Institute or Neuroinformatics

ClassChip 2018

Design and Implementation of an Architectural Framework for a Neuromorphic Class Chip

CoACH Chip Architecture

Author:
Greg Burman

Supervisor:
Dr. Giacomo Indiveri

Revised and extended by:
Adrian M. Whatley

15 September 2020

Acknowledgements

I wish to thank Carsten Nielsen, for starting this project up with me, and sticking it through as far as he could, and Ole Richter, who regardless of the time of day (or current location), would advise me on any of the very many questions I had.

– Greg Burman

My thanks go to Mohammadali Sharifshazileh, Ning Qiao, Melika Payvand and Elisa Donati for helping me revise and update this document.

– Adrian M. Whatley

Contents

1 Introduction

This report documents the design and implementation of a chip architectural-framework for operating 30 test circuits, created for use in a neuromorphic engineering class and described in a companion report.

This report details these 30 test circuits as they constitute a core block, the peripheral circuits required to interface with input/output signals of this core, and and an overview of the entire project structure used to implement this design.

This entire design forms the basis of a $2 \times 2mm$ test chip, designed for the AMS $180\mu m$ process using the Cadence EDA programme.

2 Project Structure

Project files are hosted on INI's Gitlab server in the `CoACH` group: `https://code.ini.uzh.ch/CoACH`. It is split into two separate repositories:

- CoACH-cadence (`https://code.ini.uzh.ch/CoACH/CoACH-cadence`)

  This repository contains the `CoACH180` Cadence library of the project, as well as a `gitignore` for disregarding generated Cadence files that can cause git merge errors if tracked and committed.

- CoACH-docs (`https://code.ini.uzh.ch/CoACH/CoACH-docs`)

  This repository contains the schematics, waveforms, layout images, `MATLAB` script code and `LATEX` source files necessary to generate the two reports that constitute the documentation of this project.

Currently, the CoACH group visibility is set to `Public`, which is necessary to make the CoACH-docs repository publicly visible. However, the CoACH-cadence repository visibility is set to `Internal`, to control access to the design files (specifically the technology HitKit, which comes with specific licensing requirements). If access to design files is thought to be required, please contact one of the group members listed on the Gitlab group.

3 Signal Naming

The signal naming convention used is the ETH standard, modified to be compatible with asynchronous and analog circuit designs. The standard form is:

<circuit name>_<signal name>_<class1><class2><B><F*><I/O><!>

The <circuit_name> is a three letter abbreviation of the the full circuit name. The following list provides this mapping for the 30 test circuits provided in the core (as described in the companion report):

NFT: N-FET Device
PFT: P-FET Device
NFA: N-FET array
PFA: P-FET array
NSF: N-type Source-Follower
PSF: P-type Source-Follower
NDP: N-type Differential Pair
PDP: P-type Differential Pair
CCR: Current Correlator
BAB: Bump Antibump
NTA: N-type 5T Transamp
PTA: P-type 5T Transamp
WRT: Wide-range Transamp
CSR: Current-Splitter Array
WTA: Winner-Take-All Array
FOI: Follower-Integrator
FOD: Follower-Differentiator
RES: Resistive Element
SRE: Symmetric Resistive Element
SFP: Source-Follower Photoreceptor
DVS: DVS Pixel
SOS: Second-Order Section
LDS: Log Domain Synapse
DPI: DPI Synapse
DDI: Dual-DPI Synapse
AHN: Axon-Hillock IF Neuron
ACN: ADEXIF Classic Neuron
ATN: ADEXIF Thresholded Neuron
ASN: ADEXIF Sigma-Delta Neuron
HHN: Hodgkin-Huxley Neuron

The <signal_name> is the full name of the signal, typically prefixed by V or I as the

vast majority of signals are voltage/current-based. There are some exceptions, such as for AER REQ & ACK handshake signals, which don't have prefixes, or when signals are connecting lines between cells, in which case the prefix L is used. Examples include: VIN, V1, I2, IOUT, etc.

<class1> & <class2> are characters that specify expected/known characteristics of the signals. The following characters are used in <class1>:

A: asynchronous digital signals
U: analog current
V: analog voltage
N: nfet analog current as voltage
P: pfet analog current as voltage

Classes N and P are specifically used to indicate the type of device being biased. The following characters are used in <class2>:

C: clock (also spike)
R: reset
S: control/status/signal
D: data/address
T: test

Classes D and T are absent from this project, and are just listed for completeness. Class S is typically used for circuit enable lines. The remaining components of the signal name are optionally included:

B: active-low voltage/NFET-driven current
F*: signal buffered * times
I/O: pin direction
!: global signal

F* is also absent from this project. Originally, ! was used to indicate biases as global signals, however as this character is invalid Cadence, and biases are identified by N/P, it was also removed.

4 Pin-outs

4.1 Pad Budget

The number of pads available for use was determined by the dimensions of the chip and the width of the pad circuits available in the technology library. The chip dimensions are $2 \times 2\,mm$ (smallest allowable), and the pad widths were measured at approx. $90\,\mu m$. For each side of the chip perimeter, $180\,\mu m$ are unusable at the corners, leaving $1820\,\mu m$ available. This provides space for 20 pads per side, and a total of 80 pads for the whole chip. A count of the number of pads required is shown in Table 1.

Signal Type	No. pads
Input Voltage	16
Output Voltage	16
General IO	10
C2F Output	6
AER Bus	17
Shared	4
Power	9
Padframe	2
Total	**80**

Table 1: Count of chip pads required.

4.2 Power Rails

`GND` and `VDD` are provided from 7 power pads. This is required for separated analog (`GND`/`VDDD`) and digital power (`dGND`/`dVDD`), and an additional three separate analog `VDD` lines for each of the three ADEXIF neurons, in order to be able to compare their power consumption.

4.3 Package Pin-out

The chip is packaged into an 84-pin PLCC package. The resulting pinout is shown in Table 2. The bonding diagram is shown in Figure 20 and a diagram of the chip including the pad frame, pin numbers and signal names in Figure 21, both in Appendix C.

4.4 Reset

There are two active-low reset inputs, `PRst` and `SRst`. To reset the chip the following sequence MUST be followed: `SRst` down → `PRst` down → Wait → `PRst` up → `SRst` up. Some 5 ns–100 ns delay is required between each step. On power up, both resets inputs can be kept low initially.

Warning: If `PRst` is active without `SRst` being active, short circuits will occur on the chip!

Pin No.	Name	Pin No.	Name
1	IID9	43	AIN6
2	IID10	44	AIN7
3	IIDLY0	45	AIN8
4	IIDLY1	46	AIN9
5	PRst	47	AIN10
6	SRst	48	AIN11
7	IIACKB	49	AIN12
8	AERO0	50	AIN13
9	AERO1	51	AIN14
10	AERO2	52	AIN15
11	NC	53	XXX
12	AERACKB	54	AO0
13	AERREQB	55	AO1
14	C2FO0	56	AO2
15	C2FO1	57	AO3
16	C2FO2	58	AO4
17	C2FO3	59	AO5
18	C2FREQB	60	AO6
19	C2FACKB	61	AO7
20	IIREQB	62	AO8
21	IID0	63	AO9
22	IID1	64	AO10
23	IID2	65	AO11
24	IID3	66	AO12
25	IID4	67	AO13
26	DVDD	68	AO14
27	DGND	69	AO15
28	NCVDD	70	GO20
29	NCVDD	71	GO21
30	NCVDD	72	GO22
31	NCVDD	73	GO23
32	NCVDD	74	XXX
33	ALLVDD	75	GO5
34	ALLGND	76	GO4
35	AVDD	77	GO3
36	AGND	78	GO2
37	AIN0	79	GO1
38	AIN1	80	GO0
39	AIN2	81	IID5
40	AIN3	82	IID6
41	AIN4	83	IID7
42	AIN5	84	IID8

Table 2: Pin-out of 84-pin package. For details of GO[0:5] and GO2[0:3], see Table 10.

5 FET Device Sizing

5.1 Sizing for Mismatch

There is a tradeoff between mitigating mismatch by increasing the size of the transistors, and the amount of area the resulting design will use. Further, mismatch can only be reduced to some given proportion – it can never be completely removed. To estimate the effect of mismatch in this technology, an experiment was performed using a single N-FET device, in which its output current was observed while switching on the gate voltage, for different geometries. Results of mismatch analyses run are shown in Figure 1.

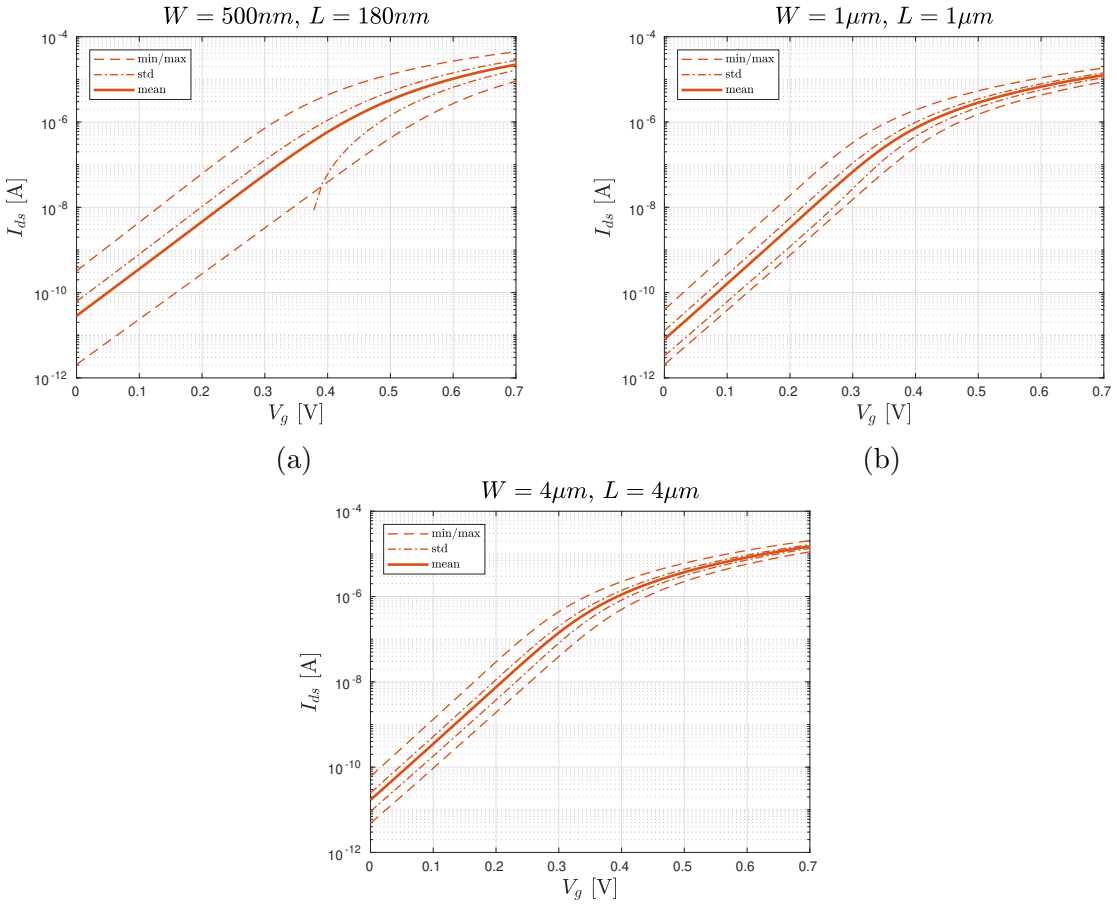


Figure 1: Mismatch analysis of an N-FET device with three different geometries: (a) minimum default technology sizing, (b) standard unit micron sizing, (c) large sizing.

.

As observed, the variation in device conductance decreases with increasing device geometry. Note that the y-axis is logarithmic. There is a large difference between the

minimum default technology sizing and the standard unit micron sizing, with diminishing returns observed with the large sizing used (at the cost of 16 times the amount of area required per device). As such, the standard unit micron sizing was chosen as the base size of all N-FET transistors used, unless otherwise required – either for additional mismatch mitigation, or for other reasons provided.

5.2 Conduction Matching

There is a lower mobility of holes in P-FET devices than of electrons in N-FETs. As a result, the output current of these devices will be lower by some proportion for the same input gate voltage. To test this, an experiment was performed using an inverter. A measure of matching is the point at which the output of the inverter switches. This was tested with both N-FET & P-FET give the same unit micron sizing, and then for increased P-FET widths. Test results are shown in Figure 2.

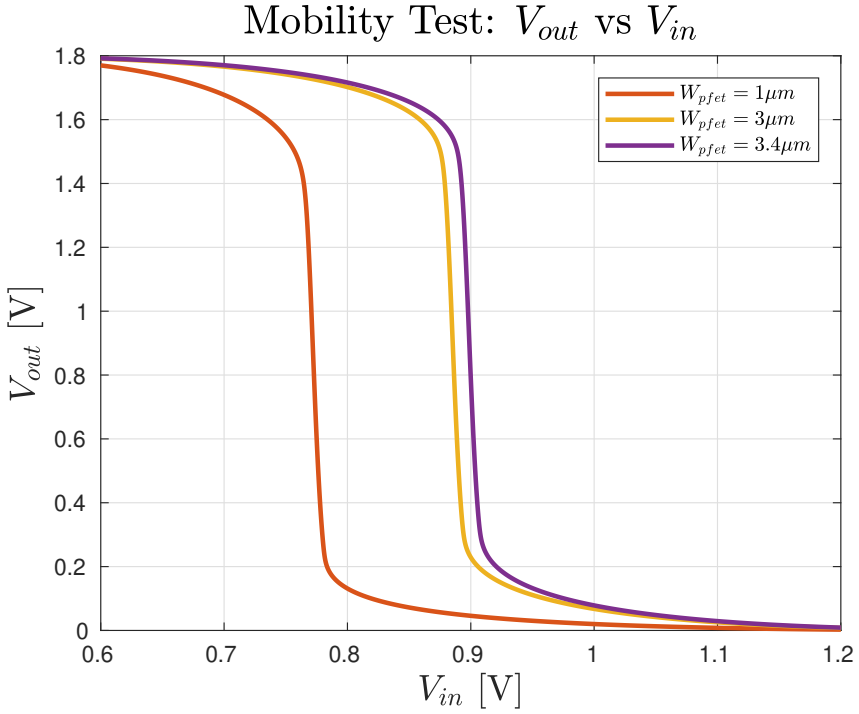


Figure 2: Output of an inverter, with different P-FET device widths.

As we are switching between $0V$ and $1.8V$, we consider $V_{out} = 0.9V$ to be the midpoint of the switch, and expect the response of devices with the same conduction to center on $V_{in} = 0.9V$. As seen in the results, when both devices have unit micron widths, the switching point is far off, at around $0.77V$. Simply multiplying the width of the P-FET device by three makes a significant difference, with the midpoint of the switch observed at $0.885V$. A slight adjustment to a width of $W_{pfet} = 3.4\mu m$ gives an

approximately exact switching point of $0.899V$. As such, N-FET sizing $W/L = 1/1\mu m$ and P-FET sizing of $W/L = 3.4/1\mu m$ were chosen.

6 Chip Architecture

The 30 test circuits described in the companion report constitute the core of the chip. To interact with these circuits, peripheral connecting blocks were designed and implemented. The 287 signals required can be divided as follows:

- Voltage inputs (40)
- Voltage outputs (38)
- Bias (current) inputs (82)
- Current outputs (95)
- Control signals (9)
- AER handshaking (11)
- Current inputs (2)
- Miscellaneous outputs (8)

The voltage signals are interfaced using an analog voltage demultiplexer for voltage inputs and an analog voltage multiplexer for voltage outputs. Bias signals are set by a 7-bit, programmable Bias Generator, of which an instance in this technology already exists. Most outputs are current-based, and these are output to an AER bus via a bank of current-monitors. Control signals are provided by an AER logic block that decodes AER words into a set of control states that are stored in latches. AER `REQ` and `ACK` lines are tied into the AER bus directly. The general IO pins are routed directly to pads. An overview of the core and connecting circuitry is shown in Figure 3.

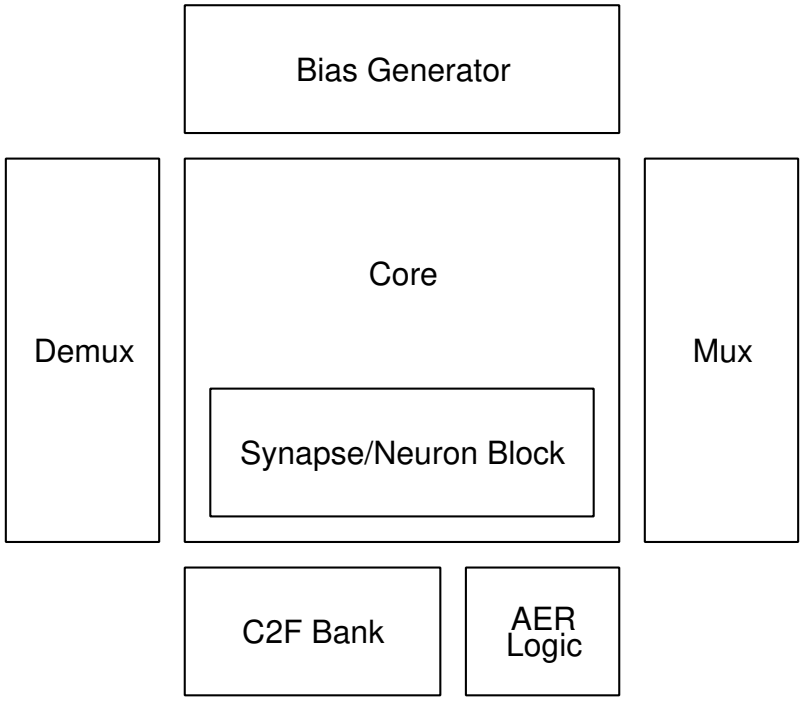


Figure 3: Overview block diagram of the chip architecture. Note that blocks are not drawn to scale – the core is in fact very small, and the bias generator is larger than the other blocks combined.

6.1 Analog Voltage Demultiplexer

The demultiplexer receives 16 voltage inputs from allocated chip pads and connects them to supply the 40 input voltage signals required by the core circuits. Select-logic is used to specify which circuits are connected to the pads, with the disabled lines pulled to `GND` or `VDD` as required (depending on the input transistors in the circuits, they are only switched 'off' if pulled to the correct rail). The full mapping of input voltage signals is shown in Table 3. Note that inputs to same circuit types, or circuits which one might want to compare, are kept on the same select line.

This is achieved by using N- and P-type T-gate switches with enable. When the switch is enabled, the input voltage is allowed to pass through. When disabled, it is prevented from passing the switch, and the output is pulled to `GND` or `VDD`, depending on the switch used. Schematics of these switches are shown in Figure 4.

Digital logic gates are used to construct the select logic. It has two selects as inputs. This is summarised in Table 4. A schematic of the logic gates are shown in Figure 5.

Input	Select Line 0 (P)	Select Line 1 (P)	Select Line 2 (N)
1	WTA_VIN1_VI	PFT_VG_VI	NFT_VG_VI
2	WTA_VIN2_VI	PFT_VB_VI	NFA_VG_VI
3	WTA_VIN3_VI	PFA_VG_VI	NSF_VIN_VI
4	WTA_VIN4_VI	PSF_VIN_VI	NTA_VP_VI
5	WTA_VIN5_VI	PDP_V1_VI	NTA_VN_VI
6	WTA_VIN6_VI	PDP_V2_VI	NDP_V1_VI
7	WTA_VIN7_VI	CSR_VG_VI	NDP_V2_VI
8	WTA_VIN8_VI	PTA_VP_VI	WRT_VP_VI
9	WTA_VIN9_VI	PTA_VN_VI	WRT_VN_VI
10	WTA_VIN10_VI	-	FOI_VIN_VI
11	WTA_VIN11_VI	-	FOD_VIN_VI
12	WTA_VIN12_VI	-	FOD_VREF_VI
13	WTA_VIN13_VI	-	BAB_V1_VI
14	WTA_VIN14_VI	-	BAB_V2_VI
15	WTA_VIN15_VI	-	SOS_VIN_VI
16	WTA_VIN16_VI	-	-

Table 3: Mapping of demultiplexer inputs to outputs, as determined by select (P and N refer to the type of switches used).

Select 0	Select 1	Select Line
0	0	-
0	1	0
1	0	1
1	1	2

Table 4: Select logic mapping, showing the select line activated for the given select inputs.

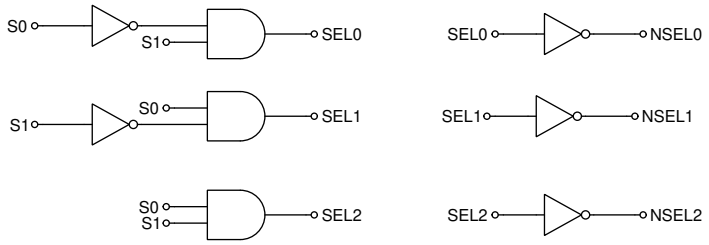
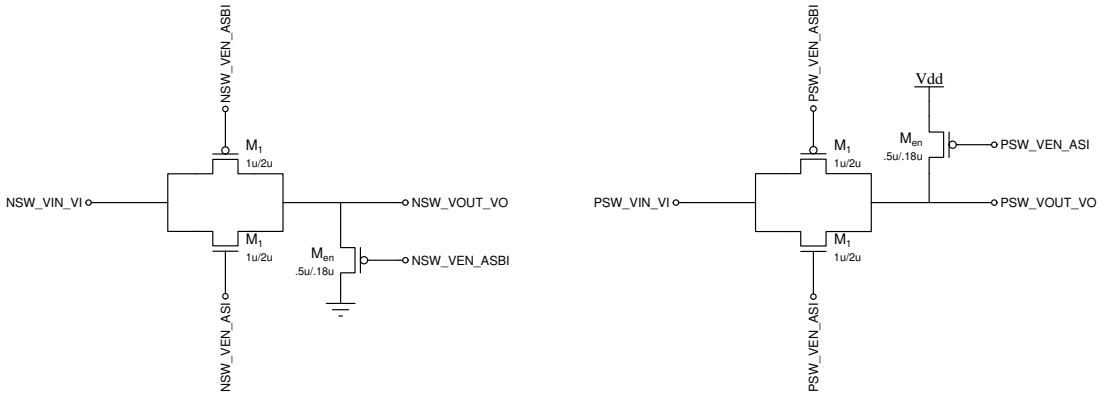


Figure 5: Logic gate circuit used to construct the select logic.

The full demultiplexer block is constructed from 16 individual demultiplexers, each consisting of two P-type switches and one N-type switch, and the select logic. A block diagram of the individual demultiplexer is shown in Figure 6, while the full block is shown in Figure 7.



(a) (b)

Figure 4: N- and P-type T-gate switches, with enable lines that pull the output to GND or VDD when disabled.

.

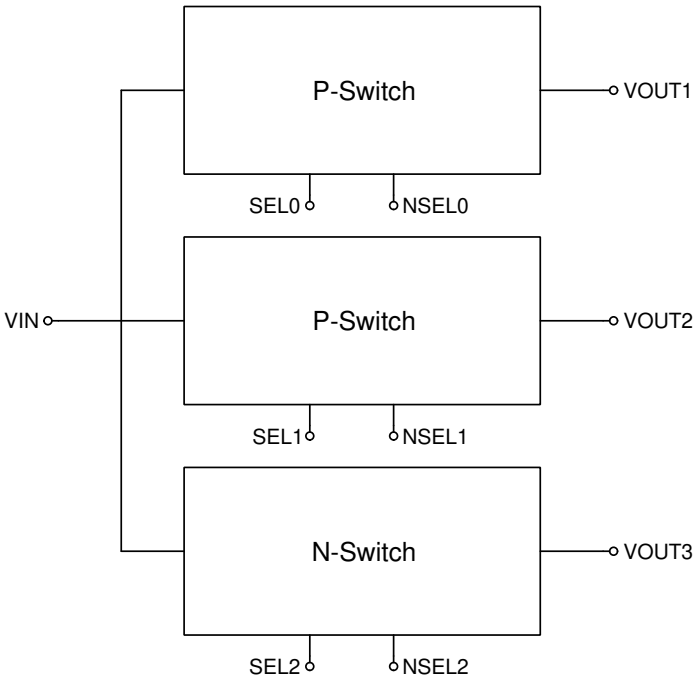


Figure 6: An individual multiplexer, which switches forwards the input voltage to the selected output.

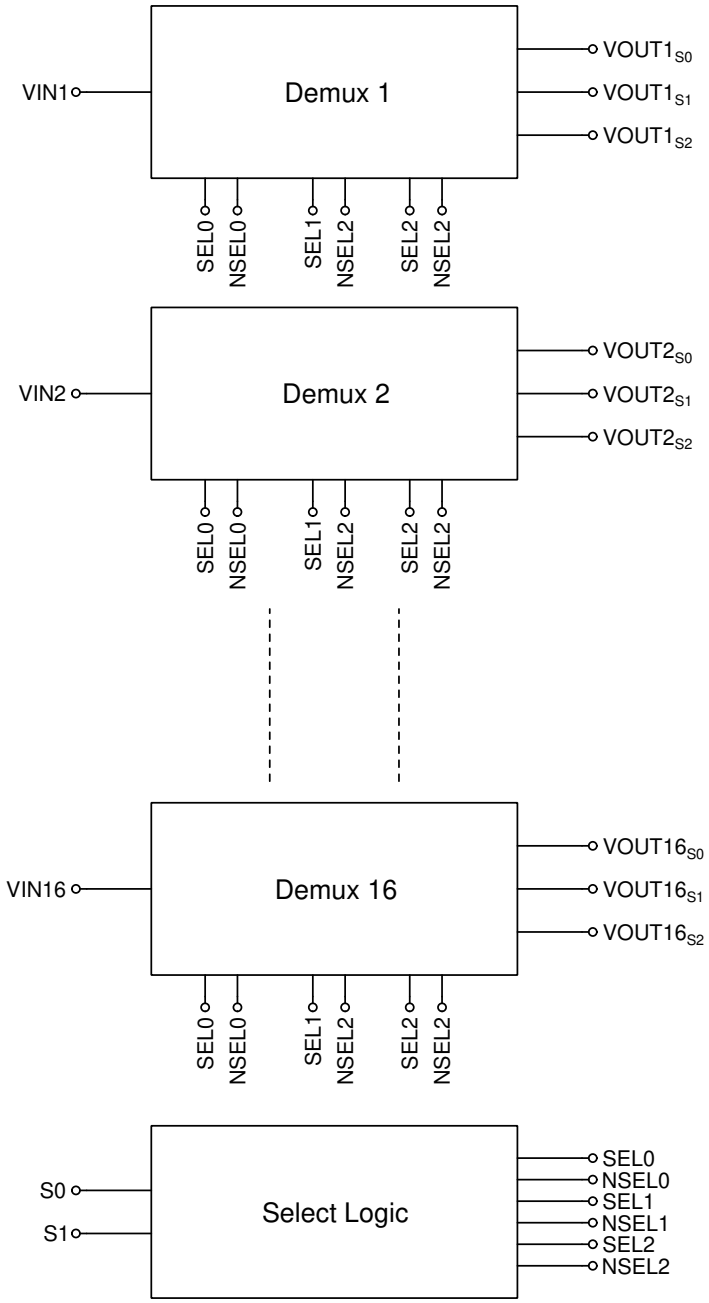


Figure 7: The full multiplexer, which aggregates 16 single multiplexers and the select logic to switch 16 voltage inputs to the selected outputs.

6.2 Analog Voltage Multiplexer

(Note: implementation has since changed)

The multiplexer provides 16 voltage outputs to allocated chip pads, received from 40

Select Line 0	Select Line 1	Select Line 2	Output
WTA_VOUT1_VO	NSF_VOUT_VO	LDS_VSYN_VO	1
WTA_VOUT2_VO	PSF_VOUT_VO	DPI_VSYN_VO	2
WTA_VOUT3_VO	NTA_VOUT_VO	DDI_VEXSYN_VO	3
WTA_VOUT4_VO	PTA_VOUT_VO	DDI_VINHSYN_VO	4
WTA_VOUT5_VO	WRT_VOUT_VO	AHN_VMEM_VO	5
WTA_VOUT6_VO	FOI_VOUT_VO	ACN_VMEM_VO	6
WTA_VOUT7_VO	FOD_VOUT_VO	ATN_VMEM_VO	7
WTA_VOUT8_VO	SOS_VS1_VO	ASN_VMEM_VO	8
WTA_VOUT9_VO	SOS_VS2_VO	HHN_VMEM_VO	9
WTA_VOUT10_VO	SFP_VOUT_VO	HHN_VNA_VO	10
WTA_VOUT11_VO	-	HHN_VK_VO	11
WTA_VOUT12_VO	-	HHN_VCA_VO	12
WTA_VOUT13_VO	-	-	13
WTA_VOUT14_VO	-	-	14
WTA_VOUT15_VO	-	-	15
WTA_VOUT16_VO	-	-	16

Table 5: Mapping of multiplexers inputs to outputs, as determined by select.

output voltage signals provided by the core circuits. As with the demultiplexer, select logic is used to specify which circuits are connected to the pads, with the disabled lines left floating (required so they don't affect the output). The full mapping of output voltage signals is shown in Table 5. Note that outputs from same circuit types, or circuits which one might want to compare, are kept on the same select line.

This is achieved by using buffers (wide-range transamps with V_- shorted to V_{out}) with enable. When the buffer is enabled, current is allowed to flow through the circuit branches. When disabled, these branches are shut off, and the output is left floating. A schematic of the buffer circuit is shown in Figure 8.

Select 0	Select 1	Select Line
0	0	-
0	1	0
1	0	1
1	1	2

Table 6: Select logic mapping, showing the select line activated for the given select inputs.

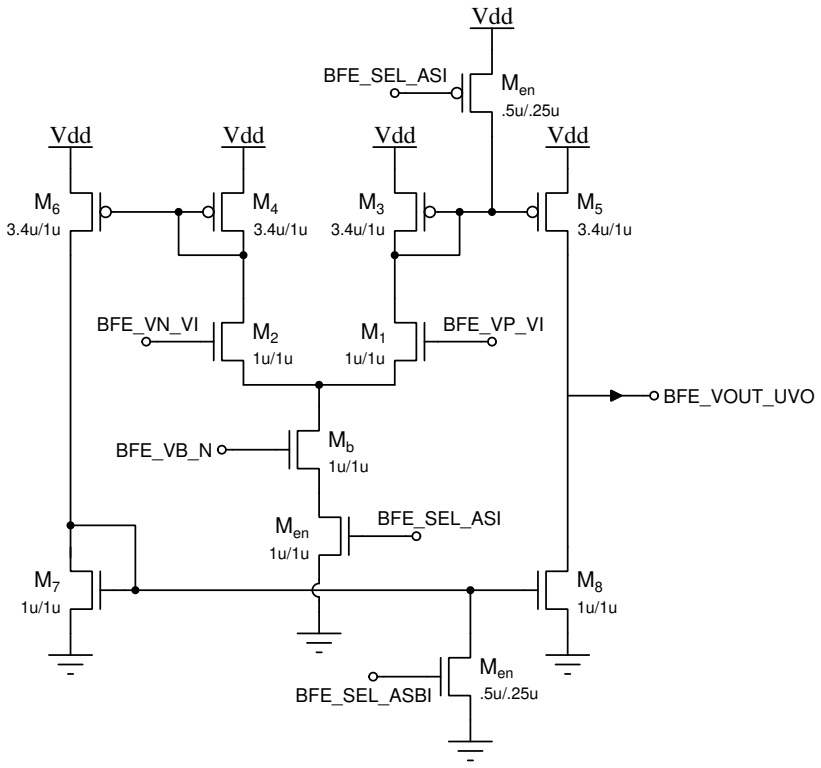


Figure 8: Buffer circuit with enable. When not selected, the input no longer drives the output, resulting in it floating.

Digital logic gates are used to construct the select logic. It has two selects as inputs. This is summarised in Table 6 and is the same logic used for the demultiplexer.

The full multiplexer block is constructed from 16 individual multiplexers, each consisting of three buffers, and the select logic. A block diagram of the individual multiplexers is shown in Figure 9, while the full block is shown in Figure 10.

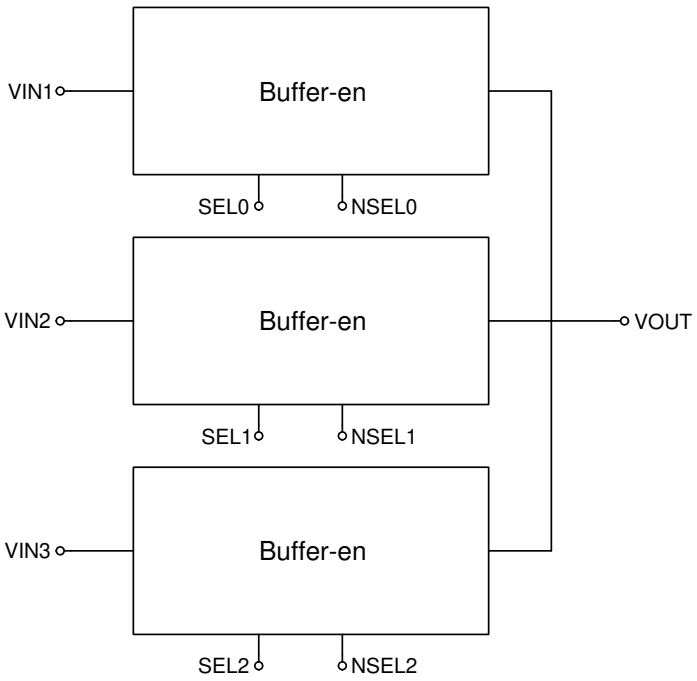


Figure 9: An individual multiplexer, which switches forwards the input voltage to the selected output.

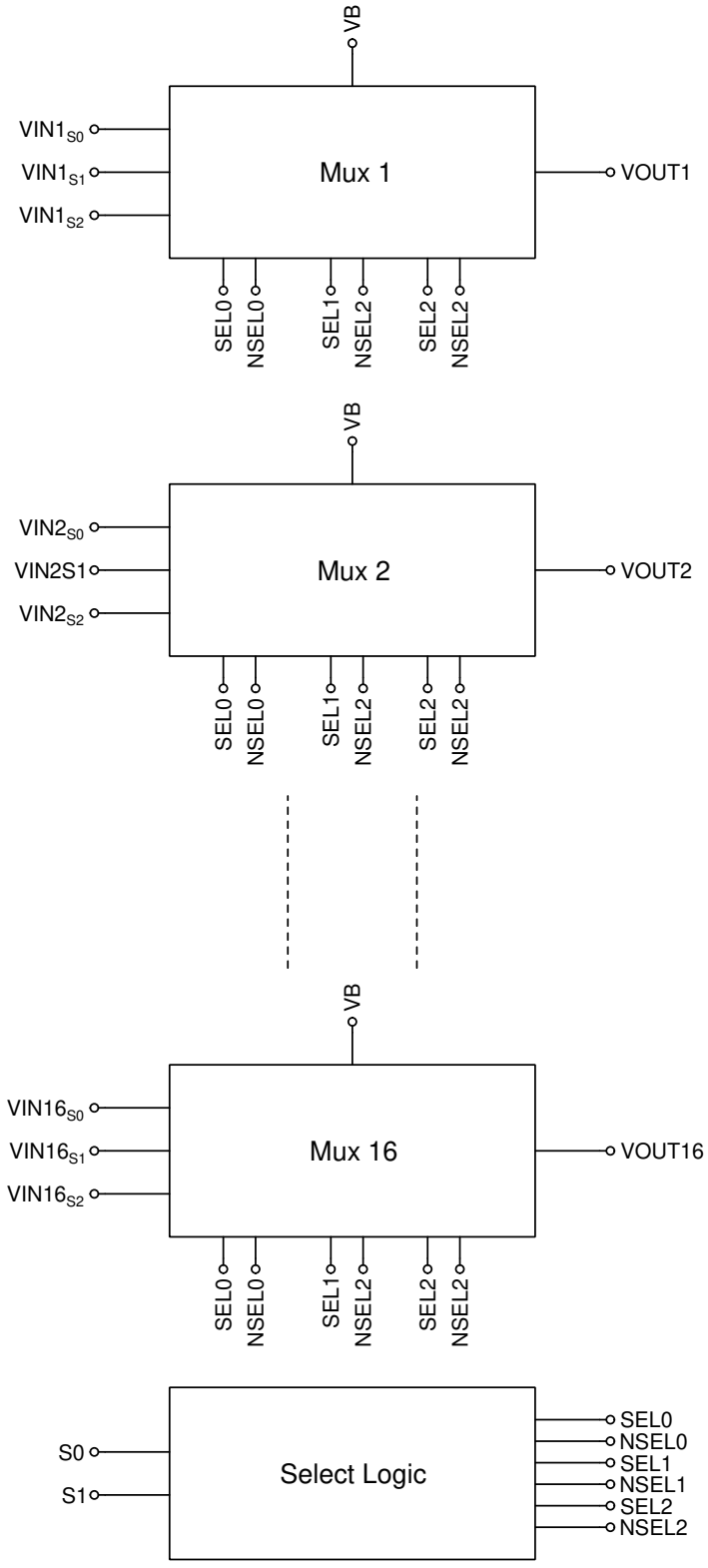


Figure 10: The full multiplexer, which aggregates 16 single multiplexers and the select logic to switch 16 voltage outputs from the selected outputs.

6.3 Bias Generator

A 7-bit bias generator is used to provide inputs to the bias inputs for all of the test circuits [1]. A full list of all biases is given in Table 7.

Address	Name	Function
0	BUFFER	
1	AHN_VPW_N	Axon-Hillock to set the refractory period
2	ACN_VADPPTAU_N	Pulse extender bias neuron adaptation
3	ACN_VADPWEIGHT_N	Classic ADEX neuron adaptation weight
4	ACN_VADPGAIN_N	Classic ADEX neuron adaptation gain
5	ACN_VADPTAU_P	Classic ADEX neuron adaptation tau
6	ACN_VADPCASC_N	Classic ADEX neuron enable AHP current
7	ACN_VREFR_N	Classic ADEX neuron refractory period
8	ACN_VLEAK_N	Classic ADEX neuron tau
9	ACN_VGAIN_N	Classic ADEX neuron gain
10	ACN_VDC_P	Classic ADEX neuron DC current
11	LDS_VTAU_P	Log-domain synapse tau
12	LDS_VWEIGHT_P	Log-domain synapse weight
13	ATN_VADPPTAU_N	Pulse extender bias for ADEXIF thresholded neuron
14	ATN_VADPWEIGHT_N	ADEXIF thresholded neuron adaptation weight
15	ATN_VADPGAIN_N	ADEXIF thresholded neuron adaptation gain
16	ATN_VADPTAU_P	ADEXIF thresholded neuron adaptation tau
17	ATN_VADPCASC_N	ADEXIF thresholded neuron enable AHP current
18	ATN_VREFR_N	ADEXIF thresholded neuron refractory period
19	ATN_VCC_N	ADEXIF thresholded neuron limiting bias current comparator
20	ATN_VSPKTHR_P	ADEXIF thresholded neuron spike threshold
21	ATN_VLEAK_N	ADEXIF thresholded neuron tau/leak
22	ATN_VGAIN_N	ADEXIF thresholded neuron gain
23	ATN_VDC_P	ADEXIF thresholded neuron DC current
24	DPI_VTAU_P	DPI synapse tau
25	DPI_VWEIGHT_N	DPI synapse weight
26	DPI_VTHR_N	DPI synapse gain
27	PEX_VTAU_N	Common pulse extender synapses
28	ASN_VADPPTAU_N	ADEXIF sigma-delta neuron pulse extender
29	ASN_VADPWEIGHT_N	ADEXIF sigma-delta neuron adaptation weight
30	ASN_VADPGAIN_N	ADEXIF sigma-delta neuron gain
31	ASN_VADPTAU_P	ADEXIF sigma-delta neuron tau
32	ASN_VADPCASC_N	ADEXIF sigma-delta neuron adaptation enable
33	ASN_VCC_N	ADEXIF sigma-delta neuron V comparator
34	ASN_VSPKTHR_P	ADEXIF sigma-delta neuron spike threshold
35	ASN_VLEAK_N	ADEXIF sigma-delta neuron tau/leak
36	ASN_VGAIN_N	ADEXIF sigma-delta neuron gain
37	ASN_VDC_P	ADEXIF sigma-delta neuron DC current
38	DDI_VWEIGHT_N	Double DPI weight
39	DDI_VTHR_N	Double DPI gain
40	DDI_VTAU_P	Double DPI tau
41	HHN_VBUF_N	Hodgkin-Huxley neuron monitor buffer
42	HHN_VAHPSAT_N	Hodgkin-Huxley neuron AHP saturation
43	HHN_VCAREST2_N	Hodgkin-Huxley neuron calcium threshold
44	HHN_VCABUF_N	Hodgkin-Huxley neuron calcium leak threshold
45	HHN_VCAREST_N	Hodgkin-Huxley neuron calcium rest

Address	Name	Function
46	HHN_VCAIN_P	Hodgkin-Huxley neuron calcium input
47	HHN_VKDSAT_N	Hodgkin-Huxley neuron potassium saturation
48	HHN_VPUWIDTH_N	Hodgkin-Huxley neuron adaptation pulse width
49	HHN_VKDTAU_N	Hodgkin-Huxley neuron potassium tau
50	HHN_VTHRES_N	Hodgkin-Huxley neuron neuron threshold
51	HHN_VNASAT_N	Hodgkin-Huxley neuron sodium saturation
52	HHN_VDC_P	Hodgkin-Huxley neuron DC current
53	HHN_VELEAK_N	Hodgkin-Huxley neuron leak
54	HHN_VNATAU_N	Hodgkin-Huxley neuron sodium tau
55	HHN_VGLEAK_N	Hodgkin-Huxley neuron threshold leak
56	HHN_VPADBIAS_N	Hodgkin-Huxley neuron adaptation spike generator
57	HHN_VPUTHRES_N	Hodgkin-Huxley neuron adaptation spike generator
95	SFP_VB_N	Source-follower photoreceptor
96	DVS_REFR_P	DVS
97	DVS_OFF_N	DVS
98	DVS_ON_N	DVS
99	DVS_DIFF_N	DVS
100	DVS_SF_P	DVS
101	DVS_CAS_N	DVS
102	DVS_PR_P	DVS
103	RR_BIAS_P	R2R buffer
104	C2F_HYS_P	C2F monitor
105	C2F_REF_L	C2F monitor
106	C2F_REF_H	C2F comparator
107	C2F_BIAS_P	C2F comparator
108	C2F_PWLK_P	C2F pulse extender
109	NTA_VB_N	N-type 5T transamp
110	CSR_VT_N	Current splitter
111	BAB_VB_N	Bump antibump
112	FOD_VB_N	Follower-differentiator
113	FOI_VB_N	Follower-integrator
114	NDP_VB_N	N-type differential pair
115	NSF_VB_N	N-type source-follower
116	SOS_VB2_N	Second-order section
117	PDP_VB_P	P-type differential pair
118	PSF_VB_P	P-type source-follower
119	PTA_VB_P	P-type 5T transamp
120	SOS_VB1_N	Second-order section
121	SRE_VB1_N	Symmetric resistive element
122	SRE_VB2_N	Resistive element
123	WRT_VB_N	Wide-range transamp
124	WTA_VB_N	WTA bias
125	WTA_VEX_N	WTA excitatory bias
126	WTA_VINH_N	WTA inhibitory bias
127	WTA_VGAIN_P	WTA gain

Table 7: Full list of bias input signals.

6.4 Current Monitor Bank

(Note: implementation not complete)

The current outputs are read out using current-to-frequency converters [2]. The monitors are essentially neurons, and spike at a frequency that correlates with the input current. The circuit was inherited from a previous design, and so device sizing, signal naming, etc. are left as-is. However, the layout did need to be re-implemented, as it was originally done in a different technology. Simulation result showing the spiking output against given current-inputs is shown in Figure 11.

The current-monitor bank comprises of 16 cells, with a 7-way select current-multiplexer to output the 98 current output signals (the bulk of current outputs come from the array circuits). The full mapping of output current signals is shown in Table 8.

Note that it is necessary to insert current-mirrors between the current-monitors and current outputs that are type `UO` (i.e. are driven by P-FETs). This is because the implemented current-monitor circuit expects output currents that are type `UBO` (i.e. are driven by N-FETs).

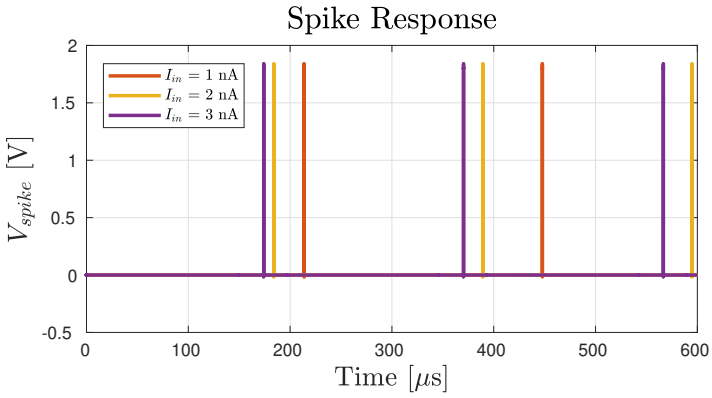


Figure 11: Spiking response of the current-monitor circuit for different input currents.

Select Line 0	Select Line 1	Select Line 2	Select Line 3	Select Line 4	Select Line 5	Select Line 6	Output
NFA_I1_UBO	PFA_I1_UO	WTA_IOUT1_UO	WTA_IALL1_UBO	CSR_I1_UO	NDP_I1_UBO	ACN_VMEM_VO	1
NFA_I2_UBO	PFA_I2_UO	WTA_IOUT2_UO	WTA_IALL2_UBO	CSR_I2_UO	NDP_I2_UBO	ATN_VMEM_VO	2
NFA_I3_UBO	PFA_I3_UO	WTA_IOUT3_UO	WTA_IALL3_UBO	CSR_I3_UO	PDP_I1_UO	ASN_VMEM_VO	3
NFA_I4_UBO	PFA_I4_UO	WTA_IOUT4_UO	WTA_IALL4_UBO	CSR_I4_UO	PDP_I2_UO	-	4
NFA_I5_UBO	PFA_I5_UO	WTA_IOUT5_UO	WTA_IALL5_UBO	CSR_I5_UO	CCR_IOUT_UBO	-	5
NFA_I6_UBO	PFA_I6_UO	WTA_IOUT6_UO	WTA_IALL6_UBO	CSR_I6_UO	BAB_I1_UBO	-	6
NFA_I7_UBO	PFA_I7_UO	WTA_IOUT7_UO	WTA_IALL7_UBO	CSR_I7_UO	BAB_I2_UBO	-	7
NFA_I8_UBO	PFA_I8_UO	WTA_IOUT8_UO	WTA_IALL8_UBO	CSR_I8_UO	BAB_IMID_UBO	-	8
NFA_I9_UBO	PFA_I9_UO	WTA_IOUT9_UO	WTA_IALL9_UBO	CSR_I9_UO	LDS_ISYN_UO	-	9
NFA_I10_UBO	PFA_I10_UO	WTA_IOUT10_UO	WTA_IALL10_UBO	CSR_I10_UO	DPI_ISYN_UO	-	10
NFA_I11_UBO	PFA_I11_UO	WTA_IOUT11_UO	WTA_IALL11_UBO	CSR_I11_UO	DDI_ISYN_UO	-	11
NFA_I12_UBO	PFA_I12_UO	WTA_IOUT12_UO	WTA_IALL12_UBO	CSR_I12_UO	NTA_IOUT_UO	-	12
NFA_I13_UBO	PFA_I13_UO	WTA_IOUT13_UO	WTA_IALL13_UBO	CSR_I13_UO	NTA_IOUT_UBO	-	13
NFA_I14_UBO	PFA_I14_UO	WTA_IOUT14_UO	WTA_IALL14_UBO	CSR_I14_UO	PTA_IOUT_UO	-	14
NFA_I15_UBO	PFA_I15_UO	WTA_IOUT15_UO	WTA_IALL15_UBO	CSR_I15_UO	PTA_IOUT_UBO	-	15
NFA_I16_UBO	PFA_I16_UO	WTA_IOUT16_UO	WTA_IALL16_UBO	CSR_I16_UO	-	-	16

Table 8: Full list of current output signals.

6.5 Control Signals

There are a total of 9 control signals in the test circuits that enable/disable specific circuit functions. Their state is maintained by corresponding latches that are set by an AER logic block (not in the scope of this report). The idea is that an AER word is read by the logic block and decoded to set the state of the control signal latches. The control signals are listed with their corresponding circuits in Table 9. They include two synapse select signals which are decoded to select one of three synapses according to Table 12.

AERC Bit No.	Control Signal	Circuit
0	SRE_VEN_VSI	Symmetric resistive element
1	WTA_VHEN_SI	Winner-take-all
2	DSY_S0_ASI	Synapse select 0
3	DSY_S1_ASI	Synapse select 1
4	ACN_ADPEN_ASI	ADEXIF classic neuron
5	ACN_DCEN_ASBI	ADEXIF classic neuron
6	ATN_DCEN_ASBI	ADEXIF thresholded neuron
7	ATN_ADPEN_ASI	ADEXIF thresholded neuron
8	ASN_DCEN_ASBI	ADEXIF sigma-delta neuron

Table 9: Full list of input control signals. AERC Bit No. refers to the position of the bit within the AERC event type on the Input Interface – see Section 8 and Table 14 for details. See Table 12 for the decoding of DSY_S1_ASI and DSY_S0_ASI.

6.6 Current Inputs

The current correlator is the only circuit which requires a sweeping input current. The original idea was to supply the circuit inputs, CCR_I1_UBI and CCR_I2_UBI, with currents from the bias generator. However, while it is able to provide very accurate currents, the generator is unable to provide a continuous sweeping output. As such, it was decided that these circuit inputs must be provided directly from pad inputs. See Table 10.

6.7 Miscellaneous Outputs

Four circuits, the FET devices and the resistive elements, have their currents read at the same terminals as their voltages are set. As a result, current monitors can't be used to read the currents without affecting the circuits themselves and thus resulting in erroneous measurements being made, and so their input terminals must be accessed directly from the pads. These output signals are listed with their corresponding circuits and pin names in Table 10.

No.	IO Signal	Circuit	Pin Name
1	NFT_VD_UVIO	N-FET device	GO22
2	NFT_VS_UVIO	N-FET device	GO20
3	PFT_VD_UVIO	P-FET device	GO21
4	PFT_VS_UVIO	P-FET device	GO23
5	RES_V1_UVIO	Resistive element	GO2
6	RES_V2_UVIO	Resistive element	GO4
7	SRE_V1_UVIO	Symmetric resistive element	GO0
8	SRE_V2_UVIO	Symmetric resistive element	GO1
	CCR_I1_UBI	Current correlator	GO5
	CCR_I2_UBI	Current correlator	GO3

Table 10: Full list of general IO signals.

6.8 AER Handshaking

The AER protocol is an asynchronous handshake. The DVS pixel and the ADEXIF neurons produce spikes as REQ events, and "wait" for an ACK event in response before resetting and producing the next spike. As such, each of these pixels has two signals which tie directly into the AER bus. These AER signals are listed with their corresponding circuits in Table 11.

No.	AER Signal	Circuit
1	DVS_ON_ASO	DVS pixel
2	DVS_OFF_ASO	DVS pixel
3	DVS_RESET_ARI	DVS pixel
4	AHN_ACK_ASBI	Axon-Hillock neuron
5	AHN_REQ_ASO	Axon-Hillock neuron
6	ACN_ACK_ASBI	ADEXIF classic neuron
7	ACN_REQ_ASO	ADEXIF classic neuron
8	ATN_ACK_ASI	ADEXIF thresholded neuron
9	ATN_REQ_ASO	ADEXIF thresholded neuron
10	ASN_ACK_ASI	ADEXIF sigma-delta neuron
11	ASN_REQ_ASO	ADEXIF sigma-delta neuron

Table 11: Full list of AER handshake signals.

7 Core Construction

The 30 test circuits mostly operate independently of one another, and so there is very little internal interaction in the core block. The exceptions to this are the duplication of the N- and P-type transamps, as well as the connection of the synapses and neurons in an internal block (described in the sections that follow). A capture of the Cadence schematic is shown in Figure 14 in Appendix A.

7.1 N-type & P-type Transamps

We are interested in both a voltage and current output from the N- and P-type 5T transamps. However, there are two difficulties here: first, reading both voltage and current is difficult to do without one measurement affecting the other, and second, the current output of the transamps can be both types UO and UBO.

The proposed solution was to have three duplicate instances of each transamp, with inputs and biases tied together, the output voltage read out of one instance and the two currents read out of the other two instances (measurements of each current could then added later for a measure of the full output current).

While it is expected that this method is susceptible to mismatch across the three instances, note that the mismatch analysis done in the companion report on these circuits included process mismatch (i.e. across chips on the same wafer), which carries significantly more variance than mismatch present within a chip. As such, it is expected that while the measurements will have variance, calculations of device transconductance based on them should yield reasonable results.

7.2 Synapse-Neuron Block

The synapses and neurons are arranged and connected so that each of the five neuron models can be driven by one of the three synapses (or a DC input current). The input to the synapses is an AER event, which is extended by a pulse extender (described in

the companion report). Each synapse mirrors its output current six times: once for each neuron and one to output directly. A block diagram of the configuration is shown in Figure 12.

A demultiplexer is used to switch the output of the pulse extender to the chosen synapse input, with the input to the other two synapses pulled to GND, disabling them. The demultiplexer uses the same T-gate switches as the full demultiplexer used to provide voltage inputs to the core, however in this case it costs only three N-type switches. The switching logic is summarised in Table 12.

DSY_S0_ASI	DSY_S1_ASI	Synapse Selected	
0	0	-	
0	1	LDS	Log domain synapse
1	0	DPI	DPI synapse
1	1	DDI	Dual DPI synapse

Table 12: Decoding of synapse select control signals DSY_S0_ASI and DSY_S1_ASI.

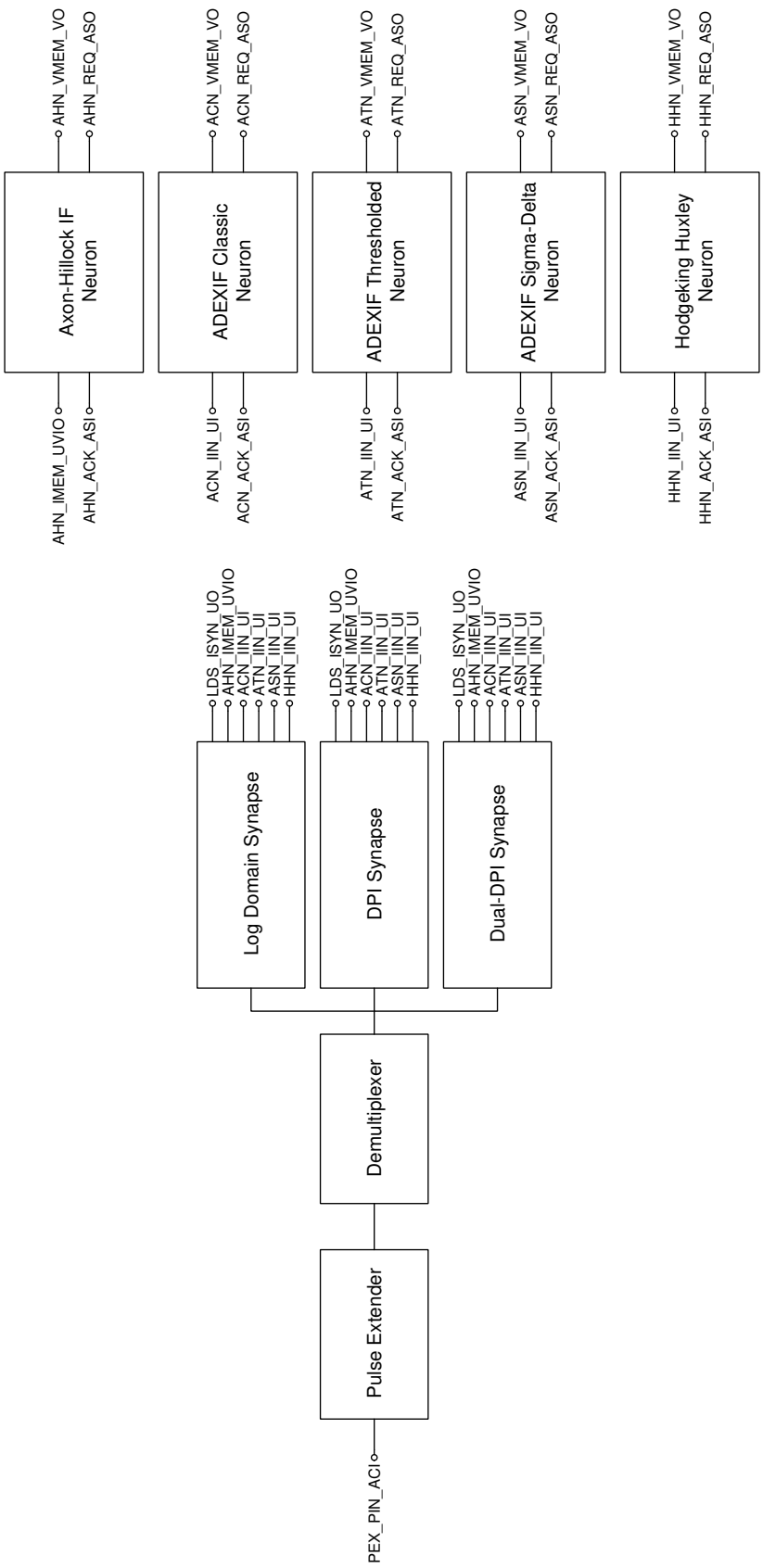


Figure 12: Block diagram of the synapse-neuron configuration.

8 Input Interface

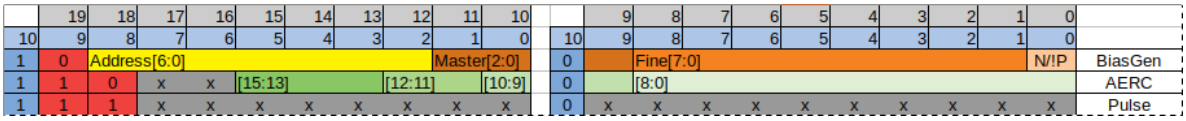


Figure 13: AER Input Interface. The chip requires a 20-bit AE word as input, with bits numbered as on the top row in grey. The physical Input Interface bus is however only 11 bits wide (`IID[10:0]`), so two AE cycles need to be sent one after the other (with bits numbered as on the second row from the top in light blue) to deliver the full 20 bits. The value of `IID10` distinguishes whether the high or low half of the full 20-bit word is being sent. There are three types of event recognized by the Input Interface: (1) BiasGen events; (2) AER Control (AERC) events; and (3) Pulse events. These event types are distinguished by the values of the bits shaded in red. Further details of the fields within the BiasGen and AERC events are given in Sections 8.1 and 8.2 respectively. x indicates a 'don't care' bit.

8.1 BiasGen Events

As shown in Figure 13, BiasGen events have four fields, the Address field (shown in yellow), the Master current field (shown in dark orange), the Fine current field (shown in mid-orange) and the N/!P field (shown in pale orange). The value in the Address field, taken from the Address column of Table 7, selects the particular bias being addressed. The value in the Master current field determines which master current branch of the bias generator is used as shown in Table 13. The value in the Fine current field can then take a value between 0 and 255. The N/!P field is set to 0 for P-type biases and 1 for N-type biases.

Master[2:0]			Master current
0	0	0	60 pA
0	0	1	460 pA
0	1	0	3.8 nA
0	1	1	30 nA
1	0	0	240 nA
1	0	1	Invalid
1	1	0	Invalid
1	1	1	Invalid

Table 13: BiasGen event Master current field values, showing the master current selected for the given field values.

8.2 AERC Events

AERC events contain several fields (shown in Figure 13 in various shades of green) as described in Table 14.

AERC event bits	Description
[15:13]	Current output mux select. The value in these three bits determines which of the current output signals are actually output according to Table 8. 000 ⇒ Select Line 0, 001 ⇒ Select Line 1 etc. Note that 111 is invalid.
[12:11]	Voltage output mux select. These two bits are used to provide the Select signals referred to in Table 6 which turn on a given Select Line which in turn selects which voltage output signals are actually output according to Table 5. Bit 12 ≡ Select 1 & bit 11 ≡ Select 0, but due to the decoding shown in Table 6, 01 ⇒ Select Line **1**, 10 ⇒ Select Line **0** & 11 ⇒ Select Line 2.
[10:9]	Voltage input demux select. These two bits are used to provide the Select signals referred to in Table 4 which turn on a given Select Line which in turn selects which voltage input signals actually receive input according to Table 3. Note that bit 10 ≡ Select 0 & bit 9 ≡ Select 1! However, due to the decoding shown in Table 4, here 01 ⇒ Select Line 0, 10 ⇒ Select Line 1 & 11 ⇒ Select Line 2.
[8:0]	Miscellaneous Control Signals. As described in Section 6.5 / Table 9.

Table 14: Description of the fields within the AERC event type.

8.3 Pulse Events

A pulse event is used to generate an input pulse to the currently selected synapse with a pulse width of around 28 ns. The synapse is selected according to the contents of AERC[3:2] in the most recent AERC event, equivalent to `DSY_S1_ASI` and `DSY_S0_ASI`, see Tables 9 and 12.

9 AER Output

The chip can produce seven different valid addresses on its AER output bus `AERO[2:0]` as shown in Table 15.

AERO[2:0]	Event source
000	?? neuron
001	?? neuron
010	?? neuron
011	?? neuron
100	?? neuron
101	DVS ON event
110	DVS OFF event
111	Invalid

Table 15: AER output addresses.

References

[1] T. Delbruck and P. Lichtsteiner, "Fully programmable bias current generator with 24 bit resolution per bias," in *Circuits and Systems, 2006. ISCAS 2006. Proceedings. 2006 IEEE International Symposium on*, pp. 4–pp, IEEE, 2006.

[2] N. Qiao and G. Indiveri, "An auto-scaling wide dynamic range current to frequency converter for real-time monitoring of signals in neuromorphic systems," in *Biomedical Circuits and Systems Conference, (BioCAS), 2016*, pp. 160–163, IEEE, 2016.

Appendices

A Core Schematic

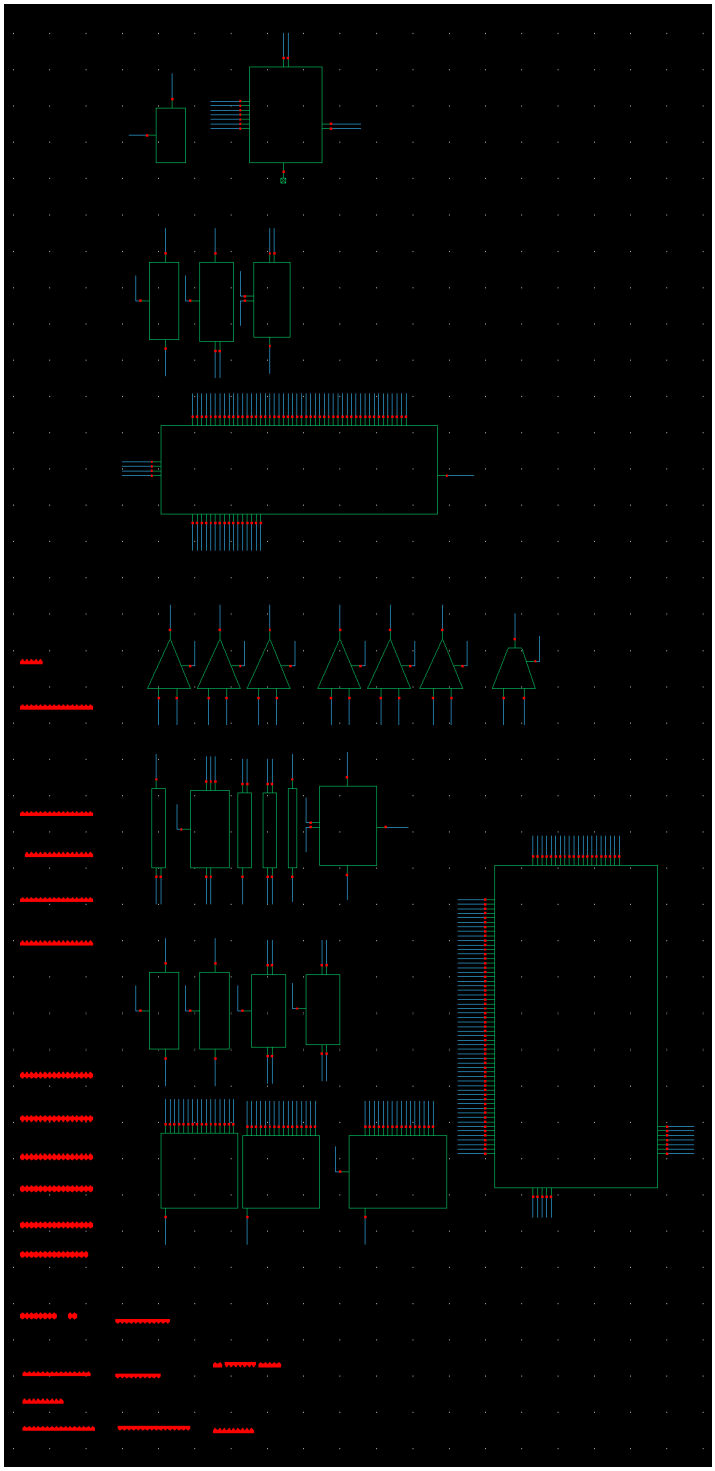


Figure 14: Core schematic.

B Layouts

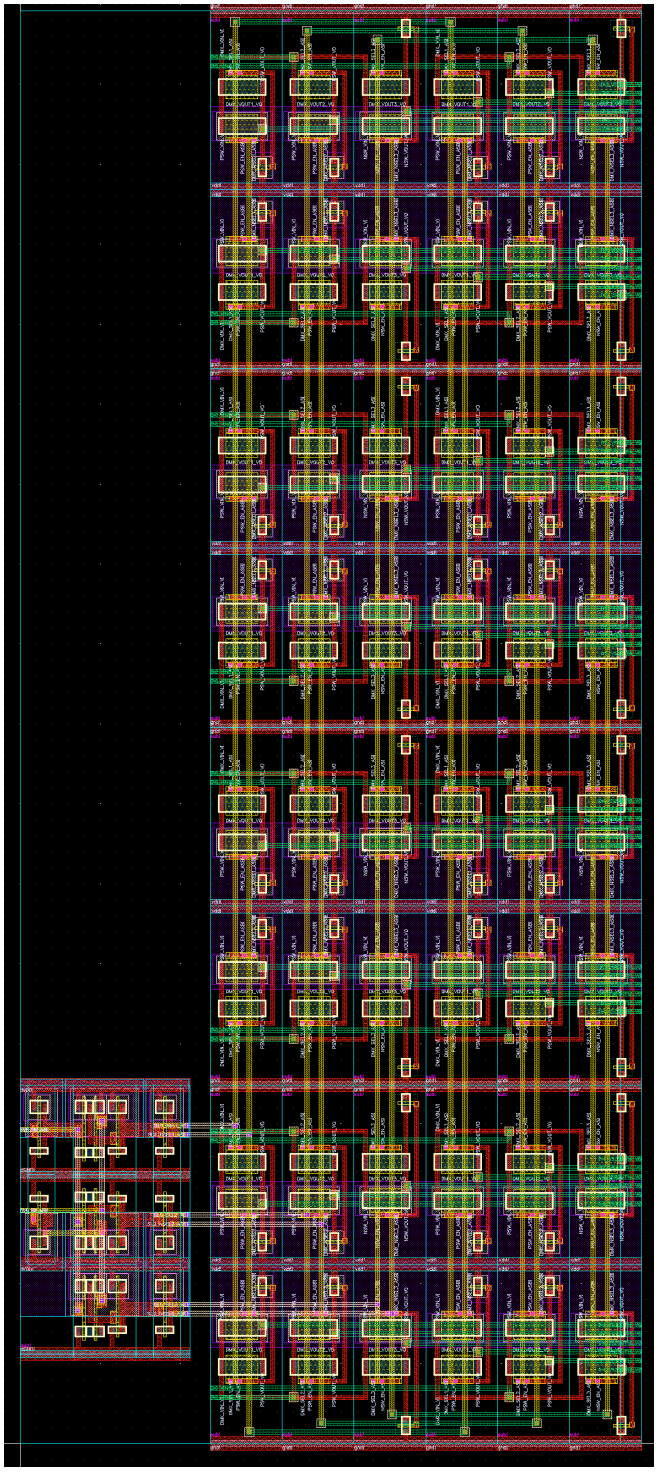


Figure 15: Demultiplexer layout.

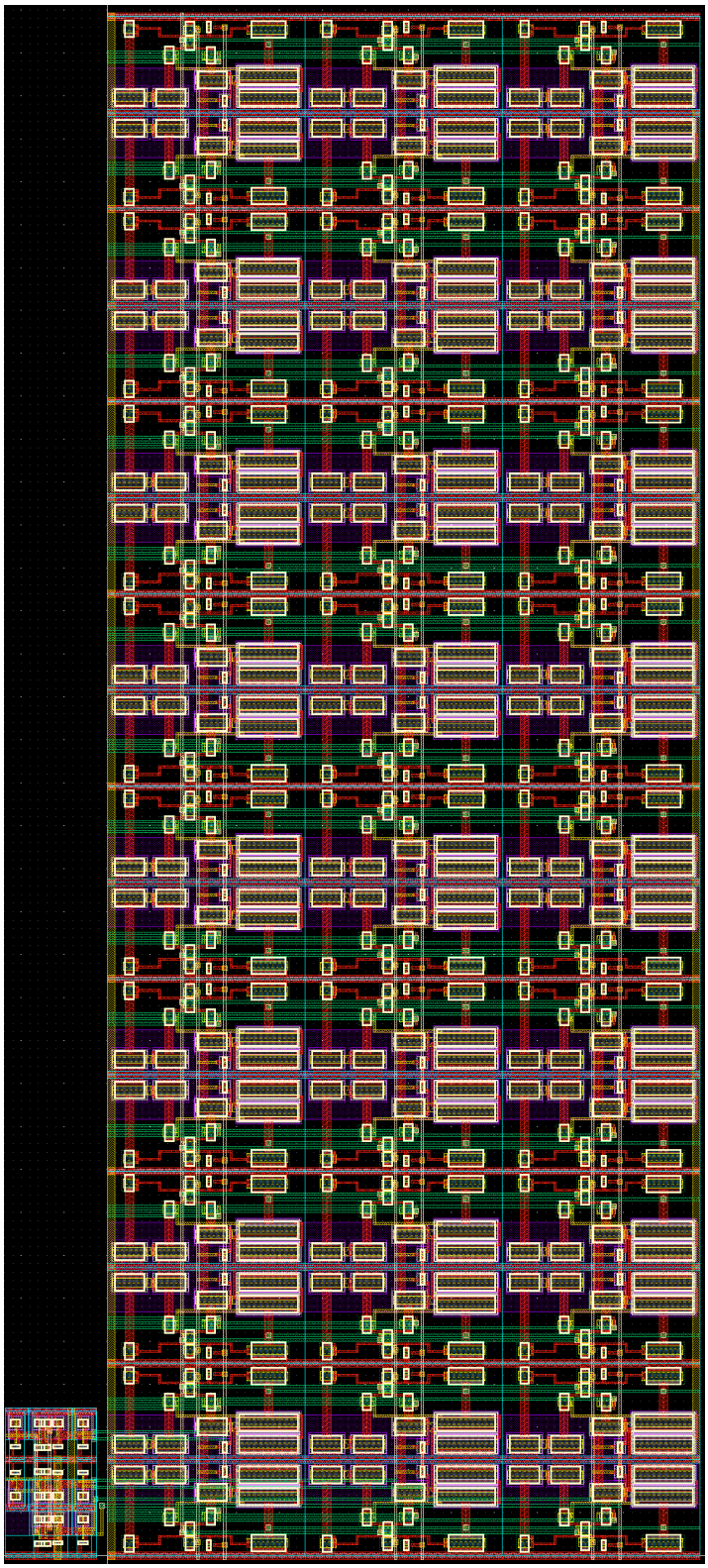


Figure 16: Multiplexer layout.

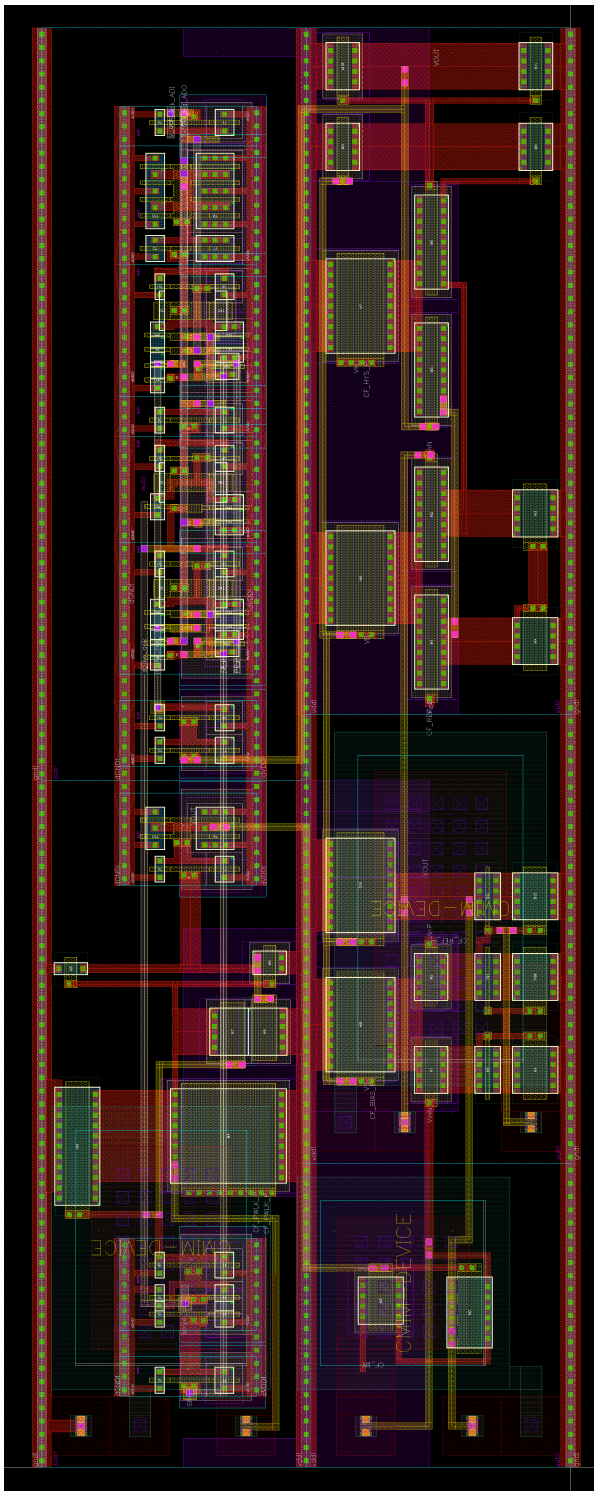


Figure 17: Current monitor layout.

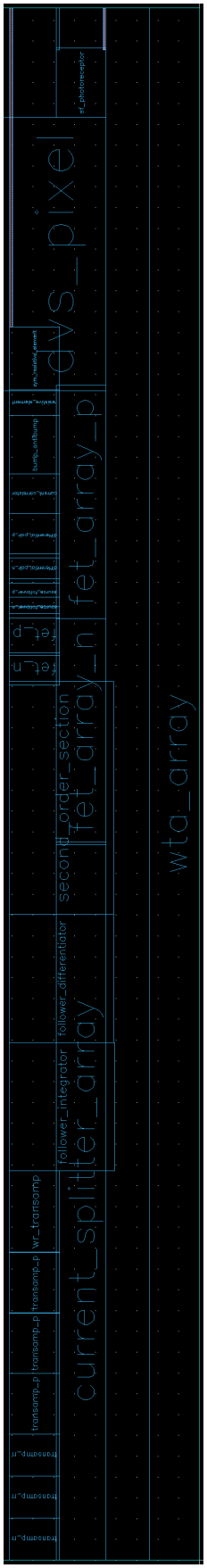


Figure 18: Core outline layout. (incomplete)

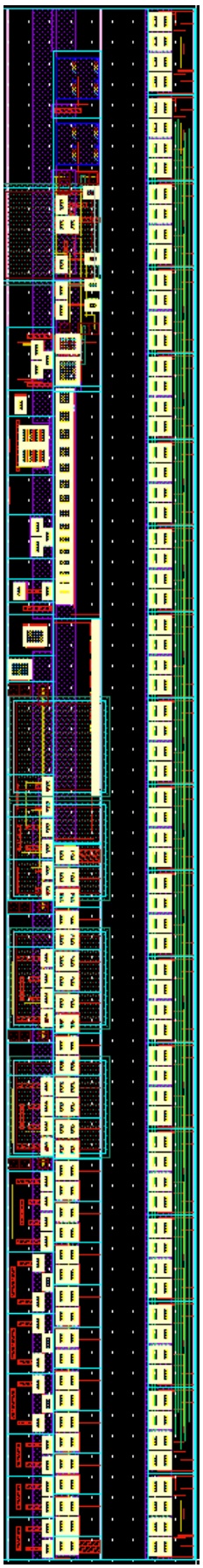


Figure 19: Core detailed layout. (incomplete)

C Bonding and Pin Assignment

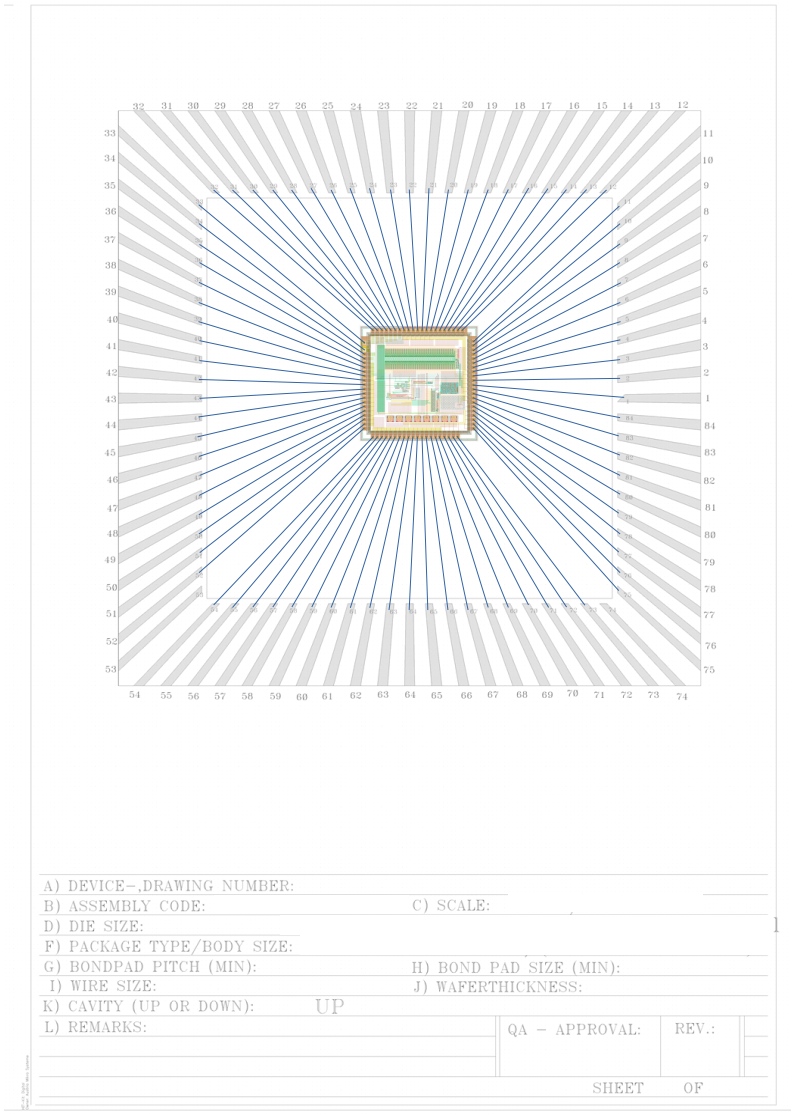


Figure 20: Bonding diagram for PLCC84 package.

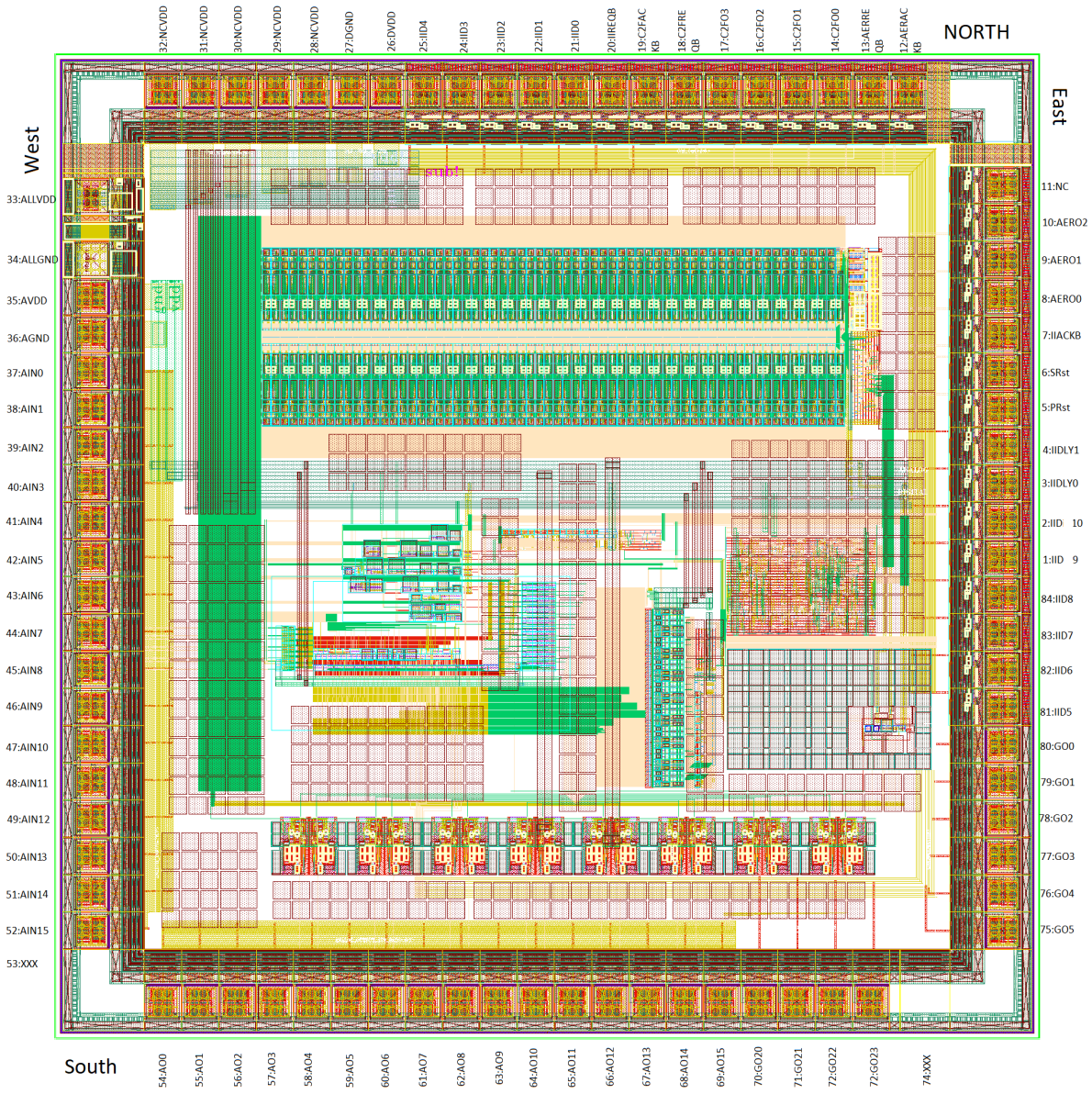


Figure 21: Chip layout with pad frame and pin assignments with signal names.