1 Upper bound for thin rectangles

Theorem 1. *The tile complexity of a self-assembled, just-barely 3D, $k \times N$ rectangle at temperature 1 is $O\left(N^{\frac{1}{k}} + k\right)$ in the abstract tile assembly model.*

The following is a proof by construction. Let $d = \left\lfloor \frac{k-4}{3} \right\rfloor$, $r = remainder(k-4, 3)$, $m = \left(\frac{N}{10}\right)^{\frac{1}{d}}$, $l = |bin\,(m-1)|$, and $c = ?NumericalValueOfTheStartCounter?$. The assembly constitutes a d-digit, base-m counter. The value of c is chosen such that the counter stops just before reaching a height of N tiles, at which point, the construction is given a flat "roof" and adds a small number of rows to finish reaching a length of N. We define a `Gadget Unit` as a collection of gadgets with a singular purpose. Gadgets belonging to the same gadget unit will have their figures grouped together, although, since `Vertical Column Tiles` are present in a majority of these gadget units, they will only be shown in Figure 1.

(a) `Up_Column_Tile`

(b) `Down_Column_Tile`

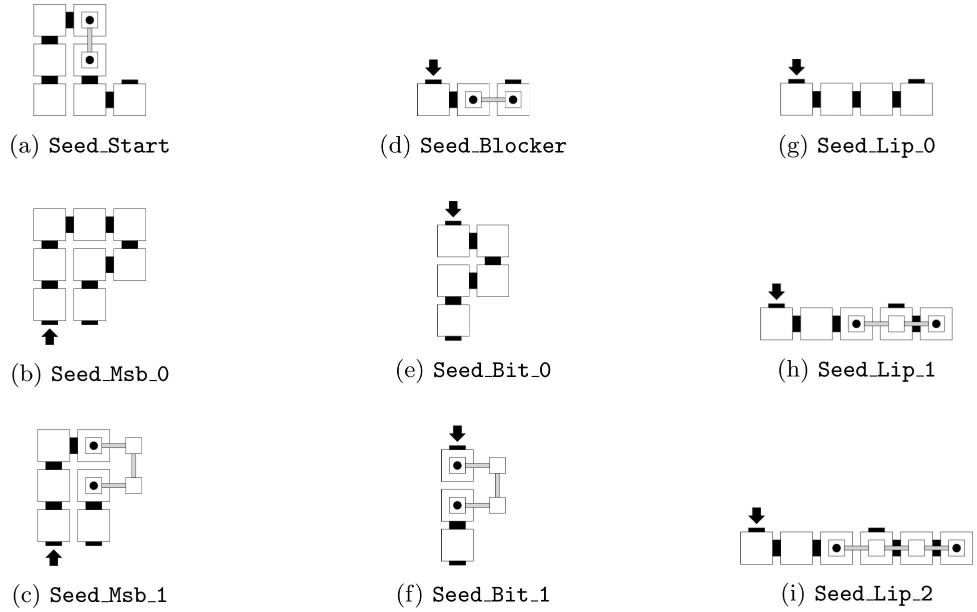
Figure 1: `Vertical Column Tiles` are used throughout the construction to adjust the height of gadget units.

1.1 Seed gadgets

We begin by hard-coding the base of the construction with the `Seed` unit. It has d columns, where each column represents a digit (most significant digit first) of c in base m, and a collection of geometric bit-bumps on the columns' east sides encodes the digits into binary. A small "lip" is added on the east side of the `Seed` gadget in cases where $k-1$ is not divisible by 3; this catches the vertical fill tiles at the end of the construction. See Figure 2.

We define the `Seed` gadgets as followed:

- Create `Seed_Start` $(\langle \texttt{seed}, \texttt{col}, d, 1\rangle)$ from the general gadget in Figure 2a.
- For each $i = 1, \ldots, d$:
  - For each $j = 1, \ldots, 3l - 3$:
    - Create `Up_Column` $(\langle \texttt{seed}, \texttt{col}, i, j\rangle, \langle \texttt{seed}, \texttt{col}, i, j+1\rangle)$ from the general gadget in Figure 1a.
  - Create `Seed_Msb` $(\langle \texttt{seed}, \texttt{col}, i, 3l-2\rangle, \langle \texttt{seed}, \texttt{bit}, i, l-1\rangle)$ from the general gadget in Figure 2b if $bit(c, m, i, l) = 0$ or Figure 2c if $bit(c, m, i, l) = 1$.
  - For each $j = 1, \ldots, l-1$:
    - Create `Seed_Bit` $(\langle \texttt{seed}, \texttt{bit}, i, j\rangle, \langle \texttt{seed}, \texttt{bit}, i, j-1\rangle)$ from the general gadget in Figure 2e if $bit(c, m, i, j) = 0$ or Figure 2f if $bit(c, m, i, j) = 1$.
- For each $i = 1, \ldots, d-1$:
  - Create `Seed_Blocker` $(\langle \texttt{seed}, \texttt{bit}, i+1, 0\rangle, \langle \texttt{seed}, \texttt{col}, i, 1\rangle)$ from the general gadget in Figure 2d.
- Create `Seed_Lip` $(\langle \texttt{seed}, \texttt{bit}, 1, 0\rangle, \langle \texttt{rw}, \texttt{up}, 1\rangle)$ from the general gadget in Figure 2g if $r = 0$, Figure 2h if $r = 1$, or Figure 2i if $r = 2$.



(a) Seed_Start (d) Seed_Blocker (g) Seed_Lip_0

(b) Seed_Msb_0 (e) Seed_Bit_0 (h) Seed_Lip_1

(c) Seed_Msb_1 (f) Seed_Bit_1 (i) Seed_Lip_2

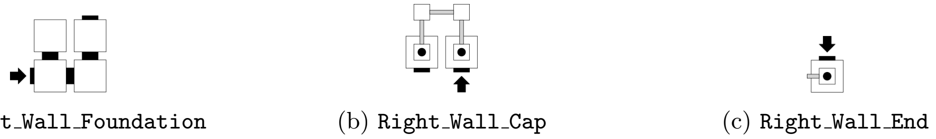
Figure 2: The Seed gadget unit.

1.2 Right wall gadgets

After the Seed unit, a Right_Wall unit attaches to the vacant north-facing glue without a Right_Wall_Foundation gadget (all other Right_Wall units will have this gadget). The Right_Wall unit's purpose is to initiate a row of Counter units, and then to block the Return gadget so that a new Right_Wall unit and subsequent counter row can form westward. Its general gadgets are shown in Figure 3.

We define the Right_Wall gadgets as followed:

- Create Right_Wall_Foundation $(\langle \texttt{rw}, \texttt{found} \rangle, \langle \texttt{rw}, \texttt{up}, 1 \rangle)$ from the general gadget in Figure 3a.
- For each $i = 1, \ldots, 3l$:
  - Create Up_Column $(\langle \texttt{rw}, \texttt{up}, i \rangle, \langle \texttt{rw}, \texttt{up}, i+1 \rangle)$ from the general gadget in Figure 1a.
- Create Right_Wall_Cap $(\langle \texttt{rw}, \texttt{up}, 3l+1 \rangle, \langle \texttt{rw}, \texttt{down}, 1 \rangle)$ from the general gadget in Figure 3b.
- For each $i = 1, \ldots, 3l-1$:
  - Create Down_Column $(\langle \texttt{rw}, \texttt{down}, i \rangle, \langle \texttt{rw}, \texttt{down}, i+1 \rangle)$ from the general gadget in Figure 1b.
- Create Right_Wall_End $(\langle \texttt{rw}, \texttt{down}, 3l \rangle, \langle \texttt{inc}, \texttt{start} \rangle)$ from the general gadget in Figure 3c.



(a) Right_Wall_Foundation (b) Right_Wall_Cap (c) Right_Wall_End

Figure 3: The Right_Wall gadget unit.

1.3 Counter gadgets

The set of Counter gadget units consist of m units that increment an individual digit of the counter and m units that copy an individual digit of the counter. Each row of the counter has d Counter units and begins by adjoining the west facing glue of the Right_Wall gadget with an incrementing unit. Each Counter unit reads over a series of bit-bumps protruding into their row from the preceding Seed unit or counter row. After reading the bit patterns via Guess gadgets, the set of possible Counter units is narrowed to one, and then a bit pattern is either copied or incremented onto the row above by the unit. To elaborate on this "narrowing", each set of copy and increment units uses the same starting Guess gadgets. Then, depending on which glue of the Counter gadget is blocked or unblocked by the preceding counter row, a gadget will be initiated but from a subset of eligible Counter units with those gadgets in common. The Counter unit that increments $m-1$ to 0 is special because its west-facing glue initiates another increment unit and its south-facing glue initiates the End_Left_Wall unit (whichever glue is not blocked). Other increment units initiate a copy unit with their west-facing glues and a regular Left_Wall unit with their south-facing glues. The copy units similarly initiate another copy unit or a Left_Wall unit. The gadgets belonging to the Counter units are shown in Figure 4.

We define the Counter gadgets as followed:

- Create Counter_Start $(\langle \texttt{inc}, \texttt{start}\rangle, \langle \texttt{inc}, \texttt{read}, 0\rangle, \langle \texttt{inc}, \texttt{read}, 1\rangle)$ from the general gadget in Figure 4a.
- Create Counter_Start $(\langle \texttt{copy}, \texttt{start}\rangle, \langle \texttt{copy}, \texttt{read}, 0\rangle, \langle \texttt{copy}, \texttt{read}, 1\rangle)$ from the general gadget in Figure 4a.
- For each $i = 0, \ldots, l-2$ and each $u \in \{0,1\}^i$:
  - Create Counter_Read $(\langle \texttt{inc}, \texttt{read}, u0\rangle, \langle \texttt{inc}, \texttt{read}, 0u0\rangle, \langle \texttt{inc}, \texttt{read}, 1u0\rangle)$ from the general gadget in Figure 4g.
  - Create Counter_Read $(\langle \texttt{inc}, \texttt{read}, u1\rangle, \langle \texttt{inc}, \texttt{read}, 0u1\rangle, \langle \texttt{inc}, \texttt{read}, 1u1\rangle)$ from the general gadget in Figure 4h.
  - Create Counter_Read $(\langle \texttt{copy}, \texttt{read}, u0\rangle, \langle \texttt{copy}, \texttt{read}, 0u0\rangle, \langle \texttt{copy}, \texttt{read}, 1u0\rangle)$ from the general gadget in Figure 4g.
  - Create Counter_Read $(\langle \texttt{copy}, \texttt{read}, u1\rangle, \langle \texttt{copy}, \texttt{read}, 0u1\rangle, \langle \texttt{copy}, \texttt{read}, 1u1\rangle)$ from the general gadget in Figure 4h.
- For each $i = 0, \ldots, m-2$:
  - Create Counter_Read_Msb $(\langle \texttt{inc}, \texttt{read}, bin(i,l)\rangle, \langle \texttt{copy}, \texttt{write}, i+1, 1\rangle)$ from the general gadget in Figure 4e if $bit(i,m,1,l) = 0$ or Figure 4f if $bit(i,m,1,l) = 1$.
- Create Counter_Read_Msb $(\langle \texttt{inc}, \texttt{read}, bin(m-1,l)\rangle, \langle \texttt{inc}, \texttt{write}, 1\rangle)$ from the general gadget in Figure 4e if $bit(i,m,1,l) = 0$ or Figure 4f if $bit(i,m,1,l) = 1$.
- For each $i = 0, \ldots, m-1$:
  - Create Counter_Read_Msb $(\langle \texttt{copy}, \texttt{read}, bin(i,l)\rangle, \langle \texttt{copy}, \texttt{write}, i, 1\rangle)$ from the general gadget in Figure 4e if $bit(i,m,1,l) = 0$ or Figure 4f if $bit(i,m,1,l) = 1$.
- For each $i = 0, \ldots, m-1$ and each $j = 1, \ldots, l-1$:
  - Create Counter_Write $(\langle \texttt{copy}, \texttt{write}, i, j\rangle, \langle \texttt{copy}, \texttt{write}, i, j+1\rangle)$ from the general gadget in Figure 4k if $bit(i,m,1,j) = 0$ or Figure 4l if $bit(i,m,1,j) = 1$.
- For each $i = 1, \ldots, l-1$:

  - Create Counter_Write $(\langle \texttt{inc}, \texttt{write}, i\rangle, \langle \texttt{inc}, \texttt{write}, i+1\rangle)$ from the general gadget in Figure 4k.

- For each $i = 0, \ldots, m-1$:
  - Create Counter_Write_Msb $(\langle \texttt{copy}, \texttt{write}, i, l\rangle, \langle \texttt{copy}, \texttt{down\_z\_0}, 1\rangle)$ from the general gadget in Figure 4i if $bit(i, m, 1, l) = 0$ or Figure 4j if $bit(i, m, 1, l) = 1$.
- Create Counter_Write_Msb $(\langle \texttt{inc}, \texttt{write}, l\rangle, \langle \texttt{inc}, \texttt{down\_z\_0}, 1\rangle)$ from the general gadget in Figure 4i.
- For each $i = 1, \ldots, 3l-1$:
  - Create Down_Column $(\langle \texttt{inc}, \texttt{down\_z\_0}, i\rangle, \langle \texttt{inc}, \texttt{down\_z\_0}, i+1\rangle)$ from the general gadget in Figure 1b.
  - Create Down_Column $(\langle \texttt{copy}, \texttt{down\_z\_0}, i\rangle, \langle \texttt{copy}, \texttt{down\_z\_0}, i+1\rangle)$ from the general gadget in Figure 1b.
- Create Counter_Return_Column_Start $(\langle \texttt{inc}, \texttt{down\_z\_0}, 3l\rangle, \langle \texttt{inc}, \texttt{down\_z\_1}, 1\rangle)$ from the general gadget in Figure 4b.
- Create Counter_Return_Column_Start $(\langle \texttt{copy}, \texttt{down\_z\_0}, 3l\rangle, \langle \texttt{copy}, \texttt{down\_z\_1}, 1\rangle)$ from the general gadget in Figure 4b.
- For each $i = 1, \ldots, l-1$:
  - Create Counter_Return_Column $(\langle \texttt{inc}, \texttt{down\_z\_1}, i\rangle, \langle \texttt{inc}, \texttt{down\_z\_1}, i+1\rangle)$ from the general gadget in Figure 4c.
  - Create Counter_Return_Column $(\langle \texttt{copy}, \texttt{down\_z\_1}, i\rangle, \langle \texttt{copy}, \texttt{down\_z\_1}, i+1\rangle)$ from the general gadget in Figure 4c.
- Create Counter_End $(\langle \texttt{inc}, \texttt{down\_z\_1}, l\rangle, \langle \texttt{inc}, \texttt{start}\rangle, \langle \texttt{elw}, \texttt{found}\rangle)$ from the general gadget in Figure 4d.
- Create Counter_End $(\langle \texttt{copy}, \texttt{down\_z\_1}, l\rangle, \langle \texttt{copy}, \texttt{start}\rangle, \langle \texttt{lw}, \texttt{found}\rangle)$ from the general gadget in Figure 4d.

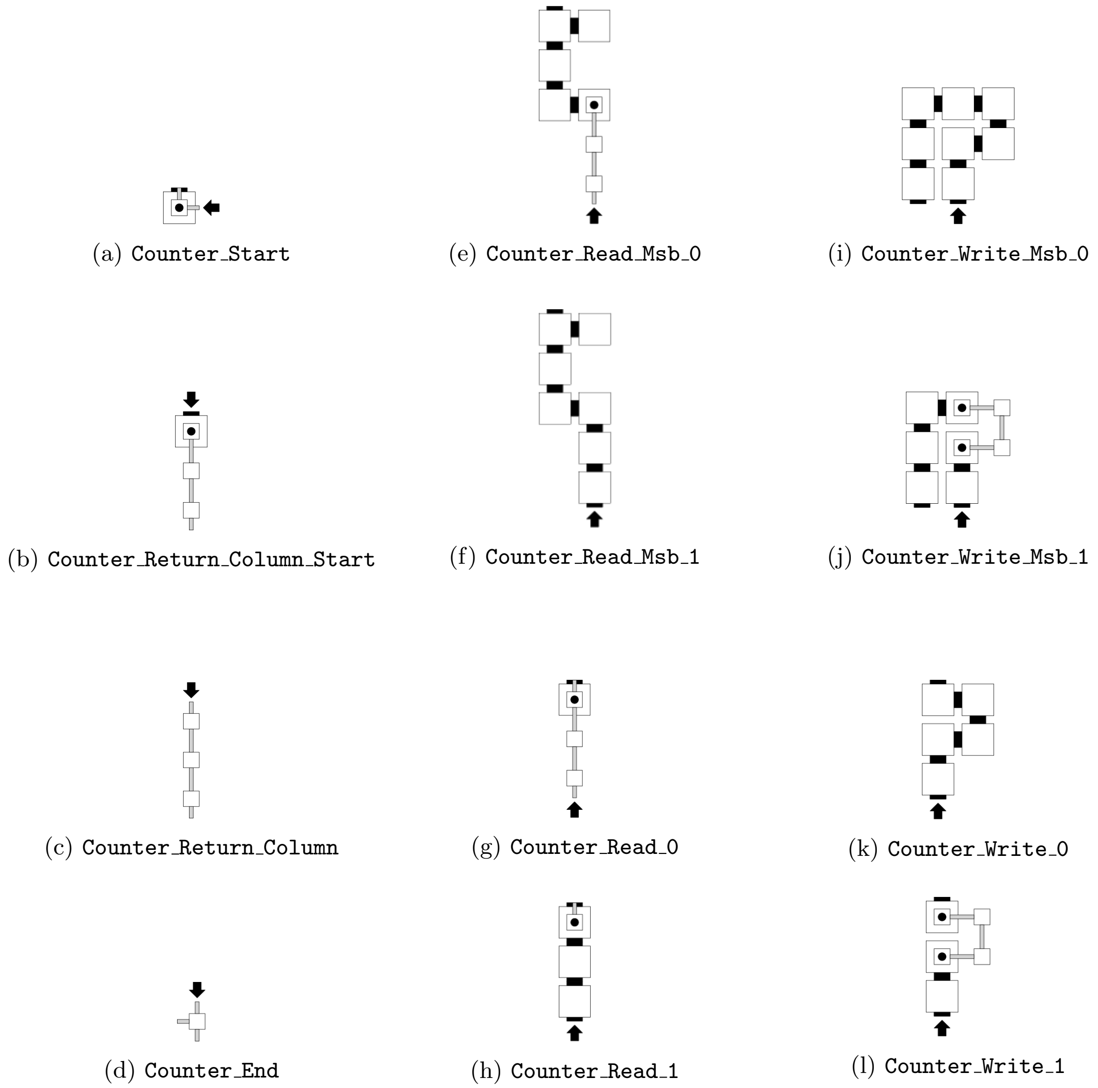
1.4 Left wall gadgets

The purpose of the Left_Wall gadget is to block the construction of additional Counter units for the next row, forcing them to produce a Left_Wall or End_Left_Wall unit. The preceding counter row will have been blocked by the preceding Left_Wall unit or the Seed unit. Additionally, the Left_Wall unit initiates a Return gadget with its east-facing glue.

1.5 Return gadget

The Return gadget ends with a Guess gadget on its east side which produces more Return gadgets in that direction, passing over the tiles that were left by the Counter gadgets until this row of Return gadgets reaches the preceding Right_Wall gadget and is forced to initiate a Right_Wall_Foundation gadget. The Right_Wall_Foundation gadget simply initiates another Right_Wall gadget.

This entire process repeats itself until an End_Left_Wall gadget is produced, at which point all Counter gadgets will be outputting 0.



(a) Counter_Start

(b) Counter_Return_Column_Start

(c) Counter_Return_Column

(d) Counter_End

(e) Counter_Read_Msb_0

(f) Counter_Read_Msb_1

(g) Counter_Read_0

(h) Counter_Read_1

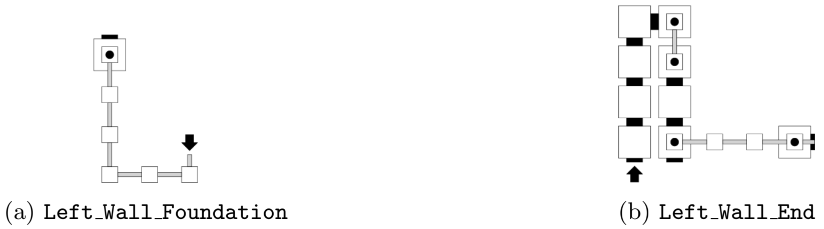
(i) Counter_Write_Msb_0

(j) Counter_Write_Msb_1

(k) Counter_Write_0

(l) Counter_Write_1

Figure 4: The Counter gadget unit.



(a) Left_Wall_Foundation

(b) Left_Wall_End

Figure 5: The Left_Wall gadget unit.

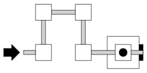
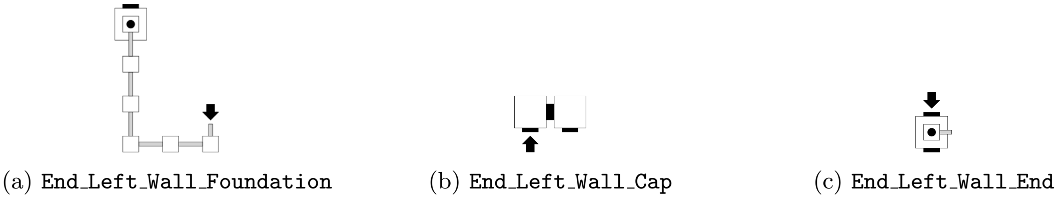


Figure 6: The Return_Row gadget.



(a) End_Left_Wall_Foundation (b) End_Left_Wall_Cap (c) End_Left_Wall_End

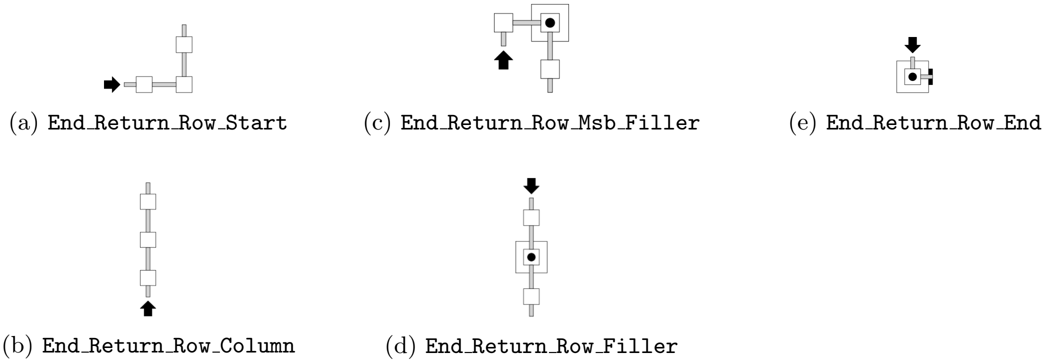
Figure 7: The End_Left_Wall gadget unit

1.6 End left wall gadgets

The End_Left_Wall gadget is made tall enough to block all future Shingle Tiles, and then initiates an End_Return gadget from its east-facing glue.

1.7 End return gadgets

The Guess gadget on the east end of End_Return gadget initiates either another End_Return gadget or the Roof gadget. The End_Return gadget is also extended northward in order to cover any empty tile spaces that would have been filled by the, never to return, counter row.



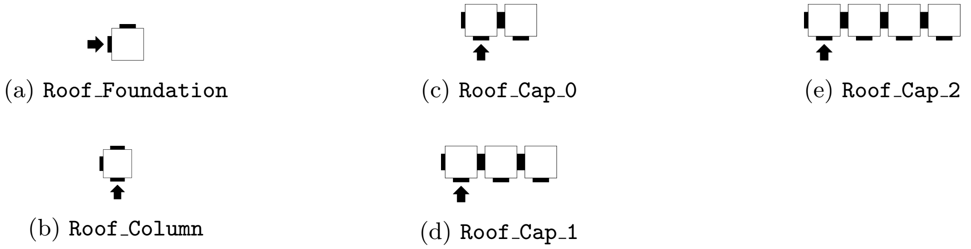
(a) End_Return_Row_Start (c) End_Return_Row_Msb_Filler (e) End_Return_Row_End

(b) End_Return_Row_Column (d) End_Return_Row_Filler

Figure 8: The End_Return_Row gadget unit.

1.8 Roof gadgets

The Roof gadget contains a hard-coded tile column that reaches above the protruding tiles from the last counter row, that then extends the assembly to a height of N. Each tile in the vertical column that extends past the counter row has a west-facing glue. The west-facing glue accepts a Shingle Tile which extend the roof westward until blocked by the End_Left_Wall gadget. The east-facing glue on the Roof gadget's north-most column tile is appended until it reaches $(k-1, N-1)$, which is the northeast corner of the construction's rectangle shape. Each tile that extends the roof eastward has a south-facing glue that accepts a Drop Tile. The Drop Tiles will replicate southward until blocked by a Right_Wall gadget or the "lip" on the Seed gadget.



(a) Roof_Foundation

(b) Roof_Column

(c) Roof_Cap_0

(d) Roof_Cap_1

(e) Roof_Cap_2

Figure 9: The Roof gadget unit.

1.9 Overview

An entire 2-digit, ternary example of the counter is illustrated in Figure ??.

1.10 Tile complexity

The tile complexity for each gadget is $O(...)$, but since we have $2m$ Counter gadgets, the consolidation of each gadget set brings the complexity of the entire construction to $O(...)$. □