Designing a calculator
FSM

https://www.theonlinecalculator.com/
Design the FSM for parsing an input string for a calculator.

What must the finite state machine remember to execute the operation $1+2$?
Try one on your own:
Draw a state diagram for

\[2 + 2 + 5\]
Let’s build a FSM that allows for several operations and data sequences
Separate the **data** from the **control** using **indirection** (self loops are implicit to minimize clutter)

Operand1, operand2, operator, and result are data

Start state and “which data has been entered” are control
Offload data onto external variables

Start

operand1 (op1)
operand2 (op2)
operator (op)
result (res)
Our generalized calculator FSM so far

What else do we still need this state machine to do?

Start
*op1

Parse Op1
*op1

Op entered
*op1

Parse Op2
*op2

Result
*res

No button
No button
No button
No button

num
num
num
num

 +/-/x/÷

operand1 (op1)
operand2 (op2)
operator (op)
result (res)
Use brackets to indicate the data stored at an address in an array (a.k.a. a register file)

<table>
<thead>
<tr>
<th>Addr</th>
<th>00</th>
<th>01</th>
<th>10</th>
<th>11</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>125</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>01</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>+</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>144</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[ R[\text{Addr}] \]
Use the FSM and system input to control the register file and ALU (the datapath)