ADM Unit 1 | Mission Brief 1.1

Optimize the Pick-and-Place Workcell

Reduce robot cycle time while maintaining 100% successful part transfers.

Customer	Department	Difficulty	Time
Lockwood Aerospace Manufacturing	Automated Assembly	Intermediate	1 class period

Production Goal: Increase output without purchasing a second robot.

Background

A small aerospace supplier uses a 6-axis robot to move components from an incoming fixture to a pallet. The program works, but the robot travels through several unnecessary positions and pauses longer than needed. Management wants a software-only improvement before purchasing additional equipment.

Learning Objectives

- Analyze robot motion and identify non-value-added travel
- Measure and calculate average cycle time
- Revise movement commands while preserving safety
- Use test evidence to justify an improvement

Equipment

- VEX CTE Workcell
- 6-Axis Arm with magnet end effector
- Cubes or disks
- Stopwatch or elapsed-time display
- Python project
- Engineering notebook

Student Instructions

1. Load the current pick-and-place program and verify that it completes one full cycle correctly.
2. Run five cycles and record each cycle time. Calculate the average.
3. Sketch the robot path and label every pickup, clearance, travel, and drop-off point.
4. Identify at least three moves, waits, or height changes that may be reduced or combined.
5. Create a copy of the program and make one change at a time.
6. After each change, run five cycles and record accuracy and average time.
7. Select the fastest version that completes ten consecutive transfers without an error.
8. Calculate the percent improvement and summarize the changes that mattered most.

Engineering Requirements

- Reduce average cycle time by at least 15%
- Complete ten consecutive transfers with no drops or collisions
- Begin and end at the approved Home position
- Keep all movement within the approved work envelope

Constraints & Safety

- Do not change hardware locations
- Robot speed may not exceed 100%
- The magnet must remain the end effector
- A safety stop must remain available

Deliverables

- Original and revised Python programs
- Robot-path sketch with identified waste
- Five-trial before/after timing table
- Percent-improvement calculation
- Notebook reflection

Reflection Questions

1. Which change produced the greatest time savings?
2. What tradeoff did you observe between speed and reliability?
3. Which motion would you redesign if the workcell layout could change?
4. How would one second saved per cycle affect a factory running 8 hours per day?

Engineering Support

Engineering Tip
Optimize path length and unnecessary waits before increasing motor speed.

Common Mistake
Removing every clearance point may create a collision even when the start and end coordinates are correct.

Stretch Goal
Achieve a 25% improvement and verify the program for 25 consecutive parts.

Career Connection
Manufacturing and automation engineers optimize robot motion because small cycle-time savings compound across thousands of parts.

Before You Submit

- Notebook evidence included
- Requirements checked
- Testing documented
- Reflection completed