ROBODEXS; Multi-robot Deployment & Extraction System

Jeremy P. Gray, James R. Mason, Michael S. Patterson, Matthew W. Skalny
U.S. Army Tank-Automotive Research, Development, and Engineering Center (TARDEC)
6501 E. 11 Mile Rd., RDTA-RS MS264, Warren, MI, USA 48397

ABSTRACT

The importance of Unmanned Ground Vehicles (UGV's) in the Military's operations is continually increasing. All Military branches now rely on advanced robotic technologies to aid in their missions' operations. The integration of these technologies has not only enhanced capabilities, but has increased personnel safety by generating larger standoff distances. Currently most UGV's are deployed by an exposed dismounted Warfighter because the Military possess a limited capability to do so remotely and can only deploy a single UGV.

This paper explains the conceptual development of a novel approach to remotely deploy and extract multiple robots from a single host platform. The Robotic Deployment & Extraction System (ROBODEXS) is a result of our development research to improve marsupial robotic deployment at safe standoff distances. The presented solution is modular and scalable, having the ability to deploy anywhere from two to twenty robots from a single deployment mechanism. For larger carrier platforms, multiple sets of ROBODEXS modules may be integrated for deployment and extraction of even greater numbers of robots. Such a system allows mass deployment and extraction from a single manned/unmanned vehicle, which is not currently possible with other deployment systems.

Keywords: ROBODEXS, Marsupial, Deployment, Extraction, Multiple UGV, Modular, Scalable, Robot, Unmanned

1. INTRODUCTION

Until recently, small robots did not have a specialized storage area and were generally carried loosely in the back of an armored vehicle such as an MRAP. A Soldier would be required to exit the vehicle while carrying the robot, or have the robot handed down to the Warfighter in order to place it on the ground. To increase Soldier safety and facilitate ease of use, several versions of externally mounted single robot deployment systems were developed by the US Army. These ranged from a scissor lift underbelly "robot elevator", to a side-mounted clamshell box, to a rear-mounted scoop that was developed in theater. What emerged from this was the RG-31Robot Deployment System (RDS), allowing the RG-31 to carry, deploy, and extract a ***single small robot*** from a hydraulically actuated box attached to the rear of the vehicle. The box carries the robot in a high, vertical position and is hinged to bring the box down to the ground level for deployment.. Research has shown commercial robotics developers have also experimented with marsupial capabilities, allowing a medium to large class UGV to carry and deploy one or two small robots [1-4]. Each of these systems are custom tailored to attach to the host vehicle and are generally limited to deployment-only [4] or deployment-and-extraction [3] of one or two smaller robots.

The interiors of currently fielded manned armored vehicles that carry small robots are space limited. Many were originally designed to carry only Soldiers, but seats have been removed and floor space has been utilized by the increased electronics payloads of these vehicles. Most small robots in use are low profile, and would benefit from being arranged vertically for stowage and transport. However, manual vertical stowage of small robots is cumbersome, raises the risk of damage to the robot during lifting and increases the risk of injury to the Soldier. ROBODEXS has been designed to automatically deploy and extract small robots while stowing them in a compact, vertical position. While all the systems are stowed within the ROBODEXS, the ROBODEXS space requirements will be slightly larger than the space required to vertically place each platform on its own (see Figure 1). Each of the chosen small robot's vertically stowed space claim dimensions are approximately 11.5 inches deep, 27 inches tall, and 20.25 inches wide; thus the for three modules (one master and two slave), the ROBODEXS will be approximately 39.2 inches deep, 28.5 inches tall, and 31 inches wide (with added size for stowing mechanics).

Report Documentation Page			*Form Approved* *OMB No. 0704-0188*
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.			
1. REPORT DATE **03 APR 2012**	2. REPORT TYPE **Journal Article**		3. DATES COVERED **01-01-2012 to 01-04-2012**
4. TITLE AND SUBTITLE **ROBODEXS; Multi-robot Deployment & Extraction System**			5a. CONTRACT NUMBER
			5b. GRANT NUMBER
			5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S) **Jeremy Gray; James Mason; Michael Patterson; Matthew Skalny**			5d. PROJECT NUMBER
			5e. TASK NUMBER
			5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **U.S. Army TARDEC,6501 East Eleven Mile Rd,Warren,Mi,48397-5000**			8. PERFORMING ORGANIZATION REPORT NUMBER **#22700**
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) **U.S. Army TARDEC, 6501 East Eleven Mile Rd, Warren, Mi, 48397-5000**			10. SPONSOR/MONITOR’S ACRONYM(S) **TARDEC**
			11. SPONSOR/MONITOR’S REPORT NUMBER(S) **#22700**
12. DISTRIBUTION/AVAILABILITY STATEMENT **Approved for public release; distribution unlimited**			
13. SUPPLEMENTARY NOTES **For SPIE DSS 2012 Conference.**			
14. ABSTRACT **The importance of Unmanned Ground Vehicles (UGV?s) in the Military?s operations is continually increasing. All Military branches now rely on advanced robotic technologies to aid in their missions? operations. The integration of these technologies has not only enhanced capabilities, but has increased personnel safety by generating larger standoff distances. Currently most UGV?s are deployed by an exposed dismounted Warfighter because the Military possess a limited capability to do so remotely and can only deploy a single UGV. This paper explains the conceptual development of a novel approach to remotely deploy and extract multiple robots from a single host platform. The Robotic Deployment & Extraction System (ROBODEXS) is a result of our development research to improve marsupial robotic deployment at safe standoff distances. The presented solution is modular and scalable, having the ability to deploy anywhere from two to twenty robots from a single deployment mechanism. For larger carrier platforms, multiple sets of ROBODEXS modules may be integrated for deployment and extraction of even greater numbers of robots. Such a system allows mass deployment and extraction from a single manned/unmanned vehicle, which is not currently possible with other deployment systems.**			
15. SUBJECT TERMS **ROBODEXS, Marsupial, Deployment, Extraction, Multiple UGV, Modular, Scalable, Robot, Unmanned**			

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Public Release**	**11**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

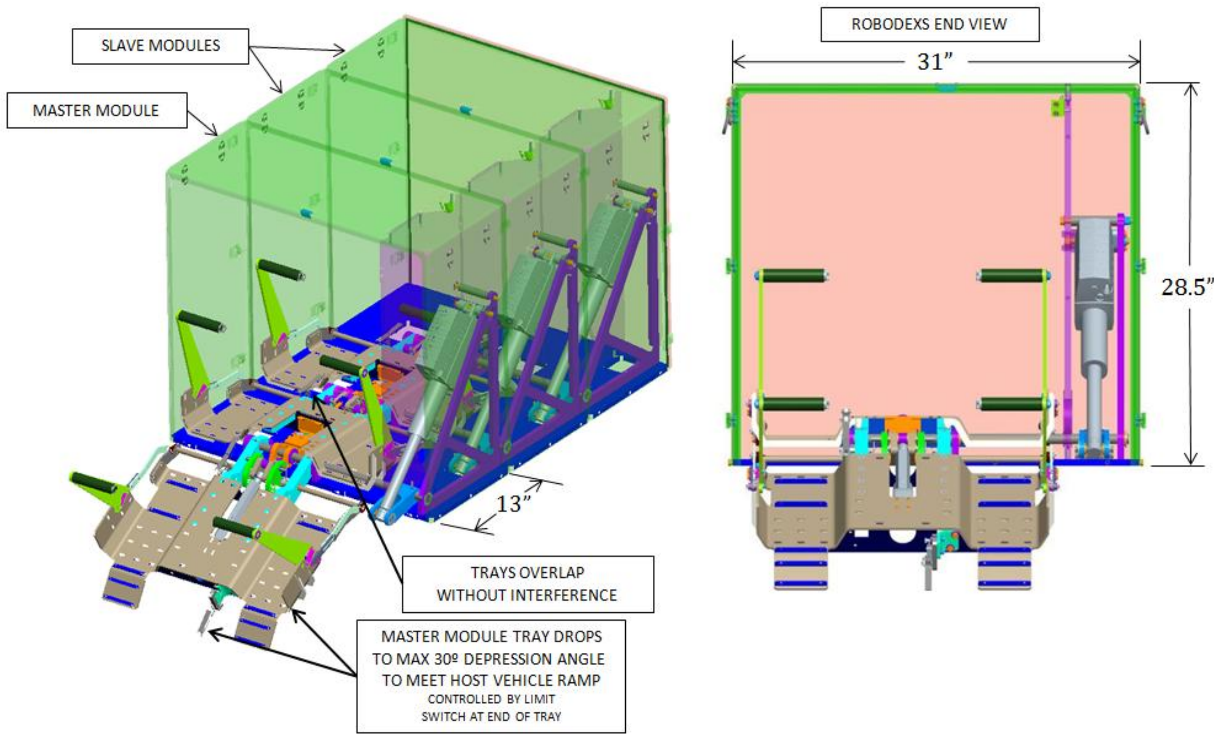

Figure 1. CAD 3D transparent images of 3 ROBODEXS modules in the deployed position, dimensioned for space claim representation

With the National Defense Authorization Fiscal Year 2001, Congress established the following: "It shall be a goal of the Armed Forces to achieve the fielding of unmanned, remotely controlled technology such that… by 2015, one-third of the operational ground combat vehicles are unmanned."[5] With Congress setting forth such a high goal for the Military, there exists a need for a mass deployment system for robots. ROBODEXS may assist in this effort by eventually being integrated into a high volume of larger robots to enhance mission capabilities and increase the percentage of ground missions being executed by unmanned means and increasing the stand-off distance for Warfighters.

At the 2011 National Defense Industrial Association (NDIA) Ground Vehicle Systems Engineering and Technology Symposium (GVSETS) held in Dearborn, MI, the PM - Robotic Systems Joint Project Office stated that future small robots will incorporate semi-autonomy. With semi-autonomy, small robots may exploit the benefits of swarm behavior and will be able to negotiate and inspect an area of interest without requiring direct operator control. This semi-autonomy will allow a paradigm shift from one operator: one robot to one operator: many robots. In order to fully realize this future concept of operation, an effort must be undertaken to transport, deploy, and extract many robots from a single host platform. Lessons-learned from and technology based on ROBODEXS will directly support this.

2. HARDWARE

2.1 MODULARITY AND SCALABILITY

ROBODEXS consists of a set of modules, each containing a single robot, allowing the system to be scaled to carry, deploy, and extract anywhere from one to twenty small robot's. Theoretically, ROBODEXS could support an infinite number of Slave modules, but due to voltage drops and interference, each additional module addition beyond twenty would interfere and degrade the ROBODEXS performance. Each module set requires one Master Module and anywhere from zero to nineteen Slave Modules. Figure 2 illustrates 20 modules linked together, and displays how each module overlays one another to create a uniform ingress and egress track for the robots to maneuver along. Each module has features designed to align and latch together without tools. Adding modules is accomplished by sliding the modules together, making power and communications connections, and flipping the draw-latches to fasten the modules together. Alignment of the modules to each other is paramount, as the robot tray for each overlaps the adjacent module during deployment. Four tie-down rings are mounted to the exterior of each module, allowing flexible mounting in/on a variety of platforms using hook-end ratchet straps without requiring modification of the platform.

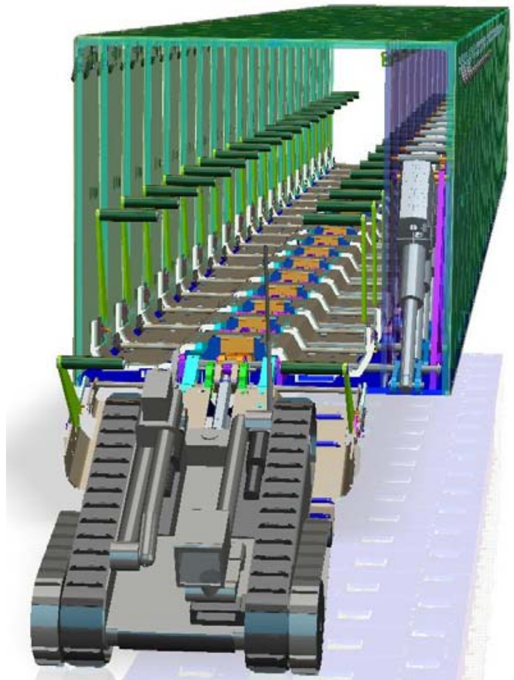

Figure 2. 3D rendering of 20 modules and robot driving into ROBODEXS

2.2 MASTER MODULE FUNCTIONAL DESCRIPTION AND ROBOT INTERACTION

The modules are arranged serially, with the Master Module acting as the sole ingress/egress point. The Master Module robot tray is unique from that of the Slave Module, swinging down to a maximum 30º depression angle to align to a rear-mounted ramp on some vehicles. A limit switch at the end of the ramp controls the motion of the ramp, signaling to the Master Module controller when contact is made. Since the Master Module tray may be at a severe depression angle, it includes tread ribs spaced at the same interval as the treads on the robot's tracks. This allows the robot to grip and climb an incline that would ordinarily cause loss of traction. The trays of each module have a center hump that forces the robot to self-align while being driven into the module. When all of the trays are in the horizontal position, they create a continuous path for the robot to traverse, as illustrated in Figure 2.

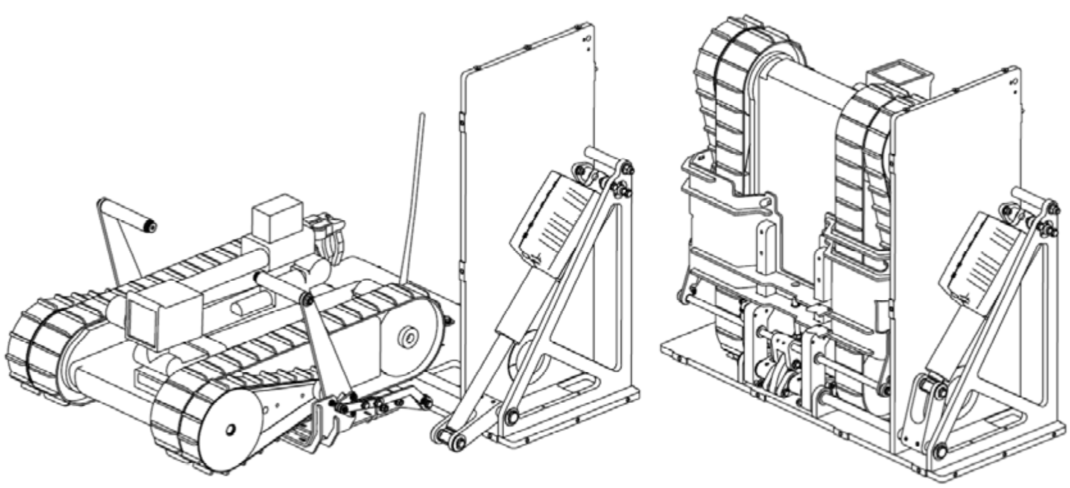

Figure 3. 3D image of a slave module ready to load a robot, unclamped (left), clamped and loaded robot (right)

2.3 SLAVE MODULE FUNCTIONAL DESCRIPTION AND ROBOT INTERACTION

The Slave Module utilizes a novel mechanism that allows a single linear actuator to clamp the robot to the tray and then lift it to a vertical stowed position over the stroke of the actuator (Figure 3, right). When the actuator direction is reversed, the robot is lowered from a vertical to horizontal position (Figure 3, left) at which point the tray contacts the adjacent module. Continued actuator motion then unclamps the robot. Counterbalance springs are used to prevent the mechanical system from unclamping the robot while in a vertical position. The actuator is connected to an arm at the end of a main drive shaft. The main drive shaft is connected via bar linkages to a counter-rotating intermediate shaft. Clamping is accomplished by a clamp dog that is linked to the intermediate shaft, locating and holding the robot body. The intermediate shaft is also mechanically linked to a pair of clamping arms that swing down, holding the top of the tracks in place. Through the combination of the clamp dog and clamping arms, ROBODEXS grips the robot body and robot’s tracks for stowage. Both mechanisms utilize geometry that cam-lock the arms and dog, providing a secure hold on the robot. When unclamping, the clamp dog rotates downward, below the plane of the center hump of the tray. Similarly, the clamp arms rotate upward, creating a clear path for robot ingress/egress.

2.4 REMOTE ROBOT POWER ACTUATION

The clamp dog contains a pogo pin that aligns to the robot push-button power switch when the robot is clamped. Once in stowed position, the free end of the pin is actuated by a solenoid. This allows the robot to be remotely powered off after stowage and to be remotely powered on prior to deployment.

2.5 ROBOT LOADING OPERATION

When loading an empty system, all trays are brought to a deploy state with all Slave Module trays in a horizontal position and the Master Module tray at whatever position is required for its limit switch to contact the floor or ramp. The ROBODEXS high-level control system, Master Controller (MC), identifies which module is ready to be loaded. Communication between the two modules, Master Module and the identified Slave Module, take place activating a green Light Emitting Diode (LED). This LED is easily visible through the camera of the robot to be loaded. The robot is driven into the Master Module and through the unloaded Slave Modules until it reaches the Slave Module with the lit green LED to be loaded. When the robot is in the proper position, identified through limit switches, the red LED is illuminated, along with the green LED. If the robot has overdriven its position by a predetermined distance, an overhead optical sensor senses the overrun, which in turn begins flashing the red LED. The flashing red LED signals to the operator that the robot must be backed up. Once the robot has backed up far enough to clear the optical sensor, while still contacting the positioning limit switch, the red LED ceases flashing and turns steady again. After a brief delay, the module is then ready to be stowed. The system then can load the robot into its stowage position automatically or by operator command. After the robot has been stowed, the next module upstream could then be loaded. Stowage of the robots continue in this way until all modules, including the Master Module are loaded, or until the user indicates that only a partial stowage operation is complete. The later sections will discuss the software and control states in depth associated with the loading, unloading, and failsafe procedures.

3. SOFTWARE

ROBODEXS consists of three primary software items – the Operator Control Unit (OCU), the ROBODEXS Master Controller (MC), and the Individual Motor Controllers (IMCs). The OCU is responsible for providing a basic operator interface for controlling and getting feedback from the ROBODEXS Master Controller and IMCs. The MC communicates with the IMCs, interfaces with the user (through communications to and from the OCU), and provides for hierarchical control over master and slave modules. This section focuses primarily on the design and functionality of the MC, with limited information on the OCU. The IMCs are detailed in the later section 3 (Control). The items covered in this section include: (i) primary goals in design of MC and OCU software, (ii) overview of the MC software design and its interface to the OCU and IMCs, (iii) functionality and use case examples for the MC, and (iv) software interoperability definitions.

3.1 PRIMARY GOALS IN DESIGN OF MASTER CONTROLLER AND OCU SOFTWARE

The ROBODEXS Master Controller and OCU software were designed to meet interoperability requirements from the Unmanned Ground Vehicle (UGV) Interoperability Profiles (IOPs). The IOPs are a set of documents developed by the Robotic Systems Joint Project Office (RSJPO) designed to facilitate the creation of interoperable robot systems and payloads (“plug and play”). Versions 0 of these documents were published in December of 2011, and describe selectable interoperability attributes in the areas of Communications, Payloads, and Control. These documents use the SAE AS-4 JAUS (Joint Architecture for Unmanned Systems) as the basis for command and control and reporting between OCUs, robots, and payloads. Making ROBODEXS compliant with version 0 of the IOPs, to the maximum extent possible, was an early goal of the project. In addition to supporting version 0 requirements, ROBODEXS also had to introduce changes or new items that will be considered for inclusion of version 1 of the IOPs. The specific IOP requirements that ROBODEXS is designed to meet are:

- ROBODEXS is a compound payload (modular payload) represented by a single JAUS node.
- ROBODEXS is capable of registering its provided services with the Discovery Service on a platform manager or on one provided by the ROBODEXS JAUS node itself.
- ROBODEXS supports core JAUS services including Management, Access Control, Liveness, and Events.
- ROBODEXS uses IOP specified data and power connector interfaces
- ROBODEXS OCU discovers and utilizes available JAUS services to control ROBODEXS and get state and error information from ROBODEXS
- JAUS Over UDP (JUDP) Transport Protocol

Some items that ROBODEXS implemented that are not covered in Version 0 of the IOPs, but may potentially be proposed as additions in version 1 of the IOPs include a second power connector to support current limits higher than the options currently specified, and two new JAUS services (Discrete State Driver and Discrete State Sensor).

3.2 OVERVIEW OF THE MASTER CONTROLLER SOFTWARE DESIGN AND ITS INTERFACE TO THE OCU AND IMCS

The ROBODEXS Software Architecture for the ROBODEXS MC consists of several software entities. Most of the software entities on the MC are organized around JAUS Subsystems, JAUS Nodes, and JAUS Components. Figure 4 illustrates the ROBORDEXS software architecture.

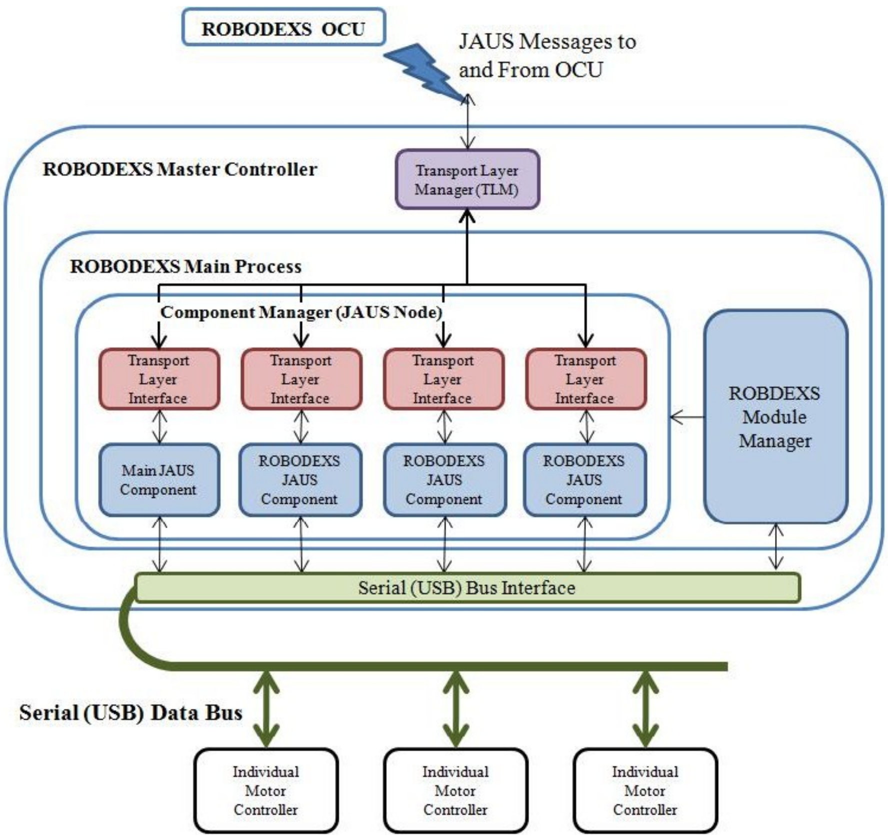

Figure 4. ROBODEXS Software Architecture Block Diagram

The MC is designed to be a modular payload that would be "plugged in" to a platform, which would typically be represented as a JAUS Subsystem. For this reason, ROBODEXS is represented as a single JAUS Node, which the Component Manager provides. The Component Manager defines one Main JAUS Component which contains the Discovery that the OCU uses to discovery how many ROBODEXS modules are present and what other services are being provided by the ROBODEXS. The Component Manager also creates and maintains one JAUS Component per IMC. These JAUS Components provide the interface for communicating with the IMCs. In addition to mandatory core services, each ROBODEXS JAUS Component provides two JAUS Services specifically created to meet the needs of ROBODEXS that are not defined in V0 of the IOPs – a Discrete State Driver service and a Discrete State Sensor service.

The Discrete State Driver service is designed to provide a way to "drive" a device using discrete states as opposed to a form of continuous control, such as velocity or position based control. The Discrete State Driver service defines a state machine, including all the states available and what the valid external transitions are (internal transitions unavailable outside the services can also happen). The Discrete State Driver service's input message set allows a client to query states, valid transitions, and current state, and to send a command to change to a new state. Its output message set includes messages for reporting current state, all states, and all valid transitions, and a transition completed notification message. The Discrete State Driver service is used to control the individual ROBODEXS modules.

The Discrete State Sensor service is designed to provide a simple way to report back on entities like switches that have a limited set of states they can be in, as opposed to a continuous feedback range. The Discrete State Sensor service is closely related to the Discrete State Driver service in that it reports the current state of something, but it does not define any transitions. A Discrete State Driver service could actually be used for the same purposes as the Discrete State Sensor service, but for ROBODEXS a separate service was needed to report on the presence of a robot within each module (yes or no). Therefore, the input and output message set for the Discrete State Sensor service was made to be different than that of the Discrete State Driver service. Table 1 shows the IOP Attributes implemented by ROBODEXS.

Table 1. Interoperability (IOP) Attributes

IOP Document	IOP Attribute	Value(s)
Overarching [8]	Transport	UDP
Overarching [8]	Platform Management	Basic
Payloads [9]	Power	Power Attribute A
Payloads [9]	Connector	Connector A
Payloads [9]	Connector	Custom (possible option for V1)
JAUS [7]	Core::Core Services	Default
JAUS [7]	Core::Access Control	Default
JAUS [7]	Core::ID Assignment	Static
JAUS [7]	Core::Transport	JUDP
JAUS [7]	Core::Component Liveness	Default
JAUS [7]	Platform::Platform Management	Basic
JAUS [7]	Custom Discrete State Driver	Consider for V1
JAUS [7]	Custom Discrete State Sensor	Consider for V1

3.3 FUNCTIONALITY AND USE CASE EXAMPLES FOR THE MC

The MC represents each ROBODEXS module, master and slave, as a state machine. Figure 5 illustrates the states and valid transitions of this state machine. Green arrows indicate internal transitions that cannot be commanded by an external source.

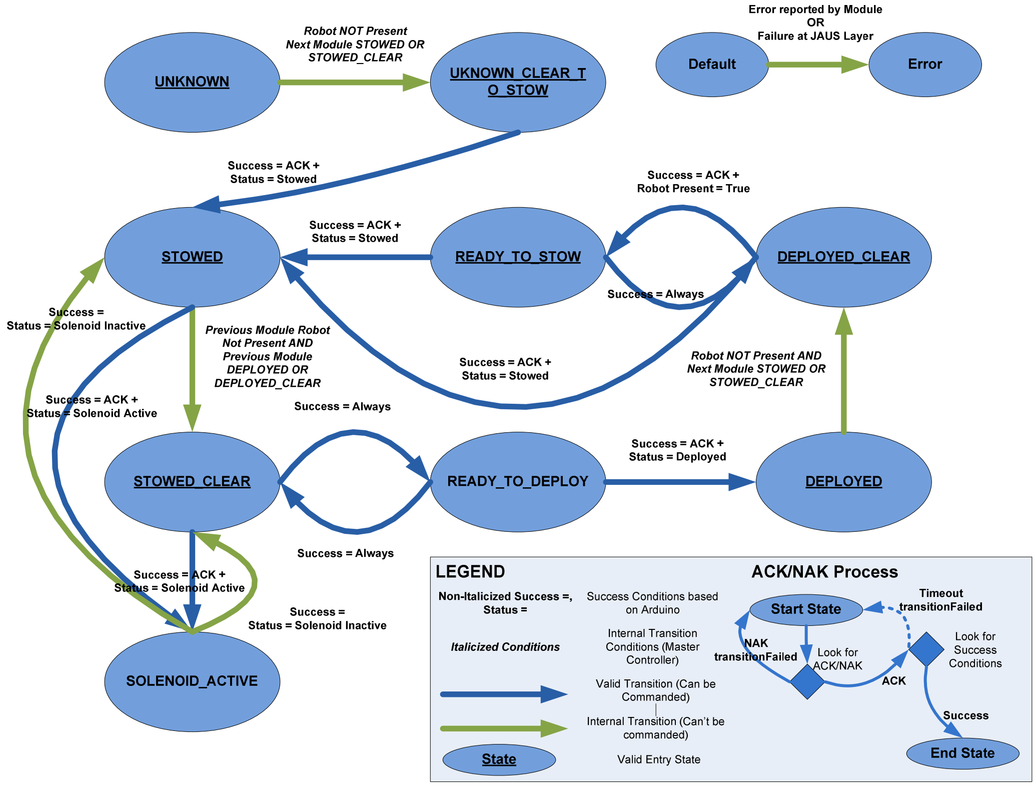

Figure 5. ROBODEXS state transition diagram

3.3.1 DETERMINATION OF MODULE STATE

The Component Manager is responsible for regularly assessing and updating the state of a module (ROBODEXS JAUS Component). The state is determined based on the internal state of the module's IMC and the state of the neighboring two modules. There are 6 internal states that the IMC can report: Stowed, Stowing, Deployed, Deploying, Error, and Unknown. The rule set for determining module state is fairly large, but can be summarized as:

1. If more than one module is moving (transitioning) at a time, those modules go to an error state.
2. If the internal state of the IMC is stowed, then the state of the module is either STOWED or STOWED_CLEAR. Clear is determined by seeing if anything would prevent a module from being deployed (i.e. a module in front of it is not deployed or still contains a robot).
3. If the internal state of the IMC is deployed, then the state of the module is either DEPLOYED or DEPLOYED_CLEAR. Clear is determined by seeing if anything would prevent a module form being stowed (i.e. a module behind it is not stowed completely).
4. The module stays in the SOLENOID_ACTIVE state until the IMC reports that it has completed the solenoid activation process.
5. If a module is clear to stow and a robot is already present, it is considered to be in the READY_TO_STOW state regardless of whether the user commanded it there or not.
6. Error states are propagated – if one module is in an error state, all modules will be set to error state.

3.3.2 EXAMPLES OF DETERMINING MODULE STATE

An example of module state determination using three modules is illustrated in Figure 6. Module 0 is the master module and is at the front of the assembly, while module 2 is the last slave module in the assembly. Module state determinations start from module 0 and go back to module 2. Module 0's state determination depends only on itself and the internal state of module 1, which is behind it. The internal state of the IMC associated with module 0 is "deployed" with "robot present", which means the state of 0 is either DEPLOYED or DEPLOYED_CLEAR. Since the internal state of the IMC associated with module 1 is "stowed", module 0 is assigned a state of DEPLOYED_CLEAR. Module 1's state is dependent on its own internal state as well as that of modules 0 and 2. In this case, module 1's internal state is "stowed". The only way that module 1 could be cleared to deploy is if module 0 is fully deployed with no robot present, so in this situation module 1's state is STOWED. Module 2's state depends on its own internal state and the internal state of module 1 in front of it. Module 2's internal state is also "deployed", so just like module 1, it's state is STOWED because it is not clear to be deployed (blocked by module 1).

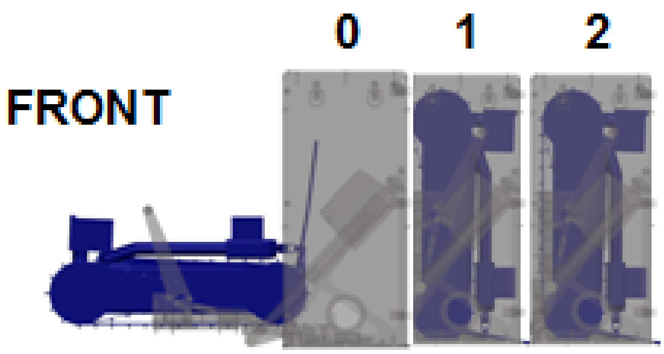

Figure 6. Example of Module State Determination

3.4 SOFTWARE INTEROPERABILITY DEFINITIONS

Definitions for select IOP previously used terms are as follows:

[1] JAUS Subsystem – A JAUS Subsystem is a logical grouping of one or more JAUS Nodes.

[2] JAUS Node – A JAUS Node is a logical grouping of one or more JAUS Components.

[3] JAUS Component – A JAUS Component is a logical grouping of one or more JAUS Services. The input message set (defines part of the interface of the JAUS Service) for any service on a single JAUS Component must be unique.

[4] JAUS Service – A JAUS Service is the lowest level in the JAUS hierarchy, and defines an interface and protocol for a unique capability (i.e. a Global Pose Service provides an interface for querying and setting global pose information). A JAUS Service contains and input and output message set which defines the messages that the service is able to accept (i.e. queries and commands) and to send out (i.e. reports and confirmations). The JAUS Service also defines a protocol behavior, which is a state machine that specifies what the state of the JAUS Service will be given an initial state and some form of excitation such as a received message or an internally triggered event.

4. CONTROL

The development effort for the individual module's control used the traditional V-model development process. Initially, the control of the Slave Modules were drafted using behavior state charts for basic functionality. This technique allows for the programmers as well as the mechanical design engineers to participate in peer reviews of the behavior model. As more requirements were derived and discovered, the behavior state chart was updated to reflect the progress and the review process was repeated. Parallel to the behavior state chart review process, software code was being written for the major states of the model. The state chart is the resultant of the peer review process (Figure 7).

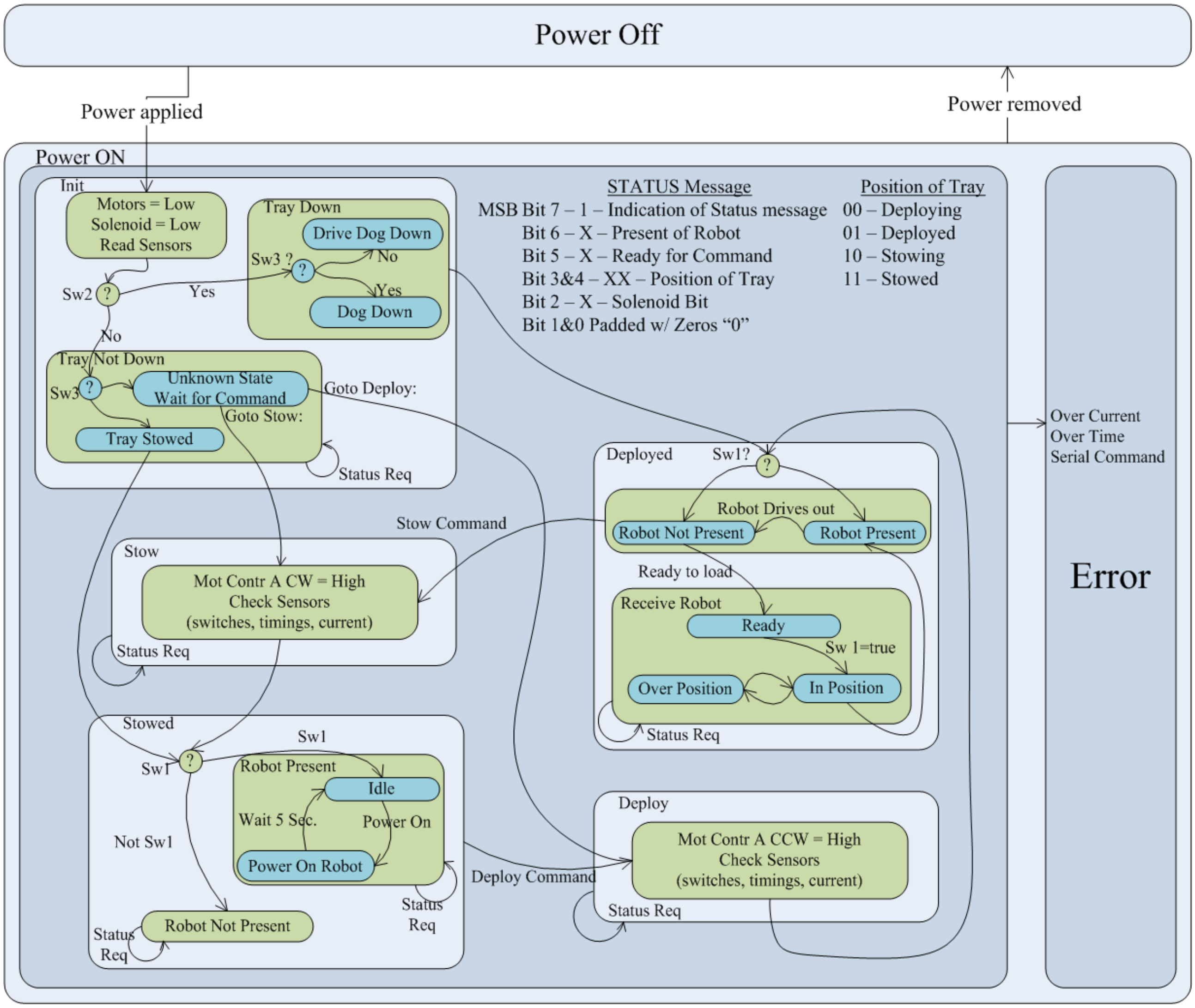

Figure 7. ROBODEXS State Chart

As illustrated in Figure 7, there are five main states in the system. Each state, besides the Initialize state, correlates to a physical configuration of the module. Messages from the Master Controller (MC) or interaction with the robot can cause transition to other states. Once the module enters a state it sends out a status message of the system. When the module receives power, it automatically transitions to the Initialize state. The function of this state is to determine what physical configuration the module is currently in. If the module is in-between states and the configuration cannot be determined through the sensors, the module will wait for the master controller to direct the module to a particular state. This is important especially since the modules overlap with one another in the deployed position.

The heart of an individual module, regardless if it's a Slave Module or the Master Module, is the Individual Motor Controller (IMC). It powers the actuator to physically move the tray that the robot rests upon. The IMC consists of a micro controller that runs the state machine and performs the general purpose input and output and an H-bridge motor driver. The micro controller is programmed in a language that is a subset of C++. The micro controller has an Analog to Digital Converter (ADC), serial communications, and digital input / output pins. For this project, serial communications are performed over Universal Serial Bus (USB).

4.1 COMMUNICATIONS

As the design of the state chart was evolving, the engineering team was also working out the communications protocol between the MC and the IMCs of ROBODEXS. Each Slave Module and the Master Module contain an IMC for deployment and extraction operations. The communication between the units was kept to a minimal. The final command structure includes 6 commands from the MC to the IMC. The IMCs would reply to the commands with 4 unique messages. To ensure that communications between the MC and the IMC are maintained, the status request message is used as heartbeat from the IMCs to the MC. The status message of the IMC uses an 8 bit encoding to convey the position of the tray, robot presences, solenoid position, and overall module readiness.

4.2 SENSORS

There are three categories of discrete sensors on the modules; distance, potentiometers, and position switches. The position switches are used to indicate the position of the tray, clamping dog, and whether a robot is located on the tray or not. The distance sensor checks for an overrun condition during the robot loading procedure. Lastly, the potentiometers are used to measure the position of the actuator in the system. An additional sensor that is integrated into the motor controller is the current sensor for the linear actuator(s). Timing measurements along with reading and interpreting all of the sensors signals is the function of the micro controller which is running the state machine.

4.3 VISUAL INDICATORS

The individual models contain two visual indicators for the user. The indicators are one green Light Emitting Diode (LED) and one red LED. The indicators are positioned in such a way that they are visible to the user through the drive camera of the robot. During start-up, if the tray is in an unknown position, the green indicator will blink until the user selects a known state, stow or deploy. Under normal conditions, the green indicator is used to indicate when the individual modules are ready to load a robot. The red indicator is illuminated when the robot is properly positioned on the loading tray. If the robot over travels on the loading tray, and is out of position, the red indicator will blink while the green indicator remains solidly illuminated. In the event of a fault or error, the red and green LEDs will alternately blink until the reported issue is corrected.

4.4 TESTING

As sections of code were written it was first tested on a breadboard. At this time in the process motor controllers, IMCs were not connected to the module so LEDs were used in place for all the outputs. Switch sensors were simulated by holding an input pin to +5v or ground. This process helped flush out the logical programming errors. As more code was completed, testing progressed to encompass the entire individual module. The state chart was used to derive the initial 45 test cases system test cases. Most of the test cases were verified on the breadboard. The test cases that could not be tested on the breadboard dealt with timings, positioning, and current draw. Once all the breadboard test cases passed, testing transitioned to the real hardware. At this point in the development, operational timing requirements, motor position, and current draws were recorded and analyzed for consistency under various conditions. After the data gathering efforts were completed, limits were established for motor position and current, as well as system timings. If an individual module hits these limits the system is halted and the error code is sent to the MC.

5. FUTURE CONSIDERATIONS

The system described herein has been realized in the form of a single proof-of-concept demonstrator consisting of one Master Module and three Slave Modules. The prototype system has yet to be evaluated by the user community at the date of this writing. Lessons-learned from the development to this point are included below. Functional testing, user community assessment, and added experience with the mass robot deployment concept of operations will likely flush out additional future requirements.

Design changes may be incorporated allowing ROBODEXS to carry different models of robots in the same size class, giving flexibility to a unit desiring to deploy a variety of different robots from a single host platform. Once the concept of operations is proven, the mechanism could be hardened to survive dynamic forces generated within a tactical vehicle driving over rough terrain. The system also has many components that could be optimized for weight savings.

Future control improvements in the system may include an advanced motor control algorithm like Pulse Width Modulation (PWM). To incorporate this feature, the main actuator may need to be replaced with a model that supports this control technique. Some additional control improvements would include detailed position reporting to the user interface and error recoverability with direction from the Master Controller.

There are a few areas of future work related to ROBODEXS MC/OCU software. The newly created JAUS services will continue to be analyzed and evaluated and potentially changed based on that analysis. The services will be presented to the RSJPO IOP JAUS Profiling WIPT for consideration in version 1 of the IOPs. Additional, more robust error and automatic unknown state handling (i.e. automatically moving all unknown state modules into a known state safely) will also be addressed.

REFERENCES

[1] D. Carrol, H.R. Everett, G. Gilbreath, K. Mullens, "Extending Mobile Security Robots to Force Protection Missions", SPAWAR Technical Report No. ADA422161, Defense Technical Information Center, July 2002.

[2] P. Rybski, N. Papanikolopoulos, A. Stoeter, D. Krantz, K. Yesin, M. Gini, R. Voyles, D. Hougen, B. Nelson, M. Erickson, "Enlisting Rangers and Scouts for Reconnaissance and Surveillance", Robotics & Automation Magazine, IEEE, Vol. 7 Issue: 4, Pgs 14 – 24, Dec 2000.

[3] ACER Multi-mission Robotic Platform, https://www.youtube.com/watch?v=3_eM7jE58Mo

[4] Black-I Robotics LandShark with Armadillo Marsupial, http://blackirobotics.com/LandShark_UGV_UC0M.html

[5] 106th Congress, National Defense Authorization Fiscal Year 2001, Public Law 106-398. Section 220, Page 38. October 30, 2000

[6] UGV Interoperability Profile (IOP) Communications Profile, Version 0 (Comms)

[7] UGV Interoperability Profile (IOP) – Overarching Profile JAUS Profiling Rules, Version 0 (JAUS)

[8] UGV Interoperability Profile (IOP) Overarching Profile, Version 0 (Overarching)

[9] UGV Interoperability Profile (IOP) Payloads Profile, Version 0 (Payloads)