Foundations and Methodology for a Finite-QEC Substrate: Code, Crystallisation, Ledgers, and Audit Protocol

David Elliman
Neuro-symbolic Ltd
`dave@neusym.ai`

Revised canon snapshot: 2026-06-25

Abstract

This companion paper states the foundational objects and audit methodology used by the finite-QEC (quantum-error-correction) substrate program summarized in the overview paper. The physics is not presented as a continuum field theory with a hidden discretization. It is presented as a finite record-writing system whose ordered crystalline phase supplies the visible substrate and whose correction, strain, service, and boundary ledgers determine which quantities become physical observables. Since the first version of this paper, the foundation has sharpened: the eight-bit cell is no longer merely an assumed starting object. Under explicit stable-record hypotheses — binary local records, commuting read/write CSS structure, balanced self-duality, closed-record-pair evenness, and distance-4 erasure protection — the unique minimal cell is the self-dual doubly-even $[8, 4, 4]$ code. The record-reconstruction layer also splits the old "why quantum?" residual into a philosophical-operational floor, a mostly theorem-grade complex-QEC/Born structure, and a quantitative service-rate result $\alpha_0 = 1/137$ derived from the symmetric record-pair alphabet. What remains open is therefore narrower: the naturalness of the record/locality axioms, recovery holonomy and CP orientation, the dressed-α precision residual, and sector-specific billing maps where they are still needed. The paper introduces the local code, the right bi-cubic crystallisation picture, the distinction between syndrome and strain readouts, the $28 = 2 \times 14$ service-clock construction, and the methodological rules that classify claims as locked, computed, foundationally grounded, conditional, retired, or open. The goal is to make the later physics papers auditable: every coefficient must name its carrier, event unit, scheduler, observable map, and reproducibility check. The methodology's own kill condition is concrete: a companion-paper coefficient that cannot exhibit this ledger is, by the standard set here, not canon.

Contents

1 Role of This Paper

The overview paper [9] gives the shortest map of the current canon. This paper gives the rules of construction. It is not primarily a phenomenology paper. Its task is to define the objects that later papers use. The accompanying code repository is part of the evidential record rather than an optional supplement [11].

- the record-reconstruction ladder and its remaining irreducible premises;
- local finite registers and their code constraints;
- the crystallised cubic phase and the meaning of vertices, branches, faces, cells, and defects;
- the ledgers that count strain, syndrome, service, boundary printing, and wall-shadow records;
- the $28 = 2 \times 14$ service-clock structure;
- the protocol by which claims enter, leave, or remain conditional in the canon; and
- the reproducibility standard for scripts and finite enumerations.

The central methodological rule is simple: a numerical or structural claim is not canon because it sounds natural. It is canon only after its load-bearing map is stated. In this program, the map usually has the form

$$\text{finite substrate} \longrightarrow \text{event ledger} \longrightarrow \text{observable}.$$

Most historical overclaims failed in the middle arrow.

2 Record Reconstruction and the Forced Byte

The deepest premise of the framework is not a particular particle or cosmological number. It is that nature contains stable local records: facts that can be read repeatably, copied redundantly, reset irreversibly, and tested compatibly by local observers. This is the philosophical-operational floor of the programme. It is not something the framework can derive without circularity; every physical theory has such a floor. The useful question is what follows once this floor is accepted.

The current canon answers that question in three layers.

Layer	Status	Content
A. Operational floor	irreducible premise	Stable local records, compatible tests, repeatable readout, reusable reset, finite noise, and local composition.
B. Mathematical reconstruction	foundationally grounded	Repeatability gives orthogonal/projective records; Stinespring/Naimark dilation supplies record isometries; local tomography plus reversible dynamics selects complex Hilbert composition over real or quaternionic alternatives [4, 15]; commuting stabilizer records give a non-contextual compatible-test structure; closed record pairs give the Born square inside that setting.
C. Quantitative service rate	foundationally grounded for the bare rate	The monitored service alphabet is a one-hot projective record observable with the symmetric record-pair count $\mathrm{Sym}^2(16) = 136$, plus idle/latch, so the bare firing probability is $$p_{\text{fire}} = \frac{1}{136+1} = \frac{1}{137} \equiv \alpha_0.$$ The residual is no longer the seed count itself, but dressed-α renormalisation and sector-specific proof that a given process bills the relevant service observable.

The most concrete consequence is the minimal balanced record-cell theorem. If the local record system is binary, has repeatable commuting read/write CSS stabilizers, is balanced ($\dim C = n/2$), is closed under the record-pair evenness condition (doubly even), and protects against distance-4 erasures, then the first possible cell is forced:

$$n = 8, \qquad C \simeq [8, 4, 4],$$

with weight enumerator $1 + 14y^4 + y^8$. Exhaustive enumeration finds no $n = 2, 4, 6$ Type-II self-dual distance-4 cells and one coordinate-permutation class at $n = 8$. Thus the slogan “start with a byte” is no longer an extra assumption. Under the stated balanced-record hypotheses, the byte is the minimal cell.

This does not make the whole framework unconditional. The remaining load is now sharply named: why the operational floor is the right floor, how recovery holonomy and CP orientation enter the record calculus, and which observable a given sector actually bills. The latest dressed-α

escape programme is an example of the distinction. The direct Ward/Kubo/Peierls audits still block the strongest claim: the monitored service occupation is a diagonal pointer observable, while the measured low-energy charge is a retarded electromagnetic response. A literal LSZ/Ward kernel therefore reads the charge-weighted $\sum Q^2$ slot, not a charge-blind service count, and the historical $N_1 = 31$ Dyson–Schwinger coefficient remains a mode-count near-hit if it is presented as a self-energy kernel.

The current route is more precise but still conditional. Closed record-pair counting can select a finite local Maxwell F^2 contact or normal-ordering constant at the service cutoff, with

$$N_{\text{contact}} = 2\sum Q^2 - 1 = 31,$$

which gives $\alpha^{-1} = 137.036013$ at the observed scale. This is the right magnitude and the right subtraction pattern, but the missing theorem is the Euclidean/in–out normal-ordering map that turns endpoint covariance (a Keldysh/noise-block object) into the finite retarded/local Maxwell contact. Thus the bare service rate is derived, while the dressed electromagnetic value is a bounded conditional Maxwell-contact candidate, not a foundation axiom and not yet a Locked QED prediction. These are different questions from the older, vaguer residual "why quantum?".

3 Local Register and Code Structure

The local substrate is therefore described by an eight-bit register. The exact labels are paper-dependent, but the common structure is a finite binary space

$$V = \mathbb{F}_2^8$$

with physical states selected by code constraints. The code-theoretic language is that of the extended Hamming / first-order Reed–Muller $[8, 4, 4]$ structure [14, 19] and its CSS relatives [3, 22]. For this foundations paper, the important point is not a single notation convention, but the role the code plays:

1. it makes the local state space finite;
2. it separates valid register states from invalid or strained states;
3. it supplies finite syndrome and strain readouts;
4. it gives a natural set of affine-hyperplane service channels; and
5. it makes many claims finite enumerations rather than continuum ansatzes.

The code has two lives in the canon. First, it is a local matter-code structure: finite states, charges, chirality, colour, generation-like organization, and anomaly arithmetic. Second, it is an error-correction instrument in the standard quantum-error-correction sense [13, 17, 21]: it determines what can be detected, corrected, billed, erased, or left as residual strain.

3.1 Syndrome, Strain, and Record Content

It is tempting to say that the engine decodes "the code syndrome." The current canon is sharper: the physical record channel is the strain ledger. The distinction matters.

Let a local state be $x \in V$. A syndrome readout asks which parity constraints are violated. A strain readout records a richer geometric frustration pattern, for example the edge or cut content associated with the cell. Schematically,

$$x \longmapsto s(x) \quad \text{syndrome}, \qquad x \longmapsto \mathcal{F}(x) \quad \text{strain}.$$

The syndrome is the minimal logical error-correction message of stabilizer theory [13]. The strain ledger is the physical frustration record that later gravitational, Bekenstein, and black-hole arguments read. A claim that bills the syndrome when the framework's own mechanics reads strain is using the wrong carrier.

This is why later thermodynamic claims are required to state the readout explicitly. The difference between a four-bit code syndrome and a twelve-edge strain ledger is not cosmetic; it changes event counts, blind slots, and service scheduling.

4 Crystallisation of the Register-Bearing Substrate

The visible substrate is not assumed to be a smooth manifold. It is the ordered phase of a constrained register network. The useful mental model is right bi-cubic crystallisation — bi-cubic because two cubic structures are in play: the ambient lattice is periodic $\mathbb{Z}^3$, and the ordered phase tiles it with disjoint eight-vertex register cells that are themselves combinatorial cubes. Local finite register cells lock into this embedded cubic phase, whose vertices, branches, and faces have different physical jobs.

4.1 Vertices

Vertices are the local register-bearing sites. A crystallised vertex is instrumented: it has code content, can carry strain records, can participate in QEC service, and can be read by the boundary or bulk ledgers. A vertex outside the instrumented crystal may still carry geometric energy or disorder, but it does not automatically carry a register record.

This distinction is central to the dark-sector interpretation [6]. Unrecorded wall energy is not simply charged to gravity. Gravity reads the recorded boundary strain or its shadow, not an uninstrumented internal energy as a free particle gas.

4.2 Branches

Branches or bonds specify adjacency. They carry the finite move graph, line support, and local transport possibilities. In the embedded crystallisation simulations the physical state is a bond subset of a periodic $\mathbb{Z}^3$ lattice, with each site constrained to hold exactly three of its six possible bonds. A local state can therefore be discussed as

$$B \subset E(\mathbb{Z}^3/L\mathbb{Z}^3), \qquad \deg_B(v) = 3.$$

This embedded ensemble is essential. A looser configuration-model ensemble allowed non-physical ordered phases such as $K_{3,3}$ clusters. The embedded cubic lattice forbids these patterns: the local neighbour geometry does not permit the shared-neighbour structure required by $K_{3,3}$, and triangular K_4-type artifacts are likewise absent in the bipartite substrate. The physical crystallisation problem is therefore not the old toy problem; it is the embedded degree-constrained problem.

4.3 Faces, Plaquettes, and Cells

Faces and plaquettes are where loop constraints live. A cube is not merely a collection of vertices and branches; it has plaquette structure. Plaquette moves preserve degree while changing local bonding. These moves define the computationally accessible dynamics used in the K04 crystallisation audits (K04 is the canon's fourth foundational kernel: the topological crystallisation rule favouring 4- and 6-cycles while penalising degree defects); plaquette structure plays the same organising role here that plaquette operators play in surface codes [16].

The ordered crystal maximizes the appropriate square and hexagon cycle content. In lattice units, the perfect cube tiling has the characteristic cycle counts used by the simulations. Defects are then not arbitrary missing sites; they are deviations in the cycle and bond ledger.

4.4 Sectors and Locality Protection

Plaquette moves are local and degree-preserving. They do not necessarily connect the full degree-three state space. The move set has conserved cut-parity sectors. This non-ergodicity is not merely a nuisance: it is part of the hidden-sector structure. The physical protocol compares hot and cold branches inside the sector containing the crystal anchors, so the comparison is meaningful even though the global state space splits into many sectors.

Durability is measured by the healing spectrum. Given a defect state and a nearest crystal anchor, the difference decomposes into alternating occupied and unoccupied closed trails. Plaquette moves implement the length-four trails. The interpretation is:

Healing spectrum	Interpretation
Only length-four trails	plaquette-healable artifact
Short six/eight trails	move-class artifact candidate; enlarge the move set and recheck
Extensive trails	locality-protected wall or domain structure

This rule prevents the common mistake of calling every trapped defect a dark matter candidate. A surviving defect must have positive excess energy and a durability spectrum dominated by extensive trails before it can even enter the dark-sector discussion.

5 Boundary Printing and Service Ledgers

The framework has several ledgers. Keeping them separate is the main technical discipline.

Ledger	Role
Code content	finite state labels and physical-code constraints
Syndrome	logical parity-violation readout
Strain	geometric frustration record; carrier for several gravitational and horizon arguments
Service current	count of QEC repair, reset, or boundary-service events
Boundary printing	creation of new instrumented cells at the cosmological boundary
Wall shadow	truncated readout of strain adjacent to uninstrumented domain-wall material

A physical claim must specify which ledger supplies the observable. For example, the scalar spectral tilt uses the 28-channel service clock only after a holographic-boundary-crystallisation (HBC) lift to a radial log-scale power ledger [5]. The debris sector uses wall-shadow strain rather than bare wall energy [6]. The black-hole sector reads horizon strain records rather than an unconstrained entropy count [7].

6 The $28 = 2 \times 14$ Service Clock

The code supplies a finite service structure. The fourteen weight-four logical service channels can be identified with affine hyperplanes of $\mathbb{F}_2^3$. The affine group $\mathrm{AGL}(3, 2)$ acts transitively on these hyperplanes. Adding the two transverse modes gives a single 28-channel service orbit:

$$N_{\text{service}} = 2 \times 14 = 28.$$

The microscopic incidence refinement is

$$8 \text{ point labels} \longrightarrow 112 \text{ incidence flags} \longrightarrow 28 \text{ coarse service channels.}$$

The important feature is not the raw number of flags, but the coarse-grained service clock. Under a one-jump finite-bandwidth scheduler, covariance gives uniform service weights:

$$p_x = \frac{1}{28}.$$

The serial absorbing service clock then has first gap

$$\Delta_1 = \frac{1}{28}.$$

This finite result is strong, but the methodology requires one more question for every application: which physical ledger does this clock act on? Acting on an amplitude, a power spectrum, a boundary shell, or a late-time activation fraction gives different physics. The code clock is finite; the observable map is the conditional part.

7 Thermodynamic-Claim Protocol

The thermodynamic and dark-sector parts of the project had the largest historical retraction cluster. The root problem was not the use of thermodynamic language itself — the Landauer accounting of irreversible information processing is standard [1, 18] — it was coefficient claims made before the ledger map was explicit. The current protocol requires the following gates.

Gate	Required statement
T1	carrier: what object is being counted or billed?
T2	event unit: what is one event, erasure, service, fault, or record?
T3	scheduler: serial, parallel, active-address, round-robin, or stochastic?
T4	observable map: how does the ledger become $w(a)$, A_s, G, mass, pressure, or another physical quantity?
T5	covariance/correlation volume: which events are independent, common-mode, or projected out?
T6	no double count: what carrier is replaced by the record, and what remains inactive?
T7	competing branches: what alternatives would give a different number?
T8	reproducibility: what script, finite enumeration, or proof checks the claim?

The HBC amplitude target is a useful example. The finite 28-clock gives a relative clock and tilt candidate. It does not automatically give the scalar amplitude. The current canon conditionally closes the amplitude only after the separate local saturated-printer premises are stated: the shell event unit is a weight-4 topology commit, the stop rule gives $N_{\text{shell}}\alpha_0^4 = 4/3$, and the local product current gives $S_j(k = aH) = 1$. A nonlocal horizon-mode scalar source, or a different observable carrier, would reopen the amplitude while leaving parts of the 28-clock slope intact. This is exactly why the protocol requires the carrier and covariance map, not only the attractive final number.

8 Canon, DRIFT, and PTMS

The canon is maintained as a structured document rather than as a collection of disconnected papers. `ANCHOR.md` records current claims and their status. `DRIFT.md` records retractions, demotions, supersessions, and scope changes. PTMS extracts a machine-readable registry from these documents and runs consistency checks.

The important methodological point is that retractions are not hidden. They are first-class data. A later paper should not silently rewrite an old claim as if it had always been clean. It should cite the retired form, state why it failed, and then state the surviving form.

8.1 Claim States

State	Use
Locked	finite theorem or exact enumeration with no named load-bearing gap
Computed	reproducible script result; may still require physical interpretation
Conditional	all algebra after named premises is clear, but at least one premise remains open
Foundationally grounded	standard theorem or canon-derived structure resting only on the record-reconstruction floor
Retired	historical claim withdrawn or superseded
Open frontier	active target with a specified closure or kill condition

This vocabulary is deliberately plainer than journal rhetoric. It is meant to make overstatement hard. The intent is the classical falsificationist one — a claim's scientific content is carried by what could kill it [20] — enforced in the bend-over-backwards reporting style Feynman asked for [12].

8.2 PTMS Checks

The PTMS checks do not prove the physics. They enforce structural integrity: duplicate item numbers, live citations to retired claims, missing scripts, registry drift, and soft tier-coverage debt. A clean PTMS run is not a truth certificate; it is a statement that the corpus is internally auditable.

9 Reproducibility Standard

Every technical paper in the series should include a script ledger, in the spirit of reproducible computational research [2]. The minimal entry is:

Field	Example	Meaning
Claim	28-channel clock	result being supported
Status	COMPUTED	epistemic status
Script	`python_code/item131_w_to_28_instrument.py`	exact script path
Invocation	`~/bin/py13_7/bin/python...`	reproducible command
Expected output	$\Delta_1 = 1/28$ branch	what must be seen
Commit	`ee0bb08`	canon/code snapshot

Numerical scans must also state whether they are toy models, physical embedded ensembles, finite-size probes, or proof-enumerations. The old K04 configuration-model debris plateau is a textbook reason for this rule: it was useful as a negative artifact, not as the physical measurement.

10 What This Foundation Does Not Claim

This paper does not claim that all continuum physics has been derived. It does not claim that every numerical match in the corpus is meaningful. It does not claim that a clean PTMS run makes a speculative theory true. It does not claim that TeV photons are Bloch modes of the $\Lambda_{\rm QCD}$ crystal. The current relativity canon represents trans-$\Lambda_{\rm QCD}$ quanta as framed causal-set/null-chain external legs with normalized endpoint residue, so the old "support problem" is closed only at that interface grade [10]. What remains is precision QED and astrophysical-transfer phenomenology around that external leg, not a hidden fine oscillator lattice.

It claims something narrower: the framework now has a record-reconstruction ladder, forced minimal byte, finite substrate, code, ledger vocabulary, crystallisation ensemble, service-clock structure, and audit protocol precise enough that later physics claims can be tested against them. That is the necessary foundation for the companion papers [5–10].

A Starter Script Ledger

Purpose	Script or command	Expected result
PTMS extraction and check	`PYTHONPATH=ai_methodology~/bin/py13_7/bin/python-mptmscheck`	Hard findings zero for clean snapshots.
Section-15 numbering guard	`python_code/verify_s15_numbering.py`	No duplicated or missing canonical 100s headers.
Methodology count audit	`ai_methodology/methodology_metrics.py`	Authoritative DRIFT/audit counts parse internally; paper-prose issues named.
Record reconstruction tier split	`python_code/record_reconstruction_tier_split.py`	Separates operational floor, structure-only reconstruction, and quantitative service-rate layers.
Minimal balanced record cell	`python_code/minimal_balanced_record_cell_theorem.py`	Exhaustive $n < 8$ exclusion and unique $[8, 4, 4]$ Type-II self-dual distance-4 cell at $n = 8$.
Bare α_0 service rate	`python_code/alpha0_record_pair_symmetry_theorem.py`	Derives the symmetric record-pair count $\mathrm{Sym}^2(16) + 1 = 137$.
Sector billing-map audit	`python_code/alpha0_downstream_billing_map_audit.py`	Checks which downstream quantities bill the R14 monitored service observable.
Dressed-α continuum escape	`python_code/dressed_alpha_monitor_web_continuum_dos.py`	Pins the non-unital bridge–Wilson-web service-occupation value at 0.9956 of the observed shift at the native tick.
Dressed-α readout and contact audits	`python_code/dressed_alpha_rate_readout_theorem_audit.py`; `python_code/dressed_alpha_sector_billing_no_go.py`; `python_code/dressed_alpha_ward_kubo_peierls_observable_audit.py`; `python_code/dressed_alpha_service_kubo_moment_no_go.py`; `python_code/dressed_alpha_field_content_31.py`; `python_code/dressed_alpha_maxwell_contact_selector_theorem.py`; `python_code/dressed_alpha_endpoint_contact_map_attempt.py`	Shows the internal service readout is sharply selected, while the physical Ward/Kubo/Peierls self-energy is a different billing slot with the one-loop-sized undershoot. The current positive candidate is a finite Maxwell-contact selector $2\sum Q^2 - 1 = 31$; it remains conditional on an endpoint-covariance $\rightarrow F^2$ normal-ordering map.
28-channel incidence bridge	`python_code/item131_w_to_28_instrument.py`	Finite $8 \rightarrow 112 \rightarrow 28$ service-channel construction.
28-channel covariance/equipartition	`python_code/equipartition_channels_120_131.py`	$\mathrm{AGL}(3, 2)$-transitivity and uniform 1/28 channel weights.
Strain-decoder cell law	`python_code/d_to_p_map.py`	Exact cell-failure law under strain readout.
Embedded crystallisation worker	`python_code/k04_embedded_sweep.py`	Physical embedded ensemble; no configuration-model $K_{3,3}$ artifact.
Orphan-policy audit	`python_code/k04_orphan_policy_audit.py`	Rescue/orphan-policy status and registered larger-run surface.
Thermodynamic target protocol example	`python_code/item131_hbc_stop_rule_proof.py`	Conditional local-printer candidate $N_{\mathrm{shell}}\alpha_0^4 = 4/3$ and $A_s = (3/4)\alpha_0^4$, with channel-lock/spatial-whitening still named.
Trans-Λ_{QCD} support closure	`python_code/foundations_trans_lambda_precision_residuals.py`	Moves the high-energy bridge from support/LSZ to precision-QED and Lorentz-transfer tests.

References

[1] C. H. Bennett. The thermodynamics of computation—a review. *Int. J. Theor. Phys.*, 21: 905–940, 1982.

[2] J. B. Buckheit and D. L. Donoho. Wavelab and reproducible research. In A. Antoniadis and G. Oppenheim, editors, *Wavelets and Statistics*, volume 103 of *Lecture Notes in Statistics*, pages 55–81. Springer, 1995.

[3] A. R. Calderbank and P. W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54:1098–1105, 1996.

[4] Giulio Chiribella, Giacomo Mauro D'Ariano, and Paolo Perinotti. Informational derivation of quantum theory. *Physical Review A*, 84:012311, 2011. doi: 10.1103/PhysRevA.84.012311.

[5] D. G. Elliman. Cosmology, dark energy, and inflation in the finite-QEC substrate. Zenodo preprint, `https://doi.org/10.5281/zenodo.20671860`, 2026.

[6] D. G. Elliman. Dark matter, MOND, and K04 debris in the finite-QEC substrate. Zenodo preprint, `https://doi.org/10.5281/zenodo.20671862`, 2026.

[7] D. G. Elliman. Gravity, horizons, and black holes in the finite-QEC substrate. Zenodo preprint, `https://doi.org/10.5281/zenodo.20671864`, 2026.

[8] D. G. Elliman. Matter, gauge structure, and spectroscopy in the finite-QEC substrate. Zenodo preprint, `https://doi.org/10.5281/zenodo.20671858`, 2026.

[9] D. G. Elliman. A finite-QEC substrate program for particle physics and cosmology: current canon and audit methodology. Zenodo preprint v5.0, `https://doi.org/10.5281/zenodo.20682177`, 2026.

[10] D. G. Elliman. Special and general relativity from the finite-QEC substrate: the propagation clock, the equivalence principle, and the horizon ledger. Zenodo preprint v3.0, `https://doi.org/10.5281/zenodo.20682176`, 2026.

[11] David Elliman. It-for-bit model: Reproducibility repository. `https://github.com/dgedge/itforbit_model.git`, 2026. Code and reproducibility repository for the canon snapshot series.

[12] R. P. Feynman. Cargo cult science. *Engineering and Science*, 37(7):10–13, 1974.

[13] D. Gottesman. *Stabilizer codes and quantum error correction*. PhD thesis, California Institute of Technology, 1997. arXiv:quant-ph/9705052.

[14] R. W. Hamming. Error detecting and error correcting codes. *Bell Syst. Tech. J.*, 29:147–160, 1950.

[15] Lucien Hardy. Quantum theory from five reasonable axioms. *arXiv preprint quant-ph/0101012*, 2001.

[16] A. Yu. Kitaev. Fault-tolerant quantum computation by anyons. *Ann. Phys.*, 303:2–30, 2003.

[17] E. Knill and R. Laflamme. Theory of quantum error-correcting codes. *Phys. Rev. A*, 55:900–911, 1997.

[18] R. Landauer. Irreversibility and heat generation in the computing process. *IBM J. Res. Dev.*, 5:183–191, 1961.

[19] F. Jessie MacWilliams and Neil J. A. Sloane. *The Theory of Error-Correcting Codes*. North-Holland, 1977.

[20] K. R. Popper. *The Logic of Scientific Discovery*. Hutchinson, London, 1959.

[21] P. W. Shor. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A*, 52:R2493–R2496, 1995.

[22] A. M. Steane. Error correcting codes in quantum theory. *Phys. Rev. Lett.*, 77:793–797, 1996.