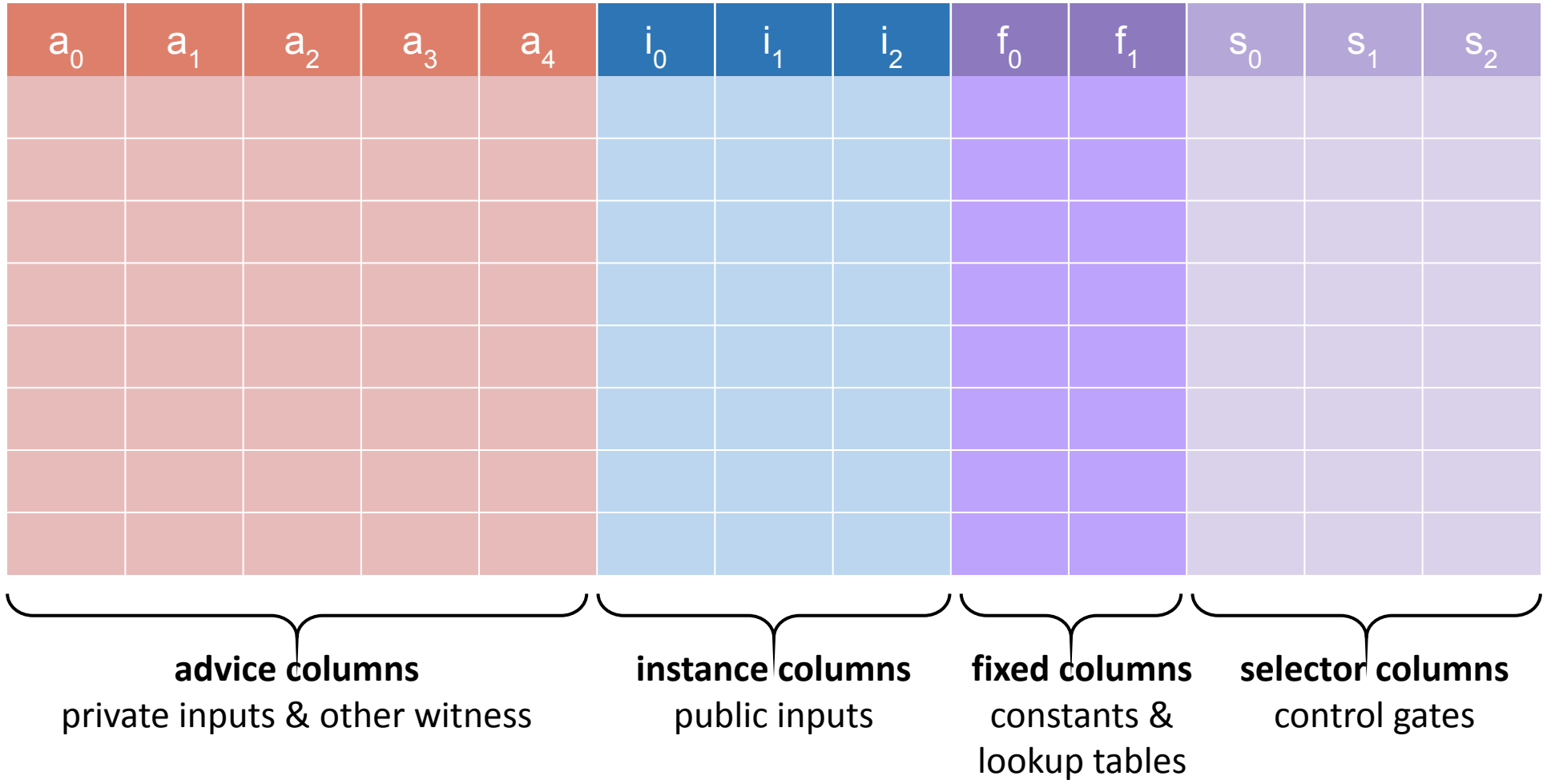
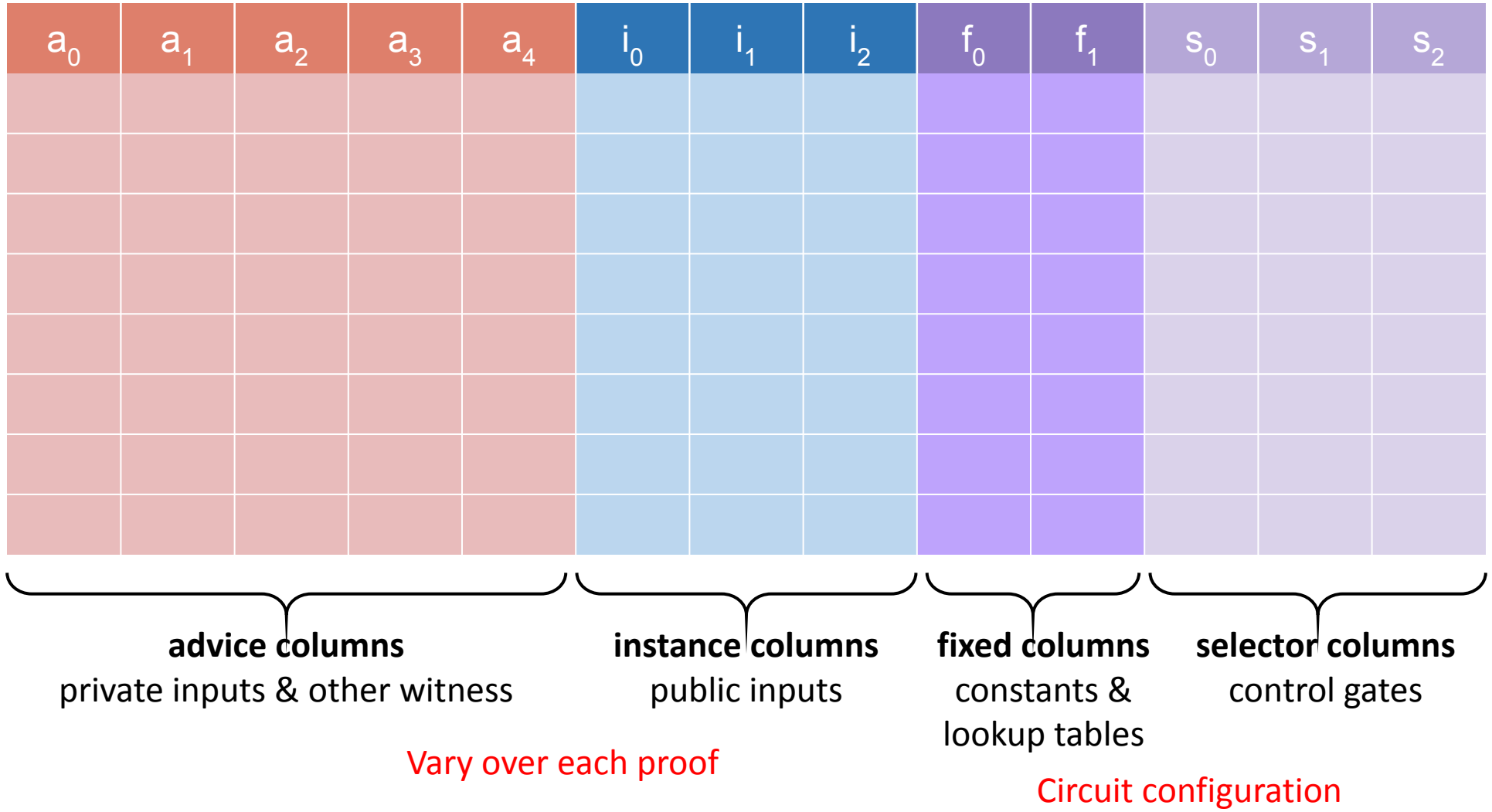
Intro to Halo2 API
Build Fibonacci Circuit together

0xPARC Halo2 Learning Group
Haichen Shen

Column types in Halo2



Column types in Halo2



Column data structure

```
/// Advice column.
Column<Advice>
/// Column for public input.
Column<Instance>
/// A fixed column for constants.
Column<Fixed>
/// A fixed column that holds binary constants.
Selector
/// A fixed column for a lookup table.
TableColumn
```

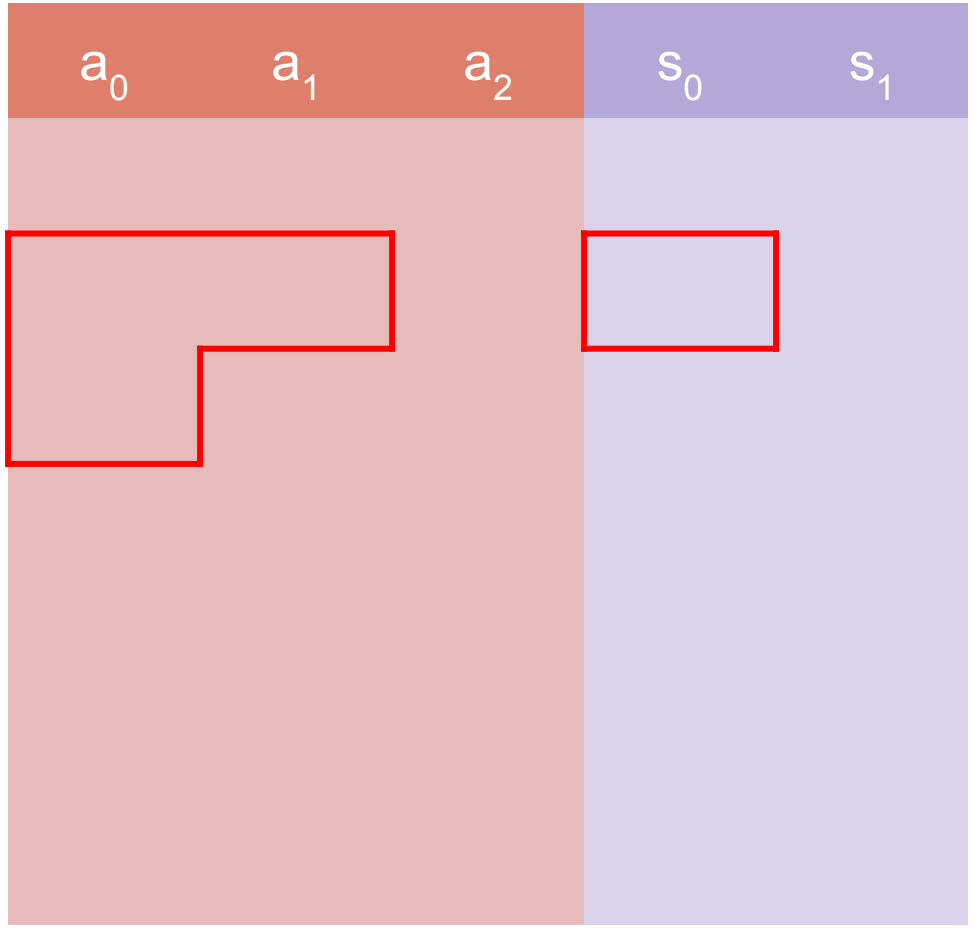
Constraint system

The constraint system is used to create columns and define custom gates.

```
fn advice_column(&mut self) -> Column<Advice>
fn instance_column(&mut self) -> Column<Instance>
fn fixed_column(&mut self) -> Column<Fixed>
fn selector(&mut self) -> Selector
fn complex_selector(&mut self) -> Selector
fn lookup_table_column(&mut self) -> TableColumn
/// Enable the ability to enforce equality over cells in this column
fn enable_equality<C: Into<Column<Any>>>(&mut self, column: C)
/// Creates a new gate.
fn create_gate<C: Into<Constraint<F>>, Iter: IntoIterator<Item = C>>(
    &mut self,
    name: &'static str,
    constraints: impl FnOnce(&mut VirtualCells<'_, F>) -> Iter,
)
```

Layouter lays out regions in the table

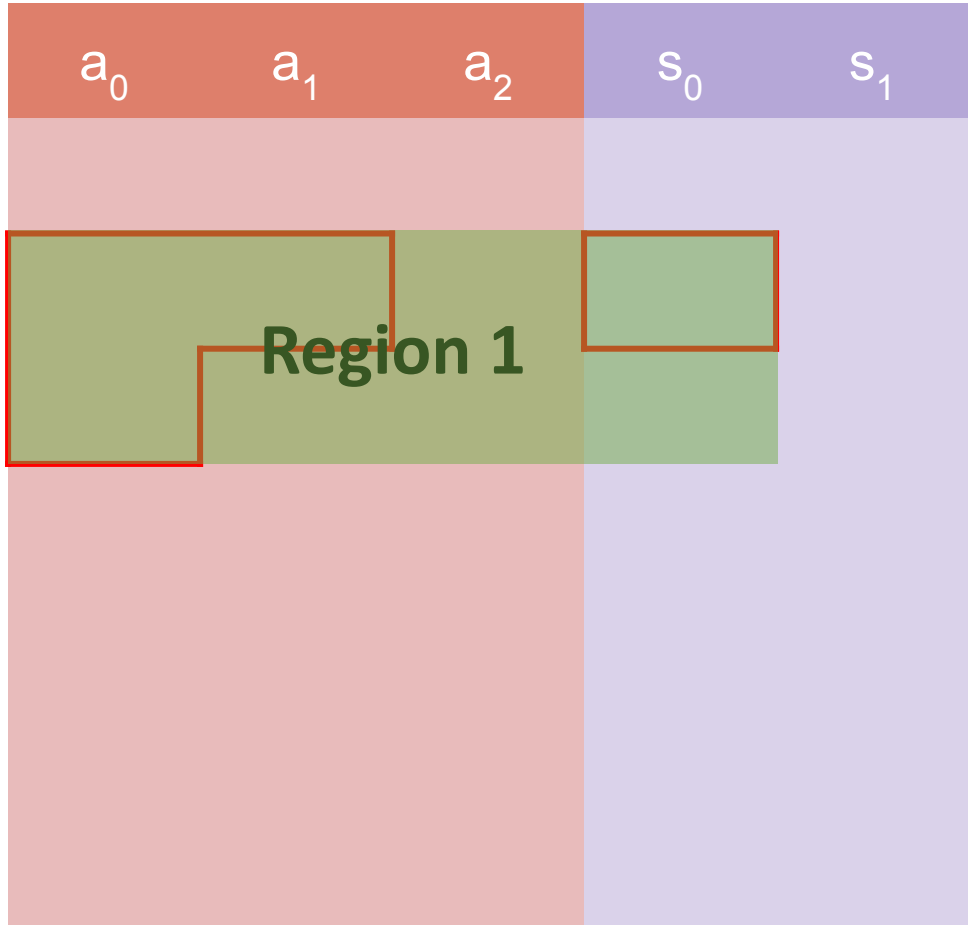
A region doesn’t need to have the same shape as custom gate but must cover all related custom gates.



Layouter lays out regions in the table

A region doesn’t need to have the same shape as custom gate but must cover all related custom gates.

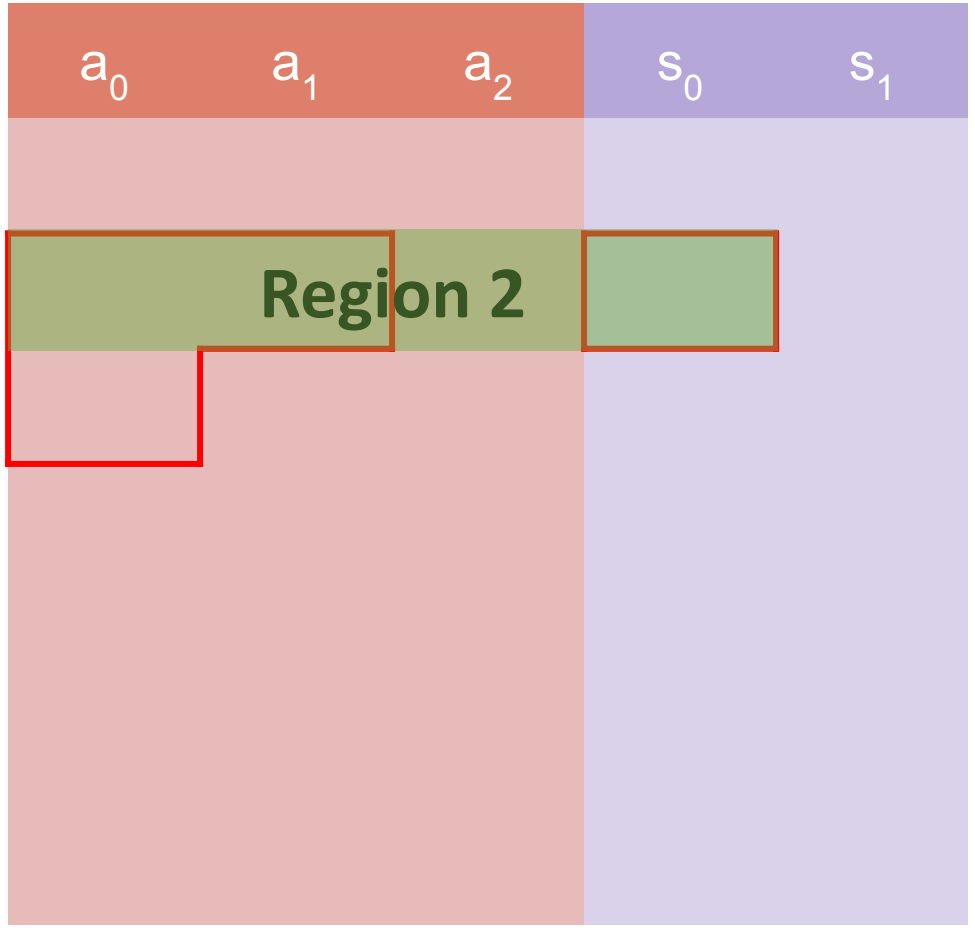
Valid region



Layouter lays out regions in the table

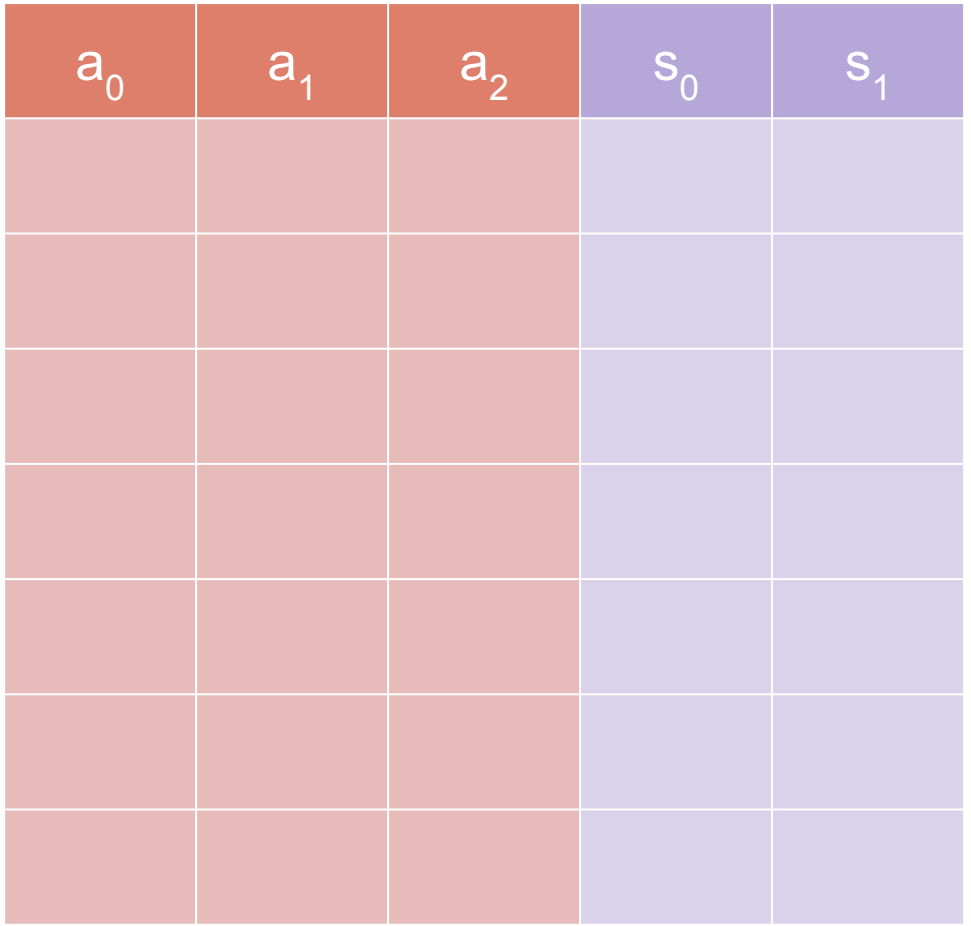
A region doesn’t need to have the same shape as custom gate but must cover all related custom gates.

Invalid region



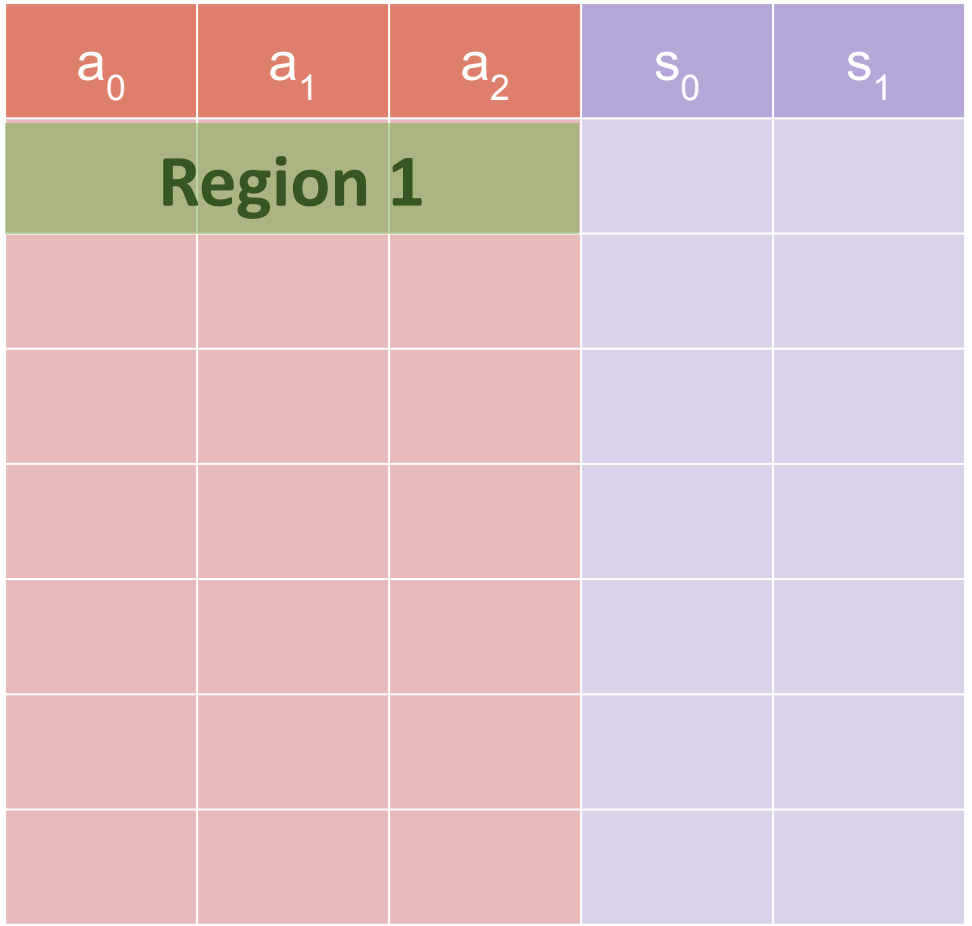
How SimpleFloorPlanner works?

- It is a single-pass layouter.
- It finds the first empty row for each column used in the region and takes a maximum.



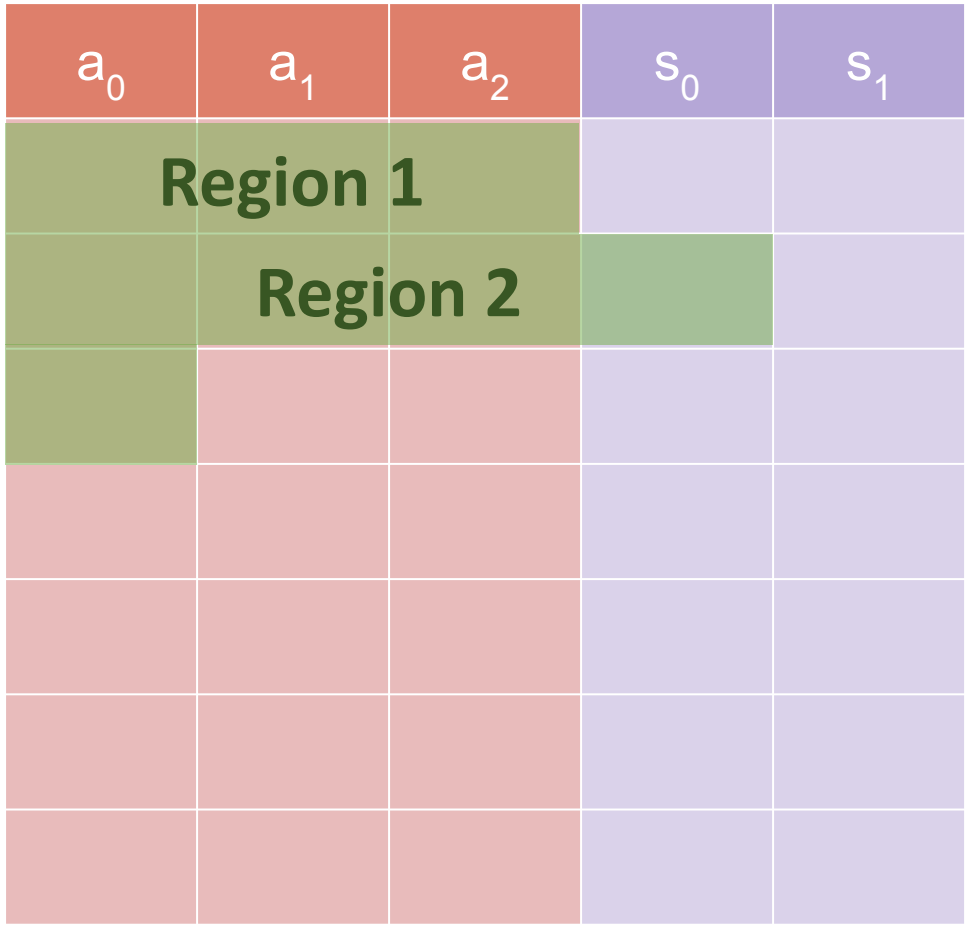
How SimpleFloorPlanner works?

- It is a single-pass layouter.
- It finds the first empty row for each column used in the region and takes a maximum.



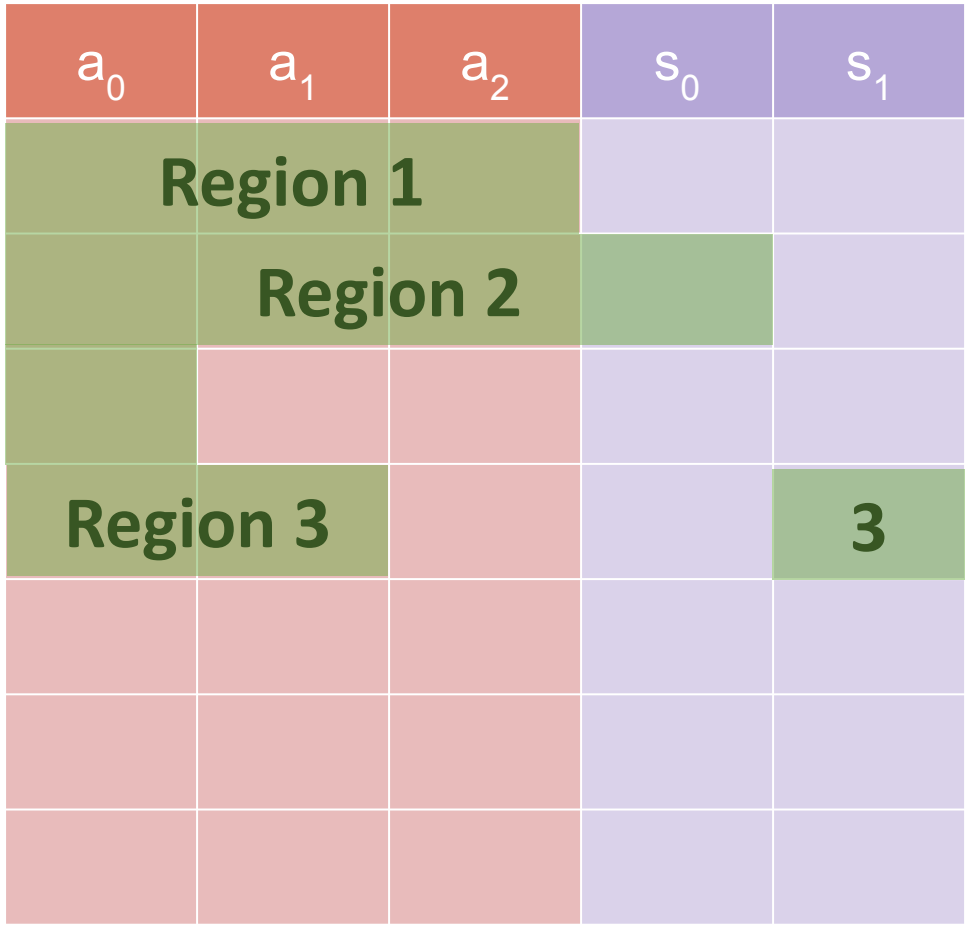
How SimpleFloorPlanner works?

- It is a single-pass layouter.
- It finds the first empty row for each column used in the region and takes a maximum.



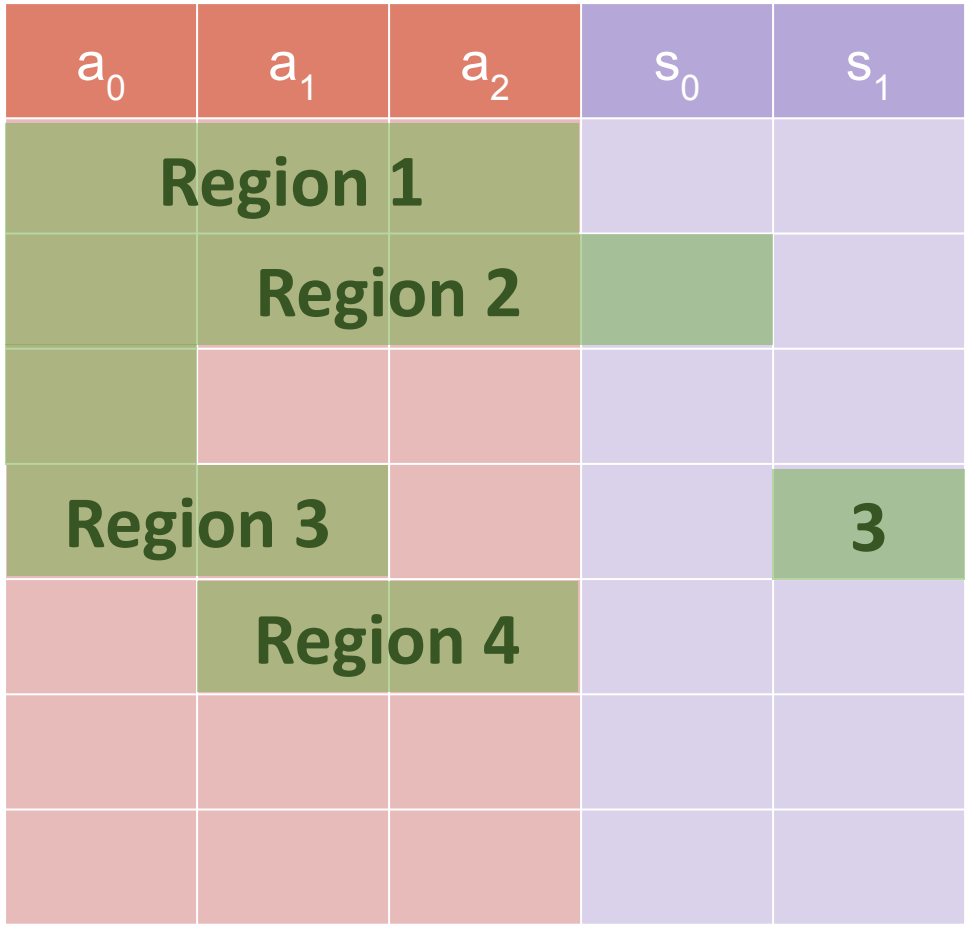
How SimpleFloorPlanner works?

- It is a single-pass layouter.
- It finds the first empty row for each column used in the region and takes a maximum.



How SimpleFloorPlanner works?

- It is a single-pass layouter.
- It finds the first empty row for each column used in the region and takes a maximum.



Three steps to implement a circuit

1. Define a `Config` struct that includes the columns used in the circuit
2. Define a `Chip` struct that configures the constraints in the circuit and provides assignment functions
3. Define a `Circuit` struct that implements the `Circuit` trait and instantiates a circuit instance that will be fed into the prover

Circuit trait

```
pub trait Circuit<F: Field> {
    type Config: Clone;
    type FloorPlanner: FloorPlanner;

    /// Returns a copy of this circuit with no witness values
    fn without_witnesses(&self) -> Self;

    /// The circuit is given an opportunity to describe the exact gate
    /// arrangement, column arrangement, etc.
    fn configure(meta: &mut ConstraintSystem<F>) -> Self::Config;

    /// Given the provided `cs`, synthesize the circuit.
    fn synthesize(&self, config: Self::Config, layouter: impl Layouter<F>) ->
Result<(), Error>;
}
```

Fibonacci Circuit in Halo2

Fibonacci circuit

Given f(0) = x, f(1) = y, we will prove f(9) = z.

Fibonacci example 1

Row	a_0	a_1	a_2	s	i
0	1	1	2	1	
1	1	2	3	1	
2	2	3	5	1	
3	3	5	8	1	
4	5	8	13	1	
…	…	…	…	…	

Fibonacci example 1

Row	a_0	a_1	a_2	s	i
0	1	1	2	1	
1	1	2	3	1	
2	2	3	5	1	
3	3	5	8	1	
4	5	8	13	1	
...	...	...	...	...	

Custom gate:

```
s * (a0 + a1 - a2) == 0
```

Fibonacci example 1

Row	a_0	a_1	a_2	s	i
0	1	1	2	1	
1	1	2	3	1	
2	2	3	5	1	
3	3	5	8	1	
4	5	8	13	1	
…	…	…	…	…	

Permutation argument

Fibonacci example 2

Row	a_0	a_1	s	i
0	1	1	1	
1	2	3	1	
2	5	8	1	
3	13	21	1	
…	…	…	…	

Fibonacci example 3

Row	a_0	s	i
0	1	1	
1	1	1	
2	2	1	
3	3	1	
4	5	1	
…	…	…	

Thank you!
Questions?