PCB Component Placement Assignment – Documentation

1. Assignment Overview

The goal of this assignment is to implement a **2D rectangular packing algorithm** to solve a PCB component placement problem. The algorithm must position rectangular components on a 50×50 unit PCB **algorithmically**, satisfying **all hard constraints** while optimizing **soft constraints** like compactness and centrality.

Key Components:

Component	Size (units)	Placement Constraint
USB Connector	5 × 5	Must touch a board edge
Microcontroller	5 × 5	Can be placed anywhere
Crystal	5 × 5	Must be within 10 units of Microcontroller
MikroBus Connector 1	5 × 15	Must touch a board edge
MikroBus Connector 2	5 × 15	Must touch the opposite edge, parallel to MB1

Hard Constraints:

- Edge placement of USB and MikroBus connectors
- Parallel orientation for MikroBus connectors
- Crystal within 10-unit radius of Microcontroller
- No overlapping of components
- Components completely within board boundaries
- Global center-of-mass within 2-unit radius of board center
- Crystal-to-Microcontroller path cannot intersect USB keep-out zone (10×15 units)

Soft Constraints (Optimization Goals):

- Minimize wasted space
- Maximize compactness of layout
- Prefer center placement for unconstrained components

2. Algorithmic Approach

Step 1 – Pre-placement of Edge-Constrained Components

- USB connector and MikroBus connectors are deterministically placed along edges.
- Ensures edge and parallel constraints are automatically satisfied.

Step 2 – Algorithmic Search for Free Components

- Microcontroller positions are explored around the center of the board.
- Crystal positions are explored within the **10-unit proximity radius** around the Microcontroller.
- Candidate placements are aligned to a **1-unit grid** for consistency.

Step 3 – Constraint Validation

The validate_placement function ensures each candidate satisfies:

1. Edge constraints
2. Parallel alignment of MikroBus connectors
3. Proximity of Crystal to Microcontroller
4. No overlaps
5. Boundary constraints
6. Global center-of-mass constraints
7. Keep-out zone avoidance

Geometric methods used:

- Rectangle intersection checks (touching counts as overlap)
- Euclidean distance for proximity
- Line-segment intersection with rectangle for keep-out zone
- Center-of-mass calculation for balance

Step 4 – Scoring & Optimization

- **Bounding Box Compactness:** Measures wasted space
- **Centrality Score:** Distance of Microcontroller from board center
- Total score = Bounding box area + 10 × centrality distance (lower is better)

- Best valid placement is selected using the optimizer function under a **2-second time limit**

---

3. Python Implementation

- Fully algorithmic solution with **no hardcoded placements**
- Modular utility functions for geometric operations
- Quick demo solver and optimizer provided

Example Functions:

- validate_placement(placement): Validates all hard constraints
- score_placement(placement): Scores placements for optimization
- plot_placement(placement): Visualizes placements using Matplotlib
- basic_solver_with_output(): Finds the first valid placement quickly
- optimizer(time_budget=2.0): Searches multiple candidate placements and returns the best

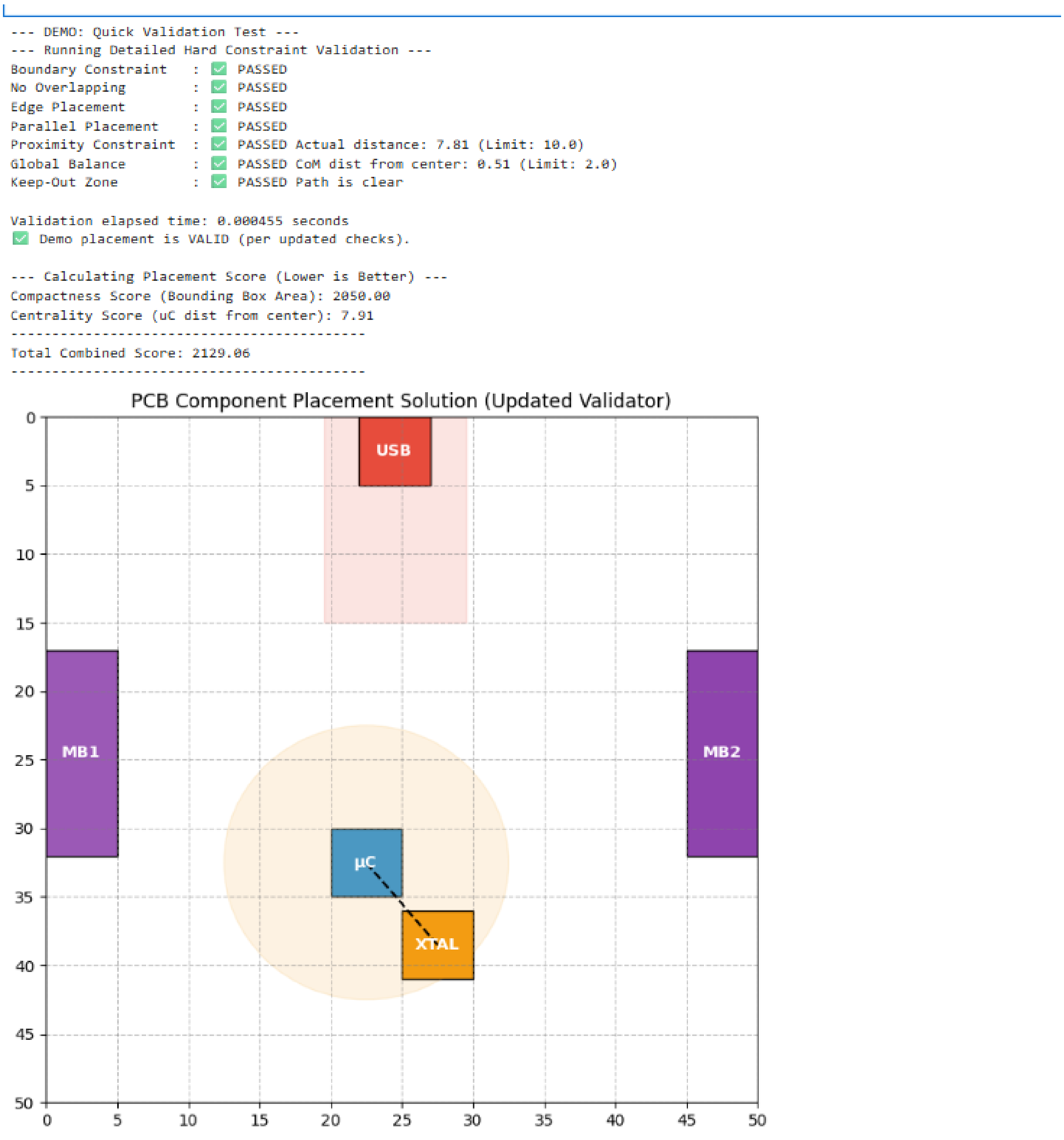
---

4. Visualization

Demo Placement Plot

Features of the plot:

- Rectangular components with color-coded labels:
    - USB (red)
    - Microcontroller (blue)
    - Crystal (orange)
    - MB1 (purple)
    - MB2 (dark purple)
- **Proximity Circle:** Shows 10-unit Crystal radius around Microcontroller

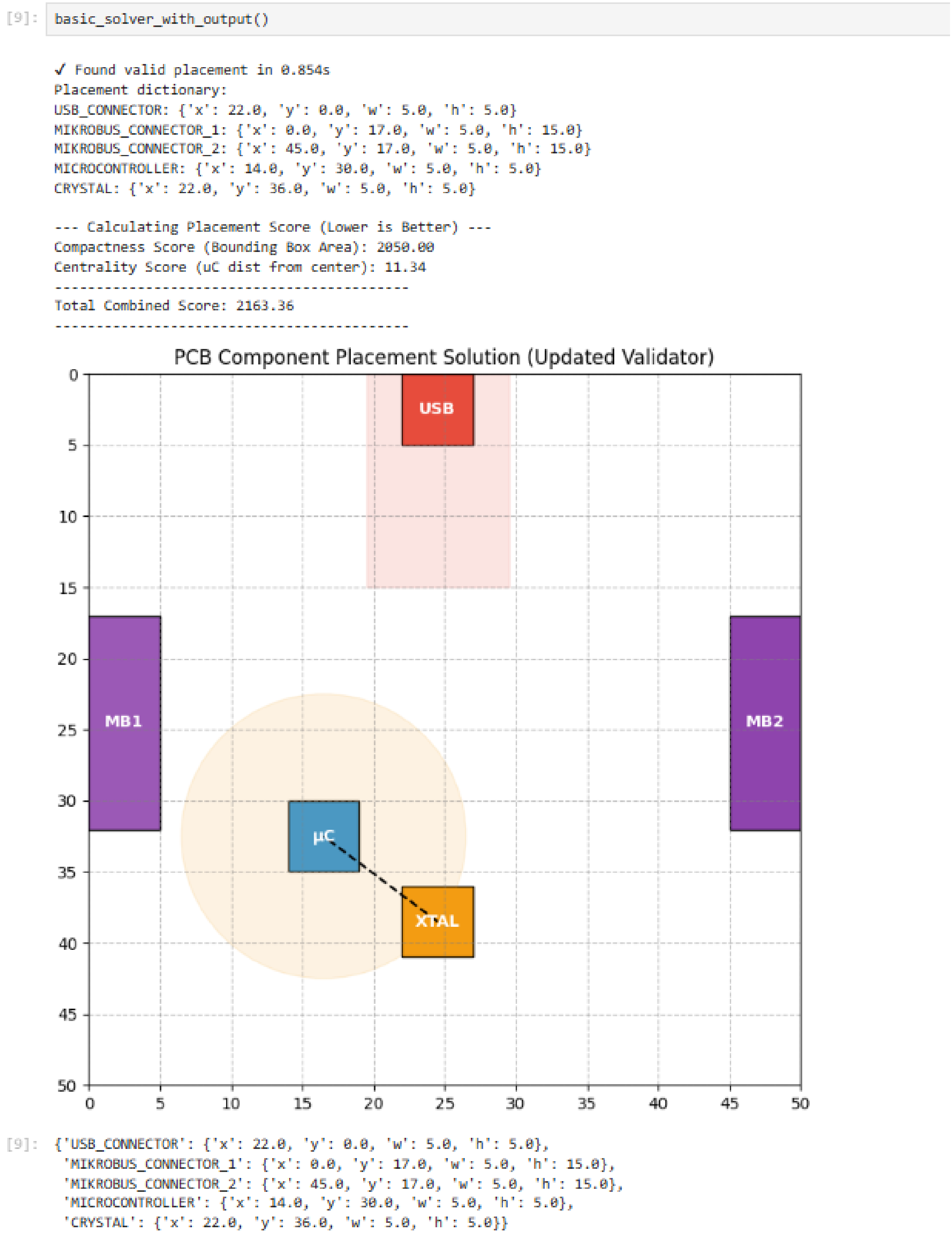


```
--- DEMO: Quick Validation Test ---
--- Running Detailed Hard Constraint Validation ---
Boundary Constraint   : ✅ PASSED
No Overlapping        : ✅ PASSED
Edge Placement        : ✅ PASSED
Parallel Placement    : ✅ PASSED
Proximity Constraint  : ✅ PASSED Actual distance: 7.81 (Limit: 10.0)
Global Balance        : ✅ PASSED CoM dist from center: 0.51 (Limit: 2.0)
Keep-Out Zone         : ✅ PASSED Path is clear

Validation elapsed time: 0.000455 seconds
✅ Demo placement is VALID (per updated checks).

--- Calculating Placement Score (Lower is Better) ---
Compactness Score (Bounding Box Area): 2050.00
Centrality Score (uC dist from center): 7.91
-------------------------------------------
Total Combined Score: 2129.06
-------------------------------------------
```

- **Keep-Out Zone:** Semi-transparent red rectangle extending 15 units from USB center
- **Connection Line:** Dashed line representing the clock signal from Crystal to Microcontroller
- Board grid with 1-unit spacing and proper aspect ratio

This visual confirms that all hard constraints are satisfied.

Demo / Basic Solver Placement Plot



```
[9]: basic_solver_with_output()
```

```
✓ Found valid placement in 0.854s
Placement dictionary:
USB_CONNECTOR: {'x': 22.0, 'y': 0.0, 'w': 5.0, 'h': 5.0}
MIKROBUS_CONNECTOR_1: {'x': 0.0, 'y': 17.0, 'w': 5.0, 'h': 15.0}
MIKROBUS_CONNECTOR_2: {'x': 45.0, 'y': 17.0, 'w': 5.0, 'h': 15.0}
MICROCONTROLLER: {'x': 14.0, 'y': 30.0, 'w': 5.0, 'h': 5.0}
CRYSTAL: {'x': 22.0, 'y': 36.0, 'w': 5.0, 'h': 5.0}

--- Calculating Placement Score (Lower is Better) ---
Compactness Score (Bounding Box Area): 2050.00
Centrality Score (uC dist from center): 11.34
-------------------------------------------
Total Combined Score: 2163.36
-------------------------------------------
```

```
[9]: {'USB_CONNECTOR': {'x': 22.0, 'y': 0.0, 'w': 5.0, 'h': 5.0},
      'MIKROBUS_CONNECTOR_1': {'x': 0.0, 'y': 17.0, 'w': 5.0, 'h': 15.0},
      'MIKROBUS_CONNECTOR_2': {'x': 45.0, 'y': 17.0, 'w': 5.0, 'h': 15.0},
      'MICROCONTROLLER': {'x': 14.0, 'y': 30.0, 'w': 5.0, 'h': 5.0},
      'CRYSTAL': {'x': 22.0, 'y': 36.0, 'w': 5.0, 'h': 5.0}}
```

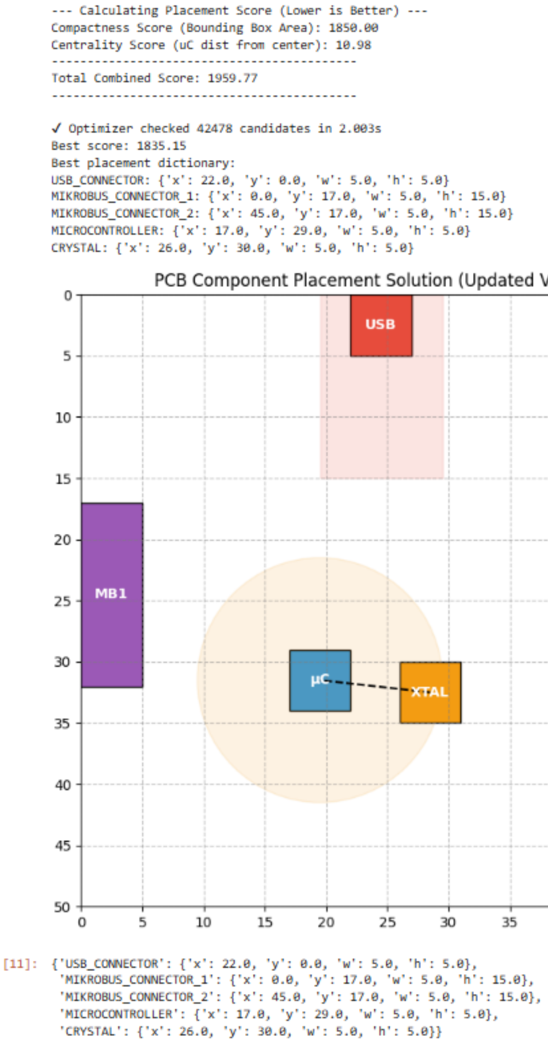
Features of the plot:

- Rectangular components with color-coded labels:
  - USB (red)
  - Microcontroller (blue)

    - Crystal (orange)
    - MB1 (purple)
    - MB2 (dark purple)
- **Proximity Circle:** Shows 10-unit Crystal radius around Microcontroller
- **Keep-Out Zone:** Semi-transparent red rectangle extending 15 units from USB center
- **Connection Line:** Dashed line representing the clock signal from Crystal to Microcontroller
- Board grid with 1-unit spacing and proper aspect ratio
- Confirms **all hard constraints are satisfied**

Optimizer Placement Plot

- Shows the **best-scoring placement** according to compactness and centrality
- Demonstrates that **soft constraints are also considered**



```
--- Calculating Placement Score (Lower is Better) ---
Compactness Score (Bounding Box Area): 1850.00
Centrality Score (uC dist from center): 10.98
---------------------------------------------
Total Combined Score: 1959.77
---------------------------------------------

✓ Optimizer checked 42478 candidates in 2.003s
Best score: 1835.15
Best placement dictionary:
USB_CONNECTOR: {'x': 22.0, 'y': 0.0, 'w': 5.0, 'h': 5.0}
MIKROBUS_CONNECTOR_1: {'x': 0.0, 'y': 17.0, 'w': 5.0, 'h': 15.0}
MIKROBUS_CONNECTOR_2: {'x': 45.0, 'y': 17.0, 'w': 5.0, 'h': 15.0}
MICROCONTROLLER: {'x': 17.0, 'y': 29.0, 'w': 5.0, 'h': 5.0}
CRYSTAL: {'x': 26.0, 'y': 30.0, 'w': 5.0, 'h': 5.0}
```

```
[11]: {'USB_CONNECTOR': {'x': 22.0, 'y': 0.0, 'w': 5.0, 'h': 5.0},
       'MIKROBUS_CONNECTOR_1': {'x': 0.0, 'y': 17.0, 'w': 5.0, 'h': 15.0},
       'MIKROBUS_CONNECTOR_2': {'x': 45.0, 'y': 17.0, 'w': 5.0, 'h': 15.0},
       'MICROCONTROLLER': {'x': 17.0, 'y': 29.0, 'w': 5.0, 'h': 5.0},
       'CRYSTAL': {'x': 26.0, 'y': 30.0, 'w': 5.0, 'h': 5.0}}
```

5. Key Highlights & Improvements

- Deterministic edge placement + algorithmic search ensures **validity and efficiency**
- Robust geometric checks for overlaps, proximity, and keep-out zones
- Optimizer balances **hard and soft constraints**
- Performance: Completes in **< 2 seconds** for the 50×50 board
- Grid-aligned integer coordinates ensure reproducibility

6. Additional Notes

- **Orientation:** Components can rotate 0° or 90° if needed
- **Extensibility:** Code can be extended to larger boards or additional components easily
- **Code Quality:** Functions are modular, documented, and reusable
- **Visualization:** Color-coded components and clear proximity/keep-out zones enhance clarity