# CLEAR Module
# User Manual

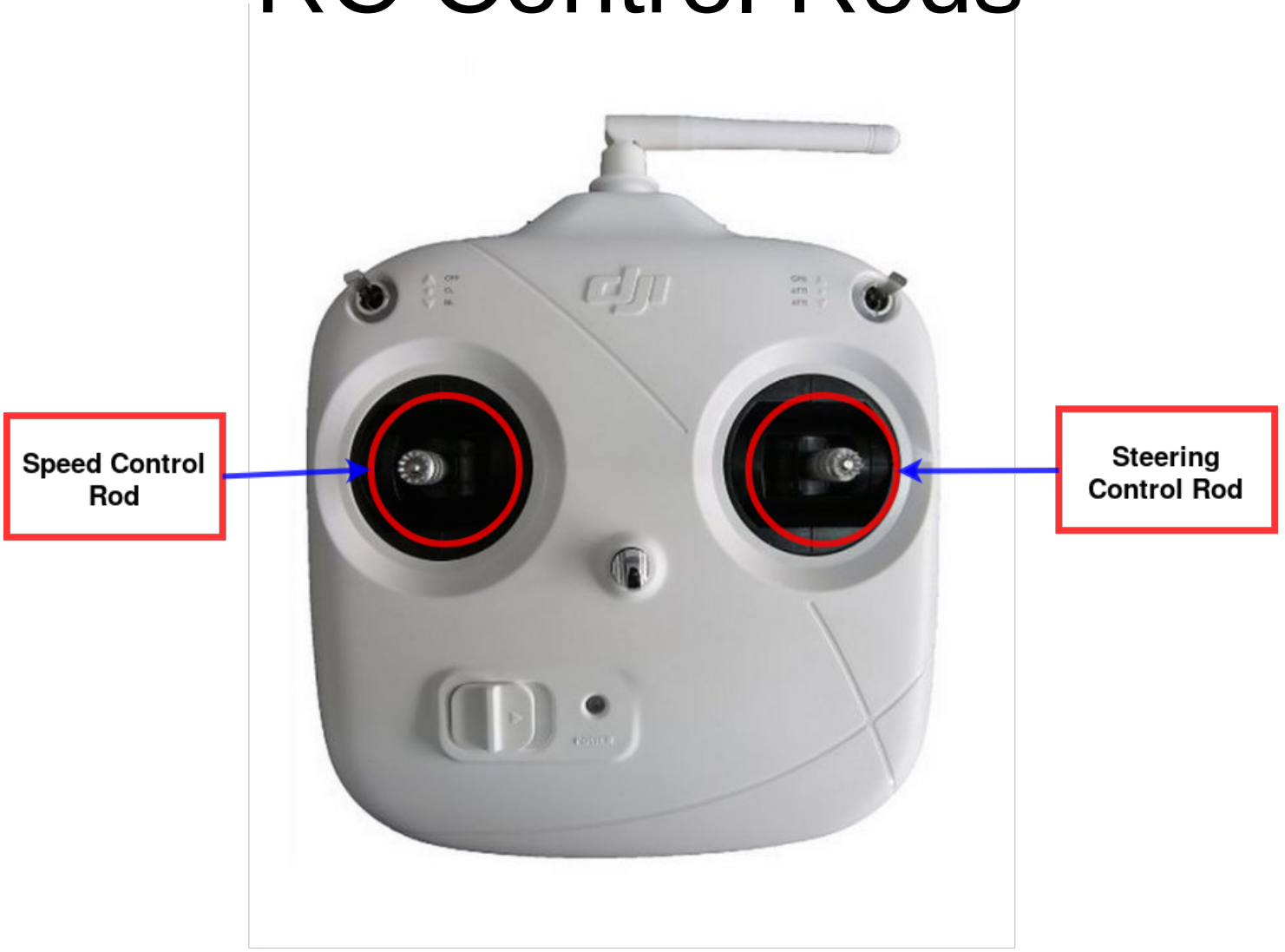
Created: 07/09/2018

Last revision: 11/09/2018

# Remote Control Interface Description

# Remote Control: DJI-DT7

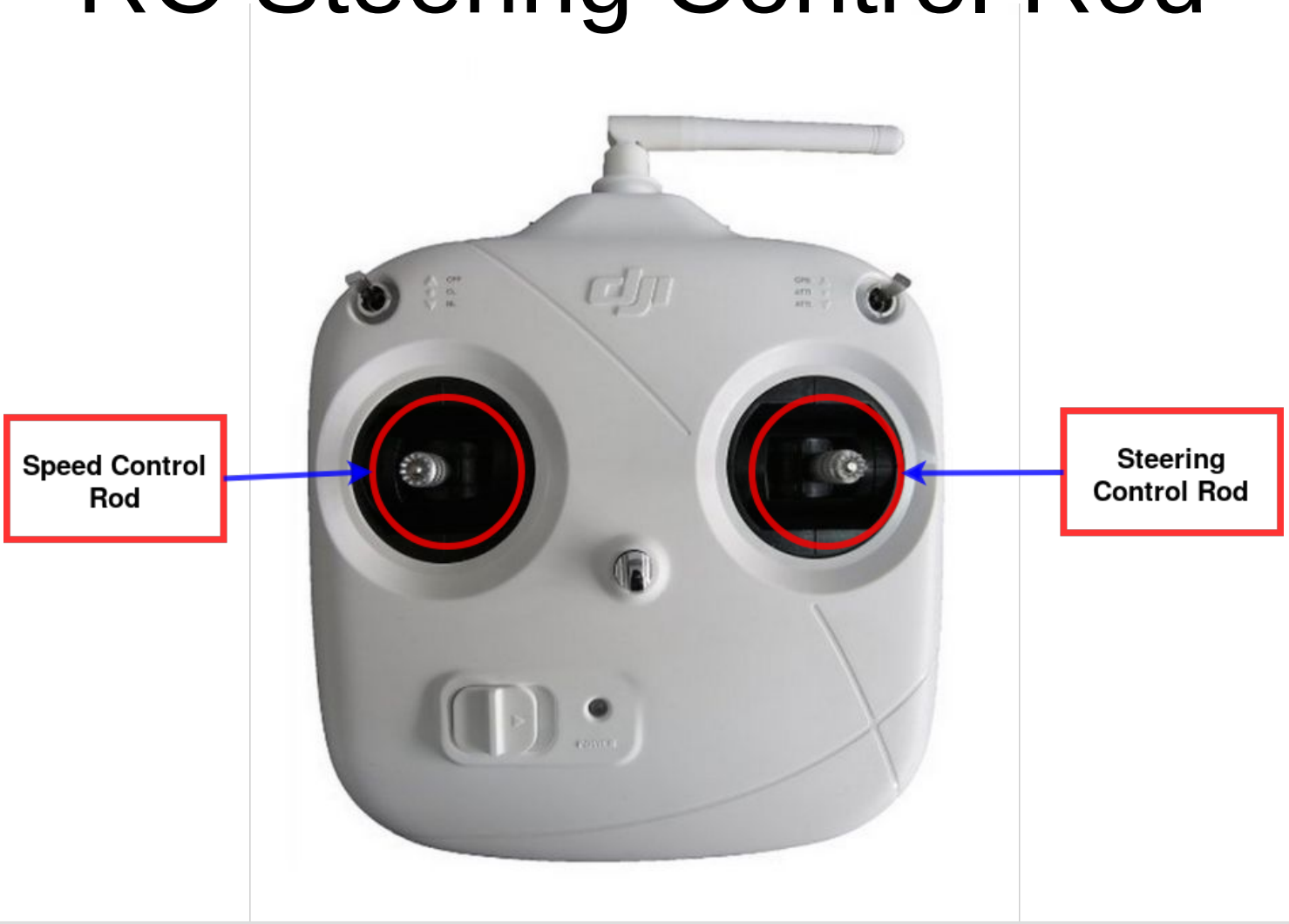
# RC Control Rods



# RC Speed Control Rod


**Speed Control Rod:**
Move it Up and Down to control the robot speed (forward and reverse speed respectively). The position of this control will be used either to adjust the speed PID setpoint or to set the output voltage, bypassing the controller (see Emergency/Disable PID Switch).

# RC Steering Control Rod




**Steering Control Rod:**
Move it to the Left and Right to control the steering angle. If the PID is enabled (see Emergency /Disable PID Switch) the control is performed in angular position, while if the PID is disabled, the control rod changes the PWM value sent to the motor, thus controlling the angular velocity.

# RC Operational Mode Switch

# RC Operational Mode Switch


**Up:** **ROS Control**
**Center:** **Manual Safe**
**Down:** **Manual NOT-SAFE**

# ROS Control Mode (On-board LED = Blue)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- Speed and steering angle values received trough the *desired_ackermann_state* topic will be used to adjust the PID controllers setpoint.

# ROS Control Mode (On-board LED = Blue)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- Speed and steering angle values received trough the *desired_ackermann_state* topic will be used to adjust the PID controllers setpoint.

- The presence of the “speed recommender” safety system is required. This system uses on-board sensors to adjust the robot speed, approaching obstacles slowly and stopping the robot if the distance becomes smaller than a user-configurable safety distance threshold.

# ROS Control Mode (On-board LED = Blue)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- Speed and steering angle values received trough the *desired_ackermann_state* topic will be used to adjust the PID controllers setpoint.

- The presence of the “speed recommender” safety system is required. This system uses on-board sensors to adjust the robot speed, approaching obstacles slowly and stopping the robot if the distance becomes smaller than a user-configurable safety distance threshold.

- Whenever the “speed recommender” overrides the PID speed setpoint, the on-board LED will turn to Yellow. If the “speed recommender” messages are not received within a safety time threshold the system state changes automatically to Emergency mode. (Both conditions also applies to Manual Safe mode).

# Manual Safe Mode
# (On-board LED = Green)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- PID setpoints are controlled using the RC control rods.

# Manual Safe Mode
# (On-board LED = Green)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- PID setpoints are controlled using the RC control rods.

- The presence of the “speed recommender” safety system is required. If it is not detected, the system will change to Emergency mode.

# Manual NOT-Safe Mode (On-board LED = White)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- The safety system is **<u>disabled</u>**, so the robot won't stop to avoid collisions.

# Manual NOT-Safe Mode (On-board LED = White)


**Up: ROS Control**
**Center: Manual Safe**
**Down: Manual NOT-SAFE**

- The safety system is **disabled**, so the robot won't stop to avoid collisions.

- The RC control rods can be used to adjust the PID's setpoints, or altermatively, to actuate directly the motors (speed voltage and steering PWM), bypassing the PIDs. (see Emergency/Disable PID switch).

# RC Emergency / Disable PID Switch

**Up:** **Emergency**
**Center:** **No-Emergency**
**Down:** **Emergency or Disable PID (in Manual NOT-SAFE mode)**

# Emergency Mode (On-board LED = Red)


**Up: Emergency**
**Center: No-Emergency**
**Down: Emergency or Disable PID (in Manual NOT-SAFE mode)**

- In Emergency Mode, the motors are disabled and the brakes are activated immediately.

# Emergency Mode (On-board LED = Red)


**Up: Emergency**
**Center: No-Emergency**
**Down: Emergency or Disable PID (in Manual NOT-SAFE mode)**

- In Emergency Mode, the motors are disabled and the brakes are activated immediately.

- To enter in Emergency Mode from the RC, just change the Emergency Switch to one of the extreme positions (Up or Down) except in Manual NOT-SAFE mode, where Down Position is reserved for PID controllers bypass (On-board LED → Yellow).

# Emergency Mode (On-board LED = Red)


**Up: Emergency**
**Center: No-Emergency**
**Down: Emergency or Disable PID (in Manual NOT-SAFE mode)**

- In Emergency Mode, the motors are disabled and the brakes are activated immediately.

- To enter in Emergency Mode from the RC, just change the Emergency Switch to one of the extreme positions (Up or Down) except in Manual NOT-SAFE mode, where Down Position is reserved for PID controllers bypass (On-board LED → Yellow).

- To exit from Emergency mode: Set the Operational Mode Switch to “Manual NOT-SAFE” mode, set the Emergency/Disable PID switch to “No-Emergency” and rearm the system using the Horn/Rearm wheel (see next slide).

# RC Horn / Rearm Wheel

**Full left: Horn**
**Full right: Rearm**


**<u>Horn / Rearm Wheel:</u>**
Drive the wheel to its left extreme position to activate the horn.
To finalize the rearm process (see Emergency / Disable PID Switch section) drive the wheel to its right extreme position.

# On-board Interface Description

# On-board Emergency Switch


- CLEAR enters in Emergency mode if this switch is activated (LED = Red)

# On-board Emergency Switch


- CLEAR enters in Emergency mode if this switch is activated (LED = Red)
- The Remote Control rearm process will not finish until this switch is turned to “no emergency” state

# On-board Emergency Switch


- CLEAR enters in Emergency mode if this switch is activated (LED = Red)
- The Remote Control rearm process will not finish until this switch is turned to “no emergency” state
- Obviously, the different Emergency Switches (RC and On-board) are independent, so any of them can activate the Emergency Mode, whatever would be the state of the others.

# On-board LED Indicator


- The CLEAR Module, in addition to the CLEAR Status topic, indicates some of its internal information by means of a LED indicator.

# On-board LED Indicator


- **Green means Remote Control Safe Mode.**

# On-board LED Indicator


- **Green means Remote Control Safe Mode.**
- The RC control rods are used to give the setpoint values for the Speed and Steering controllers (m/s and degrees)

# On-board LED Indicator


- **Green means Remote Control Safe Mode.**
- The RC control rods are used to give the setpoint values for the Speed and Steering controllers (m/s and degrees)
- This mode requires the presence of the reactive Safety System (Speed Recommender) that adjust --or stops if needed-- the robot speed to avoid collisions.

# On-board LED Indicator


- **Green means Remote Control Safe Mode.**
- The RC control rods are used to give the setpoint values for the Speed and Steering controllers (m/s and degrees)
- This mode requires the presence of the reactive Safety System (Speed Recommender) that adjust --or stops if needed-- the robot speed to avoid collisions.
- In this mode, if the Safety System messages are not received, CLEAR enters in Emergency Mode.

# On-board LED Indicator


**RED means Emergency Mode, and can be reached in several ways:**

# On-board LED Indicator


**RED means Emergency Mode, and can be reached in several ways:**

- RC communication lost. If the RC signal is lost, CLEAR enters in Emergency mode, this implies that the robot can not be operated without the presence of the RC, even in ROS mode (for safety reasons).

# On-board LED Indicator


**RED means Emergency Mode, and can be reached in several ways:**

- RC communication lost. If the RC signal is lost, CLEAR enters in Emergency mode, this implies that the robot can not be operated without the presence of the RC, even in ROS mode (for safety reasons).
- On-board Emergency Switch.

# On-board LED Indicator


**RED means Emergency Mode, and can be reached in several ways:**

- RC communication lost. If the RC signal is lost, CLEAR enters in Emergency mode, this implies that the robot can not be operated without the presence of the RC, even in ROS mode (for safety reasons).
- On-board Emergency Switch.
- RC Emergency Switch.

# On-board LED Indicator


**RED means Emergency Mode, and can be reached in several ways:**

- RC communication lost. If the RC signal is lost, CLEAR enters in Emergency mode, this implies that the robot can not be operated without the presence of the RC, even in ROS mode (for safety reasons).
- On-board Emergency Switch.
- RC Emergency Switch.
- Reactive Safety System communication lost (all modes except Manual Not-Safe).

# On-board LED Indicator


**RED means Emergency Mode, and can be reached in several ways:**

- RC communication lost. If the RC signal is lost, CLEAR enters in Emergency mode, this implies that the robot can not be operated without the presence of the RC, even in ROS mode (for safety reasons).
- On-board Emergency Switch.
- RC Emergency Switch.
- Reactive Safety System communication lost (all modes except Manual Not-Safe).
- Desired Ackermann State topic not received (only in ROS Control Mode).

# On-board LED Indicator


**Blue means ROS Control Mode.**

# On-board LED Indicator


**Blue means ROS Control Mode.**

- This is the mode used during autonomous navigation.

# On-board LED Indicator


**Blue means ROS Control Mode.**

- This is the mode used during autonomous navigation.
- In this mode CLEAR expects to receive the Speed and Steering controllers setpoints through the Desired Ackermann State topic.

# On-board LED Indicator


**Blue means ROS Control Mode.**

- This is the mode used during autonomous navigation.
- In this mode CLEAR expects to receive the Speed and Steering controllers setpoints through the Desired Ackermann State topic.
- The presence of the reactive Safety System and the Remote Control are also required.

# On-board LED Indicator


**Yellow could have two different meanings:**

# On-board LED Indicator


**Yellow could have two different meanings:**

- In Manual Not-Safe mode, it indicates that the RC control rods are used to drive the motors at the lowest possible level, Voltage for the Speed and PWM for the Steering, so **PIDs are disabled**.

# On-board LED Indicator


**Yellow could have two different meanings:**

- In Manual Not-Safe mode, it indicates that the RC control rods are used to drive the motors at the lowest possible level, Voltage for the Speed and PWM for the Steering, so **PIDs are disabled**.
- In Manual Safe Mode and ROS Control Mode, this color means that the reactive Safety System is currently limiting the speed.

# On-board LED Indicator


**The rest of the colors are currently unused, but this could change in the future! → Please keep updated this document!**

# CLEAR ROS Interface Description

# CLEAR ROS Interface

- The  CLEAR module provides a rich interface with ROS using Rosserial.
- For a detailed description of the input, configuration, debugging and output topics, please refer to the software documentation (the present document aims to be just a user-friendly introduction!)
- In the next slides we will describe the main input and output topics: Desired Ackermann State and CLEAR Status.

# Desired Ackermann State

- This topic represents the main control input of the CLEAR module.
- It is an Ackermann message that contains five state variables: steering angle, steering angle rate, speed, acceleration and jerk.
- Currently only two variables are used: steering angle (degrees) and speed (meters per second).
- This two values are used as setpoints for the steering and speed PID’s controllers.
- This topic controls the robot movement only when the CLEAR module is in ROS Control Operational Mode (on-board LED = blue)
- The reactive Safety System can override these controls in any moment (on-board LED = yellow) for safety reasons.

# CLEAR Status

- This topic shows a code using five integrer numbers.
- These numbers are always integers, however we have choose to use floating point variables, to allow more flexibility during the debugging process: with floating point variables one can easily change one of the status codes for the value of any non-discrete internal variable of interest (maybe some distance, voltage…).
- Reading the topic with rostopic echo we have, from left to right:
  - Current Operational Mode.
  - Error Code.
  - ROS Interface Warnings.
  - Robot Warnings.
  - Verbose Level.

# CLEAR Status

| CODE | Operational Mode | Error | ROS Interface Warnings | Robot Warnings | Verbose Level |
|---|---|---|---|---|---|
| **0** | Emergency | No Error | No Warning | No Warning | No Verbose |
| **1** | RC Safe | RC Lost | ROS controls received in not ROS mode | - | **Regular:** status + state estimation |
| **2** | ROS | On-board Emerg. Switch | - | Safety System Limiting Speed | **Debug:** All info (including echoes) |
| **3** | RC NOT Safe | RC Emerg.Switch | - | - | - |
| **4** | - | Safety System Lost | - | - | - |
| **5** | - | ROS Control Lost | - | - | - |