A 16-Bit Parallel Prefix Carry Look-Ahead Kogge-Stone Adder Implemented in Adiabatic Quantum-Flux-Parametron Logic

Tomoyuki TANAKA¹, Student Member, Christopher L. AYALA¹, and Nobuyuki YOSHIKAWA¹, Members

SUMMARY
Extremely energy-efficient logic devices are required for future low-power high-performance computing systems. Superconductor electronic technology has a number of energy-efficient logic families. Among them is the adiabatic quantum-flux-parametron (AQFP) logic family, which adiabatically switches the quantum-flux-parametron (QFP) circuit when it is excited by an AC power-clock. When compared to state-of-the-art CMOS technology, AQFP logic circuits have the advantage of relatively fast clock rates (5 GHz to 10 GHz) and 5 – 6 orders of magnitude reduction in energy before cooling overhead. We have been developing extremely energy-efficient computing processor components using the AQFP. The adder is the most basic computational unit and is important in the development of a processor. In this work, we designed and measured a 16-bit parallel prefix carry look-ahead Kogge-Stone adder (KSA). We fabricated the circuit using the AIST 10 kA/cm² High-speed STandard Process (HSTP). Due to a malfunction in the measurement system, we were not able to confirm the complete operation of the circuit at the low frequency of 100 kHz in liquid He, but we confirmed that the outputs that we did observe are correct for two types of tests: (1) critical tests and (2) 110 random input tests in total. The operation margin of the circuit is wide, and we did not observe any calculation errors during measurement.

key words: superconductor logic circuit, adiabatic quantum-flux-parametron, Kogge-Stone adder, superconductor electronics, digital circuits

1. Introduction

Most computers that support the information infrastructure in recent years are made from conventional CMOS integrated circuits. The process dimension of CMOS has reached several nanometers but it has been getting more difficult and expensive to scale further [1]. In order to develop the next generation of computing infrastructures to overcome the information processing demand of today, a new integrated circuit technology is required [2]. Based on this, we are focusing on extremely energy-efficient adiabatic quantum-flux-parametron (AQFP) logic [3], [4], and we are conducting various studies on how to systematically develop it for applications related to information technology [5]. The latest development towards this goal is a 4-bit AQFP microprocessor consisting of over 20,000 JJs [6], [7].

To continue to push the complexity of AQFP circuits, we focused on developing a 16-bit parallel prefix carry look-ahead Kogge-Stone adder (KSA) [8] in this work. The adder is one of the very basic components of a microprocessor, and it serves as a useful benchmark for evaluating and refining design strategies for new circuit technologies. In addition, this work is one of the largest superconductor-based parallel adders fabricated and demonstrated so far. This is an important milestone towards large-scale integration of AQFP circuits and towards superconductor-based computation using more practical data word sizes. In this paper, we report the design, fabrication, and demonstration of our 16-bit KSA implemented in AQFP logic.

2. Design of the 16-Bit KSA

The logic design of the 16-bit KSA is done by hand using an environment for AQFP semi-custom design [5]. The basic idea of the logic design is the same as the majority-based 8-bit KSA using majority-3 gates described in [9]. The 16-bit KSA block diagram is shown in Fig. 2 with each of the 16-bit inputs labeled at the top of the diagram. The area consisting of the black and gray blocks in the figure is the carry prefix tree which is the main component of the adder.
as described in [8], [9]. The depth of the logic in this region increases logarithmically, so it is faster than the ripple-carry adder.

The difference in the structure of the circuits between the previous study [9] and this study is the fabrication process and the clocking scheme. First, we switched the fabrication process from the STandard Process (STP) [10] to the High-speed STandard Process (HSTP) [11] of AIST. As a result, the critical current density was increased from 2.5 kA/cm² to 10 kA/cm² and the capacitance of the Josephson junction (JJ) has been reduced. This allowed us to remove the shunt resistor from the JJ. This change reduced the overall area of the cell by two-thirds, and it allowed us to fit more logic in a given footprint. Further by using unshunted JJs, we also reduced the switching energy of the logic cells compared to the shunted version (STP) by about an order of magnitude.

We also changed the structure of the power-clock network to excite the AQFP from 3 phases to 4 phases. The 3-phase power-clock network used three separate AC currents with a relative phase difference of 120° between them. The 4-phase network uses two AC currents (AC1 and AC2) that differ by 90° coupled to a single DC offset which enables 4 different phases to be generated from the two AC currents. Implementation using the 4-phase network becomes a little bit more complex compared to the previous 3-phase approach as we have to pay careful attention to relative directions of the AC and DC currents as they meander throughout the circuit to ensure that the appropriate clock phase is generated correctly [11]. Nonetheless, the 4-phase power-clock network allows data to propagate through 4-stages of logic per cycle instead of 3 in the prior work, and the timing windows also becomes wider thanks to the larger sampling overlap that exists between adjacent clock phases [5]. These benefits are important as we try to scale AQFP circuits to larger complexities.

With the logic netlist prepared, we then performed the automated place and route of the KSA using the genetic algorithm (GA) and channel-based routing [12], [13]. The reason for using GA was to try to solve the constraints of signal propagation in AQFP circuits by finding a quality placement of cells such that the interconnect length between adjacent logic rows do not surpass the driving length constraint. The AQFP signal line propagates the logic state using the current flowing through a superconductor ring of the AQFP. When the inductance of the superconductor ring becomes large, it attenuates the amplitude of the loop current. When the amplitude of the input data signal becomes too small such that it falls within the gray zone of the receiving AQFP, the resulting output is produced stochastically, and would result in bit errors. The exact propagation limit is currently under thorough investigation, but we assume a soft limit of approximately 0.8 mm via preliminary simulation results. In order to propagate signals over even longer distances, it is necessary to insert buffers at appropriate distances to amplify the signals once again. However, the insertion of buffers increases the area, energy and latency of the circuit. The problem is further compounded by the fact that the inserted buffers are also clocked, meaning that even if a single net between two adjacent logic rows is too long and requires a buffer insertion, all other nets between these two logic rows must also be buffered to maintain data synchronization from one phase to the next phase. The insertion of this extra row of buffers is expensive in terms of area, latency, and energy. In manual design, this insertion also introduces complications in the subsequent rows in which the designer may have already assigned clock phases to. If an extra clock phase has to be inserted in the middle of a circuit, the designer may have to redesign the clock network for the subsequent logic rows. Thus, it is important to search for a sufficient cell placement result that does not require the addition of buffers.

Since it is very difficult and time-consuming to do this work by hand, we opted to use our GA-based automatic placement tool which can insert buffers and reconcile the clock network when necessary. The most critical routing section of a conventionally placed KSA is in the final stage of the carry prefix tree where it is necessary for the interconnect to span over half of the bit slices of the adder to perform the calculation. The 16-bit KSA has roughly a horizontal distance of 1.3 mm spanning across 8 bit slices, and the actual wiring will be longer than this when considering the vertical distance. Therefore, when the adder scales to 16-bit or higher, it is necessary to insert buffers for signal amplification. This problem was not an issue in the design of the 8-bit KSA which had a worse-case horizontal interconnect distance of 0.75 mm across 4 bit slices.

To make the circuit more robust and to relax the constraints of the GA placer, we inserted an additional buffer after every logic stage to act as a driver for the next stage. This eliminated the need for the GA to handle gate-dependent driving strengths and ensured all signals from one logic stage to the next are transmitted with good amplification at the cost of doubling the latency of the entire circuit. This design guideline for the GA placer certainly impacts latency.
When compared to a handmade design which carefully considered the gate-dependent driving strengths of each cell, there is a 27% reduction in latency which is the tradeoff in using automated place and route tools in this work. The microphotograph of the resulting 16-bit KSA chip is shown in Fig. 1. The circuit area of the adder is 2.8 mm × 3.6 mm, and it consists of nearly 5000 JJs.

3. Measurement Result

Using a low-frequency immersion probe, the KSA circuit was placed in liquid He and measured. Due to poor contact between the measurement system at room temperature and the custom-made terminal box connecting to the test probe, it was difficult to observe the 17 readout signals simultaneously. So a total of nine outputs were checked: bits 0 to 3, bits 12 to 15, and the carry bit. We tried two sets of tests: critical tests and random tests. The critical tests were five test vectors that we explicitly defined to demonstrate pass-through of propagate signals, generation of carry from every bit, and full-propagation along the carry chain from the least significant bit (LSB) to the most significant bit (MSB). If the carry output is correct in the latter case, it is a very good indication that the prefix signals are properly propagating along the long interconnects between stages, and that the carry prefix tree is working properly. Table 1 lists the critical test vectors we applied and the expected result. We then proceeded to apply a set of random vectors for a total of 110 random additions, as shown in Table 2.

Figure 3 shows the results of the critical tests and 20 random tests, and Fig. 4 shows the results of 90 additional random tests at 100 kHz. The dc-SQUID-based output interface of the KSA produces a unipolar return-to-zero signal. When the output is a logic ‘1’, the output signal rises and then falls back to zero proportional to the applied clock period of the AC clock. When the output is a logic ‘0’, the output signal remains low. We confirmed that both the critical tests and the random tests passed. Even though our experiment was not fully exhaustive, these successful tests provide a strong indication that our design is working correctly.

Some of the outputs were rather noisy with S14 in particular being very unstable. We attribute this to the measurement equipment condition at the time of this writing. Furthermore, we measured the operating margins of the circuit by adjusting how much we can change the amplitudes of AC1 and AC2 before the circuit malfunctions. The operating range is −4.29 dBm to 0.90 dBm for AC1, and −3.41 dBm to 1.02 dBm for AC2. Both ranges are sufficiently wide.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0x0044A</td>
<td>0x0075B</td>
<td>0x01A5A</td>
</tr>
<tr>
<td>2</td>
<td>0x0349A</td>
<td>0x08D8B</td>
<td>0x0F248</td>
</tr>
<tr>
<td>3</td>
<td>0x32911</td>
<td>0x055B9</td>
<td>0x38411</td>
</tr>
<tr>
<td>4</td>
<td>0x0A9FC</td>
<td>0x234D3</td>
<td>0x02E9</td>
</tr>
<tr>
<td>5</td>
<td>0x06C5C</td>
<td>0x75EB3</td>
<td>0x1557F</td>
</tr>
<tr>
<td>6</td>
<td>0x03EAD</td>
<td>0x128A8</td>
<td>0x98E4</td>
</tr>
<tr>
<td>7</td>
<td>0x085FA</td>
<td>0x0F66E</td>
<td>0x17E60</td>
</tr>
<tr>
<td>8</td>
<td>0x076A8</td>
<td>0x08659</td>
<td>0x0F6FD</td>
</tr>
<tr>
<td>9</td>
<td>0x0CA13</td>
<td>0x3339D</td>
<td>0xF2DA</td>
</tr>
<tr>
<td>10</td>
<td>0x0E972</td>
<td>0x75A9F</td>
<td>0x15F1B</td>
</tr>
<tr>
<td>11</td>
<td>0x06B60</td>
<td>0x71070</td>
<td>0x0DCC</td>
</tr>
<tr>
<td>12</td>
<td>0x04F84</td>
<td>0x0C01C</td>
<td>0x16F9A</td>
</tr>
<tr>
<td>13</td>
<td>0x02A26</td>
<td>0x3A5E0</td>
<td>0x0DD4</td>
</tr>
<tr>
<td>14</td>
<td>0x08C19</td>
<td>0x199D4</td>
<td>0x0A5ED</td>
</tr>
<tr>
<td>15</td>
<td>0x01B9A</td>
<td>0x87944</td>
<td>0x944AE</td>
</tr>
<tr>
<td>16</td>
<td>0x0B3SC</td>
<td>0x0CE35</td>
<td>0x18231</td>
</tr>
<tr>
<td>17</td>
<td>0x060C3</td>
<td>0x0B2B6</td>
<td>0x08BE3</td>
</tr>
<tr>
<td>18</td>
<td>0x04FEC</td>
<td>0x04CE9</td>
<td>0x9C5D</td>
</tr>
<tr>
<td>19</td>
<td>0x06661</td>
<td>0x013C2</td>
<td>0x07A23</td>
</tr>
<tr>
<td>20</td>
<td>0x0B11F</td>
<td>0x0FF4A</td>
<td>0x18B69</td>
</tr>
<tr>
<td>21</td>
<td>0x0B380</td>
<td>0x068BF</td>
<td>0x11F78</td>
</tr>
<tr>
<td>22</td>
<td>0x07C9D</td>
<td>0x38E3B</td>
<td>0x0FAB8</td>
</tr>
<tr>
<td>23</td>
<td>0x0939A</td>
<td>0x3856B</td>
<td>0x14C59</td>
</tr>
<tr>
<td>24</td>
<td>0x02B01</td>
<td>0x0A72C</td>
<td>0x94C78</td>
</tr>
<tr>
<td>25</td>
<td>0x02D0E</td>
<td>0x38576</td>
<td>0x06662</td>
</tr>
<tr>
<td>26</td>
<td>0x06D5B</td>
<td>0x0B819</td>
<td>0x12574</td>
</tr>
<tr>
<td>27</td>
<td>0x07F5D</td>
<td>0x0258A</td>
<td>0x11B95</td>
</tr>
<tr>
<td>28</td>
<td>0x09F5E</td>
<td>0x0F771</td>
<td>0x19756</td>
</tr>
<tr>
<td>29</td>
<td>0x0E74F</td>
<td>0x4C905</td>
<td>0x1A7EC</td>
</tr>
<tr>
<td>30</td>
<td>0x02FA2</td>
<td>0x0992B</td>
<td>0x18FCD</td>
</tr>
<tr>
<td>31</td>
<td>0x06203</td>
<td>0x5797F</td>
<td>0x089CF</td>
</tr>
<tr>
<td>32</td>
<td>0x00974</td>
<td>0x02029</td>
<td>0x0B8D7</td>
</tr>
<tr>
<td>33</td>
<td>0x0C993</td>
<td>0x85F24</td>
<td>0x12B87</td>
</tr>
<tr>
<td>34</td>
<td>0x0AEB9</td>
<td>0x0882D</td>
<td>0x195A6</td>
</tr>
<tr>
<td>35</td>
<td>0x08E74</td>
<td>0x028F6</td>
<td>0x166FA</td>
</tr>
<tr>
<td>36</td>
<td>0x082E9</td>
<td>0x38309</td>
<td>0x165F2</td>
</tr>
<tr>
<td>37</td>
<td>0x063A3</td>
<td>0x04485</td>
<td>0x06858</td>
</tr>
<tr>
<td>38</td>
<td>0x01895</td>
<td>0x0F683</td>
<td>0x16F18</td>
</tr>
<tr>
<td>39</td>
<td>0x03590</td>
<td>0x0C2B8</td>
<td>0x0A84E</td>
</tr>
<tr>
<td>40</td>
<td>0x069C9</td>
<td>0x06D0D</td>
<td>0x06D6</td>
</tr>
<tr>
<td>41</td>
<td>0x0267C</td>
<td>0x0B668</td>
<td>0x006E7</td>
</tr>
<tr>
<td>42</td>
<td>0x0C662</td>
<td>0x59398</td>
<td>0x1169A</td>
</tr>
<tr>
<td>43</td>
<td>0x0D329</td>
<td>0x06D48</td>
<td>0x1469B</td>
</tr>
<tr>
<td>44</td>
<td>0x0F5B9</td>
<td>0x06668</td>
<td>0x0FBE</td>
</tr>
<tr>
<td>45</td>
<td>0x0513A</td>
<td>0x68287</td>
<td>0x08C61</td>
</tr>
<tr>
<td>46</td>
<td>0x0D07A</td>
<td>0x0F8D7</td>
<td>0x1D951</td>
</tr>
<tr>
<td>47</td>
<td>0x03FDC</td>
<td>0x0F748</td>
<td>0x05628</td>
</tr>
<tr>
<td>48</td>
<td>0x0F5B9</td>
<td>0x16610</td>
<td>0x180BA</td>
</tr>
<tr>
<td>49</td>
<td>0x03F8E</td>
<td>0x0E5A8</td>
<td>0x07C49</td>
</tr>
<tr>
<td>50</td>
<td>0x04741</td>
<td>0x0C2B8</td>
<td>0x0111E</td>
</tr>
<tr>
<td>51</td>
<td>0x05696</td>
<td>0x0A6A9</td>
<td>0x0C4AC</td>
</tr>
<tr>
<td>52</td>
<td>0x02D28</td>
<td>0x0284A</td>
<td>0x0192C</td>
</tr>
<tr>
<td>53</td>
<td>0x088B3</td>
<td>0x0F92F</td>
<td>0x018DA</td>
</tr>
<tr>
<td>54</td>
<td>0x03C5C</td>
<td>0x0E38C</td>
<td>0x0183D</td>
</tr>
<tr>
<td>55</td>
<td>0x041E8</td>
<td>0x0C088</td>
<td>0x117D3</td>
</tr>
</tbody>
</table>

The first 20 vectors in this list correspond to the random tests in Fig. 3, and the last 90 vectors correspond to Fig. 4.

4. Comparison with Different Technologies

We compare several parallel adders under various technologies including superconductor RSFQ/ERSFQ [14], RQL [15], and 90-nm adiabatic CMOS [16]. We decided to...
compare RSFQ/ERSFQ as it is the most widespread logic family in superconductor electronics. We include RQL as well even though the reported design is only 8-bit because it is an AC-biased logic just like AQFP, and it is also considered a very energy-efficient technology. Finally, we included the 90-nm adiabatic CMOS work reported in [16] because even though the designs have not been experimentally demonstrated, it is one of the more recent works using a 90-nm fabrication process to manufacture devices that operate adiabatically like AQFP. The area, number of Josephson
junctions (JJs) when applicable, bias magnitude, target operating frequency, and energy/op are compared in Table 3.

Firstly, the footprint area of the adder in this study is larger than that of the RSFQ implementation. Compared to RQL, the area of the AQFP circuit is also larger. The primary cause for this is the large output transformers of each AQFP. Section 5 briefly discusses how this can be improved. The adiabatic CMOS adder occupies even less area despite using a relatively old 90-nm CMOS technology. We expect this difference to grow when considering 7-nm FinFET technology [17].

In terms of the number of JJs, this study is almost half the number of junctions used in the RSFQ implementation. This may be due to the fact that RSFQ circuits are designed for very high-speed operation, so many JJs are inserted to adjust the delay of data propagation. Furthermore, the clock and reset paths of the wave-pipelined RSFQ adder also require active JJs, whereas the AQFP power-clock acts as both the synchronizing clock and reset. This power-clock is distributed through a meandering microstripline and thus requires no active JJs. RQL, like AQFP, distributes power through AC power-clocks along microstriplines, so it also has a low number of JJs. In our work, 3000 JJs are used for the amplification of signal currents, so there is potential to reduce the number of JJs to around the levels of RQL through more intelligent cell placement and interconnect routing.

The target operating frequency of RSFQ is higher than our AQFP implementation as we need to operate our circuit at relatively low clock rates to remain in the adiabatic regime. Despite this, the AQFP adder is still faster than adiabatic CMOS. When the operating frequency of the adiabatic CMOS adder is increased, the dynamic switching energy becomes more dominant compared to the static power consumption. This power becomes more prevalent in the adiabatic operation region as reported in [16]. Considering that our design is more energy-efficient and still operates at a practical clock rate for high-performance computing, this makes AQFP logic a good candidate for building the next generation of low-power supercomputers and data centers.

5. Scaling of the AQFP KSA

We estimate how the adder scales in terms of area and latency when we increase the data word size as shown in Fig. 5. Based on these estimates, we discuss where AQFP can use improvement and how such improvements can be carried out. For the estimation, we assumed a simple structure where the bit slices of KSAs are connected directly below each other, and we also considered the insertion of buffers for long-distance wiring. For example, in the final stage of a 128-bit KSA, the data travels a distance of 64-bit slices. In this case, at least 13 levels buffer insertions will be

### Table 3 Comparison with other 16-bit parallel adders in literature.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>10.1 mm²</td>
<td>8.5 mm²</td>
<td>2.8 mm²</td>
<td>0.0048 mm²</td>
</tr>
<tr>
<td>Complexity</td>
<td>4976 JJs</td>
<td>9941 JJs</td>
<td>815 JJs</td>
<td>972 Trs</td>
</tr>
<tr>
<td>Bias</td>
<td>AC 3.0 mA</td>
<td>DC 1.61 A</td>
<td>AC 0.9 mA</td>
<td>DC 1 V</td>
</tr>
<tr>
<td>Target Freq.</td>
<td>5 GHz</td>
<td>30 GHz</td>
<td>10 GHz</td>
<td>0.5 GHz</td>
</tr>
<tr>
<td>Energy/op</td>
<td>6.97 $\text{fJ/mm}^2$</td>
<td>3.13 $\text{pJ/mm}^2$</td>
<td>90.2 $\text{nJ/mm}^2$</td>
<td>182 $\text{fJ/mm}^2$</td>
</tr>
</tbody>
</table>

* Includes 1000 W/W$_{4,2K}$ cooling efficiency.
* Design was originally in RSFQ but we assumed ERSSFQ biasing.
* Note that this is an 8-bit RQL adder, not 16-bit.
* Extrapolated from energy per transistor of shift register in [16].

![Fig. 5](image-url) Word size scaling of the AQFP KSA for area and latency.
required. To estimate the area, we calculated the ratio of the area used by the gate to the area used by the internal wiring and excitation lines for the 16-bit KSA we designed in this work, and estimated the area based on that. The number of buffer insertions increases in proportion to the square of the number of bits in the adder. Therefore, the slope of the curve with respect to area becomes a little larger. Compared to the previous study, the scalability is slightly improved because the gate area is smaller. From these estimates, it was found that, for example, a 64-bit KSA circuit can be implemented on a 1 cm × 1 cm chip. Compared to the CMOS circuits that are widely used today, this area is very large, and efforts are needed to reduce the area. One approach is using a directly coupled quantum-flux-parametron (DQFP) [21], [22] where the large output transformer is completely removed, and logic gates connect directly to each other. This can reduce the cell size in half but unlike conventional AQFP cells, the DQFP buffer and inverter have different core structures, introducing complexities in cell library development. Another approach is to use a more advanced process with more layers [23]–[25]. With more layers, the transformer and the SQUID can be stacked vertically, roughly halving the footprint of the logic cell. Interconnections of the AQFP can also be stacked, potentially reducing the area further. The DQFP approach can also benefit from more advanced processes since the directly coupled inductor structures can be implemented using dedicated kinetic inductance layers such as those in the MIT Lincoln Laboratory SFQSee process [26], which opens more opportunities for further scaling.

The increase in latency for KSA should be logarithmic, but according to our estimates, it is increasing a little bit faster than expected. This is due to our conservative design approach of adding buffers after each logic stage in addition to buffer insertions for long interconnects. In any case, the absolute latency is quite high. We can improve the latency by reducing the phase difference of the excitation current or, in other words, adding more clock phases within the same target clock period. This can be achieved through power clock dividers or delay line clocking [27]–[29]. The latency improvement between these techniques is shown in the lower sub-plot of Fig. 5. In the present 4-phase design, the phase difference is 90°, but by using the aforementioned methods, the phase difference can be reduced up to 1/5 of the present design. This means more clock phases become available per cycle and thus data can propagate through more stages of logic in a cycle. Furthermore, we still use the same clock frequency in these methods so the switching energy is unchanged. Lastly, the current design uses only 3-input majority gates, but by using 5-input majority gates, the latency of the prefix carry tree can be reduced by half [9], [30], [31].

6. Conclusion

We have designed and fabricated the first 16-bit AQFP adder chip. Due to a malfunction in the measurement system, we were not able to confirm the complete operation of the circuit, but it worked properly for a total of 115 validated test vectors at the low frequency of 100 kHz in liquid He. The operation range of the AC1 and AC2 excitation currents are −4.29 dBm to 0.90 dBm and −3.41 dBm to 1.02 dBm, respectively, and the circuit consists of just under 5000 JJJs. We briefly discussed a few ways in which the latency and area of the circuit can be improved as we scale to larger adders. With this demonstration, we are moving closer towards functionally meaningful AQFP circuits operating on more practical data word sizes.

Acknowledgments

The present study was supported by JSPS KAKENHI (Grants No. 19H05614). CAD and EDA support was provided by the VLSI Design and Education Center (VDEC) of the University of Tokyo in collaboration with Cadence Design Systems, Inc.

The circuits were fabricated in the Clean Room for Analog-digital superconductivity (CRAVITY) of the National Institute of Advanced Industrial Science and Technology (AIST) using the high-speed standard process (HSTP).

References


**Tomoyuki Tanaka** received the B.E. degrees in 2018 and the M.E. degrees in 2020 from Yokohama National University. His research interests include superconductor computing devices and computer architecture. Mr. Tanaka is a student member of the Institute of Electronics, Information and Communication Engineers of Japan, Cryogenics and Superconductivity Society of Japan.

**Christopher L. Ayala** received the combined B.E./M.S. degree in 2009 and the Ph.D. degree in 2012 from Stony Brook University, Stony Brook, New York, USA, all in electrical and computer engineering. From 2013 to 2015, he was a Post-Doctoral Fellow with IBM Research, Zurich, Switzerland. Since 2015, he has been with the Institute of Advanced Sciences, Yokohama National University, Yokohama, Japan, where he is currently an Associate Professor. His research interests include emerging circuit technologies, superconductor logic, NEMS-MEMS, novel computer architectures, and electronic design automation (EDA). Dr. Ayala is a member of the Institute of Electrical and Electronics Engineers, the Japan Society of Applied Physics, the Institute of Electronics, Information and Communication Engineers of Japan, Eta Kappa Nu Honor Society, and the Tau Beta Pi Engineering Honor Society.

**Nobuyuki Yoshikawa** received the B.E., M.E., and Ph.D. degrees in electrical and computer engineering from Yokohama National University, Japan, in 1984, 1986, and 1989, respectively. Since 1989, he has been with the Department of Electrical and Computer Engineering, Yokohama National University, where he is currently a Professor. His research interests include superconductive devices and their applications in digital and analog circuits. He is also interested in single-flux-quantum circuits, quantum computing devices and cryo-CMOS devices. Prof. Yoshikawa is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Japan Society of Applied Physics, the Institute of Electrical Engineering of Japan, Cryogenics and Superconductivity Society of Japan, and Institute of Electrical and Electronics Engineers.