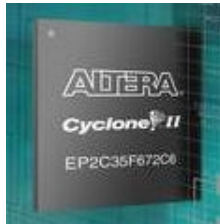
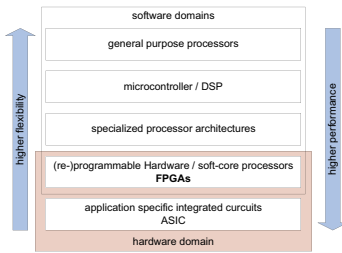


Hardware-Software Co-Design in Practise: A Case Study in Image Processing

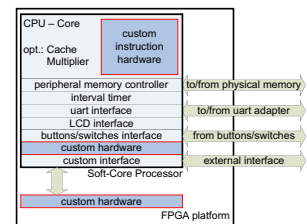


```

getCurrentPixelByte(row, column)
getCurrentPixelStatusBit(index_x, index_y)
for all neighbours
    calculateNeighbourIndex()
    getPixelByte(row, column)
    getNeighbourPixelStatusBit(index_x, index_y)
    storeStatusOfNeighbour()
end
evaluateStatusOfAllNeighbours()
decideForCurrentPixel()
storeResult()
    
```

Task

- modular approach
- different skill levels
- arbitrary order of execution
- students choose according to their demands
- enforce and support self-study



Problem Description

- complete tool chain from one vendor
- tutorials provided for easy entry
- reuse in all exercises
- graphical user interfaces
- focus on content rather than on the tool

```

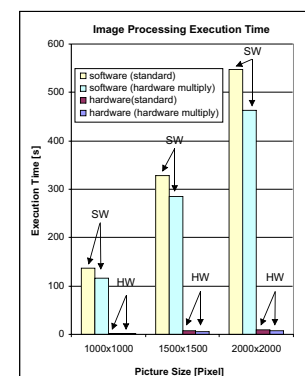
getCurrentPixelByte(row, column)
calculateNeighbourIndex()
getNeighbourPixelBytes(row, column)
callHardwareForCurrentPixel(NeighbourBytes)
storeResult()
    
```

Results

- low-cost boards with peripheral components
- resources allow for advanced topics
- provide "real-world" experiences to students
- e.g., Altera UP3 Education Kit
- other vendors available

Solution

- e.g., Nios II CPU from Altera
- 32bit RISC Processor
- configurable system
- configuration via GUI
- configuration of memory, cache, I/O ...
- support for custom hardware
- HW/SW co-design supported by custom instructions



UNIVERSITÄT ROSTOCK

University of Rostock

Ralf Joost, Ralf Salomon
 Department of Computer Science and Electrical Engineering
 Institute of Applied Microelectronics and Computer Engineering

