Appendix A

FlatTop processor schematics and layouts

This appendix gives the detailed Cadence schematics and layout for the FlatTop universal parallel reversible processor which we discussed in §6.7. These diagrams were also included in our conference paper [60].

A.1 High-level blocks

Figure A-1 shows a schematic block diagram of a single processing element cell. Note the three blocks, one for each of the three stages in the 3-phase SCRL pipeline which makes up the logic of the cell. The lollipop-shaped icon above each stage represents the set of swinging supply rails, in a particular phase, which drive the stage adiabatically. The cell has four inputs A, B, C, D which come from the four neighboring cells, and four corresponding outputs which go back out to those cells.

The SHIFT in and out signals are global signals shared by all cells; they tell the cells whether to operate in initialization mode or normal mode. In initialization mode the array of cells behaves as a shift register, and the array contents may be shifted in and out; in normal mode, the array just obeys the BBMCA update rules.

The boxes attached to the input are for setting initial conditions during HSPICE simulation of the circuit.

Figure A-2 shows our schematic icon representing an entire processing element. The icon portrays the 2×2 block of BBMCA cells which the PE is updating, with the PE inputs and outputs placed in the appropriate cells. The cell grid is rotated 45 degrees from the representation in fig. 6-19 to show how the CA array is orientated with respect to the edges of the chip. With this orientation, the array of processing elements can communicate along pathways that run parallel to the chip edges, making layout easier.

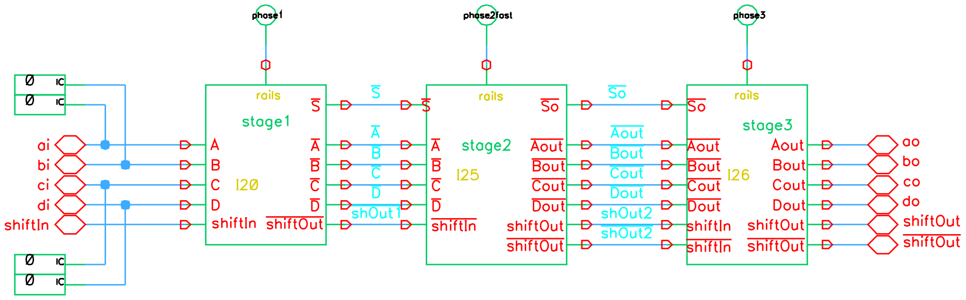


Figure A-1: Cadence schematic block diagram of the FLATTOP PE cell. The three blocks are the 3 SCRL logic stages, each clocked by a separate set of clock rails, indicated by the "lollipop" icons above the blocks, whose phases are offset by 1/3 cycle from each other.

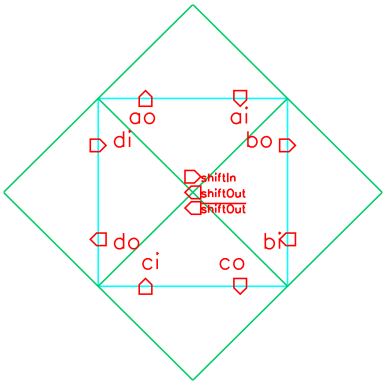


Figure A-2: Icon for a single FLATTOP cell. The depiction graphically illustrates the block of 4 CA cells, oriented by 45° to the inter-PE wiring, that is updated by the PE. The two terminals in each cell connect to and from the neighboring PE in the given direction, whose icon is overlapped with this one, as a reminder that the two PEs take turns updating the same CA cell.

Figure A-3 shows one corner of an array of these processing elements. In the upper left corners are pathways used for initialization and reading out the whole array when used as a shift-register. Along the edges of the array are edge cells which provide connections to pins, allowing the chip to communicate with other chips during normal operation. There are not enough pins to allow communication everywhere along the edge, so in other places the wires at the edge just loop around to feed back into the array. Every PE receives the global shift signals, which run horizontally.

Figure A-4 shows the entire 20×20 array of processing elements which we fabricated for testing purposes.

A.2 Detailed gate schematics

Below are the schematics for the logic in the three stages of each cell. We have not yet had time to size the transistors in our design so as to minimize power, but within each gate, we have, for uniformity, sized its transistors so that the worst-case conductances are the same as that of an inverter with a minimum-width n-FET and a twice-minimum-width p-FET.

Figure A-5 shows the logic in stage 1 which computes the inverses of the inputs, together with the inverse of the S (static) signal used in stage 2. Note that S is turned on when in *sh* (shift) mode. This special case was added to support array initialization.

Figure A-6(a) shows the gate used for computing each second-stage output. The forward part of stage 2 includes four repititions of this gate, differing as to which inputs are fed to the A, B, C, D pins, plus 6 "fast" inverters for generating the $A, B, C, D, S, shift$ signals from their inverses using the $f\phi_2$ and $\overline{f\phi_2}$ rails, plus one other inverter for generating $\overline{S}$ on the stage 2 output.

Finally, figure A-6(b) shows the gate that is repeated four times (with different pin assignments) in the forwards part of stage 3. This gate selects either A or the bit in the opposite corner of the block for passing through to the output, depending on *sh*. Thus if *sh* is on, input bits go to the opposite output bits, enabling the array as a whole to act as one large shift register, which can be used to initialize the array contents.

A.3 Cell layout

Figure A-7 shows the complete layout of a single PE cell. For clarity, the metal3 rails that run vertically across the entire cell are not shown. The cell measures 167.6 μm × 91.3 μm.

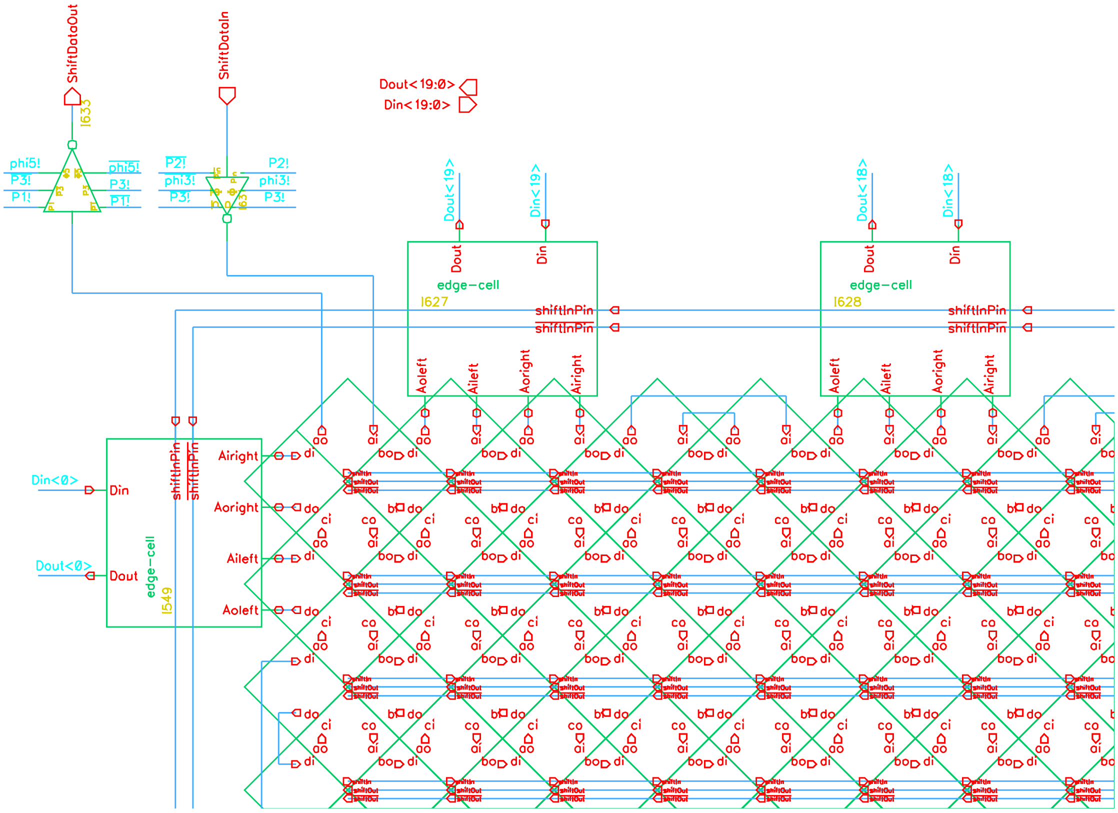


Figure A-3: Schematic diagram of the upper-left corner of a large array of FlatTop processing elements. The rows and columns of cells of CA array being simulating can be seen running diagonally across the array. You can see how the PE icons (fig. A-2) are overlapped to represent the shared management of each CA cell. Meanwhile, clock-power signals are strapped horizontally across the array. Periodically along the edge of the array are units for inter-chip communication. Other signals along the edges are simply wrapped around to a neighboring edge cell. The buffers in the far upper left corner provide initialization and readout capabilities.

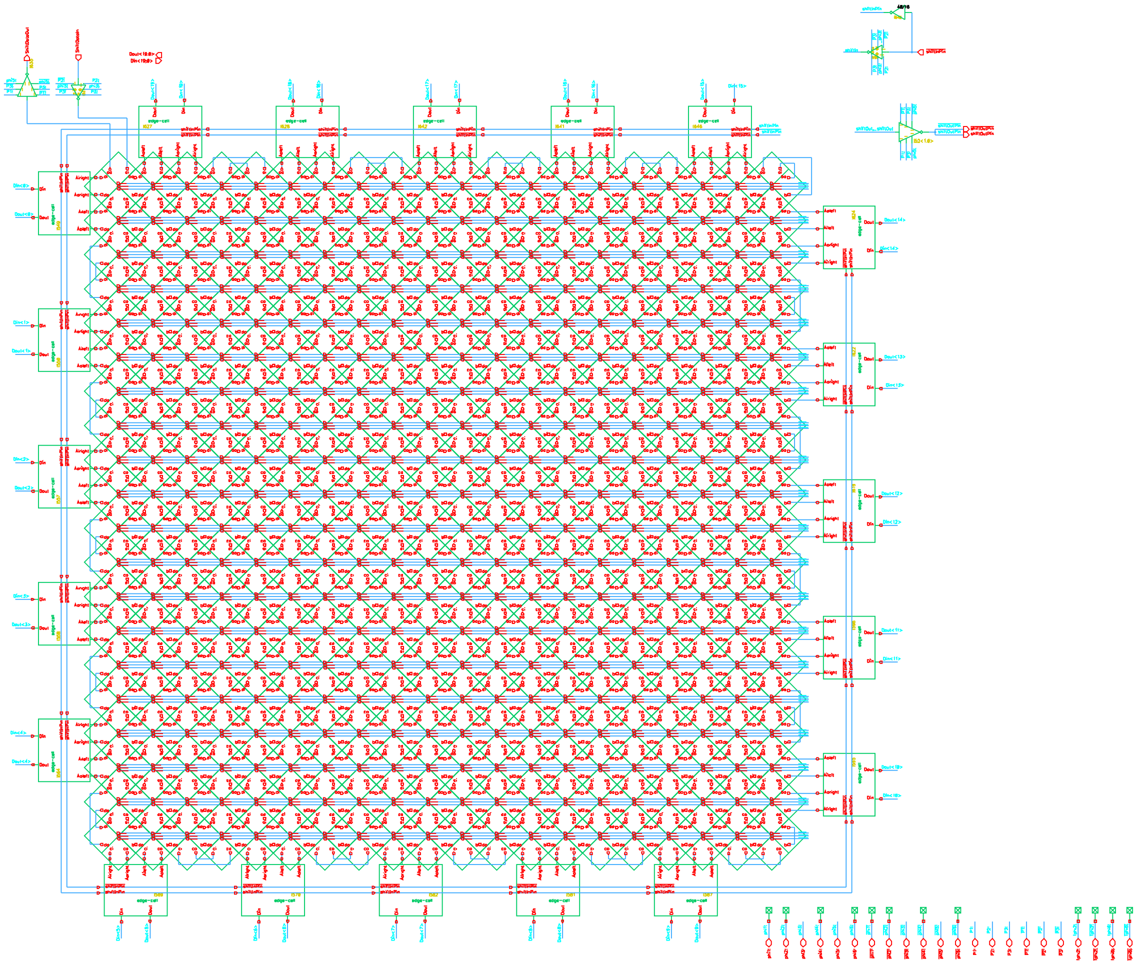


Figure A-4: Schematic block diagram of the full 20×20 array of FLATTOP processing elements which we fabricated on each die. This simply extends the structure shown in fig. A-3.

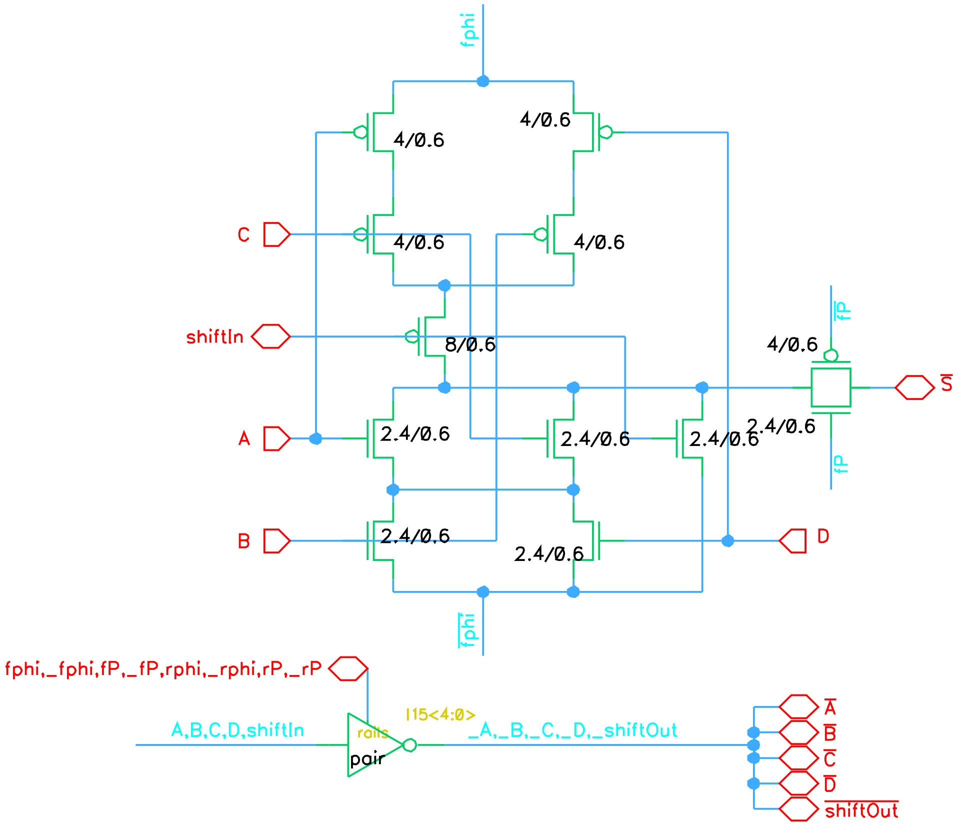
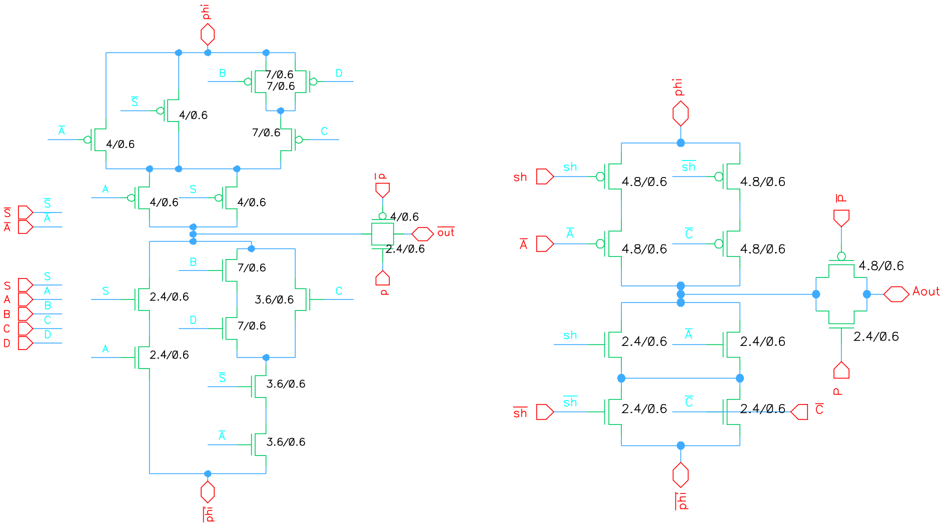


Figure A-5: Stage 1 logic, $\overline{S} = \overline{sh + (A + C)(B + D)}$.



(a) (b)

Figure A-6: (a) Gate for computing $\overline{A_{out}}$ in stage 2. (b) Gate for computing $A' = \overline{\overline{sh}\ \overline{A} + sh\ \overline{C}}$ in stage 3.