MagneMotion

A Rockwell Automation Company

User Manual Addendum, Vehicle Platooning

Original Instructions

MagneMotion, Inc.
A Rockwell Automation Company
139 Barnum Road
Devens, MA 01434
USA

Phone: +1 978-757-9100
Fax: +1 978-757-9200

rok.auto/ict

Contents

Figures

Tables

Overview

Vehicle platooning allows moving multiple vehicles simultaneously without the restriction of brick-wall headway between the vehicles by controlling only the lead vehicle. Brick-wall headway is still maintained before the lead vehicle in the platoon. Platooning is useful for stations that have multiple processes, as it allows a set of vehicles to move into, and out of, position in the station simultaneously, which can minimize change over time.

Platoons consist of following vehicles that are coupled to a leading vehicle. Once a platoon of two or more vehicles is created, additional vehicles are added to it by coupling the additional vehicles to the end of the platoon. Or, by coupling the platoon to a new lead vehicle. Once a platoon is created, control of the vehicle at the front of the platoon (the leader) controls the behavior of all vehicles in the platoon (acceleration, velocity, position).

This feature currently supports the following functions:

- Vehicles can be coupled into a platoon only when the platoon is not moving.
- Vehicles can be uncoupled from a platoon either when the platoon is not moving or when the Vehicle Follow Order for a specific vehicle is configured for auto-decouple.
- All vehicles in a platoon must be the same size (that is, all single pucks or all tandem pucks).
- Platooning supports motion on any path and motion across a node.
- Platooning supports coupling vehicles to a platoon that is located across a node.
- Platooning supports uncoupling a vehicle from a platoon that is located across a node.
- The vehicle follow order to the HLC must include acceleration and velocity values.
- Platoons cannot reverse direction. To change the direction of motion, the platoon must stop and be uncoupled. Once uncoupled, the vehicles are recoupled in the opposite direction and the last vehicle in the original platoon becomes the lead vehicle in the new platoon.

This document describes the use of the NCHost utility to configure a vehicle to follow another vehicle on a MagneMover® LITE (MM LITE™) or QuickStick® 100 transport system. This document also describes the communication protocol messages that are used between the Host Controller and the MagneMotion high-level controller (HLC). These messages are used to configure a vehicle to follow another vehicle on the transport system and to monitor vehicles on the system.

This document describes the changes to the NCHost TCP Interface Utility, the Host Controller TCP/IP Communication Protocol, and the Host Controller EtherNet/IP Communication Protocol.

Related Documents

This document references the following manuals:

- *NCHost TCP Interface Utility User Manual*, MMI-UM010.
- *Host Controller TCP/IP Communication Protocol User Manual*, MMI-UM003.
- *Host Controller EtherNet/IP Communication Protocol User Manual*, MMI-UM004.
- *MagneMover LITE Configurator User Manual*, MMI-UM008.
- *QuickStick Configurator User Manual*, MMI-UM009.

NOTE: Distribution of this manual and all addendums and attachments are not controlled. Changes to the document set or the software can be made at any time. To identify the current revisions or to obtain a current version, see Rockwell Automation Support on page 85.

Glossary

Couple: The joining of a vehicle into a platoon, where one vehicle follows another vehicle at a defined distance.

Uncouple: Remove a vehicle from a platoon.

Following Vehicle: A vehicle following another vehicle in a platoon. This vehicle can be following either the lead vehicle or another following vehicle.

Lead Vehicle: The vehicle at the front of a platoon. This vehicle determines the destination, acceleration, and velocity of the platoon.

Vehicle Platoon: A set of vehicles that are moving together and following a lead vehicle. This group of vehicles is allowed to maintain a distance between each other while in motion that is less than the Brick-wall Headway.

Using The Platooning Function

The platooning feature couples a following vehicle to the vehicle in front of it at an ordered position offset to mimic the move profile (acceleration and velocity) of the platoon leader. This feature can be used to create a platoon of N vehicles to move out of a station while a new platoon of N vehicles moves in.

NOTE: Vehicles following another vehicle in a platoon do not support collision avoidance (brick-wall headway) between vehicles in the platoon.

When a platoon is moving through a node, the following vehicles in the platoon continue to follow the lead vehicle. Ownership of the node is only released after the entire platoon passes through the node.

The functioning of Single Vehicle areas, Keepout areas, and Traffic Lights is not guaranteed with vehicles that are in a platoon. Only the lead vehicle obeys these functions.

A vehicle ordered to follow another vehicle with a platooning command will obey the specified platoon following distance, and will NOT use wide vehicle spacing. A platoon spacing distance must be specified that is large enough to avoid collisions along the route that the platoon will travel (for example, through curves).

- When using multiple platoons, the distance from the following platoon to the leading platoon must obey brick-wall headway.
- Platoons can be “coupled” and “uncoupled” to move vehicles into and out of a station without the constraint of brick-wall headway between the vehicles in the platoon.
- All vehicles in a platoon must be the same size and use the same type of magnet array.
- When coupling a vehicle to a platoon, the center-to-center distance between vehicles must keep the vehicles from contacting each other when moving through a curve.
- Attempting to couple across a node that another platoon is crossing is rejected.
- For MagneMover LITE, to reduce the effect of competing forces between adjacent magnet arrays and the coils in the motor, the recommended minimum center-to-center space from leading to following vehicles when creating a platoon is:
    - 100 mm for standard 62 x 62 mm single pucks with a 77 mm configured length.
    - 173 mm for standard 62 x 150 mm tandem pucks with a 165 mm configured length.

  If the vehicle length is longer than the minimal puck length described above, the minimum gap between vehicles must be the value that is returned from the following formula.

  $$MinGap = 23.3 \text{ mm} + VehicleLength$$

  Where:

  MinGap – Minimum gap between vehicles in a platoon (in mm).

  VehicleLength – Length of the vehicle (in mm).

The gap that is calculated must be added to the actual vehicle length in deriving minimal vehicle spacing in the platoon, subject to practical minimal distance recommendations.

If the vehicle overhangs the puck, the vehicle edges can be placed closer than the distance that is defined by the formula. However, if the vehicles are placed too close to each other, tolerances in the vehicle following the profile may cause vehicles to touch.

- For QuickStick 100 and QuickStick Plus, to reduce the effect of competing forces between adjacent magnet arrays and the coils in the motor, the recommended minimum center-to-center space from leading to following vehicles when creating a platoon is:

  $$MinGap = 102 \text{ mm} + VehicleLength + LargestDownstreamGap$$

  Where:

  MinGap – Minimum gap between vehicles in a platoon (in mm).

  VehicleLength – Length of the vehicle (in mm).

  LargestDownstreamGap – Length of the largest downstream gap (in mm).

  The gap that is calculated must be added to the actual vehicle length in deriving minimal vehicle spacing in the platoon, subject to practical minimal distance recommendations.

- To make effective use of platooning, the maximum distance between the vehicles in a platoon should be no greater than:

  $$MaxDistance = MaxCommandedVelocity^2 / CommandedAcceleration * 2$$

  Where:

  MaxDistance – Maximum distance between vehicles in a platoon (in meters).

  MaxCommandedVelocity – Maximum velocity defined in the Node Controller Configuration File (in m/s).

  CommandedAcceleration – Maximum acceleration being used in a motion command for the platoon (in m/s^2).

- When coupling a vehicle to a platoon, the maximum difference between the physical distance and the requested distance (distance in the command) between the vehicles must be within 30 mm

- When coupling a vehicle to a platoon using the vehicle follow order, the vehicle and the platoon must be stopped. Stopped is defined as movement must be below the **Arrival Velocity Tolerance** defined on the **Global Settings** page in the Configurator when creating the Node Controller Configuration File.

- When uncoupling a vehicle from a platoon using a vehicle position order, the platoon must be stopped.

- When uncoupling a vehicle from a platoon using a vehicle follow order with a decouple destination, the vehicle decouples from the platoon when the platoon approaches the decouple destination. The vehicle then stops at the decouple destination when the platoon passes that position.

- The leading vehicle in a platoon must be leading in the direction of travel.
- Each vehicle added to the platoon must be added in the direction of travel of the platoon.
- Deleting the lead vehicle from a platoon causes the platoon to stop due to lack of profile data from the leader. The next vehicle in the platoon becomes the new leader. To continue motion of the platoon, send a vehicle motion order to the new leader.

  If the vehicle behind the lead vehicle was the end of a two-vehicle platoon, the platoon no longer exists.
- Platoons cannot couple or uncouple vehicles while using Sync.
- Sync can only be used on the lead vehicle of a platoon, but the sync profile must continue in the direction of the platoon. Followers in the platoon ignore all sync messages.

Creating and Moving Platoons

Platoons can be created for vehicle motion either downstream or upstream. Once a platoon is created, it is limited to motion in the direction that is specified when it was created. The direction of a platoon can be reversed by decoupling the vehicles in the platoon and creating a platoon moving in the opposite direction (see Change Platoon Direction on page 18).

To couple a vehicle into a platoon, or uncouple a vehicle from a moving platoon, use the vehicle position command and the vehicle follow order command as described in the following examples.

Follow a Vehicle Moving Downstream

Create a two-vehicle platoon that moves downstream by specifying a vehicle for another vehicle to follow.

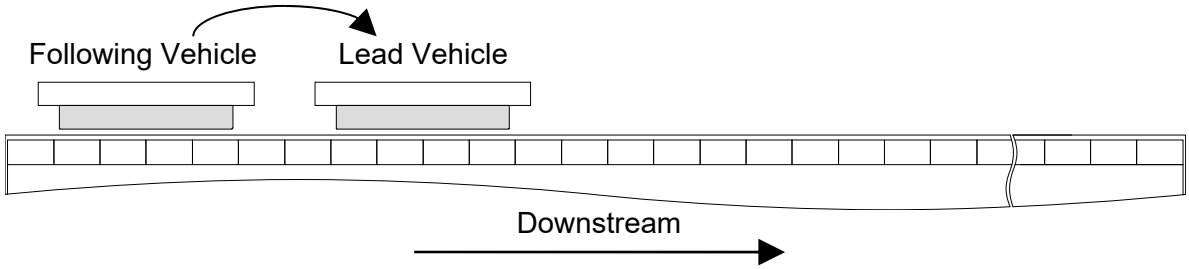

Figure 1: Create a Downstream Platoon

1. While the vehicle that is going to be the platoon leader is stopped, command the following vehicle to move downstream into position behind (upstream of) it. The vehicle follow order command must specify this distance between the vehicles as the follow distance.

   The following vehicle moves to its commanded position.

2. Couple the following vehicle to the lead vehicle and specify the distance (from Step 1) between the vehicles and the direction of motion (downstream) using a vehicle follow command.

   The vehicles are linked into a platoon at the specified distance between the vehicles.

3. Command the lead vehicle to move downstream as required.

   The following vehicle follows behind the lead vehicle at the specified distance.

To add additional vehicles to the platoon, see Add Additional Vehicles to the End Of a Platoon on page 13.

Follow a Vehicle Moving Upstream

Create a two-vehicle platoon that moves upstream by specifying a vehicle for another vehicle to follow.

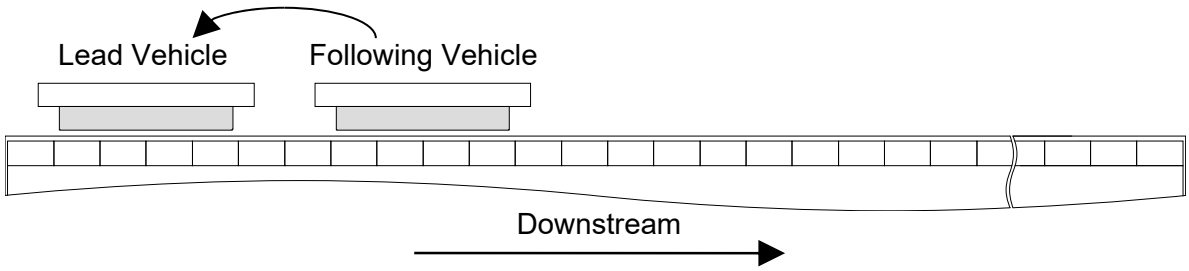

Figure 2: Create an Upstream Platoon

1. While the vehicle that is going to be the platoon leader is stopped, command the following vehicle to move upstream into position behind (downstream of) it. The vehicle follow order command (Step 2) must specify this distance between the vehicles as the follow distance.

   The following vehicle moves to its commanded position.

2. Couple the following vehicle to the lead vehicle and specify the distance (from Step 1) between the vehicles and the direction of motion (upstream).

   The vehicles are linked into a platoon at the specified distance between the vehicles.

3. Command the lead vehicle to move upstream as required.

   The following vehicle follows behind the lead vehicle at the specified distance.

To add additional vehicles to the platoon, see Add Additional Vehicles to the End Of a Platoon on page 13.

Add Additional Vehicles to the End Of a Platoon

Extend a multi-vehicle platoon that moves either downstream or upstream by adding additional vehicles to the end of the platoon. The additional vehicles are added in the direction of motion of the platoon by specifying the last vehicle in the platoon as the vehicle to follow for the vehicle being added. The vehicle being added can be either one vehicle or the lead vehicle in another platoon.

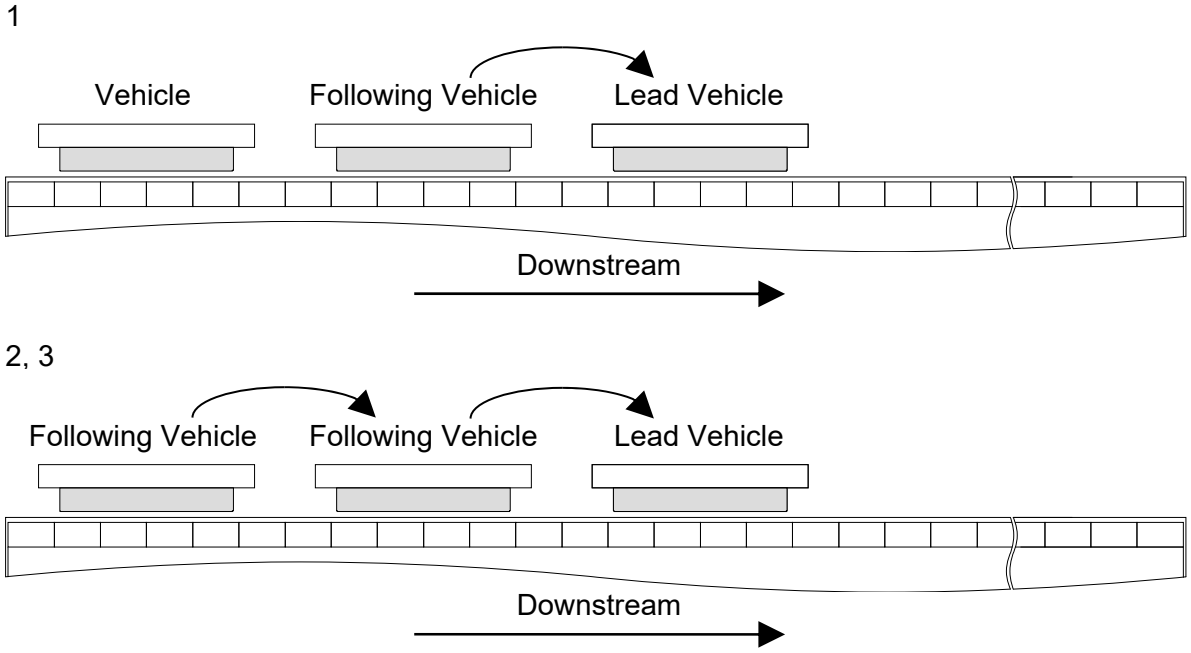

Figure 3: Add Vehicles to an Existing Platoon

1. While the platoon is stopped, command the next following vehicle (either one vehicle or the lead vehicle in another platoon) to move into position behind the platoon. The vehicle follow order command must specify this distance between the vehicles as the follow distance.

   The following vehicle moves to its commanded position.

2. Couple the next following vehicle to the platoon by specifying the last vehicle in the platoon as the vehicle to follow. Specify the distance (from Step 1) between the vehicles and the direction of motion of the platoon.

   The vehicles are linked into a platoon at the specified distance between the vehicles.

3. Command the lead vehicle as required.

   All following vehicles follow behind the lead vehicle at the specified distances.

Add a New Leader to a Platoon

Extend a multi-vehicle platoon that moves either downstream or upstream by coupling the lead vehicle in the platoon to a vehicle, which creates a new lead vehicle for the platoon. The platoon is coupled to the vehicle in the direction of motion of the platoon by specifying the vehicle for the current lead vehicle in the platoon to follow. The vehicle being coupled can be either one vehicle or the lead vehicle in another platoon.

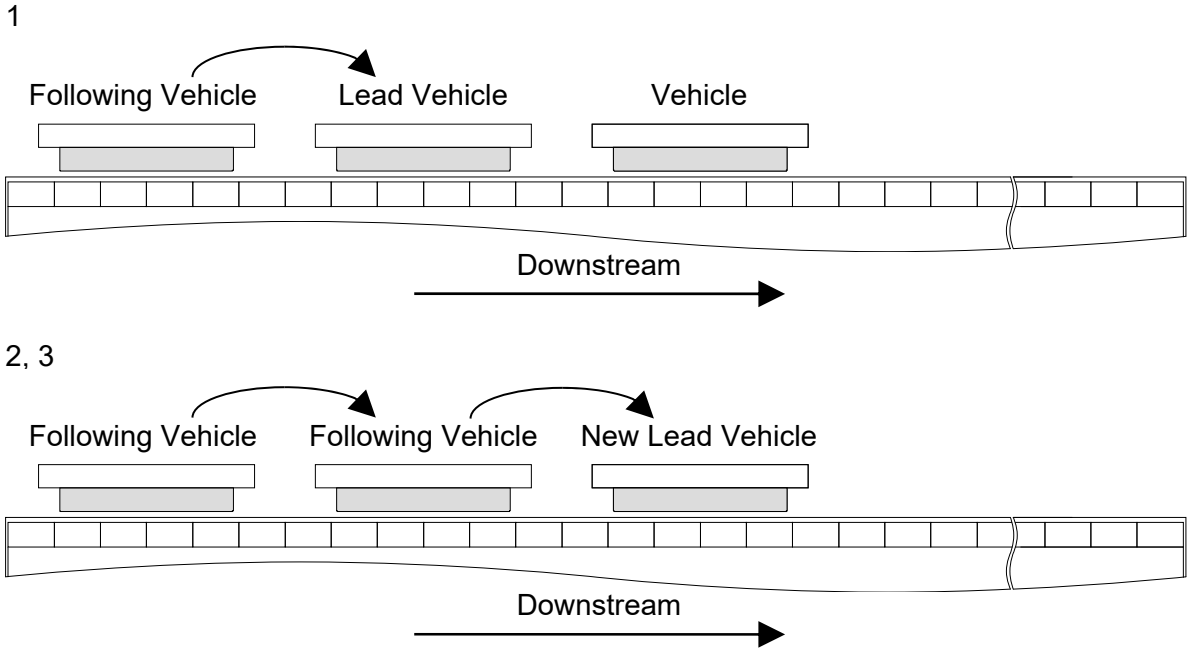

Figure 4: Add a New Lead Vehicle to a Platoon

1. While the vehicle that becomes the new lead vehicle (or the last vehicle in a platoon) is stopped, command the lead vehicle in the existing platoon to move into position behind that vehicle. The vehicle follow order command must specify this distance between the vehicles as the follow distance.

   The existing platoon follows the lead vehicle as it moves to its commanded position.

2. Couple the platoon by specifying the new lead vehicle (or the last vehicle in a platoon) as the vehicle for the platoon leader to follow. Specify the distance between the vehicles (from Step 1) and the direction of motion of the platoon.

   The platoon is linked to its new lead vehicle at the specified distance between the vehicles.

3. Command the new lead vehicle as required.

   All following vehicles follow behind the lead vehicle at their specified distances.

Uncoupling Platoons

Existing platoons can be uncoupled once the need for platooning is complete. Platoons can be split into their individual vehicles, or they can be split into smaller platoons.

Separate a Platoon Into Vehicles

Once a platoon has been created, it can be separated into uncoupled vehicles when the need for platooning is complete. To remove a vehicle from a platoon, send it a new vehicle movement command.

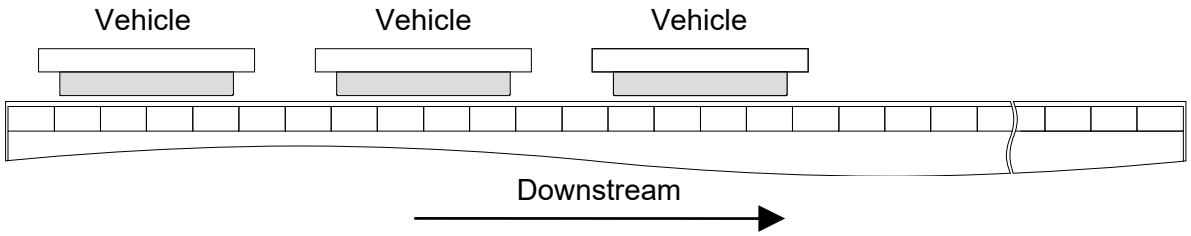

Figure 5: Separate a Platoon into Vehicles

1. With the platoon stopped, command vehicles to be uncoupled from the platoon.
   - To separate a vehicle from the vehicle it is following in the platoon without moving the vehicle, command it to move to its current target as returned in by either Vehicle Status or Extended Vehicle Status. The move vehicle command must specify bidirectional motion.

     The vehicle moves to its current target position (no actual motion occurs), which uncouples it from the vehicle it is following. If there is a vehicle that is coupled to this vehicle, it remains coupled until it is uncoupled.
   - To separate the last vehicle from the platoon and move the vehicle, command it to move to its new location. The move vehicle command must specify motion in the direction away from the platoon.

     The vehicle moves to its new position, which uncouples it from the vehicle it is following.
2. Once the vehicles are uncoupled, command them to their new positions.

Separate a Platoon into Smaller Platoons

Once a platoon has been created, one section can be uncoupled from the platoon, creating smaller platoons when the need for the larger platoon is complete. To split the platoon, remove the vehicle designated to be the leader of the new platoon from the existing platoon by sending it a new vehicle movement command.

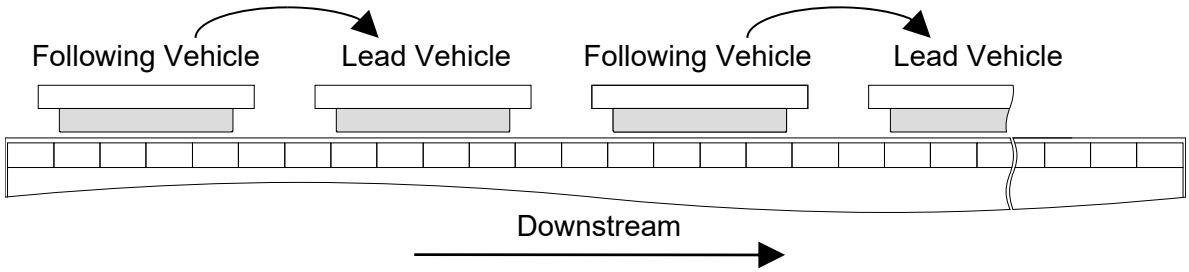

Figure 6: Separate a Platoon into Smaller Platoons

1. With the platoon stopped, determine where to split the platoon into two. Command the vehicle that becomes the lead vehicle for the new platoon to move to its current target as returned in by either Vehicle Status or Extended Vehicle Status. The move vehicle command must specify bidirectional motion.

   The vehicle moves to its current target position (no actual motion occurs), which uncouples it from the vehicle it is following. All vehicles that are coupled to this vehicle remain coupled and this vehicle becomes the leader of a new platoon.

2. Once the platoon is separated into smaller platoons, command the lead vehicle of each platoon to its new position.

Separate a Vehicle from a Platoon at a Specified Position

When a platoon is created, the follow order for a vehicle can specify a decouple position. When the platoon with the following vehicle with that position specified as part of its follow order approaches that position, the vehicle decouples from the platoon, slows down, and stops at that position. Any vehicles that are following the vehicle that decouples from the platoon continue to follow it after it decouples and stops. The remaining platoon continues to move under its original motion order.

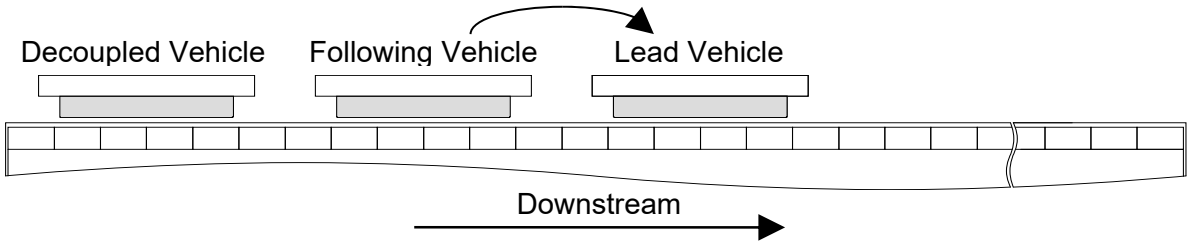

Figure 7: Separate a Vehicle from a Moving Platoon

1. While the platoon is stopped, command the following vehicle that is going to decouple from the platoon to move into position behind it. The vehicle follow order command must specify this distance between the vehicles as the follow distance.

2. Couple the following vehicle to the lead vehicle and specify the distance (from Step 1) between the vehicles and the direction of motion using a vehicle follow command. This vehicle follow command must also define the decouple path and the final position for this vehicle.

3. Command the lead vehicle to move as required, the following vehicle follows it at the specified distance.

   When the platoon approaches the decouple position the vehicle decouples from the platoon, slows down, and stops at that position. Any vehicles that are following the vehicle that decouples from the platoon continue to follow it after it decouples and stops.

Change Platoon Direction

Once a platoon has been created to move in a specific direction, the direction of the platoon can be reversed by stopping it, separating the vehicles in the platoon, and creating a platoon in the opposite direction. The lead vehicle for the new platoon is the vehicle that was the last following vehicle.

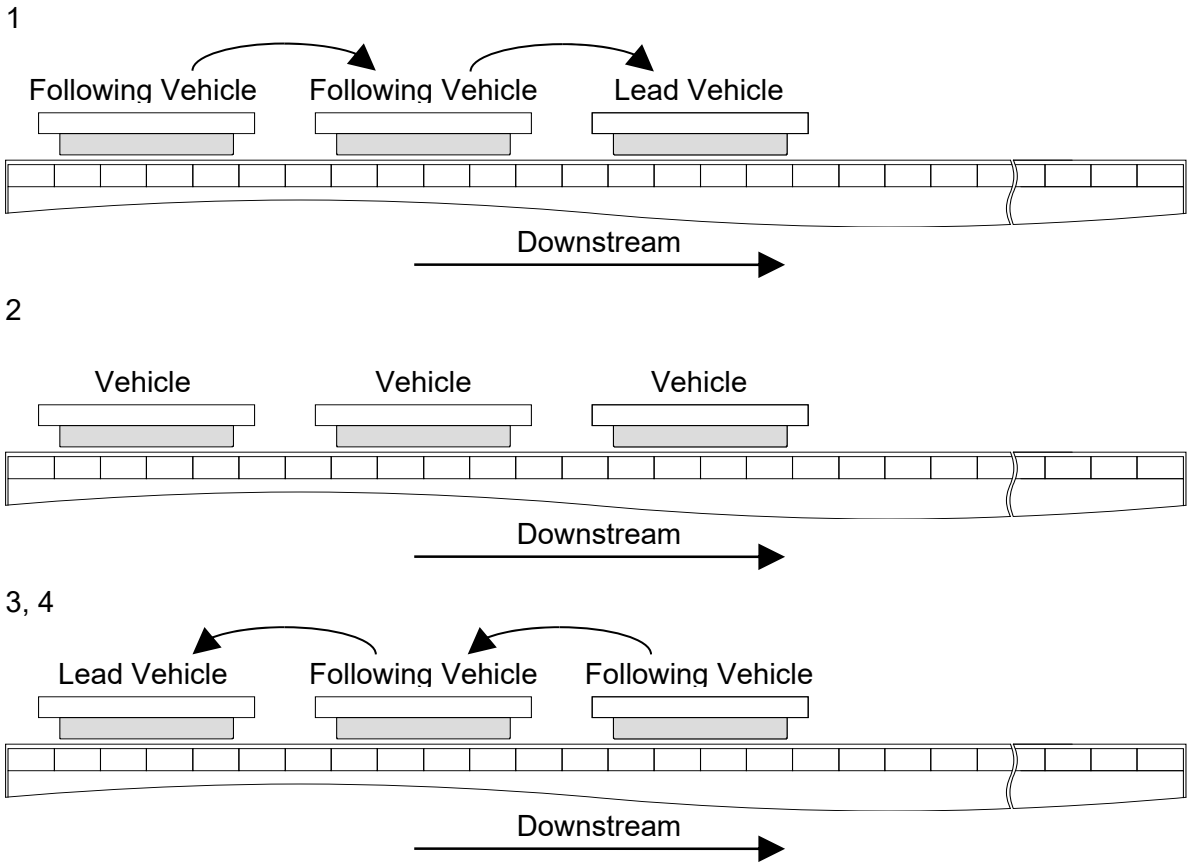

Figure 8: Change Platoon Direction

1. With the platoon stopped, command each following vehicle to move to its current target as returned through the Vehicle Status or Extended Vehicle Status. The move vehicle command must specify bidirectional motion.

   Each vehicle moves to its current target position (no actual motion occurs), which uncouples it from the vehicle it is following.

2. Using the last vehicle from the previous platoon as the new platoon leader, couple the new following vehicle to the new lead vehicle and specify the distance (from Step 1) between the vehicles and the new direction of motion. The vehicle follow order command must specify this distance between the vehicles as the follow distance.

   The vehicle is linked into the new platoon as specified.

3. Couple the next following vehicle to the new platoon by specifying the last vehicle in the new platoon as its vehicle to follow and specify the distance between the vehicles and the

new direction of motion. The vehicle follow order command must specify this distance between the vehicles as the follow distance.

The vehicle is linked into the new platoon as specified.

4. Repeat Step 3 for each additional vehicle in the platoon.

5. Command the new lead vehicle as required, all following vehicles follow it at their specified distances.

   The following vehicles follow behind the lead vehicle at their specified distances.

Troubleshooting

Platooning Across a Node

Attempting to couple across a node that is already part of a platoon causes the Vehicle Follow Order command to be rejected. For instance, Figure 9 shows a platoon with vehicle 3 coupled to vehicle 4. If a Vehicle Follow Order to couple vehicle 1 to vehicle 2 is sent, this order is rejected since the vehicle 3 – vehicle 4 platoon has locked the Merge-Diverge Node.

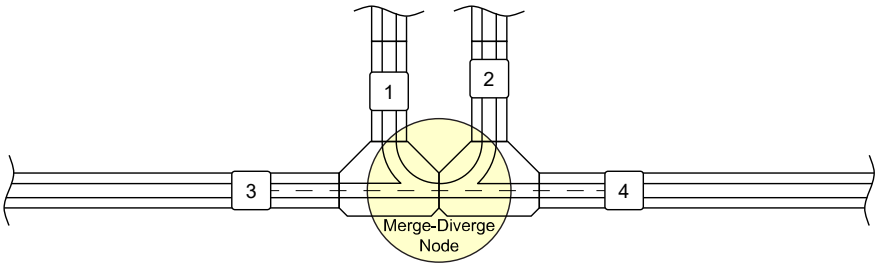

Figure 9: Platooning Across a Node

NCHost TCP Interface Utility User Manual Addendum

This section is an addendum to the *NCHost TCP Interface Utility User Manual*, MMI-UM010, and describes changes made to the NCHost User Interface UI and additions made for support of vehicle platooning.

Support for the changes that are described in this section starts with the following NCHost versions:

- NCHost – 15.0.x.

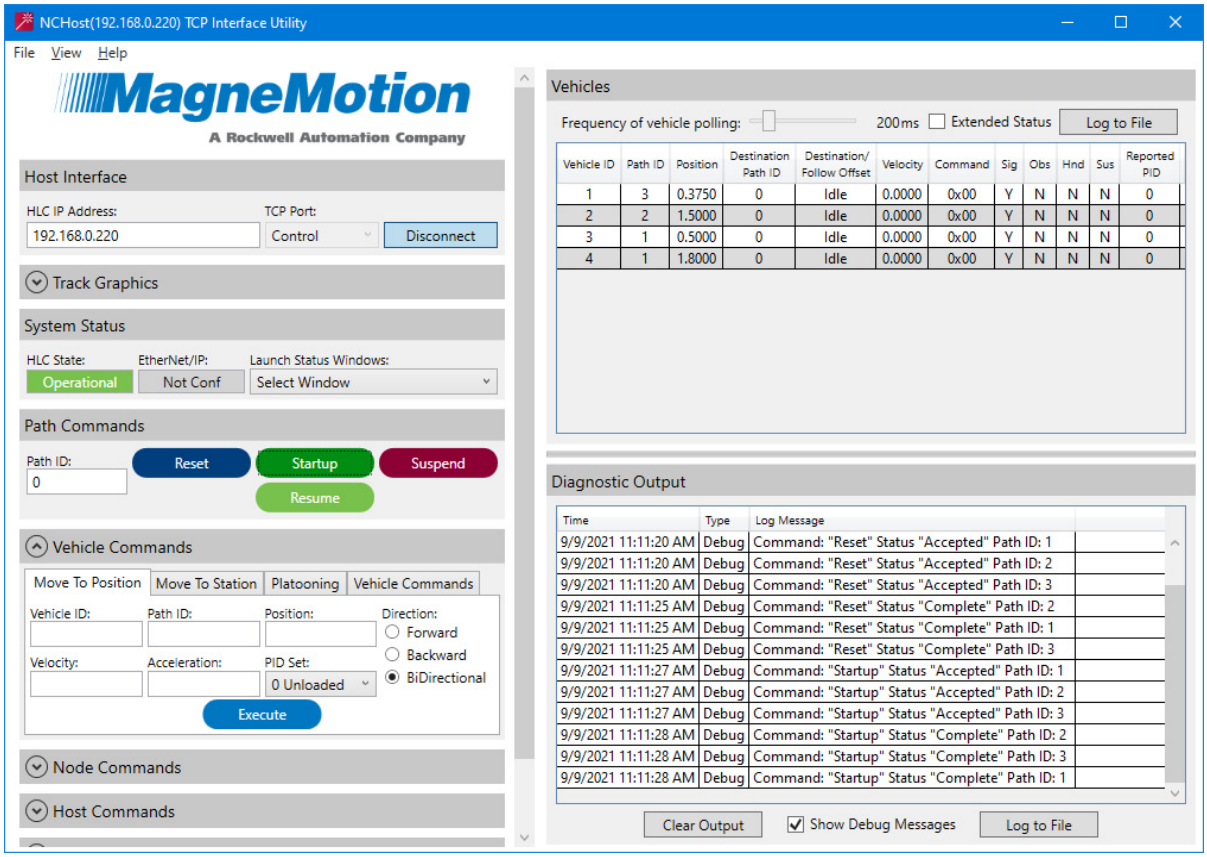

Figure 10: NCHost TCP Interface Utility for MagneMover LITE

Vehicle Commands

The Vehicle Commands section of NCHost is the same for both MM LITE and QuickStick and is used to send vehicle-specific commands. This section is expanded and redesigned to simplify access to the vehicle commands as shown in Figure 10. This section now includes a new tab with commands for vehicle platooning. The following sections provide a description of each of the new command tabs and the functions available through those tabs.

Move To Position

Use this tab to move the specified vehicle to the position on the specified path using the load, direction, acceleration, and velocity values provided.

CRUSH HAZARD: Moving mechanisms have no obstruction sensors.

Do not operate the transport system without barriers in place or personal injury could result from the squeezing or compression of fingers, hands, or other body parts between moving mechanisms.

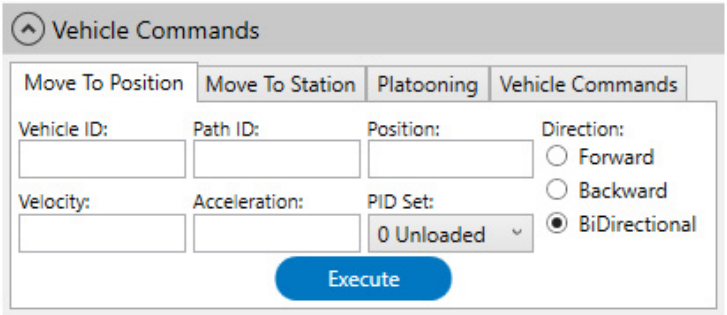

Figure 11: Move To Position Commands

- **Vehicle ID** – The ID number of the vehicle.
- **Path ID** – The ID number of the path where the vehicle destination is located.
- **Position** – The vehicle destination from the start of the path that is specified in **Path ID**, in meters.
- **Direction** – Provides an option list for selecting the direction of movement for the vehicle.
    - **Forward** – The vehicle must move forward (downstream) to go to the designated position.
    - **Backward** – The vehicle must move backward (upstream) to go to the designated position.
    - **BiDirectional** – The vehicle can move forward (downstream) or backward (upstream) to go to the designated position.
- **Velocity** – The maximum velocity for the vehicle to use (in m/s) when moving (system dependent).
- **Acceleration** – The maximum acceleration/deceleration for the vehicle to use (in m/s^2) when moving (system dependent).

- **PID Set** –Provides a menu to select one of the PID sets defined in the Node Controller Configuration File to use when moving the vehicle.

  NOTE: If the transport system is not configured for loaded/unloaded PID values, the unloaded values are always used.

  - **0 Unloaded** – The vehicle is unloaded. If the transport system is configured for loaded/unloaded PID values, the unloaded values are used.
  - **1 Loaded** – The vehicle is loaded. If the transport system is configured for loaded/unloaded PID values, the loaded values are used.
  - **2…15** – The vehicle uses the specified custom PID set.

- **Execute** – Select to move the specified vehicle to the position on the specified path using the acceleration and velocity values provided.

Move To Station

Use this tab to move the specified vehicle to the specified station using the load, direction, acceleration, and velocity values provided.

CRUSH HAZARD: Moving mechanisms have no obstruction sensors.

Do not operate the transport system without barriers in place or personal injury could result from the squeezing or compression of fingers, hands, or other body parts between moving mechanisms.

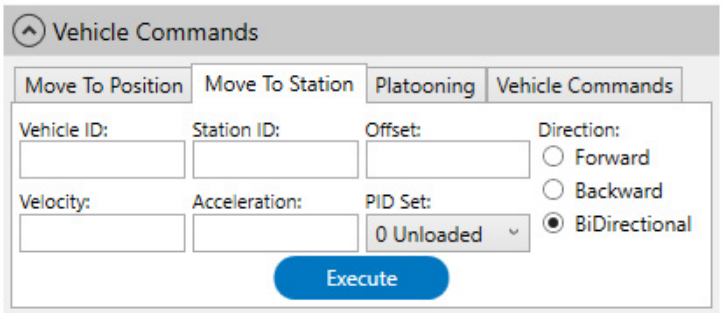

Figure 12: Move To Station Commands

- **Vehicle ID** – The ID number of the vehicle.
- **Station ID** – The ID number of the station for the vehicle destination on the path.

  NOTE: At least one station must be defined.
- **Offset** – Station offset, which is used when moving a vehicle to a station to shift the final position from the actual station position. Positive values are downstream of the station, negative values are upstream of the station.
- **Direction** – Provides an option list for selecting the direction of movement for the vehicle.
  - **Forward** – The vehicle must move forward (downstream) to go to the designated position.
  - **Backward** – The vehicle must move backward (upstream) to go to the designated position.
  - **BiDirectional** – The vehicle can move forward (downstream) or backward (upstream) to go to the designated position.
- **Velocity** – The maximum velocity for the vehicle to use (in m/s) when moving (system dependent).

- **Acceleration** – The maximum acceleration/deceleration for the vehicle to use (in m/s^2) when moving (system dependent).

- **PID Set** –Provides a menu to select one of the PID sets defined in the Node Controller Configuration File to use when moving the vehicle.

  NOTE: If the transport system is not configured for loaded/unloaded PID values, the unloaded values are always used.

  - **0 Unloaded** – The vehicle is unloaded. If the transport system is configured for loaded/unloaded PID values, the unloaded values are used.
  - **1 Loaded** – The vehicle is loaded. If the transport system is configured for loaded/unloaded PID values, the loaded values are used.
  - **2…15** – The vehicle uses the specified custom PID set.

- **Execute** – Select to move the specified vehicle to the specified station using the offset, acceleration, and velocity values provided.

Platooning

Use this tab to control vehicle platooning for the specified vehicle.

CRUSH HAZARD: Moving mechanisms have no obstruction sensors.

Do not operate the transport system without barriers in place or personal injury could result from the squeezing or compression of fingers, hands, or other body parts between moving mechanisms.

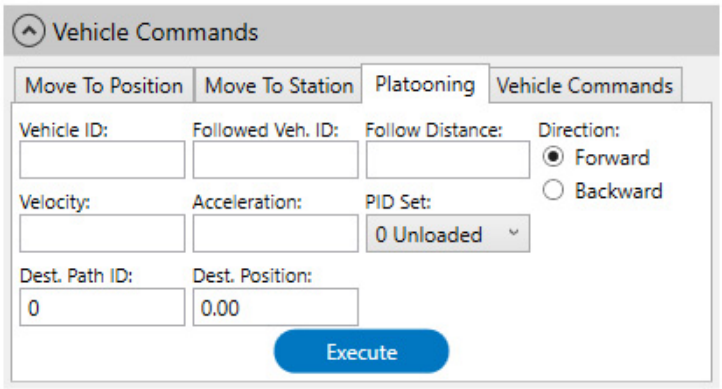

Figure 13: Platooning Commands

- **Vehicle ID** – The ID number of the vehicle being configured to follow another vehicle.
- **Followed Veh. ID** (Followed Vehicle ID) – The ID number of the vehicle to be followed (that is, leading this vehicle in the platoon).
- **Follow Distance** – The distance center-to-center that this vehicle maintains behind the vehicle being followed (specified by **Followed Veh. ID**) in the direction that is specified, once the pair of vehicles are coupled into a platoon.

  NOTE: When coupling to a platoon, the vehicle must be within 30 mm of this distance before sending the vehicle follow order.

- **Direction** – Provides an option list for selecting the direction of motion for the vehicle.
  - **Forward** – The vehicle to follow is downstream of this vehicle (motion is forward).
  - **Backward** – The vehicle to follow is upstream of this vehicle (motion is backward).
- **Velocity** – The maximum velocity for the vehicle to use (in m/s) when the vehicle is catching up to a moving vehicle to follow or when decoupling from a moving vehicle.

  NOTE: Currently not used for coupling as the vehicles must not be moving when creating a platoon, but it must be defined.

- **Acceleration** – The maximum acceleration/deceleration for the vehicle to use (in m/s^2) when the vehicle is catching up to a moving vehicle to follow or when decoupling from a moving vehicle.

  NOTE: Currently not used for coupling as the vehicles must not be moving when creating a platoon, but it must be defined.

- **PID Set** –Provides a menu to select one of the PID sets defined in the Node Controller Configuration File to use when moving the vehicle.

  NOTE: If the transport system is not configured for loaded/unloaded PID values, the unloaded values are always used.

  - **0 Unloaded** – The vehicle is unloaded. If the transport system is configured for loaded/unloaded PID values, the unloaded values are used.
  - **1 Loaded** – The vehicle is loaded. If the transport system is configured for loaded/unloaded PID values, the loaded values are used.
  - **2…15** – The vehicle uses the specified custom PID set.

- **Dest. Path ID** (Destination Path ID) – The path ID number for the automatic decoupling position.

  NOTE: This feature is only available on MagneMover LITE configurations.

  - **1…65535** – The ID number of the path where the automatic decoupling position is located (where the vehicle stops after auto-decoupling while in motion).
  - **0** – The automatic decoupling feature is disabled.

- **Dest. Position** (Destination Position) – The decoupling destination for the vehicle from the start of the path specified in **Dest. Path ID** in meters (where the vehicle stops after automatic decoupling while in motion).

  NOTE: This feature is only available on MagneMover LITE configurations.

- **Execute** – Select to implement the specified platooning command using the values provided.

Vehicle Commands

Use this tab to send vehicle-specific commands to the specified vehicle.

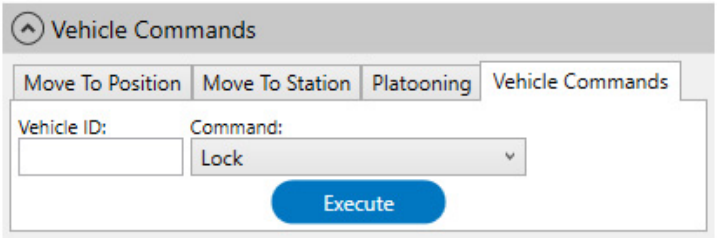

Figure 14: Vehicle Commands

- **Vehicle ID** – The ID number of the vehicle the command is applied to.
- **Command** – Provides a menu to select vehicle-specific commands.
  - **Lock** – Select to keep the specified vehicle from accepting new motion commands.

    NOTE: The vehicle lock status set using the NCHost TCP Interface Utility is temporary and is not restored if the node controller is reset or rebooted.
  - **Unlock** – Select to unlock a previously set lock, which allows the vehicle to be moved.

    NOTE: The vehicle lock status set using the NCHost TCP Interface Utility is temporary and is not restored if the node controller is reset or rebooted.
  - **Delete** – Deletes the specified vehicle record from the node controller and all motors. This function is typically used when a vehicle is physically removed from the guideway because it is inoperable. It can also be used when a vehicle entry is canceled, or a vehicle has been manually moved and the motor can no longer track it. If a vehicle entry or exit is active for the vehicle, the entry/exit state machine is also cleared.

IMPORTANT	Using the Delete Vehicle command to delete a vehicle that is physically present on the transport system can cause collisions as the system no longer accounts for that vehicle as it moves other vehicles.

  - **Clear Suspect Bit** – Allows the Suspect bit to be cleared once control of the vehicle is regained.
- **Execute** – Select to implement the specified vehicle command using the values provided.

Vehicles Pane

The Vehicles pane in NCHost is used to monitor vehicle behavior. This section has been redesigned to include information about vehicle in a platoon. Updates have been made to both the basic vehicle status (see Figure 15) and the advanced vehicle status (see Figure 16).

Vehicle Status

The Vehicles pane displays the status of all vehicles currently on the transport system. The display is available in both a basic and an extended version. Figure 15 shows the Basic Vehicle Status. Figure 16 shows the Advanced Vehicle Status.

Vehicles

Frequency of vehicle polling: 200ms ☐ Extended Status Log to File

Vehicle ID	Path ID	Position	Destination Path ID	Destination/ Follow Offset	Velocity	Command	Sig	Obs	Hnd	Sus	Reported PID
1	3	0.3750	0	Idle	0.0000	0x00	Y	N	N	N	0
2	2	1.5000	0	Idle	0.0000	0x00	Y	N	N	N	0
3	1	0.5000	0	Idle	0.0000	0x00	Y	N	N	N	0
4	1	1.8000	0	Idle	0.0000	0x00	Y	N	N	N	0

Figure 15: Basic Vehicle Status

Vehicles

Frequency of vehicle polling: 200ms ☑ Extended Status Log to File

Vehicle ID	Path ID	Position	Destination Path ID	Destination/ Follow Offset	Velocity	Command	Sig	Obs	Hnd	Sus	Stl	Lkd	Loc	FIU	FID	CUp	PSE	EFE	Dec	Reported PID	Ordered PID	Target	Followed ID	Ordered Accel Limit	Ordered Velocity Limit	Active Move To Station Offset
1	2	0.7745	2	1.000	0.0000	0xB1	Y	Y	N	N	N	N	N	N	N	N	N	N	N	1	1	0.7745	0	0.9996	0.5000	0.0000
2	1	1.8243	3	0.500	0.0000	0xB1	Y	Y	N	N	N	N	N	N	N	N	N	N	N	0	0	1.8243	0	0.9996	0.5000	0.0000
3	1	0.5000	1	Idle	0.0000	0x00	Y	N	N	N	N	N	N	N	N	N	N	N	N	1	1	0.5000	0	0.9996	0.5000	0.0000
4	2	2.6435	3	0.500	0.0000	0xB1	Y	Y	N	N	N	N	N	N	N	N	N	N	N	0	0	2.6435	0	0.9996	0.5000	0.0000

Figure 16: Extended Vehicle Status

Select any of the column headers to sort the contents of the Vehicles pane by that column. An arrow is displayed that indicates the sort order. Select any of the dividers between columns to change the width of the column to the left.

- **Frequency of vehicle polling** – Defines the rate at which the data within the **Vehicles** pane is updated. The slider range is 20 milliseconds–1000 milliseconds (1 second).
- **Extended Status** – Select to display extended vehicle status.
- **Log To File/Close Log** – Toggles between **Log to File** and **Close Log** based on the current log status.
    - **Log To File** – Displayed when there is no log file currently open. Select to open a dialog box to either create a file or select an existing file and start logging vehicle data to that file. Data is logged at the rate set by the **Frequency of vehicle polling** slider.
    - **Close Log** – Displayed when there is a log file in use. Select to stop file logging. The log file is locked while in use, once the log file is closed, the file can be opened to view the contents using a text editor.

- **Vehicle ID** – Displays the ID number of each vehicle in the transport system as assigned by the high-level controller.

- **Path ID** – Displays the ID number of the path, as defined in the Node Controller Configuration File, where the vehicle is located.

- **Position** – Displays the last reported position of the vehicle in meters from the beginning of the path where the vehicle is located.

- **Destination Path ID** – Displays the ID of the path where the vehicle is headed. Equal to the ID of the path where the vehicle is located if motion has completed. Equal to 0 if the vehicle has never moved.
    - ***value*** – When the vehicle is under a motion command, this value is the ID of the path where the vehicle is heading. When vehicle motion has completed, this value is the ID of the path where the vehicle is located. If the vehicle has never moved, this value is zero.
    - **Following** – The vehicle is following another vehicle in a platoon. The destination of the lead vehicle determines the destination for all vehicles in the platoon.

- **Destination/Follow Offset** – Displays the commanded position on the destination path or the vehicle movement status for the vehicle.
    - ***value*** – For one vehicle or the lead vehicle in a platoon, when the vehicle is under a motion command, this value is the commanded position on the destination path. When the vehicle is coupled into a platoon, this value is the distance to the vehicle it is following even if it is not moving.
    - **Idle** – The vehicle is at the commanded position. Or the vehicle has never moved.

- **Velocity** – Displays the current velocity of the vehicle (in m/s).

- **Command** – Displays the current command from the host controller for the vehicle.

- **Sig** (Signal Detected) – Displays the detection status for the vehicle as determined by the motor currently in charge of the vehicle.
    - **N** – Detection and location of the vehicle on the path has been lost.
    - **Y** – Vehicle has been detected and its location identified.

- **Obs** (Obstructed) – Displays the vehicle obstruction status.
    - **N** – The vehicle is not obstructed and is free to move.
    - **Y** – The vehicle is obstructed and waiting for permission to move further. Obstructions occur when a vehicle is in the way (entering a switch) or a hardware fault.

- **Hnd** (Hindered) – Displays the vehicle hindered (jam) status.
    - **N** – The vehicle is not hindered and is moving as expected.

- **Y** – The vehicle is not moving despite having permission to move. Vehicles are hindered when they are held back by an external force including a foreign object jamming a vehicle on the guideway or someone physically holding the vehicle back. The motor continues to apply force on the vehicle to try to move it indefinitely.

- **Sus** (Suspect) – Displays the vehicle suspect bit status.

  NOTE: This feature is only available on MM LITE and QuickStick configurations.

  - **N** – The vehicle is not suspect and is free to move.
  - **Y** – The vehicle is suspect. The motor has detected that the vehicle has been manually moved out of the control region and the motor cannot guarantee vehicle control, even after moving the vehicle back into its control region.

    NOTE: Suspect vehicles should be deleted, then restart each path reporting an unlocated vehicle fault.

- **Stl** (Stall) – Displays the vehicle stall bit status (only displayed when **Extended Status** is selected).
  - **N** – The vehicle is not stalled and is free to move.
  - **Y** – The vehicle has stalled. The motor has detected that the vehicle is not moving, and the motor has had to reduce power to help prevent overheating.

- **Lkd** (Locked) – Displays the vehicle lock status (only displayed when **Extended Status** is selected).
  - **N** – The vehicle is not locked and is free to move.
  - **Y** – The vehicle is locked and cannot be moved.

- **Loc** (Located) – Displays the vehicle locate bit status (only displayed when **Extended Status** is selected).
  - **N** – Vehicle locate is not completed (startup is in progress).
  - **Y** – Vehicle locate has completed.

- **FlU** (Following Upstream) – The vehicle in a platoon is following another vehicle upstream (only displayed when **Extended Sta**tus is selected).
  - **N** – This vehicle is not following another vehicle upstream.
  - **Y** – This vehicle is following the vehicle that is specified in the Followed ID field and that vehicle is upstream from this vehicle.

- **FlD** (Following Downstream) – The vehicle in a platoon is following another vehicle downstream (only displayed when **Extended Status** is selected).
  - **N** – This vehicle is not following another vehicle downstream.
  - **Y** – This vehicle is following the vehicle that is specified in the Followed ID field and that vehicle is downstream from this vehicle.

- **CUp** (Caught Up) – The vehicle in a platoon is caught up to the vehicle it is following (only displayed when **Extended Status** is selected).
    - **N** – This vehicle is not caught up to the vehicle it is following, or it is not following another vehicle.
    - **Y** – This following vehicle has caught up to the vehicle it is following and is at the requested following distance.

- **PSE** (Profile Stale Error) – PSE (Profile Stale Error) – The vehicle in a platoon is not receiving profile data (acceleration and velocity) from the vehicle it is following (only displayed when **Extended Status** is selected).
    - **N** – There is no Profile Stale Error, or this vehicle is not following another vehicle.
    - **Y** – This following vehicle has not received profile data from the vehicle it is following, in the expected time (~10 ms). The vehicle slows to a stop. If the profile data is restored, the vehicle catches up to its leader using the catchup acceleration and velocity.

- **EFE** (Excessive Following Error) – The vehicle in a platoon is too far from the vehicle it is following (only displayed when **Extended Status** is selected).
    - **N** – There is no Excessive Following Error, or this vehicle is not following another vehicle.
    - **Y** – This following vehicle is too far from where it is expected to be (possibly hindered or stopped receiving following profile data).

- **Dec** (Decoupled) – Displays the auto-decouple status of the vehicle (only displayed when **Extended Status** is selected).
    - **N** – This vehicle is a platoon follower that is not in the decoupling state, or not a platooned vehicle.
    - **Y** – This vehicle is a platoon follower that is decoupling from the vehicle it is following. This bit stays set from when the motor reports decoupling, until the decoupled vehicle reaches its new destination under the 0xB1 position order.

- **Reported PID** – Displays the PID Set being used by the vehicle for motion.
    - **0** – The vehicle is not loaded (using PID Set index 0).
    - **1** – The vehicle is loaded (using PID Set index 1).
    - **2…14** – The vehicle is using the specified PID Set.
    - **15** – Startup PID values. This PID Set is used during startup. If it is not defined, PID Set 0 is scaled by 25% and used for startup.

- **Ordered PID** – Displays the currently commanded PID Set for use for vehicle motion during the current move command (only displayed when **Extended Status** is selected).
    - **0** – The vehicle is not loaded (PID Set index 0).

  - **1** – The vehicle is loaded (PID Set index 1).
  - **2…14** – The vehicle uses the requested PID Set.
  - **15** – The vehicle uses startup PID values.

- **Target** – Displays the next interim target location (in meters) that the vehicle has permission to move to from the beginning of the path where the vehicle is located. It is equal to the position of the vehicle when the vehicle is not moving (only displayed when **Extended Status** is selected).

- **Followed ID** – Displays the ID number of the vehicle that this vehicle is following when this vehicle is under a vehicle follow order and is part of a platoon. Equal to 0 if this vehicle is not under a vehicle follow order (only displayed when **Extended Status** is selected).

- **Ordered Accel Limit** (Ordered Acceleration Limit) – Displays the commanded acceleration and deceleration for vehicle motion during the current move command (only displayed when **Extended Status** is selected).

- **Ordered Velocity Limit** – Displays the commanded maximum velocity for vehicle motion during the current move command (only displayed when **Extended Status** is selected).

- **Active Move To Station Offset** – Displays the commanded offset for the vehicle from the station that is the current destination (only displayed when **Extended Status** is selected).

- **Horizontal Scroll Bar** – Scrolls the status columns to display all columns. The scroll bar appears below the display when the vehicle display contains more vehicle status columns than can be displayed in the screen space provided (typically when **Extended Status** is selected.

- **Vehicle Scroll Bar** – Scrolls the list of vehicles vertically to display all vehicles on the transport system. The scroll bar appears to the left of the display when the vehicle display is populated with more vehicles than can be displayed in the screen space provided.

Create a Platoon

When coupling vehicles to create a platoon, the vehicles must be stopped. Stopped is defined as movement must be below the **Arrival Velocity Tolerance** defined on the **Global Settings** page in the Configurator when creating the Node Controller Configuration File.

1. In the **Vehicle Commands** section, select the **Move to Position** tab (see Figure 17) and move the second vehicle into position behind the leader (this distance must be the same as the **Follow Distance**).

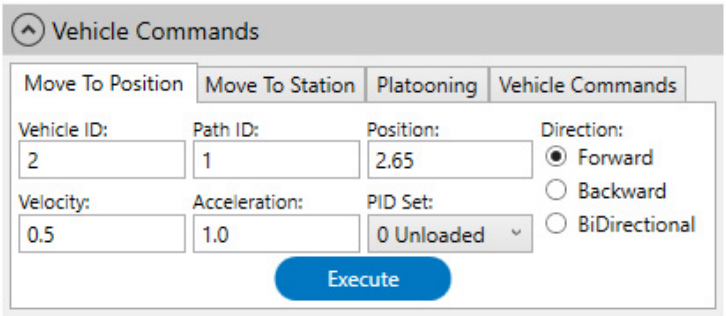

Figure 17: Manual Vehicle Commands Section – Move Vehicle Into Position

2. In the **Vehicle Commands** section, select the **Platooning** tab (see Figure 18) and couple the second vehicle to the first vehicle.

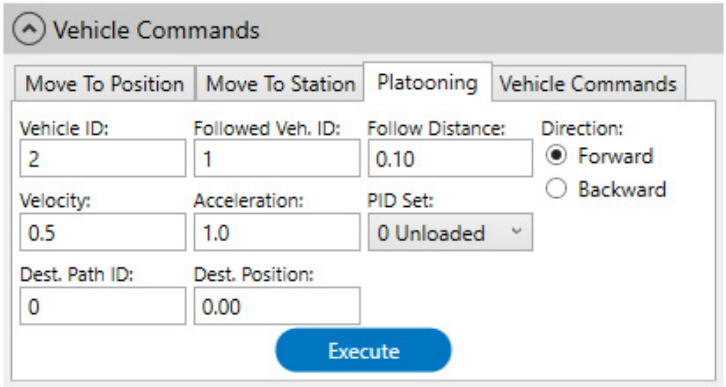

Figure 18: Manual Vehicle Commands Section – Couple Vehicle to Platoon

A. While the vehicle that is going to be the following vehicle is stopped, enter the **Vehicle ID** of that vehicle.

B. While the vehicle that is going to be the platoon leader is stopped, enter the **Followed Veh. ID** of that vehicle (the leading vehicle or the last vehicle in the platoon).

C. Enter the **Follow Distance** to maintain between the vehicles.

NOTE: When coupling to a platoon, the vehicle must be within 30 mm of this distance before sending the vehicle follow order.

D. Select the **Direction** of motion for the vehicle to use to follow the leading vehicle, which is the direction the platoon is moving.

E. Enter the **Velocity** and **Acceleration** (system dependent).

NOTE: Currently only used when decoupling as vehicles must not be moving when creating a platoon, but it must always be defined.

F. Select the **PID Set** to use.

G. Select **Execute**.

The second (following) vehicle is coupled to the first (lead) vehicle.

3. Command the first vehicle as required, the second vehicle follows it at the specified distance.

Separate a Platoon

When uncoupling a vehicle from a platoon, the platoon must be stopped. Stopped is defined as movement must be below the **Arrival Velocity Tolerance** defined on the **Global Settings** page in the Configurator when creating the Node Controller Configuration File.

1. In the **Vehicle Commands** section, select the **Move to Position** tab (see Figure 19).

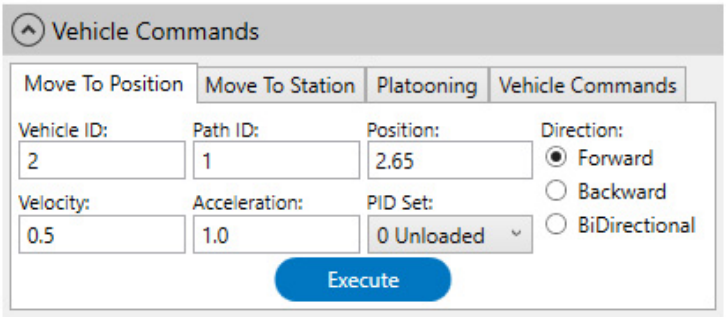

Figure 19: Manual Vehicle Commands Section – Uncouple Vehicle From Platoon

A. While the platoon is stopped, enter the **Vehicle ID** of the following vehicle.

B. Enter the current **Path ID**.

C. Enter the current target for the vehicle as returned through the Extended Vehicle Status as the **Position**.

D. Select BiDirectional as the **Direction** of motion.

E. Enter an **Acceleration** and **Velocity** for the vehicle.

F. Select the **PID Set** to use.

G. Select **Execute**.

The vehicle moves as directed, which uncouples it from the vehicle it is following. Since the vehicle was commanded to its current target position, no actual motion occurs. If there is a vehicle that is coupled to this vehicle, it remains coupled and this vehicle becomes the leader of a new platoon.

4. Command the individual vehicles as required.

Host Controller TCP/IP Communication Protocol User Manual Addendum

This section is an addendum to the *Host Controller TCP/IP Communication Protocol User Manual*, MMI-UM003, which describes the commands that support vehicle platooning. The additions to the protocol for communication between the high-level controller (HLC) and a host controller that is equipped with a TCP/IP interface are described.

Support for the changes that are described in this section starts with the following NC Software Image versions:

- MM LITE – 15.11.20.
- QuickStick – 15.10.23.

Host Controller to HLC Communications

This section describes the commands (listed in Table 1) sent from the host controller to the HLC as asynchronous requests for the transport system to perform an action. These requests are responded to by the HLC by routing the command to the appropriate node controller for completion and sending responses to the host controller (listed in Table 2).

Table 1: Host Controller to HLC Commands

Description and Value	Use	Page
Move Vehicle To Position (0xB1)	MM LITE, QS 100	37
Vehicle Follow Order (0xB7)	MM LITE, QS 100	40
Get Extended Vehicle Status, Ver 3 (0xBF 03 06)	MM LITE, QS 100	44

Move Vehicle To Position (0xB1)

Type

Host Controller → HLC

Purpose

Moves the specified vehicle to a specified position, relative to the start of a path.

To monitor vehicle status, the Vehicle Status (0xD5) or Extended Vehicle Status, Ver. 3 (0xDF 03 06) commands can be used.

NOTE: The high-level controller, node controllers, and the path the vehicle resides on must be in the operational state. If the path where the vehicle is located is not in an operational state, the command is rejected. If any component required to execute the command is not in operational state, the command does not complete.

If the commanded acceleration or velocity values are higher than the limit that is defined in the Node Controller Configuration File, the command is rejected.

If the commanded velocity value is higher than the value for a specific motor, the motor velocity value overrides the commanded value while the vehicle is on that motor.

If a move command is issued to a vehicle already in motion and the command has a lower acceleration than the previous command to that vehicle, the command is rejected.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

Format

Offset	Item	Bytes	Range
0	Command Header	1	0xB1
1	Order Number	4	0x00000000…0xFFFFFFFF
5	Vehicle ID	2	1…65535
7	Flags and Direction	1	Bits 0…3 are direction values Bits 4…7 are the PID set index
8	Position	4	-41.0…+41.0 (m, floating point)
12	Path ID	2	1…65535
14	Acceleration/Deceleration	4	0.0…60.0 (m/s^2, floating point)
18	Velocity	4	0.0…5.0 (m/s, floating point)

Arguments

Command Header – Fixed message type that identifies this message as a Move Vehicle to Position command.

Order Number – Host controller derived unique order number that is used for tracking, which is included in the command status response messages that the high-level controller returns.

Vehicle ID – The ID number of the vehicle to move. The ID must be a nonzero 16-bit positive integer that references a vehicle that exists in the transport system.

Flags and Direction – Bits 0…3 indicate the direction to move the vehicle. Bits 4…7 indicate the PID set index to use when executing the move command.

Bit	Description
0…3	**Direction:** **0 – BiDirectiona**l – If the destination is on the same path the vehicle resides on, the vehicle can move either direction as required to get to the destination in the shortest distance. If the destination is on a path other than the path where the vehicle is located, the forward direction takes precedence for a transport system that is a closed-loop. **1 – Forward** – Force the vehicle to move forward (downstream) only, useful to implement a unidirectional loop. If the destination is not reachable the command fails, and the vehicle is not moved. **2 – Backward** – Force the vehicle to move backward (upstream) only, useful to implement a unidirectional loop in the backwards direction. If the destination is not reachable the command fails, and the vehicle is not moved. 3…15 – Reserved
4…7	**PID Loop Set Index:** 0 – Use user-defined PID set 0 – Unloaded PID values. 1 – Use user-defined PID set 1 – Loaded PID values. 2 – Use user-defined PID set 2. 3 – Use user-defined PID set 3. 4 – Use user-defined PID set 4. 5 – Use user-defined PID set 5. 6 – Use user-defined PID set 6. 7 – Use user-defined PID set 7. 8 – Use user-defined PID set 8. 9 – Use user-defined PID set 9. 10 – Use user-defined PID set 10. 11 – Use user-defined PID set 11. 12 – Use user-defined PID set 12. 13 – Use user-defined PID set 13. 14 – Use user-defined PID set 14. 15 – Use user-defined PID set 15 – Startup PID values. This PID set is automatically used during startup. If it is not defined, PID set 0 is scaled by 25% and used for startup.

Position – The destination (in meters) of the vehicle, relative to the start of the specified path. The position is expressed as a 32-bit single-precision floating point number using little-endian format. Zero position is defined as the midpoint of the vehicle at the beginning of the path. When decoupling a vehicle from a stationary platoon, the position that is specified must be the current target for the vehicle as returned through the Extended Vehicle Status. Negative positions can only be used with node types that support movement to a position before the beginning of a path.

Path ID – The ID number of the path where the position is located. The ID must be a nonzero 16-bit positive integer that references a path that exists in the configuration.

Acceleration/Deceleration – The maximum acceleration/deceleration for the vehicle (in m/s^2) to use when moving. This value is a positive number that is expressed as a 32-bit single-precision floating point number using little-endian format. This value is checked against the limit that is defined in the Node Controller Configuration File, and if higher, the command is rejected. If this value is within the limit that is defined in the Node Controller Configuration File but higher than the value for any specific motor in the route of the vehicle, the value for that motor is used and the command is updated to that lower value.

This value is system-dependent, see the *Host Controller TCP/IP Communication Protocol User Manual*, MMI-UM003.

Velocity – The maximum velocity for the vehicle (in m/s) to use when moving. This value is a positive number that is expressed as a 32-bit single-precision floating point number using little-endian format. This value is checked against the limit that is defined in the Node Controller Configuration File, and if higher, the command is rejected. If this value is within the limit that is defined in the Node Controller Configuration File, but higher than the value for any specific motor in the route of the vehicle, the value for that motor overrides the command value while the vehicle is on that motor.

This value is system-dependent, see the *Host Controller TCP/IP Communication Protocol User Manual*, MMI-UM003.

Response

After receiving the command and verifying the command parameters, the HLC sends either a "Command Accepted" response (0x00) or a "Command Rejected" response as appropriate (see Command Status (0xD0)). If the command is accepted, it is forwarded to the appropriate node controller for execution. If the command is rejected, the Command Status (0xD0) response includes a rejection code that shows the reason for rejection.

On completion of the command (when the HLC is notified that the vehicle has arrived), the HLC sends a Command Status (0xD0) message with a "Command Completed Successfully" response (0x80).

NOTE: If the vehicle never reaches its destination, this command never completes.

See Also

Command Status (0xD0) on page 47

Vehicle Follow Order (0xB7)

Type

Host Controller → HLC

Purpose

Orders a vehicle to catch up to another vehicle using a specified profile and then maintain a specified following distance. Once the vehicle has caught up to the specified vehicle at the commanded following distance, the system maintains the following distance indefinitely, matching the movement profile of the followed vehicle. The vehicle continues to follow the specified vehicle at the specified distance until the follow order is overridden with a Move Vehicle to Position Order, a Move Vehicle to Station Order, or a new Vehicle Follow Order.

There is an auto decouple feature, where a decouple path ID and position are defined. When the platoon with the following vehicle approaches that position, the vehicle with that position specified as part of its follow order decouples from the platoon, slows down, and stops at that position.

To monitor vehicle status, the Vehicle Status (0xD5) or Extended Vehicle Status, Ver. 3 (0xDF 03 06) commands can be used.

NOTE: The high-level controller, node controllers, and the path the vehicle resides on in the transport system must be in the operational state. If the path where the vehicle is located is not in an operational state, the command is rejected. If any component required to execute the command is not in operational state, the command does not complete.

Vehicles must be stopped to couple them into a platoon.

Platoons must be stopped to decouple vehicles unless the command coupling a vehicle to the platoon specified a decouple destination for that vehicle.

If the acceleration or velocity values in a move command are higher than the limit that is defined in the Node Controller Configuration File, the command is rejected.

If the commanded velocity value is higher than the value for a specific motor, the motor overrides that value while the vehicle is on that motor.

If a move command is issued to a vehicle already in motion and the command has a lower acceleration than the previous command to that vehicle, the command is rejected.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

Format

Offset	Item	Bytes	Range

Offset	Item	Bytes	Range
0	Command Header	1	0xB7
1	Order Number	4	0x00000000…0xFFFFFFFF
5	Vehicle ID	2	1…65535
7	Direction	1	1…2
8	Follow Distance	4	0.0…41.0 (m, floating point)
12	Followed Vehicle ID	2	1…65535
14	PID Set Index	1	0…15
15	Catchup or Decouple Acceleration	4	0.0…60.0 (m/s^2, floating point)
19	Catchup or Decouple Velocity	4	0.0…5.0 (m/s, floating point)
23	Decouple Destination Path ID (MM LITE only):	2	0…65535
25	Decouple Destination Position (MM LITE only):	4	0.0…41.0 (m, floating point)

Arguments

Command Header – Fixed message type that identifies this message as a Vehicle Follow Order command.

Order Number – Host derived order number included in the command status response message that the high-level controller returns. Allows the host to track internally any specific vehicle follow order message that is sent to the HLC.

Vehicle ID – The ID number of the vehicle to execute this vehicle follow order. This vehicle follows the vehicle that **Followed Vehicle ID** specifies. The ID must be a nonzero 16-bit positive integer that references a vehicle that exists in the transport system.

Direction – Direction on the track relative to this vehicle where the vehicle specified by **Followed Vehicle ID** is located:

Value	Description
1	Vehicle to follow is downstream of this vehicle (motion is downstream).
2	Vehicle to follow is upstream of this vehicle (motion is upstream).

Follow Distance – The distance (in meters) center-to-center that this vehicle maintains behind the vehicle being followed (specified by **Followed Vehicle ID**) in the **Direction** that is specified, once the pair of vehicles are coupled into a platoon. The distance is expressed as a 32-bit single-precision floating point number using little-endian format.

NOTE: When coupling to a platoon, the vehicle must be within 30 mm of this distance before sending the vehicle follow order.

Followed Vehicle ID – The ID number of the vehicle being followed by this vehicle (executing this vehicle follow order). The Followed Vehicle ID must be a nonzero 16-bit integer.

PID Set Index – The PID set to use when executing the command (see flags_and_direction in Move Vehicle To Position (0xB1)).

Catchup or Decouple Acceleration – The maximum acceleration/deceleration for the vehicle (in m/s^2) to use when catching up to the vehicle to follow or to use when auto-decoupling from the vehicle it is following (system dependent). This value is a positive number that is expressed as a 32-bit single-precision floating point number using little-endian format. This value is checked against the system limit that is defined in the Node Controller Configuration File, and if higher, the command is updated to the value from the Node Controller Configuration File. Only available in MagneMover LITE systems.

NOTE: Currently only used when decoupling as vehicles must not be moving when creating a platoon, but it must always be defined.

Catchup or Decouple Velocity – The maximum velocity for the vehicle (in m/s) to use when catching up to the vehicle to follow or to use when auto-decoupling from the vehicle it is following (system dependent). This value is a positive number that is expressed as a 32-bit single-precision floating point number using little-endian format. This value is checked against the system limit that is defined in the Node Controller Configuration File, and if higher, the command is updated to the value from the Node Controller Configuration File. Only available in MagneMover LITE systems.

NOTE: Currently only used when decoupling as vehicles must not be moving when creating a platoon, but it must always be defined.

Decouple Destination Path ID – The ID number of the path where the decoupling position is located (where the vehicle stops after decoupling). The ID must be a nonzero 16-bit positive integer that references a path that exists in the configuration. If the value is zero, the decoupling feature is disabled. Only available in MagneMover LITE systems.

Decouple Destination Position – The location (in meters) of the decouple location, relative to the start of the specified path (where the vehicle stops after decoupling). The position is expressed as a 32-bit single-precision floating point number using little-endian format. Zero position is defined as the midpoint of the vehicle at the beginning of the path. This value is only used when Decouple Path ID is not zero. Only available in MagneMover LITE systems.

Response

After receiving the command and verifying the command parameters, the HLC sends a Command Status (0xD0) message with either a “Command Accepted” response (0x00) or a “Command Rejected” response as appropriate (see HLC Status Codes). If the command is accepted, it is forwarded to the appropriate node controller for execution. If the command is rejected, the Command Status (0xD0) message includes a rejection code that shows the reason for rejection.

- **Vehicle Following** – If the auto-decouple feature is not configured in the vehicle follow order (Destination Path ID = 0), when the command completes the following events occur:

  1. When the vehicle is within the following distance from the specified vehicle, the **Caught Up** bit is set in the Extended Vehicle Status, Ver. 3 (0xDF 03 06) Vehicle Flags.

  2. The HLC sends a Command Status (0xD0) message with a "Command Status – Vehicle has caught up" response (0x81).

     NOTE: If the vehicle never catches up to the specified vehicle, this command never completes.

- **Vehicle Auto-decouple** – If the auto-decouple feature is configured in the vehicle follow order (Destination Path ID > 0), when the vehicle begins to brake for arrival at the destination path and position that is supplied in the follow order the following events occur:

  1. The HLC sends a Command Status (0xD0) message with a "Vehicle Decoupling" response (0x82).

  2. The **Vehicle Decoupled** bit is set in the Extended Vehicle Status, Ver. 3 (0xDF 03 06) Vehicle Flags).

  3. The order type for the command the vehicle is under changes from a Vehicle Follow Order (0xB7) to a Vehicle Position Order (0xB1).

  4. A "Command Complete" response (0x80) is sent for the Vehicle Position Order (0xB1) when the vehicle arrives at the defined decouple position. If the vehicle has followers during auto decoupling, it becomes the head of a new platoon for these followers.

  5. The **Vehicle Decoupled** bit is cleared in the Extended Vehicle Status, Ver. 3 (0xDF 03 06) Vehicle Flags).

See Also

Command Status (0xD0) on page 47

Get Extended Vehicle Status, Ver 3 (0xBF 03 06)

Type

Host Controller → HLC

Purpose

Requests the extended vehicle status, version 3 for the specified vehicle. If the vehicle ID specified is zero, the extended vehicle status for all vehicles is returned.

The extended vehicle status returns all status information that the Vehicle Status (0xD5) message returns and includes additional status information.

NOTE: The high-level controller and node controllers in the transport system must be in the operational state.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

Format

Offset	Item	Bytes	Range
0	Command Header	1	0xBF
1	Extension Type	1	0x03
2	Extension Subtype	1	0x06
3	Vehicle ID	2	0…65535

Arguments

Command Header – Fixed message type that identifies this message as a Host Extension command.

Extension Type – Fixed extension type that identifies this command as a Vehicle extension command.

Extension Subtype – Fixed extension subtype that identifies this command as a Get Extended Vehicle Status extension command.

Vehicle ID – The ID number of the vehicle for which status is requested. The ID must be either a nonzero 16-bit positive integer that references a vehicle that exists in the transport system or zero to request the status of all vehicles.

Response

After receiving the command and verifying the command parameters, the HLC sends one or more Extended Vehicle Status, Version 3 responses as appropriate (see Extended Vehicle Status, Ver. 3 (0xDF 03 06)).

If the specified vehicle does not exist, an Extended Vehicle Status, Ver. 3 (0xDF 03 06) response is returned with the **Vehicle Present** field set to zero, which indicates that no such vehicle exists.

See Also

Extended Vehicle Status, Ver. 3 (0xDF 03 06) on page 53

HLC to Host Controller Communications

This section describes the responses that are sent from the high-level controller (HLC) to the host controller as asynchronous responses from the transport system after performing an action. Table 2 lists the responses that are used with the vehicle platooning function.

Table 2: HLC to Host Controller Responses

Description and Value	Use	Page
Command Status (0xD0)	MM LITE, QS 100	47
Vehicle Status (0xD5)	MM LITE, QS 100	49
Extended Vehicle Status, Ver. 3 (0xDF 03 06)	MM LITE, QS 100	53

Command Status (0xD0)

Type

HLC → Host Controller

Purpose

Acknowledges the reception or rejection of a command, signals command execution failure, or signals command completion to the host controller.

NOTE: This response is sent as an asynchronous message with the appropriate command header when certain commands complete or fail.

Support

This response is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

Format

Offset	Item	Bytes	Range
0	Response Header	1	0xD0
1	Command	1	0xB0…0xB4, 0xB6, 0xB7…0xBC, 0xBF
2	Command Status	1	0x00…0x82
3	Command Data Detail	Varies	Varies for each command

Arguments

Response Header – Fixed message type that identifies this message as a Command Status response message.

Command – Header of the command that this message is acknowledging.

Command Header Codes

Cmd	Ext	Sub	Command	Valid Command Status Values
0xB1	–	–	Move Vehicle To Position	0x00, 0x01, 0x03, 0x04, 0x05, 0x06, 0x07, 0x0A, 0x0B, 0x0C, 0x0D, 0x10, 0x13, 0x41, 0x80
0XB7	–	–	Vehicle Follow Order	0x00, 0x01, 0x04, 0x0A, 0x0B, 0x0C, 0x0D, 0x10, 0x13, 0x1D, 0x41, 0x81, 0x82

Command Status – The status of the command that this message is acknowledging, see HLC Status Codes on page 83.

Command Data Detail – Variable amount of data that provides the details of the command that is specified in the **Command** field.

Cmd	Ext	Sub	Command Description	Command Data
0xB1	–	–	Move Vehicle To Position	Order Number (4 bytes)
				Vehicle ID (2 bytes)
				Position* (4 bytes)
				Path ID (2 bytes)
				Acceleration Limit* (4 bytes)
				Velocity Limit* (4 bytes)
				Flags and Direction (1 byte)
0xB7	–	–	Vehicle Follow Order	Order Number (4 bytes)
				Vehicle ID (2 bytes)
				Direction (1 byte)
				Follow Distance (4 bytes)
				Followed Vehicle ID (2 bytes)
				PID Set Index (1 byte)
				Catchup or Decouple Acceleration (4 bytes)
				Catchup or Decouple Velocity (4 bytes)
				Decouple Path ID (2 bytes)
				Decouple Position (4 bytes)

See Also

Move Vehicle To Position (0xB1) on page 37
Vehicle Follow Order (0xB7) on page 40

Vehicle Status (0xD5)

Type

HLC → Host Controller

Purpose

Reports the status of the vehicles that are specified in a Status Request (0xB5) command.

NOTE: When a Status Request (0xB5) is received for all vehicles, a separate status response is sent for each vehicle.

This response is sent as an asynchronous message when vehicles become obstructed and **Obstructed Status Notification** is selected in the Node Controller Configuration File. Enabling **Obstructed Status Notification** can result in many Vehicle Status messages being sent when vehicles are waiting in a queue.

Support

This response is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

Format

Offset	Item	Bytes	Range
0	Response Header	1	0xD5
1	Vehicle ID	2	1…65535
3	Vehicle Present	1	0…1
4	Path ID	2	1…65535
6	Destination Path ID	2	0…65535
8	Vehicle Position	4	0.0…41.0 (m, floating point)
12	Vehicle Velocity	4	-5.0…+5.0 (m/s, floating point)
16	Vehicle Command Type	1	0x00, 0xB0, 0xB1, 0xB7
17	Vehicle Flags	1	Bits 0…7 are Flags
18	Commanded Position or Station ID	4	0.0…41.0 (m, floating point)
		2	1…2048

Arguments

Response Header – Fixed message type that identifies this message as a Vehicle Status response.

Vehicle ID – The ID number of the vehicle for which status is being reported.

Vehicle Present – Equal to 0 if the vehicle is not present (remainder of data in message is undefined) or 1 if the vehicle is present.

Path ID – The ID number of the path where the vehicle is located.

Destination Path ID – The ID number of the destination path for the vehicle based on command type.

Command	Description
0x00, 0xB7	Zero if the vehicle has never moved or the vehicle is under a Vehicle Follow Order (0xB7).
0xB0, 0xB1	The ID number of the path where the vehicle is headed. Equal to the ID of the path where the vehicle is located if motion has completed.

Vehicle Position – The last reported position (in meters) of the vehicle, relative to the start of the specified path. Vehicle position is expressed as a 32-bit single-precision floating point number using little-endian format. Zero position is defined as the midpoint of the vehicle at the beginning of the path.

Vehicle Velocity – The last reported velocity (in m/s) of the vehicle on the specified path. Vehicle velocity is expressed as a 32-bit single-precision floating point number using little-endian format.

Vehicle Command Type – Indicates the type of command currently active for the vehicle.

Value	Description
0x00	No command is in progress.
0xB0	Move vehicle to station.
0xB1	Move vehicle to position.
0xB7	Vehicle Follow Order.

Vehicle Flags – Flags indicating vehicle status.

Bit	Description
0	**Vehicle Signal Status:** 0 – The motor does not detect the vehicle. 1 – The motor currently in charge of the vehicle detects the vehicle.
1	**Obstructed Status:** 0 – The vehicle is not obstructed and able to acquire permission to move further. 1 – The vehicle is obstructed and unable to acquire permission to move further. Obstructions are due to either a vehicle in the way, a hardware fault, or motion is suspended. Vehicles in the way occur during normal operation when vehicles are in a queue or when a vehicle is in a switch, which keeps another vehicle from entering the switch. If after three consecutive reports the obstructed bit is still set, check for an actual obstruction.

Bit	Description
2	**Hindered Status:** 0 – The vehicle is making progress on its current move profile. 1 – The vehicle is not making progress towards the position it has most recently been granted permission to move to. Lack of progress can happen when a vehicle is held back by some external force including a foreign object jamming a vehicle on the guideway or if the vehicle is not in motion while under sync control. The motor continues to apply force on the vehicle to try to move it indefinitely.
3…6	**PID Loop Set Index:** 0 – Use user-defined PID set 0 – Unloaded PID values. 1 – Use user-defined PID set 1 – Loaded PID values. 2 – Use user-defined PID set 2. 3 – Use user-defined PID set 3. 4 – Use user-defined PID set 4. 5 – Use user-defined PID set 5. 6 – Use user-defined PID set 6. 7 – Use user-defined PID set 7. 8 – Use user-defined PID set 8. 9 – Use user-defined PID set 9. 10 – Use user-defined PID set 10. 11 – Use user-defined PID set 11. 12 – Use user-defined PID set 12. 13 – Use user-defined PID set 13. 14 – Use user-defined PID set 14. 15 – Use user-defined PID set 15 – Startup PID values. This PID set is automatically used during startup. If it is not defined, PID set 0 is scaled by 25% and used for startup.
7	**Suspect Status (MM LITE, QS 100 only):** 0 – The vehicle is not suspect and is free to move. 1 – The vehicle is suspect. The motor has detected that the vehicle has been manually moved out of the control region. Typically, control of the vehicle cannot be regained. Even if control is regained, the vehicle continues to be suspect until a Vehicle Command (0xBF 03 04) is issued to clear the Suspect Bit. Or, the vehicle must be deleted using a Delete Vehicle (0xB9) command and each path that is reporting an unlocated vehicle fault must be restarted.

Commanded Position or Station – The commanded destination for the vehicle.

Cmd	Command Type	Description
0x00	No command or command complete	4-byte value that is set to zero
0xB0	Vehicle station order	2-byte value that is the commanded Station (1…2048).

Cmd	Command Type	Description
0xB1	Vehicle position order	4-byte value that is the commanded destination on the specified path (0.0…41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).
0xB7	Vehicle follow order	4-byte value that is the distance to the vehicle being followed (0…41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).

See Also

Move Vehicle To Position (0xB1) on page 37
Vehicle Follow Order (0xB7) on page 40

Extended Vehicle Status, Ver. 3 (0xDF 03 06)

Type

HLC → Host Controller

Purpose

Reports the status of the vehicles that are specified in a Get Extended Vehicle Status, Ver 3 (0xBF 03 06) command.

NOTE: When a Get Extended Vehicle Status, Ver 3 (0xBF 03 06) is received for all vehicles, a separate status response is sent for each vehicle.

The Extended Vehicle Status response returns all status information that the Vehicle Status (0xD5) message returns and includes additional status information.

Support

This response is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

Format

Offset	Item	Bytes	Range
0	Response Header	1	0xDF
1	Extension Type	1	0x03
2	Extension Subtype	1	0x06
3	Vehicle ID	2	1…65535
5	Vehicle Present	1	0…1
6	Path ID	2	1…65535
8	Destination Path ID	2	0…65535
10	Vehicle Position	4	-41.0…+41.0 (m, floating point)
14	Vehicle Velocity	4	-41.0…+41.0 (m/s, floating point)
18	Vehicle Command Type	1	0x00, 0xB0, 0xB1, 0xB7
19	Vehicle Flags	2	Bits 0…15 are Flags
21	Commanded Position	4	-41.0…+41.0 (m, floating point)
25	Current Target	4	-41.0…+41.0 (m, floating point)
29	Followed Vehicle ID	2	Command types 0x00, 0xB0, 0xB1: 0 Command type 0xB7: 1…65535
31	Time Since Last Report	4	Seconds (floating point)
35	Destination Station ID	2	0…2048

Offset	Item	Bytes	Range
37	Reported PID Set Index	1	0…15
38	Ordered PID Set Index	1	0…15
39	Ordered Acceleration Limit	4	Float in m/s^2 (limits system dependent)
43	Ordered Velocity Limit	4	Float in m/s (limits system dependent)
47	Active Move to Station Offset	4	-41.0…+41.0 (m, floating point)

Arguments

Response Header – Fixed message type that identifies this message as a Host Extension response.

Extension Type – Fixed extension type that identifies this message as a Vehicle extension response.

Extension Subtype – Fixed extension subtype that identifies this message as an Extended Vehicle Status, Version 3 extension message.

Vehicle ID – The ID number of the vehicle for which status is being reported.

Vehicle Present – Equal to 0 if the vehicle is not present (remainder of data in message is invalid) or 1 if the vehicle is present.

Path ID – The ID number of the path where the vehicle is located.

Destination Path ID – The ID of the path where the vehicle is heading when the vehicle is under a motion command. Equal to the ID of the path where the vehicle is located if motion has completed. Equal to 0 if the vehicle has never moved or if the vehicle is under a Vehicle Follow Order (0xB7).

Vehicle Position – The last reported position (in meters) of the vehicle, relative to the start of the specified path. Vehicle position is expressed as a 32-bit single-precision floating point number using little-endian format. Zero position is defined as the midpoint of the vehicle at the beginning of the path.

Vehicle Velocity – The last reported velocity (in m/s) of the vehicle on the specified path. Vehicle velocity is expressed as a 32-bit single-precision floating point number using little-endian format.

Vehicle Command Type – Indicates the type of command currently active for the vehicle.

Value	Description
0x00	No command is in progress.
0xB0	Move vehicle to station.
0xB1	Move vehicle to position.
0xB7	Vehicle Follow Order

Vehicle Flags – Flags indicating vehicle status.

Bit	Description
0	**Vehicle Signal Status:** 0 – The motor does not detect the vehicle. 1 – The motor currently in charge of the vehicle detects the vehicle.
1	**Obstructed Status:** 0 – The vehicle is not obstructed and able to acquire permission to move further. 1 – The vehicle is obstructed and unable to acquire permission to move further. Obstructions are due to either a vehicle in the way, a hardware fault, or motion is suspended. Vehicles in the way occur during normal operation when vehicles are in a queue or when a vehicle is in a switch, which keeps another vehicle from entering the switch. If after three consecutive reports the obstructed bit is still set, check for an actual obstruction.
2	**Hindered Status:** 0 – The vehicle is making progress on its current move profile. 1 – The vehicle is not making progress towards the position it has most recently been granted permission to move to. Lack of progress can happen when a vehicle is held back by some external force including a foreign object jamming a vehicle on the guideway or if the vehicle is not in motion while under sync control. The motor continues to apply force on the vehicle to try to move it indefinitely.
3	**Stall Status:** 0 – The vehicle is not stalled. 1 – The vehicle has stalled. A stalled vehicle is defined as a vehicle that has jammed and the motor in control of the vehicle has had to reduce power to keep from overheating.
4	**Locked Status:** 0 – The vehicle is not locked. 1 – The vehicle is locked. Until the vehicle is unlocked, the system rejects execution of move commands for the vehicle.
5	**Locate Completed Status:** 0 – Vehicle locate is not completed (startup is in progress). 1 – Vehicle locate has completed.
6	Reserved/unused
7	**Suspect Status (MM LITE, QS 100 only):** 0 – The vehicle is not suspect and is free to move. 1 – The vehicle is suspect. The motor has detected that the vehicle has been manually moved out of the control region. Typically, control of the vehicle cannot be regained. Even if control is regained, the vehicle continues to be suspect until a Vehicle Command (0xBF 03 04) is issued to clear the Suspect Bit.

Bit	Description
8	**Following Upstream:** 0 – This vehicle is not following another vehicle. 1 – This vehicle is following the vehicle that is specified in the **Followed Vehicle ID** field and that vehicle is upstream from this vehicle.
9	**Following Downstream:** 0 – This vehicle is not following another vehicle. 1 – This vehicle is following the vehicle that is specified in the **Followed Vehicle ID** field and that vehicle is downstream from this vehicle.
10	**Caught Up:** 0 – This vehicle is not following another vehicle. 1 – The motor controller is reporting that this following vehicle has caught up to the vehicle it is following and is at the requested following distance.
11	**Profile Stale Error:** 0 – This vehicle is not following another vehicle. 1 – The motor controller is reporting that this following vehicle has not received profile data from the vehicle it is following, in the expected time (~10 ms). The vehicle slows to a stop. If the profile data is restored, the vehicle catches up to its leader using the catchup acceleration and velocity.
12	**Excessive Following Error:** 0 – This vehicle is not following another vehicle. 1 – The motor controller is reporting that this following vehicle is too far from where it is expected to be (possibly stopped receiving following profile data).
13	**Vehicle Decoupled:** 0 – A platoon follower vehicle not in the decoupling state, or a vehicle not in a platoon. 1 – A platoon follower vehicle is decoupling from the vehicle it is following. This bit stays set from when the MC reports decoupling, until the decoupled vehicle reaches its new destination under the 0xB1 position order that was converted from the 0xB7 follow order.
14, 15	Reserved/unused

Commanded Position – The commanded destination for the vehicle.

Cmd	Command Type	Description
0x00	No command or command complete	4-byte value that is set to zero
0xB0	Vehicle station order	4-byte value that is the commanded destination on the specified path (-41.0…+41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format). For a 0xB0 command, this position is the location of the station.
0xB1	Vehicle position order	
0xB7	Vehicle follow order	4-byte value that is the distance to the vehicle being followed (0…41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).

Current Target – The position that the vehicle has permission to move to (in meters) as measured from the upstream end of the current path. The target is expressed as a 32-bit single-precision floating point number using little-endian format. If the vehicle has completed its current order and is stopped, this value is equal to the position of the vehicle.

If the vehicle is under a Vehicle Follow Order (0xB7), this value is the Follow Distance (the distance (in meters) center-to-center that this vehicle maintains behind the vehicle being followed.

Followed Vehicle ID – The ID number of the vehicle that this vehicle is following when this vehicle is under a vehicle follow order and is part of a platoon (vehicle command type is 0xB7). Equal to 0 if this vehicle is not under a vehicle follow order (vehicle command type is 0x00, 0xB0, or 0xB1).

Time Since Last Report – The time in seconds since the last vehicle status for this vehicle was received from the motor. The time is expressed as a 32-bit single-precision floating point number using little-endian format.

Destination Station ID – The ID number of the station the vehicle is commanded to when under a Move to Station command. This field is only valid when the Vehicle Command Type field is 0xB0. If Vehicle Command Type is 0x00, 0xB1, or 0xB7, this value is zero.

Reported PID Set Index – The PID set currently in use.

Table 3: PID Control Loop Sets

Value	Description
0	Use user-defined PID set 0 – Unloaded PID values.
1	Use user-defined PID set 1 – Loaded PID values.
2	Use user-defined PID set 2.
3	Use user-defined PID set 3.
4	Use user-defined PID set 4.
5	Use user-defined PID set 5.
6	Use user-defined PID set 6.
7	Use user-defined PID set 7.
8	Use user-defined PID set 8.
9	Use user-defined PID set 9.
10	Use user-defined PID set 10.
11	Use user-defined PID set 11.
12	Use user-defined PID set 12.
13	Use user-defined PID set 13.
14	Use user-defined PID set 14.
15	Use user-defined PID set 15 – Startup PID values. This PID set is automatically used during startup. If it is not defined, PID set 0 is scaled by 25% and used for startup.

Ordered PID Set Index – The commanded PID Control Loop set for vehicle movement during the movement order when a movement order is in progress. This field is zero if the vehicle has not been given an order since startup. See Table 3: PID Control Loop Sets.

Ordered Acceleration Limit – The last acceleration order from the host (in m/s^2) for the vehicle on the specified path, expressed as a 32-bit single-precision floating point number using little-endian format when a movement order is in progress. This field is zero if the vehicle has not been given an order since startup.

Ordered Velocity Limit – The last velocity order from the host (in m/s) for the vehicle on the specified path, expressed as a 32-bit single-precision floating point number using little-endian format when a movement order is in progress. This field is zero if the vehicle has not been given an order since startup.

Active Move to Station Offset – Commanded offset from the destination station relative to the position of the station expressed as a 32-bit single-precision floating point number using little-endian format. Offsets downstream from a station are positive values, offsets upstream from a station are negative values.

NOTE: If the move to station was commanded using a Move Vehicle To Station (0xB0) command, this field returns the configured offset. If the move to station was commanded using an MMI_vehicle_station_order tag, this field always returns 0.

See Also

Get Extended Vehicle Status, Ver 3 (0xBF 03 06) on page 44

Host Controller EtherNet/IP Communication Protocol User Manual Addendum

This section is an addendum to the *Host Controller EtherNet/IP Communication Protocol User Manual*, MMI-UM004, which describes the tags that support vehicle platooning. The additions to the protocol for communication between the high-level controller (HLC) and a host controller that is equipped with an EtherNet/IP™ interface are described.

Support for the changes that are described in this section starts with the following NC Software Image versions:

- MM LITE – 15.11.20.
- QuickStick – 15.10.23.

Host Controller to HLC Communications

This section describes Explicit Message types and Tag memory formats that traverse TCP connections between the host controller and the HLC.

Command (explicit) messages (listed in Table 4) are sent from the host controller to the HLC to request the HLC to perform an action. These requests are responded to by the HLC by routing the command to the appropriate node controller for completion and updating the corresponding response tags in the host controller memory (listed in Table 9).

Table 4: Host Controller to HLC Explicit Message Requests

Message Name	Message Element	Use	Page
MMI_vehicle_position_order	udt_MMI_vehicle_follow_order	MM LITE, QS 100	60
MMI_vehicle_follow_order	udt_MMI_vehicle_follow_order	MM LITE, QS 100	64

MMI_vehicle_position_order

Type

Host Controller → HLC

Purpose

Moves the specified vehicle to a specified position, relative to the start of a path.

Each Vehicle Position Order can be monitored via its corresponding udt_MMI_vehicle_order_status entry in the MMI_vehicle_order_status tag. The status of the vehicle can be monitored via its corresponding entry in the MMI_vehicle_status or MMI_extended_vehicle_status tags.

NOTE: The high-level controller, node controllers, and the path the vehicle resides on must be in the operational state. If the path where the vehicle is located is not in an operational state, the command is rejected. If any component required to execute the command is not in operational state, the command does not complete.

If the commanded acceleration or velocity values are higher than the limit that is defined in the Node Controller Configuration File, the command is rejected.

If the commanded velocity value is higher than the value for a specific motor, the motor velocity value overrides the commanded value while the vehicle is on that motor.

If a move command is issued to a vehicle already in motion and the command has a lower acceleration than the previous command to that vehicle, the command is rejected.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

UDT Field Descriptions

Table 5: UDT Configuration for udt_MMI_vehicle_position_order

Field Name	Data Type	Style	Range
vehicle_id	INT	Decimal	1…65535
path_id	INT	Decimal	1…65535
position	REAL	Float	-41.0…+41.0 (m, floating point)
velocity_limit	REAL	Float	0.0…5.0 (m/s, floating point)
acceleration_limit	REAL	Float	0.0…60.0 (m/s^2, floating point)
order_number	DINT	Decimal	0…65535

Table 5: UDT Configuration for udt_MMI_vehicle_position_order

Field Name	Data Type	Style	Range
flags_and_direction	SINT	Binary	Bits 0…3 are direction values Bits 4…7 are Flags
active_flag	SINT	Hex	0x00, 0x01

vehicle_id – The ID of the vehicle to move. The ID must be a nonzero 16-bit positive integer that references a vehicle that exists in the transport system.

path_id – The ID of the path where the desired position is located. The ID must be a nonzero 16-bit positive integer that references a path that exists in the configuration.

position – The destination (in meters) of the vehicle, relative to the start of the specified path (expressed as a 32-bit single precision floating point number). Zero position is defined as the midpoint of the vehicle at the beginning of the path. When decoupling a vehicle from a stationary platoon, the position that is specified must be the current target for the vehicle as returned through the Extended Vehicle Status. Negative positions can only be used with node types that support movement to a position before the beginning of a path.

velocity_limit – A positive number (in m/s) that defines the maximum velocity for vehicle motion during the move command (expressed as a 32-bit single precision floating point number). This value is checked against the limit that is defined in the Node Controller Configuration File, and if higher the command is rejected. If this value is within the limit that is defined in the Node Controller Configuration File but higher than the value for any specific motor in the vehicle's route, the value from the motor overrides the command value while the vehicle is on that motor.

This value is system-dependent, see the *Host Controller EtherNet/IP Communication Protocol User Manual*, MMI-UM004.

acceleration_limit – A positive number (in m/s^2) that defines the maximum acceleration and deceleration rate for the move command (expressed as a 32-bit single precision floating point number). This value is checked against the limit that is defined in the Node Controller Configuration File, and if higher the command is rejected. If this value is within the limit that is defined in the Node Controller Configuration File but higher than the value for any specific motor in the vehicle's route, the value from the motor is used and the command is updated to that lower value.

This value is system-dependent, see the *Host Controller EtherNet/IP Communication Protocol User Manual*, MMI-UM004.

order_number – Host controller derived unique order number, which is used to confirm that the HLC has received the command and to determine command status. This counter provides a convenient handshake mechanism to determine order status by matching the counter with the fields in the MMI_vehicle_order_status array.

MagneMotion suggests creating a scratch DINT tag for use as a counter. The scratch DINT tag can then be incremented and copied to order_number each time a new vehicle command

is issued so the new command can be tracked against the various count fields that are described in MMI_vehicle_order_status.

flags_and_direction – Bits 0…3 indicate the direction to move the vehicle. Bits 4…7 indicate the PID set to use when executing the move command.

Bit	Description
0…3	Direction: 0 – BiDirectional – If the destination is on the same path the vehicle resides on, the vehicle can move either direction as required to get to the destination in the shortest distance. If the destination is on a path other than the path where the vehicle is located, the forward direction takes precedence for a transport system that is a closed-loop. 1 – Forward – Force the vehicle to move forward (downstream) only, useful to implement a unidirectional loop. If the destination is not reachable, the command fails and the vehicle is not moved. 2 – Backward – Force the vehicle to move backward (upstream) only, useful to implement a unidirectional loop in the backwards direction. If the destination is not reachable, the command fails and the vehicle is not moved. 3…15 – Reserved
4…7	PID Loop Set Index: Use user-defined PID set 0 – Unloaded PID values. Use user-defined PID set 1 – Loaded PID values. Use user-defined PID set 2. Use user-defined PID set 3. Use user-defined PID set 4. Use user-defined PID set 5. Use user-defined PID set 6. Use user-defined PID set 7. Use user-defined PID set 8. Use user-defined PID set 9. Use user-defined PID set 10. Use user-defined PID set 11. Use user-defined PID set 12. Use user-defined PID set 13. Use user-defined PID set 14. Use user-defined PID set 15 – Startup PID values. This PID set is automatically used during startup. If it is not defined, PID set 0 is scaled by 25% and used for startup.

active_flag – This flag is used in configurations that use a static table of udt_MMI_vehicle_position_order entries. The same table is sent every time with the active_flag for each entry set or cleared to direct the HLC whether to process the entry or not.

Value	Description
0x00	The HLC skips the entry.
0x01	The HLC processes the entry.

Message Configuration

Message Element	Description
Source Element	The source element vehicle_position_order shown is an array tag of type udt_MMI_vehicle_position_order. It can be an array or a single tag with Number Of Elements set to 1.
Number of Elements	Min: 1 Max: Max vehicles + 1 Specify the number of MMI_vehicle_position_order tags to issue (one per vehicle) in the transport system with one message.
Destination Element	MMI_vehicle_position_order

Response

After receiving the command, the HLC verifies the command parameters and updates MMI_vehicle_order_status as appropriate. If the command is accepted, it is forwarded to the appropriate node controller for execution. Once the command completes the HLC updates MMI_vehicle_order_status to indicate completion.

See Also

MMI_vehicle_order_status on page 70
MMI_vehicle_status on page 73
MMI_extended_vehicle_status on page 77

MMI_vehicle_follow_order

Type

Host Controller → HLC

Purpose

Orders a vehicle to catch up to and follow another vehicle using a specified profile and then maintain a specified following distance. Once the vehicle is positioned at the commanded following distance, the system maintains the following distance indefinitely, matching the movement profile of the followed vehicle. The vehicle continues to follow the specified vehicle at the specified distance until the follow order is overridden with an MMI_vehicle_position_order, an MMI_vehicle_station_order, or a new MMI_vehicle_follow_order.

There is an auto decouple feature, where a decouple path ID and position are defined. When the platoon with the following vehicle approaches that position, the vehicle with that position specified as part of its follow order decouples from the platoon, slows down, and stops at that position.

Each Vehicle Follow order can be monitored via the corresponding udt_MMI_vehicle_order_status entry in the MMI_vehicle_order_status tag. The status of the vehicle can be monitored via its corresponding entry in the MMI_extended_vehicle_status or MMI_vehicle_status tags.

NOTE: The high-level controller, node controllers, and the path the vehicle resides on in the transport system must be in the operational state. If the path where the vehicle is located is not in an operational state, the command is rejected. If any component required to execute the command is not in operational state, the command does not complete.

Vehicles must be stopped to couple them into a platoon.

Platoons must be stopped to decouple vehicles unless the command coupling a vehicle to the platoon specified a decouple destination for that vehicle.

If the acceleration or velocity values in a move command are higher than the limit that is defined in the Node Controller Configuration File, the command is rejected.

If the commanded velocity value is higher than the value for a specific motor, the motor overrides that value while the vehicle is on that motor.

If a move command is issued to a vehicle already in motion and the command has a lower acceleration than the previous command to that vehicle, the command is rejected.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

UDT Field Descriptions

Table 6: UDT Configuration for udt_MMI_vehicle_follow_order

Field Name	Data Type	Style	Range
vehicle_id	INT	Decimal	1…65535
follow_distance	REAL	Float	0…41.0 (m, floating point)
followed_vehicle_id	INT	Decimal	1…65535
pid_set_index	SINT	Decimal	0…15
catchup_or_decouple_acceleration	REAL	Float	0.0…60.0 (m/s^2, floating point)
catchup_or_decouple_velocity	REAL	Float	0.0…5.0 (m/s, floating point)
decouple_destination_path_id (MM LITE only):	INT	Decimal	0…65535
decouple_destination_position (MM LITE only):	REAL	Float	0.0…41.0 (m, floating point)
order_number	DINT	Hex	0x00000000…0xFFFFFFFF
direction	SINT	Decimal	1, 2
active_flag	SINT	Hex	0x00, 0x01

vehicle_id – The ID number of the vehicle executing this vehicle follow order. This vehicle follows the vehicle that **followed_vehicle_id** specifies. The ID must be a nonzero 16-bit positive integer that references a vehicle that exists in the transport system.

follow_distance – The distance (in meters) center-to-center that this vehicle maintains behind the vehicle being followed (specified by **followed_vehicle_id**) in the **direction** that is specified, once the pair of vehicles are coupled into a platoon.

NOTE: When coupling to a platoon, the vehicle must be within 30 mm of this distance before sending the vehicle follow order.

followed_vehicle_id – The ID number of the vehicle being followed by this vehicle (executing this vehicle follow order). The followed_vehicle_id must be a nonzero 16-bit integer.

pid_set_index – The PID set to use when executing the command (see flags_and_direction in MMI_vehicle_position_order).

catchup_or_decouple_acceleration –The maximum acceleration/deceleration for the vehicle (in m/s^2) to use when catching up to the vehicle to follow or to use when auto-decoupling from the vehicle it is following (system dependent). This value is checked against the system limit that is defined in the Node Controller Configuration File, and if higher, the command is updated to the value from the Node Controller Configuration File. Only available in MagneMover LITE systems.

NOTE: Currently only used when decoupling as vehicles must not be moving when creating a platoon, but it must always be defined.

catchup_or_decouple_velocity – The maximum velocity for the vehicle (in m/s) to use when catching up to the vehicle to follow or to use when auto-decoupling from the vehicle it is following (system dependent). This value is checked against the system limit that is defined in the Node Controller Configuration File, and if higher, the command is updated to the value from the Node Controller Configuration File. Only available in MagneMover LITE systems.

NOTE: Currently only used when decoupling as vehicles must not be moving when creating a platoon, but it must always be defined.

decouple_destination_path_id – The ID number of the path where the decoupling position is located (where the vehicle stops after decoupling). The ID must be a nonzero 16-bit positive integer that references a path that exists in the configuration. If the value is zero, the decoupling feature is disabled. Only available in MagneMover LITE systems.

decouple_destination_position – The location (in meters) of the decouple location, relative to the start of the specified path (where the vehicle stops after decoupling). The position is expressed as a 32-bit single-precision floating point number using little-endian format. Zero position is defined as the midpoint of the vehicle at the beginning of the path. This value is only used when Decouple Path ID is not zero. Only available in MagneMover LITE systems.

order_number – Host controller derived unique order number, which is used to confirm that the HLC has received the command and to determine command status. This counter provides a convenient handshake mechanism to determine order status by matching the counter with the fields in the MMI_vehicle_order_status array.

MagneMotion suggests creating a scratch DINT tag for use as a counter. The scratch DINT tag can then be incremented and copied to order_number each time a new vehicle command is issued. This counter allows the new command to be tracked against the various count fields that are described in MMI_vehicle_order_status.

direction – Direction on the track relative to this vehicle where the **followed_vehicle_id** is located:

Value	Description
1	Vehicle to follow is downstream of this vehicle (motion is downstream).
2	Vehicle to follow is upstream of this vehicle (motion is upstream).

active_flag – This flag is used in configurations that use a static table of udt_MMI_vehicle_follow_order entries. The same table is sent every time with the active_flag for each entry set or cleared to direct the HLC whether to process the entry or not.

Value	Description
0x00	The HLC skips the entry.
0x01	The HLC processes the entry.

Message Configuration

Table 7: Message Configuration for MMI_vehicle_follow_order

Source Element	The source element vehicle_follow_order shown is an array tag of type udt_MMI_vehicle_follow_order. It can be an array or a single tag with Number Of Elements set to 1.
Number of Elements	Min: 1 Max: Max vehicles + 1 Specify the number of MMI_vehicle_follow_order tags to issue (one per vehicle) in the transport system with one message.
Destination Element	MMI_vehicle_follow_order

Response

After receiving the command, the HLC verifies the command parameters and updates the MMI_vehicle_order_status tag with either a "Command Accepted" response (0x00) or a "Command Rejected" response as appropriate (see HLC Status Codes). If the command is accepted, it is forwarded to the appropriate node controller for execution. If the command is rejected, the MMI_vehicle_order_status tag includes a rejection code that shows the reason for rejection.

- **Vehicle Following** – If the auto-decouple feature is not configured in the vehicle follow order (Destination Path ID = 0), when the command completes the following events occur:

  1. When the vehicle is within the following distance from the specified vehicle, the **Caught Up** bit is set in the MMI_extended_vehicle_status Vehicle Flags.

  2. The HLC updates the MMI_vehicle_order_status tag with a "Command Status – Vehicle has caught up" response (0x81).

  NOTE: If the vehicle never catches up to the specified vehicle, this command never completes.

- **Vehicle Auto-decouple** – If the auto-decouple feature is configured in the vehicle follow order (Destination Path ID > 0), when the vehicle begins to brake for arrival at the destination path and position that is supplied in the follow order the following events occur:

  1. The HLC updates the MMI_vehicle_order_status tag with a "Vehicle Decoupling" response (0x82).

  2. The **Vehicle Decoupled** bit is set in the MMI_extended_vehicle_status Vehicle Flags.

  3. The order type for the command the vehicle is under changes from a Vehicle Follow Order (0xB7) to a Vehicle Position Order (0xB1).

  4. The HLC updates the MMI_vehicle_order_status tag with a "Command Complete" response (0x80) for the Vehicle Position Order when the vehicle arrives at the defined

decouple position. If the vehicle has followers during auto decoupling, it becomes the head of a new platoon for these followers.

5. The **Vehicle Decoupled** bit is cleared in the MMI_extended_vehicle_status Vehicle Flags.

See Also

MMI_vehicle_order_status on page 70
MMI_vehicle_status on page 73
MMI_extended_vehicle_status on page 77

HLC to Host Controller Communications

This section describes the Status and Response Memory Tags that the HLC updates to interoperate with the host controller.

The HLC updates the Response Tags (listed in Table 8) only in response to specific commands from the host controller issued via explicit messages. The HLC updates the Status Tags (listed in Table 9) on state change or when the EtherNet/IP link between the HLC and the host controller has been re-established after going down. Status Tags are informational for the host controller and provide status about the vehicle platooning functions.

Table 8: HLC to Host Controller Response Memory Tags

Message Name	Message Element	Use	Page
MMI_vehicle_order_status	udt_MMI_vehicle_order_status	MM LITE, QS 100	70

Table 9: HLC to Host Controller Status Memory Tags

Message Name	Message Element	Use	Page
MMI_vehicle_status	udt_MMI_vehicle_status	MM LITE, QS 100	73
MMI_extended_vehicle_status	udt_MMI_extended_vehicle_status	MM LITE, QS 100	77

MMI_vehicle_order_status

Type

HLC → Host Controller

Purpose

Reports the status of all vehicle commands in the transport system. This tag is a one-dimensional array of type udt_MMI_vehicle_order_status indexed by vehicle ID. Index 0 is not used since a vehicle ID of 0 in the transport system is invalid and not used.

This array must be sized the same as the vehicle status array being used. When using MMI_vehicle_status the minimum array size is 225, see MMI_vehicle_status on page 73 for more details. When using MMI_extended_vehicle_status the minimum array size is 145, see MMI_extended_vehicle_status on page 77 for more details. This value is system-dependent, see the *Host Controller EtherNet/IP Communication Protocol User Manual*, MMI-UM004, for the maximum number of vehicles per system.

The high-level controller updates the MMI_vehicle_order_status array only in response to an MMI_vehicle_position_order, an MMI_vehicle_station_order, an MMI_vehicle_delete_order, or an MMI_vehicle_follow_order sent from the host controller to the HLC.

This tag array can be used to handshake vehicle commands so the host controller logic can know that the HLC received a vehicle command, accepted or rejected it, and that the command completed (the vehicle has arrived at the commanded destination). For transport systems with a few vehicles and many Stations, or no Stations at all, this tag array is the suggested method for handshaking vehicle commands between the host controller and the HLC. For transport systems with many vehicles and a smaller number of Stations, the MMI_station_arrivals tag array can be used as an alternative form of vehicle command handshaking.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

UDT Field Descriptions

Table 10: UDT Configuration for udt_MMI_vehicle_order_status

Field Name	Data Type	Style	Range
last_order_number_received	DINT	Hex	0x00…0xFFFFFFFF
last_order_number_accepted	DINT	Hex	0x00…0xFFFFFFFF
last_order_type_received	SINT	Hex	0xB0, 0xB1, 0xB7, 0xB9

Table 10: UDT Configuration for udt_MMI_vehicle_order_status

Field Name	Data Type	Style	Range
last_command_received_status	SINT	Hex	**Vehicle Station Order (0xB0)** 0x00, 0x01, 0x02, 0x04, 0x0A, 0x0D, 0x10, 0x13, 0x41, 0x80 **Vehicle Position Order (0xB1)** 0x00, 0x01, 0x03, 0x04, 0x0A, 0x0D, 0x10, 0x13, 0x41, 0x80 **Vehicle Follow Order (0xB7)** 0x00, 0x01, 0x04, 0x0A, 0x0B, 0x0C, 0x0D, 0x10, 0x13, 0x1D, 0x41, 0x81, 0x82 **Vehicle Delete Order (0xB9)** 0x00, 0x01, 0x0C, 0x41
last_order_type_accepted	SINT	Hex	0xB0, 0xB1, 0xB7, 0xB9
last_order_accepted_is_complete	SINT	Hex	0x00…0x02 (lower 4 bits)
		Binary	Bits 4…7 are Flags (upper 4 bits)

last_order_number_received – The HLC writes the **order_number** field of the most recently received vehicle command for the corresponding vehicle ID to this field. The host controller logic can use this field for a specific vehicle to determine if the HLC received a command. The host controller can then optionally retry the command after a timeout (nominally 5 seconds) if the **order_number** in the original command does not match this field, which indicates the HLC never received the command.

last_order_number_accepted – The HLC writes the **order_number** field of the most recently accepted vehicle command for the corresponding vehicle ID to this field when the node controller has accepted the command. The host controller logic can use this field to determine if a command received by the HLC (**last_order_number_received** = **order_number** for the most recently issued command) has been accepted or not. When this field matches **last_order_number_received**, the host controller knows that the HLC has accepted the command. This acceptance of the command also implies the HLC has written a 0x00 to the **last_order_received_status** field.

last_order_type_received – The HLC writes the command type of the most recently received vehicle command for the corresponding vehicle ID to this field.

Value	Description
0xB0	Vehicle Station Order
0xB1	Vehicle Position Order
0xB7	Vehicle Follow Order
0xB9	Vehicle Delete Order

last_order_received_status – The HLC writes an acceptance or rejection status to this field upon receiving a new vehicle command from the host controller. See HLC Status Codes on page 83 for the meaning of the failure codes in the specified range.

last_order_type_accepted – The HLC writes the command type of the most recently accepted vehicle command for the corresponding vehicle ID to this field.

Value	Description
0xB0	Vehicle Station Order
0xB1	Vehicle Position Order
0xB7	Vehicle Follow Order
0xB9	Vehicle Delete Order

last_order_accepted_is_complete – The HLC writes the order completion status of the most recently accepted vehicle command for the corresponding vehicle ID to the lower four bits of this field as a value. The upper four bits of this field are used as flags.

Value (lower 4 bits)	Description
0x0	The last_order_number_accepted is still in progress.
0x1	The last_order_number_accepted is complete. The vehicle has completed its commanded motion or has been deleted.
0x2	The last_order_number_accepted failed.
0x3…0xF	Reserved

Bit (upper 4 bits)	Description
4	Auto-decouple not complete and vehicle arrived.
5…7	Reserved

Message Configuration

Table 11: Message Configuration for MMI_vehicle_order_status

Source Element	vehicle_order_status Array tag of type udt_MMI_vehicle_order_status. Array index corresponds to HLC vehicle ID. Since vehicle ID 0 is invalid, array element [0] is not used.
Number of Elements	Min: MMI_vehicle_status or MMI_extended_vehicle_status) Max: Max vehicles + 1 Specify the number of MMI_vehicle_order_status tags to read (one per vehicle) with one message.
Destination Element	MMI_vehicle_order_status

See Also

MMI_vehicle_position_order on page 60
MMI_vehicle_follow_order on page 64

MMI_vehicle_status

Type

HLC → Host Controller

Purpose

Reports the status of all vehicles in the transport system. This tag is a one-dimensional array of type udt_MMI_vehicle_status indexed by vehicle ID. Index 0 is not used since a vehicle ID of 0 in the transport system is invalid and not used. A vehicle entry is in use when the **path_id** field in the entry is nonzero. A vehicle entry is not in use when the **path_id** field in the entry is zero (that is, no vehicle exists at that index).

This array must be sized to the maximum number of vehicles in the transport system plus one (vehicle 0 is invalid). The **PLC Max Vehicle ID** setting establishes the maximum vehicle ID that the HLC reports vehicle status for to the MMI_vehicle_status tag in the host controller memory. PLC Max Vehicle ID is defined on the **PLC EtherNet/IP Settings** page in the Configurator when creating the Node Controller Configuration File. See either the *MagneMover LITE Configurator User Manual*, MMI-UM008, or the *QuickStick Configurator User Manual*, MMI-UM009. The HLC updates the MMI_ vehicle_status tag in the host controller memory based on the **Vehicle Records per Status Period** and **Send Vehicle Status Period** settings. Vehicle Records per Status Period and Send Vehicle Status Period are defined on the **PLC EtherNet/IP Settings** page in the Configurator when creating the Node Controller Configuration File.

The maximum update rate is 224 vehicle records per status period. When sizing this tag, make sure that the number of vehicle records per status period does not exceed the number of entries in the MMI_vehicle_status tag. The HLC round-robins, if needed, the writing of vehicle records in batches that range from 1 (minimum) to 224 (maximum) records as specified by **Vehicle Records per Status Period**.

NOTE: When updating more than 224 vehicles, the HLC still updates at **Send Vehicle Status Period** intervals, but round-robins 224 vehicle records at a time. For instance, for 448 vehicles, vehicles 1…224 get updated, then next status period vehicles 225…448 get updated, next status period vehicles 1…224 get updated, and so on.

MMI_vehicle_status must be defined if **Use Extended Vehicle Status** is not selected (cleared) in the Configurator when creating the Node Controller Configuration File.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

UDT Field Descriptions

Table 12: UDT Configuration for udt_MMI_vehicle_status

Field Name	Data Type	Style	Range
path_id	INT	Decimal	0…65535
dest_path_id	INT	Decimal	0…65535
position	REAL	Float	0.0…41.0 (m, floating point)
commanded_position	REAL	Float	-41.0…+41.0 (m, floating point)
velocity	REAL	Float	-5.0…+5.0 (m/s, floating point)
dest_station_id	INT	Decimal	0…2048
command	SINT	Hex	0x00, 0xB0, 0xB1, 0xB7
flags	SINT	Binary	Bits 0…7 are Flags

path_id – The ID of the path where the vehicle is located. If this field is a zero, the vehicle at this index is not in use (for example, no such vehicle exists). In a multi-path transport system, this field changes as the vehicle progresses from path to path.

dest_path_id – The ID of the path where the vehicle is headed. Equal to the ID of the path where the vehicle is located if motion has completed. Equal to 0 if the vehicle has never moved or if the vehicle is under an MMI_vehicle_follow_order.

position – The last reported position (in meters) of the vehicle, relative to the start of the specified path, expressed as a 32-bit single-precision floating point number. Zero position is defined as the midpoint of the vehicle at the beginning of the path.

commanded_position – The destination for the vehicle.

Cmd	Command Type	Description
0x00	No command or command complete	4-byte value that is set to zero
0xB0	Vehicle station order	
0xB1	Vehicle position order	4-byte value that is the commanded destination on the specified path (-41.0…+41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).
0xB7	Vehicle follow order	4-byte value that is the commanded distance to the vehicle being followed (0…41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).

velocity – The last reported velocity (in m/s) of the vehicle on the specified path, expressed as a 32-bit single-precision floating point number.

dest_station_id – The ID number of the Station the vehicle is heading to when under a Station command. This field is only valid when the **command** field is 0xB0.

command – Shows the type of command currently active for the vehicle.

Value	Description
0x00	No command active
0xB0	Vehicle Station Order
0xB1	Vehicle Position Order
0xB7	Vehicle Follow Order

flags – Vehicle status indicators.

Bit	Description
0	**Vehicle Signal Status:** 0 – The motor does not detect the vehicle. 1 – The motor currently in charge of the vehicle detects the vehicle.
1	**Obstructed Status:** 0 – The vehicle is not obstructed and able to acquire permission to move further. 1 – The vehicle is obstructed and unable to acquire permission to move further. Obstructions are due to either a vehicle in the way, a hardware fault, or motion is suspended. Vehicles in the way occur during normal operation when vehicles are in a queue or when a vehicle is in a switch, which keeps another vehicle from entering the switch. If after three consecutive queries of the vehicle status the obstructed bit is still set, check for an actual obstruction.
2	**Hindered Status:** 0 – The vehicle is making progress on its current move profile. 1 – The vehicle is not making progress towards the position it has most recently been granted permission to move to. Lack of progress can happen when a vehicle is held back by some external force including a foreign object jamming a vehicle on the guideway or if the vehicle is not in motion while under sync control). The motors continue to apply force on the vehicle to try to move it indefinitely.

Bit	Description
3…6	**PID Loop Set Index:** 0 – Use user-defined PID set 0 – Unloaded PID values. 1 – Use user-defined PID set 1 – Loaded PID values. 2 – Use user-defined PID set 2. 3 – Use user-defined PID set 3. 4 – Use user-defined PID set 4. 5 – Use user-defined PID set 5. 6 – Use user-defined PID set 6. 7 – Use user-defined PID set 7. 8 – Use user-defined PID set 8. 9 – Use user-defined PID set 9. 10 – Use user-defined PID set 10. 11 – Use user-defined PID set 11. 12 – Use user-defined PID set 12. 13 – Use user-defined PID set 13. 14 – Use user-defined PID set 14. 15 – Use user-defined PID set 15 – Startup PID values. This PID set is automatically used during startup. If it is not defined, PID set 0 is scaled by 25% and used for startup.
7	**Suspect Status (MM LITE, QS 100 only):** 0 – The vehicle is not suspect and is free to move. 1 – The vehicle is suspect. The motor has detected that the vehicle has been manually moved out of the control region. Typically, control of the vehicle cannot be regained. Even if control is regained, the vehicle continues to be suspect until an MMI_vehicle_command is issued to clear the Suspect Bit.

Message Configuration

Table 13: Message Configuration for MMI_vehicle_status

Source Element	vehicle_status Array tag of type udt_MMI_vehicle_status. Array index corresponds to HLC vehicle ID. Since vehicle ID 0 is invalid, array element [0] is not used.
Number of Elements	Min: 2 Max: Max vehicles + 1 Specify the number of MMI_vehicle_status tags to read (one per vehicle) with one message.
Destination Element	MMI_vehicle_status

See Also

MMI_vehicle_position_order on page 60
MMI_vehicle_follow_order on page 64
MMI_extended_vehicle_status on page 77

MMI_extended_vehicle_status

Type

HLC → Host Controller

Purpose

Reports the version 3 extended status of all vehicles in the transport system. This tag is a one-dimensional array of type udt_MMI_extended_vehicle_status indexed by vehicle ID. Index 0 is not used since a vehicle ID of 0 in the transport system is invalid and not used. A vehicle entry is in use when the **path_id** field in the entry is nonzero. A vehicle entry is not in use when the **path_id** field in the entry is zero (that is, no vehicle exists at that index).

This array must be sized to the maximum number of vehicles in the transport system plus one (vehicle 0 is invalid). The **PLC Max Vehicle ID** setting establishes the maximum vehicle ID that the HLC reports vehicle status for to the MMI_extended_vehicle_status tag in the host controller memory. PLC Max Vehicle ID is defined on the **PLC EtherNet/IP Settings** page in the Configurator when creating the Node Controller Configuration File. See either the *MagneMover LITE Configurator User Manual*, MMI-UM008, or the *QuickStick Configurator User Manual*, MMI-UM009. The HLC updates the MMI_extended_vehicle_status tag in the host controller memory based on the **Vehicle Records per Status Period** and **Send Vehicle Status Period** settings. Vehicle Records per Status Period and Send Vehicle Status Period are defined on the **PLC EtherNet/IP Settings** page in the Configurator when creating the Node Controller Configuration File.

The maximum update rate is 144 vehicle records per status period. When sizing this tag, make sure that the number of vehicle records per status period does not exceed the number of entries in the MMI_extended_vehicle_status tag. The HLC round-robins, if needed, the writing of vehicle records in batches that range from 1 (minimum) to 144 (maximum) records as specified by **Vehicle Records per Status Period**.

NOTE: When updating more than 144 vehicles, the HLC still updates at **Send Vehicle Status Period** intervals, but round-robins 144 vehicle records at a time. For instance, for 288 vehicles, vehicles 1…144 get updated, then next status period vehicles 145…288 get updated, next status period vehicles 1…144 get updated, and so on.

MMI_extended_vehicle_status must be defined if **Use Extended Vehicle Status** is selected (checked) in the Configurator when creating the Node Controller Configuration File.

Support

This command is supported in the latest software release for the following product lines:

- MagneMover LITE transport systems.
- QuickStick 100 transport systems.

UDT Field Descriptions

Table 14: UDT Configuration for MMI_extended_vehicle_status

Field Name	Data Type	Style	Range
path_id	INT	Decimal	0…65535
dest_path_id	INT	Decimal	0…65535
position	REAL	Float	-41.0…+41.0 (m, floating point)
commanded_position	REAL	Float	-41.0…+41.0 (m, floating point)
velocity	REAL	Float	-5.0…+5.0 (m/s, floating point)
dest_station_id	INT	Decimal	0…2048
command	SINT	Hex	0x00, 0xB0, 0xB1, 0xB7
flags	INT	Binary	Bits 0…15 are Flags
followed_vehicle_id	INT	Decimal	Command = 0x00, 0xB0, 0xB1 – 0 Command = 0xB7 – 1…65535
current_target	REAL	Float	-41.0…+41.0 (m, floating point)
reported_pid_set_index	SINT	Decimal	0…15
ordered_pid_set_index	SINT	Decimal	0…15
time_since_last_update	REAL	Float	Seconds (floating point)
ordered_acceleration_limit	REAL	Float	0…60.0 (m/s^2, floating point)
ordered_velocity_limit	REAL	Float	0…41.0 (m/s, floating point)
active_move_to_station_offset	REAL	Float	-41.0…+41.0 (m, floating point)

path_id – The ID of the path where the vehicle is located. If this field is a zero, the vehicle at this index is not in use (for example, no such vehicle exists). In a multi-path transport system, this field changes as the vehicle progresses from path to path.

dest_path_id – The ID of the path where the vehicle is heading when the vehicle is under a motion command. Equal to the ID of the path where the vehicle is located if motion has completed. Equal to 0 if the vehicle has never moved or if the vehicle is under an MMI_vehicle_follow_order.

position – The last reported position (in meters) of the vehicle, relative to the start of the specified path, expressed as a 32-bit single-precision floating point number. Zero position is defined as the midpoint of the vehicle at the beginning of the path.

commanded_position – The destination for the vehicle.

Cmd	Command Type	Description
0x00	No command or command complete	4-byte value that is set to zero
0xB0	Vehicle station order	

Cmd	Command Type	Description
0xB1	Vehicle position order	4-byte value that is the commanded destination on the specified path (-41.0…+41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).
0xB7	Vehicle follow order	4-byte value that is the commanded distance to the vehicle being followed (0…41.0 m) expressed as a 32-bit single-precision floating point number using little-endian format).

velocity – The last reported velocity (in m/s) of the vehicle on the specified path, expressed as a 32-bit single-precision floating point number.

dest_station_id – The ID number of the Station the vehicle is heading to when under a Station command. This field is only valid when the **command** field is 0xB0.

command – Shows the type of command currently active for the vehicle.

Value	Description
0x00	No command active
0xB0	Vehicle Station Order
0xB1	Vehicle Position Order
0xB7	Vehicle Follow Order

flags – Vehicle status indicators.

Bit	Description
0	**Vehicle Signal Status:** 0 – The motor does not detect the vehicle. 1 – The motor currently in charge of the vehicle detects the vehicle.
1	**Obstructed Status:** 0 – The vehicle is not obstructed and able to acquire permission to move further. 1 – The vehicle is obstructed and unable to acquire permission to move further. Obstructions are due to either a vehicle in the way, a hardware fault, or motion is suspended. Vehicles in the way occur during normal operation when vehicles are in a queue or when a vehicle is in a switch, which keeps another vehicle from entering the switch. If after three consecutive queries of the vehicle status the obstructed bit is still set, check for an actual obstruction.

Bit	Description
2	**Hindered Status:** 0 – The vehicle is making progress on its current move profile. 1 – The vehicle is not making progress towards the position it has most recently been granted permission to move to. Lack of progress can happen when a vehicle is held back by some external force including a foreign object jamming a vehicle on the guideway or if the vehicle is not in motion while under sync control). The motors continue to apply force on the vehicle to try to move it indefinitely.
3	**Stall Status:** 0 – The vehicle is not stalled. 1 – The vehicle has stalled. A stalled vehicle is defined as a vehicle that has jammed and the motor in control of the vehicle has had to reduce power to keep it from overheating.
4	**Locked Status:** 0 – The vehicle is not locked. 1 – The vehicle is locked. Until the vehicle is unlocked, the system rejects execution of move commands for the vehicle.
5	**Locate Completed Bit:** 0 – Vehicle locate is not completed (startup is in progress). 1 – Vehicle locate has completed.
6	Reserved
7	**Suspect Status (MM LITE, QS 100 only):** 0 – The vehicle is not suspect and is free to move. 1 – The vehicle is suspect. The motor has detected that the vehicle has been manually moved out of the control region. Typically, control of the vehicle cannot be regained. Even if control is regained, the vehicle continues to be suspect until an MMI_vehicle_command is issued to clear the Suspect Bit.
8	**Following Upstream:** 0 – This vehicle is not following another vehicle. 1 – This vehicle is following the vehicle that is specified in the **followed_vehicle_id** field and that vehicle is upstream from this vehicle.
9	**Following Downstream:** 0 – This vehicle is not following another vehicle. 1 – This vehicle is following the vehicle that is specified in the **followed_vehicle_id** field and that vehicle is downstream from this vehicle.
10	**Caught Up:** 0 – This vehicle is not following another vehicle. 1 – The motor controller is reporting that this following vehicle has caught up to the vehicle it is following and is at the requested following distance.
11	**Profile Stale Error:** 0 – This vehicle is not following another vehicle. 1 – The motor controller is reporting that this following vehicle has not received a profile from the vehicle it is following, in the expected time frame.

Bit	Description
12	**Excessive Following Error:** 0 – This vehicle is not following another vehicle. 1 – The motor controller is reporting that this following vehicle is too far from where it is expected to be (possibly stopped receiving following profile data).
13	**Vehicle Decoupled:** 0 – A platoon follower vehicle not in the decoupling state, or a vehicle not in a platoon. 1 – A platoon follower vehicle is decoupling from the vehicle it is following. This bit stays set from when the MC reports decoupling, until the decoupled vehicle reaches its new destination under the 0xB1 position order that was converted from the 0xB7 follow order.
14, 15	Reserved/unused

followed_vehicle_id – The ID number of the vehicle that this vehicle is following when this vehicle is under a vehicle follow order and is part of a platoon (vehicle command type is 0xB7). Equal to 0 if this vehicle is not under a vehicle follow order (vehicle command type is 0x00, 0xB0, or 0xB1).

current_target – The position that the vehicle has permission to move to (in meters) as measured from the upstream end of the current path, expressed as a 32-bit single-precision floating point number. If the vehicle has completed its current order and is stopped, this value is equal to the position of the vehicle.

If the vehicle is under an MMI_vehicle_follow_order, this value is the follow_distance (the distance (in meters) center-to-center that this vehicle maintains behind the vehicle being followed.

reported_pid_set_index – The PID control loop set currently in use for the vehicle.

Table 15: PID Control Loop Sets

Value	Description
0	Use user-defined PID set 0 – Unloaded PID values.
1	Use user-defined PID set 1 – Loaded PID values.
2	Use user-defined PID set 2.
3	Use user-defined PID set 3.
4	Use user-defined PID set 4.
5	Use user-defined PID set 5.
6	Use user-defined PID set 6.
7	Use user-defined PID set 7.
8	Use user-defined PID set 8.
9	Use user-defined PID set 9.
10	Use user-defined PID set 10.
11	Use user-defined PID set 11.
12	Use user-defined PID set 12.

Table 15: PID Control Loop Sets

Value	Description
13	Use user-defined PID set 13.
14	Use user-defined PID set 14.
15	Use user-defined PID set 15 – Startup PID values. This PID set is automatically used during startup. If it is not defined, PID set 0 is scaled by 25% and used for startup.

ordered_pid_set_index – The commanded PID control loop set for vehicle movement if a movement order is in progress. This field is zero if the vehicle has not been given an order since startup. See Table 15: PID Control Loop Sets.

time_since_last_update – The time in seconds since the last vehicle status for this vehicle was received from the motor expressed as a 32-bit single-precision floating point number.

ordered_acceleration_limit – The last acceleration order from the host (in m/s^2) for the vehicle on the specified path, expressed as a 32-bit single-precision floating point number when a movement order is in progress. This field is zero if the vehicle has not been given an order since startup.

ordered_velocity_limit – The last velocity order from the host (in m/s) for the vehicle on the specified path, expressed as a 32-bit single-precision floating point number when a movement order is in progress. This field is zero if the vehicle has not been given an order since startup.

active_move_to_station_offset – Commanded offset from the destination station relative to the position of the station expressed as a 32-bit single-precision floating point number. Offsets downstream from a station are positive values, offsets upstream from a station are negative values.

Message Configuration

Table 16: Message Configuration for MMI_extended_vehicle_status

Source Element	Extended_vehicle_status Array tag of type udt_MMI_extended_vehicle_status. Array index corresponds to HLC vehicle ID. Since vehicle ID 0 is invalid, array element [0] is not used.
Number of Elements	Min: 2 Max: Max vehicles + 1 Specify the number of MMI_extended_vehicle_status tags to read (one per vehicle) with one message.
Destination Element	MMI_extended_vehicle_status

See Also

MMI_vehicle_position_order on page 60
MMI_vehicle_follow_order on page 64
MMI_vehicle_status on page 73

HLC Status Codes

Table 17 lists the status codes that the HLC returns when it accepts, rejects, or completes platooning commands.

Table 17: HLC Status Codes

Value	Description
0x00	Command Accepted
0x01	Command Rejected – Invalid vehicle ID
0x03	Command Rejected – Invalid path ID
0x04	Command Rejected – Invalid position (off path)
0x08	Command Rejected – Invalid parameter (acceleration, velocity, or direction)
0x0A	Command Rejected – Startup sequence not initiated/not complete
0x0B	Command Rejected – Invalid parameter (acceleration, velocity, or direction)
0x0C	Command Rejected – Initialization not complete
0x0D	Command Rejected – Reset active
0x10	Command Rejected – Programming active
0x13	Command Rejected – Vehicle lock active
0x1D	Command Rejected – Vehicle in motion
0x1E	Command Rejected – Decouple not complete
0x41	Command Failed – Unable to complete
0x80	Command Completed Successfully
0x81	Command Status – Vehicle has caught up to the vehicle in a platoon it is following
0x82	Command Status – Vehicle in a platoon is in the process of decoupling from the vehicle it is following
0x83	Command Status – Vehicle decoupling from a platoon has arrived at its destination before completing the decouple process. Arrival position tolerance permits the decoupling vehicle to be identified as having arrived at its destination in the following cases: • Waiting for decoupling to start • Waiting for decoupling to complete

Revision History

Ver.	Change Description
A	Initial release
B	Added support for QuickStick 100 and information on minimum space between vehicles on page 8.
C	Updated the description of the platooning function and procedures for platooning. Added the troubleshooting section. Corrected the minimum tag array sizes. Updated the support information on the back page.
D	Updated the follow order to include the Decouple Destination. Updated the extended vehicle status to report decoupling. Added an example of using a Decouple Destination. Restructured the EtherNet/IP tag descriptions.
E	Identified support for the Decouple Destination as MM LITE only. Corrected NCHost PID Set Menu and Direction Menu descriptions.
F	Added additional HLC Status Codes (0x1E, 0x83). Updated NCHost references to new version (15.0.x). Updated the description of the last_order_accepted_is_complete field in the MMI_vehicle_order_status tag.

Rockwell Automation Support

Use the following resources to access support information.

Technical Support Center	Knowledgebase Articles, How-to Videos, FAQs, Chat, User Forums, Product Notification Updates, and the Direct Dial Code for your product	rok.auto/support
Literature Library	Installation Instructions, Manuals, Brochures, and Technical Data.	rok.auto/literature
Product Compatibility and Download Center (PCDC)	Get help determining how products interact, check features and capabilities, and find associated firmware.	rok.auto/pcdc.
Product Catalog and Configurator	Additional product information including CAD drawing files, 3D models, photos, and more.	rok.auto/configure

Documentation Feedback

Your comments help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the How Are We Doing? form at rok.auto/docfeedback.

Rockwell Automation maintains current product environmental information on its website at rok.auto/pec.

Product certificates are available on the Rockwell Automation website: rok.auto/certifications.

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control, and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

139 Barnum Road, Devens, MA 01434, USA
+1 978-757-9100
Publication MMI-UM036F-EN-P - October 2022 DIR 10005354930
Supersedes MMI-UM036E-EN-P – April 2021