ExoNaut Robot Library

This guide shows every function available in the ExoNaut library.

Space Trek ExoNaut Programming Reference
Created by Ryan Bori

Last Updated: May 29, 2025

Contents

1 Basic Setup 4

2 ExoNaut (Main Robot Control) 4
2.1 Motor Functions 4
2.2 Encoder Functions 4
2.3 LED Functions 4

3 ExoNaut_AICam (AI Camera) 4
3.1 Setup Functions 4
3.2 Face Detection Functions 4
3.3 Object Detection Functions 5
3.4 Color Detection Functions 5
3.5 Line Detection Functions 5
3.6 QR Code Functions 5
3.7 Barcode Functions 5
3.8 AprilTag Functions 5
3.9 Classification Functions 6
3.10 Feature Learning Functions 6
3.11 Number Recognition Functions 6
3.12 Landmark Recognition Functions 6

4 ExoNaut_AICamLF (AI Camera Line Following) 6
4.1 Setup Functions 6
4.2 Detection Functions 6
4.3 Movement Functions 6

5 ExoNaut_Ultrasonic (Distance Sensor) 7
5.1 Measurement Functions 7
5.2 LED Functions 7

6 ExoNaut_TempHumid (Temperature & Humidity) 7
6.1 Setup Functions 7
6.2 Reading Functions 7

7 ExoNaut_MP3 (Music Player) 7
7.1 Setup Functions 7
7.2 Playback Functions 7
7.3 Volume Functions 7

8 ExoNaut_7Segment (Number Display) 8
8.1 Setup Functions 8
8.2 Display Functions 8
8.3 Animation Functions 8

9 ExoNaut_DotMatrix (Dot Display) 8
9.1 Setup Functions 8
9.2 Display Functions 8
9.3 Text Functions 8

10 ExoNaut_Knob (Dial Control) 9
10.1 Setup Functions 9

10.2 Reading Functions 9

11 ExoNaut_IMU (Motion Sensor) 9

11.1 Setup Functions 9

11.2 Raw Values - Exact Numbers 9

11.3 Easy Questions - True/False 9

11.4 Simple Words - Descriptions 10

12 ExoNaut_LineFollower (Simple Line Sensor) 10

12.1 Reading Functions 10

13 ExoNaut_RGB_LED (External LED Strip) 10

13.1 Setup Functions 10

13.2 LED Functions 10

14 Quick Reference 10

14.1 Camera Application Modes 10

14.2 Motor IDs 11

14.3 Port Compatibility 11

14.4 Port Pin Mappings 11

1 Basic Setup

Every ExoNaut program starts with this basic setup:

```
#include "ExoNaut.h"
exonaut robot;

void setup() {
    robot.begin();  // Always start with this!
}
```

2 ExoNaut (Main Robot Control)

2.1 Motor Functions

```
robot.begin()                                    // Initialize the robot
robot.set_motor_speed(m1, m2)                   // Set motor speeds (-100 to 100)
robot.encoder_motor_set_speed(motorid, speed)   // Set individual motor speed
robot.encoder_motor_get_speed(items[])          // Get current motor speeds
robot.stop_motor(motorid)                       // Stop motor (0=both, 1=left, 2=right)
robot.encoder_motor_turn(speed, angle)          // Turn robot by degrees
```

2.2 Encoder Functions

```
robot.reset_encoder_counter(motorid)            // Reset wheel counters
robot.get_encoder_count(items[])                // Get wheel rotation counts
robot.set_motor_type(motortype)                 // Set motor type (1 or 2)
```

2.3 LED Functions

```
robot.setColor(n, r, g, b)                      // Set individual LED color (0-5)
robot.setColorAll(r, g, b)                      // Set all LEDs same color
robot.show()                                    // Update LED display
robot.clear()                                   // Turn off all LEDs
```

3 ExoNaut_AICam (AI Camera)

3.1 Setup Functions

```
camera.begin()                                  // Initialize camera
camera.changeFunc(new_func)                     // Change camera mode
camera.updateResult()                           // Get new camera data
camera.setLed(state)                           // Turn camera LED on/off
camera.currentFunc()                           // Get current camera mode
```

3.2 Face Detection Functions

```
camera.anyFaceDetected()                        // Check if any face found
camera.numOfTotalFaceDetected()                 // Count total faces
camera.anyLearnedFaceDetected()                 // Check for known faces
camera.numOfTotalLearnedFaceDetected()          // Count known faces
camera.anyUnlearnedFaceDetected()               // Check for unknown faces
camera.numOfTotalUnlearnedFaceDetected()        // Count unknown faces
camera.faceOfIdDetected(id)                 // Check for specific face ID
camera.getFaceOfId(id, result)              // Get face data by ID
camera.getFaceOfIndex(index, result)        // Get face data by position
camera.printFaceTable()                     // Print all face info
```

3.3 Object Detection Functions

```
camera.anyObjDetected()                     // Check if any object found
camera.numOfObjDetected()                   // Count objects detected
camera.objIdDetected(id)                    // Check for specific object
camera.numOfObjIdDetected(id)               // Count specific objects
camera.objDetected(id, index, result)       // Get object data
```

3.4 Color Detection Functions

```
camera.anyColorDetected()                   // Check if any color found
camera.numOfColorDetected()                 // Count colors detected
camera.colorIdDetected(id)                  // Check for specific color
camera.colorId(id, result)                  // Get color data
```

3.5 Line Detection Functions

```
camera.anyLineDetected()                    // Check if line found
camera.numOfLineDetected()                  // Count lines detected
camera.lineIdDetected(id)                   // Check for specific line
camera.lineId(id, result)                   // Get line data
```

3.6 QR Code Functions

```
camera.qrCodeDetected()                     // Check if QR code found
camera.qrCodeDataLength()                   // Get QR code data length
camera.qrCodeData(buffer)                   // Get QR code text
```

3.7 Barcode Functions

```
camera.barCodeDetected()                    // Check if barcode found
camera.barCodeDataLength()                  // Get barcode data length
camera.barCodeData(buffer)                  // Get barcode text
```

3.8 AprilTag Functions

```
camera.anyTagDetected()                     // Check if AprilTag found
camera.numOfTotalTagDetected()              // Count AprilTags
camera.tagIdDetected(id)                    // Check for specific tag
camera.numOfTagIdDetected(id)               // Count specific tags
camera.tagId(id, index, result)             // Get tag data
camera.printAllTagDetails()                 // Print all tag info
camera.getTagInfo(tagId, result)            // Get specific tag info
camera.estimateTagDistance(tag, realSize)   // Calculate distance to tag
camera.getTagOrientation(tag)               // Get tag rotation
camera.listDetectedTagIds()                 // List all tag IDs
```

3.9 Classification Functions

```
camera.classIdOfMaxProb()                       // Get most likely class ID
camera.classMaxProb()                           // Get highest probability
camera.classProbOfId(id)                        // Get probability of class
```

3.10 Feature Learning Functions

```
camera.featureIdOfMaxProb()                     // Get most likely feature ID
camera.featureMaxProb()                         // Get highest probability
camera.featureProbOfId(id)                      // Get probability of feature
```

3.11 Number Recognition Functions

```
camera.numberWithMaxProb()                      // Get most likely number
camera.numberMaxProb()                          // Get highest probability
camera.numberProbOfId(id)                       // Get probability of number
```

3.12 Landmark Recognition Functions

```
camera.anyLandmarkDetected()                    // Check if landmark found
camera.numOfLandmarksDetected()                 // Count landmarks
camera.landmarkIdDetected(id)                   // Check for specific landmark
camera.numOfLandmarkIdDetected(id)              // Count specific landmarks
camera.getLandmarkById(id, result)              // Get landmark data
camera.landmarkIdWithMaxProb()                  // Get most likely landmark
camera.landmarkMaxProb()                        // Get highest probability
camera.landmarkProbOfId(id)                     // Get probability of landmark
```

4 ExoNaut_AICamLF (AI Camera Line Following)

4.1 Setup Functions

```
lineFollower.begin(robot, camera)               // Initialize line follower
lineFollower.update()                           // Update line detection
lineFollower.setBaseSpeed(speed)                // Set driving speed
```

4.2 Detection Functions

```
lineFollower.getLineStatus(lineId)              // Get line status
lineFollower.getLineAngle(lineId)               // Get line angle
lineFollower.getLineOffset(lineId)              // Get line offset from center
lineFollower.getLineData(lineId, data)          // Get complete line data
lineFollower.isLineDetected(lineId)             // Check if line detected
lineFollower.getLineCount()                     // Count lines detected
```

4.3 Movement Functions

```
lineFollower.followLine(lineId, speed, turnFactor)  // Manual line following
lineFollower.simpleFollowLine()                 // Automatic line following
```

5 ExoNaut_Ultrasonic (Distance Sensor)

5.1 Measurement Functions

```
sensor.getDistance()                             // Get distance in millimeters
```

5.2 LED Functions

```
sensor.color(r1, g1, b1, r2, g2, b2)           // Set LED colors (0-255)
sensor.breathing(r1, g1, b1, r2, g2, b2)       // Set breathing effect
sensor.getColor()                               // Get current LED colors
sensor.getBrightness()                          // Get LED brightness
sensor.getLEDMode()                             // Get LED mode
```

6 ExoNaut_TempHumid (Temperature & Humidity)

6.1 Setup Functions

```
weather.begin()                                  // Initialize sensor
weather.isConnected()                           // Check if sensor connected
```

6.2 Reading Functions

```
weather.readSensor()                            // Take new measurement
weather.temperature                             // Temperature in Fahrenheit
weather.temperatureC                            // Temperature in Celsius
weather.humidity                                // Humidity percentage
weather.getStatus()                             // Get sensor status
weather.debugSensor()                           // Print debug information
```

7 ExoNaut_MP3 (Music Player)

7.1 Setup Functions

```
music.begin()                                   // Initialize MP3 player
```

7.2 Playback Functions

```
music.play()                                    // Play current song
music.pause()                                   // Pause playback
music.next()                                    // Next song
music.previous()                                // Previous song
```

7.3 Volume Functions

```
music.volumeUp()                                // Increase volume
music.volumeDown()                              // Decrease volume
music.maxVolume()                               // Set maximum volume
music.mute()                                    // Mute sound
music.setVolumePercent(percent)                 // Set volume 0-100\%
```

8 ExoNaut_7Segment (Number Display)

8.1 Setup Functions

```
display.begin()                                // Initialize display
display.setBrightness(level)                   // Set brightness 0-7
```

8.2 Display Functions

```
display.showNumber(number)                     // Show number 0-9999
display.showText(text)                         // Show text (4 letters max)
display.clear()                                // Clear display
display.showDigit(digit, position)             // Show single digit
display.clearDigit(position)                   // Clear single position
display.showDecimal(number, decimalPlace)      // Show number with decimal
```

8.3 Animation Functions

```
display.countUp(start, end, delayMs)           // Count up animation
display.countDown(start, end, delayMs)         // Count down animation
display.blink(number, times, delayMs)          // Blink number
display.scroll(text, delayMs)                  // Scroll long text
```

9 ExoNaut_DotMatrix (Dot Display)

9.1 Setup Functions

```
matrix.begin()                                 // Initialize display
matrix.setBrightness(level)                    // Set brightness 0-7
```

9.2 Display Functions

```
matrix.clear()                                 // Clear display
matrix.setAllOn()                              // Turn all dots on
matrix.setAllOff()                             // Turn all dots off
matrix.setLED(row, col, state)                 // Set individual dot
matrix.setRow(row, pattern)                    // Set entire row
matrix.setMatrix(data[])                       // Set entire display
```

9.3 Text Functions

```
matrix.displayNumber(number)                   // Show number 0-99
matrix.displayNumberWithEffect(number)         // Show number with animation
matrix.drawChar(character, x, y)               // Draw single character
matrix.scrollText(text, numScrolls, speed)     // Scroll text across
matrix.stopScroll()                            // Stop scrolling
matrix.isScrolling()                           // Check if scrolling
matrix.updateScroll()                          // Update scroll (call in loop)
```

10 ExoNaut_Knob (Dial Control)

10.1 Setup Functions

```
knob.begin(port)                               // Initialize knob on port 1,2,6,8
knob.setCalibration(rawMin, rawMax)            // Set custom range
```

10.2 Reading Functions

```
knob.getPercent()                              // Get position 0-100\%
```

11 ExoNaut_IMU (Motion Sensor)

11.1 Setup Functions

```
motion.start()                                 // Initialize sensor
motion.calibrate()                             // Calibrate (keep still!)
motion.resetDirection()                        // Reset starting direction
```

11.2 Raw Values - Exact Numbers

```
// Angles (in degrees)
motion.getPitchAngle()                         // Tilt forward/back (-180 to +180)
motion.getRollAngle()                          // Tilt left/right (-180 to +180)
motion.getYawAngle()                           // Direction facing (0 to 360)
motion.getTurnAngle()                          // Turn from start (-180 to +180)

// Motion Speed (degrees per second)
motion.getPitchSpeed()                         // How fast tilting forward/back
motion.getRollSpeed()                          // How fast tilting left/right
motion.getTurnSpeed()                          // How fast turning left/right

// Forces (in g-force, 1g = gravity)
motion.getForceX()                             // Side force (left/right)
motion.getForceY()                             // Forward force (forward/back)
motion.getForceZ()                             // Up force (up/down)
```

11.3 Easy Questions - True/False

```
// Nose direction
motion.isLevel()                               // Is robot flat/level?
motion.isNoseUp()                              // Is nose tilted up?
motion.isNoseDown()                            // Is nose tilted down?

// Side position
motion.isFlat()                                // Is robot standing upright?
motion.isOnLeftSide()                          // Is robot on left side?
motion.isOnRightSide()                         // Is robot on right side?

// Direction facing
motion.isFacingStraight()                      // Facing starting direction?
motion.isTurnedLeft()                          // Has robot turned left?
motion.isTurnedRight()                         // Has robot turned right?

// Movement detection
```

```
motion.isShaking()                              // Is robot shaking/vibrating?
motion.isMoving()                               // Is robot currently moving?
motion.isUpsideDown()                           // Is robot upside down?
```

11.4 Simple Words - Descriptions

```
motion.getNoseDirection()                       // Returns "up", "down", "level"
motion.getSidePosition()                        // Returns "flat", "left side", "right side"
motion.getTurnDirection()                       // Returns "straight", "left", "right"
```

12 ExoNaut_LineFollower (Simple Line Sensor)

12.1 Reading Functions

```
lineSensor.readLineFollower(value)              // Read all sensors
lineSensor.CH1                                  // Sensor 1 state (true/false)
lineSensor.CH2                                  // Sensor 2 state
lineSensor.CH3                                  // Sensor 3 state
lineSensor.CH4                                  // Sensor 4 state
```

13 ExoNaut_RGB_LED (External LED Strip)

13.1 Setup Functions

```
RGB myStrip(numLeds, pin, rmtChannel)           // Create LED strip
myStrip.begin()                                 // Initialize strip
myStrip.setBrightness(level)                    // Set brightness 0-255
```

13.2 LED Functions

```
myStrip.setColor(n, r, g, b)                    // Set LED color
myStrip.setColor(n, packedColor)                // Set LED with packed color
myStrip.show()                                  // Update strip display
myStrip.clear()                                 // Turn off all LEDs
myStrip.numPixels()                             // Get number of LEDs
```

14 Quick Reference

14.1 Camera Application Modes

```
APPLICATION_FACEDETECT        // Face detection
APPLICATION_OBJDETECT         // Object detection
APPLICATION_COLORDETECT       // Color detection
APPLICATION_LINEFOLLOW        // Line detection
APPLICATION_QRCODE            // QR code reading
APPLICATION_BARCODE           // Barcode reading
APPLICATION_APRILTAG          // AprilTag detection
APPLICATION_CLASSIFICATION    // Image classification
APPLICATION_FEATURELEARNING   // Feature learning
APPLICATION_NUMBER_REC        // Number recognition
APPLICATION_LANDMARK          // Landmark recognition
```

14.2 Motor IDs

- **0** = Both motors
- **1** = Left motor
- **2** = Right motor

14.3 Port Compatibility

Which modules work on which ports:

- **RGB LED Strip**: Ports 6, 8
- **Temperature & Humidity**: Ports 3, 4, 5, 9
- **7-Segment Display**: Ports 6, 8
- **Dot Matrix Display**: Ports 6, 8
- **IMU (Motion Sensor)**: Ports 3, 4, 5, 9
- **Avoid Obstacle Sensor**: Ports 6, 8
- **Knob (Dial)**: Ports 1, 2, 6, 8
- **Fan**: Ports 6, 8
- **MP3 Player**: Ports 3, 4, 5, 9
- **AI Camera**: Ports 3, 4, 5, 9

14.4 Port Pin Mappings

Pin connections for each port:

- **Port 1**: DIN = Pin 36, CLK = Pin 39
- **Port 2**: DIN = Pin 32, CLK = Pin 35
- **Port 6**: DIN = Pin 33, CLK = Pin 25
- **Port 8**: DIN = Pin 26, CLK = Pin 27