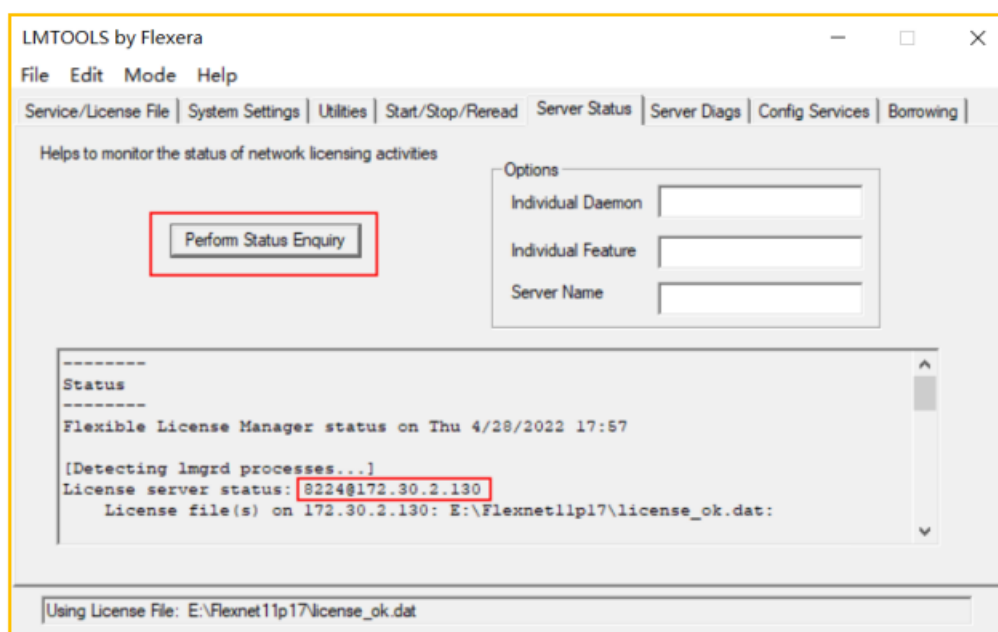
注：选填各个路径，所有的文件最好都在同一个目录下：

lmgrd.exe 在下载
的软件的文件夹里
license.dat从官方网
站下载
debug log file 需要新建。

- 4) 以上步骤完成后，点击 **save service** 保存操作。点击 **Start/Stop/Reread** 栏，点击 **Start Server**，启动服务（**Server Starts Successfully**）。



- 5) 用户可点击 **Server Status** 查看服务状态。



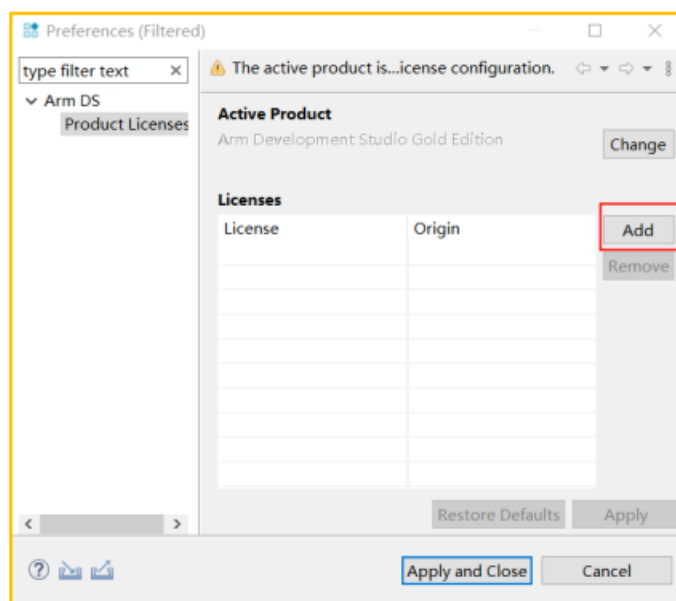
点击 Perform Status Enquiry下方会出现当前license的状态。到此，服务器端安装完成。

五、 激活软件

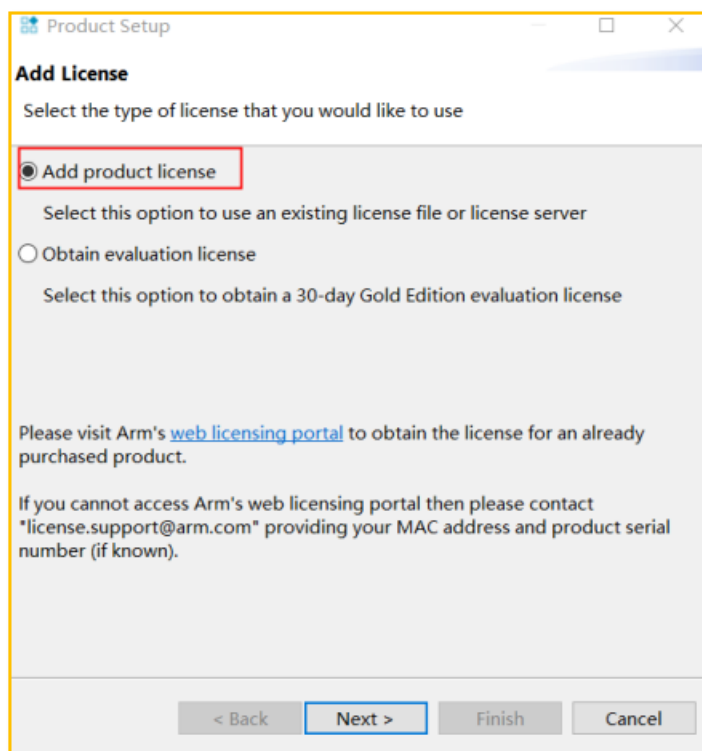
5.1 Windows 版

➤ ARM DS软件

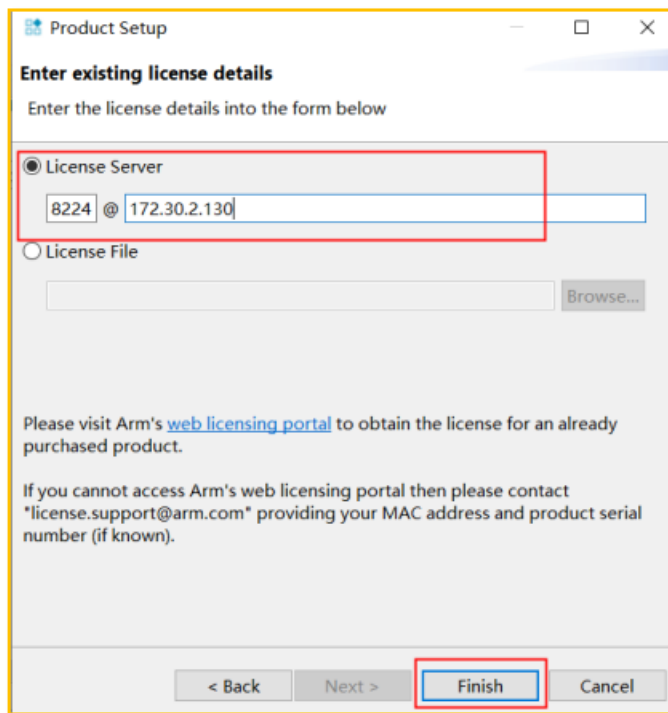
- 1) 打开Eclipse for ARM DS。
- 2) 选择“ Help> ARM License Manager”，再点击“Add ”。



- 3) 选择第一个选项，再点击“Next”。



- 4) 选择License Server，Host选择IP地址，Port是服务器启动的Port，再点击“Finish”。



➤ 单独编译器

- 1) 产品类型说明文件所在路径如下，也可以把对应的产品类型说明文件复制到 armclang 的路径下。

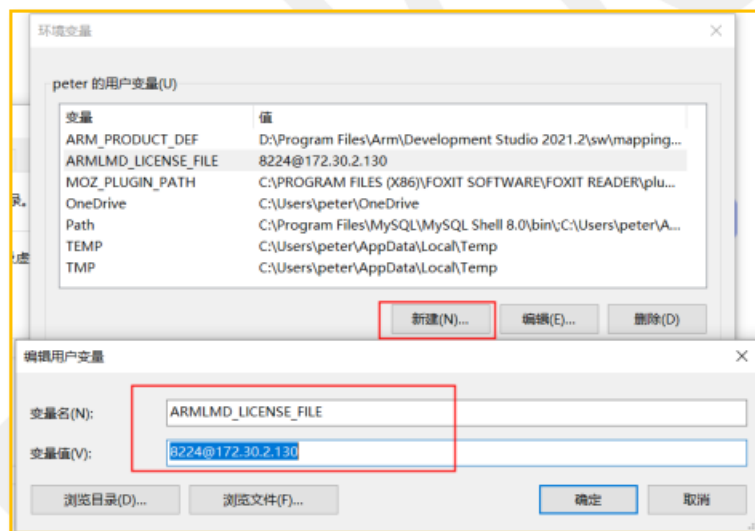
此电脑 > Program (D:) > Program Files > Arm > Development Studio 2021.2 > sw > mappings

名称	修改日期	类型	大小
bronze.elmap	2021/11/18 10:49	ELMAP 文件	35 KB
designstart_pro.elmap	2021/11/18 10:49	ELMAP 文件	37 KB
eval.elmap	2021/11/18 10:49	ELMAP 文件	18 KB
flexible_access.elmap	2021/11/18 10:49	ELMAP 文件	37 KB
gold.elmap	2021/11/18 10:49	ELMAP 文件	37 KB
intel_fpga.elmap	2021/11/18 10:49	ELMAP 文件	26 KB
intel_fpga_eval.elmap	2021/11/18 10:49	ELMAP 文件	27 KB
silver.elmap	2021/11/18 10:49	ELMAP 文件	35 KB

注意：对于 arm 原厂给的 1 年(给 FAE 的)都是 eval.elmap，正常购买的才是按照 arm ds

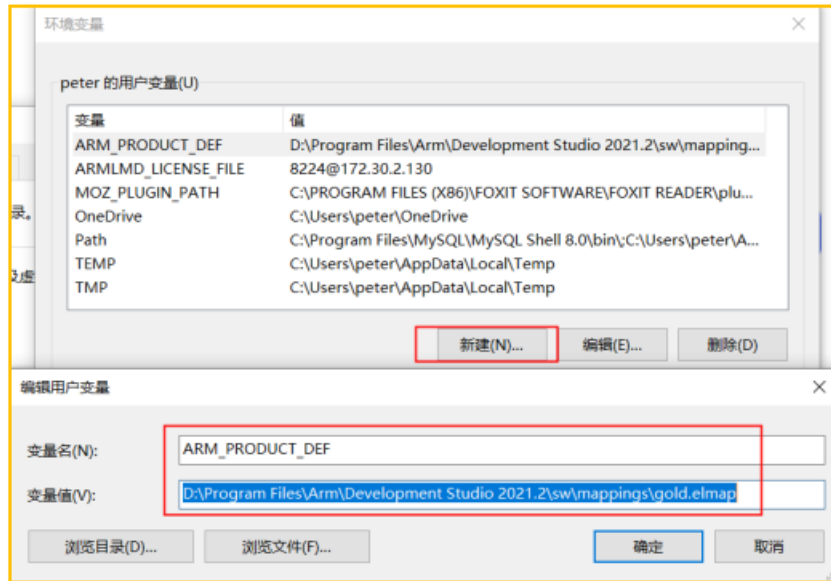
的版本类型来选择不同的文件。如果不确定用哪个，可以逐个尝试。

- 2) 设置 License 环境变量 ARMLMD_LICENSE_FILE
ARMLMD_LICENSE_FILE=license 文件路径，如：



3) 创建 ARM_PRODUCT_DEF 环境变量

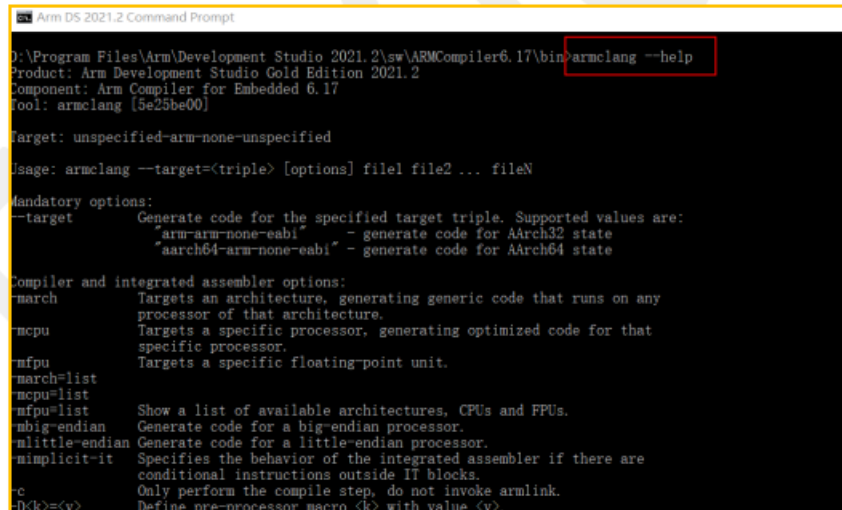
ARM_PRODUCT_DEF=elmap 文件的绝对路径设置步骤和 ARMLMD_LICENSE_FILE 类似



4) 验证:

在 cmd 窗口中切换到 arm 编译器路径, 输入编译器名称并回车。比如: armclang

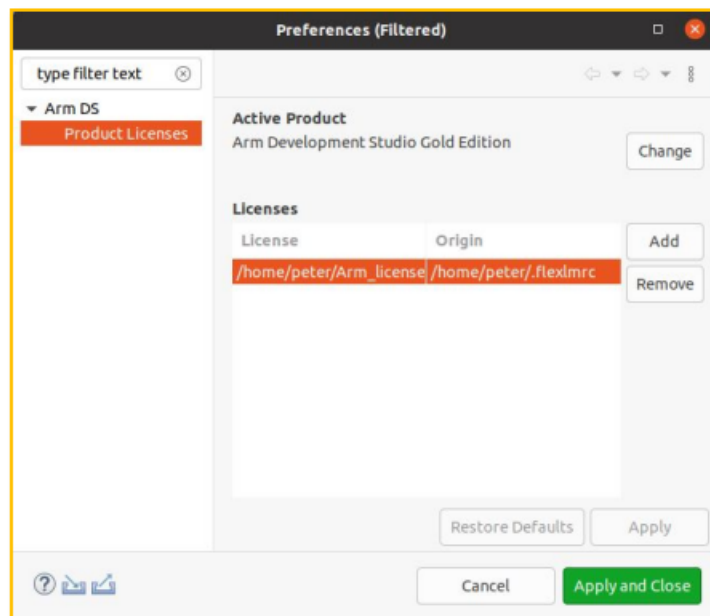
--help



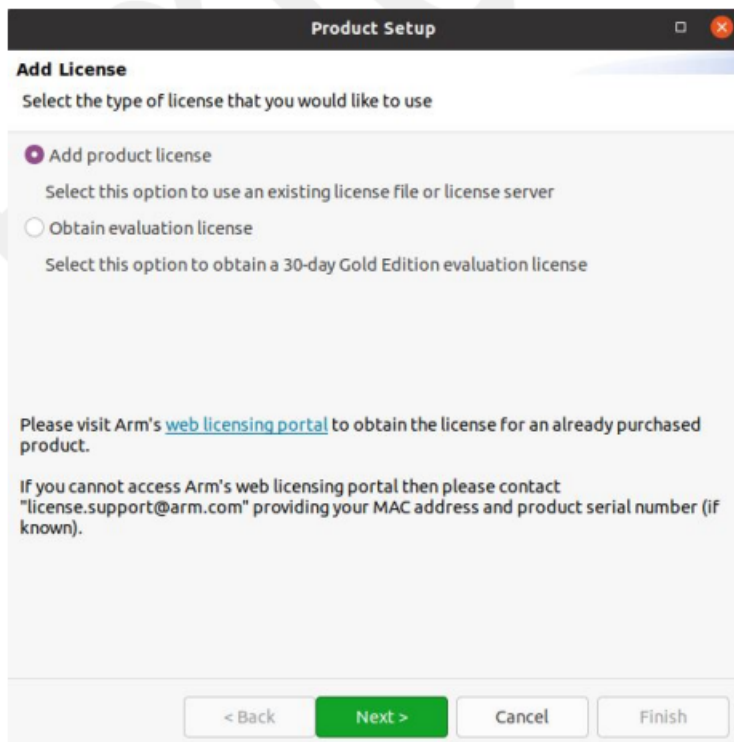
5.2 Linux 版

➤ ARM DS 软件

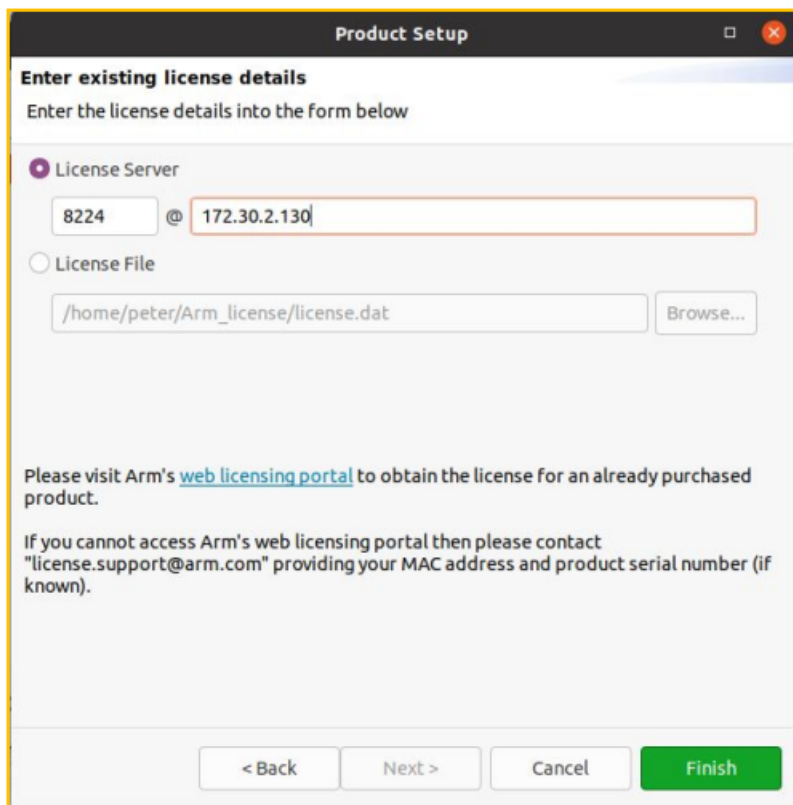
- 1) 打开Eclipse for ARM DS。
- 2) 选择“ Help> ARM License Manager”，再点击“Add”。



- 3) 选择第一个选项，再点击“Next”。



4) 选择License Server ，Host选择IP地址，Port是服务器启动的Port。



The image shows a 'Product Setup' dialog box with the title 'Enter existing license details'. It contains two radio buttons: 'License Server' (selected) and 'License File'. Under 'License Server', there are input fields for '8224' and '172.30.2.130' separated by an '@' symbol. Under 'License File', there is a text field with the path '/home/peter/Arm_license/license.dat' and a 'Browse...' button. Below these fields, there is a paragraph of text: 'Please visit Arm's [web licensing portal](#) to obtain the license for an already purchased product. If you cannot access Arm's web licensing portal then please contact "license.support@arm.com" providing your MAC address and product serial number (if known).' At the bottom, there are four buttons: '< Back', 'Next >', 'Cancel', and a green 'Finish' button.

➤ 单独编译器

注：只能是Arm Development Studio Silver, Gold or Platinum版本的license才兼容以前的旧版本编译器，Arm Development Studio Evaluation or Bronze版本不支持。

- 1) 在用户目录下的“.bashrc”文件中添加“ARMLMD_LICENSE_FILE”和“ARM_PRODUCT_DEF”环境变量的相关内容，如下图所示：

```
peter@ubuntu: ~
if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
    if [ -f /usr/share/bash-completion/bash_completion ]; then
        . /usr/share/bash-completion/bash_completion
    elif [ -f /etc/bash_completion ]; then
        . /etc/bash_completion
    fi
fi
export ARMLMD_LICENSE_FILE=8224@172.30.2.130
export ARM_PRODUCT_DEF=/home/peter/developmentstudio-2021.2/sw/mappings/gold.elmap
# export PATH="$PATH:/home/peter/ARM_Compiler_5.06u7/bin"
```

ARMLMD_LICENSE_FILE=license 文件的所在服务器的端口号@IP

ARM_PRODUCT_DEF=elmap 文件的绝对路径

对于arm原厂给的1年(给FAE的)都是eval.elmap，正常购买的才是按照arm ds的版本类型来选择不同的文件。如果不确定用哪个，可以逐个尝试。

2) 验证:

在 cmd 窗口中切换到arm 编译器路径，输入编译器名称并回车。比如: armclang -help

```
peter@ubuntu: /opt/arm/developmentstudio-2021.2/sw/AR...
peter@ubuntu: /opt/arm/developmentstudio-2021.2/sw/ARMCompiler6.17/bin$ ./armclang --help
Product: Arm Development Studio Gold Edition 2021.2
Component: Arm Compiler for Embedded 6.17
Tool: armclang [5e25bf00]

Target: unspecified-arm-none-unspecified

Usage: armclang --target=<triple> [options] file1 file2 ... fileN

Mandatory options:
--target          Generate code for the specified target triple. Supported values
are:
                  "arm-arm-none-eabi"      - generate code for AArch32 state
                  "aarch64-arm-none-eabi" - generate code for AArch64 state

Compiler and integrated assembler options:
-march            Targets an architecture, generating generic code that runs on an
y
                  processor of that architecture.
-mcpu            Targets a specific processor, generating optimized code for that
specific processor
```