✅ GOAL

Implement controlled unload and automatic preload logic into the GUI-based Bambu G-code post-processor.

🛠️ REQUIRED CHANGES

1. Dynamic ComboBox Items (GUI Input Section)

- Replace:

python

CopyEdit

```
type_cb.addItems(["feedrate", "flowrate", "temperature", "unload right", "unload left"])
```

- With dynamically generated items like:

python

CopyEdit

```
["feedrate", "flowrate", "temperature", "Unload S0A", ..., "Unload S7A"]
```

(Set during open_file after parsing M620 SxxA slots.)

---

2. Tracking of M620 → Slot → Tool

- In open_file():
  - Parse every M620 SxxA and map it to T0 or T1 via user input.
  - Save this in self.tool_slot_map.

---

3. Find Insert Locations (for Unload)

- For each "Unload SxA" entry in the GUI:
  - Find the Z1 height.
  - Go **upward** from that height until the last M620 **of the opposite tool**.
  - Insert unload macro:

gcode

CopyEdit

```
; --- UNLOAD SxA ---

M620 SxA

M1002 gcode_claim_action : 4

M204 S9000
```

---

4. Automatic Preload Insertion

- When an unload is inserted:
  - Automatically search **downward** from the Z1 layer.
  - Find the next M620 SxA for the **same slot**.
  - Insert preload macro *just before it*:

gcode

CopyEdit

```
; --- PRELOAD SxA ---

Tn   ; n = T0 or T1 from slot mapping

M104 S140
```

---

5. Preview for Every Insertion

- After inserting code:
  - Show **~30 lines of G-code** centered around insertion point in a message box.
  - Highlight inserted lines (e.g. prefixed with >>>).

---

6. Validation and Messaging

- If no matching M620 for **unload** is found above Z1:
  - ❌ Show error: *"Cannot insert unload before next toolchange for opposite nozzle."*
- If no matching M620 for **preload** is found below Z1:
  - ❌ Show warning: *"No place to insert preload for slot SxA."*

7. (Optional but Good): Label Z List Items

- Show slot used in the Z-height list on the right:
    - Example: Z=10.3 mm: T1 (S3A)