# Assignment Embedded Computer Architectures 1

The assignment for the course Embedded Computer Architectures 1 consists of 2 parts. The first part consists of creating the firmware for data acquisition in a physical embedded system. The system consists of a microprocessor board, an actuator, and a sensor. The collected data will then be processed in the second part, where the computation needs to be mapped to and optimized on a different platform.

## PART 2: Data Processing

Within Part 2, you need to map and optimize the functionality of the provided source code on a platform of your choice. Each student group can decide which alternative platform to use, choosing from the following options: an FPGA platform, a GPU, or a multi-core GPP (using multi-threading and/or vector extensions). If you want to target another platform not listed here, you can discuss this with the teacher.

The provided source code implements a convolutional kernel on a 2D matrix, with the following details:

- The kernel size is 3x3
- The matrix size is 80x300[1]
- The section of interest is marked between /*****/ comments and 2 timing function calls.
- The section of interest is repeated 1000 times to obtain a more accurate execution time measurement.

**Implementation steps**

1. Modify the provided source code to load the distance_vector data from a file.
2. Run the modified source code of step 1 on your own computer and measure two performance metrics: 1) average execution time for one complete convolution operation and 2) throughput (measured as the total number of 80x300 matrices that can be processed in 1 second). Do not use any compiler optimization flags.
3. Implement the functionality of the provided source code on the target platform of your choice with the aim to improve performance (define which metric you are optimizing for).
   Specific considerations per target platform:
   a) If you are targeting an FPGA and use an HDL for the design description, you can assume that the input matrix data is stored in on-chip memory. This on-chip memory is not part of your design under test (your hardware module). You only need to map the section of interest on reconfigurable hardware.
   b) If you are targeting an FPGA and use HLS for the design description, you can use the modified source code as the testbench and perform

[1] If you want to use the data you collected, the number of positions might change, so set the sizes as input parameters accordingly.

C/RTL co-simulation. You only need to map the section of interest on reconfigurable hardware.

c) If you are targeting a GPP or GPU, do not use any compiler optimization flags.

4. Quantify the performance improvements achieved in step 3 by comparing them with the modified source code of step 2 using the two performance metrics.

**Value-specific optimizations are not allowed** (i.e., your implementation should perform computations even if all values are zero).

The results will have to be presented in week 8. Furthermore, Part 2 has to result in a report that includes (at least) the following:

a) a short explanation to motivate the choice of platform
b) the key specifications of the chosen platform
c) the designed hardware architecture and implementation details (for FPGA targets), the data layout and description of the GPU kernel and mapping (for GPU targets), details about parallelism, task granularity, and synchronization (for multi-threaded implementations on GPP), and details about the vectorized implementation (for GPP implementation using vector extensions).
d) performance comparison with the modified source code executed on your personal computer and analysis of the results.
e) an overall discussion and observations about the chosen platform (comment on its performance potential and possible limitations for the problem at hand and challenges that you encountered in this project)

This report has to be handed in in week 10.

## Requirements for the presentations

- 5-minute presentation only addressing key aspects of your optimizations
- The slides have to be sent to the lecturers in PDF format before the presentation.

## Requirements for the reports

- The report should be written In English
- It should contain a table of contents
- Pages should be numbered.
- PDF format
- 11 point font
- Part 2: maximum 5 pages

## Grading the assignment

The final mark for the assignment will be: 0.3 * grade for part 1 + 0.7 * grade for part 2