

Article

Integrating Sustainability into Embedded Systems Education: A CDIO-Based Framework

Xiangjin Zeng ^{1,2} 

¹ School of Computer Science and Engineering, Wuhan Institute of Technology, Wuhan 430205, China; xjzeng21@163.com

² Hubei Key Laboratory of Intelligent Robot, Wuhan Institute of Technology, Wuhan 430205, China

Abstract

While existing curricula often focus on theoretical aspects of sustainability, they frequently fail to equip students with practical design skills required by the green industry. To address this disconnect, this study seeks to answer: How can a structured pedagogical framework effectively enhance students' ability to translate abstract sustainability principles into concrete technical solutions? This study introduces a comprehensive CDIO-based framework reform for Embedded Intelligent Systems education, weaving sustainability throughout every phase. We put forward a "Sustainable CDIO Capability Model" that charts a progressive pathway—starting from basic resource awareness and advancing through to sophisticated sustainable system innovation. Our four-dimensional teaching strategy brings this model to life: first, project-based learning driven by real sustainability challenges; second, a hybrid ecosystem blending online resources, hands-on practice, and immersion in green industry contexts; third, hierarchical team-based pedagogy backed by personalized support mechanisms; and fourth, a multi-dimensional assessment system that weights energy efficiency, resource stewardship, and social value creation alongside conventional metrics. We implemented this approach with Intelligent Science and Technology majors at Wuhan Institute of Technology. The results show the model effectively bridges the persistent gap between dry technical content and the practical demands of green industry. Students made substantial gains not merely in core engineering capabilities—system architecture, hardware-software co-development—but crucially in sustainable design awareness and their capacity to untangle complex sustainability challenges. This work offers a readily transferable framework for embedding Education for Sustainable Development (ESD) into engineering curricula worldwide. It provides practitioners with a concrete, tested model for cultivating the next generation of engineers who naturally think and act with sustainability in mind.



Academic Editors: Grigorios L. Kyriakopoulos and Jordi Segalàs Coral

Received: 18 April 2026

Revised: 14 June 2026

Accepted: 18 June 2026

Published: 25 June 2026

Copyright: © 2026 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: engineering education; Sustainable Development; CDIO; embedded systems; Curriculum Reform

1. Introduction

The Fourth Industrial Revolution is happening alongside global efforts to meet the Sustainable Development Goals. This dual pressure has fundamentally altered what we expect from engineering schools. It's no longer enough to produce graduates with strong technical competence—universities now face the daunting task of systematically developing engineers who genuinely grasp social responsibility and can drive sustainable innovation [1–6]. At

Wuhan Institute of Technology, our Intelligent Science and Technology and Artificial Intelligence programs aim to train professionals ready to tackle complex, ill-structured societal challenges using intelligent robotics, smart production systems, and embodied intelligence. Yet here's the problem: our current approach to teaching embedded systems largely ignores environmental realities—the energy footprint of devices across their entire lifespan, how wasteful our design processes can be, and what happens to all that electronic waste when we're done with it [7–13]. Embedded systems classes are notoriously broad and intensely hands-on. They're supposed to be foundational for any electronics student. While we've seen some teaching reforms—like building interdisciplinary progressive systems [14] competency-based practical frameworks [15] and constructivist scaffolding approaches [16]—most of these innovations fixate on boosting technical skills and creative thinking. They miss the sustainability piece almost entirely. Conventional courses obsess over getting things to work while paying little attention to whether students can design with the planet in mind, optimize for energy use, or manage product lifecycles responsibly [17–19]. This gap between what we teach and what greening industries actually need is becoming untenable. To address the identified gap in pedagogical frameworks for sustainability education, this study seeks to answer the following research question: How can a structured curricular framework effectively bridge the divide between abstract sustainability theories and the practical design skills demanded by the green industry? While global initiatives like the CDIO framework emphasize integrated learning [20–22], regional implementations in China have often faced challenges. This study contributes to this international dialogue by proposing teaching reform for the Embedded Intelligent Systems course that weaves sustainability throughout the CDIO (Conceive-Design-Implement-Operate) model. We've built a theoretical framework for developing sustainability-oriented engineering capabilities and designed a hierarchical, spiral-progressive teaching methodology. The course follows what we call "Green CDIO". Unlike prior CDIO adaptations that treat sustainability as a supplementary module, Green CDIO restructures all four CDIO stages by embedding ecological literacy, lifecycle thinking, and quantifiable sustainability metrics into every phase of engineering design. It further introduces institutionalized sustainability roles (e.g., Sustainability Officer), transforming sustainability from an optional consideration into a mandatory design constraint. This manuscript is positioned as a mixed-methods empirical case study of curriculum innovation. Its primary aim is to evaluate the impact of the Green CDIO framework on students' technical and sustainability competencies, supported by replicable curricular and assessment details, rather than serving as a purely descriptive teaching reform report.

2. Theoretical Framework: Building Sustainability-Integrated CDIO Capabilities

2.1. Core Idea: Developing Engineers Who Can Handle Sustainability

Applying sustainability-integrated CDIO to embedded systems education means blending technical knowledge, practical skills, and sustainable design thinking into one coherent, step-by-step learning journey. Our approach rests on three theoretical pillars that tackle both how students think and how they eventually practice sustainable engineering.

2.1.1. How Knowledge Actually Transforms When Sustainability Matters

Getting good at embedded intelligent systems isn't just cramming hardware facts (ARM architecture, register setups, interface technologies) together with software concepts (RTOS, algorithms, data structures). Students need to weave discrete pieces of knowledge—GPIO control, task scheduling, communication protocols—into project-based, hardware-software co-design situations where sustainability constraints are real and present. We lean on Cognitive Load Theory here. The idea is to ease the burden on

working memory by grouping complex information into meaningful mental frameworks centered on solving sustainable engineering problems. Take a robot navigation system: students must juggle the hardware limits of STM32 microcontrollers, the software architecture of ROS (Robot Operating System), and sustainability factors like power consumption, thermal management, and material choices that minimize environmental harm. This integrated approach trains dual-track thinking—balancing technical demands with sustainability concerns—helping isolated knowledge evolve into genuine engineering capability.

2.1.2. Building Capabilities Step by Step for Sustainable Innovation

We've mapped out a three-tier progression model that aligns with Bloom's taxonomy but folds in sustainability competency development: Foundation Tier (Sustainable Basics): Students nail core technical skills while getting introduced to fundamental sustainability principles. They learn minimum system construction, peripheral driver development, and basic RTOS task management—all while developing awareness of power efficiency, sleep modes, and energy profiling. We emphasize resource-conscious design from day one. System Tier (Sustainable Co-Design and Optimization): This tests students' ability to integrate hardware and software within sustainability constraints. Projects like an STM32 ROS environmental monitoring robot force them to think across multiple dimensions: driver efficiency, algorithmic optimization for reduced energy use, data synchronization that cuts communication overhead, and system architecture that enables maintenance and upgrades. This tier emphasizes applying sustainable design principles to complex system development. Innovation Tier (Sustainable Integration and Real Impact): Students tackle messy, real-world sustainability challenges through creative system integration. Multi-robot systems for precision agriculture, renewable energy facility inspection, or smart waste management involve wireless protocols, collaborative algorithms, distributed architectures, and comprehensive impact assessment. Their work gets judged not just on traditional performance metrics but on environmental, social, and economic impact—the triple bottom line. Table 1 shows mapping of CDIO stages, bloom's taxonomy, sustainability competencies, and embedded system learning outcomes.

Table 1. Mapping of CDIO Stages, Bloom's Taxonomy, Sustainability Competencies, and Embedded System Learning Outcomes.

CDIO Stage	Bloom Level	Sustainability Competency	Embedded System Learning Outcome
Conceive	Understand	Lifecycle awareness	Define energy and material constraints
Design	Apply	Low-power optimization	Select MCU peripherals for minimal energy use
Implement	Analyze	Resource efficiency	Optimize RTOS scheduling for energy savings
Operate	Evaluate	Environmental impact assessment	Measure, log, and report system energy profile

2.1.3. Learning in Real Sustainable Engineering Contexts

Embedded systems knowledge is deeply contextual and must grow through authentic engineering experiences. The abstract notion of interrupt priority, for instance, plays out differently in bare-metal versus RTOS environments. Real understanding emerges when students wrestle with actual problems in development settings. Our course uses sustainability-driven situated learning: genuine industry cases (multi-robot warehouse systems that optimize energy use), competitions focused on sustainable tech (energy-efficient embedded design challenges),

and school-enterprise collaborations tackling environmental monitoring or resource conservation. This aligns with Situated Learning Theory's core premise—knowledge gets constructed within specific contexts, not through abstract, decontextualized instruction. Embedding sustainability directly into authentic engineering scenarios helps students develop both technical skills and ethical frameworks for responsible innovation.

2.2. Theoretical Backing and Instructional Design for Sustainability Integration

Constructivist principles guide students to actively build their knowledge within real sustainable design constraints. Unlike pure software courses, embedded system design is fundamentally constrained by physical hardware while needing to address system integrity, performance, and sustainability goals. Designing a smart energy monitoring system, for example, requires considering STM32 GPIO allocation, sensor interface timing, task scheduling efficiency, and minimizing the device's environmental footprint through strategic component choices and power management software. In this model, instructors shift from lecturers to project mentors and sustainability coaches. They provide scaffolding (schematic templates, code frameworks, lifecycle assessment tools, energy profiling instruments) while students own the final solutions. Outcome-Based Education (OBE), applied to course evaluation, expands traditional assessment to explicitly measure sustainability competency alongside technical proficiency. Technical outputs now include not just standard engineering processes (chip selection, schematic design, PCB layout, hardware-software debugging) but also justifying sustainable design choices through comparative analysis and impact assessment. Engineering documentation must meet both enterprise standards and emerging environmental compliance frameworks. Innovation shows up through participation in sustainability competitions (energy conservation and emission reduction contests), green technology patents, or research publications addressing environmental challenges in embedded systems. This end-to-end design philosophy creates a virtuous cycle connecting student knowledge and skills, curriculum objectives, industrial sustainability needs, and responsible engineering capability development. Figure 1 shows the theoretical framework: building sustainability-integrated CDIO capabilities.

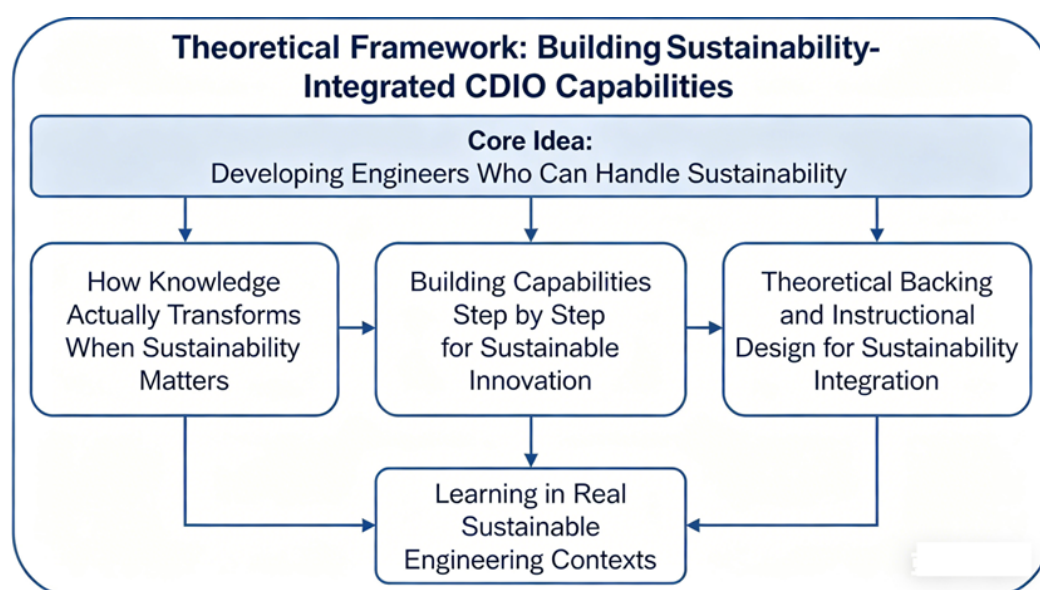


Figure 1. The theoretical framework: building sustainability-integrated CDIO capabilities.

3. Teaching Methodology: Four Strategies for Sustainable Embedded Systems Education

3.1. Project-Based Learning with Sustainability and Hardware-Software Co-Design

Project-based learning (PBL) formed the core instructional strategy, with all projects explicitly framed around sustainability challenges. At the foundational level, the curriculum integrated hardware understanding with software concepts, emphasizing low-power design and efficient resource utilization. System-level projects, such as smart agricultural monitoring robots, required students to design STM32-ROS communication protocols and timing synchronization strategies that balanced reliability with energy minimization. Debugging processes emphasized systematic problem-solving within constraints that included functional requirements, energy budgets, and thermal limits. This pedagogical approach aimed to cultivate engineers who treat sustainability as integral to technical excellence. Figure 2 shows the teaching methodology: four strategies for sustainable embedded systems education capabilities.

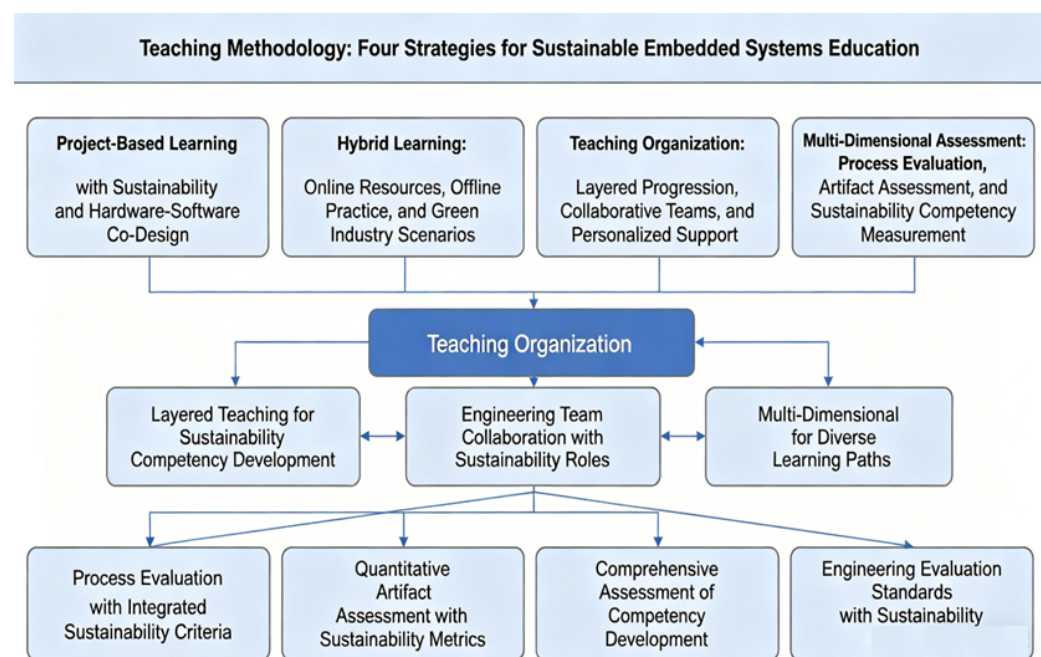


Figure 2. The teaching methodology: four strategies for sustainable embedded systems education capabilities.

3.2. Hybrid Learning: Online Resources, Offline Practice, and Green Industry Scenarios

A hybrid learning ecosystem was implemented to enhance accessibility and flexibility. Online components included a dedicated library of micro-lectures (6–8 min each) covering technical fundamentals (e.g., register configuration, interrupt nesting) and sustainability topics (e.g., low-power modes, thermal management). Simulation tools (e.g., Proteus) allowed students to verify circuits and conduct preliminary energy analyses prior to physical prototyping, which helped reduce material waste associated with iterative design cycles. Electronic textbooks with QR codes linked to sustainable design guidelines and component lifecycle databases. Offline sessions maintained a hands-on focus. Classroom time centered on project-guided activities where teams developed sustainable solutions. Industry engagement was integrated through authentic enterprise projects (e.g., multi-robot systems for waste sorting, IoT solutions for precision irrigation) and partnerships with green technology companies. Students were encouraged to participate in competitions focused on sustainability themes, such as the “Internet Green Innovation” contest.

3.3. Teaching Organization: Layered Progression, Collaborative Teams, and Personalized Support

3.3.1. Layered Teaching for Sustainability Competency Development

Layered implementation addresses diverse learning needs while systematically building sustainability competencies. The foundation tier ensures everyone masters sustainable basics: low-power design principles, efficient coding, basic environmental impact assessment. The system tier offers differentiated project pathways with varying complexity—from basic robot obstacle avoidance to advanced SLAM implementations for environmental monitoring—letting students develop appropriate skills while progressively tackling more sophisticated sustainability challenges. The innovation tier provides advanced exploration through sustainability research projects, cutting-edge technology investigations, or high-level green technology competitions, supporting future leaders in sustainable engineering innovation.

3.3.2. Engineering Team Collaboration with Sustainability Roles

Team-based projects develop both technical collaboration and sustainability leadership. Embedded projects typically need multidisciplinary teams covering hardware design, software development, testing, and verification. In our course, teams tackle substantial projects like smart energy monitoring systems or autonomous environmental sensing platforms. Teams are explicitly structured with sustainability-focused roles, including a designated “Sustainability Officer” overseeing environmental impact throughout design. Members assume specialized responsibilities (hardware engineer focusing on energy-efficient component selection and longevity, software engineer optimizing algorithms for computational efficiency, testing engineer developing validation protocols including power metrics) while collaborating through clear interface specifications. Students use professional tools like Git for version control and collaborative documentation platforms, developing technical collaboration skills and awareness of sustainability-conscious engineering norms. Regular team meetings, progress reviews, and problem-solving sessions enhance communication, coordination, and collective responsibility for technical and sustainability objectives.

3.3.3. Personalized Support for Diverse Learning Paths

We’ve established comprehensive personalized support addressing varied backgrounds, interests, and career aspirations. While everyone must master core sustainable engineering fundamentals, the system provides differentiated skill development pathways. Students needing extra support get targeted assistance through one-on-one consultations, and extended lab access. Those showing advanced capabilities face optional complex projects, opportunities to contribute to faculty sustainable technology research, or participation in high-level innovation competitions. The support system is flexible and responsive, with regular progress assessment informing adjustments to teaching strategies, resource allocation, and challenge levels. This ensures all students can develop their potential as sustainability-conscious engineers, regardless of starting point.

3.4. Multi-Dimensional Assessment: Process Evaluation, Artifact Assessment, and Sustainability Competency Measurement

3.4.1. Process Evaluation with Integrated Sustainability Criteria

Our assessment framework systematically evaluates both technical proficiency and sustainability competency throughout learning. Evaluation covers traditional engineering quality dimensions (schematic correctness, code quality, documentation completeness) while explicitly incorporating sustainability criteria: energy efficiency analysis, environmental impact consideration, and social value creation. We use progressive evaluation rubrics

(Tables 1–3) aligned with the layered learning structure, providing clear expectations and feedback at each development stage.

To ensure rigor and reproducibility, the measurement procedures for all key performance indicators were strictly standardized. Completion Time was defined as the total calendar days elapsed between the official start and final delivery dates, with data automatically logged by the Project Management System (PMS) to eliminate recall bias. Code Reuse was quantified objectively using static code analysis to calculate the percentage of functional code invoked from pre-existing verified libraries relative to the total lines of code. Sustainability Score, addressing the abstract nature of software longevity, was evaluated by a panel of three independent senior architects using a predefined rubric assessing Maintainability, Scalability, and Documentation Quality on a 0–100 scale. As reported in Table 9 the inter-rater reliability for this subjective assessment was excellent ($ICC > 0.85$), confirming the robustness of the evaluation process.

Table 2. Basic Stage Evaluation Framework with Sustainability Integration.

Evaluation Dimension	Specific Criteria	Score	Performance Indicators
Task Completion and Quality	1. Hardware/software functions per specs 2. Reports follow standard format with accurate data 3. Basic power consumption in all testing	30 pts	Functional completeness (0–10) Report quality (0–10) Energy measurement (0–10)
Knowledge Application	1. Explain code logic with efficiency focus 2. Proficient use of KEIL, Proteus, power analyzers 3. Apply basic low-power patterns	25 pts	Code interpretation (0–8) Tool proficiency (0–9) Power pattern use (0–8)
Collaboration and Teamwork	1. Timely completion of responsibilities 2. Active participation in discussions/decisions 3. Shared responsibility for sustainable practices	20 pts	Task reliability (0–7) Discussion contribution (0–7) Sustainability collaboration (0–6)
Basic Problem Solving	1. Independent debugging of integration issues 2. Code optimization based on feedback 3. Systematic power issue diagnosis	25 pts	Debugging independence (0–9) Optimization implementation (0–8) Power issue resolution (0–8)

3.4.2. Quantitative Artifact Assessment with Sustainability Metrics

We’ve developed comprehensive quantitative evaluation frameworks for projects at each level, incorporating traditional performance metrics and specific sustainability indicators. These provide objective measurement of student achievement while emphasizing sustainable design outcomes. Figure 3 Shows the teaching activities in embedded system courses.

3.5. Research Methodology and Data Analysis

3.5.1. Research Design

This study employed a mixed-methods case study design to investigate the implementation and impact of the pedagogical framework described above. Quantitative data were collected through pre- and post-intervention assessments of project artifacts, while qualitative data were gathered via student reflections, interviews, and analysis of competition outcomes.

Table 3. Advanced Stage Evaluation Framework.

Evaluation Dimension	Specific Criteria	Score	Performance Indicators
Project Logical Design	1. Requirements docs including sustainability specs 2. Modular, hierarchical architectures supporting maintainability 3. Lifecycle thinking in design decisions	25 pts	Requirements completeness (0–9) Architecture quality (0–8) Lifecycle integration (0–8)
Technical Depth and Innovation	1. Advanced techniques (multithreading, interrupts) for efficiency 2. Optimized communication protocols (UART, I2C, SPI) 3. Energy-efficient algorithms and system management	30 pts	Advanced technique use (0–10) Protocol implementation (0–10) Algorithm efficiency (0–10)
Process Management	1. Milestones achieved per schedule 2. Tracking performance and sustainability metrics 3. Iterative refinement based on testing and impact assessment	20 pts	Milestone achievement (0–7) Metric tracking (0–7) Iterative improvement (0–6)
Sustainable Design Integration	1. Document design choices considering energy, material, end-of-life 2. Implement features supporting energy efficiency and environmental responsibility 3. Comparative analysis of alternatives from sustainability perspective	25 pts	Design documentation (0–9) Feature implementation (0–8) Comparative analysis (0–8)

**(a)** The diagram illustrates the teaching process.**(b)** It shows the process of students answering**(c)** The diagram shows the experimental process of an embedded system.**Figure 3.** Teaching activities in embedded system courses.

3.5.2. Participants

The study involved 30 undergraduate students majoring in Intelligent Science and Technology at Wuhan Institute of Technology during the Semester 2/2024. Participants had diverse academic backgrounds and prior exposure to embedded systems.

3.5.3. Data Collection Instruments and Procedure

Data were collected through three primary sources: (1) Final project submissions were evaluated using the multi-dimensional rubrics (Tables 1–3); (2) Quantitative Artifact Assessment: Detailed performance metrics were recorded for projects at each level (Tables 4–6); (3) Competency Attainment Surveys: A custom-designed questionnaire was administered to measure students' self-efficacy in sustainability design (Table 7).

Table 4. Innovation Stage Evaluation Framework.

Evaluation Dimension	Specific Criteria	Score	Performance Indicators
Innovativeness and Originality	1. Original technical or sustainability solutions 2. Novel technology application for environmental/social challenges 3. Creative system integration and problem-solving	30 pts	Solution originality (0–10) Technology application (0–10) Creative problem-solving (0–10)
Standard Compliance and Quality	1. Code quality emphasizing efficiency and maintainability 2. Hardware meeting specs and environmental regulations 3. Documentation following industry and sustainability standards	25 pts	Code standard compliance (0–9) Hardware regulation adherence (0–8) Documentation quality (0–8)
Social and Environmental Value	1. Clear demonstration of practical problem-solving with environmental/social benefits 2. Public dissemination through competitions, publications, community engagement 3. Assessment of potential impact and scalability	25 pts	Problem-solving relevance (0–9) Results dissemination (0–8) Impact assessment (0–8)
Interdisciplinary Team Leadership	1. Effective coordination of interdisciplinary collaboration 2. Comprehensive documentation with clear interface specs 3. Managing team processes for technical and sustainability objectives	20 pts	Leadership effectiveness (0–7) Documentation completeness (0–7) Process management (0–6)

Table 5. Foundation Layer Project Evaluation (Example: Low-Power Digital Oscilloscope).

Evaluation Category	Specific Indicators	Criteria	Weight	Sample Data
Hardware Function	ADC Sampling Rate	≥ 1 MHz	25 per	1.2 MHz
Software Function	Trigger Success Rate	≥ 90	25 per	92 per
System Performance	Response Time	≤ 50 ms	30 per	45 ms
Documentation	Schematic/PCB/ Test Report	Complete	15 per	90 per complete
Team Collaboration	Role Definition and Progress	Clear, ≤ 3 day deviation	5 per	Clear, 2-day deviation

Table 6. System Layer Project Evaluation (Example: Energy-Aware Robot Navigation).

Evaluation Category	Specific Indicators	Criteria	Weight	Sample Data
Functional Integrity	SLAM Map Creation Success	≥ 90 per	30 per	92 per
System Performance	Communication Reliability	≤ 1 per packet loss	20 per	0.8 per
Hardware-Software Integration	Interface Standardization	Complete, consistent	15 per	90 per
Engineering Standards	Code Quality Metrics	80 per compliance	15 per	85 per
Sustainability Performance	Power Management Efficiency	85 per low-power time	20 per	88 per

Table 7. Innovation Layer Project Evaluation (Example: Multi-Robot Coordination System).

Evaluation Category	Specific Indicators	Criteria	Weight	Sample Data
Communication Performance	Operational Range	≥ 50 m	25 per	60 m
Coordination Capability	Positioning Accuracy	≤ 10 cm error	15 per	8 cm
System Reliability	Continuous Operation	≥ 1 h fault-free	25 per	2 h
Innovation and Optimization	Green Technology Integration	Use of sustainable techniques	20 per	Solar charging implemented
Documentation and Analysis	Sustainability Impact Assessment	Comprehensive analysis report	15 per	85 per complete assessment

3.5.4. Data Analysis

Quantitative data from rubric scores and project metrics were analyzed using descriptive statistics and paired *t*-tests to identify significant improvements (Table 8). Qualitative data from interviews and open-ended survey responses were analyzed using thematic analysis.

Table 8. Student Sustainable Engineering Competency Attainment (30 Students).

Competency Category	Specific Items	Students Achieving	Rate (%)	Description
Foundational Technical	Minimum System Construction with Power Awareness	28	93.3	Independently build STM32 systems with integrated power management
	RTOS Task Creation and Management	25	83.3	Create/manage tasks in uC/OS with power state attention
	Serial Communication with Power Optimization	20	66.7	Implement UART with sleep/wake cycling for power savings
System Integration	Hardware-Software Interface Design	25	83.3	Design interfaces with efficiency focus
	Multitask Scheduling and Synchronization	23	76.7	Implement scheduling balancing performance and power
	Communication Protocol Implementation	21	70.0	Implement I2C, SPI with error handling and power management
Sustainability and Innovation	Participation in Sustainability Competitions	19	63.3	Active participation in environmental/social impact competitions
	Competition Awards and Recognition	11	36.7	Awards in provincial/national sustainability/innovation competitions
	Lifecycle Thinking Application	15	50.0	Consider full product lifecycle in design and documentation

Table 9. Paired -Sample *t*-tests Analysis of Project Achievement Metrics (N = 30).

Category	Metric	Pre M (SD)	Post M (SD)	<i>t</i> (29)	<i>p</i>	Cohen's <i>d</i>	ICC (95 per CI)
Basic Projects	Completion Time (days)	4.0 (0.8)	1.5 (0.4)	−9.12	<0.001	3.98	0.92 (0.87–0.96)
	Code Reuse (%)	25 (10)	52 (12)	9.85	<0.001	2.45	0.94 (0.89–0.97)
	Sustainability Score	45 (15)	82 (10)	10.14	<0.001	2.87	0.91 (0.84–0.95)
Comprehensive Projects	Completion Time (days)	6.0 (0.8)	4.0 (0.5)	−15.59	<0.001	3.00	0.93 (0.88–0.96)
	Code Reuse (%)	30 (12)	68 (14)	15.90	<0.001	2.91	0.95 (0.91–0.97)
	Sustainability Score	62 (18)	88 (16)	8.33	<0.001	1.53	0.89 (0.80–0.94)
Innovation Projects	Completion Time (days)	10.0 (0.8)	8.0 (0.6)	−15.24	<0.001	2.83	0.90 (0.82–0.95)
	Code Reuse (%)	40 (14)	62 (12)	9.21	<0.001	1.68	0.93 (0.87–0.96)
	Sustainability Score	62 (15)	70 (11)	3.25	<0.01	0.61	0.88 (0.78–0.93)

Note: M = Mean, SD = Standard Deviation. Statistical significance: $p < 0.05$. "Pre-intervention data were obtained from the students' prior-semester embedded systems projects, assessed using the same rubrics to ensure comparability."

3.5.5. Validity

To ensure the trustworthiness of the quantitative assessment, Inter-Rater Reliability was calculated. Three independent raters (two faculty members and one industry expert) scored a random sample of project portfolios. The results indicated acceptable levels of agreement.

3.5.6. Results

Figures 4–8 illustrate representative student projects realized within the proposed CDIO-oriented framework. Rather than remaining at a conceptual level, these artifacts demonstrate how sustainability considerations were explicitly embedded into system-level design tasks. For instance, Figures 4 and 5 showcase hardware implementations that optimize energy efficiency under real-world constraints, whereas Figures 6–8 highlight the iterative refinement of sustainability-oriented design documentation. Collectively, these examples substantiate the claim that the framework was not only theoretically sound but also practically enacted in authentic classroom settings.



Figure 4. Digital oscilloscope prototype demonstrating low-power hardware optimization.

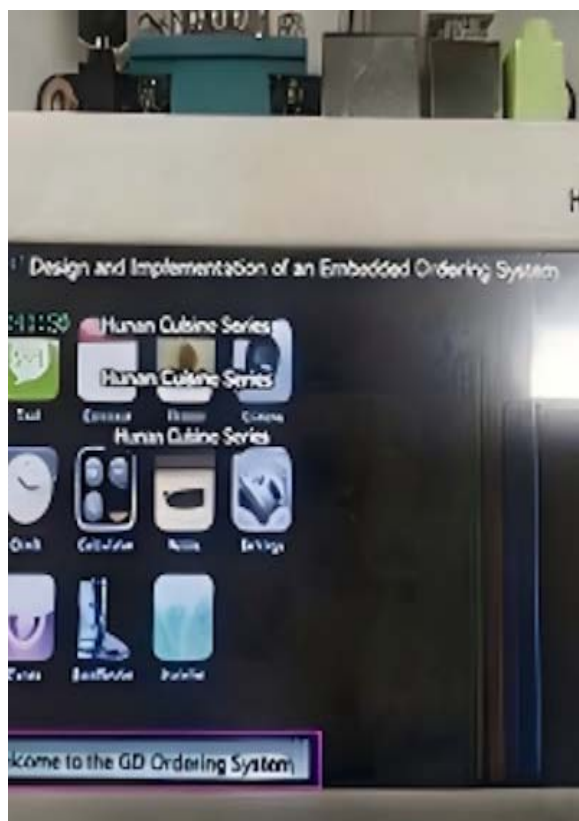


Figure 5. The ordering system.

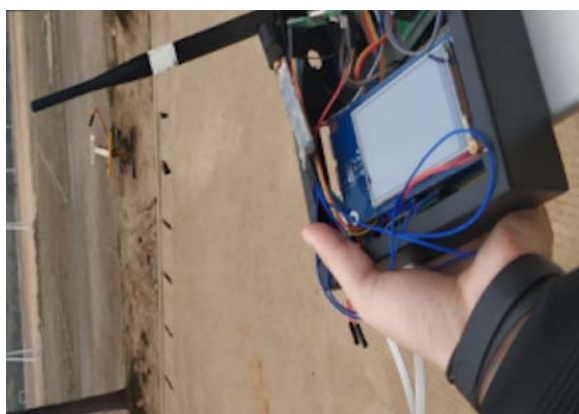


Figure 6. Multi-robot system illustrating system-level energy-aware coordination.

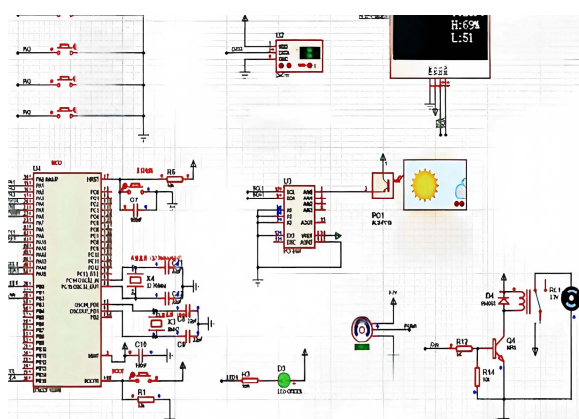


Figure 7. The smart home control system.

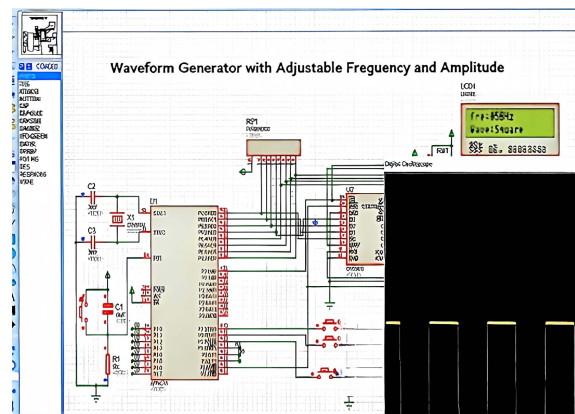


Figure 8. The waveform generator.

4. Theoretical Contribution and Practical Value

4.1. Contributions to Sustainable Embedded Systems Education

4.1.1. Integrated Sustainability-Technical Teaching Theory

This study makes two key contributions to Engineering Education for Sustainable Development. First, it demonstrates a scalable pedagogical model that effectively integrates sustainability principles into embedded systems education. This is evidenced by the percent 88 average score achieved by students in the Innovation Layer projects (Table 6) and the percent 63.3 participation rate in national sustainability competitions (Table 7). Second, We've proposed and implemented a comprehensive teaching theory integrating hardware knowledge, software skills, and sustainability competencies within a unified framework. This addresses the historical imbalance in embedded education that often emphasized hardware or software in isolation while neglecting environmental and social dimensions. Through the CDIO framework, we've created a coherent capability development pathway progressing from fundamental hardware awareness through software abstraction to sustainable system design. This theoretical contribution provides guidance for embedded course reform amid global sustainability challenges, offering a model for how technical education can evolve to address pressing environmental concerns without sacrificing engineering rigor.

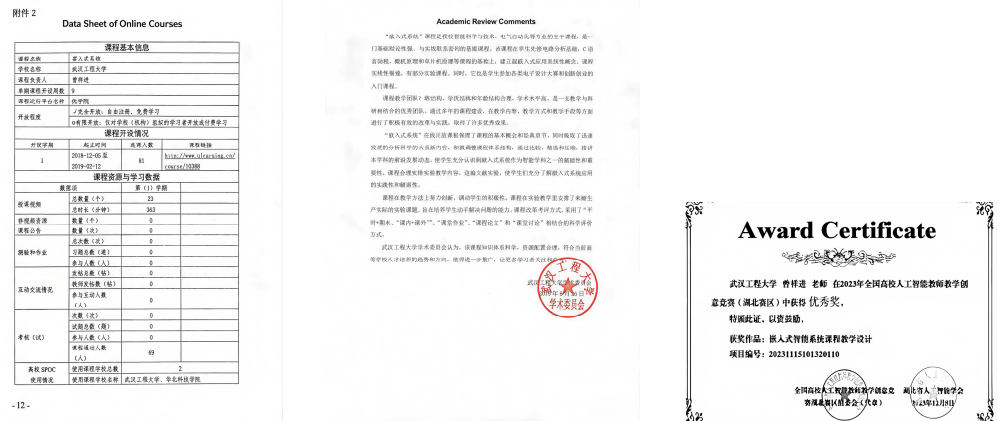
4.1.2. Three-Tier Progressive Curriculum Model for Sustainability Competency Development

We've developed and validated a three-tier progressive curriculum model (foundation-system-innovation) systematically defining sustainability competency development stages for embedded systems education. This addresses fundamental pedagogical questions of "what to teach" and "to what level" in sustainable engineering education. The model demonstrates strong practical applicability; other institutions can adapt it according to their disciplinary characteristics, student backgrounds, and resource constraints. By providing clear progression pathways and competency definitions, this model supports systematic development of engineers prepared to address sustainability challenges throughout their careers. Figure 9 demonstration of course impact and quality assurance with sustainability-oriented learning. Figure 9a Enrollment statistics of students in the online course. Figure 9b Formal academic assessment report issued by the university. Figure 9c Certificates of teaching honors awarded to the course.

4.1.3. Enhanced Multi-Dimensional Assessment System with Sustainability Integration

We've substantially enhanced traditional assessment by systematically incorporating sustainability metrics. Moving beyond exam-focused evaluation, we've developed a comprehensive system assessing project documentation quality, code efficiency, physical artifact

functionality, competition outcomes, and explicit sustainability performance indicators. Our assessment design incorporates industry sustainability standards, guiding students to value process quality, environmental impact, and social responsibility alongside technical achievement. This better reflects the comprehensive capabilities required in modern engineering practice and aligns educational outcomes with the sustainability priorities of forward-looking enterprises.



(a) Enrollment statistics of students in the online course	(b) Formal academic assessment report issued by the university	(c) Certificates of teaching honors awarded to the course
		

Figure 9. Online analytics confirming sustained engagement with sustainability-oriented learning.

4.1.4. Limitations and Future Work

While the results are promising, several limitations warrant attention. First, the study is constrained by its single-institution, single-cohort sample ($N = 30$). This limits the generalizability of the findings to diverse institutional contexts with varying resource levels. Second, the absence of a control group prevents us from making definitive causal claims that the observed improvements were solely due to the intervention, as maturation effects cannot be entirely ruled out. Third, the evaluation of the “Sustainability Officer” role relied primarily on qualitative self-reports, which may be subject to social desirability bias. Future research should address these limitations by conducting multi-institutional randomized controlled trials and developing automated tools to reduce the assessment burden on instructors.

4.2. Practical Implications for Engineering Education Reform

4.2.1. Curriculum Reconstruction Focus: Systematic Design of Sustainability-Driven Project Carriers

Our experience shows meaningful curriculum reconstruction requires more than adding isolated experiments or topics. Effective transformation involves designing coherent project sequences systematically developing capabilities from basic experimentation through comprehensive implementation to innovative practice. Each project must incorporate clearly defined sustainability-oriented competency objectives, specific technical requirements aligned with these objectives, and appropriate assessment criteria measuring both technical and sustainability outcomes. This project-based approach ensures sustainability is integrated throughout learning rather than treated as an isolated supplement to technical content.

4.2.2. Teacher Capability Transformation: From Content Lecturer to Sustainability Project Mentor

Successful implementation requires significant transformation of teacher roles and capabilities. Instructors must evolve from content deliverers to multifaceted mentors pos-

sessing sustainable engineering practice experience, pedagogical design skills for green technology projects, and current industry insight into sustainability trends and challenges. This transformation may require targeted professional development, industry engagement opportunities, and support for faculty participation in sustainability-focused RD initiatives. Teachers with direct sustainable technology development experience are better positioned to guide students in addressing real-world environmental challenges through their engineering work.

4.2.3. Development of Sustainability-Focused Teaching Resources

We've found effective sustainability integration requires specially designed teaching resources rather than adaptation of conventional materials. Our approach involves developing online resources specifically focused on sustainable embedded design—including micro-lectures on green design principles, simulations for energy and environmental impact analysis, and case studies of sustainable technology implementations. Physical resource development includes hardware platforms with integrated power measurement capabilities, debugging tools supporting efficiency analysis, and component libraries with environmental impact data. These specifically designed resources provide the material foundation for effective sustainability-integrated education, enabling students to develop both knowledge and practical skills needed for sustainable engineering practice.

4.2.4. Continuous Improvement Mechanism for Sustainability Education

We've implemented a systematic continuous improvement mechanism ensuring ongoing relevance and effectiveness of our sustainability-integrated approach. This incorporates multiple feedback channels including student surveys focused on sustainability competency development, analysis of competition outcomes and industry partnership results, and assessment of graduate performance in sustainability-focused roles. Evaluation findings inform targeted improvements such as adding prerequisite modules on environmental science fundamentals, updating technical content to reflect emerging green technologies, or enhancing industry collaboration around specific sustainability challenges. This iterative improvement process ensures the curriculum remains responsive to evolving sustainability priorities, technological developments, and industry needs.

5. Conclusions

This study demonstrates that embedding sustainability principles within a hybrid, project-based CDIO framework leads to statistically significant improvements in students' technical proficiency and sustainability competency. Beyond the specific context of embedded systems education, this research contributes to the broader discourse on Engineering Education for Sustainable Development by offering a scalable pedagogical blueprint. It validates that sustainability can be taught not as a separate topic, but as an integral dimension of quality and innovation in engineering design. Through such adaptation and implementation, engineering education can contribute to developing the sustainability-conscious engineers needed to address global challenges, creating a virtuous cycle in which educational theory informs practice, and practical experience enriches theoretical understanding in the vital domain of sustainable technology development. Several limitations temper the scope of these findings. First, the study was conducted with a single cohort ($N = 30$) at one institution, limiting generalizability. Second, the absence of a control group means causal inferences are tentative. Third, the reliance on instructor-generated data for some qualitative assessments may introduce bias. Future research should prioritize multi-institutional validation studies to test the framework's adaptability across diverse resource environments.

Funding: Major Program (JD) of Hubei Province (2025BEA005); Hubei Provincial Regional Science and Technology Innovation Plan Project (2025EIA091).

Institutional Review Board Statement: Ethical review and approval were waived for this study by Wuhan Institute of Technology as per internal research policy, this type of low-risk, anonymous educational survey qualifies for exemption from formal Institutional Review Board review.

Informed Consent Statement: Informed consent for participation was obtained from all subjects involved in the study.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to restrictions (privacy, legal or ethical reasons).

Acknowledgments: Author thanks Major Program (JD) of Hubei Province (2025BEA005) Foundation and Graduate Education and Teaching Project of Wuhan Institutes of Technology Foundation for a support grant.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Li, J.; Wang, Z.; Xue, L. Research and Exploration of Embedded System Teaching Based on CDIO Mode. *Comput. Educ.* **2010**, *12*, 122–124. [\[CrossRef\]](#)
2. Han, L.; Cao, Y.; Wu, X. Construction of Embedded System Curriculum System Based on CDIO. *Comput. Educ.* **2014**, *18*, 39–42. [\[CrossRef\]](#)
3. Li, L.; Gao, X.; Qin, H. Reform and Exploration of Embedded Artificial Intelligence System Experimental Course. *Lab. Res. Explor.* **2025**, *44*, 147–152.
4. Cao, L.; Yuan, Y.; Wang, T. Construction and Implementation of “Embedded System Design” Course under CDIO Mode—Taking “Smart Agriculture System” as an Example. *South. Agric. Mach.* **2021**, *52*, 185–187.
5. Webster, R. Shaping Engineering Technology Students’ Perceptions of Manufacturing Through Experiential Learning in a Flipped Classroom—A Case Study. *Int. J. Eng. Educ.* **2025**, *41*, 1342–1349.
6. Zheng, W.; Wen, S.; Lian, B.; Nie, Y. Research on a Sustainable Teaching Model Based on the OBE Concept and the TSEM Framework. *Sustainability* **2023**, *15*, 5656. [\[CrossRef\]](#)
7. Chen, B.; Xie, X.; Jiang, X. A Graduate Innovation Ability Cultivation Model Based on CDIO Mode—Taking System Simulation as an Example. *J. High. Educ. Res.* **2019**, *42*, 53–63.
8. Wang, B.; Ji, X.; Jia, H. Teaching Reform of “ARM Embedded System” Course. *J. Electr. Electron. Educ.* **2015**, *37*, 18–23.
9. Du, P. Application of Online Offline (O2O) Interactive Teaching Mode in Medical Ethics Teaching. *Chin. Med. Ethics* **2017**, *30*, 893–896.
10. Alani, F.; Grewal, R. Comparison Between In-person Versus Virtual Case-based Learning for an Upper Year Course in Engineering Technology Education. *Int. J. Eng. Educ.* **2024**, *40*, 499–510.
11. Zeng X.; Cai, D. Discussion on Innovative Teaching Reform of Embedded System Course Combined with Robot Project. *Innov. Educ. Res.* **2018**, *6*, 462–465.
12. Garcia, J.; Sinfield, J.; Yadav, A.; Adams, R. Learning Through Entrepreneurially Oriented Case-Based Instruction. *Int. J. Eng. Educ.* **2021**, *28*, 448–457.
13. Xie, X.; Ma, C.; Huang, Q. Research on Teaching Mode Reform Based on CDIO Concept—Taking “Principles and Technology of Mechanical Manufacturing Engineering” as an Example. *High. Educ. Res.* **2022**, *45*, 108–111.
14. Peng, Y.; Cheng, L. Reform and Practice of Embedded Teaching Integrating Maker Education Model. *J. North China Univ. Sci. Technol. (Soc. Sci. Ed.)* **2021**, *21*, 86–90.
15. Tong, Q.; Chang, F. Construction of a Practical Teaching System for the New Engineering Discipline of Electronic Information Based on Multi-dimensional Integration. *Res. Explor. Lab.* **2024**, *43*, 197–202.
16. Xu, T.; Zhang, C.; Liu, M. Application of Scaffolding Instruction in Embedded System Course Teaching for Graduate Students. *Comput. Educ.* **2024**, *2*, 187–191, 197.
17. Xu, J.; Wang, Z.; Shi, Y. Emerging Engineering Construction and Talent Cultivation Leading Future Industrial Transformation: Exploration and Practice of Microelectronics Talent Cultivation. *High. Eng. Educ. Res.* **2017**, *2*, 13–18.
18. Zhu, J.; Song, S.; Qin, L. Basic Practical Teaching Reform of Electronic Information Specialties Based on the Innovative Concept of “New Engineering”. *Exp. Technol. Manag.* **2017**, *11*, 171–173.
19. Xi, J.; Wang, X. Development of Landscape Architecture Design Students’ Pro-Environmental Awareness by Project-Based Learning. *Sustainability* **2022**, *14*, 2164.

20. Gutierrez-Bucheli, L.; Reid, A.; Kidman, G.; Lamborn, J. Exploring Response Levels to Sustainability in Engineering Curricula: A Cross-case Analysis. *Int. J. Sustain. High. Educ.* **2025**, *27*, 202–219. [[CrossRef](#)]
21. Dresel, M. Exploring the Role of Teacher Self-Efficacy and Personal Environmental Practices in Integrating Sustainability into Teaching: A Network Analysis of German Teachers. *Sustainability* **2025**, *17*, 7533. [[CrossRef](#)]
22. Crawley, E.F.; Malmqvist, J.; Ostlund, S.; Brodeur, D.R. *Rethinking Engineering Education: The CDIO Approach*; Springer: New York, NY, USA, 2007.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.