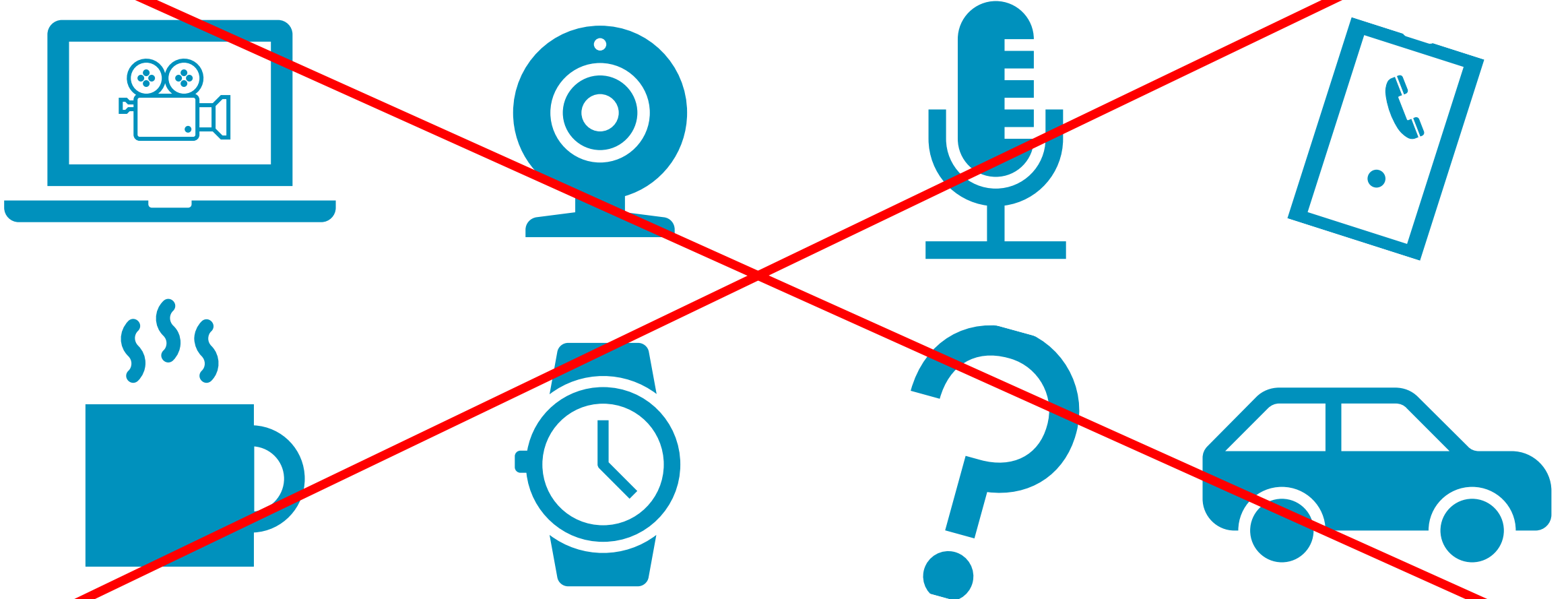




Introduction to the Arm Cortex-M Architecture and Software Development Specialization

Virtual house rules

Just a few things before we begin



What does Arm do?

Arm founded in November 1990

- Advanced RISC Machines

Best known for its range of RISC processor cores designs

- Other products – fabric IP, software tools, models, cell libraries - to help partners develop and ship Arm-based SoCs

Company headquarters in Cambridge, UK

- Processor design centers in Cambridge, Austin, Sophia Antipolis and Hsinchu
- Sales, support, and engineering offices all over the world

Arm does not manufacture silicon

More information about Arm and our offices on our web site:

- <https://www.arm.com/company>



Arm Architecture Profiles



Cortex-A

Highest performance

Designed for high-level operating systems



Cortex-R

Faster responsiveness

Designed for high performance, hard real-time applications



Cortex-M

Smallest/lowest power

Designed for discrete processing and microcontrollers



SecurCore

Tamper resistant

Designed for physical security

Development of the Arm Architecture

v4T

Halfword and signed halfword/byte support
System mode
Thumb instruction set

v5TE

Improved Arm/Thumb Interworking
CLZ
Saturated arithmetic
DSP multiply-accumulate instructions

v6

SIMD Instructions
Multi-processing
v6 Memory architecture
Unaligned data support
Extensions
Thumb-2 (v6T2)
TrustZone (v6Z)
Multicore (v6K)
Thumb-only (v6-M)

v7

Thumb-2
NEON
TrustZone
Virtualization
Architecture Profiles
v7-A (Applications):
NEON
v7-R (Real-time):
Hardware divide
v7-M (Microcontroller):
Hardware divide,
Thumb-only

v8/v9

64-bit registers
Privilege Levels
New exception model
New memory model
New instructions
Armv7 compatible

v8-M (Microcontroller)
32-bit only
Baseline / Mainline
TrustZone for Armv8-M

Note that implementations of the same architecture can be different:

- Cortex-M3 - architecture v7-M with a 3-stage integer pipeline
- Cortex-M7 - architecture v7-M with a 6-stage integer pipeline

Arm Cortex Advanced Processors

Architectural compatibility across a wide application range

Arm Cortex[®]-**A** family:

- Applications processors for feature-rich OS and 3rd party applications

Arm Cortex-**R** family:

- Embedded processors for real-time signal processing, control applications

Arm Cortex-**M** family:

- Microcontroller-oriented processors for MCU, ASSP, and SoC applications
- Arm SecurCore[™] and Cortex-M35P - Tamper-resistant security

Arm Neoverse and Cortex-X processors also fall under the **A**pplication profile

Armv7 & Armv6-M

- Cortex-A17
- Cortex-A15
- Cortex-A12
- Cortex-A9
- Cortex-A8
- Cortex-A7
- Cortex-A5

- Cortex-R8
- Cortex-R7
- Cortex-R5
- Cortex-R4

- Cortex-M7
- Cortex-M4
- Cortex-M3
- SC300
- Cortex-M1
- SC000
- Cortex-M0

- Cortex-M0+
<12k gates...

Armv8 & Armv9

- Cortex-A710
Best performance

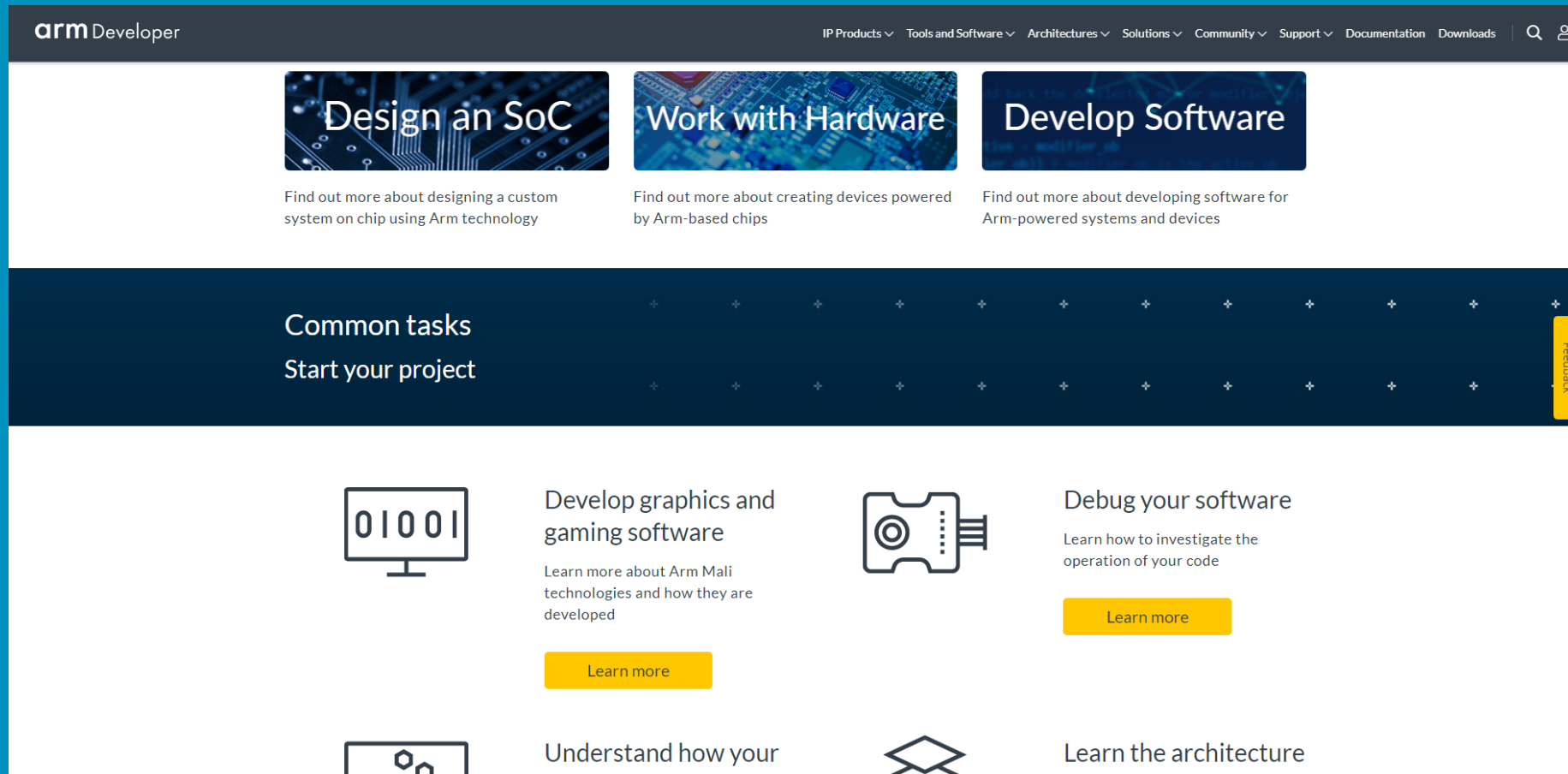
- Cortex-A510
- Cortex-A78
- Cortex-A78C
- Cortex-A78
- Cortex-A78AE
- Cortex-A77
- Cortex-A77AE
- Cortex-A76
- Cortex-A76AE
- Cortex-A75
- Cortex-A73
- Cortex-A72
- Cortex-A65AE
- Cortex-A65
- Cortex-A55
- Cortex-A53
- Cortex-A35
- Cortex-A34
- Cortex-A32

- Cortex-R82
- Cortex-R52+
- Cortex-R52

- Cortex-M55
- Cortex-M35P
- Cortex-M33
- Cortex-M23

Arm Developer

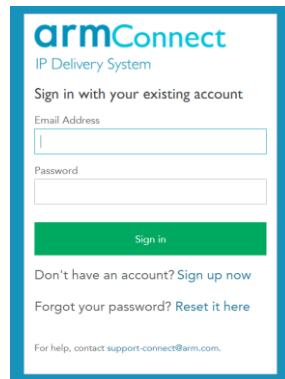
Developer resources website



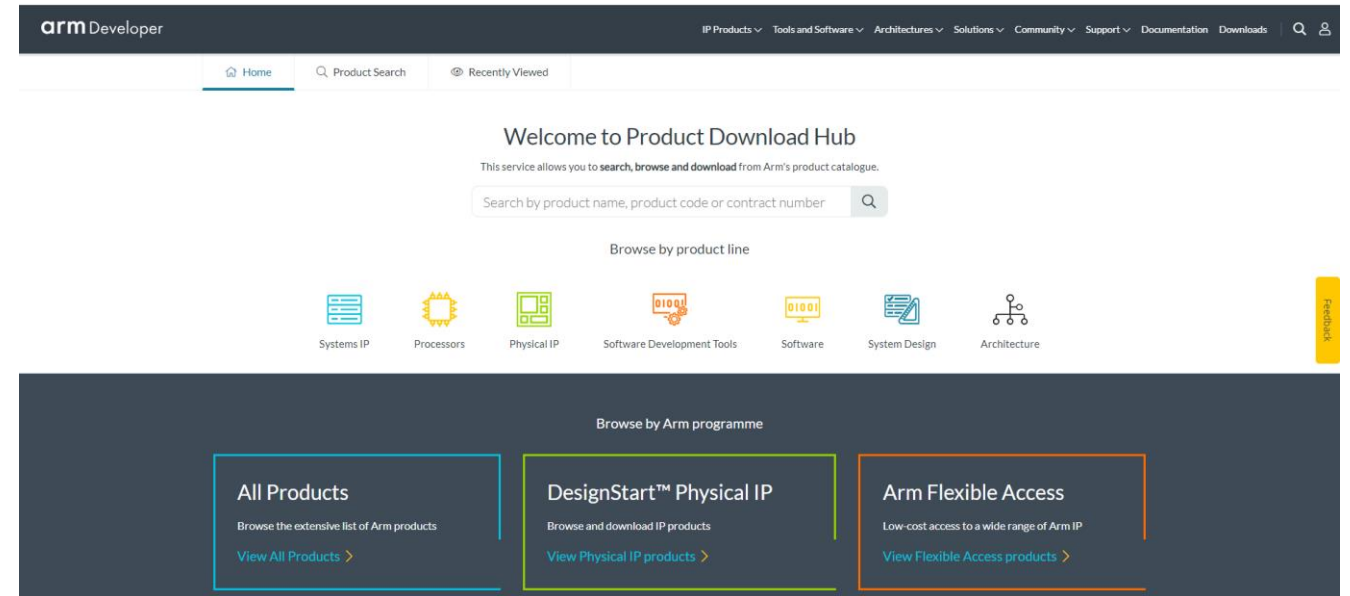
Downloading Arm products

Products bundles are currently delivered via two IP delivery systems:

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



The image shows a login form for the armConnect IP Delivery System. It has a blue header with the 'armConnect' logo and 'IP Delivery System' text. Below this, it says 'Sign in with your existing account'. There are two input fields: 'Email Address' and 'Password'. A green 'Sign in' button is below the password field. At the bottom, there are links for 'Don't have an account? Sign up now' and 'Forgot your password? Reset it here'. A small note at the very bottom says 'For help, contact support-connect@arm.com.'



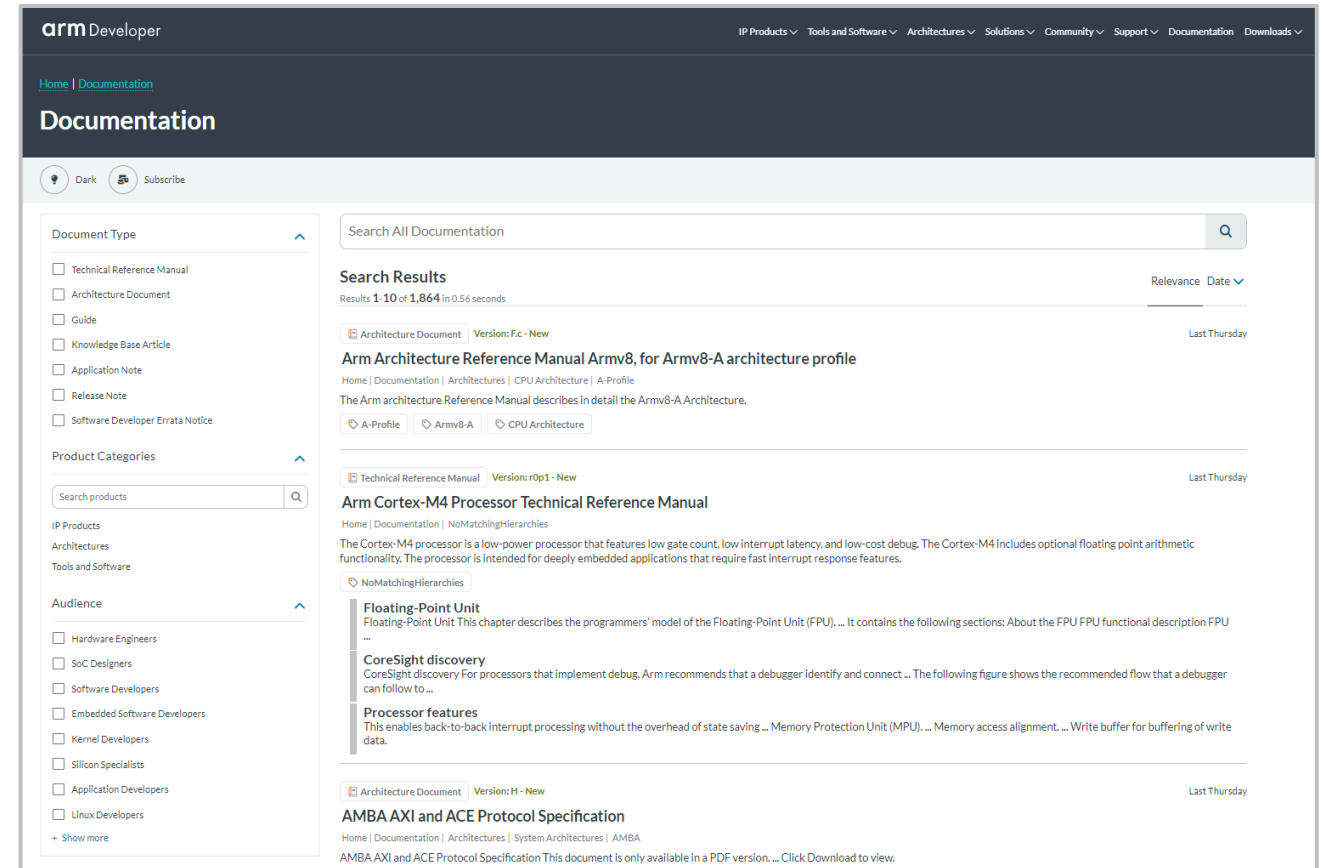
Most of Arm's IP is restricted to licensees

Non-restricted access to other downloads (license sometimes required)

Arm Documentation

Useful sections include

- Arm architecture
- Cortex-A/R/M series processors
- Arm Technical Support Knowledge Articles
- Application Notes and Tutorials
- Developer Guides and Articles

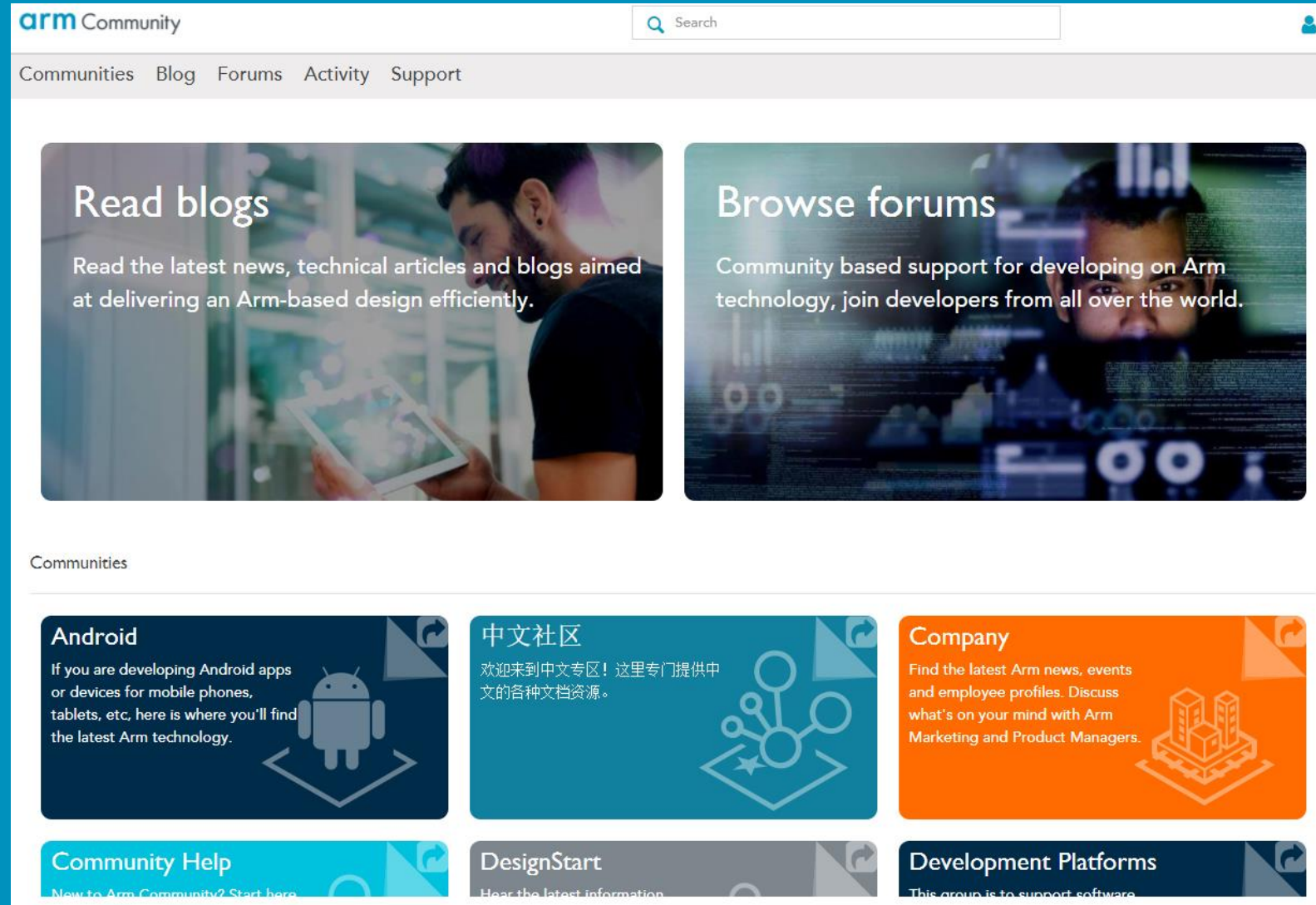


<https://developer.arm.com/documentation>

Arm Community

- Arm blogs
- Forum posts

<https://community.arm.com/developer/ip-products/processors/b/processors-ip-blog/posts/cortex-m-resources>



<http://community.arm.com/>

Technical Support

Arm Support: <https://developer.arm.com/support>



Technical Support Options

Features	Standard Support	Enhanced Support
Unlimited number of questions/cases	✓	✓
Case tracking system	✓	✓
Maintenance	✓	✓
Online documentation	✓	✓
Named AEs		✓
Proactive project engagement		✓
Onsite presence available		✓

arm

Search cases...

SEARCH

MATT RUSH...

MY CASES

Open a New Case

* What would you like to open a case about?

☐ I have a technical question about my product

☐ I have an issue with my license

☐ I have an issue downloading my product

☐ I have a website or account issue

☐ Something else

Previous

Next

Instructions

Take a look at the following pages for more information about managing your Arm support cases:

• Support Hub FAQs

• What information to provide when raising a support case

• What to expect once you have opened a case.

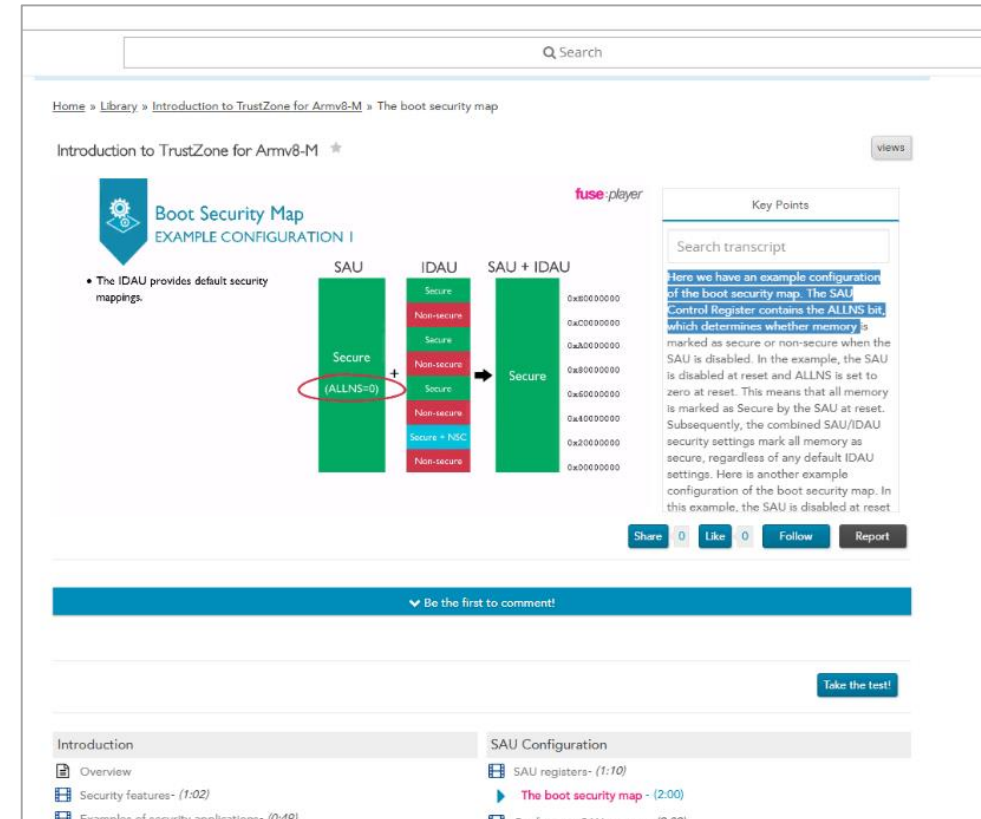
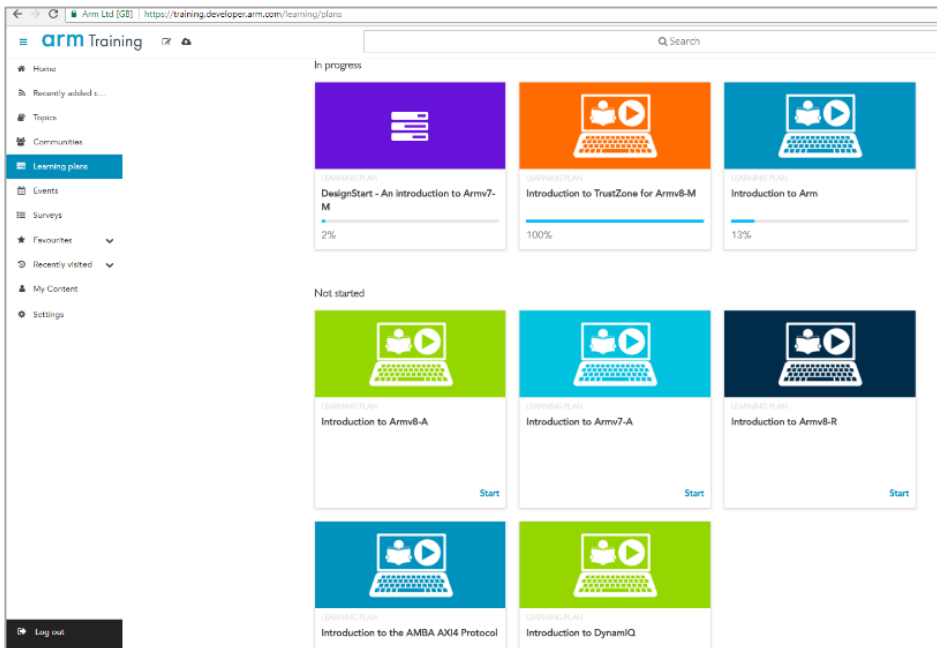
Cookie Policy | Terms of Use | Privacy Policy | Accessibility | Trademarks |

Copyright © 1995-2020 Arm Limited (or its affiliates). All rights reserved.

Arm Training

Arm offers different types of training: <https://developer.arm.com/training>

- Face-to-face training
- Virtual training
- Online learning: <https://developer.training.arm.com>



Fundamental course objectives

At the end of this course you will be able to:

- Describe the implementation differences and configuration options of the different Cortex-M processors.
- Describe the programming model (register set, modes and states of the processor).
- Describe standards and APIs that support the architecture.
- Interpret and program in assembly language.
- Target applications at the default system memory.
- Configure the MPU to change the default system memory map.
- Handle processor exceptions such as interrupts and faults.
- Build an application targeting a Cortex-M system.
- Debug an application running on a Cortex-M system.

Further course objectives

At the end of this course you will be able to:

- Develop software that can safely access memory which is shared between multiple threads or processors.
- Utilise and manage caches in software.
- Optimise software to make use of the optional DSP Extension.
- Describe how floating-point hardware can help speed-up software that contains floating-point operations.
- Partition memory to allow secure and non-secure applications to run on the same device.
- Program the SysTick timer for different use cases.
- Authenticate pointers and identify branch targets to make software more secure.
- Optimise software to make use of the optional Microcontroller Vector Extension (Helium).

arm

Thank You

Danke

Gracias

Grazie

谢谢

ありがとう

Asante

Merci

감사합니다

धन्यवाद

Kiitos

شكراً

ধন্যবাদ

תודה