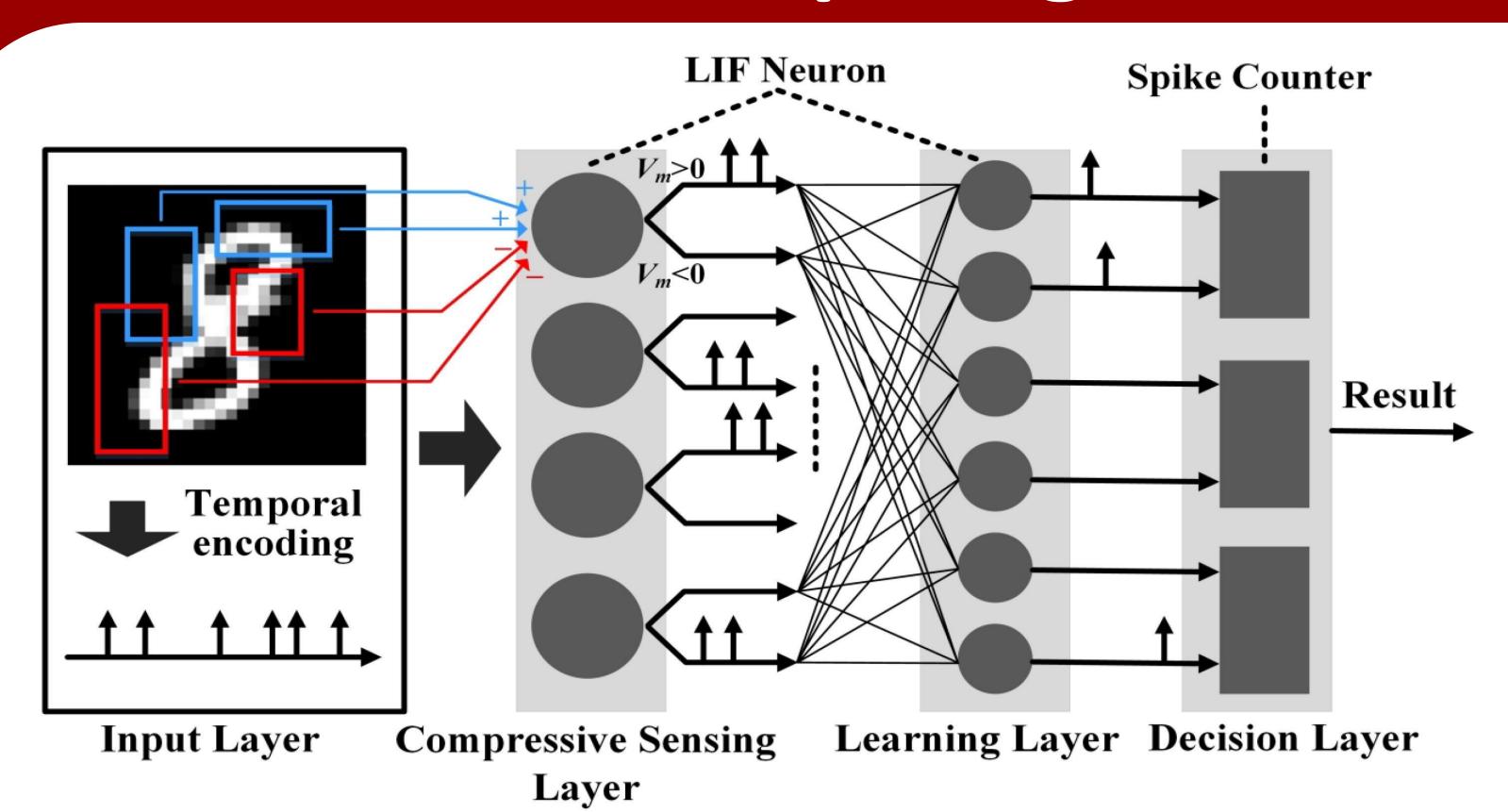


Bachelor Projects

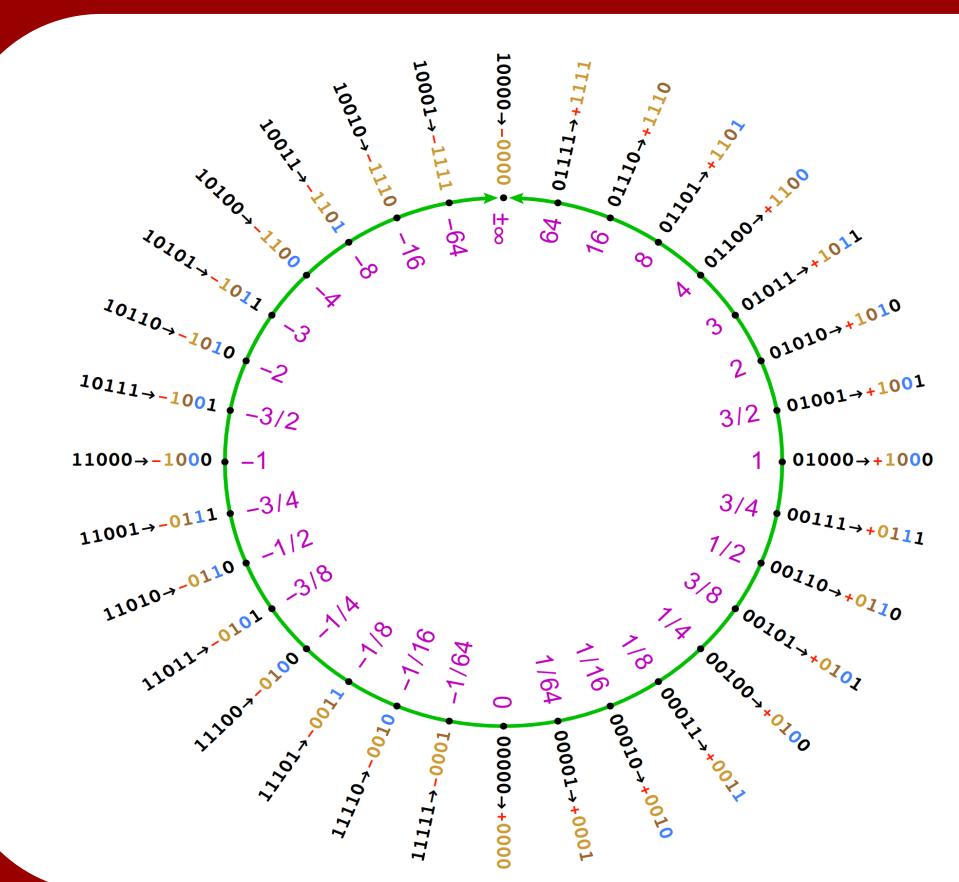
Oliver Keszöcze — olike@dtu.dk

Spiking Neural Networks

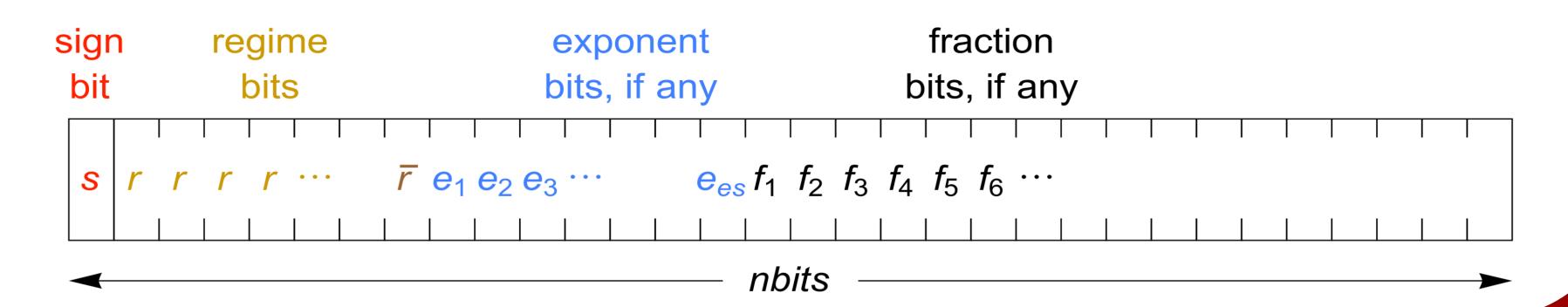


- Different take on Neural Networks
- Uses alternative number formats
- Networks with memory
- Implement and investigate!

Posits: Beating IEEE 754 floating point?



- Different encoding of floating point values
- More accurate?
- More efficient?
- Implement and find out!



Modern Hardware Description Languages

```
class Memory(prog: String, size: Int) extends Module {
       val io = IO(new Bundle {
         val rdAddr = Input(UInt(9.W))
         val rdData = Output(UInt(8.W))
         val wrEna = Input(Bool())
         val wrData = Input(UInt(8.W))
         val wrAddr = Input(UInt(9.W))
30
31
32
       val regPC = RegInit(0.U(8.W))
       val rdAddrReg = RegInit(0.U(9.W))
       rdAddrReg := io.rdAddr
35
36
       val program = VecInit(Assembler.getProgram(prog).map( .U))
37
       val instr = program(rdAddrReg(7, 0))
38
39
       /* Chisel 2 val mem = Mem(UInt(width = 8), 256, seqRead = true) */
       val mem = Mem(size, UInt(8.W))
40
       val data = mem(rdAddrReg(7, 0))
41
42
       when(io.wrEna) {
43
         mem(io.wrAddr) := io.wrData
44
45
46
       // Output MUX for now
47
       io.rdData := Mux(rdAddrReg(8), data, instr)
48
```

- Develop hardware like software!
- Follow a functional approach
- Interesting ideas welcome

```
memory :: forall dom n. (HiddenClockResetEnable dom, KnownNat n) =>
        Vec n Word ->
134
135
        Signal dom (ReadAddress, Maybe (WriteAddress, Word)) ->
        Signal dom Word
136
137
      memory instructions memCtrl = mux readFromROM programROMOutput memoryOutput
138
        where
          (addr,writeCtrl) = unbundle memCtrl
139
          programROMOutput = programROM actualAddress
140
141
          memoryOutput = lipsiRAM actualAddress writeCtrl
142
          programROM = rom instructions
          lipsiRAM = blockRamPow2 (replicate d256 0)
143
144
          readFromROM = testBit <$> addr <*> 8 -- 0 is LSB, 8 is MSB
145
          actualAddress = fmap truncateB addr :: Signal dom Word
```