Software Development Engineers
Automation Tools & Flows

Arm Sophia Antipolis
Le Paros – 25 allée Pierre Ziller
Sophia-Antipolis
06560 VALBONNE
Tel: 04.97.23.51.00

[2018-F-1] Generic Packaging Flow
[2018-F-2] Automatic RAM integration &Verification

[2018-A-1] Get lucky: Use machine learning to make random based verification more efficient.
[2018-A-2] Next generation cross-simulator RTL build flow.

Essential skills:
Different sets of skills and knowledge are required depending on the selected internship topic. These should desirably include:

1- Programming language (C/C++, Python, …)
2- Use of UNIX and shell programming
3- Strong analytical skills
4- Ability to express ideas and communicate effectively

Interested in the latest processor technologies? Join us at:http://www.arm.com/about/careers/students/student-vacancies.php

Tools & Software Solutions
goo.gl/9kcWNA

[2018-F-1] Generic Packaging Flow

At different phases of CPU development, it is essential to be able to create packages containing the IPs (Intellectual Properties) that we will deliver to our clients. These packages are tested to validate their content and make sure our clients build their product successfully. During this internship, the candidate will explore the current flow, define the requirements and needs for a new solution and implement it.

[2018-F-2] Automatic RAM integration &Verification

Internal RAM accesses are critical for CPU performance, from an area, power budget and performance perspective.
Therefore, it is important to use the most optimized ram structure that will address all these challenges altogether.
Finally, it is essential to adapt to the various kinds of RAM that would be available from a given foundry, with a given technology Node.

Electronic Design Automation
goo.gl/Lds3db

[2018-A-1] Get lucky: Use machine learning to make random based verification more efficient

RTL Verification is key in CPU development to find bugs as soon as possible in the process.
Today this is mainly randomly driven since the space is wide. Let's direct this randomness by integrating Machine Learning techniques to better drive job selection.
After getting familiar with the flow and existing algorithms, you'll be in charge of creating a solution that allows this new capability in the flow.
You'll have then to benchmark your solutions against current random solution for different applications such as reaching coverage targets faster or discover RTL bugs sooner.

[2018-A-2] Next generation cross-simulator RTL build flow

Before simulating a design you have to compile it, this step is a must, but this has a cost in terms of run time.
It's key to provide an efficient engine to achieve this compilation for the different simulators.
After getting familiar with the flow, and all constraints, you'll be in charge of defining this new flow and find the best framework to adapt to global Arm verification flow.
Then you'll have to benchmark its efficiency compared to current solution.