

## Multi-Robot Operator Control Unit for Unmanned Systems



Aug 01 2008

The U.S. military has incorporated many robotic systems into battlefield scenarios over the past several years, ranging from a single vehicle or unattended sensor to multiple vehicles and sensors. The majority of these systems use proprietary protocols requiring the creation and maintenance of a custom (stovepipe) operator control unit (OCU) able to control only a single asset type or a very limited subset of assets. With missions ranging from neutralization of an improvised explosive device (IED) to remote surveillance, the mix of hardware requires mission-specific control of functionality, and mission-specific information displayed to the operator. Some control units must be handheld, while others have large computational requirements and must be located in a Humvee or at a stationary land-based location.

Space and Naval Warfare Systems Center, San Diego (SSC San Diego) has developed an unmanned vehicle and sensor operator control interface capable of simultaneously controlling and monitoring multiple sets of heterogeneous systems. The Multi-robot Operator Control Unit (MOCU) accommodates a wide range of vehicles and sensors in various mission scenarios. MOCU currently controls all of the SSC San Diego developmental vehicles (land, air, sea, and undersea), including the SPARTAN Advanced Concept Technology Demonstration (ACTD) Unmanned Surface Vehicle (USV), the iRobot PackBot, and the Family of Integrated Rapid Response Equipment (FIRRE) vehicles and sensors.

MOCU has been designed using a more modular, scalable, and flexible architecture. Modularity allows for a breadth of functionality, such as communicating in unrelated protocols or displaying video with a proprietary video codec, and also makes third-party expansion possible. Scalability enables MOCU to be installed on a wide range of hardware. MOCU also allows the user to define what information is displayed and determine what control is needed for each system.



Figure 1: MOCU uses both aerial photo and DNC maps to control and monitor land, sea, and air vehicles.

### MOCU Architecture

MOCU was created to provide the ability to simultaneously control and monitor multiple existing or newly developed heterogeneous unmanned systems. Heterogeneous describes vehicles that are dissimilar in both modality (land, air, sea, or undersea) and communications protocol. The goal is to not just minimally control, but to have full access to, the vehicle/sensor and its payloads. The core acts as the glue to tie modules to other modules and to the user interface. The interface between the module and the core is defined by the type of module. The user interface, including display, joystick, and keyboard, is defined by a configuration file. Multiple configuration files are loaded by the core, but only one is used at a given time.

Modules contain specific functionality that modifies what MOCU can display and control. By achieving specific capabilities through associated modules, the core remains relatively constant. The core manages information from the modules, such as waypoints, vehicle or sensor status, tracks, etc. Each loaded module has one or more predefined types, depending on the functionality it performs, and must conform to at least a subset of the API for its predefined type(s). By standardizing the interface, multiple modules of the same type may be loaded, which allows MOCU to simultaneously control heterogeneous vehicles or display different types of maps.

To communicate with a vehicle or sensor, MOCU must load a protocol module, which then communicates with the vehicle and provides control when MOCU has requested it. MOCU supports a



Figure 2: The command and control station for the FIRRE project. MOCU monitors and controls multiple vehicles and stationary sensor suites.

broad range of vehicle autonomy, from autonomous vehicles with obstacle avoidance, to teleoperated vehicles, to stationary sensors.

The creation of new modules is greatly simplified by the core handling the data management. For example, the map module type is vital for missions where geographic awareness is important. The core maintains the information on items on the maps and handles user input for actions such as panning or interfacing with a vehicle icon. The type of map(s) displayed within MOCU is chosen depending on the mission and modality of the vehicle (see Figure 1). The modular architecture of MOCU allows for third-party development, so that a module of any of the defined types can be created to completely alter the functionality of MOCU.

The capabilities of MOCU are adjusted by adding or removing modules with no need to modify the core. If the application requires a minimal installation, only the required modules to control or monitor a specific vehicle or sensor are installed on the target machine. A large control structure with multiple semiautonomous vehicles communicating in different protocols and modalities may require more functionality including maps, vehicle status, a planner, and a scheduler (see Figure 2).

## **Flexible User Interface**

The user interface, which includes the display and user input, is defined by XML configuration files. The display configuration specifies the location and types of maps, video, gauges, etc. The user input includes the mapping of the joystick, keyboard, and mouse to commands in MOCU. MOCU has two general modes of operation: monitor and control. The user interface in both is defined by the configuration files. One operator can use MOCU to monitor multiple vehicles and sensors, but at any given time, can only control one.

In monitor mode, all the vehicles in communication with MOCU are shown geographically on all displayed maps. Vehicle status information is also presented for each vehicle along with the option to display video if the status gauge is available. No control is available to the vehicle or sensors while in monitor mode. At any point, the user can switch to control mode by selecting a vehicle.

While in control mode, MOCU has complete control of a vehicle or sensor and its payloads. More status information is available, and the vehicle or sensor can be controlled through teleoperation or by sending waypoints to the vehicle. For sea vehicles, the operator will require digital nautical charts (DNCs) detailing features in the waterways, whereas for ground vehicles, detailed terrain information is required.

The configuration files specify how the information from the selected vehicle is presented on the display. Graphical gauges are used to give quick feedback, while textual gauges provide absolute values where appropriate. The style of a gauge, including dashboard, bar graph, textual or indicator, may be chosen to present the information in the format that is best suited to the nature of the data.

The military is moving towards network- centric warfare where systems will be required to communicate with one another to share information. MOCU has the capability to communicate with existing command and control (C2) systems, and thus provide data from robotic assets to enrich the battle-space awareness. Communication with a C2 system may be unidirectional or bidirectional.

In addition to monitoring the route that a particular vehicle is executing, the overarching C2 system can also send routes to the vehicle via MOCU. Depending on how MOCU is set up, these routes can either be immediately passed on unchanged, or they can be displayed to the MOCU operator as a C2-Link suggested route. In the latter case, the MOCU operator can accept the route as is, modify the route before sending it down to the vehicle, or reject the route. Whichever option the operator chooses, the action is reported to the C2 Link so that the higher level C2 system is aware of what the vehicle is actually doing.

## **Implementations**

MOCU was originally created in 2001 in response to a need for the Unmanned Ground Vehicle (UGV) program for a small OCU capable of creating pixelaccurate routes for a small ground robot using aerial photography. This

early version had a moving map that used aerial photos to display the location of the vehicle as well as allow the user to define routes. The status display was limited to textual output.

The SPARTAN ACTD work began in late 2003. The goal of the SPARTAN program is to field an Unmanned Surface Vehicle (USV). Rather than create another custom OCU, the SPARTAN team evaluated current OCUs and decided MOCU was the most viable candidate. The Family of Integrated Rapid Response Equipment (FIRRE) program was the next to adopt MOCU in 2005. The purpose of the FIRRE program is to integrate UGVs, unattended ground sensors (UGSs) and ground surveillance radars (GSRs) into a system that provides force protection of high-value areas for forward-deployed forces. MOCU is being modified to monitor the status of ground sensors, monitor and control the GSRs, and control the Tactical Amphibious Ground Support System (TAGS) vehicles.

### **The Future of MOCU**

MOCU has continued to be adopted for more unmanned systems programs. There are plans to interface to additional UAVs and add features to enhance the control and mission planning for UAVs, such as route verification. There will be additional types of modules developed and additional variants of the existing types, possibly including new types of map modules with 3D display capabilities, automatic mission planning modules and path planning modules, generic payload modules, etc.

SSC San Diego is planning on developing a distributed control architecture, where multiple MOCUs communicating through a common network would have the ability to pass vehicle, sensor, and database information among them, as well as control of individual vehicles. There is a growing objective within the unmanned system community for a common OCU that will control many, if not all, of the vehicles in a particular mission. MOCU is currently capable of controlling multiple vehicles, but there is a definite limit on the number of systems one operator can control or even supervise.

Even as unmanned systems gain continuing acceptance on the battlefield, most require a stovepipe OCU to control them. MOCU provides the capability to control and monitor multiple heterogeneous systems while remaining flexible to handle different missions. MOCU was created to be modular and scalable to prevent it from becoming a large software program that requires an inordinate amount of effort to make a relatively small change. As technology advances, MOCU will be able to advance by adding or replacing functionality.

*This article was written by D. Powell, G. Gilbreath, and M. Bruch of the Space and Naval Warfare Systems Center, San Diego. For more information, [click here](#).*

Close Window