

정확한 결함 허용 양자 컴퓨팅을 위한 논리 큐비트 측정 장치의 구조 및 중요성 분석

변일권, 김장우

서울대학교 전기정보공학부

e-mail : ik.byun@snu.ac.kr, jangwoo@snu.ac.kr

Analyzing Logical Measure Unit Architecture for Accurate Fault-tolerant Quantum Computing

Ilkwon Byun, Jangwoo Kim

Department of Electrical and Computer Engineering
Seoul National University

Abstract

Fault-tolerant quantum computing is essential to achieve a true sense of quantum supremacy. With the recent research effort, architects have revealed a fault-tolerant quantum control processor's hardware components and scalability. However, there still has been a lack of detailed discussion for the logical measure unit, especially regarding the logical-qubit sign tracking. This paper clarifies the logical measure unit's microarchitecture and importance focusing on the sign tracking.

I. 서론

양자 컴퓨팅은 다양한 분야에 걸쳐 기존의 컴퓨터 시스템으로는 풀지 못하는 중요한 문제들을 풀 수 있을 것으로 촉망받는 새로운 컴퓨팅 방식이다. 그러나, 양자 컴퓨터에서 연산이 수행되는 큐비트는 근본적으로 에러에 매우 취약하기 때문에 에러를 낮추지 않고는 실제로 유용한 문제들을 푸는 것은 거의 불가능하다. 이를 해결하기 위해, 양자 오류 정정

(quantum error correction; QEC) 기술을 적용하여 큐비트의 에러를 크게 낮춘 뒤 양자 연산을 수행하는 결함 허용 컴퓨팅 (fault-tolerant quantum computing; FTQC)이 필수적이다. 따라서, 진정한 의미의 양자 우월성 [1]을 달성하기 위해서는 FTQC에 필요한 모든 기능을 지원하는 양자 컴퓨터 시스템을 개발해야 한다.

이에 따라, FTQC를 지원하는 양자 제어 프로세서와 이를 구성하는 세부 하드웨어에 대한 연구들이 활발히 진행되고 있다. 예를 들면, 최근 선행 연구 [2]는 surface code [3] 기반 lattice surgery [4] 기법을 사용해 FTQC를 수행하는 양자 제어 프로세서의 구조를 제안했다. 그러나 [2]에서 대부분의 논의는 전체 양자 제어 프로세서의 확장성에 초점을 맞추고 있어, 정확한 FTQC 수행을 위한 각 세부 하드웨어의 역할 및 중요성에 대한 논의가 부족한 부분이 있다. 특히 논리 큐비트 측정 장치 (logical measure unit; LMU)의 경우, 핵심적인 역할 중 하나인 논리 큐비트 부호 추적 (logical-qubit sign tracking) 기능은 아직까지도 구체적으로 다뤄지지 않은 상황이다.

따라서, 본 연구에서는 논리 큐비트 부호 추적 기능에 초점을 맞춰 논리 큐비트 측정 장치의 구조와 중요성에 대해 분석하고자 한다. 이를 위해 우선 기존에 논의됐던 논리 큐비트 측정 장치의 역할과 기능에 대해

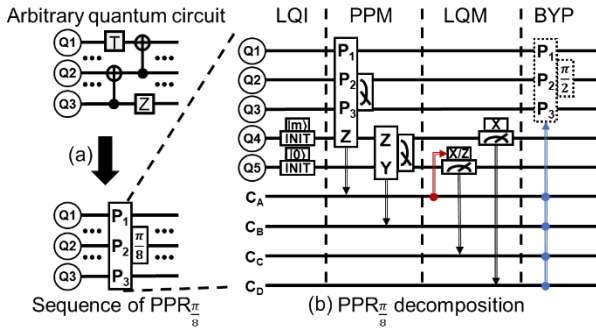


그림 1. (a) 임의의 논리 양자 회로의 $PPR_{\pi/8}$ 변환과
(b) $PPR_{\pi/8}$ 의 세부 논리 연산으로의 분해

소개한다. 그 다음, 논리 큐비트 부호가 변화하는 현상과 이를 추적하기 위해 필요한 기능과 이를 지원하는 하드웨어 구조에 대해 논의한다. 마지막으로, 논리 큐비트 부호 추적 기능의 유무에 따른 FTQC 연산의 실패율을 분석함으로써 해당 기능을 지원하는 장치의 중요성을 실험을 통해 보인다.

II. 본문

2.1 논리 큐비트 측정 장치의 역할과 기능

논리 큐비트 측정 장치 (LMU)는 물리 큐비트들의 측정값을 토대로 논리 큐비트의 측정값을 도출하는 하드웨어 요소이다. 본 단락에서는 먼저 대표적인 양자 오류 정정 기법인 surface code 기반 lattice surgery [5]를 이용하는 선행 연구 [2]에서 소개된 LMU의 역할을 구체적으로 다룬다.

2.1.1. FTQC에서 요구되는 논리 큐비트 측정

QEC가 적용된 논리 큐비트에 대해 양자 연산을 수행하기 위해서, 임의의 논리 양자 회로는 $\frac{\pi}{8}$ Pauli-product rotation ($PPR_{\pi/8}$) 양자 연산들의 나열로 변환된다. 이때, $PPR_{\pi/8}$ 는 다시 네 가지 세부 연산들로 분해되는데 여기엔 (1) 단일 논리 큐비트 측정 (logical qubit measure, LQM)과 (2) Pauli-product measure (PPM)으로 불리는 다중 논리 큐비트 측정이 포함된다. 따라서, FTQC를 위해 LMU가 수행해야 역할은 바로 LQM과 PPM 측정값의 정확한 해석이다.

2.1.2. 논리 큐비트 측정의 측정값 해석

단일 논리 큐비트 측정 (LQM). 그림 2(a)는 surface code에서 code distance (d)가 3인 논리 큐비트 패치의 예시를 보여준다. 논리 큐비트는 양자 상태 정보를 담은 데이터 큐비트 (검은색, D)과 오류 탐지에 이용되는 보조 큐비트 (흰색, X 혹은 Z)의 두 종류의 물리

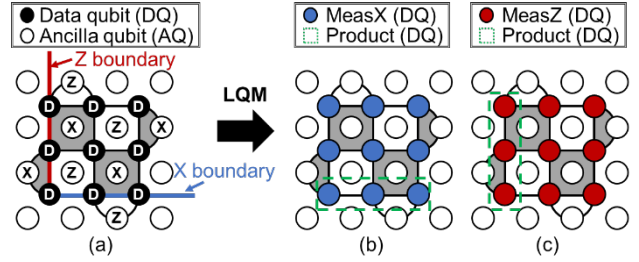


그림 2. (a) Surface code 논리 큐비트 패치의 구조 ($d = 3$)와 (b) X축 및 (c) Z축에 대한 LQM 예시

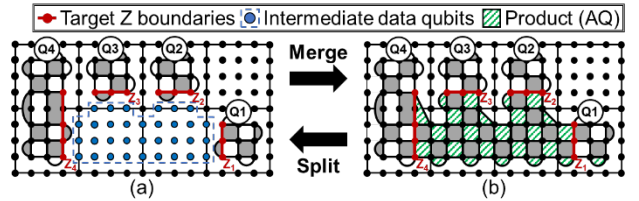


그림 3. $PPM(Z_1Z_2Z_3Z_4)$ 를 위한 lattice surgery 예시;
(a) Merge 이전과 split 이후 (b) Merge 이후

큐비트로 구성된다. 논리 큐비트는 보조 큐비트들을 인접한 데이터 큐비트들과 엮고 측정하는 error syndrome measurement (ESM)을 통해 생성된다.

LQM을 위해, 먼저 원하는 논리 큐비트의 측정 축과 동일한 축으로 모든 데이터 큐비트에 측정을 가한다. 그 다음, LMU에서 측정값들을 전달받아 측정 축에 대응되는 경계만을 선별해 곱하여 논리 큐비트의 측정값을 도출한다. 그림 2(b), 2(c)는 각각 X축과 Z축에 대한 LQM의 예시를 보여준다.

다중 논리 큐비트 측정 (PPM). PPM은 논리 큐비트 패치들을 ESM을 통해 합쳤다가 (merge) 분리하는 (split) lattice surgery 기법을 통해 구현된다. 그림 3은 $PPR_{\pi/8}(Z_1Z_2Z_3)$ 에서 요구되는 $PPM(Z_1Z_2Z_3Z_4)$ 의 수행 과정을 예시로 보여준다.

먼저, PPM의 목표 Pauli product (i.e., $Z_1Z_2Z_3Z_4$)에 대응되는 논리 큐비트들의 경계들을 연결하는 영역에 존재하는 데이터 큐비트들을 초기화한 뒤, 이들을 포함하여 ESM을 수행한다 (merge). 그 다음, 경계들을 연결하는 영역의 데이터 큐비트들을 측정한 뒤 이들을 다시 제외하고 ESM을 수행한다 (split). 이 과정이 논리 큐비트들을 목표 Pauli product에 대해 측정하는 것과 같은 효과를 가지기 때문에 이를 PPM이라고 부른다. 그림 3의 예시에서는 데이터 큐비트들을 $|+\rangle$ 로 초기화하고 X 축에 대해 측정한다.

PPM의 측정값은 merge 됐던 논리 큐비트 경계들 상의 모든 데이터 큐비트 측정값을 곱한 것과 같다. 이때, LMU는 merging ESM에서 얻은 보조 큐비트들의 측정값을 선별하여 곱함으로써 이 값을 얻어낼 수 있다. 예를 들어, 그림 3(b)의 예시의 경우 경계들을

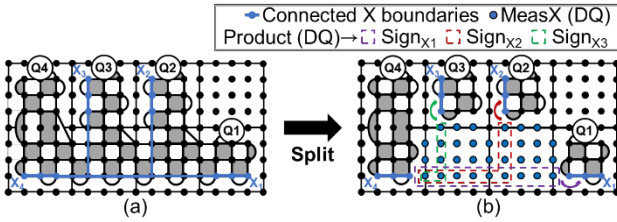


그림 4. $PPM(Z_1Z_2Z_3Z_4)$ 를 위한 lattice surgery 예시; (a) Merge로 연결된 X boundary들과 (b) Split을 통해 발생하는 X sign flip

연결하는 영역에 포함된 Z 보조 큐비트들 (녹색 빗금)의 측정값들을 전부 곱한 것이 측정값이 된다.

논리 측정에서 이뤄지는 양자 오류 정정. 양자 오류 정정은 데이터 큐비트에서 발생한 에러를 알아내고 (error decoding) 이를 수정하는 연산을 가하는 것이다. 그러나, 실제로는 발생한 에러를 직접 수정하는 대신 이를 기록해 뒀다가 논리 측정 단계에서 반영하는 기법을 사용한다. 예를 들면, Z축에 대한 논리 큐비트 측정값을 해석하는 경우, LMU는 Z boundary에 기록된 X 에러의 수만큼 측정값을 뒤집어 줌으로써 에러를 수정한다. 이때 누적된 에러의 기록을 Pauli frame이라고 한다 [6].

$PPR_{\pi/2}$ 추적. LMU는 $PPR_{\pi/8}$ 를 수행하는 과정에서 LQM과 PPM 측정값을 바탕으로 추가 $PPR_{\pi/2}$ 연산이 발생하는 지 판단 (그림 1 BYP)하고 이를 기록한다. 만약 이전까지 누적된 $PPR_{\pi/2}$ 의 Pauli-product가 PPM 혹은 LQM의 목표와 commute하지 않는 경우, LMU는 최종 해석 단계에서 논리 측정값을 뒤집어준다.

2.2 논리 큐비트의 측정 부호 추적

앞선 모든 기능을 LMU가 완벽히 지원하더라도 lattice surgery로 인해 변하는 논리 큐비트의 측정 부호를 추적 (logical-qubit sign tracking) 하지 못하면 FTQC를 정확히 수행하지 못한다.

논리 큐비트의 측정 부호 변화. 먼저, 그림 4(a)와 같이 PPM의 목표 Pauli product를 따라 merge를 수행한 상황에는 반대 쪽 경계들이 전부 연결되는 것을 확인할 수 있다. 이어지는 split에서 연결됐던 경계들은 다시 분리되는데, 주목할 점은 이때 연결됐던 경계들의 논리 측정 값이 의도치 않게 반대로 뒤집어질 수 있다는 점이다 [7]. 이 현상을 논리 큐비트의 측정 부호가 변화한다고 일컫는다 (sign flip). 그림 4의 예시에서 연결된 X boundary에 sign flip이 발생한 경우 이어지는 논리 측정에서 논리 큐비트의 X 측정값이 실제와는 반대로 해석된다.

논리 큐비트의 측정 부호 변화 기록. 따라서, 정확한 논리 측정값의 해석을 위해 LMU는 lattice surgery

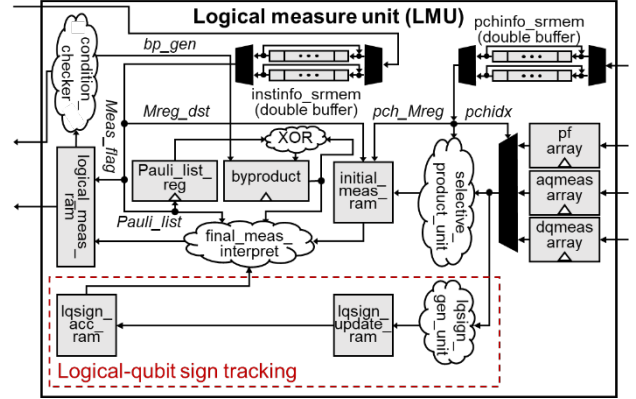


그림 5. 논리 큐비트의 측정 부호 추적 기능을 지원하는 논리 측정 장치의 구조

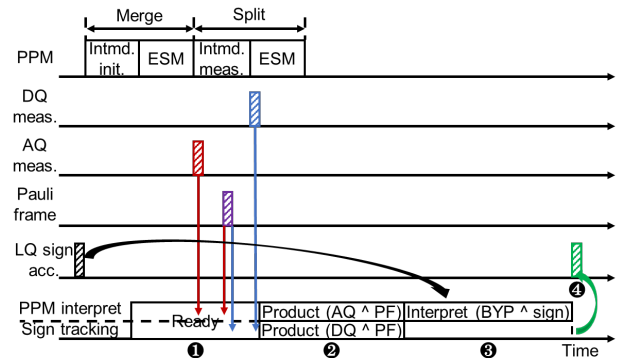


그림 6. 논리 측정 장치의 PPM 해석과 논리 큐비트 부호 추적 수행 과정

과정에서 발생하는 sign flip을 알아내고 기록해야 한다. 이때 sign flip의 발생 여부는 merge 됐을 때 경계를 연결했던 선 상의 모든 데이터 큐비트들에 대해 split에서 얻은 측정값들을 전부 곱하는 것이다. 예를 들면, 그림 4(b)의 예시에서 X_4 를 기준으로 삼았을 때 X_1, X_2, X_3 의 부호는 각각 보라색, 빨간색, 녹색 점선으로 표시된 영역의 데이터 큐비트들의 X 측정값의 곱이 1인 경우 뒤집어진다. 이를 알아내는 과정에서 정확한 데이터 큐비트 측정값이 필요하므로 Pauli frame에 누적된 에러 또한 고려해야 한다.

논리 큐비트의 측정 부호 변화 반영. 또한, LMU는 논리 측정의 최종 해석 단계에서 이전까지 기록했던 논리 큐비트의 측정 부호를 반영해야 한다. 구체적으로, 논리 큐비트가 홀수 번의 sign flip을 겪었다면 음의 부호 (-)로 취급해, 측정값을 뒤집어야 한다.

2.3 논리 큐비트 측정 장치의 구조와 동작 방식

그림 5는 선행 연구 [2]에서 제안한 구조를 바탕으로 sign tracking을 지원하도록 설계한 LMU의 구조를 보여준다. 본 연구에서는 sign tracking이 이뤄지는 PPM 관련 동작을 집중적으로 논의한다.

그림 6은 PPM을 통해 얻은 물리 큐비트 측정값들을 기반으로 LMU가 PPM 해석과 sign tracking을 수행하는 과정을 시간 순서에 따라 요약하고 있다.

준비 단계 (①). LMU는 PPM 측정값 해석과 sign tracking 모두에 대해 merging ESM이 끝난 시점의 Pauli frame을 필요로 한다. 따라서, 두 작업을 병렬적으로 수행하기 위해 필요한 데이터들을 레지스터에 수집하며 모든 데이터가 준비될 때까지 기다린다.

연산 단계 (②). 연산 단계에서는 물리 큐비트 측정값과 Pauli frame들 중 논리 측정값을 도출하거나 논리 큐비트의 sign flip을 감지하기 위해 필요한 부분의 값들만을 선택적으로 곱하여 원하는 값을 얻어낸다. PPM 해석과 sign flip 감지는 전혀 다른 값들을 필요로 하고 동작 방식도 다르기 때문에 별도의 하드웨어 유닛들 (selective_product_unit과 lqsign_gen_unit)을 병렬적으로 사용해 수행한다. 이때 도출된 값들도 각각 별도의 메모리에 저장된다 (init_meas_ram과 lqsign_update_ram).

최종 해석 단계 (③). 최종적인 논리 측정값을 도출하는 단계로, PPM 해석의 경우 이때 현재까지 누적된 $PPR_{\pi/2}$ (byproduct)와 더불어 논리 큐비트의 측정 부호를 고려한다 (final_meas_interpret). 이때, PPM의 측정은 merge 단계에서 일어나고 sign의 변화는 그 이후인 split 단계에서 일어나기 때문에, 이전 단계에서 함께 도출한 sign flip은 제외하고 이전 PPM까지 누적된 sign (lqsign_acc_ram)을 반영한다. PPM 해석은 여기서 마무리된다.

갱신 단계 (④). 이전 단계에서 얻은 sign flip들을 메모리 (lqsign_acc_ram)에 누적한다. 이는 이어지는 LQM과 그 다음 PPM의 해석에 사용된다.

III. 실험 및 결과

본 논문에서는 기존에 자세히 논의되지 않았던 논리 큐비트 부호 추적 기능 (sign tracking)과 이를 지원하는 LMU 구조를 밝혔다. 앞서 설명한 구조의 중요성을 보이기 위해 해당 기능의 유무가 FTQC의 실패율에 미치는 영향을 분석하였다.

3.1 실험 방법

본 실험을 위해 선행 연구 [2]에서 개발된 XQsim 시뮬레이터를 활용했다. 먼저, XQsim의 LMU 모듈을 수정하여 sign tracking 기능을 선택적으로 켜고 끌 수 있게 변형했다. 그 다음, 각 경우에 대해 물리 큐비트의 에러율을 바꿔가며 code distance가 5인 다섯 개의 논리 큐비트들을 대상으로 $PPR_{\pi/2}(Z_1Z_2Z_3Z_4Z_5)$ 를

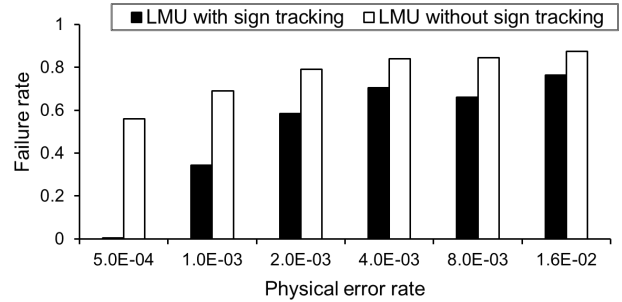


그림 7. 물리 큐비트의 에러율 변화에 따라 LMU의 sign tracking이 $PPR_{\pi/2}$ 의 실패율에 미치는 영향 분석

40회씩 시뮬레이션 했다. 마지막으로, 시뮬레이션 별로 다섯 개의 논리 큐비트를 X, Y, Z 축에 대해 측정했을 때 확률 분포를 도출하고 이를 이론적 정답과 비교했다. 그 결과, 각 물리 큐비트 에러율에 대해 총 200개의 논리 큐비트의 측정 결과를 분석했으며, X, Y, Z 측정 결과 중 하나라도 이론적 정답과 일치하지 않는 논리 큐비트의 비율을 $PPR_{\pi/2}$ 의 실패율 (failure rate)로 정의했다. 이외에 시뮬레이션에 필요한 모든 파라미터들은 선행 연구 [2]에서 사용했던 값을 적용했다.

3.2 실험 결과

그림 7은 LMU가 논리 큐비트 부호 추적 기능을 지원하는 경우 (LMU with sign tracking)과 지원하지 않는 경우 (LMU without sign tracking) 각각에 대해 물리 큐비트 에러율을 변화시키며 $PPR_{\pi/2}$ 의 실패율에 미치는 영향을 분석한 것이다.

먼저, 실험의 결과를 통해 LMU의 sign tracking이 논리 연산의 실패율에 매우 큰 영향을 끼친다는 것을 확인할 수 있다. 구체적으로, 물리 큐비트 에러율이 충분히 낮은 0.0005일 때의 결과를 보면 sign tracking을 지원하는 경우에는 0에 가까운 매우 낮은 실패율을 보이지만 그렇지 못한 경우 절반 이상의 경우 실패하는 것을 확인할 수 있다. 따라서, 해당 결과로부터 본 논문의 주요 목적인 sign tracking을 지원하는 LMU의 중요성을 보일 수 있었다.

한편, 0.001~0.16의 높은 물리 큐비트 에러율에 대한 실험에서는 sign tracking의 중요성이 낮아 보일 수 있다. 그러나, 이 경우는 높은 에러율로 인해 양자 오류 정정이 제대로 이뤄지지 않는 상황으로, FTQC를 수행하는 것이 불가능하다. 이 문제를 해결하기 위해서는 더 높은 정확도를 갖는 양자 오류 탐지 장치 (EDU)를 사용해야 하며, 이는 본 연구의 주제를 벗어나는 별도의 연구 주제이다.

IV. 결론 및 향후 연구 방향

본 논문에서는 정확한 결함 허용 컴퓨팅을 위한 논리 큐비트 측정 장치의 구조와 중요성에 대해 분석했다. 구체적으로, 기존에 알려진 기능들에 더해 제대로 분석된 적이 없는 논리 큐비트 측정 부호의 변화 현상을 논의하고 이를 추적하기 위한 하드웨어 구조를 밝혔다. 그 다음, 해당 기능이 부재할 경우 FTQC의 기본 논리 연산의 수행 과정에서 매우 높은 실패 확률을 보이는 것을 실험을 통해 확인했다. 한편, 실험 과정에서 논리 큐비트 측정 장치와는 별도로, 양자 오류 탐지 장치의 오류 감지 능력이 실패의 주된 원인이 됨을 함께 확인했다. 따라서, 이를 고려해 높은 성능과 낮은 실패율을 동시에 달성하는 양자 오류 탐지 장치의 연구가 필요하다고 여겨진다.

overhead quantum computation using lattice surgery." arXiv preprint arXiv:1808.06709 (2018).

ACKNOWLEDGMENTS

본 연구는 IDEC에서 EDA Tool을 지원 받아 수행하였습니다.

참고문헌

- [1] Terhal, Barbara M. "Quantum supremacy, here we come." *Nature Physics* 14.6 (2018): 530–531.
- [2] Byun, Ilkwon, et al. "XQsim: modeling cross-technology control processors for 10+ K qubit quantum computers." *Proceedings of the 49th Annual International Symposium on Computer Architecture*. 2022.
- [3] Fowler, Austin G., et al. "Surface codes: Towards practical large-scale quantum computation." *Physical Review A* 86.3 (2012): 032324.
- [4] Horsman, Clare, et al. "Surface code quantum computing by lattice surgery." *New Journal of Physics* 14.12 (2012): 123011.
- [5] Litinski, Daniel. "A game of surface codes: Large-scale quantum computing with lattice surgery." *Quantum* 3 (2019): 128.
- [6] Riesebo, Leon, et al. "Pauli frames for quantum computer architectures." *Proceedings of the 54th Annual Design Automation Conference* 2017. 2017.
- [7] Fowler, Austin G., and Craig Gidney. "Low