面向单磁通量子电路的多数逻辑综合框架

张俊尧南加州大学洛杉矶，加利福尼亚州
junyaozh@usc.edu

Paul Bogdan南加州大学洛杉矶，加利福尼亚州
pbogdan@usc.edu

Shahin Nazarian南加州大学洛杉矶，加利福尼亚州
shahin.nazarian@usc.edu

arXiv:2301.10695v1 [cs.LO] 25 Jan 2023

摘要——**百亿亿次级计算及其相关应用对效率提出了越来越高的要求。基于半导体晶体管的电路（STbC）在提高 GHz 频率的同时，难以解决功耗问题。作为 STbC 的一种替代方案，超导电子（SCE）技术中的单磁通量子（SFQ）逻辑有望在超低功耗下实现更高的时钟频率。然而，其量子化脉冲操作、高环境要求、工艺偏差以及其他 SFQ 特有的非理想因素，都是导致 SFQ 电路发生逻辑错误的重要原因。减少电路中的约瑟夫森结（JJ）数量，是降低上述误差源影响的一种有效方法，因此也是大型 SFQ 电路设计流程中的关键环节。本文提出了一种新型 SFQ逻辑综合框架：该框架以网表为输入，提供自动化映射方案，其中包含多数门（MAJ）逻辑，旨在尽可能减少约瑟夫森结数量，同时满足设计的独特特征和要求。实验结果表明，我们的综合框架显著优于当前最先进的学术界 SFQ 技术映射器，平均可将约瑟夫森结数量减少 35.0%**。

I. 引言

作为超导电子产品领域中与最先进半导体电路强劲竞争的技术，SCE 在超高时钟频率和超低功耗方面展现出巨大潜力 [1]。作为 SCE 中最受瞩目的候选技术，单磁通量子受到越来越多追求更高性能和更高能效的研究群体关注。然而，单磁通量子电路对环境因素高度敏感，这一直是个问题 [1], [2]。已有一些研究探讨制造后测试中的硬故障 [3], [4]。不过，与 CMOS 技术相比，单磁通量子技术的特征尺寸要大得多 [5]。例如，近期一种约瑟夫森结制造工艺基于 200 nm 的量产级工艺 [6]。较大的特征尺寸意味着较低的缺陷密度，因此，固定型故障及其他硬故障的影响较小。

相比之下，工艺偏差以及其他单磁通量子特有问题（电感耦合、偏置电流转向、磁通俘获）等非理想因素，会对制造出的芯片 [7]的运行产生重大影响。这些故障会导致逻辑门产生错误输出，正如高度失真的脉冲无法被单元正确地进行逻辑解释，或意外的抖动可能产生逻辑高电平 [8]一样。有人提出了一种以工艺偏差为重点的自动测试图样生成范式，用于生成针对这些随机故障的测试图样 [7]。然而，这些故障仍无法得到有效消除或减轻。因此，开发一种能够减轻这些偶发故障负面影响的有效方法至关重要。

逻辑综合是数字电路设计流程中的关键步骤，因为它能够使目标指标达到最优 [9]。逻辑综合分为两个不同阶段：**1）与工艺无关的优化：**布尔优化，例如重构、重新代换、公共子表达式提取和节点最小化。**2）技术映射：**将逻辑表达式与给定单元库中的实际逻辑门相关联。根据理论和实验结果 [2], [7], ，通过减少单磁通量子电路中约瑟夫森结的数量（记为 #JJs），可以有效减少上述故障。因此，减轻单磁通量子电路中非理想因素影响的一种有效方法，是采用旨在减少 #JJs 数量的映射算法。

此外，与其完全依赖基于与门-或门-反相器（AOI）的表示法，还可以通过将多数逻辑（MAJ）门纳入表达式来改进逻辑综合 [10]。尽管动态单磁通量子（DSFQ）中已提出多数电路结构 [11]，，且互补金属氧化物半导体（CMOS） [12]和绝热量子磁通参量子（AQFP） [13]领域均已有多数逻辑综合方面的研究，但SFQ中的多数电路结构以及能够将多数逻辑系统地集成到表达式中的综合框架仍有待实现。

本文提出一种逻辑综合框架，可根据给定网表生成包含多数门的SFQ网表。映射后的网表可减少约瑟夫森结的数量，从而降低SFQ电路中出现非理想因素的风险。如图1所示，该框架分为三个不同阶段：**1）前后处理：**这两个过程涉及SFQ特有的电路修改，例如路径深度平衡、插入分路器二叉树以及门转换。**2）映射优化：**使用改进的奎因–麦克拉斯基算法优化每个节点的割电路，然后采用贪心方法，利用各节点对应的代表割修改节点，以降低约瑟夫森结总数与网表网络深度的乘积（记为PND）。本文的主要贡献总结如下：

- 我们提出了一种单磁通量子逻辑中的多数门电路配置，并将其应用于我们提出的逻辑综合框架。
- 我们为单磁通量子逻辑开发了一种新颖的贪心映射算法，可修改每个节点及其代表

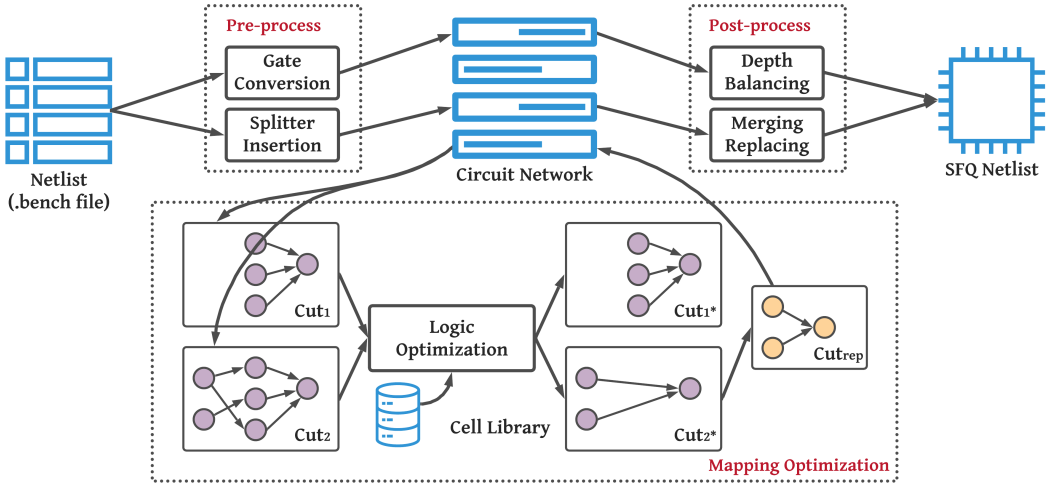

图1：所提框架概览。1）预处理：在输入网表中插入分路器；将网表中的所有单元转换为单元库中的单元。**2）映射优化：**为每个节点找出所有满足 K 条件的切割；使用改进的奎因–麦克拉斯基算法优化每个割电路的布尔函数 Cut_n；根据优化后的布尔函数重新生成割电路 Cut_n*；通过代表割修改该节点 Cut_{rep}（即重新生成的割电路中 PND 值改善最显著的一个）。**3）后处理：**插入D触发器以平衡每个节点扇入的逻辑层级；合并并替换路径深度平衡D触发器，以进一步降低其在单磁通量子网表中的开销。

割，以降低 PND值。

- 我们提出了一种用于单磁通量子逻辑的逻辑综合框架，可将网表处理为单磁通量子网表，同时兼顾单磁通量子逻辑的独特特性，并减轻单磁通量子电路中非理想因素的影响

本文其余部分将在后续各节依次介绍预备知识、相关工作、研究动机、非理想因素下的单元表征、多数门、单磁通量子逻辑综合框架、评估和结论。

II. 预备知识

A. 单磁通量子逻辑

单磁通量子中的过阻尼约瑟夫森结采用电阻性电流偏置，这意味着二进制信息以持续数皮秒的电压脉冲 $V(t)$ 表示，而非像半导体技术那样通过电压电平表示。由于该电压脉冲等效于一个磁通量子 $\phi_o = 2.07\times10^{-15} V\cdot s$，因此也称为单磁通量子脉冲（SFQ 脉冲）。这些独特属性从根本上改变了人们对比特表示方式的传统理解。基本约定是：在当前时钟周期内到达一个单磁通量子脉冲表示逻辑“1”，而该周期内没有任何脉冲则表示逻辑“0”。此外，单磁通量子电路还有若干特有属性，下面将对此进行讨论。

1) 扇出：鉴于单磁通量子逻辑的量子通信特性，单元的扇出严格等于 1。要实现更高的扇出，需要使用一种特殊的异步单磁通量子门（分路器单元）[1]。分路器接收一个单磁通量子脉冲后，只能产生两个输出脉冲。可以将额外的分路器组成二叉树结构，以实现更多扇出。要满足 n 个扇出，需要$n-1$ 个分路器。图 2 展示了扇出等于 4 时的分路器二叉树示例。

2) 门级流水线：几乎所有单磁通量子门（某些异步门除外，例如分路器或约瑟夫森传输线）都与时钟同步，因此单磁通量子门可以被视作一个输出端带有边沿触发触发器的 CMOS门。只有时钟脉冲到达时，输出才会响应各自的输入。本工作 [14] 概述了时钟分配网络的设计，其核心结论是，单磁通量子电路必须完全采用门级流水线结构。

3) 路径深度平衡：由于门级流水线的前提要求，单磁通量子门的所有输入必须处于相同的逻辑层级才能正确工作。如果各扇入的逻辑层级不同，则应在路径中插入一些 D触发器，以平衡输入的逻辑层级。假设或门的扇入路径深度不同：$in_1 = 1$、$in_2 = 2$。若没有路径深度平衡触发器，in_1 所代表的正确脉冲将比 in_2 提前 1 个时钟周期被消耗。因此，需要路径深度平衡D触发器来确保逻辑值正确。

B. 逻辑综合

逻辑综合分为两个阶段：技术无关阶段和技术相关阶段 [9]。第一阶段会进行若干优化，以减少给定网络中的文字总数。一些功能操作包括公共子表达式提取、分解和重代入 [15]。第二阶段称为技术映射，在该阶段中，从给定库中为给定网络中的节点分配合适的门，以满足特定约束，例如最小化某些代价函数 [9]。提供的网表需要先转换为布尔网络，然后才能进行技术映射。布尔

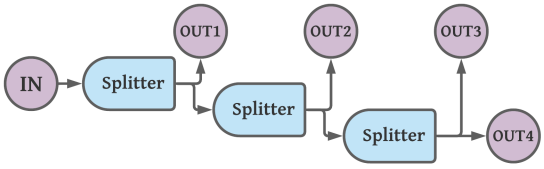

图 2： 分路器二叉树（扇出 = 4）

网络是有向无环图（DAG） [9],，其中节点表示逻辑门，有向边对应连接各逻辑门的连线。本文中，网络和电路这两个术语可以互换使用。主输入（PIs）/主输出（POs）是当前网络中没有扇入/扇出的节点 [16]。节点 n 的切割定义如下： n 是根节点（该切割的输出）， l_n 是网络中称为叶节点的一组节点（该切割的输入） [16]。从主输入到 n 的每条路径都至少经过一个叶节点。如果切割包含的叶节点不超过 K 个，则称其为 K-可行切割 [17]。节点的逻辑层级是从任一主输入到该节点的最长路径长度。逻辑深度是网络中内部节点的最大层级。

III. 相关工作

在单磁通量子电路逻辑综合的相关文献中，有少数论文探讨了技术无关问题和技术映射问题 [18]–[21]。在 [18], 中，研究者通过构建名为“2- AND/XOR”的虚拟单元，开发了一个框架，使得可以将互补金属氧化物半导体逻辑综合工具用于单磁通量子电路。在 [21], 中，研究者将路径深度平衡D触发器和分路器添加到网表中，随后应用标准重定时算法 [22] ，以减少所需的路径深度平衡D触发器数量。在 [20], 中，研究者提出了一种单磁通量子逻辑电路技术映射工具（称为SFQmap），重点在于最小化逻辑深度以及最坏情况级延迟与逻辑深度的乘积（PSD）。 [19]的工作与SFQmap类似，能够生成具有平衡结构的映射方案。 [19], [20] 中的映射算法通过应用基于切割枚举的技术实现，与 [17]中的工作较为接近。上述单磁通量子电子学领域的论文 [19]–[21]均基于与非图（AIG）逻辑综合与验证工具ABC [23]开发。

此外，除了互补金属氧化物半导体技术之外，纳米技术等领域也有一些应用受益于多数逻辑综合，例如量子元胞自动机（QCA） [24]和单电子隧穿（SET）{v2）。在单磁通量子电子学领域，提及多数逻辑的论文寥寥无几。其中一篇提出了适用于AQFP的多数逻辑综合框架 [13], ，另一篇则提出了异步动态单磁通量子多数门设计 [11]。

IV. 研究动机

尽管在其他领域利用逻辑综合取得了显著的研究进展，但单磁通量子逻辑中的最先进综合工具仍有很大的改进空间。在 [19], [20], 其映射算法基于切割枚举技术开发。该技术旨在通过查找表（LUT）将割电路映射到超级门。具体而言，映射算法会建立一个庞大的超级门库，并针对每个割电路在其中进行搜索。如果某个超级门的 LUT 与该割电路相同，则可以使用该超级门映射该割电路。这项技术的缺点是 K 取值范围有限，以及超级门库带来的内存开销；随着逻辑深度和基本单元库规模的增加，超级门库会呈指数级增长。例如，若原始库中有 20 个逻辑门并使用最高至第 3 级的超级门，则超级门数量约为 4000 个。尽管建立了大规模的超级门库，但随着 K 取值增加 [19], 仍然很难找到能够实现某个切割函数的映射超级门。此外，上述技术主要侧重于深度优化和路径深度平衡开销。除了这些优化之外，我们还希望将多数逻辑集成到映射算法中，以减少约瑟夫森结数量，从而减轻随机故障的影响。

我们提出了一种新颖的SFQ逻辑综合框架，通过减少约瑟夫森结数量，倾向于生成受SFQ相关非理想因素影响较小的映射方案。我们为以下方面推导了若干闭式公式：切割选择、改进的奎因–麦克拉斯基算法，以及具有SFQ特定属性的网表处理算法。据我们所知，这是首篇提出SFQ逻辑综合框架的论文，该框架：1）涉及多数门；2）减轻非理想因素对SFQ电路的影响。

V. 非理想因素下的单元表征

单磁通量子电路利用约瑟夫森结和超导运行的独特特性，实现高速运行和低功耗。然而，约瑟夫森结对环境条件高度敏感。一些单磁通量子电路特有的问题，如电感耦合、偏置电流转向和磁通俘获，会严重影响其运行 [1], [2]。在 [7], 中，使用蒙特卡洛仿真来表征非理想因素下单磁通量子逻辑单元的行为，并在电路参数发生变化时，利用高斯分布对平均值进行归一化。仿真结果表明，单元包含的约瑟夫森结越多，错误率相对越高。因此，约瑟夫森结数量较多的单磁通量子电路更容易因非理想因素而出现错误。如 II-A2 节所述，单磁通量子电路的延迟主要由其逻辑深度决定。网表可以通过减少约瑟夫森结数量来减轻非理想因素的影响，但不应以牺牲逻辑深度等其他关键电路属性为代价来实现这一目标。因此，我们引入一种新指标 PND，用于评估映射方案的延迟、可靠性和组合性能。对于一个节点集为 N 的网络 $\mathcal{C}$ ，PND 定义为约瑟夫森结总数与其逻辑深度的乘积 $d_{\mathcal{C}}$。

$$P(\mathcal{C}) = (\sum_{n \in N} J_n) \times d_{\mathcal{C}} \tag{1}$$

其中，P()为PND值，而 J_n 为节点 n的约瑟夫森结数。本文使用的单元库由以下单元组成，每个单元均按以下格式列出：单元名称（约瑟夫森结数）。

单元库：D触发器（8）、二输入与门（9）、三输入与门（12）、四输入与门（15）、二输入或门（9）、三输入或门（11）、四输入或门（13）、反相器（5）、异或门（7）、分路器（3）以及本文提出的多数门（12）。

VI. 多数门模型

本节将讨论多数门，并介绍一种适用于大规模单磁通量子电路的门电路配置示例。n 输入（n 为奇数）多数函数 M 返回超过半数输入所代表的逻辑值 [26]。约瑟夫森

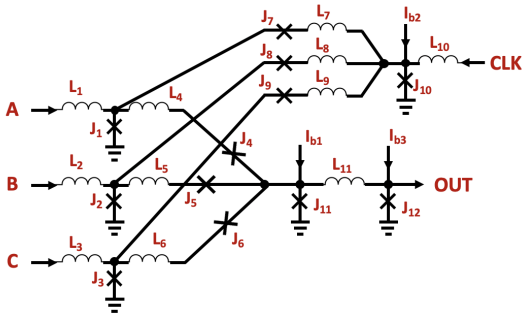

图 3： 多数门的电路配置

结的工作原理是：当其偏置电流接近临界电流时，会引发电压相位的 2π 跃变，进而触发单磁通量子脉冲。图 3 展示了一种提出的三输入单磁通量子多数门。该门由三个并联的超导量子干涉器件（SQUID）环组成 [1]、 $S1$： $\{J1, L4, J4, J11\}$、 $S2$： $\{J2, L5, J5, J11\}$和 $S3$ ： $\{J3, L6, J6, J11\}$。这些环终止于接地约瑟夫森结 J11，该结驱动输出端口 OUT。

所提出的三输入多数门的工作方式与单磁通量子三输入与门 [27]类似。然而，与三输入与门相比，多数门中的直流偏置 I_{b1} 提供的电流相对较大。当输入脉冲在一个时钟周期内输入时，相应的环路跃迁到“1”状态，J11 处的总偏置电流等于 I_{b1} 加上跃迁环路中的电流。在初始状态下，所有环路均初始化为“0”。如果一个时钟周期内只有一个输入脉冲或没有输入脉冲，J11 中的偏置电流就达不到其临界电流。因此，时钟脉冲无法诱发 J11 处的2π 跃迁。当单元接收到两个或更多输入脉冲时，由于直流偏置 I_{b1} 提供了较大的电流，总偏置电流接近或超过其临界值。此时，时钟脉冲将在 J11 处诱发一个单磁通量子脉冲，使 OUT处的输出变为“1”。所提出门的仿真如图 4 所示，其工作过程表示为

$$M(A, B, C) = AB + BC + AC$$

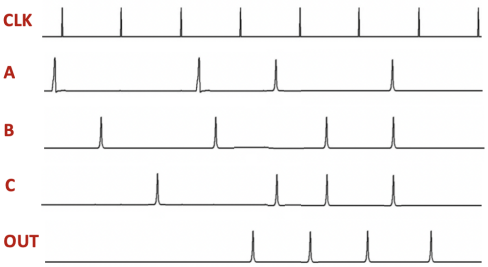

图 4： 所提出的 3 输入多数门的工作原理， A、 B 和C 为输入， CLK 为时钟输入， OUT 为输出

VII. 单磁通量子逻辑综合框架

对于给定的布尔表达式： $S = (ab + bc + ac)d$。我们希望将其映射为单磁通量子电路。如图 5 所示，最先进的映射器（SFQmap [20]）生成了左侧电路，其 PND值为 201，逻辑深度为 3。然而，也可能找到更优的映射方案

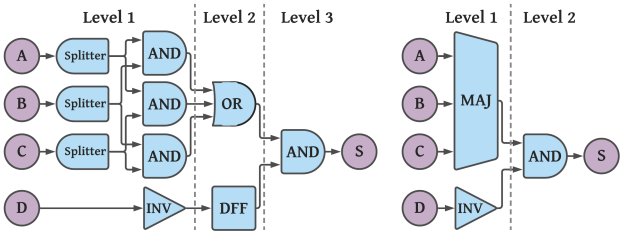

图 5： $S = (ab + bc + ac)d$的两种映射方案。左侧方案由 SFQmap [20]，生成，其 PND值为 201，最大逻辑层级为 3，且需要 1 个路径深度平衡触发器；右侧方案则可分别将上述 3 项指标降低至 52、2、0。

该方案具有更低的 PND值和更优的逻辑优化，如图 5 右侧的图所示。原因在于，目前最先进的技术映射器尚未实现任何用于减少约瑟夫森结数量并纳入多数门的算法，更不用说优化 PND 这样一个多目标值了。我们的框架能够生成具有最小 PND值且路径深度平衡开销较低的新型映射方案。假设对于网络 $\mathcal{C}$，其节点集为 N， $\mathcal{K}_n$ 是节点 $n \in N$ 的所有 K 可行切割的集合， c_n^{j*} 是切割 $c_n^j \in \mathcal{K}_n$ 的再生成割电路，那么我们将节点集为 N 的网络 $\mathcal{C}$ 的目标形式化为：

$$\min(P(\sum_{\forall n \in N}(c_n^{rep}))) \tag{2}$$

其中， $P()$ 是式 (1) 中的 PND值函数。在下一节中，我们将首先详细介绍针对所提出的 PND 值最小化问题的贪心映射算法，随后对该映射算法进行理论分析，并介绍在我们的框架中针对单磁通量子独有特性所做的修改流程。

A. 映射优化

在我们的框架中，我们采用以下启发式贪心方法来降低网络的 PND值。网络映射的最优结果定义为：遍历网络中的所有节点，并使用其代表割电路对每个节点进行修改后得到的结果。代表割 c_n^{rep} 是指根据最优布尔函数重新生成后，与原始切割 c_n^j 相比，PND值降幅最大的切割。该方法分解为以下子任务。

(1) 查找 $\mathcal{K}_n$：我们在第二节 B 中解释了 K可行切割。算法 1 对根节点 n 的上游节点执行广度优先搜索，并收集边界为 K 的所有可能割电路，具体来说，每个切割的叶节点数量 c_n^j 都在范围 $[2, K]$内。假设 I_n^j 表示切割 $c_n^j \in \mathcal{K}_n$的叶节点集合。通常，当搜索到 PI节点、分流器节点，或叶节点数量超过 K (size(I_n^j) $> K$) 时，搜索就会终止，因为重新生成的割电路 c_n^{j*} 仅保留与其前一个割电路相同的叶节点和根节点 c_n^j。如果 c_n^j 内部包含一个分流器节点，且该分流器的某些分支被省略，那么后续的重新生成过程会影响未包含分支中的逻辑值，因为该节点可能被移除或重新连接。我们的算法是一种启发式算法，能够检测分流器之后的可能切割。图6中的切割 c_H^5 就是一个例子。

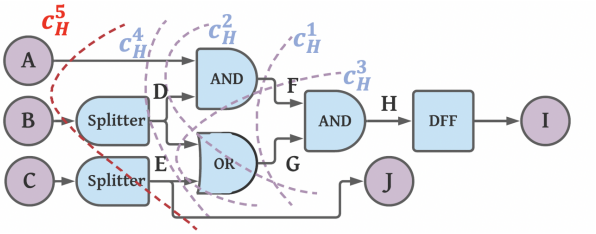

图6： 节点 H 的 3-可行割，c_H^1：{F, G}，c_H^2：{A, D, G}，c_h^3：{D, E, F}，c_h^4：{A, D, E}，c_H^5：{A, B, E}

算法 1： 查找所有 K可行割

输入： 根节点 n，割参数 K
输出： 集合 $\mathcal{S}$，每个切割 c_j对应的所有叶节点集合
$\mathcal{U}$：扇入集合；N：节点集；g()：门类型
集合 curPI = $\{n\}$，idx = 0
当 $size(curPI) > idx$ **时**
  n_c =当前PI[索引]
  s =当前PI的大小 + ($\mathcal{U}_{n_c}$) 的大小 - 1
  添加所有 $\mathcal{U}_{n_x}$ 到 N_{in}, ($n_x \in curPI$)
  对每个 $i \in N_{in}$ **执行**
    如果 $\exists j = i$；$j \in N_{in}$，$\mathcal{U}_a$；$i \in \mathcal{U}_b$；a，b ∈当前 PI **则**
      将 $\mathcal{U}_a$，$\mathcal{U}_b - i$ 添加到 N_x，并将 a、b 添加到 N_d
      从 N_{in} 中移除 i、j
  end
  如果 当前 PI-N_d + N_x ≤k **则**
    当前 PI = 当前 PI - N_d + N_x
    将当前 PI 添加到 $\mathcal{S}$
  否则如果 $g(n_c) \notin$ {PI/分路器} 与 $s \leq K$ **则**
    将 $\mathcal{U}_{n_c}$ 添加到 curPI，从 curPI 中移除 n_c
    将 curPI 添加到 $\mathcal{S}$
  else
    索引++
end

(2) 为每个切割查找最优布尔函数 $c_n^j \in \mathcal{K}_n$： 在第四节中，我们讨论了最先进映射算法的缺陷。我们提出了一种基于奎因-麦克拉斯基算法[28] 的新型布尔函数最小化算法，以克服这些缺陷。我们的优化算法更具优势，因为它将异或门和多数逻辑纳入布尔代数。此外，在许多基于多数逻辑的研究中 [10]，[12]，[13]，，所有与/或门都被多数门取代。尽管这些基于多数逻辑的映射算法已展现出显著的性能提升，且第六节证明我们提出的多数门与三输入与门具有相同的开销，但这并不意味着多数门是所有逻辑优化的万能方案。例如，表示一个异或门函数需要 3 个多数门和 2 个反相器。如果插入分路器并进行路径深度平衡，开销还会进一步增加。在我们的框架中，只有在特定的最佳情况下，我们才会采用包含多数门的切割。

为开始进行逻辑优化，我们对切割 c_n^j 进行模拟，以确定其所有主蕴含项。假设切割 c_n^j有三个叶节点：{a、b、c}，且每一位均用 0 或 1 表示。我们采用标准表示法来表示逻辑，即乘积之和（SOP） [29]。假设蕴含项集合如下：

$$c_n^j = \{111, 110, 101, 011, 001, 000\}$$

蕴含项 {110} 表示 c_n^j = 1，当 a = 1、b = 1且 c = 0 时。每个蕴含项中比特的数量和顺序均不可更改。然后，我们的算法评估每个主蕴含项之间的关系，并将其组合起来，生成本质主蕴含项。最终的本质主蕴含项集合即为切割的最优布尔函数，其形式为SOP。算法2中使用以下关系来验证蕴含项：格雷码对、异或/同或逻辑和多数逻辑。格雷码对检查用于判断两个蕴含项之间是否恰好有一位不同，并将这对互补比特替换为无关项符号“-”。多数逻辑检查会同时比较三个蕴含项，并将这些蕴含项中可实现多数逻辑的比特替换为“⋆”。异或/同或逻辑检查与多数逻辑检查类似，但每次处理两个蕴含项。“⊕”表示具有异或关系的比特，而“⊖”表示具有同或关系的比特。组合过程首先进行格雷码对检查，直至在此检查阶段中所有蕴含项均成为本质主蕴含项；随后，再对输出蕴含项进行另外两种逻辑检查（多数逻辑和异或/同或）。这两种检查互斥：若某一检查修改了某个蕴含项，另一检查就会忽略该蕴含项。我们的算法将前述蕴含项集合缩减为以下本质主蕴含项。

$$c_n^j = \{11-, 1-1, -11, 00-\} = \{\star\star\star, 00-\}$$

这些蕴含项表示的最优布尔函数如下：

$$c_n^j = M(a, b, c) + \overline{ab}$$

算法2： 蕴含项检查

输入： 蕴含项字符串 s1、s2、s3
函数 格雷码检查($s1$, $s2$)：
  $flag = 0$，$idx = 0$
  对于每个位 $i \in s1$ **执行**
    如果 $s1[i] \neq s2[i]$，**则**
      $flag$++，$idx = i$
  如果 $flag == 1$，**则**
    返回将 s2 合并到 s1 和 s1[idx]→ −
函数 异或/同或逻辑检查($s1$, $s2$)：
  f1 = 0, f2 = 0, s = {}
  对于 每个位 $i \in s1$ **执行**
    如果 ($s1[i]$, $s2[i]$) == (0, 1) **则**
      $f1$++，将 i 添加到 s
    如果 ($s1[i]$, $s2[i]$) == (1, 0) **则**
      $f2$++，将 i 添加到 s
  如果 $f2 == 2$ 或 $f2 == 2$，**则**
    返回 将 s2 合并到 s1，并针对 $j \in s$ s1[j]→ ⊖
  如果 $f1 == 1$ 与 $f2 == 1$，**则**
    返回 将 s2 合并到 s1，并针对 $j \in s$ s1[j]→ ⊕
函数 多数逻辑检查($s1$, $s2$, $s3$)：
  $f1 = 0$, $f2 = 0$, $f3 = 0$, s = {}
  对每个比特 $i \in s1$ **执行**
    如果($s1[i]$, $s2[i]$, $s3[i]$) == (1, 1, −) **则**
      $f1$++，将 i 添加到 s
    如果($s1[i]$, $s2[i]$, $s3[i]$) == (1, −, 1) **则**
      $f2$++，将 i 添加到 s
    如果($s1[i]$, $s2[i]$, $s3[i]$) == (−, 1, 1) **则**
      $f3$++，将 i 添加到 s
  如果 $f1 == 1$ 与 $f2 == 1$ 与 $f3 == 1$ **则**
    返回将 s2、s3 合并到 s1，并针对 $j \in s$ s1[j]→ ⋆

(3) 从 c_n^{j*} 的最优布尔函数重新生成c_n^j： K 值仅限制 K可行切分搜索中的叶节点数量，这意味着每个叶节点可以处于不同的逻辑层级。考虑图6中的 c_H^2，重新生成的割电路 c_H^{2*} 是一个三输入与门，其最优布尔函数来自 $H = ADF$ c_H^2。这似乎

通过将其降低到12，大幅改善了PND值。然而， c_H^{2*} 会增加约瑟夫森结的开销，因为需要2个D触发器来平衡节点A和D的路径深度。PND值骤升至56，甚至高于 c_H^2 （42）。因此，必须在 c_n^j中考虑叶节点的逻辑层级，重新生成 c_n^{j*}。

（4）比较 c_n^{j*}与 c_n^j，更新 c_n^{rep}，并将节点 n 修改为其代表割 c_n^{rep}： 重新生成的切割 c_n^{j*}和切割 c_n^j 表示一对割对。每对割对的改进程度通过以下公式计算。

$$d_n^j = P(c_n^j) - P(c_n^{j*}) \tag{3}$$

其中， d_n^j 是 c_n^j与 c_n^{j*}之间PND值的改进值，P()是根据式（1）计算出的PND值。如果 d_n^j在所有割对（c_n^i、 c_n^{i*}）中具有最大值，则 $c_n^i \in \mathcal{K}_n$，这表明与 c_n^j相比， c_n^{j*}的PND值有最大的正向改进，随后将 c_n^{j*}更新为代表割 c_n^{rep}。如果没有任何一个 c_n^{j*}能够改善节点 n的PND值，则将 c_n^{rep}设为null。遍历节点 n的所有切割 $c_n^j \in \mathcal{K}_n$ 后，将其修改为代表割 c_n^{rep}，以最小化网络 $\mathcal{C}$ 的PND值。如果网络中没有任何节点被修改，则表明经过预处理阶段后，该网络已经是最优映射解。

B. 理论分析

网络 $\mathcal{C}$ 的最优映射解由以下方程形式化表示:

1) 计算节点 n 的代表割 c_n^{rep}:

$$c_n^{rep} = c_n^{j*} \quad | \quad d_n^j = \max_{\forall c_n^i \in \mathcal{K}_n} (d_n^i) \tag{4}$$

2) 使用其代表割更新每个节点:

$$\mathcal{C} = \sum_{\forall n \in N} (c_n^{rep}) \tag{5}$$

定理 1.目标函数式 (2) 是次模的。

证明。 对于给定网络 $\mathcal{C}$ 及其节点集 N，我们定义两个可由本框架生成代表割的节点集： $\alpha \subseteq \{a_1, \ldots, a_n\}$、 $\beta \subseteq \{b_1, \ldots, b_n\}$，其中两个集合 $\alpha, \beta \subseteq \Omega$，且 Ω 是该问题的解空间；我们将 $f(x)$ 定义为式 (2) 中的目标函数。如果 $\alpha \cap \beta = \epsilon$，则存在 $\epsilon = \emptyset$。

$$\begin{aligned} & f(\alpha \cap \beta) + f(\alpha \cup \beta) \\ &= \sum_{n \in N} \{|P(\alpha(n))| + |P(\beta(n))| - |P(\epsilon(n))| + |P(\epsilon(n))|\} \\ &= \sum_{n \in N} |P(\alpha(n))| + \sum_{n \in N} |P(\beta(n))| \\ &= f(\alpha) + f(\beta) \end{aligned}$$

根据前述方程，我们可以推断目标函数是次模的。由于对于任意两个集合α， $\beta \subseteq \Omega$， $f(\alpha) + f(\beta) = f(\alpha \cap \beta) + f(\alpha \cup \beta)$。

定理 2。目标函数式（2）是单调的。

证明。 对于一个节点集为 N 的网络 $\mathcal{C}$ ，我们定义一个任意节点集可以通过我们的框架生成代表割 $\alpha' = \alpha \cup \tau$，由此可以得到 $f(\alpha \cup \tau) - f(\alpha) \geq 0$。

$$\begin{aligned} f(\alpha \cup \tau) &= \sum_{n \in N} \{|P(\alpha'(n))| \\ &= \sum_{n \in N} \{|P(\alpha(n))| + |P(\tau(n))|\} \\ &= \sum_{n \in N} |P(\alpha(n))| + \sum_{n \in N} |P(\tau(n))| \\ &= f(\alpha) + f(\tau) \end{aligned}$$

C. 预处理

我们在上一节介绍了映射算法。然而，在进行映射优化之前，框架需要对网表进行预处理，将其转换为基于SFQ电路的网络，其中节点表示网表中的逻辑门。该表示方案的预处理阶段需要执行以下两个步骤：*1)* 门转换：将网表中的所有逻辑门转换为单元库中已有的单元。假设输入网表包含单元库中没有的与非门。因此，每个与非门都会转换为一个与门，并串接一个新的反相器节点。同理，转换函数也会修改同或门和或非门。此外，由于单元库的限制，与门/或门的扇入不能超过四个。扇入更大的门会由扇入较小的对应门转换为门树；*2)* 分路器插入： 第二节A-1介绍了SFQ电路的扇出特性。对于每个扇出大于二的节点，都必须插入一个分路器二叉树。算法3详细描述了门转换和分路器插入。

算法3： SFQ电路独特特性的修改

输入： 网络 $\mathcal{C}$的节点集合 N，
$\mathcal{U}$: 扇入； $\mathcal{D}$: 扇出； l: 逻辑层级；g(): 门类型

函数路径深度平衡(N):
- **对于**每个节点 $n \in N$ **执行**
  - **如果**$g(n) \neq$ 分路器 **则**
    - l_{max} = 最大值($l_j, j \in \mathcal{U}_n$)
    - **对于**每个扇入 $i \in \mathcal{U}_n$ **执行**
      - 在 i处插入 $(l_{max} - l_i)$ 个D触发器

函数 分路器插入($\mathcal{N}$):
- **对于**每个 $n \in N$，**执行**
  - **如果** $g(n) \neq$ 分路器 且 $\mathcal{D}_n > 1$ **则**
    - $c = n$
    - **对于** 每个扇出 $o_j \in \mathcal{D}_n$ **执行**
      - 在 c和 o之间插入一个**分路器** s^j $_j$
      - 创建边 $c \rightarrow s^j$, $s^j \rightarrow o_j$
      - **如果** $j = size(D_n) - 1$ **则**
        - 创建边 $s^j \rightarrow o_{j+1}$
      - $c = s^j$

函数门转换($\mathcal{N}$):
- **如果**$g(n) \in$ {与非门/或非门/同或门} **则**
  - 插入 $n*$ 在 n 与其扇出 $\mathcal{D}$之间 $_n$
  - g(n_*) $\in$ {与门/或门/异或门}, g(n) $\in$ 反相器
- **如果** $g(n) \in$ {与门/或门} 且大小 $(\mathcal{U}_n) > 4$ **那么**
  - 在 n 与 $\mathcal{D}_n$之间插入 n_*， g(n_*)$\in${与门/或门}
  - **对于** $f_j \in \mathcal{U}_n$ 且 $j \leq 4$ **执行**
    - 移动边 $f_j \rightarrow n_*$

D. 后处理

在后处理阶段，框架需要根据第二节A-3中的特性修改网络：每个门的所有扇入都应具有相同的延迟（时钟相位）。为避免重复进行切分搜索和计算，这项修改交由优化后的处理流程执行。通过路径深度平衡，D触发器与其扇入节点具有相同的切割，具体可参见图6中的节点I和H。路径深度平衡的伪代码也列于算法3中。

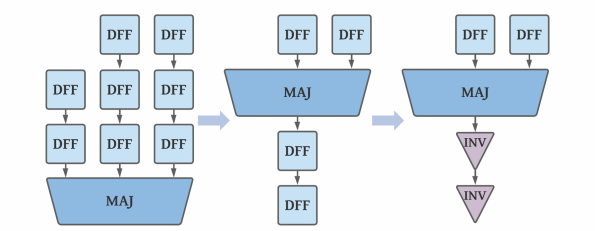

图 7 2, 3, **：** 合并与替换示例，目标门的扇入分别具有 D触发器。此过程在此特定情况下可将#(D触发器 反相器)的数量减少 50%。

此外，由于 D触发器消耗的#约瑟夫森结多于反相器，因此采用合并与替换过程，进一步减少路径深度平衡D触发器的消耗。在此阶段，框架仅处理路径深度平衡算法插入的D触发器；输入网表中原有的 D触发器不会被更改，以保留网络的正确逻辑函数。对于每个单元，如果其所有扇入都至少插入了y 个D触发器， y 就是该门共享的D触发器数量。因此，其所有扇入都可以移除 y 个D触发器，然后在其扇出上插入 y个D触发器，且不改变逻辑。同时，对于每条路径，如果插入了 x 个路径深度平衡D触发器（$x > 2$），则将 $x-$（x%2）个D触发器替换为反相器。如图7所示，目标门的三个扇入分别有 2、3 和 3 个D触发器。通过将两个D触发器移至目标门的扇出路径，可以简化此结构，使扇入上的D触发器数量（#D触发器）分别降至 0、1 和 1，然后将扇出路径上的D触发器替换为反相器。此步骤可确保路径深度平衡D触发器的总数尽可能少，同时减轻电路中非理想因素的影响。

VIII. 评估

在本节中，我们提供两种实验设置和实验结果，以验证所提框架的有效性。我们选取若干 ISCAS 算术基准电路 [30]，用于测试所开发的框架。所提框架的伪代码如算法 4 所示。首先，我们展示

表I： 所提框架在不同K 值下的实验结果。 3^- 表示 $K = 3$，且禁用了合并与替换过程。

K	逻辑深度	#JJs	PND($\times 10^5$)	#(D触发器+反相器)
$K = 3^-$	46.7	108662	63.48	11247
$K = 4^-$	44.8	106253	62.42	10943
$K = 3$	46.1	81401	45.71	11032
$K = 4$	44.8	80038	45.0	10733

算法4： 映射优化

输入： 网络 $\mathcal{C}$，割参数 K
输出： 映射后的电路 $\mathcal{C}_{map}$
l：逻辑层级； I：叶节点集合；P()：PND值
对于每个节点 $n \in N$ **执行**
  计算 n 的所有 K可行切割 $\mathcal{K}_n$，使用 K
  $d = 0$，代表割 c_n^{rep} = NULL;
  对每个切割 $c_n^j \in \mathcal{K}_n$ **do**
    计算切割 c_n^j 的最优布尔函数 F_j
    生成 c_n^{j*}，使用 F_j 和 l_{I^j}
    如果 $P(c_n^j) - P(c_n^{j*}) > d$ **则**
      $d = P(c_n^j) - P(c_n^{j*})$， $c_{rep} = c_n^{j*}$
    end
  end
  如果 $c_n^{rep} \neq$ 为空，**则**
    通过切割 c_n^{rep} 修改了 $\mathcal{C}$
  end
end

不同 K 值的实验，并比较不同 K 值下的性能，以及未经过合并与替换过程时的相应结果。然后，我们将所提框架与最先进的技术映射器进行比较。

1) K **值测试：** 在第七节 A 部分，我们介绍了 K可行切割查找算法和映射优化算法。现在，我们对所提框架在一系列 K 值下的输出，以及合并与替换过程的影响进行实证评估。 K值表示切割电路的最大扇入；因此，最大 K 值设为 4，这对应于我们单元库中单元的最大扇入。表 I 展示了 15 个基准电路（列于表 II）的平均结果，其中框架将 K 值设为 3 或 4，并与未执行合并与替换过程时的相应结果进行比较。正如第七节所述，我们开发的技术映射器着重改进单磁通量子电路的四项关键参数，包括逻辑深度、#约瑟夫森结、PND 和 #(D触发器+反相器)。通常，#D触发器指标用于确定路径深度平衡所需的开销。由于我们的后处理算法会在允许的条件下以D触发器替换反相器，从而实现路径深度平衡，因此在评估中以 #(D触发器+反相器) 取代 #D触发器作为指标。与 $K = 3$相比，采用 $K = 4$ 的框架改善了所有指标。这是因为，具有较大 K 值的割搜索算法所得到的结果集，是较低 K值对应的切割电路集合的超集。具体而言，该算法能够为根节点找到更多切割电路，并重新生成 PND 值更低的代表割电路。在逻辑结果相同的条件下，合并与替换过程取得了相当的性能：具有较多 #约瑟夫森结的D触发器被所需 #约瑟夫森结较少的反相器替换，而分支中的D触发器和反相器则被合并到根节点。平均而言，此操作使 #约瑟夫森结减少了 24.88%。

2）与最先进技术映射器的比较： 表 II 展示了我们的框架与两个基准映射器的实验结果。第一个基准映射器以输入网表 $\mathcal{B}_1$ 为基础，应用算法3中的 SFQ 特性修改方案来插入 D触发器和

表 II： 实验结果：所提框架 $\mathcal{F}$ ($K = 4$) 与基线方法（$\mathcal{B}_1$ ： 输入网表， $\mathcal{B}_2$ ： SFQmap [20]）

	逻辑深度			#JJs			PND($\times 10^5$)			#(D触发器+反相器)		
电路	$\mathcal{B}_1$	$\mathcal{B}_2$	$\mathcal{F}$	$\mathcal{B}_1$	$\mathcal{B}_2$	$\mathcal{F}$	$\mathcal{B}_1$	$\mathcal{B}_2$ ：	$\mathcal{F}$	$\mathcal{B}_1$	$\mathcal{B}_2$ ：	$\mathcal{F}$
b04	51	40	33	35297	30141	18532	18.0	12.1	6.12	3855	3294	2332
b11	61	35	36	48190	31251	22636	29.4	10.9	8.15	5480	3267	3137
c880	42	35	23	19764	14551	8380	8.3	5.1	1.93	2172	1566	1096
c1355	44	14	12	33995	10915	7587	15.0	1.53	0.91	3792	1024	896
c1908	68	52	50	70512	58294	25983	47.9	30.3	13.0	8436	6773	3963
c2670	42	38	33	30768	29475	21998	12.9	11.2	7.3	3205	2970	2601
c3540	75	64	58	65143	58443	39179	48.9	37.4	22.7	7215	6357	5320
c5315	73	53	52	114273	91323	66002	83.4	48.4	34.3	12783	9784	9257
c6288	246	168	168	402824	248038	166283	991	416.7	279.4	47935	28874	28159
c7552	68	49	48	127514	106264	79554	86.7	52.7	38.2	13998	10981	10209
s1423	67	59	57	82400	55746	30214	55.2	32.9	17.2	9723	6518	5174
s1494	18	16	15	19328	20447	14721	3.48	3.27	2.21	1722	1774	1569
s5378	33	30	26	58191	55137	37101	19.2	16.5	9.65	6490	6097	4354
s35932	42	16	13	1121734	310597	239681	471.1	49.7	31.2	126866	26715	24724
s38584	70	63	48	815009	727571	422713	570.5	458.3	202.9	89538	79792	58209
Avg	66.7	48.8	44.8	202996	12312	80037	164.1	79.1	45.0	22881	13017	10773
Imp(Avg)	↑32.8%	↑8.2%		↑60.5%	↑35.0%		↑72.5%	↑43.1%		↑53.1%	↑17.5%	

分路器。第二个基准映射器是 SFQmap $\mathcal{B}_2$ [20]，其 K 值设为 3，以与原有工作保持一致。为确保公平性，我们为所有映射器统一采用了第 V 节所述的单元库。本实验评估了与 K 值测试相同的指标。与$\mathcal{B}_1$相比，我们的框架在所有基准电路的这些关键参数上均取得了显著改进。逻辑深度、约瑟夫森结数量、PND值以及 #(D触发器+反相器) 的平均降幅分别为 32.8%、60.5%、72.5% 和 53.1%。这主要是因为 $\mathcal{B}_1$ 未能提供有效的映射算法来降低开销。

总体而言，我们的框架 $\mathcal{F}$ 在约瑟夫森结数量、PND值以及D触发器和反相器总数方面均优于基准SFQmap$\mathcal{B}_2$ 。这验证了我们的预期：所提出的框架能够减少约瑟夫森结的数量（#JJs），以及约瑟夫森结总数与逻辑深度的乘积（PND）。平均而言，与 $\mathcal{B}_2$——一种面向D触发器数量和逻辑深度的优化映射算法——相比，$\mathcal{F}$ 将所有基准电路的逻辑深度降低了8.3%。这体现了我们算法的优势：该算法纳入多数门，并根据原始电路中叶节点的逻辑层级重新生成切割电路，而不是仅利用超级门最小化切割电路的关键路径。在电路b11中，$\mathcal{F}$ 生成的映射结果的逻辑深度略高于 $\mathcal{B}_2$。这表明， $\mathcal{F}$ 是一种面向多目标优化的映射器，会综合考虑整体指标来寻找映射结果。因此，它可以牺牲逻辑深度，以换取PND值更低的结果。尽管它生成的结果逻辑深度更高，但在b11中，PND值和#JJs均显著降低。

为了分析与最先进的技术映射器相比所观察到的改进是否主要归功于我们的映射算法，我们在 $\mathcal{F}$ 中禁用合并与替换，然后将其与

表 III： 所提框架的实验结果 $\mathcal{F}$ ($K = 4$⁻)与基线方法（$\mathcal{B}_2$：SFQmap [20]）

映射器	逻辑深度	#JJs	PND($\times 10^5$)	#(D触发器+反相器)
$\mathcal{F}$	44.8	106253	62.42	10943
$\mathcal{B}_2$ ：	48.8	123212	79.10	13017
平均改进	↑8.2%	↑13.7%	↑21.1%	↑15.9%

SFQmap $\mathcal{B}_2$ 再次进行比较。由于合并与替换操作也有助于减少约瑟夫森结数量，这一点在 K 值测试中有所体现。表 III 汇总了前述 15 个基准电路的平均实验结果。虽然禁用合并与替换过程后，约瑟夫森结数量方面的优化受到抑制，但逻辑深度、PND值和 #(D触发器+反相器) 等其他关键指标仍持续显著改善。

IX. 结论

本文提出了一种全面的逻辑综合框架，能够将网表映射为与单磁通量子技术兼容的最优网表；同时还提出了单磁通量子电路中的多数门电路配置，并将其应用于我们的框架。与基准方法相比，我们提出的映射优化算法在所有测试基准电路中，均能有效降低路径深度平衡所带来的开销，以及约瑟夫森结数量与逻辑深度的乘积（PND），证明了其优越性，从而减轻了这些单磁通量子电路中的非理想因素。与最先进的技术映射器相比，实验结果表明，在15个基准电路中，我们的框架平均分别将逻辑深度、#(DFFs+INVs)、#JJs和PND降低了8.2%、17.5%、35.0%和43.1%。

参考文献

[1] K. K. Likharev 和 V. K. Semenov，“RSFQ逻辑/存储器系列：一种适用于亚太赫兹时钟频率数字系统的新型约瑟夫森结技术”，IEEE应用超导学报，第1卷，第1期，第3–28页，1991年。[2] P. Bunyk、K. Likharev 和 D. Zinoviev，“RSFQ技术：物理学与器件”，《国际高速电子与系统杂志》，第11卷，第01期，第257–305页，2001年。[3] K. Gaj、Q. Herr 和 M. Feldman，“RSFQ电路的参数变化与同步”，载于《物理学会会议系列》，第148卷，IOP PUBLISHING LTD，1995年，第1733–1736页。[4] I. V. Vernik、Q. P. Herr、K. Gaij 和 M. J. Feldman，“RSFQ电路局部时序参数变化的实验研究”，IEEE应用超导学报，第9卷，4344，1999 2，4341 no. pp. – .[5] S. Narasimha、B. Jagannathan、A. Ogino、D. Jaeger、B. Greene、C. Sheraw、K. Zhao、B. Haran、U. Kwon、A. Mahalingam 等，“面向移动与高性能计算应用的7纳米CMOS技术平台”，载于2017年IEEE国际电子器件会议（IEDM）。IEEE，2017年，第29–5页。[6] S. K. Tolpygo、V. Bolkhovsky、T. J. Weir、L. M. Johnson、M. A. Gouker 和 W. D. Oliver，“用于超大规模集成电路的全平坦化深亚微米 $nb/al-alo_x/nb$ 约瑟夫森结的制造工艺与特性”，IEEE应用超导学报，第25卷，第3期，第1–12页，2014年。[7] F. Wang 和 S. Gupta，“RSFQ电路时序验证与延迟测试的自动测试向量生成”，载于2019年IEEE第37届超大规模集成电路测试研讨会（VTS）。IEEE，2019年，第1–6页。[8] M. E. Celik 和 A. Bozbey，“RSFQ布线单元及时钟门信号延迟、抖动和时序的统计方法”，IEEE应用超导学报，第23卷，第3期，第1 701 305–1 701 305页，2012年。[9] G. D. Hachtel 和 F. Somenzi，《逻辑综合与验证算法》。Springer Science & Business Media，2007年。[10] L. Amar´u、P.-E. Gaillardon 和 G. De Micheli，“多数反相器图：一种用于高效逻辑优化的新型数据结构与算法”，载于2014年第51届ACM/EDAC/IEEE设计自动化会议（DAC）。IEEE，2014年，第1–6页。[11] G. Krylov 和 E. G. Friedman，“异步动态单磁通量子多数门”，IEEE应用超导学报，第30卷，第5期，第1–7页，2020年。[12] L. Amaru、P.-E. Gaillardon 和 G. De Micheli，“ 多数反相器图：逻辑优化的新范式 ”，IEEE 集成电路与系统计算机辅助设计学报，第35卷，第5期，第806–819页，2015年。[13] R. Cai 、 O. Chen 、 A. Ren 、 N. Liu 、 C. Ding 、 N. Yoshikawa 和 Y. Wang，“绝热量子磁通参数超导电路的多数逻辑综合框架”，载于2019年五大湖超大规模集成电路研讨会论文集，2019年，第189–194页。[14] E. G. Friedman，“同步数字集成电路中的时钟分配网络”，IEEE会议录，第89卷，第5期，第665–692页，2001年。[15] R. L. Rudell，《面向超大规模集成电路设计的逻辑综合》，博士论文，加州大学伯克利分校，1989年。[16] A. Mishchenko、S. Cho、S. Chatterjee 和 R. Brayton，“采用优先级割的组合与时序映射”，载于2007年IEEE/ACM国际集成电路与系统计算机辅助设计会议。IEEE，2007年，第354–361页。[17] J. Cong 和 Y. Ding，“FlowMap：一种针对基于查找表的FPGA设计进行延迟优化的最优技术映射算法”，IEEE集成电路与系统计算机辅助设计学报，第13卷，第1期，第1–12页，1994年。[18] S. Yamashita、K. Tanaka、H. Takada、K. Obata 和 K. Takagi，“用于综合RSFQ电路的基于转换的框架”，载于亚洲及南太平洋设计自动化会议，2006年。IEEE，2006年，第7–页。[19] G. Pasandi 和 M. Pedram，“PBMap：一种面向单磁通量子逻辑电路的路径均衡技术映射算法”，IEEE应用超导学报，第29卷，第4期，第1–14页，2018年。[20] G. Pasandi、A. Shafaei 和 M. Pedram，“SFQmap：一种面向单磁通量子逻辑电路的技术映射工具”，载于2018年IEEE国际电路与系统研讨会（ISCAS）。IEEE，2018年，第1–5页。[21] N. Katam、A. Shafaei 和 M. Pedram，“复杂快速单磁通量子单元设计及其在逻辑综合中的应用”，载于2017年第16届国际超导电子学会议（*ISEC*）。IEEE，2017年，第1–3页。[22] C. E. Leiserson 和 J. B. Saxe，“同步电路重定时”，*Algorithmica*，第6卷，第1期，第5–35页，1991年。[23] A. Mishchenko 等，“ABC：一种用于时序综合与验证的系统”，网址：*http://www. eecs. berkeley. edu/alanmi/abc*，第17卷，2007年。[24] C. S. Lent 和 P. D. Tougaw，“一种基于量子点计算的器件架构”，*IEEE*会议录，第85卷，第557页，1997年，第4期，第541页。[25] T. Oya、T. Asai、T. Fukui 和 Y. Amemiya，“采用一对平衡单电子盒的多数逻辑纳米器件”，《纳米科学与纳米技术杂志》，第2卷，第3–4期，第333–342页，2002年。[26] T. Sasao，用于逻辑综合的开关理论。Springer Science & Business Media，2012年。[27] N. K. Katam 和 M. Pedram，“单磁通量子电路的逻辑优化、复杂单元设计与重定时”，*IEEE*应用超导学报，第28卷，第7期，第1–9页，2018年。[28] E. J. McCluskey，“布尔函数最小化”，《贝尔系统技术期刊》，第35卷，第6期，第1417–1444页，1956年。[29] G. De Micheli，数字电路的综合与优化。McGraw Hill，1994年，图书。[30] F. Brglez、D. Bryan 和 K. Kozminski，“时序基准电路的组合特征”，载于*IEEE*国际电路与系统研讨会。IEEE，1989年，第1929–1934页。