

Realisierung eines *Spiking Neural Network* in Chisel

Spiking Neural Networks (SNNs) sind eine alternative zu den »klassischen« neuronalen Netzen, welche derzeit hauptsächlich im Bereich der künstlichen Intelligenz werden.

Im Hardwareentwurf wird die Funktionalität von komplexen System häufig mittels einer Hardware-Beschreibungssprache (engl. Hardware Description Language, HDL) beschrieben. Die moderne HDL Chisel setzt auf die Programmiersprache Scala auf.

Im Rahmen dieser Arbeit soll ein Beschleuniger für ein SNN in der HDL Chisel implementiert und auf einem FPGA prototypisch getestet werden.

Die Arbeit eignet sich für Bachelor (Fokus: Implementierung) und Master (Fokus: offene Forschungsfragen).

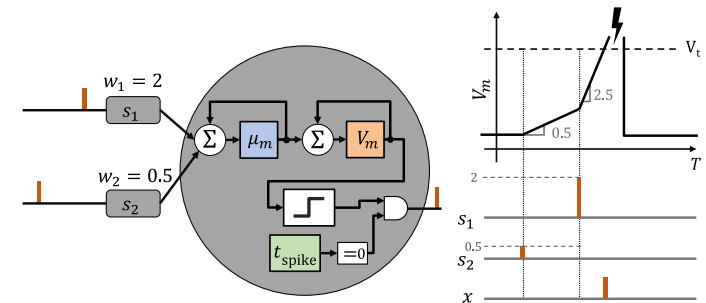
Ziele der Arbeit sind:

- Verständnis der Arbeitsweise von SNNs
- Verständnis der Hardware-Beschreibungssprache Chisel
- Implementierung eines SNN-Beschleunigers mittels Chisel

Voraussetzungen: Erfahrung FPGA Programmierung (z.B. Verilog/VHDL) hilfreich, Scala hilfreich

Art der Arbeit: variabel, z.B. Konzeption (30%), Implementierung (70%)

Betreuer: Oliver Keszöcze (oliver.keszoecze@fau.de)



```
class ClassificationUnitIO(  
  nNeurons: Int,  
  memWidth: Int)  
  extends Bundle {  
    val enable = Input(Bool())  
    val OriginAES = Input(UInt())  
    val validAddr = Input(Bool())  
    val neurons = Output(Vec(nNeurons, UInt(memWidth.W)))  
  }  
class ClassificationUnit extends Module {  
  val io = IO(ClassificationUnitIO(128, 16))  
}
```