Keypad Module Non-blocking Driver

1 Description

The keypad driver depends on a switch-case state to determine which button is pressed.

2 Keypad Module APIs

2.1 Keypad Initialization

```
/**
* @brief Initializes the direction of the row keypad pins as output pins,
and the columns keypad pins as input pins.
*/
void KEYPAD_init(void);
```
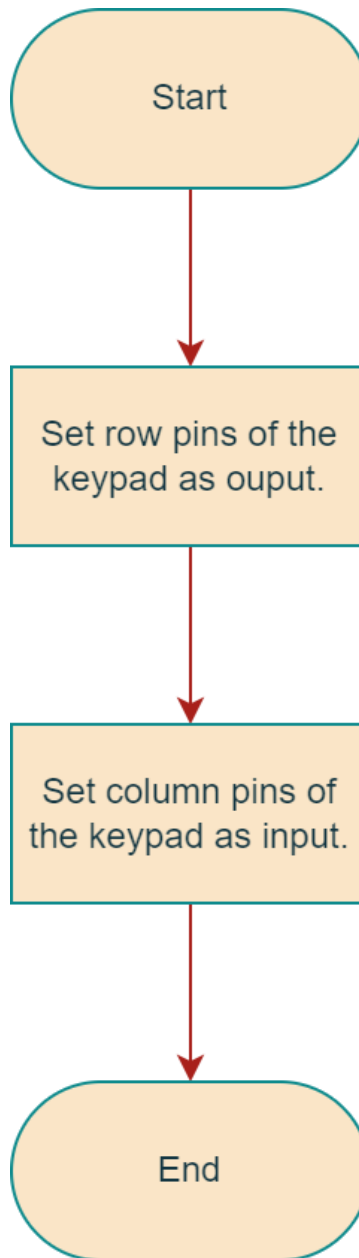
2.2 Keypad Get Key Pressed

```
/**
* @brief Returns the number of position of the key pressed on the keypad
matrix.
* @return An unsigned integer value that represents the position of the
key pressed.
*/
uint8 KEYPAD_getKeyPressed(void);
```

2.3 Keypad Get Mapped Key

```
/**
* @brief Maps the position of the key with the actual value of the keypad.
* @param current_key_pressed An unsigned integer value that represents the
position of the key pressed.
* @return An unsigned integer of the actual reading of the keypad.
*/
uint8 KEYPAD_mappedKey(uint8 current_key_pressed);
```
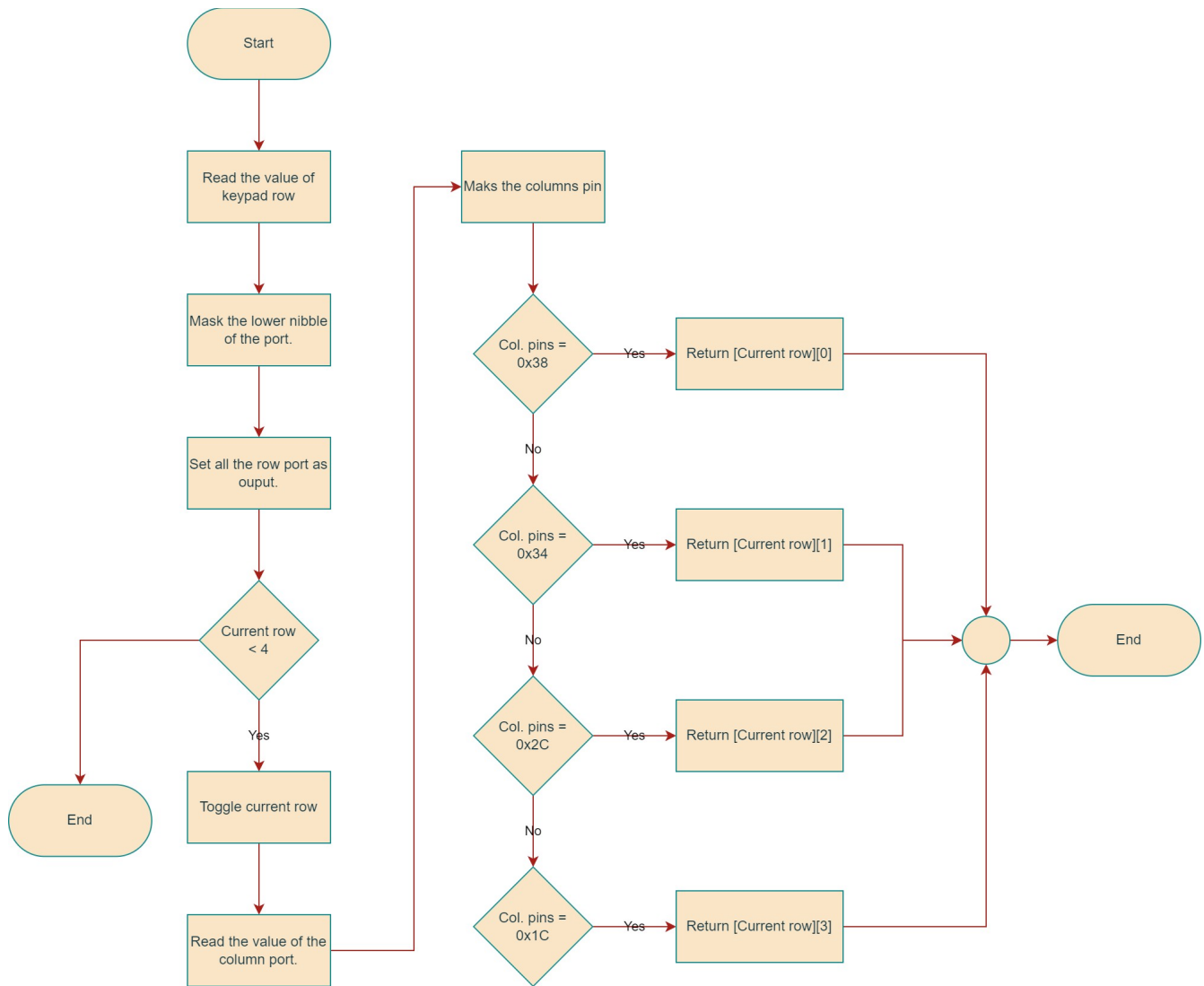
3 Keypad Module Low Level Design Flowchart

3.1 Keypad Initialization Flowchart

3.2 Keypad Get Key Pressed Flowchart

Start

Read the value of keypad row

Mask the lower nibble of the port.

Set all the row port as ouput.

Current row < 4

End

Yes

Toggle current row

Read the value of the column port.

Maks the columns pin

Col. pins = 0x38 → Yes → Return [Current row][0]

No

Col. pins = 0x34 → Yes → Return [Current row][1]

No

Col. pins = 0x2C → Yes → Return [Current row][2]

No

Col. pins = 0x1C → Yes → Return [Current row][3]

End

3.3 Keypad Get Key pressed Flowchart

```
          ┌─────────────────┐
          │      Start       │
          └─────────────────┘
                    │
                    ▼
          ┌─────────────────┐
          │ Return [Current Key │
          │   Pressed - 1]   │
          └─────────────────┘
                    │
                    ▼
          ┌─────────────────┐
          │       End        │
          └─────────────────┘
```