Label Printing Reference:

Media Feed and Positioning

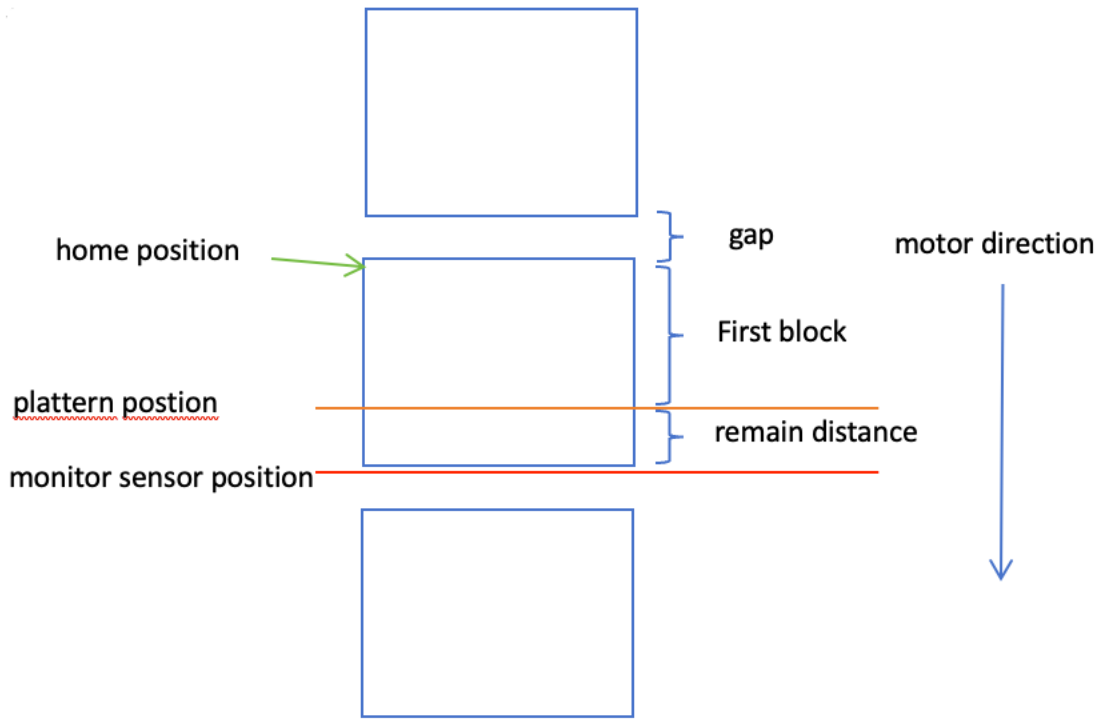
1. Overview

This section defines the hardware behavior and corresponding command protocol for managing printhead positioning and label media advancement. The system requires specific user commands to compensate for the physical offset between the printhead and the gap sensor.

1.1 Key Concepts

Term	Definition
Platen / Printhead	The physical print mechanism. Its position (yellow line) defines the actual print start point.
Monitor Sensor	Optical sensor that detects gaps between labels (red line).
Sensor Offset	Fixed physical distance between the Platen and Monitor Sensor positions.
Home Position	Zero reference point where the platen initiates printing.
First Block	Media section from Home Position to the next detectable label gap.
Remaining Distance	Additional media advance required after print stop to complete platen movement (handled automatically).
Gap-to-Home Distance	Media advance required to position next label at Home Position (requires manual command).

1.2 Physical Layout



Media Path: [HOME] → [PRINT AREA] → [PLATEN (Yellow)] → [SENSOR OFFSET] → [MONITOR SENSOR (Red)]

Critical: The platen and gap sensor are not vertically aligned. All positioning logic must account for this fixed offset.

---

2. Command Reference

2.1 Initialization & Test Print

Command: 0x31 0x0A
Description: Initiates a test print from the current platen position.
Pre-condition: Platen must be at **Home Position**.
Post-condition: Prints character "1" starting from Home Position.
Usage:

```
c
// Example: Request test print
uint8_t test_cmd[] = {0x31, 0x0A};
uart_send(test_cmd, sizeof(test_cmd));
```

2.2 Automatic Short Advance

Command: 1D 06
Description: Automatically advances media to the next Home Position.
Condition: Only valid when **print length < First Block distance**.
Behavior: Full automatic cycle including Remaining Distance and Gap-to-Home advance.
Usage:

```
c
// Example: Advance after short print
if (print_length < first_block_distance) {
   uint8_t auto_advance[] = {0x1D, 0x06};
   uart_send(auto_advance, sizeof(auto_advance));
}
```

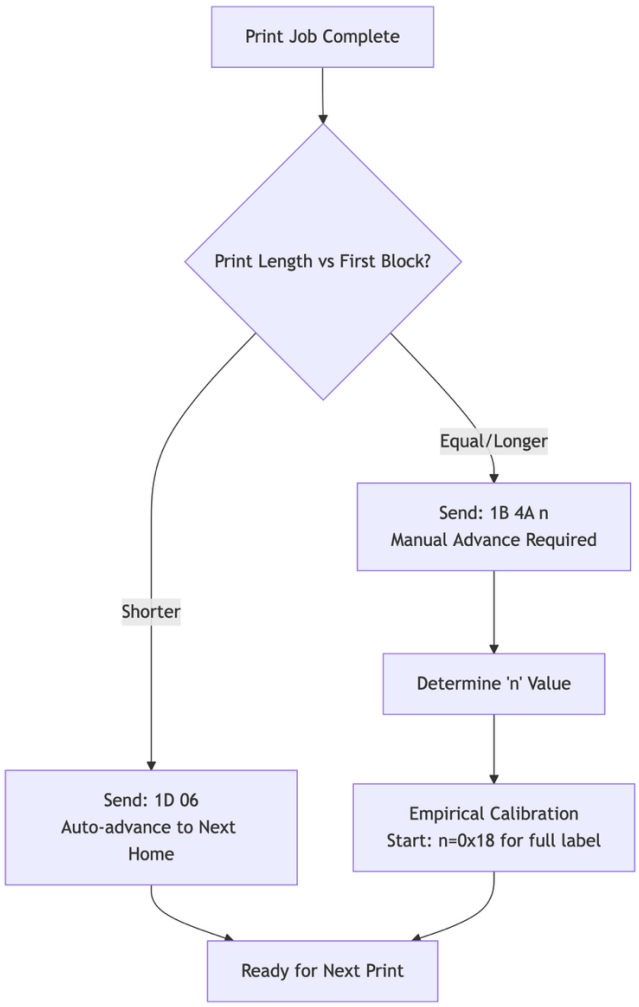
2.3 Manual Long Advance

Command: 1B 4A n
Description: Manually advances platen to next Home Position after long prints.
Condition: Required when **print length ≥ First Block distance**.
Parameter: n = Number of dot lines to advance (8-bit value).
Calibration: Empirical determination required. Reference: n ≈ 0x18 for full-label-length prints.
Usage:

```
c
// Example: Manual advance after long print
if (print_length >= first_block_distance) {
   uint8_t manual_advance[] = {0x1B, 0x4A, calibration_value};  // calibration_value = 0x18 for full label
   uart_send(manual_advance, sizeof(manual_advance));
}
```

---

3. Operational Flowchart



---

4. Error Scenarios & Troubleshooting

Scenario	Symptom	Root Cause	Solution
Incomplete Advance	Platen stops before next label start	Using `1D 06` for long prints	Switch to `1B 4A n` with calibrated `n`
Over-advance	Platen passes next Home Position	`n` value too large	Reduce `n` incrementally (try 0x10, 0x14, etc.)
Under-advance	Platen stops short of Home Position	`n` value too small	Increase `n` incrementally (try 0x1C, 0x20, etc.)
No Advance	Command ignored	Printer not in valid state	Ensure print job completed and sensor not faulted

---

5. Calibration Procedure

5.1 Determining `n` for `1B 4A n`

1. Print content equal to one full label length
2. Send `1B 4A 0x18` as initial value
3. Check if platen reaches next Home Position:
    - **Too short**: Increase `n` by 2-4 units
    - **Too far**: Decrease `n` by 2-4 units
4. Repeat until position is correct
5. Store calibrated value in non-volatile memory

5.2 First Block Distance Calculation

First Block Distance = (Sensor Position - Home Position) + Label Gap Length

Note: This value is hardware-dependent and should be provided in the printer specifications.

---

6. Code Integration Example

c

```c
/**
 * @brief Advances media to next label position
 * @param print_length Length of just-completed print job in dots
 * @param first_block Pre-calculated First Block distance for this printer
 * @param calibrated_n Empirically determined advance parameter
 */
void advance_to_next_label(uint32_t print_length, uint32_t first_block, uint8_t calibrated_n) {
    if (print_length < first_block) {
        // Short print: use automatic advance
        uint8_t cmd_auto[] = {0x1D, 0x06};
        uart_send(cmd_auto, sizeof(cmd_auto));
        log("Auto-advance command sent");
    } else {
        // Long print: manual advance required
        uint8_t cmd_manual[] = {0x1B, 0x4A, calibrated_n};
        uart_send(cmd_manual, sizeof(cmd_manual));
        log("Manual advance command sent with n=0x%02X", calibrated_n);
    }

    // Wait for operation to complete
    while (!printer_ready()) {
        delay_ms(10);
    }
}
```