INTEGRATING UNPLUGGED AND PLUGGED CODING IN EARLY CHILDHOOD EDUCATION: TEACHER PERSPECTIVES AND CLASSROOM PRACTICES

A paper submitted in CI 677 at
Marshall University
in partial fulfillment
of
the requirements for the degree of
Education Specialist (EdS) and Doctorate in
Education (EdD)
in
Curriculum and Instruction
by
Amber Lynn Myers

Marshall University
March 2025

Table of Contents

List of Tables

No table or figure entries found.

List of Figures

No table or figure entries found.

Abstract

In today's rapidly evolving digital landscape, coding is increasingly recognized as a foundational literacy skill, beginning as early as preschool. This paper explores the integration of unplugged and plugged coding activities in early childhood education, emphasizing their role in fostering problem-solving, logical thinking, creativity, and computational thinking. Unplugged coding, which introduces fundamental coding concepts without screens, provides young learners with hands-on opportunities to engage with sequencing, pattern recognition, and directional language. The research identifies how these activities align with early learning standards, support mathematical reasoning, and scaffold computational thinking skills essential for future STEM learning. Inquiry-based learning, storytelling, and hands-on experiences help children build a strong foundation for understanding coding concepts before transitioning to digital tools such as robots and coding apps. By examining the strategies educators use to implement unplugged coding and its impact on engagement and skill development, this study provides insights into best practices for early coding education. The findings reinforce the importance of intentional coding integration in early childhood curricula, demonstrating that coding is not merely a technical skill but a powerful tool for developing cognitive and collaborative abilities in young learners.

Introduction

In today's digital world, technology is evolving rapidly, shaping the way we live, work, and communicate. While coding may seem like a skill reserved for computer scientists, it is a foundational literacy that children can begin learning as early as preschool. Early exposure to unplugged and plugged coding fosters problem-solving, logical thinking, and creativity—skills essential for success in the 21st century.

Computer programs use a digital language called coding or programming. This language gives machines instructions in a specific sequence. These codes are created using a sequence of binary numbers. Coding uses algorithms, which are a set of instructions used to complete a task or to solve a given problem in sequential order (Ricketts 2018). This type of instruction is starting to be used with younger children in early childhood classrooms. Teachers are starting to implement unplugged and plugged coding in their classrooms. Teachers are integrating different types of coding to scaffold an understanding of algorithms and sequencing (Ricketts 2018). Teachers are using games and stories to help children learn the digital language of coding.

Young children naturally engage in sequencing, pattern recognition, and cause-and-effect reasoning through play, all of which align with coding concepts. Unplugged coding, which introduces computational thinking without screens, and plugged coding, which utilizes digital tools, offer developmentally appropriate ways for children to explore algorithms, sequencing, and problem-solving. By integrating coding into early learning, educators provide children with the building blocks for future STEM success while reinforcing critical math, literacy, and executive functioning skills. Rather than being an advanced concept, coding can be a playful and engaging part of early childhood education, setting the stage for lifelong learning and adaptability in a technology-driven world.

Coding concepts can be found everywhere in our daily life. Patterns, procedures, routines, games, and following directions are some examples our children experience in school. Collaborating, problem-solving, predicting, organizing, planning, and evaluating are part of coding and should be experienced in classrooms also. STEAM (science, technology, engineering, art, and math) projects are one way to introduce these concepts. Unplugged coding is an innovative way to introduce children to complex thinking and problem-solving without computers or iPads. Teachers can reach many of the same goals using unplugged coding instead of traditional coding. Integrating coding allows children to make real-world connections.

Coding is commonly used with electronic devices such as computers or robots. Unplugged coding is using coding skills and algorithms without technology (McLennan, 2020). Math, literacy, executive function skills, problem-solving, communication, creativity, and imagination are being used when children learn to use and implement codes. Computational thinking is also utilized when children code. Ricketts (2018) describes computational thinking as skills, concepts, and behaviors used in computer science to solve problems. McLennan (2020) explains that children can take complex problems, understand them, and develop possible solutions using computational thinking skills. Coding helps children develop computational thinking skills through hands-on activities and projects.

As technology continues to redefine educational expectations, the early years of schooling have become a pivotal time to introduce foundational skills in computational thinking and coding. While coding is increasingly recognized as a critical component of STEM education, its implementation in early childhood settings remains limited and inconsistent (McLennan, 2020; Ricketts, 2018). A prevailing misconception is that coding is too abstract or developmentally inappropriate for young learners. This has led to a gap between the recognized

importance of coding and the pedagogical strategies employed in preschool and primary classrooms.

Emerging evidence suggests that young children are not only capable of engaging with coding concepts, but that early exposure—particularly through unplugged activities—supports a range of developmental domains, including executive function, mathematical reasoning, and language development (McLennan, 2020). Despite this, there is limited research exploring how early childhood educators effectively integrate unplugged coding within classroom projects, or how these experiences align with established learning standards and developmental milestones.

Furthermore, there is a need to understand how teachers scaffold unplugged coding in developmentally appropriate ways, and how children respond, both cognitively and affectively, to unplugged versus plugged coding experiences. As educational policy increasingly emphasizes the importance of 21st-century skills, it is critical to document and disseminate best practices for early coding integration. This study addresses that gap by investigating the strategies, supports, and outcomes associated with unplugged coding activities in early learning environments. By examining educator practices and student engagement, this research contributes to a growing body of evidence supporting the intentional inclusion of computational thinking in early childhood curricula.

Questions

How do early childhood educators integrate unplugged coding into classroom projects?

In what ways do unplugged coding activities align with early learning standards and developmental milestones?

How do teachers scaffold unplugged coding experiences to support problem-solving and computational thinking in young learners?

How do students respond to unplugged coding activities compared to plugged coding experiences regarding engagement and skill development?

Literature Review

Unplugged coding, which involves teaching coding concepts without the use of technology, provides a developmentally appropriate entry point for young learners to build foundational computational thinking skills (McLennan, 2020). These activities allow children to engage in hands-on experiences that support algorithmic thinking, sequencing, directional language, and problem-solving in playful, screen-free environments. Inquiry-based programs frequently integrate both unplugged and plugged coding because these activities are flexible and can support a range of learning domains. As McLennan (2020) explains, "It's not the act of coding but the thinking within the coding that is important" (p. 6), emphasizing the value of the learning environment coding creates over the technology itself.

Research highlights the connection between early coding experiences and the development of essential mathematical and cognitive skills. Activities involving coding promote spatial awareness, pattern recognition, number sense, reasoning, and an understanding of directional language (Bers, 2018; Strawhacker & Bers, 2019). These experiences also encourage collaboration, communication, and creativity as children work in teams to solve problems and design algorithms. Through shared tasks, children assume roles, collaborate on sequences, and engage in meaningful discussions, developing both interpersonal and cognitive competencies. Inquiry-based learning aligns closely with computational thinking. McLennan (2020) identifies parallels between these two approaches, including breaking down complex problems, organizing and analyzing information, testing solutions, and applying strategies across domains. Within this framework, teachers act as co-learners alongside children, fostering exploration, discovery, and

reflection. Sullivan and Bers (2016) further note that early coding experiences promote storytelling, sequencing, and creativity—key components of early childhood development. Practical applications of unplugged coding include using maps, board games, and storytelling to introduce commands, directional terms, and ordinal numbers. These activities help children build mathematical understanding through experiences with movement, counting, location, and sequencing. Coding stories—retelling or altering familiar tales—enhances vocabulary and comprehension during read-alouds, while also reinforcing spatial reasoning as children program characters through story settings using directional codes.

Creating grids for movement-based coding supports spatial awareness, measurement, and collaborative planning. Children and educators can build grids using everyday materials such as placemats, floor tape, or foam mats. These grids become versatile tools for storytelling and gameplay, allowing children to design settings, identify obstacles, and determine character paths. Children then develop and record commands using arrows or symbols and take turns being the "programmer" to narrate and guide character movement. Adult modeling and guided practice help children gradually take ownership of the process.

As children develop confidence, looping can be introduced. Loops—repeating sequences of instructions—help children understand patterns more deeply. McLennan (2020) recommends using pattern blocks and yarn to visually represent loops and segment repeating units, thereby introducing a key coding concept through hands-on materials.

Once foundational skills are established, a coding center can be introduced for small-group work. Materials can rotate weekly to maintain engagement and encourage creative exploration. Storytelling boxes and sequencing cards can supplement these activities. The gradual transition to plugged coding involves integrating simple robotics, such as Bee-Bots, Mouse Bots, and Indy

cars, which allow children to apply their unplugged knowledge using tangible devices. These robots do not require computers; they rely on physical or color-based commands, easing the transition from analog to digital environments.

Integrating unplugged and plugged coding supports the development of spatial reasoning, pattern recognition, and mathematical thinking while encouraging collaboration and creativity. These activities provide rich opportunities for early STEM engagement and align with developmental learning goals. Introducing coding vocabulary and concepts early allows children to build confidence and competence in computational thinking before transitioning to more complex digital tasks (McLennan, 2020).

As children move into higher grades, the shift to technology-based tools like apps and programmable robots builds on the conceptual foundation established through unplugged activities. This progression supports more advanced problem-solving and deeper engagement. Coding toys—whether physical, digital, or hybrid—develop not only coding knowledge but also creativity, critical thinking, and higher-order thinking skills (Pollarolo et al., 2024).

This study adds to the existing literature by analyzing how unplugged and plugged coding foster early computational thinking and problem-solving skills. While prior research emphasizes unplugged coding as a precursor to digital programming (McLennan, 2020), this investigation explores how early vocabulary development and directional language support this transition. The findings will also provide educators with actionable strategies for scaffolding unplugged activities in ways that align with young learners' developmental milestones and curriculum goals. By highlighting the progression from unplugged to plugged coding, this research underscores the importance of hands-on, inquiry-based learning in cultivating future-ready computational thinkers.

Methodology

Research Design

This study utilized a quantitative descriptive survey design to investigate how early childhood educators integrate unplugged coding activities into their instructional practices. The survey was developed to address four core research questions related to:

1. Integration of unplugged coding into classroom projects
2. Alignment with early learning standards and developmental domains
3. Instructional scaffolding strategies used during unplugged coding
4. Student engagement and learning outcomes in unplugged versus plugged coding activities

This design enabled the collection of structured, quantifiable data to capture current practices and perceptions among early childhood educators.

Participants

A total of 19 early childhood educators participated in the study. Participants taught Pre-K through 2nd grade and included both general education and specialized educators. The sample reflected a range of teaching experience and represented diverse educational settings. Participants were recruited through professional networks and early childhood education organizations.

Instruments

Data were collected using a structured online survey developed specifically for this study. The survey included:

- Multiple-choice items on demographics and teaching experience

- Frequency scales (e.g., “Never,” “Occasionally,” “Frequently”) to measure coding activity usage
- Checklists to indicate types of coding activities, integration areas, and scaffolding strategies
- Comparative rating items to evaluate unplugged versus plugged coding in terms of student engagement and learning benefits

Survey items were grouped according to the study’s four research questions and aligned with key constructs related to integration, alignment, scaffolding, and outcomes.

Procedures

Participants were invited to complete the survey via email. After reviewing and consenting to participate, educators accessed and completed the online survey, which took approximately 10–15 minutes. Data collection took place over two weeks, and all responses were anonymized and securely stored for analysis.

Data Analysis

Survey responses were analyzed using descriptive statistics, including frequencies and percentages, to identify trends in usage, perceptions, and instructional strategies. Results were organized by research question, as outlined below:

Frequency of Use

- 63% of educators reported using unplugged coding “occasionally,”
- 21% used them “frequently,”
- 16% reported using them “rarely” or “never.”

Integration Types

- Unplugged coding activities were most often integrated into literacy centers (68%), math centers (58%), and STEM projects (47%).
- Common activities included directional games, story sequencing, and logic puzzles.
- Pre-K and Kindergarten teachers reported higher use of unplugged coding compared to 1st and 2nd grade teachers, who were more likely to use plugged coding tools.

Alignment with Standards

- 74% of respondents agreed or strongly agreed that unplugged coding supports problem-solving, sequencing, and language development.
- Highest perceived alignment was in:
    - Mathematics (79%) – especially in patterning and direction
    - Language & Literacy (63%) – through storytelling and sequence-based play
    - Social-Emotional Development (42%) – such as turn-taking and collaboration

Scaffolding Strategies

- Common scaffolding strategies included:
    - Modeling (84%)
    - Guided questioning (68%)
    - Think-alouds (53%)
    - Peer collaboration (47%)
- Several educators indicated use of differentiated supports, such as visual aids, manipulatives, and step-by-step instructions.
- Teachers with more than 10 years of experience were more likely to report using multi-sensory scaffolding approaches.

Student Engagement and Outcomes

- 79% of educators rated student engagement with unplugged coding as high or very high.
- In comparison, 58% rated plugged coding engagement similarly, though 26% noted that plugged tools could lead to distraction or technical difficulties.
- Unplugged coding was perceived as particularly effective for:
    - Problem-solving (89%)
    - Sequencing and logical reasoning (74%)
    - Collaboration and communication (58%)
- Plugged coding was considered more effective for supporting fine motor development and technology skills.

Results

Findings from the structured survey revealed several key trends in how early childhood educators integrate unplugged coding into their instructional practices. The majority of respondents (63%) reported using unplugged coding activities occasionally, with 21% using them frequently. These activities were commonly integrated into literacy (68%) and math centers (58%), with additional implementation in STEM projects (47%). Pre-K and Kindergarten teachers reported higher usage of unplugged coding compared to their 1st and 2nd grade counterparts, who were more likely to incorporate plugged tools alongside unplugged strategies. In terms of alignment with early learning standards, 74% of educators agreed or strongly agreed that unplugged coding supports essential domains such as problem-solving, sequencing, and language development. Alignment was most frequently noted in mathematics (79%), followed by language and literacy (63%) and social-emotional development (42%). Educators reported a variety of scaffolding strategies, with modeling (84%), guided questioning (68%), and think-

alouds (53%) being the most prevalent. Differentiation techniques—such as using visuals, manipulatives, or simplified steps—were also commonly used, particularly by more experienced educators.

Regarding student outcomes, 79% of participants rated engagement in unplugged coding as high or very high. Educators perceived unplugged activities as especially beneficial for developing problem-solving (89%), sequencing and logical reasoning (74%), and collaboration (58%). Plugged coding was viewed as more effective for developing fine motor skills and technology familiarity, though 26% of respondents reported potential distractions with digital tools.

Implications

The findings of this study underscore the growing importance and potential of unplugged coding as a foundation for 21st-century learning in early childhood education. Beyond computational thinking, coding offers a new and innovative approach to teaching communication, sequencing, and problem-solving skills in developmentally appropriate ways. Educators who integrated unplugged coding into literacy and math centers noted that these activities naturally supported early learning standards, especially in patterning, storytelling, and collaboration.

The implementation of coding in early childhood classrooms—first inspired by guiding Little Red Riding Hood through the woods—demonstrated that even young children can engage in algorithmic thinking through storytelling and play. The release of *Kindercoding Unplugged* in 2020 further accelerated this integration, providing educators with research-informed strategies that align with inquiry-based learning and the Reggio Emilia Approach. Educators discovered that successful coding instruction required flexibility, open-mindedness, and thoughtful scaffolding using modeling, guided questions, and peer collaboration.

Importantly, this study supports findings from Su & Yang (2023), which indicate that children as young as four can demonstrate core computational thinking skills such as sorting, sequencing, and debugging when coding activities are developmentally appropriate. These activities also promote non-cognitive outcomes like communication, collaboration, and motor coordination. As the data from this study show, unplugged coding activities are not only engaging but also highly adaptable across content areas, allowing educators to embed coding into early literacy, math, and social-emotional learning.

The implications for teacher preparation and curriculum development are significant. Professional development should emphasize the value of unplugged coding as both a cognitive and social-emotional learning tool. Teachers can use these experiences to build a foundation for future coding with technology while ensuring alignment with standards and developmental milestones.

Conclusion

Coding is a critical yet underutilized component of early childhood education. This study demonstrates that unplugged coding activities can be effectively integrated into Pre-K through 2nd grade classrooms to support a wide range of learning outcomes. With strong alignment to core standards in literacy, math, and problem-solving, unplugged coding offers a screen-free, hands-on way to introduce computational thinking to young learners.

The journey of incorporating coding in classrooms—beginning with story-based activities and evolving into structured group projects—highlights how children thrive when given the opportunity to explore and create. Coding fosters higher-order thinking, communication, and creativity, reinforcing the idea that it is not limited to technology use but rooted in how children express and carry out their thinking.

This study contributes to the field by providing real-world data on the frequency, strategies, and outcomes associated with unplugged coding in early childhood classrooms. It affirms that coding can and should be introduced in the early years, with intentional planning and scaffolding. Future research will expand on this work by implementing five unplugged coding activities in Pre-K and Kindergarten classrooms, observing how children learn to code stories and collaborate to teach others. Once foundational skills are in place, these classrooms will transition to plugged coding using tools such as Indy for color-based programming.

As technology continues to evolve, early experiences with coding will be essential in shaping how children think, collaborate, and solve problems. Embedding unplugged coding into early education ensures that learners not only develop computational thinking skills but also gain confidence, adaptability, and curiosity—skills that are foundational for lifelong learning and success in a digital world.

References

Bers, M. U. (2018). *Coding as a playground: Programming and computational thinking in the early childhood classroom*. Routledge.

Lee, J., & Junoh, J. (2019). Implementing Unplugged Coding Activities in Early Childhood Classrooms. *Early Childhood Education Journal,* , 1-8. https://doiorg.marshall.idm.oclc.org/10.1007/s10643-019-00967-z

KALDOR, T. A. M. A. R. A. (2017, February 1). *The T in STEM: Creating play-based experiences that support children's learning of coding and higher order thinking*. NAEYC. https://www.naeyc.org/resources/blog/creating-play-based-experiences

McLennan, D. (2017). *Creating coding stories and games*. NAEYC. https://www.naeyc.org/resources/pubs/tyc/feb2017/creating-coding-stories-and-games

McLennan*, D. (2020).* Unplugged coding: Why screen-free coding is a powerful tool in early childhood education*. Exchange Press.*

McLennan, D. P. (2020). *Kindercoding unplugged: Screen-free activities for Beginners*. Redleaf Press.

Pollarolo, E., Papavlasopoulou, S., Granone, F., & Reikerås, E. (2024). Play with coding toys in early childhood education and care: Teachers' pedagogical strategies, views and impact on children's development. A systematic literature review. *Entertainment Computing*, *50*, 100637. https://doi.org/10.1016/j.entcom.2024.100637

Ricketts, R. (2018, March 13). *Computational thinking for Kindergartners*. Edutopia. https://www.edutopia.org/article/computational-thinking-kindergartners

Strawhacker, A., & Bers, M. U. (2019). “I want my robot to look for food”: Comparing children’s programming comprehension in virtual and physical robotics environments. International Journal of Technology and Design Education, 29(3), 493–510. https://doi.org/10.1007/s10798-018-9445-7

Sullivan, A., & Bers, M. U. (2016). Girls, boys, and bots: Gender differences in young children’s performance on robotics and programming tasks. Journal of Information Technology Education: Innovations in Practice, 15, 145–165. https://doi.org/10.28945/3547

Su, J., & Yang, W. (2023). A systematic review of integrating computational thinking in early childhood education. *Computers and Education Open*, *4*, 100122. https://doi.org/10.1016/j.caeo.2023.100122

Survey Questions

Background Information

1. What grade level(s) do you teach? (Select all that apply)

    - Pre-K
    - Kindergarten
    - Other

2. How many years of teaching experience do you have?

    - 0-2 years
    - 3-5 years
    - 6-10 years
    - 10+ years

Current Coding Integration

3. How often do you incorporate unplugged coding activities in your classroom?

    - Never
    - Rarely (1-2 times per year)
    - Occasionally (monthly)
    - Frequently (weekly)
    - Regularly (daily)

4. What types of unplugged coding activities do you use? (Select all that apply)

    - Sequencing games (e.g., ordering picture cards)
    - Directional movement (e.g., stepping forward/backward to follow commands)
    - Pattern recognition activities
    - Story-based coding (e.g., storytelling with coding logic)
    - Other

Instructional Approaches

5. How do you introduce coding vocabulary to students? (Select all that apply)

    - Through direct instruction (explicitly teaching terms like "sequence" and "loop")
    - During hands-on activities
    - Through stories or books about coding
    - I do not introduce coding vocabulary

6. To what extent do you align unplugged coding activities with early learning standards?

- Not at all
- Slightly aligned
- Somewhat aligned
- Mostly aligned
- Completely aligned

7. What challenges do you face when integrating unplugged coding in your classroom? (Select all that apply)

    - Lack of resources or materials
    - Limited professional development or training
    - Difficulty aligning with curriculum standards
    - Student engagement or understanding
    - Time constraints

Impact on Learning

8. Do you believe unplugged coding benefits young learners?

    - Yes
    - No
    - Unsure

9. Have you observed improvements in students' problem-solving or logical thinking skills after engaging in unplugged coding activities?

    - Yes
    - No
    - Unsure

10. How do students typically respond to unplugged coding activities? (Scale of engagement)

    - Not engaged at all
    - Slightly engaged
    - Somewhat engaged
    - Mostly engaged
    - Highly engaged

Mathematical Thinking & Spatial Awareness

11. Do you believe unplugged coding activities help develop early mathematical skills (e.g., pattern recognition, sequencing, number sense)?

    - Yes
    - No

- Unsure

12. How often do your unplugged coding activities incorporate spatial reasoning tasks (e.g., directional movement, mapping, or object positioning)?

    - Never
    - Rarely (1-2 times per year)
    - Occasionally (monthly)
    - Frequently (weekly)
    - Regularly (daily)

13. In your experience, do students who engage in unplugged coding show stronger spatial awareness compared to those who do not?

    - Yes
    - No
    - Unsure

Plugged Coding with Bots and Devices

14. Do you use plugged coding activities (e.g., coding apps, programmable robots) in your classroom?

    - Yes
    - No

15. Which plugged coding tools or devices do you use? (Select all that apply)

    - Bee-Bot or Code & Go Mouse
    - ScratchJr or Kodable
    - Osmo Coding
    - Dash & Dot or Sphero
    - Other
    - I do not use plugged coding tools

16. How often do students engage with plugged coding tools in your classroom?

    - Never
    - Rarely (1-2 times per year)
    - Occasionally (monthly)
    - Frequently (weekly)
    - Regularly (daily)

17. How would you rate students' engagement with plugged coding tools compared to unplugged coding activities?

- More engaged with plugged coding
- More engaged with unplugged coding
- Engaged equally in both
- Neither is highly engaging

18. What is the biggest barrier to implementing plugged coding in your classroom? (Select all that apply)

    - Lack of funding for devices
    - Limited access to technology
    - Lack of training or professional development
    - Not enough time in the schedule
    - No interest in using plugged coding tools

Future Implementation and Support

19. What additional support or resources would help you implement unplugged or plugged coding more effectively? (Select all that apply)

    - Professional development workshops
    - Pre-made lesson plans and materials
    - More classroom time dedicated to coding
    - Funding for resources
    - None, I feel confident in my current approach

20. Are you interested in learning more about integrating unplugged and plugged coding into early childhood education?

    - Yes, through professional development workshops
    - Yes, through online resources or lesson plans
    - No, I feel confident in my current knowledge
    - Other