AI Agent Prompt: ZMK Dual Split Keyboard Dongle Implementation

Project Overview

You are tasked with implementing a ZMK firmware configuration that enables two different 42-key split keyboards (Crosses-42 and Corne Wireless) to connect to a single USB dongle. The Crosses-42 will have two trackballs (one for XY cursor movement, one for scrolling), while the Corne will have no trackballs.

System Architecture

Hardware Components

Central Device (Dongle):

- 1× Nice!Nano v2 (nRF52840)
- Connected to host PC via USB-C
- Acts as BLE central for all peripherals
- No physical keyboard matrix

Peripheral Devices:

- 2× Nice!Nano v2 for Crosses-42 (left half + right half)
    - Left half: Contains XY movement trackball
    - Right half: Contains scroll-only trackball
- 2× Nice!Nano v2 for Corne Wireless (left half + right half)
    - No trackballs on either half

Total BLE Connections: 4 peripherals to 1 central (dongle)

Firmware Repository Structure

You will be working with three existing ZMK configuration repositories:

1. **Crosses-42 Template:** https://github.com/Good-Great-Grand-Wonderful/crosses-42-zmk-template
2. **Corne Wireless Config:** https://github.com/typeractivexyz/corne-wireless-zmk-config
3. **New Dongle Firmware:** To be created in this implementation

Implementation Requirements

Phase 1: Dongle Firmware Creation

Create a new ZMK shield configuration for the dual-keyboard dongle with the following specifications:

File: config/boards/shields/dual_dongle/Kconfig.shield

```
config SHIELD_DUAL_DONGLE
def_bool $(shields_list_contains,dual_dongle)
```

File: config/boards/shields/dual_dongle/Kconfig.defconfig

```
if SHIELD_DUAL_DONGLE

config ZMK_KEYBOARD_NAME
default "DualKB Dongle"

config ZMK_SPLIT_ROLE_CENTRAL
default y

config ZMK_SPLIT
default y
```

Support 4 peripherals (Crosses-42 L+R, Corne L+R)

```
config ZMK_SPLIT_BLE_CENTRAL_PERIPHERALS
default 4
```

4 peripherals + 5 BT profiles = 9 connections

```
config BT_MAX_CONN
default 9

config ZMK_SPLIT_BLE_CENTRAL_BATTERY_LEVEL_FETCHING
default y

config ZMK_SPLIT_BLE_CENTRAL_BATTERY_LEVEL_PROXY
default y

endif
```

File: config/boards/shields/dual_dongle/dual_dongle.overlay

```
/ {
chosen {
zmk,kscan = &kscan0;
};
```

```
kscan0: kscan {
    compatible = "zmk,kscan-gpio-matrix";
    label = "KSCAN";

    // Dongle has no physical keys
    diode-direction = "col2row";
};
```

```
};
```

File: config/dual_dongle.conf

USB dongle mode

```
CONFIG_ZMK_USB=y
```

BLE central mode only

```
CONFIG_BT_PERIPHERAL=n
CONFIG_BT_CENTRAL=y
```

Support 4 peripherals

```
CONFIG_ZMK_SPLIT=y
CONFIG_ZMK_SPLIT_ROLE_CENTRAL=y
CONFIG_ZMK_SPLIT_BLE_CENTRAL_PERIPHERALS=4
CONFIG_BT_MAX_CONN=9
```

Queue sizes optimized for XY + scroll trackballs

```
CONFIG_ZMK_SPLIT_BLE_CENTRAL_POSITION_QUEUE_SIZE=10
CONFIG_ZMK_SPLIT_BLE_CENTRAL_SPLIT_RUN_QUEUE_SIZE=10
```

Input thread stack (moderate increase for dual trackballs)

CONFIG_INPUT_THREAD_STACK_SIZE=768

Pointing device support

CONFIG_ZMK_POINTING=y
CONFIG_ZMK_POINTING_SMOOTH_SCROLLING=y

Increase BT power for better range

CONFIG_BT_CTLR_TX_PWR_PLUS_8=y

Battery monitoring

CONFIG_ZMK_SPLIT_BLE_CENTRAL_BATTERY_LEVEL_FETCHING=y
CONFIG_ZMK_SPLIT_BLE_CENTRAL_BATTERY_LEVEL_PROXY=y

Disable advertising (dongle should not appear as keyboard)

CONFIG_BT_PERIPHERAL=n

File: config/dual_dongle.keymap

Create a 42-key keymap that matches the logical layout of both keyboards:

#include <behaviors.dtsi>
#include <dt-bindings/zmk/keys.h>
#include <dt-bindings/zmk/bt.h>

/ {
keymap {
compatible = "zmk,keymap";

```
    default_layer {
        bindings = <
            // Row 1 (12 keys)
            &kp TAB   &kp Q &kp W &kp E &kp R &kp T
            &kp Y &kp U &kp I &kp O &kp P &kp BSPC

            // Row 2 (12 keys)
            &kp LCTRL &kp A &kp S &kp D &kp F &kp G
            &kp H &kp J &kp K &kp L &kp SEMI &kp SQT

            // Row 3 (12 keys)
            &kp LSHFT &kp Z &kp X &kp C &kp V &kp B
            &kp N &kp M &kp COMMA &kp DOT &kp FSLH &kp ESC

            // Thumb cluster (6 keys)
            &kp LGUI &mo 1 &kp SPACE
            &kp RET &mo 2 &kp RALT
        >;
    };

    lower_layer {
        bindings = <
            &kp GRAVE &kp N1 &kp N2 &kp N3 &kp N4 &kp N5
            &kp N6 &kp N7 &kp N8 &kp N9 &kp N0 &kp DEL

            &bt BT_CLR &bt BT_SEL 0 &bt BT_SEL 1 &bt BT_SEL 2 &bt BT_SEL 3 &bt B
            &kp LEFT &kp DOWN &kp UP &kp RIGHT &trans &trans

            &trans &trans &trans &trans &trans &trans
            &trans &trans &trans &trans &trans &trans

            &trans &trans &trans
            &trans &trans &trans
        >;
    };

    raise_layer {
        bindings = <
            &kp ESC &kp EXCL &kp AT &kp HASH &kp DLLR &kp PRCNT
            &kp CARET &kp AMPS &kp STAR &kp LPAR &kp RPAR &kp BSPC

            &trans &trans &trans &trans &trans &trans
            &kp MINUS &kp EQUAL &kp LBKT &kp RBKT &kp BSLH &kp GRAVE

            &trans &trans &trans &trans &trans &trans
```

```
            &trans &trans &trans &trans &trans &kp TILDE

            &trans &trans &trans
            &trans &trans &trans
         >;
      };
   };
```

};

Phase 2: Crosses-42 Peripheral Configuration

Modify the existing Crosses-42 repository to support:

1. Peripheral-only mode (not central)
2. Trackball integration on both halves
3. XY movement on left half, scroll-only on right half

File: `config/crosses_42_left.conf`

Add or modify:

Split peripheral mode

```
CONFIG_ZMK_SPLIT=y
CONFIG_ZMK_SPLIT_ROLE_CENTRAL=n
CONFIG_ZMK_BLE=y
CONFIG_BT_PERIPHERAL=y
```

Pointing device support (XY trackball)

```
CONFIG_ZMK_POINTING=y
CONFIG_ZMK_SPLIT_PERIPHERAL_HAS_POINTING_DEVICE=y
```

File: `config/crosses_42_right.conf`

Add or modify:

Split peripheral mode

```
CONFIG_ZMK_SPLIT=y
CONFIG_ZMK_SPLIT_ROLE_CENTRAL=n
CONFIG_ZMK_BLE=y
CONFIG_BT_PERIPHERAL=y
```

Pointing device support (scroll trackball)

```
CONFIG_ZMK_POINTING=y
CONFIG_ZMK_SPLIT_PERIPHERAL_HAS_POINTING_DEVICE=y
```

Trackball Hardware Integration

For Crosses-42 Left Half (XY Movement):

Add to config/boards/shields/crosses_42_left/crosses_42_left.overlay:

```
&spi3 {
status = "okay";
cs-gpios = <&gpio0 XX GPIO_ACTIVE_LOW>; // Replace XX with actual pin
```

```
paw3204_left: paw3204@0 {
    compatible = "pixart,paw3204";
    reg = <0>;
    spi-max-frequency = <2000000>;
    irq-gpios = <&gpio0 YY GPIO_ACTIVE_LOW>;  // Replace YY with actual pin
};
```

```
};

/ {
chosen {
zmk,pointing = &paw3204_left;
};
```

```
paw3204_left_listener {
    compatible = "zmk,input-listener";
    device = <&paw3204_left>;

    // XY movement trackball - full XY output
    xy-mode;
};

input_split_listener {
    compatible = "zmk,input-split";
};
```

```
};
```

For Crosses-42 Right Half (Scroll Only):

Add to config/boards/shields/crosses_42_right/crosses_42_right.overlay:

```
&spi3 {
status = "okay";
cs-gpios = <&gpio0 ZZ GPIO_ACTIVE_LOW>; // Replace ZZ with actual pin
```

```
paw3204_right: paw3204@0 {
    compatible = "pixart,paw3204";
    reg = <0>;
    spi-max-frequency = <2000000>;
    irq-gpios = <&gpio0 WW GPIO_ACTIVE_LOW>;  // Replace WW with actual pin
};
```

```
};
```

```
/ {
chosen {
zmk,pointing = &paw3204_right;
};
```

```
paw3204_right_listener {
    compatible = "zmk,input-listener";
    device = <&paw3204_right>;

    // Scroll-only trackball - suppress XY, output scroll only
    scroll-mode;
};

input_split_listener {
    compatible = "zmk,input-split";
};
```

```
};
```

Crosses-42 Keymap Synchronization

Copy the exact keymap from config/dual_dongle.keymap to config/crosses_42.keymap to ensure logical key positions match.

Phase 3: Corne Wireless Peripheral Configuration

Modify the existing Corne Wireless repository to support peripheral-only mode.

File: config/corne_left.conf

Add or modify:

Split peripheral mode

CONFIG_ZMK_SPLIT=y
CONFIG_ZMK_SPLIT_ROLE_CENTRAL=n
CONFIG_ZMK_BLE=y
CONFIG_BT_PERIPHERAL=y

No pointing device

File: config/corne_right.conf

Add or modify:

Split peripheral mode

CONFIG_ZMK_SPLIT=y
CONFIG_ZMK_SPLIT_ROLE_CENTRAL=n
CONFIG_ZMK_BLE=y
CONFIG_BT_PERIPHERAL=y

No pointing device

Corne Keymap Synchronization

Copy the exact keymap from config/dual_dongle.keymap to config/corne.keymap to ensure logical key positions match across all keyboards.

Phase 4: Build Configuration

File: build.yaml (in dongle repository)

---

include:

- board: nice_nano_v2
  shield: dual_dongle
- board: nice_nano_v2
  shield: crosses_42_left
- board: nice_nano_v2
  shield: crosses_42_right
- board: nice_nano_v2
  shield: corne_left

- board: nice_nano_v2
  shield: corne_right

This will generate 5 firmware files:

1. `dual_dongle-nice_nano_v2-zmk.uf2`
2. `crosses_42_left-nice_nano_v2-zmk.uf2`
3. `crosses_42_right-nice_nano_v2-zmk.uf2`
4. `corne_left-nice_nano_v2-zmk.uf2`
5. `corne_right-nice_nano_v2-zmk.uf2`

Phase 5: Pairing and Testing Procedure

Initial Setup

1. **Clear all existing bonds:**
    - On each Nice!Nano controller, reset Bluetooth bonds
    - Use reset button combination or flash with `&bt BT_CLR` behavior
2. **Flash firmware:**
    - Flash dongle Nice!Nano with `dual_dongle-nice_nano_v2-zmk.uf2`
    - Flash Crosses-42 left Nice!Nano with `crosses_42_left-nice_nano_v2-zmk.uf2`
    - Flash Crosses-42 right Nice!Nano with `crosses_42_right-nice_nano_v2-zmk.uf2`
    - Flash Corne left Nice!Nano with `corne_left-nice_nano_v2-zmk.uf2`
    - Flash Corne right Nice!Nano with `corne_right-nice_nano_v2-zmk.uf2`
3. **Power on sequence:**
    - Plug dongle into PC via USB (should appear as HID keyboard)
    - Power on Crosses-42 left half (LED should indicate pairing)
    - Power on Crosses-42 right half
    - Power on Corne left half
    - Power on Corne right half
    - All peripherals should auto-pair to the dongle

Verification Tests

Test 1: Basic Key Input

- Type on Crosses-42 → verify characters appear on host
- Type on Corne → verify characters appear on host
- Switch between keyboards freely

Test 2: XY Trackball Movement (Crosses-42 Left)

- Move trackball on Crosses-42 left half
- Verify mouse cursor moves on screen
- Check movement is smooth and responsive

Test 3: Scroll Trackball (Crosses-42 Right)

- Move trackball on Crosses-42 right half
- Verify page scrolls vertically
- Verify cursor does NOT move (scroll-only mode)

Test 4: Layer Switching

- Press layer 1 key on either keyboard
- Verify layer-specific keys work (numbers, Bluetooth controls)
- Test Bluetooth profile switching

Test 5: Battery Reporting

- Check host OS battery indicator
- Should show battery levels for all 4 peripherals

Test 6: Connection Stability

- Leave keyboards idle for 5 minutes
- Wake up and type → should respond immediately
- No reconnection lag

Test 7: Simultaneous Input

- Use Crosses-42 XY trackball while typing on Corne
- Verify no input conflicts or dropped events

Troubleshooting Guide

Problem: Peripherals won't pair to dongle

- Solution: Clear bonds on all devices, flash again, ensure dongle is powered via USB

Problem: Choppy trackball movement

- Solution: Increase `CONFIG_ZMK_SPLIT_BLE_CENTRAL_POSITION_QUEUE_SIZE` to 15-20

Problem: Scroll trackball also moves cursor

- Solution: Verify scroll-mode configuration in right half overlay, check keymap behavior

Problem: Keys on one keyboard don't match expected output

- Solution: Ensure keymaps are identical across dongle, Crosses-42, and Corne configs

Problem: Battery levels not showing

- Solution: Enable `CONFIG_ZMK_SPLIT_BLE_CENTRAL_BATTERY_LEVEL_FETCHING=y` in dongle config

Problem: Connection drops after idle

- Solution: Increase BLE power with `CONFIG_BT_CTLR_TX_PWR_PLUS_8=y`

Technical Constraints and Considerations

Bluetooth Connection Limits

- **Maximum peripherals:** 4 (Crosses-42 L+R, Corne L+R)
- **Maximum total connections:** 9 (4 peripherals + 5 BT host profiles)
- **Do not exceed** these limits or connection stability will degrade

Trackball Event Load

With XY + Scroll configuration:

- XY trackball generates ~100-150 events/sec
- Scroll trackball generates ~50 events/sec
- Total: ~150-200 events/sec
- Default queue size (10) is sufficient for this load
- Only increase to 20 if experiencing dropped events

Keymap Synchronization Requirements

Critical: All three keymaps (dongle, Crosses-42, Corne) must have:

- Identical logical key positions
- Same layer structure
- Same layer numbers
- Matching key codes for each position

Why: The dongle processes all key events. If keymaps differ, switching between keyboards will cause unexpected output.

Power Consumption

Dongle: USB-powered, no battery concerns

Peripherals:

- Each trackball adds ~5-10mA continuous draw
- Expect ~50% battery life reduction on Crosses-42 vs Corne
- Use higher capacity batteries (e.g., 500mAh+) for Crosses-42

Physical Pin Assignments

Important: Replace placeholder pins (XX, YY, ZZ, WW) in overlay files with actual GPIO pins from your hardware design:

- Trackball SPI pins (MOSI, MISO, SCK, CS)
- Trackball IRQ pins
- Matrix row/column pins (already defined in existing configs)

Refer to:

- Nice!Nano v2 pinout diagram
- PAW3204/PMW3610 sensor datasheets
- Existing Crosses-42 and Corne schematics

ZMK Version Requirements

This implementation requires:

- **ZMK main branch** (pointing device support for split peripherals merged in PR #2477)
- **Zephyr 3.2+** (required by ZMK main)

Use recent ZMK builds; older versions lack split peripheral pointing device relay.

Deliverables Checklist

Repository Structure

- [ ] Dongle firmware repository created with shield configuration
- [ ] Crosses-42 repository modified for peripheral mode + dual trackballs
- [ ] Corne repository modified for peripheral mode
- [ ] All repositories use synchronized keymaps

Firmware Artifacts

- [ ] `dual_dongle-nice_nano_v2-zmk.uf2` builds successfully
- [ ] `crosses_42_left-nice_nano_v2-zmk.uf2` builds successfully
- [ ] `crosses_42_right-nice_nano_v2-zmk.uf2` builds successfully
- [ ] `corne_left-nice_nano_v2-zmk.uf2` builds successfully
- [ ] `corne_right-nice_nano_v2-zmk.uf2` builds successfully

Configuration Files

- [ ] All `.conf` files include required options
- [ ] All `.overlay` files have correct device tree syntax
- [ ] All keymaps are synchronized across builds
- [ ] Pin assignments match hardware design

Testing

- [ ] All 7 verification tests pass
- [ ] No Bluetooth connection issues
- [ ] Trackball XY movement smooth and accurate
- [ ] Trackball scroll works without cursor movement
- [ ] Battery reporting functional
- [ ] Layer switching works on both keyboards

Documentation

- [ ] README with setup instructions
- [ ] Pin assignment documentation
- [ ] Troubleshooting guide
- [ ] Pairing procedure documented

Success Criteria

The implementation is successful when:

1. **Dongle connects to all 4 peripherals** simultaneously over BLE
2. **Both keyboards are fully functional** for typing
3. **Crosses-42 left trackball** provides XY cursor movement
4. **Crosses-42 right trackball** provides scroll-only (no XY)
5. **Keymap is consistent** across both keyboards
6. **No input conflicts** when using trackball + typing simultaneously
7. **Battery levels** reported correctly for all peripherals
8. **Connection is stable** with no dropout or lag after idle
9. **User can switch** between Crosses-42 and Corne seamlessly

Additional Resources

- ZMK Documentation: https://zmk.dev/docs
- ZMK Split Keyboards Guide: https://zmk.dev/docs/features/split-keyboards
- ZMK Dongle Integration: https://zmk.dev/docs/development/hardware-integration/dongle
- ZMK Pointing Devices: https://zmk.dev/docs/development/hardware-integration/pointing
- Nice!Nano Documentation: https://nicekeyboards.com/docs/nice-nano/
- PAW3204 Sensor Datasheet: (obtain from PixArt)

Notes for AI Agent

- **Validate all syntax** before generating final configs
- **Check pin assignments** against Nice!Nano v2 pinout
- **Test incremental changes** (dongle first, then peripherals)
- **Monitor event queue depth** during testing (may need adjustment)
- **Document any deviations** from this specification
- **Preserve existing keymap preferences** where possible
- **Ask for clarification** on hardware-specific details (pins, sensors, etc.)

This specification provides complete implementation guidance. Proceed systematically through each phase, validating functionality before moving to the next. The configuration is designed for optimal performance with dual trackballs while maintaining clean separation between XY movement and scrolling.