

Thermal Exploration of RSFQ Integrated Circuits

Ana Mitrovic¹ and Eby G. Friedman², *Life Fellow, IEEE*

Abstract—The maturing of rapid single flux quantum (RSFQ) circuits into a VLSI complexity technology has focused the need for advanced design and analysis capabilities. For example, the temperature dependence of RSFQ circuits has created a need for a thermal analysis methodology, including an accurate thermal model and related partitioning algorithm. The Josephson critical current density and the superconductive properties of the interconnects become compromised at elevated temperatures. Heating of Josephson junctions (JJs) and niobium interconnects results in reduced margins and/or functional failure. A methodology for evaluating the thermal properties of RSFQ integrated circuits, targeting large-scale systems, is presented here. This methodology comprises a thermal model and a multistage partitioning algorithm. The algorithm, based on a layout of the IC, partitions the circuit into blocks. An average thermal model is applied to the partitioned structure, producing a netlist for thermal simulation. The hot spots are also determined with a threshold temperature indicating functional failure. A peak thermal model is presented to detect hotspots based on the threshold temperature. The model is evaluated at several block complexities and validated using a numerical solver. An error of less than 1% between the model and numerical simulations is achieved. The algorithm is applied to the AMD2901 benchmark circuit composed of more than 300 000 heating elements. The thermal profile for AMD2901 is generated in under 68 min.

Index Terms—Single flux quantum, superconductive digital electronics, superconductive integrated circuits, thermal analysis, thermal model.

I. INTRODUCTION

RAPID single flux quantum (RSFQ) technology was first introduced in 1985 and has since become the primary superconductive digital logic family [1], [2], [3], [4]. Niobium-based RSFQ circuits operate at 4.2 K—below the critical temperature of niobium (9.3 K). Recent advancements in RSFQ system design and manufacturing have yielded circuits that operate at frequencies close to 80 GHz [5], [6]. Device densities over 4.2×10^6 Josephson junctions (JJs) per cm^2 are supported by existing fabrication facilities [7], [8], [9].

The heat produced due to increasing frequencies and higher fabrication densities of JJs in cryogenic superconductive inte-

grated circuits has increased the on-chip ambient temperature. Elevated temperatures degrade the margins, increase the likelihood of functional failure due to the loss of superconductive properties, and create heat load issues that affect the refrigeration characteristics [10], [11], [12], [13]. Understanding the thermal behavior of these circuits and the effects of heat on circuit operation enhances the reliability of superconductive ICs. This insight enables the placement of sensitive circuitry to mitigate the effects of the thermal aggressors. Due to the sensitivity of RSFQ circuits to temperature, a methodology for exploring the thermal behavior of large-scale systems has become necessary. This methodology can be used to estimate the effects of heat on circuit behavior, avoid operational failure, and manage the heat transfer process [1], [13], [14].

The thermal behavior of niobium nitride (NbN) JJs placed directly on a substrate was examined using an analytic model in 1991 by Lavine and Bai [10]. Ohki et al. [15] assessed the heating properties of a bias resistor in a millikelvin superconductive structure in 2003. These models target small structures and have not been adapted to large-scale systems. In addition, RSFQ manufacturing technologies have changed since the development of these early models. Some notable modifications include the quantity and material composition of the metal layers, and characteristics of the JJs and resistors. In 2021, a tool for the thermal analysis of superconductive circuits based on a finite element method was developed by Venter [16], although computationally prohibitive for large-scale systems. The primary intention of this work is to provide a sufficiently accurate, computationally efficient compact thermal model applicable to large-scale RSFQ circuits.

An analytic thermal model of RSFQ circuit structures has previously been described in [13]. In this model, each heating element is represented as a node within a resistive mesh with a thermal resistance describing the path between each pair of heating elements. Applying this model to large-scale integrated circuits requires a high complexity mesh. In this article, a methodology for the thermal analysis of large-scale RSFQ ICs, comprising a partitioning algorithm and a thermal model, is presented. This model includes an average thermal model to determine the heat load and a peak thermal model to identify any potential loss of superconductive properties that might result in functional failure.

The methodology assumes a circuit is composed of blocks with a preset complexity, where each block is modeled as a node within a resistive mesh. The average and peak thermal models are applied to large-scale ICs based on a multistage partitioning algorithm for evaluating the thermal behavior. The algorithm receives a circuit layout as an input, evaluates the peak temperature of each block, and produces a netlist. A SPICE simulation of the netlist is used to evaluate the

Manuscript received 4 July 2023; revised 13 November 2023; accepted 20 December 2023. Date of publication 10 January 2024; date of current version 22 March 2024. This work was supported in part by the National Science Foundation under Grant 2124453 (DISCOVER Expeditions) and Grant 2308863, in part by the Department of Energy under Funding Opportunity Announcement (FOA) Grant DE-FOA-0002950, in part by the Singapore Ministry of Education under Grant MOE2019-T2-2-075, and in part by a grant from Qualcomm Corporation. (Corresponding author: Ana Mitrovic.)

The authors are with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY 14627 USA (e-mail: ana.mitrovic@rochester.edu; friedman@ece.rochester.edu).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TVLSI.2023.3348452>.

Digital Object Identifier 10.1109/TVLSI.2023.3348452

1063-8210 © 2024 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See <https://www.ieee.org/publications/rights/index.html> for more information.

average temperature of each block. The methodology is calibrated for the MIT Lincoln Laboratory SFQ5ee fabrication technology [17], [18] and is validated using a 3-D numerical solver, COMSOL Multiphysics¹ [19], due to a lack of available experimental data.

This article is organized as follows. In Section II, a thermal model of large-scale RSFQ integrated circuits is presented. In Section III, a multistage partitioning algorithm for analyzing the thermal behavior of RSFQ ICs is proposed. Validation of the model and algorithm is described in Section IV. This article is concluded in Section V.

II. THERMAL MODEL

The heat dissipated within one region of a circuit can affect neighboring regions due to thermal conductance through a connecting medium [20], [21], [22], [23]. Those regions that dissipate heat are referred to here as aggressors. The victims are those regions affected by the dissipated heat. Each portion of a circuit can be viewed as both an aggressor and a victim or only as a victim if no heating elements exist.

The thermal model assumes a system of aggressors, victims, and heat paths, as portrayed in Fig. 1. Based on a thermal–electrical analogy, the model represents the heat flow rate as a current source, thermal resistance as an electrical resistance, and temperature as a voltage [21]. The material properties used in the model are summarized in [13]. The aggressors are described by the power dissipated within a block. The individual thermal resistance of a block is evaluated for both the aggressors and victims. This system is described in Section II-A. The heat paths are represented as a thermal resistance along a path, as described in Section II-B. A peak thermal model is described in Section II-C. A temperature threshold for evaluating hotspots is described in Section II-D.

A. Aggressor/Victim Model

The largest source of heat in RSFQ circuits is the static power dissipated by the resistive bias network, P_{bias} . Dynamic power is dissipated within the resistively shunted JJs. To produce a desirable current–voltage characteristic of the JJs for RSFQ applications, an external shunt resistance is added in parallel to the junction [24], [25], [26]. During a switching event, most of the power is dissipated within the shunt resistor P_{shunt} . A portion of the power P_{JJ} is dissipated within the JJ due to the resistance of the JJ caused by the breakup of Cooper pairs [1], [13]. To approximate a block as an aggressor, the power is estimated as the total power dissipated by all of the heating elements within the block

$$P_{\text{block}} = \Sigma P_{\text{JJ}} + \Sigma P_{\text{shunt}} + \Sigma P_{\text{bias}}. \quad (1)$$

The power dissipated by the bias resistors is estimated by the Joule heating

$$P_{\text{bias}} = V_{\text{bias}} I_{\text{bias}} \quad (2)$$

where V_{bias} is the voltage on the bias bus, and I_{bias} is the bias current, typically approximated as $\approx 0.7 I_c$ [27]. The power dissipated by the JJs and shunt resistors is described by the rms voltage V_{rms} across a resistively shunted junction [13].

¹Registered trademark.

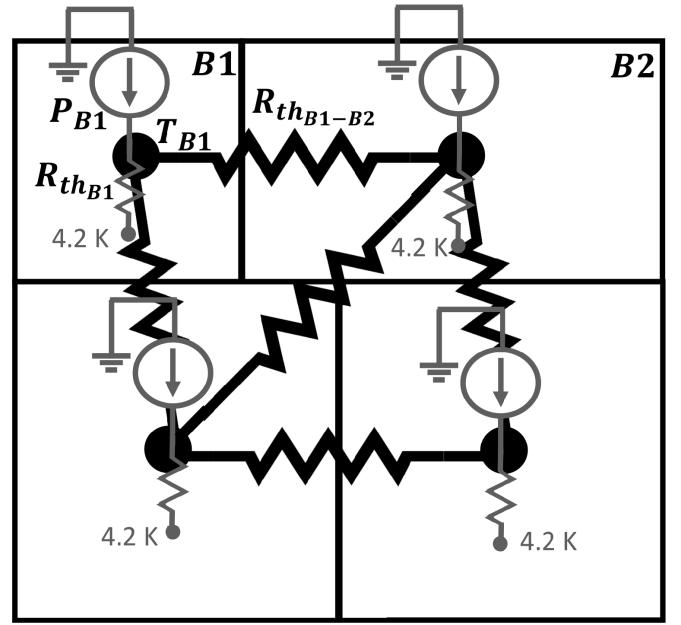


Fig. 1. Block-level thermal model of RSFQ ICs. P_{B1} represents the power produced by block B1, R_{thB1} is the thermal resistance of block B1, and $R_{thB1-B2}$ is the thermal resistance of the path between blocks B1 and B2. The temperature of block B1, T_{B1} , is evaluated as the voltage on node B1.

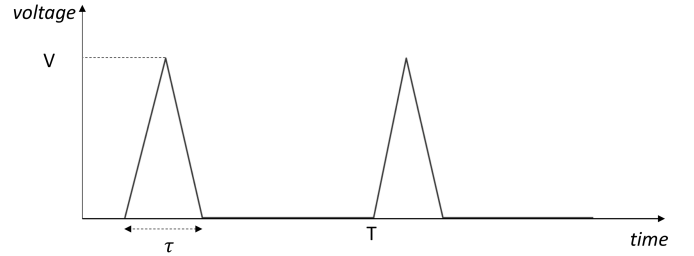


Fig. 2. Triangular wave approximation of an SFQ pulse.

To approximate V_{rms} , a triangular wave approximation is assumed [28], as shown in Fig. 2

$$V_{\text{rms}} = c_s \sqrt{\tau f \frac{V^2}{3}} \quad (3)$$

where c_s is the switching coefficient of a JJ, τ is the width of an SFQ pulse, f is the operating frequency, and V is the height of an SFQ pulse. For a resistively shunted junction, the pulse height can be approximated as $V \approx 2 I_c R_s$, leading to $\tau \approx \Phi_0 / I_c R_s$, where Φ_0 is a single flux quantum (≈ 2.07 mV·ps) [26], [29]. With these approximations of V and τ

$$V_{\text{rms}} = 2 c_s \sqrt{f \frac{\Phi_0 I_c R_s}{3}}. \quad (4)$$

Since $I_c R_s \approx V_c$

$$V_{\text{rms}} = 2 c_s \sqrt{f \frac{\Phi_0 V_c}{3}} \quad (5)$$

where V_c is the characteristic voltage corresponding to a critically damped JJ. For the SFQ5ee process, $V_c \approx 0.7$ mV [18].

Another important parameter of the thermal model is the individual thermal resistance of the block, R_{thblock} . The thermal

resistance for each block—aggressor or victim—is

$$R_{\text{thblock}} = \frac{1}{A_{\text{block}} h_{\text{eff}}} \quad (6)$$

where A_{block} is the area of the block, and h_{eff} is the effective heat transfer coefficient [13].

B. Model of the Thermal Path

To model heat conduction between blocks, the thermal resistance between adjacent blocks is introduced. The thermal resistance between two blocks, B1 and B2, either victims or aggressors, is

$$R_{\text{thB1-B2}} = \frac{1}{\kappa (d_{\text{heating_layers}})w} s_{B1-B2} \quad (7)$$

where s_{B1-B2} is the physical distance between the center of the blocks. $d_{\text{heating_layers}}$ is the thickness of the material layers where the heating elements (JJs and resistors) are located. For the MIT LL SFQ5ee fabrication technology, these layers are oxide layer 5 (O5) and metal layer 5 (M5). w is the width of the boundary between two blocks. κ is the equivalent thermal conductivity of the material conducting heat between the blocks. This thermal conductivity is approximated by the relative portion of the niobium and SiO_2 (for example, within material layers M5 and O5).

The thermal resistance between two adjacent blocks, where B1 is an aggressor block and B2 is a victim block, is

$$R_{\text{thB1-B2}} = \frac{1}{\kappa (d_{\text{heating_layers}})w} s_{\text{elB1-B2}} \quad (8)$$

where $s_{\text{elB1-B2}}$ is the physical distance between the boundary of the nearest heating element within the aggressor block B1 and the center of the victim block B2.

C. Peak Thermal Model

The peak thermal model of the aggressor blocks is based on the thermal model of a primitive [13]. The increase in temperature of the largest heater within a block is

$$\Delta T_{\text{el}} = \frac{P_{\text{el}}}{h_{\text{eff,el}} \alpha_{\text{el}} A_{\text{el}}} \quad (9)$$

where α_{el} is a unitless multiplying coefficient that increases the effective area of the element that is cooled due to the lateral dispersion of heat [13].

The peak temperature of those victim blocks that do not contain heating elements originates from elements within neighboring aggressor blocks. The peak thermal model of these victim blocks is therefore based on the lateral conduction of heat dissipated by the heating elements. At the site of the highest temperature inside a victim block, the increase in temperature is [30], [31]

$$\Delta T(x) = \Delta T_{\text{el}} * e^{-\frac{x}{\eta}} \quad (10)$$

where ΔT_{el} is the increase in temperature directly adjacent to the heating element, x is the distance between the heating element and the edge of the victim block, and η is the

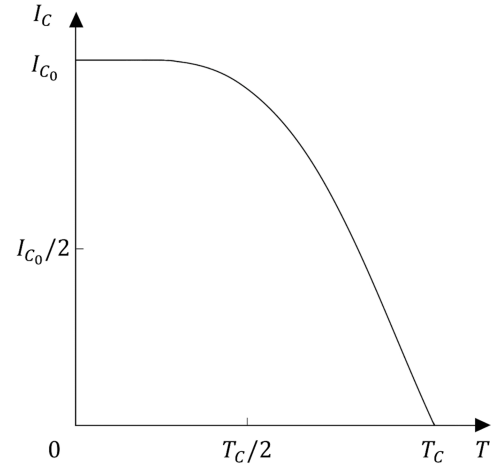


Fig. 3. Temperature dependence of the critical current of a JJ [10], [13].

thermal healing length quantifying the decrease in lateral temperature [31], [32], [33]

$$\eta = \sqrt{\frac{\kappa \cdot d_{\text{layer}}}{h_{\text{eff}}}}. \quad (11)$$

D. Model of Hot Spot

An increase in temperature above the critical temperature T_c indicates the loss of the superconductive properties of the material, leading to improper circuit operation. Compromised operation due to margin shrinkage, however, can occur at temperatures lower than T_c due to the temperature dependence of the Josephson critical current I_c [10]. The JJs in RSFQ circuits are most commonly biased to approximately 0.7 times the critical current [1], [17]. Any increase in temperature would lead to a decrease in I_c , as shown in Fig. 3, reducing the bias current margins. If the Josephson critical current decreases below the bias current due to heating within the RSFQ ICs, unwanted switching of the JJs will occur.

To determine the temperature at which I_c becomes less than I_{bias} , from BCS theory [29], [34]

$$I_c(T) = \frac{\pi \Delta(T)}{2eR_n} \tanh\left[\frac{\Delta(T)}{2k_B T}\right] \quad (12)$$

where Δ is the superconductive energy gap, R_n is the normal resistance of the JJ, and k_B is the Boltzmann constant (1.38×10^{-23} J/K).

Closer to T_c , $\Delta(T)$ is described in [35] by

$$\Delta(T) = k \cdot \Delta(0) \sqrt{1 - \frac{T}{T_c}}. \quad (13)$$

Applying the gap voltage expression [17]

$$V_g = \frac{2\Delta}{e}. \quad (14)$$

The coefficient of proportionality k from (13) is

$$k = \frac{V_g e}{2\Delta(0)} \frac{1}{\sqrt{1 - \frac{T}{T_c}}} \quad (15)$$

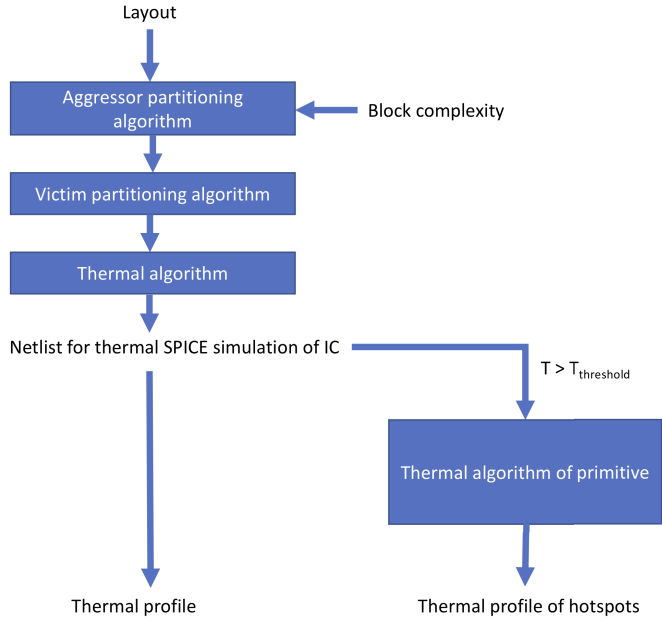


Fig. 4. Flow diagram of the multistage thermal analysis algorithm for large-scale RSFQ systems.

where k is evaluated at 4.2 K. A typical gap voltage V_g at 4.2 K is 2.6 mV [36]. Substituting (15) and (13) into (12)

$$I_c(T) = K_1 \sqrt{1 - \frac{T}{T_c}} \tanh \left[\frac{K_2 \sqrt{1 - \frac{T}{T_c}}}{T} \right] \quad (16)$$

where

$$K_1 = \frac{V_g \pi}{4R_n \sqrt{1 - \frac{4.2}{T_c}}} \quad (17)$$

and

$$K_2 = \frac{V_g e}{4k_B \sqrt{1 - \frac{4.2}{T_c}}}. \quad (18)$$

Assuming an average radius of a JJ is 1 μm , from (16), the temperature threshold $T_{\text{threshold}}$ for identifying hotspots is 5.7 K.

III. MULTISTAGE PARTITIONING ALGORITHM FOR THERMAL ANALYSIS OF SFQ ICs

The objective of the algorithm is to produce a netlist to support the thermal simulation of an RSFQ IC and to detect hotspots based on a threshold temperature. The algorithm produces a thermal profile of an IC and a thermal profile of the hotspots. A flow diagram of the multistage algorithm is shown in Fig. 4.

The aggressor partitioning stage of the multistage algorithm combines the heating elements into blocks. This stage of the algorithm is described in Section III-A. The victim partitioning stage of the algorithm, described in Section III-B, places the remaining circuit elements within the blocks. The average thermal algorithm, described in Section III-C, produces a netlist for thermal simulation using SPICE. The algorithm used to evaluate the peak temperature is described in Section III-D.

A. Aggressor Partitioning Algorithm

The temperature is greatest in the area surrounding the high power (heat) generating elements. To prevent information from being lost by averaging the temperature across areas where heat is minimally dissipated, it is crucial to evaluate the temperature close to the heating elements. The primary objective of the aggressor partitioning stage of the algorithm is therefore to form aggressor blocks by partitioning the system into blocks composed of those heating elements near each other, as described by Algorithm 1. No physical movement of the blocks occurs; the blocks are grouped together based only on the thermal properties of the component elements. The circuit layout is described as a weighted undirected graph G_a with the heating elements forming a set of graph vertices U_a . E_a is the set of edges connecting each pair of vertices. Each edge is characterized by a weight representing the Euclidean distance between nodes. The resulting graph is

$$G_a = (U_a, E_a, w : E_a \rightarrow \mathbb{R}) \quad (19)$$

$$E_a = \{\{m, n\} \in U_a^2 | m \neq n\} \quad (20)$$

$$w(m, n) = \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2} \quad (21)$$

with a total number of $(1/2)|U_a|(|U_a| - 1)$ edge weights [37].

During the aggressor partitioning stage, heating elements within a specific radius are clustered. A smaller radius produces a smaller block and provides local information about the average or peak temperature, while increasing the granularity of the resistor mesh for SPICE simulation. Once G_a is formulated, a parameter describing the minimum radius of an element r_{el} is used to characterize the complexity of a block, where subgraph $G_a^S \in G_a$ ensures that each pair of vertices $u_i, u_j \in U_a^S$

$$w(\{u_i, u_j\}) < r_{\text{el}}. \quad (22)$$

After the initial subgraph is formed, the algorithm ensures that the block borders corresponding to the subgraph do not intersect other vertices $u \in U_a \setminus U_a^S$. An example of this process is depicted in Fig. 5. Those vertices at a distance smaller than r_{el} are depicted in Fig. 5(a). Partitioning the layout according to the coordinates of these elements may intersect another heating element, as shown in Fig. 5(b). This element is incorporated within U_a^S , resulting in the aggressor block shown in Fig. 5(c).

A similar fault can occur with the borders of different aggressor blocks. The next step of the algorithm splits subgraph G_a^S into a smaller graph $G_a^{S-\text{new}}$ such that the border of the corresponding block does not intersect any of the aggressor blocks within existing subgraphs G_a^{S-e} . An example of this process is depicted in Fig. 6 with two intersecting blocks, as shown in Fig. 6(a). As described in Algorithm 1, the elements to the right of the intersected block form new subgraph vertices $U_a^{S-\text{new}}$, replacing U_a^S . As a result, none of the block borders intersects, as shown in Fig. 6(b). The vertices of each formed subgraph are stored in a set of vertices F_E . The aggressor partitioning process terminates when F_E contains all of the vertices U_a . The computational complexity of the aggressor partitioning stage of the algorithm is $O(|U_a|^2)$. An example of the boundary of the heating blocks produced by

Algorithm 1 Given Input File inputfile With Heating Element Coordinates, Generate Graph G_a Describing the Position of the Heating Elements Within the IC. Given the Minimum Radius of an Element r_{el} as a Block Complexity Parameter, Generate Subgraphs (Aggressor Blocks) $G_a^S = (U_a^S \in U_a, E_a^S \in E_a)$, Comprising Those Elements Closer Than r_{el} . Store the Coordinates of the Aggressor Blocks in x_{group} and y_{group}

Function: AGGRESSOR_PARTITION
Input : $inputfile, r_{el}$
Output : x_{group}, y_{group}

```

1  $X, Y, Xlen, Ylen, U_a \leftarrow read(inputfile);$ 
2  $E_a \leftarrow U_a^S;$ 
3  $w(m, n) \leftarrow \sqrt{(x_m - x_n)^2 + (y_m - y_n)^2};$ 
4  $G_a \leftarrow (U_a, E_a, w : E_a \rightarrow \mathbb{R});$ 
5  $F_E \leftarrow \emptyset;$ 
6 while  $F_E < |U_a|$  do
7    $U_a^S \leftarrow \emptyset;$ 
8    $u_i \leftarrow \text{first } u \in U_a, u \notin F_E;$ 
9    $U_a^S \leftarrow u_i;$ 
10  for each  $u_j \in U_a, u_j \notin F_E$  do
11    if  $w(\{u_i, u_j\}) < r_{el}$  then
12       $U_a^S \leftarrow u_j;$ 
13    end
14  end
15  while  $size(U_a^S)$  increasing do
16    for each  $u_k \in U_a^S$  and  $u_j \in U_a \setminus U_a^S, u_j \notin F_E$  do
17      if  $w(\{u_k, u_j\}) < r_{el}$  then
18         $U_a^S \leftarrow u_j;$ 
19      end
20    end
21  end
22   $x_{min}, y_{min} \leftarrow \min(X(U_a^S)), \min(Y(U_a^S));$ 
23   $x_{max}, y_{max} \leftarrow \max((X + Xlen)(U_a^S)), \max((Y + Ylen)(U_a^S));$ 
24  for each  $u_j \in U_a \setminus U_a^S, u_j \notin F_E$  do
25    if  $coord(u_j)$  intersected by  $x_{min}, y_{min}, x_{max}$  or  $y_{max}$  then
26       $U_a^S \leftarrow u_j;$ 
27    end
28  end
29  for each  $G_a^{S-e}$  do
30    if  $coord(G_a^{S-e})$  intersected by  $x_{min}, y_{min}, x_{max}$  or  $y_{max}$  then
31       $U_a^{S-new} \leftarrow \emptyset;$ 
32      if  $u \in U_a^S, u$  to the right of  $U_a^{S-e}$  then
33         $U_a^{S-new} \leftarrow u;$ 
34      else if  $u \in U_a^S, u$  to the left of  $U_a^{S-e}$  then
35         $U_a^{S-new} \leftarrow u;$ 
36        repeat for above;
37        repeat for below;
38      end
39       $U_a^S = U_a^{S-new};$ 
40    end
41  end
42   $x_{min}, y_{min} \leftarrow \min(X(U_a^S)), \min(Y(U_a^S));$ 
43   $x_{max}, y_{max} \leftarrow \max((X + Xlen)(U_a^S)), \max((Y + Ylen)(U_a^S));$ 
44   $x_{group} \leftarrow (x_{min}, x_{max});$ 
45   $y_{group} \leftarrow (y_{min}, y_{max});$ 
46   $F_E \leftarrow U_a^S;$ 
47 end
```

the aggressor partitioning stage of the algorithm is illustrated in Fig. 7(a).

Algorithm 2 Given the Coordinates of Aggressor Blocks x_{group} and y_{group} From the Aggressor Partitioning Stage of the Algorithm, Divide the Remaining Portion of the Circuit Into Blocks and Provide x_{part} and y_{part} Block Coordinates

Function: VICTIM_PARTITION

Input : x_{group}, y_{group}

Output : x_{part}, y_{part}

```

1  $x_{group}, y_{group} \leftarrow AGGRESSOR\_PARTITION(inputfile, r_{el});$ 
2  $\Gamma, L_X, L_Y \leftarrow x_{group}, y_{group};$ 
3  $Q = \{q_1, \dots, q_k\}, k \leftarrow size(x_{group})/2;$ 
4 while  $size(\Gamma) > 0$  do
5    $x_0 \leftarrow \min(D(\Gamma));$ 
6    $y_0 \leftarrow \min(R(\Gamma), D(\Gamma) = x_0);$ 
7    $\Gamma \leftarrow \Gamma \setminus (x_0, y_0);$ 
8    $x_1 \leftarrow \min(D(\Gamma), D(\Gamma) > x_0);$ 
9    $y_1 \leftarrow \min(R(\Gamma), D(\Gamma) = x_1);$ 
10  if  $(y_1 - y_0) > (x_1 - x_0)$  then
11     $y_w \leftarrow y_1;$ 
12    for each boundary  $L$  in  $L_X$  do
13      if  $L$  to the right of  $x_0,$ 
14         $L$  overlaps  $((x_0, y_0), (x_0, y_1))$  then
15         $boundaries \leftarrow L$ 
16      end
17    end
18     $x_w \leftarrow \min(x_{q_{min}} \in boundaries);$ 
19  else
20     $x_w \leftarrow x_1;$ 
21    for each boundary  $L \in L_Y$  do
22      if  $L$  above  $y_0,$ 
23         $L$  overlaps  $((x_0, y_0), (x_1, y_0))$  then
24         $boundaries \leftarrow L$ 
25      end
26    end
27     $y_w \leftarrow \min(y_{q_{min}} \in boundaries);$ 
28  end
29   $x_{vict} \leftarrow x_0, x_w;$ 
30   $y_{vict} \leftarrow y_0, y_w;$ 
31   $\Gamma \leftarrow (x_0, y_w), (x_w, y_0), (x_w, y_w);$ 
32   $\Gamma \leftarrow \Gamma \setminus duplicate(x, y);$ 
33 end
34  $x_{part}, y_{part} \leftarrow x_{group} \cup x_{vict}, y_{group} \cup y_{vict};$ 
```

B. Victim Partitioning Algorithm

After grouping the heating elements, the remaining portion of the circuit is clustered into blocks. This process is performed within the victim partitioning stage of the algorithm, as depicted in Algorithm 2. The structure is represented by the set of vertices Γ formed from the coordinates of the heating blocks $(x_{min}, x_{max}) \in x_{group}$ and $(y_{min}, y_{max}) \in y_{group}$ produced by the aggressor partitioning algorithm. Γ is the set of ordered pairs (x, y) such that for each aggressor block $q \in Q$, the vertices in Γ are

$$(x_{q_{min}}, y_{q_{min}}) \quad (23a)$$

$$(x_{q_{min}}, y_{q_{max}}) \quad (23b)$$

$$(x_{q_{max}}, y_{q_{min}}) \quad (23c)$$

$$(x_{q_{max}}, y_{q_{max}}). \quad (23d)$$

The domain of the set $D(\Gamma)$ represents the x -coordinate of the vertices, whereas the range of the set $R(\Gamma)$ represents the y -coordinate of the vertices. Two sets, $L_X, L_Y \subseteq \Gamma$, denote the boundary of the partitioned regions along the

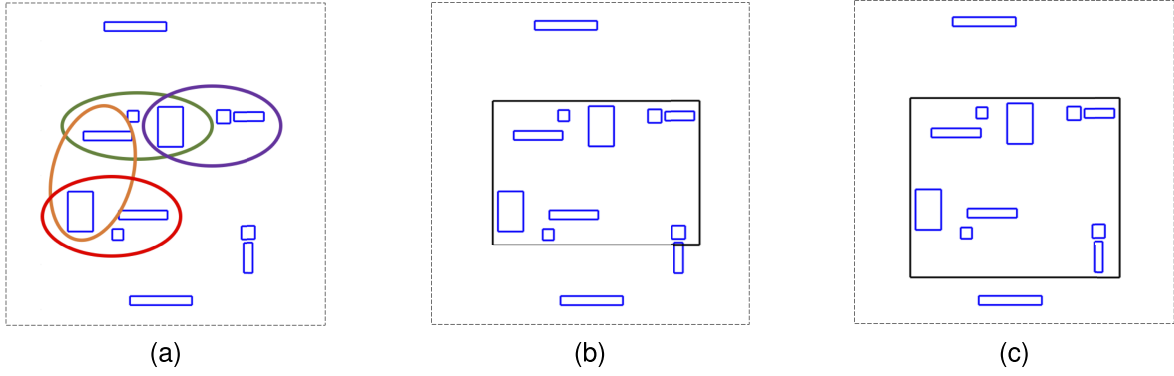


Fig. 5. Process of partitioning a structure into aggressor blocks. (a) Those elements at a distance smaller than r_{el} form subgraph U_a^S , (b) resulting block intersecting a heating element, and (c) incorporating the intersected element within the subgraph.

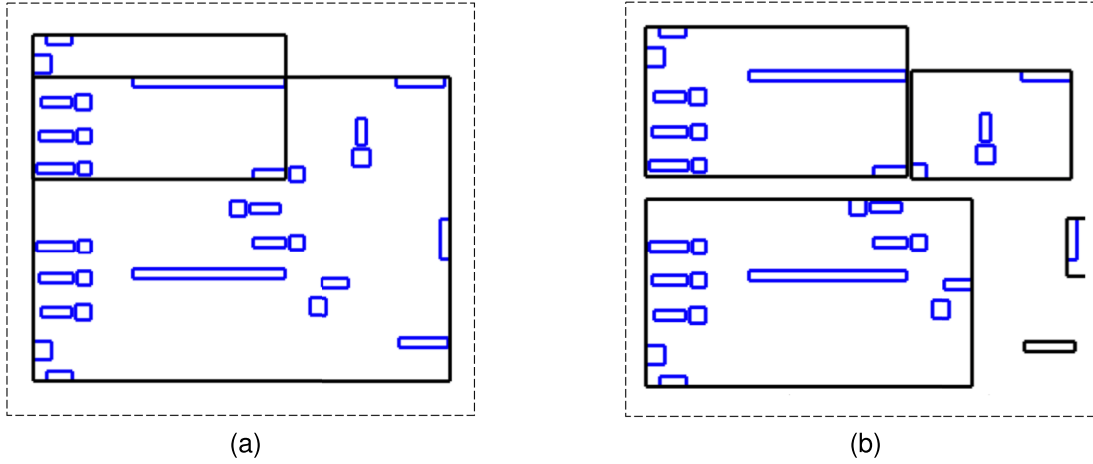


Fig. 6. Ensuring the coordinates of different aggressor blocks do not intersect during the aggressor partitioning stage. (a) Two blocks with intersecting borders, and (b) splitting the subgraph to form nonintersecting blocks.

x - and y -axes. L_X comprises (23a) and (23b), whereas L_Y comprises (23a) and (23c) for each block q .

The lower coordinates of the new victim block x_0 and y_0 are found from the minima of $D(\Gamma)$ and $R(\Gamma)$. If the nearest subsequent partitioned region denoted by coordinates (x_1, y_1) is horizontally closer to (x_0, y_0) , L_X is used to determine the coordinates of the new victim block. The nearest boundary to the right of x_0 whose y -coordinates overlap the vertical space between y_0 and y_1 is identified. This boundary sets the upper x -coordinate of the new victim block x_w . If (x_1, y_1) is vertically closer to (x_0, y_0) , y_w is determined from the boundaries in L_Y .

After the vertices of the new victim block are added to Γ , any duplicate pairs $(x, y) \in \Gamma$ are deleted. The algorithm continues to partition the circuit into blocks until no remaining vertices in Γ exist. The computational complexity of the victim partitioning stage of the algorithm is $O(|Q|^2)$, with the upper limit of $O(|U_a|^2)$ when each aggressor block contains only one heating element. An example of the output of the victim partitioning stage of the algorithm is depicted in Fig. 7(b).

C. Average Thermal Algorithm

Once all the boundaries of the blocks are determined, the thermal algorithm formulates a graph from the partitioned

structure. The structure can be described as a loopless undirected simple graph $G(B, R)$, where each block is represented as a node within the set of nodes B , and R is the set of thermal resistances connecting the nodes [11], [38]. From the block coordinates produced during the aggressor and victim partitioning stages of the algorithm, the adjacency of the blocks is determined to generate the corresponding graph. An adjacency matrix Z is formed which describes the connectivity within a graph. Element $z \in Z$ corresponding to blocks b_i and b_j is

$$z_{ij} = \begin{cases} 1, & \text{if } b_i \text{ adjacent to } b_j \\ 0, & \text{otherwise.} \end{cases} \quad (24)$$

Since G is a simple graph, meaning no self-loops, the diagonal elements of Z are zero. The adjacency matrix is characterized by weights representing the resistance of the edge and converted into an upper triangular matrix, where element $z_{ij}^w \in Z^w$ is

$$z_{ij}^w = \begin{cases} r_{ij}, & \text{if } z_{ij} = 1, i > j \\ \infty, & \text{if } z_{ij} = 0 \text{ or } i < j. \end{cases} \quad (25)$$

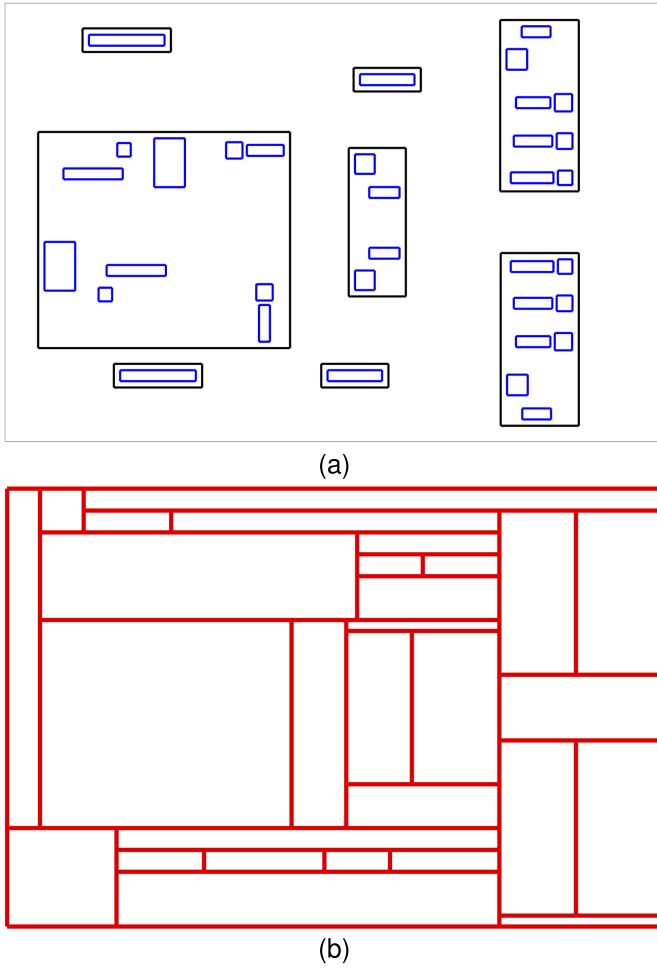


Fig. 7. Partitioning structure into blocks. (a) Location of heating blocks from the aggressor partitioning stage. (b) Location of victim blocks from the victim partitioning stage.

The resulting thermal resistance grid is

$$Z^w = R = \begin{bmatrix} b_1 & b_k & b_n \\ R_{11} & \dots & R_{1n} \\ \vdots & \ddots & \vdots \\ R_{n1} & \dots & R_{nn} \end{bmatrix} \begin{matrix} b_1 \\ b_k \\ b_n \end{matrix} \quad (26)$$

where the thermal resistances across and under the diagonal of the matrix are equated to infinity.

An example of this process is depicted in Fig. 8. The example structure is partitioned into blocks, as shown in Fig. 8(a). The adjacency matrix of the graph generated for this example structure using the thermal algorithm is

$$Z = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix} \begin{matrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \\ b_6 \end{matrix} \quad (27)$$

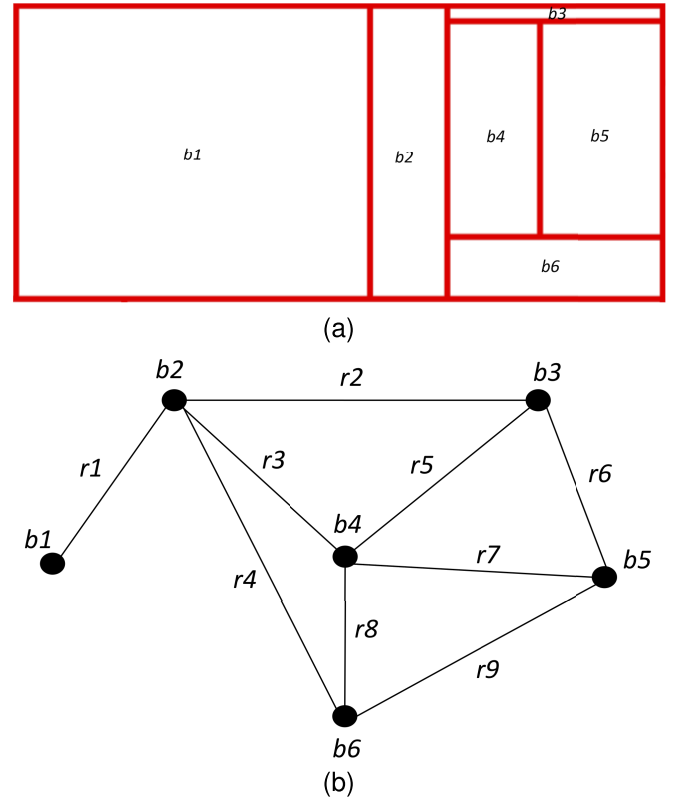


Fig. 8. Example of superconductive circuit structure. (a) Partitioned structure. (b) Equivalent graph.

as shown in Fig. 8(b). The thermal resistance grid corresponding to Z is

$$R = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 \\ \infty & R_{12} & \infty & \infty & \infty & \infty \\ \infty & \infty & R_{23} & R_{24} & \infty & R_{26} \\ \infty & \infty & \infty & R_{34} & R_{35} & \infty \\ \infty & \infty & \infty & \infty & R_{45} & R_{46} \\ \infty & \infty & \infty & \infty & \infty & R_{56} \end{bmatrix} \begin{matrix} b_1 \\ b_2 \\ b_3 \\ b_4 \\ b_5 \end{matrix} \quad (28)$$

The thermal resistance of the paths between the blocks represented by (28) is evaluated by (7) and (8), depending on whether the blocks are aggressors or victims. The aggressor blocks are determined from the aggressor block indices Q produced during the partitioning stage of the algorithm. The number and type of heating elements within the blocks are used to estimate the dissipated power [see (1)]. The operating frequency and switching coefficient to evaluate V_{rms} are assumed here [see (5)]. The dimensions of the blocks are used to evaluate the thermal resistance [see (6)]. The algorithm produces a netlist for thermal simulation, from which a thermal profile of the circuit is produced. The netlist contains a network of current sources describing the dissipated power and resistances describing the heat paths. The temperature is evaluated by extracting the node voltages from a SPICE simulation. This process is outlined by the pseudocode shown in Algorithm 3. The computational complexity of the thermal stage of the algorithm is $O(|U_a| + |B|^2)$.

Algorithm 3 Given Element Indices F_E , Aggressor Block Indices Q , and Coordinates of Blocks x_{part} and y_{part} From the Partitioning Stage of the Algorithm, Operating Frequency f , and Switching Coefficient c_s , Evaluate the Thermal Network. Produce a Netlist to Obtain the Thermal Profile of the Structure

Function: AVERAGE_THERMAL_MODEL
Input : f, c_s
Output : T_{average}

```

1  $F_E \leftarrow \text{AGGRESSOR\_PARTITION}(\text{inputfile}, r_{\text{el}})$ ;
2  $Q, x_{\text{part}}, y_{\text{part}} \leftarrow \text{VICTIM\_PARTITION}(x_{\text{group}}, y_{\text{group}})$ ;
3  $B = \{b_1, \dots, b_k\}, k \leftarrow \text{size}(x_{\text{part}})/2$ ;
4  $R \leftarrow \emptyset$ ;
5 for each pair of blocks  $(b_i, b_j) \in B$  do
6   if  $b_i$  adjacent to  $b_j$  then
7      $w(i, j) \leftarrow \text{boundary width}$ ;
8     if  $b_i \in Q$  and  $b_j \notin Q$  then
9        $R(i, j) \leftarrow \text{Eq. (8)}$ 
10    else
11       $R(i, j) \leftarrow \text{Eq. (7)}$ 
12    end
13  end
14 end
15 for each heating element  $el \in F_E$  do
16    $V_{\text{rms}}(el) \leftarrow \text{Eq. (5)}$ ;
17    $\text{Power}_B(Q(el)) \leftarrow \text{Power}_B(Q(el)) + \text{Power}(el)$ ;
18 end
19 for each block  $b_i \in B$  do
20    $R_{\text{th}}(i) \leftarrow \text{Eq. (6)}$ ;
21 end
22 Write_netlist;

```

Algorithm 4 Given Element Indices F_E , Aggressor Block Indices Q , Coordinates of Blocks x_{part} and y_{part} From the Partitioning Stage of the Algorithm, Operating Frequency f , and Switching Coefficient c_s , Evaluate the Peak Temperature of the Blocks

Function: PEAK_THERMAL_MODEL
Input : f, c_s
Output : T_{max}

```

1  $F_E \leftarrow \text{AGGRESSOR\_PARTITION}(\text{inputfile}, r_{\text{el}})$ ;
2  $Q, x_{\text{part}}, y_{\text{part}} \leftarrow \text{VICTIM\_PARTITION}(x_{\text{group}}, y_{\text{group}})$ ;
3  $B = \{b_1, \dots, b_k\}, k \leftarrow \text{size}(x_{\text{part}})/2$ ;
4 for each heating element  $el \in F_E$  do
5    $V_{\text{rms}}(el) \leftarrow \text{Eq. (5)}$ ;
6    $T_{\text{init}}(el) \leftarrow \text{Eq. (9)}$ ;
7 end
8 for each heating element  $el \in F_E$  do
9    $T_{\text{max\_init}} \leftarrow T_{\text{init}}(el)$ ;
10  if  $T_{\text{max}}(Q(el)) < T_{\text{max\_init}}$  then
11     $T_{\text{max}}(Q(el)) \leftarrow T_{\text{max\_init}}$ ;
12  end
13 end
14 for each pair of blocks  $(q_i \in Q, b_j \in B \setminus Q)$  do
15   if  $\sqrt{(x_{q_i} - x_{b_j})^2 + (y_{q_i} - y_{b_j})^2} < \eta$  then
16     for each  $el \in q_i$  do
17        $T_{\text{max\_init}} \leftarrow \text{Eq. (10)}$ ;
18       if  $T_{\text{max}}(b_j) < T_{\text{max\_init}}$  then
19          $T_{\text{max}}(b_j) \leftarrow T_{\text{max\_init}}$ ;
20       end
21     end
22   end
23 end

```

D. Peak Thermal Algorithm

An algorithm is used to evaluate the peak temperature: specifically, for hotspot detection. To evaluate the peak temperature of a block, a subset of N elements $C[0, \dots, N-1]$ is extracted. The elements in C contribute to the peak temperature of a block. For each aggressor block, C is the set of heating elements within that block.

For those victim blocks that do not contain heating elements, C is based on the proximity between the aggressor and victim blocks. A heating element from a nonadjacent aggressor block can also be the heat source for the peak temperature of a victim block. The thermal healing length η quantifies the distance between these blocks. The set of elements C_{b_j} of victim block b_j includes set C_{q_i} from aggressor block q_i

$$C_{b_j} = \begin{cases} 1, & \text{if } \sqrt{(x_{q_i} - x_{b_j})^2 + (y_{q_i} - y_{b_j})^2} < \eta \\ 0, & \text{otherwise.} \end{cases} \quad (29)$$

For each block within the partitioned structure, C determines the set of element temperatures, $T[0, \dots, N-1]$. The peak thermal algorithm finds the element k within each set, such that $T[k]$ is the maximum temperature of that set

$$T_l(k) = \begin{cases} T_{\text{max}_C}, & \text{if } T_k > T_l \\ 0, & \text{otherwise.} \end{cases} \quad (30)$$

This process is outlined by the pseudocode shown in Algorithm 4. The computational complexity of the peak thermal algorithm is $O(|U_a| + |Q| * |B \setminus Q|)$.

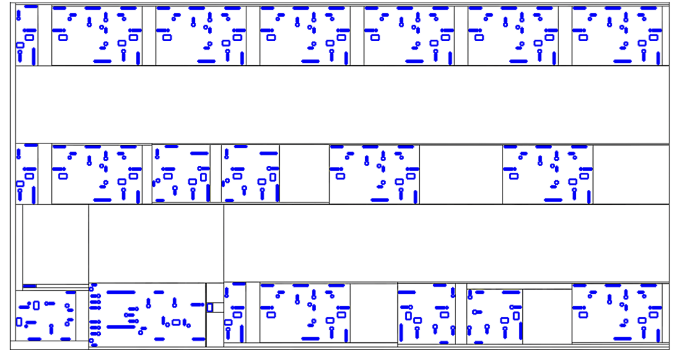


Fig. 9. Portion of the RSFQ AMD2901 used to validate the thermal model divided into aggressor and victim blocks. The aggressor blocks contain heating elements—JJs and resistors.

IV. CASE STUDY

A portion of the AMD2901 is evaluated to validate the algorithm. The RSFQ version of the structure contains several types of gates—XOR gates, Josephson transmission lines (JTLs), flip flops, and splitters. The dimensions of the structure are $380 \times 200 \mu\text{m}$, and the circuit contains 18 gates (155 resistively shunted JJs). The structure is divided into blocks based on the element radius r_{el} characterizing the complexity of the blocks. Element radius parameters, 7.5, 10, and $50 \mu\text{m}$, are chosen to evaluate the effects of the substantial variations among the block size considered in the case study. An example of a structure partitioned into blocks based on $r_{\text{el}} = 7.5 \mu\text{m}$ is shown in Fig. 9. The circuit is assumed to operate at 50 GHz with a global switching coefficient of 0.5.

TABLE I
ACCURACY OF THERMAL MODEL AS COMPARED TO THE NUMERICAL SIMULATOR, COMSOL MULTIPHYSICS [19]

r_{el} (μm)	Average Thermal Model				Peak Thermal Model			
	$ D _{ave}$ (mK)	$ D _{max}$ (mK)	δ_{ave} (%)	δ_{max} (%)	$ D _{ave}$ (mK)	$ D _{max}$ (mK)	δ_{ave} (%)	δ_{max} (%)
50	4.50	14.33	0.11	0.34	10.85	20.15	0.24	0.47
10	4.46	20.70	0.11	0.49	14.84	32.93	0.34	0.74
7.5	5.03	21.56	0.12	0.51	14.77	39.96	0.33	0.92

To validate the model, a 3-D layout of the structure is evaluated by the numerical simulator, based on the MIT LL SFQ5ee fabrication technology specifications [17], [18]. Eight superconductive niobium layers, each separated by an oxide layer, are included in this process. The bias resistors, JJs, and shunt resistors are positioned within the fifth metal (M5) and oxide layer (O5). The inductors and connections between the logic components are positioned in layer M6. The three ground planes are M1, M4, and M7. The remaining metal layers, M0, M2, and M3, are primarily used for the power distribution network and passive transmission lines [39], [40]. The structure is built above an oxidized 500- μm thick Si substrate. The material properties incorporated in the model and the heating characteristics applied to the heating elements are described in [13].

A comparison between the thermal model and a numerical simulator is listed in Table I, where $|D| = |T_{\text{numerical}} - T_{\text{model}}|$ is the absolute difference, and

$$\delta = \frac{|T_{\text{numerical}} - T_{\text{model}}|}{\text{average}(T_{\text{numerical}}, T_{\text{model}})} \quad (31)$$

is the relative difference between the thermal model and a numerical simulator. The greatest discrepancy between the average thermal model and the numerical simulation is 0.51%. Due to the averaging nature of the average thermal model, higher accuracy is produced at larger block complexities. Rather than averaging the temperature of all of the heaters within the blocks, the peak thermal model determines the temperature generated by the greatest heat producing element. The accuracy of this model therefore does not linearly depend on the complexity of the block. The largest discrepancy between the peak thermal model and the numerical simulation is 0.92%. The discrepancy of the model is below 3% of the critical operating range ($T_{\text{threshold}} - T_{\text{He}}$). Based on a threshold temperature of 5.7 K (see Section II-D), the maximum error of the peak model (40 mK) is 2.7% of the critical operating range.

A thermal profile depicting the average temperature for the block size with the greatest discrepancy is shown in Fig. 10. Note that for enhanced visibility, the color bar in Fig. 10 is limited to the range of 4.2 to 4.25 K. A thermal profile of the same structure depicting the peak temperature is portrayed in Fig. 11. The minimum and maximum temperatures are, respectively, shown at the bottom and top of the color bar for both thermal profiles. The increase in average temperature for this structure is ~ 160 mK. The increase in peak temperature of the structure is ~ 1.66 K.

The thermal algorithm is applied to an RSFQ version of the AMD2901 four-bit microprocessor. The IC comprises 11 609 gates containing a total of 335 616 heating elements

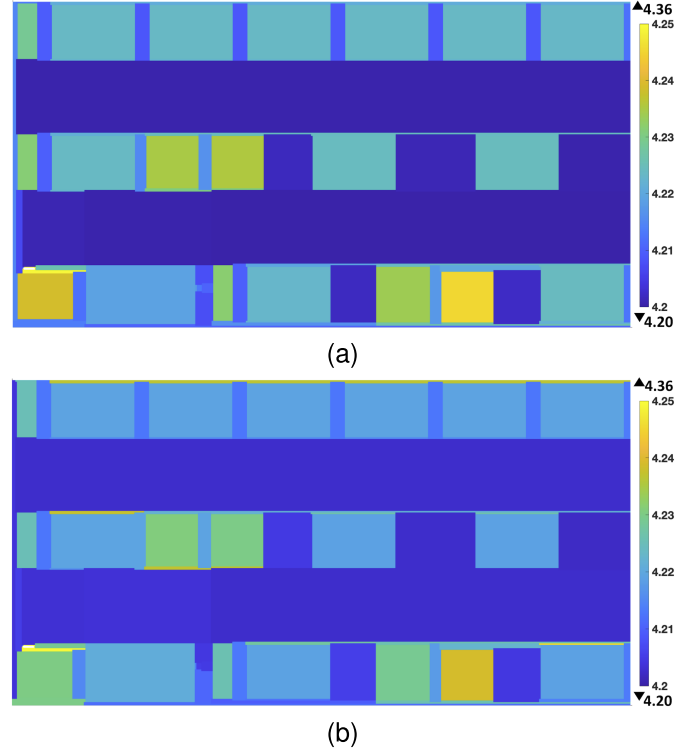


Fig. 10. Thermal profile showing average temperature of a portion of the RSFQ AMD2901 ($r_{el} = 7.5 \mu\text{m}$). (a) Thermal model. (b) Numerical simulation. The average discrepancy between the model and numerical simulation is 5.03 mK.

(114 944 JJs). The location of the gates is extracted from the floorplan of the IC in GDSII format. The input file for the algorithm is generated by populating the floorplan with a layout composed of RSFQ standard cells. The multistage algorithm and SPICE simulation are applied to an eight-core (Intel¹ Core² i7-6700) 3.4-GHz system. The results are listed in Table II, including the number of blocks and thermal resistances for each r_{el} . $|\Delta T|_{\text{max}}$ is the maximum increase in average or peak temperature within the IC. The computational runtime is listed for each stage of the algorithm. For the average temperature model stage, this computational time includes the time to run the thermal stage of the algorithm and the SPICE simulation of the netlist to evaluate the thermal behavior. The primary bottleneck of the algorithm is the aggressor partitioning stage with a complexity of $O(|U_a|^2)$. The complexity of this stage can be reduced by exploiting the repetitiveness of the standard cells within the floorplan. Since current RSFQ technology supports several hundreds of thousands of junctions, this algorithm is suitable for modern RSFQ ICs.

²Trademarked.

TABLE II
PERFORMANCE OF THERMAL ALGORITHM APPLIED TO AMD2901 IC

r_{el} (μm)	Partition stages		Average model stage			Peak model stage	
	Runtime (min)	No. of blocks	Runtime (min)	No. of thermal resistances	$ \Delta T _{max}$ (mK)	Runtime (min)	$ \Delta T _{max}$ (mK)
50	18	5	0.2	13	9	0.2	
10	35	19,365	0.3	90,776	65	0.2	2,210
7.5	67	47,070	0.6	219,806	261	0.4	

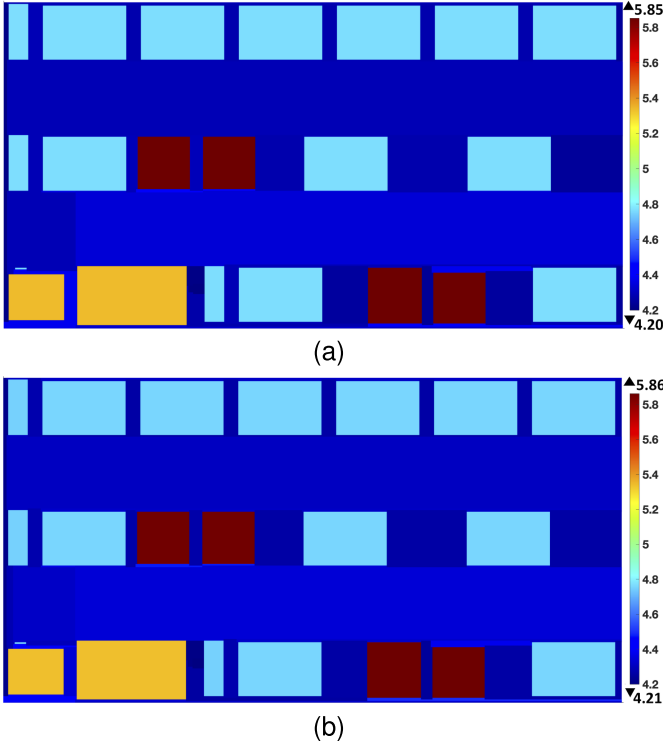


Fig. 11. Peak thermal profile of a portion of the RSFQ AMD2901 ($r_{el} = 7.5 \mu\text{m}$). (a) Thermal model. (b) Numerical simulation. The average discrepancy between the peak thermal model and numerical simulation is 14.77 mK.

V. CONCLUSION

Heating within cryogenic superconductive circuits results in reduced operational margins, operational failure, and degradation of the refrigeration characteristics due to a larger heat load. A methodology for evaluating the thermal behavior of large RSFQ systems is presented in this article. A partitioning algorithm, including an analytic thermal model of RSFQ ICs, is provided. The model considers both the average thermal and peak thermal behavior. The aggressor partitioning stage of the multistage algorithm divides the circuit into aggressor blocks based on the distance between heating elements. The remainder of the circuit is divided into blocks based on the victim partitioning stage of the algorithm. The thermal model is applied to the partitioned structure. The model is validated using a portion of the RSFQ AMD2901, comprising 155 JJs. The model is validated by a numerical solver, producing a 5-mK average error for the average thermal model and a 15-mK average error for the peak thermal model. The algorithm is applied to the AMD2901 benchmark circuit composed of more than 100 000 JJs and 300 000 heating elements, generating a thermal profile in under 68 min.

REFERENCES

- [1] G. Krylov and E. G. Friedman, *Single Flux Quantum Integrated Circuit Design*. Cham, Switzerland: Springer, 2022.
- [2] D. K. Brock, "RSFQ technology: Circuits and systems," *Int. J. High Speed Electron. Syst.*, vol. 11, no. 1, pp. 307–362, Mar. 2001.
- [3] K. K. Likharev and V. K. Semenov, "RSFQ logic/memory family: A new Josephson-junction technology for sub-terahertz-clock-frequency digital systems," *IEEE Trans. Appl. Supercond.*, vol. 1, no. 1, pp. 3–28, Mar. 1991.
- [4] K. Likharev, O. Mukhanov, and V. Semenov, "Resistive single flux quantum logic for the Josephson-junction digital technology," in *Proc. Int. Conf. Superconducting Quantum Devices*, Jun. 1985, pp. 1103–1108.
- [5] Y. Ando, R. Sato, M. Tanaka, K. Takagi, and N. Takagi, "80-GHz operation of an 8-bit RSFQ arithmetic logic unit," in *Proc. 15th Int. Superconductive Electron. Conf. (ISEC)*, Jul. 2015, pp. 1–3.
- [6] T. Jabbari and E. G. Friedman, "SFQ/DQFP interface circuits," *IEEE Trans. Appl. Supercond.*, vol. 33, no. 5, pp. 1–5, Aug. 2023.
- [7] S. K. Tolpygo et al., "Inductance of circuit structures for MIT LL superconductor electronics fabrication process with 8 niobium layers," *IEEE Trans. Appl. Supercond.*, vol. 25, no. 3, pp. 1–5, Jun. 2015.
- [8] S. K. Tolpygo, E. B. Golden, T. J. Weir, and V. Bolkhovsky, "Inductance of superconductor integrated circuit features with sizes down to 120 nm," *Supercond. Sci. Technol.*, vol. 34, no. 8, Jun. 2021, Art. no. 085005.
- [9] T. Jabbari and E. G. Friedman, "Inductive and capacitive coupling noise in superconductive VLSI circuits," *IEEE Trans. Appl. Supercond.*, vol. 33, no. 9, pp. 1–7, Dec. 2023.
- [10] A. S. Lavine and C. Bai, "An analysis of heat transfer in Josephson junction devices," *J. Heat Transf.*, vol. 113, no. 3, pp. 535–543, Aug. 1991.
- [11] N. Zhuldassov, R. Bairamkulov, and E. G. Friedman, "Thermal optimization of hybrid cryogenic computing systems," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 31, no. 9, pp. 1339–1346, Sep. 2023.
- [12] S. Razmkhah and A. Bozbey, "Heat flux capacity measurement and improvement for the test of superconducting logic circuits in closed-cycle cryostats," *TURKISH J. Electr. Eng. Comput. Sci.*, vol. 27, no. 5, pp. 3912–3922, Sep. 2019.
- [13] A. Mitrovic and E. G. Friedman, "Thermal modeling of rapid single flux quantum circuit structures," *IEEE Trans. Electron Devices*, vol. 69, no. 5, pp. 2718–2724, May 2022.
- [14] T. Jabbari, G. Krylov, S. Whiteley, E. Mlinar, J. Kawa, and E. G. Friedman, "Interconnect routing for large-scale RSFQ circuits," *IEEE Trans. Appl. Supercond.*, vol. 29, no. 5, pp. 1–5, Aug. 2019.
- [15] T. A. Ohki, J. L. Habif, M. J. Feldman, and M. F. Bocko, "Thermal design of superconducting digital circuits for Millikelvin operation," *IEEE Trans. Applied Supercond.*, vol. 13, no. 2, pp. 978–981, Jun. 2003.
- [16] B. Venter, "Development of a thermal analysis tool for superconducting circuits," M.S. thesis, Dept. Elect. Electron. Eng., MEng, Stellenbosch Univ., Stellenbosch, South Africa, 2021.
- [17] S. K. Tolpygo, "Superconductor digital electronics: Scalability and energy efficiency issues (review article)," *Low Temp. Phys.*, vol. 42, no. 5, pp. 361–379, May 2016.
- [18] S. K. Tolpygo et al., "Advanced fabrication processes for superconductor electronics: Current status and new developments," *IEEE Trans. Appl. Supercond.*, vol. 29, no. 5, pp. 1–13, Aug. 2019.
- [19] *Heat Transfer Module User's Guide, COMSOL Multiphysics® V. 5.4*, COMSOL AB, Stockholm, Sweden, 2018.
- [20] W. Huang, M. R. Stan, K. Skadron, K. Sankaranarayanan, S. Ghosh, and S. Velusam, "Compact thermal modeling for temperature-aware design," in *Proc. 41st Annu. Design Autom. Conf.*, Jun. 2004, pp. 878–883.
- [21] M. Pedram and S. Nazarian, "Thermal modeling, analysis, and management in VLSI circuits: Principles and methods," *Proc. IEEE*, vol. 94, no. 8, pp. 1487–1501, Aug. 2006.

- [22] E. C. W. de Jong, J. A. Ferreira, and P. Bauer, "Thermal design based on surface temperature mapping," *IEEE Power Electron Lett.*, vol. 3, no. 4, pp. 125–129, Dec. 2005.
- [23] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "HotSpot: A compact thermal modeling methodology for early-stage VLSI design," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 14, no. 5, pp. 501–513, May 2006.
- [24] K. K. Likharev and J. Lukens, "Dynamics of Josephson junctions and circuits," *Phys. Today*, vol. 41, no. 11, p. 122, Nov. 1988.
- [25] P. K. Hansma, G. I. Rochlin, and J. N. Sweet, "Externally shunted Josephson junctions: Generalized weak links," *Phys. Rev. B, Condens. Matter*, vol. 4, no. 9, pp. 3003–3014, Nov. 1971.
- [26] A. M. Kadin, C. A. Mancini, M. J. Feldman, and D. K. Brock, "Can RSFQ logic circuits be scaled to deep submicron junctions?" *IEEE Trans. Appl. Supercond.*, vol. 11, no. 1, pp. 1050–1055, Mar. 2001.
- [27] G. Krylov and E. G. Friedman, "Design methodology for distributed large-scale ERSFQ bias networks," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 11, pp. 2438–2447, Nov. 2020.
- [28] C. J. Fourie, "Extraction of DC-biased SFQ circuit verilog models," *IEEE Trans. Appl. Supercond.*, vol. 28, no. 6, pp. 1–11, Sep. 2018.
- [29] A. M. Kadin, *Introduction to Superconducting Circuits*. Hoboken, NJ, USA: Wiley, 1999.
- [30] M. Asheghi, M. N. Touzelbaev, K. E. Goodson, Y. K. Leung, and S. S. Wong, "Temperature-dependent thermal conductivity of single-crystal silicon layers in SOI substrates," *J. Heat Transf.*, vol. 120, no. 1, pp. 30–36, Feb. 1998.
- [31] J. Lee et al., "Thermoelectric characterization and power generation using a silicon-on-insulator substrate," *J. Microelectromech. Syst.*, vol. 21, no. 1, pp. 4–6, Feb. 2012.
- [32] E. Pop, V. Varshney, and A. K. Roy, "Thermal properties of graphene: Fundamentals and applications," *MRS Bull.*, vol. 37, no. 12, pp. 1273–1281, Nov. 2012.
- [33] E. Yalon et al., "Temperature-dependent thermal boundary conductance of monolayer MoS₂ by Raman thermometry," *ACS Appl. Mater. Interfaces*, vol. 9, no. 49, pp. 43013–43020, Dec. 2017.
- [34] B. Seeber, *Handbook of Applied Superconductivity*, vol. 2. Boca Raton, FL, USA: CRC Press, 2010.
- [35] T. Van Duzer and C. W. Turner, *Principles of Superconductive Devices and Circuits*. Upper Saddle River, NJ, USA: Prentice-Hall, 1981.
- [36] Whiteley Research Inc. *WRSPICE Reference Manual*. Accessed: Nov. 10, 2023. [Online]. Available: <http://www.wrcad.com/manual/wrsmmanual.pdf>
- [37] R. Bairamkulov, T. Jabbari, and E. G. Friedman, "QuCTS—Single-flux quantum clock tree synthesis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 41, no. 10, pp. 3346–3358, Oct. 2022.
- [38] R. Bairamkulov and E. G. Friedman, *Graphs in VLSI*. Cham, Switzerland: Springer, 2023.
- [39] S. K. Tolpygo et al., "Advanced fabrication processes for superconducting very large-scale integrated circuits," *IEEE Trans. Appl. Supercond.*, vol. 26, no. 3, pp. 1–10, Apr. 2016.
- [40] S. S. Meher, C. Kanungo, A. Shukla, and A. Inamdar, "Parametric approach for routing power nets and passive transmission lines as part of digital cells," *IEEE Trans. Appl. Supercond.*, vol. 29, no. 5, pp. 1–7, Aug. 2019.



Ana Mitrovic received the B.Sc. and M.Sc. degrees in electrical and computer engineering from the University of Novi Sad, Novi Sad, Serbia, in 2016 and 2018, respectively. She is currently working toward the Ph.D. degree at the School of Electrical and Computer Engineering, University of Rochester, Rochester, NY, USA.

In the summers of 2021 and 2022, she interned with the DDR PHY System/Timing Design Team, Qualcomm Inc., San Diego, CA, USA. Her research

interests include superconductive and cryogenic electronics, low T_c superconductive integrated circuit design, superconductive–ferromagnetic devices, and electronic design automation of emerging VLSI technologies.



Eby G. Friedman (Life Fellow, IEEE) received the B.S. degree in electrical engineering from the Lafayette College, Easton, PA, USA, in 1979, and the M.S. and Ph.D. degrees in electrical engineering from the University of California at Irvine, Irvine, CA, USA, in 1981 and 1989, respectively.

He was with Hughes Aircraft Company, Carlsbad, CA, USA, from 1979 to 1991, rising to the Manager of the Signal Processing Design and Test Department, where he was responsible for the design and test of high-performance digital and analog ICs. He has been with the Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY, USA, since 1991, where he is currently a Distinguished Professor and the Director of the High Performance VLSI/IC Design and Analysis Laboratory. He is also a Visiting Professor with the Technion—Israel Institute of Technology, Haifa, Israel. He has authored almost 600 articles and book chapters, holds 29 patents, and has authored or edited 21 books in the fields of high-speed and low-power CMOS design techniques, 3-D design methodologies, high-speed interconnect, superconductive circuits, and the theory and application of synchronous clock and power distribution networks. His current research and teaching interests include high-performance synchronous digital and mixed-signal circuit design and analysis with application to high-speed portable processors, low-power wireless communications, and server farms.

Dr. Friedman is a Senior Fulbright Fellow, a National Sun Yat-sen University Honorary Chair Professor, and an Inaugural Member of the UC Irvine Engineering Hall of Fame. He was a recipient of the IEEE Circuits and Systems Mac Van Valkenburg Award, the IEEE Circuits and Systems Charles A. Desoer Technical Achievement Award, the University of Rochester Graduate Teaching Award, and the College of Engineering Teaching Excellence Award. He was the Editor-in-Chief and the Chair for the Steering Committee of the IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION (VLSI) SYSTEMS, the Editor-in-Chief of the *Microelectronics Journal*, a Regional Editor of the *Journal of Circuits, Systems and Computers*, an editorial board member for numerous journals, and a program and technical chair for several IEEE conferences.