PMMI PROGRAMMABLE LOGIC CONTROLLERS (PLC) 1 PRACTICE TEST (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://pmmiplc1.examzify.com

Copyright © 2026 by Examzify - A Kaluba Technologies Inc. product.

ALL RIGHTS RESERVED.

No part of this book may be reproduced or transferred in any form or by any means, graphic, electronic, or mechanical, including photocopying, recording, web distribution, taping, or by any information storage retrieval system, without the written permission of the author.

Notice: Examzify makes every reasonable effort to obtain accurate, complete, and timely information about this product from reliable sources.

SAMPLE

Table of Contents

Copyright 1

Table of Contents 2

Introduction 3

How to Use This Guide 4

Questions 5

Answers 8

Explanations 10

Next Steps 15

SAMPLE

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- Practice answering questions under realistic conditions,
- Improve accuracy and speed,
- Review explanations to strengthen weak areas, and
- Approach the exam with greater confidence.

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

SAMPLE

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here's how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 – 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don't just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There's no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

SAMPLE

Questions

SAMPLE

1. What is the time base used by PLCs?

A. Milliseconds (0.001 s)

B. Seconds

C. Minutes

D. Hours

2. How would you perform an addition between two integers in PLC logic, and where is the result stored?

A. Use ADD instruction and store result in a destination integer variable.

B. Use SUB instruction and store result in a destination integer variable.

C. Use AND instruction and store result in a destination boolean.

D. Use OR instruction and store result in a source integer.

3. In ladder logic, what does a normally closed (NC) contact represent and how does it affect the coil when its input is false?

A. An NC contact is open when its input is true and closed when false; when closed it can energize the coil.

B. An NC contact is closed when input is true and energizes coil.

C. An NC contact is always open.

D. An NC contact cannot energize the coil.

4. Which bit is designated as the Preset value in retentive timer bit components?

A. .ACC

B. .PRE

C. .DN

D. .EN

5. How do I/O modules communicate with the PLC processor?

A. The input module transmits data to the processor on a regular basis so that the processor can collect the information via scan.

B. The input module stores data locally and does not communicate with the processor.

C. The input module only sends data when manually triggered by a technician.

D. The input module uses wireless to report data to a separate controller.

SAMPLE

6. What does the PLC Counter Instruction control?

A. It provides control logic by setting and resetting its Done bit after it has been turned on and off a predetermined number of times instead of after a time delay

B. It measures elapsed time

C. It delays outputs by a fixed amount of time

D. It duplicates the input on the output

7. What happens if you do not reset a latch in PLC control?

A. It maintains the condition.

B. It triggers an alarm.

C. It automatically resets when input clears.

D. It powers down the PLC.

8. What is the difference between local and global variables in PLC programming?

A. Local variables exist within a function block or program; global variables are accessible from multiple blocks.

B. Local variables are global across all blocks; global variables are only in the main program.

C. Local variables are only used for timers; global for I/O.

D. Local variables persist after power down; global do not.

9. What is the purpose of a subroutine or function block in PLC programming, and when would you use one?

A. To reuse code and organize logic; call a common function from multiple places.

B. To reduce memory usage by compressing code.

C. To disable the PLC.

D. To generate random numbers.

10. Which statement best contrasts relay and solid state switching?

A. Relay contacts and solid state switches.

B. Diodes and transistors.

C. Resistors and capacitors.

D. Open-circuit switches only.

SAMPLE

Answers

SAMPLE

1. A
2. A
3. A
4. B
5. A
6. A
7. A
8. A
9. A
10. A

SAMPLE

Explanations

SAMPLE

1. What is the time base used by PLCs?

A. Milliseconds (0.001 s)

B. Seconds

C. Minutes

D. Hours

Timing in PLCs hinges on a fixed smallest step, the time base, which is millisecond-scale. Most PLC timers count in discrete ticks of 1 millisecond, so you set delays and durations using milliseconds (0.001 s). For example, a 2-second delay is programmed as 2000 milliseconds. Minutes or hours aren't the base timing unit; they are just longer durations expressed in milliseconds. This millisecond time base lets timers and pulse logic operate with fine, predictable granularity.

2. How would you perform an addition between two integers in PLC logic, and where is the result stored?

A. Use ADD instruction and store result in a destination integer variable.

B. Use SUB instruction and store result in a destination integer variable.

C. Use AND instruction and store result in a destination boolean.

D. Use OR instruction and store result in a source integer.

Addition of two integers in PLC logic is done with the ADD instruction. It takes two integer inputs and stores their sum in a destination that is defined as an integer. This is the best approach because it directly performs arithmetic and places the result in a variable you can use later in the program. Using the SUB instruction would perform subtraction instead of addition, even though it can also write to a destination integer. The AND and OR instructions operate on bits or booleans and do not produce a numeric sum, so they wouldn't give you an integer result for adding two numbers.

3. In ladder logic, what does a normally closed (NC) contact represent and how does it affect the coil when its input is false?

A. An NC contact is open when its input is true and closed when false; when closed it can energize the coil.

B. An NC contact is closed when input is true and energizes coil.

C. An NC contact is always open.

D. An NC contact cannot energize the coil.

A normally closed contact conducts current when its input is not active and breaks the circuit when the input is active. So when the input is false, the NC contact is closed and provides a path for current to energize the coil (assuming the rest of the rung is also complete). When the input becomes true, the NC contact opens, breaking the path and de-energizing the coil. This explains why the correct interpretation is that the coil can energize when the NC input is false. The other descriptions don't fit: a normally closed contact does not stay open when the input is true, it opens then; it is not always open, it is closed when the input is false; and it can energize the coil when the path is complete.

SAMPLE

4. Which bit is designated as the Preset value in retentive timer bit components?

A. .ACC

B. .PRE

C. .DN

D. .EN

The preset value is the target time the timer counts to before it finishes. In a retentive timer bit, the value you load as the goal is stored in the PRE input. The timer uses ACC to keep the current elapsed count and compares ACC to PRE; when ACC reaches PRE, the Done flag (DN) goes high. The Enable input (EN) starts or enables the timing. So the bit designated for the preset value is PRE.

5. How do I/O modules communicate with the PLC processor?

A. The input module transmits data to the processor on a regular basis so that the processor can collect the information via scan.

B. The input module stores data locally and does not communicate with the processor.

C. The input module only sends data when manually triggered by a technician.

D. The input module uses wireless to report data to a separate controller.

I/O modules communicate with the PLC processor through a regular scan cycle. At the start of each cycle, the input modules sample their physical signals (sensors, switches) and place those values into the processor's input image. The processor then executes the program using those current input values. After the logic runs, the processor writes the results to the output image, and the output modules drive the actual physical outputs. This cyclic, synchronized data exchange keeps the system deterministic and ready to respond to changes on every scan. That's why the other options don't fit: inputs aren't stored locally without communicating with the processor, they aren't sent only when manually triggered, and standard I/O typically isn't wireless reporting to a separate controller within the core PLC operation.

6. What does the PLC Counter Instruction control?

A. It provides control logic by setting and resetting its Done bit after it has been turned on and off a predetermined number of times instead of after a time delay

B. It measures elapsed time

C. It delays outputs by a fixed amount of time

D. It duplicates the input on the output

The Counter Instruction is used to trigger actions after a specific number of signal activations rather than after a time delay. It watches an input, and each time that input goes on, it increments its accumulated count. When the count reaches the preset value, the Done bit is set, signaling that the target has been reached and allowing the rest of the logic to proceed. The counter can then be reset to start a new cycle as needed. This is distinct from measuring elapsed time, which is what a timer does, and from delaying outputs or copying the input to the output, which are not the functions of a counter.

SAMPLE

7. What happens if you do not reset a latch in PLC control?

A. It maintains the condition.

B. It triggers an alarm.

C. It automatically resets when input clears.

D. It powers down the PLC.

A latch is a memory element in PLC logic that holds whatever state it was set to until a reset action occurs. If you do not reset it, the latched condition remains active across PLC scans, even if the input that activated it goes away. This is why the output stays in its latched state until you explicitly clear it with a reset signal (or power down in some cases). So the effect is that the latch maintains the condition, rather than automatically clearing or triggering something else. The other options describe behaviors that would require additional logic (an alarm), an automatic clearing on input drop, or power-down effects, none of which are inherent to simply not resetting a latch.

8. What is the difference between local and global variables in PLC programming?

A. Local variables exist within a function block or program; global variables are accessible from multiple blocks.

B. Local variables are global across all blocks; global variables are only in the main program.

C. Local variables are only used for timers; global for I/O.

D. Local variables persist after power down; global do not.

Understanding variable scope is crucial in PLC programming. Local variables live inside the block they're declared in (such as a function block or a specific program) and aren't visible to other blocks unless they're passed as parameters. Each time the block runs, it uses its own local data, making that data private to that block's instance. Global variables sit in a global scope and can be accessed by many blocks across the program, providing a shared place to store values that multiple parts of the project need to read or update. This separation helps keep code organized and prevents unintended interactions: use locals for block-specific or temporary data, and globals for information that must be shared, like system mode, a common counter, or a status flag that multiple blocks react to. The other statements either invert the scope idea or mix up persistence concepts, which aren't the defining difference here.

9. What is the purpose of a subroutine or function block in PLC programming, and when would you use one?

A. To reuse code and organize logic; call a common function from multiple places.

B. To reduce memory usage by compressing code.

C. To disable the PLC.

D. To generate random numbers.

Subroutines and function blocks are modular units in PLC programming that let you organize logic and reuse code. The main idea is to design a piece of logic once and call it from different parts of your program, feeding it inputs and getting outputs back. This reduces duplication, makes the program easier to read, and simplifies maintenance because a single change inside the block updates all places that use it. Use them whenever a task appears in multiple places or deserves its own dedicated scope—such as a common motor-start sequence, a calculation routine, or a safety check. They aren't about shrinking memory, disabling the PLC, or generating random numbers.

SAMPLE

10. Which statement best contrasts relay and solid state switching?

A. Relay contacts and solid state switches.

B. Diodes and transistors.

C. Resistors and capacitors.

D. Open-circuit switches only.

The main idea is how the switch is implemented: a relay uses physical contacts that open and close to connect or disconnect a circuit, while solid-state switching uses semiconductor devices with no moving parts to perform the switching. This direct contrast between "relay contacts" and "solid state switches" captures the fundamental difference between mechanical and electronic switching. The other options don't illustrate this distinction: diodes and transistors are examples of components used in solid-state switching rather than describing the contrast with a relay; resistors and capacitors are passive components not themselves the switching action; open-circuit switches only describes a state, not how the switch is implemented.

SAMPLE

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

https://pmmiplc1.examzify.com

We wish you the very best on your exam journey. You've got this!

SAMPLE