Master Console, *SMC*

Héctor J. Pagán Marrero
John F. Kennedy Space Center
Major: Computer Engineering
KSC-FO Fall 2013
Date: 21 11 2013

Table of Contents

Master Console, *SMC*

Héctor J. Pagán Marrero[1]
Inter American University of Puerto Rico Bayamon Campus, Bayamón P.R.

Nomenclature

FR1	=	Firing Room 1
SMC	=	System Monitoring and Control
MCO	=	Master Console Operator
H&S	=	Health and Status
PWS	=	Portable Workstation
DE	=	Display Editor
CUI	=	Compact Unique Identifier
NASA	=	National Aeronautics and Space Administration
KGCS	=	Kennedy Ground Control System
LCS	=	Launch Control System
MC	=	Master Console
CSCI	=	Computer Software Configuration Item
KSC	=	Kennedy Space Center
NE	=	*NASA* Engineering
DWS	=	Development Workstation
ACL	=	Application Control Language
NE	=	*NASA* Engineering
SCCS	=	Spaceport Command and Control System
DSF	=	Display Software Framework
ASF	=	Application Software Framework
SDD	=	Software Design Description
MCPG	=	Master Console Product Group

Abstract

The Master Console oversees the function of Computer Systems in Firing room 1 (*FR1*). Master Console Operators, *MCOs*' for short, are our customer. I was integrated into the System Monitoring and Control (*SMC*) software team that is under the guidance of David Slaiman, who is the product group lead. I have been brought up to speed with System Monitoring and Control. The initial time spent reading *SMC* software design description and understanding how it works. The current Firing Room 1 Console Display is a floor layout giving the *MCO* two essential pieces of information which are Health and Status. When an issue arises, the *MCO* has to look on the display to find which console is affected and then the *MCO* must use the Reference designator from the display to manually search for the Portal Workstation (*PWS*) installed in the console using the hardware map; which is a long process to lookup a *PWS* if an issue is present. My project is to make the *FR1* Console Display easier for the *MCO*'s to pinpoint *PWS*'s without having to lookup additional resources in the process. My project also includes updating Firing Room 1 Console Display to include the F1R Non-Redundant Set. The display does not have good use of space and functionality. *PWS* numbers were not present in the previous design and are the critical component in efficient understanding and administration of the consoles. Part of the process includes getting feedback from the customer, instead of just emailing them with a question, we made a proposal with changes so they could respond and give us their input; which proved to be an effective method for engaging them. In order to do this I had to use the Display Editor (*DE*) tool developed by *NASA*, Paint.Net and Visio. The process I have been using has been Visio to alter the floor layout of Firing Room and take advantage of the white areas, and then I take the altered floor plan into Paint.Net. Once in Paint.Net I put the new floor plan as a background to the standard console display. Paint.Net also serves as a tool to create the new service state bars for the new background; finally I used the *DE* to assemble everything and setup the right *CUI*'s for each *PWS*.

I. Introduction

This report shows you the goals, accomplishments and process underwent to fulfill them, for the Fall 2013 System Monitoring and Control (*SMC*) internship. This process involves security training, accounts necessary for the development work and training for the tools utilized.

The *SMC* software package is an important component of the Launch Control System (*LCS*). The *LCS* is the whole bundle of software that is used in the Firing Room when launching a rocket into space. *LCS* is made of different software products one of them is *SMC*. *SMC* gathers the system Health and Status (*H&S*) from all of the different components in the *LCS*. (Servers, Portable Work Stations (*PWS*), Kennedy Ground Control System (KGCS), etc...)

The goal for *NASA* is to ensure mission success; first because a manned launch mission is expensive, second it is dangerous to launch astronauts on a rocket with at least seventy metric tons of lift. Problems can come from the physical components such as the rockets or capsule used hardware failures such as computer systems failing or software problems such as misconfigurations or just plain software crashes. For this reason *SMC* is necessary to ensure mission success. This helps the Master Console (*MC*) team know when any of these issues arise and they can make informed decisions as to be GO for launch or NO GO for launch.

The goal of my project is to make the Master Console Operator (*MCO*) job more effective by providing clear and concise displays. This consists of making the displays with all the information and functionality needed for the *MCOs* to perform their duties.

II. Body

To accomplish the task of delivering effective displays several things have to happen before one can start developing. Since *NASA* is a Federal Agency there are high security standards along with good security practices implemented in their work force. To accomplish this, security training is essential to all new employees and interns. Once you have your security qualifications you need to set up your computer accounts, because depending on your account, permissions will be assigned and they vary from user to user. The accounts and access needed take a bit of time from the time you request them to the time you get approved because this has to go through different people for approval. This is the time to get acquainted with the project documentation, to better grasp what your product is and how it works. The *SMC* product has a Software Design Description Document which details the capabilities of the Computer Software Configuration Item (*CSCI*). When you have your computer accounts and access set up you start with the training portion of the internship, training is needed because some of the tools *NASA* uses are built in-house and a user from the outside would never have access to them. Stand up team meetings are held each morning to see the status of everyone's work load and offer help if issues arise. After all of this is completed we were able to deliver functional displays like the Firing Room 1 display which helps the *MCOs* with all the tasks they carry.

A. Security Training

Using the *NASA*'s Satern portal, took training courses online to meet the security requirements. The first security training was done before the internship to have access to *NASA* computers, the other trainings were performed during the internship period. These training courses also were necessary to fulfill tasks like installing software on the ACES computer, as well as getting training for some control management tools like Clearquest.

Training Courses:

- Introduction to Information Security for new employees (ITS-013-002)
- KSC IT Security Pre-Briefing (KSC-ITS-PB)
- FY 2013 Annual Information Security Training (ITS-013-001)
- Protecting Sensitive Unclassified and Privacy Information 2013 (ITS-SBUPII-013-R)
- KSC Protecting Sensitive Unclassified and Privacy information for 2013 (KSC-ITS-SBUPII-013)
- NASA NE Information Technology Policies (KSC-NE-ROB)
- Elevated Privileges on NASA Information Systems (ITS-002-09)
- KSC Clearquest User Training – Basic (KSC-SCCS-CQUTB)

B. Accounts Setup

Access to the computer only grants you access to a certain amount of tools, but where I did most of my work wasn't with the tools provided by the ACES machine, it was on the additional tools that are available on *KSC* through web applications or through Linux boxes. To request the setup and creation of the accounts I used the *NASA* Account Management System (NAMS). The most used account is the *LCS* dev account since this is a developing internship, this account gives you access to the development Workstation (*DWS*). From that *DWS* you have access to all your development tools such as the Netbeans Ide used to program Application Control Language (*ACL*) scripts and the Display Editor (*DE*) where the entire graphical user interfaces where built and assembled.

Assigned Resources:

- *KSC* – Rational ClearQuest
- *KSC* AccuRev
- *KSC* RequisitePro
- CCC LCSdeV
- KDDMS – Windchill
- Enhanced ICE General
- ACES workflow for Elevated Privileges
- CCC *SCCS*

C. Documents

To understand the *SMC CSCI* and its capabilities initial time was spent viewing and reading the Spaceport Command and Control System (*SCCS*) presentation and the *SDD* for the *SMC CSCI*. These documents document the capabilities of both *SCCS* and *SMC*. *SCCS* is the whole system which includes *LCS* and *SMC* is a *CSCI* of *LCS*.

D. Tool trainings

Most of the tools utilized for the tasks of my internship are *NASA* controlled tools because of this I had no training or experience working with them. The *SMC* team as well as other *NASA* personnel offered us either training documents/slides on the tool or gave us an actual training.

Tool training:

- ClearQuest training by Jerry Murr
- AccuRev training by Delvin VanNorman
- Application Software Framework (*ASF*) user guide, *ACL* training
- Display Software Framework (*DSF*) Display Editor and Test Driver user guide

E. Team meetings

One of the most important aspects of being a successful computer engineer is having good communication abilities. Good communications abilities enable you to effectively understand your client's requirements and your communication towards your client to see if they are possible or not. Some clients do not have the technical knowledge or know the limitations in what they are asking. With effective communication you can make the client aware of those limitations, so you are not trying to do things out of the available scope of work. Through the course of this internship the Master Console Product Group (*MCPG*) holds meetings every morning to see the team's status on their individual tasks as well clear information about milestones. This has a positive effect since we are on track and reaching our requirements on our way, without effective communication this would not be possible. As part of my tasks I have had to contact people from different subsystems to ask them for advice, explanation and best practices regarding their software/hardware to fulfill my duties.

F. Firing Room 1 (*FR1*) Consoles Health and Status Display

The *FR1* Consoles *H&S* display is the first task I received; this display consists of the *FR1* floor layout with all the components that are inside of it, the *PWSs* as well as printers. The old layout of this screen as seen on Figure 1 only had part of the *FR1* hardware mapped with the right *H&S* CUIs. The old layout only mapped out the redundant set, but *FR1* has two sets the redundant and non-redundant set. The requirements were gathered from my mentor and after the requirements a prototype was built to proof the concept to the *MCOs*. They replied with good feedback and things they wanted to include in this display. This meant updating the *FR1* layout to include both sets, to do this I used Paint.Net image editor. Also the status icons for each *PWS* were small and not numbered, when an issue arises the *MCO* would see the status of a certain *PWS* change state but could not locate the *PWS* just by looking at it, instead they had to look at the reference designator number of that particular desk to locate the two *PWSs* present and figure out which one is the one that is not working. This is a process that can be cut by just adding the *PWS* number on the spot that it is located. The layout size was increase to make better use of the available white space, this way icons are bigger and easier to distinguish (Figure 3and Figure 4). Throughout the process constant feedback was available from my mentor and also from the *MCOs*. This project was finished and implemented into a System Test TCID for 15-1. At the moment we are using the DDE set to test the display with a test driver script.

G. EIC Display Pad B

The EIC Display for Pad B is a display used by the *MCOs* to Activate Data Acquisition (ADA) and Inhibit Data Acquisition (IDA) on the Gateways (GW) connected to the Arbiters on Pad B. This display also requests and releases an arbiter for remote or local commanding. *SMC* will only be able to command when the arbiter is in Remote grant. The physical infrastructure for this display is from the KGCS team they are the ones that control and design the PLC system because of this I had little information on the displays requirements and capabilities. I contacted Katherine Stresau, who is part of the KGCS team, and met with her to talk about requirements and displays they used for their same system. She helped narrow down the requirements and also gave us best practices for that display by giving us the best idea of how the system works and interacts with our system. This display as seen on Figure 5 is still under development and I have developed three prototypes (Figure 6, Figure 7 and Figure 8) to present to the operations team so they can decide what works better for them. Later conversations with David Miller and Guy Bedette have helped us really make a consistent display with accurate information as it can be seen on Figure 9. This displays is the one that meets the requirements and the one going through the approval process. Also this display has *ACL* code running the commanding for the remote request and release since the display is essentially talking to the GW instead of the PLC. Some of the *ACL* code is already implemented since it was used on a previous display but there might be a little bit to add to this display. The *ACL* training has proved good to get around this problem, it is a lot easier than what I thought.

III. Conclusion

I successfully completed the required trainings and requests needed to work in the computer systems at *KSC*. The changes made to the *FR1* Consoles *H&S* display were to the expectations of the *MCOs* and to my mentors' standards. I was successful in updating and implementing the *FR1* Consoles *H&S* Display for testing in the DDE set, as soon as the testing is done and successful will implement it in the final build. The second display that is still in the process of being reviewed and approved for implementation is a more complex display because there is not too much information on it. Having good communication skills helped me gather the right requirements in the best amount of time possible to continue with this display. The training slides on the *ACL* programing proved to be invaluable piece of information for understanding the code needed to execute the request and release commanding and neutral set of the commanding.

My experience as an Intern for *KSC* has been the best work experience that a computer engineer can have. The *MCPG* has been the most experienced and helpful set of engineers in the center. I have developed a better understanding of what it is to work on a team with independent tasks that are all linked together and built. This is a real software engineering process, using iterative software development and CMMI like an industry standard has given me better understanding as a leader of how to run things. I am sure that I would love to work with *NASA*, it is a lot more than just an employer it is a family environment. Being part of the best space agency in the world that is sending rockets and working towards taking humans back into space is just mind boggling.

Appendix

Firing Room 1 Consoles Health and Status Displays

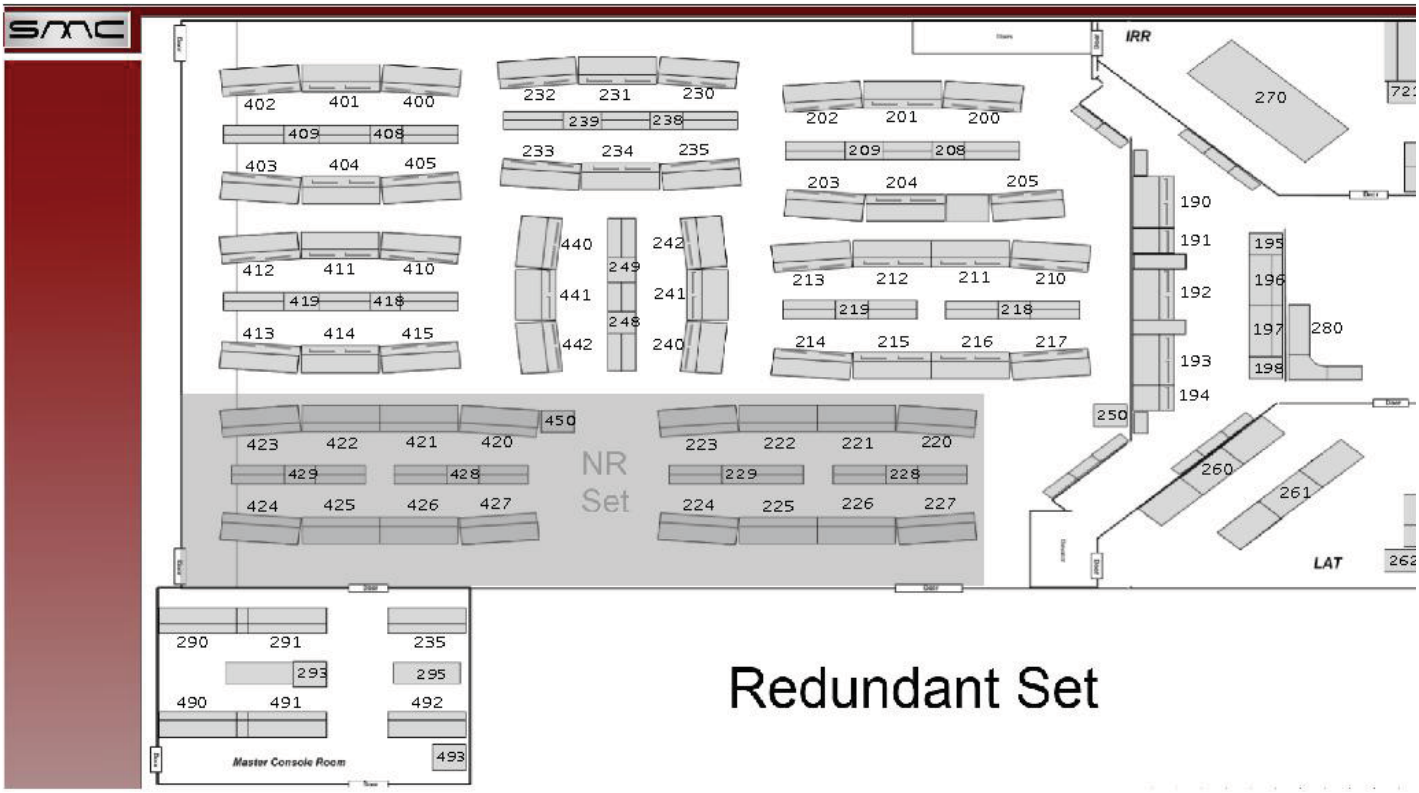


Figure 1Original FR1 Consoles Health and Status Display

This is an image of the original FR1 Consoles H&S display; it shows only the redundant set with the health and status icons. Health icons are transparent on their default state.

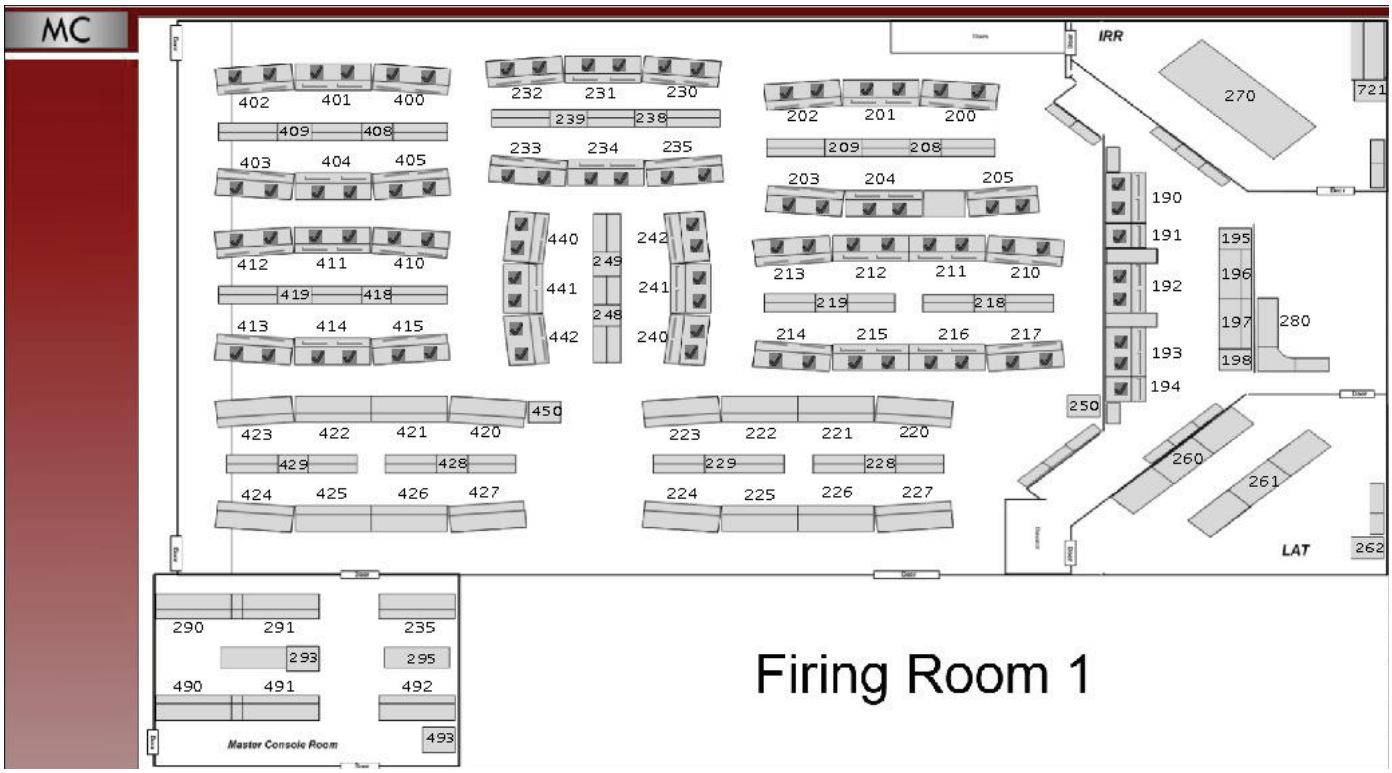


Figure 2 Updating FR1 Consoles H&S Display

This is in the process of updating the display the gray square that covered the non-redundant set is removed, the name of the display is changed to FR1 and on the top left the SMC was removed and it holds the MC now.

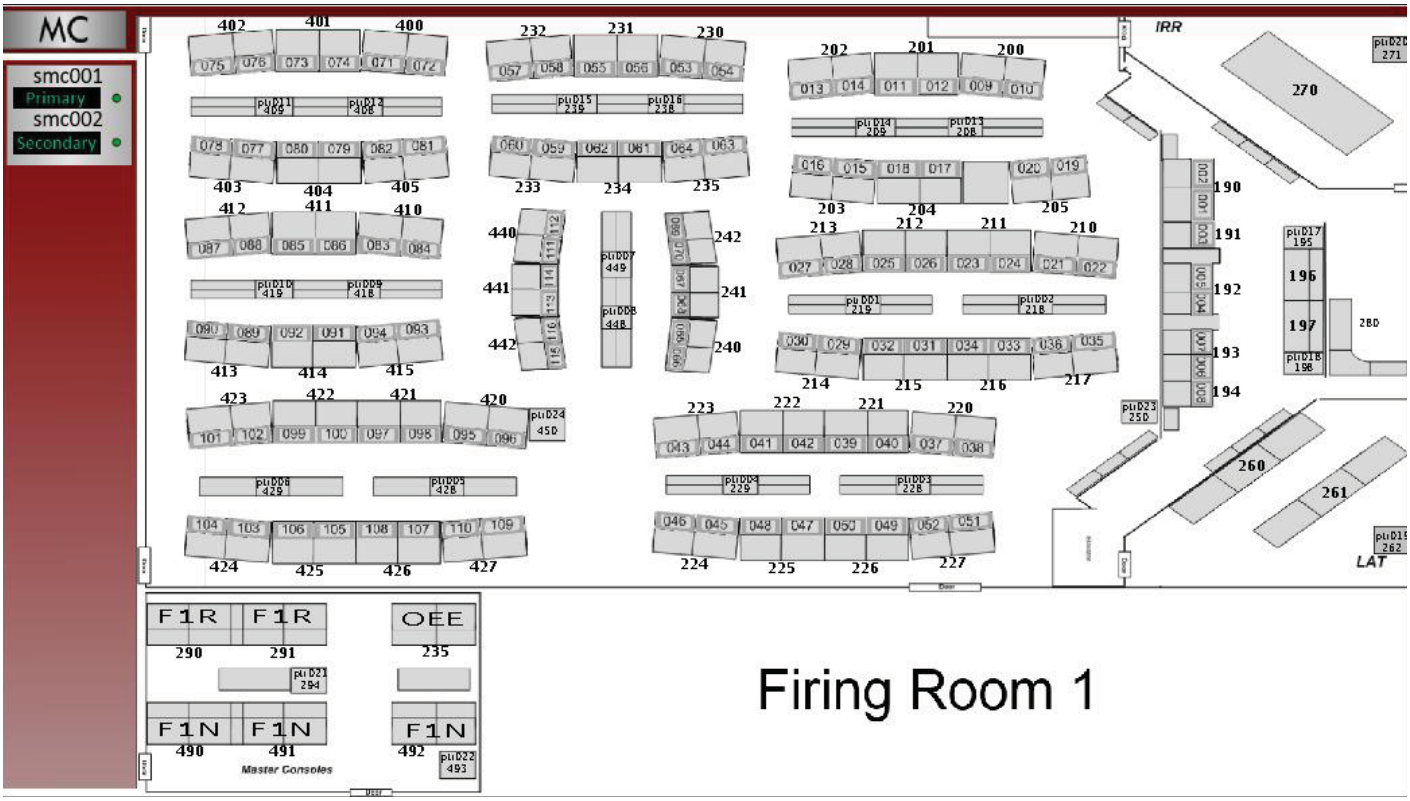


Figure 3 The completed FR1 Consoles H&S display

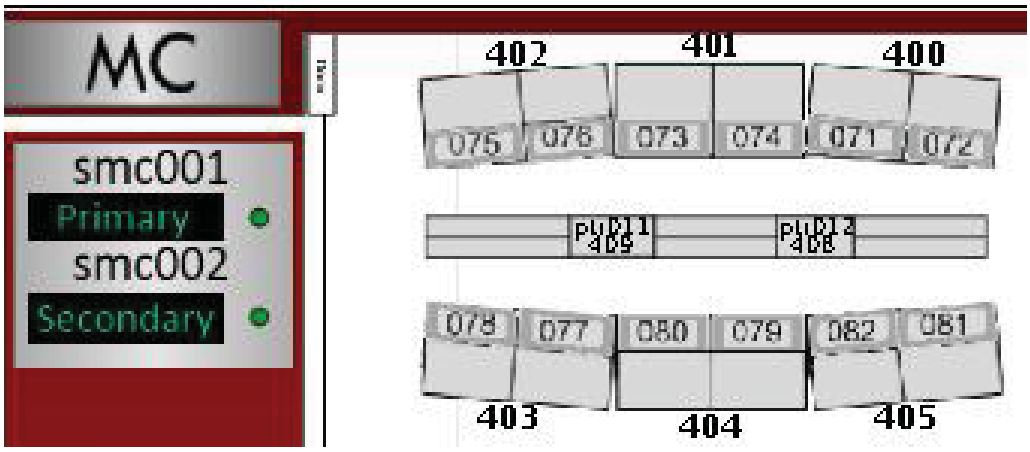


Figure 4 Close Up of cluster F and Heartbeat monitor

The changes made to the FR1 Console H&S display can be seen here, the whole layout was made bigger in order to have a better visibility. The desks have divisor lines so one can easily distinguish between one PWS and another. Each PWS has its number on the desk so the MCO does not need to look up additional information when an issue arises. On the outside of the desks there are the reference designators which help the MCOs locate issues faster. Incorporated a heartbeat monitor on the left side of the screen so that the MCOs know can know if the SMC servers are up and running at any given moment and last printer numbers were included as a request of the MCOs

EIC Displays

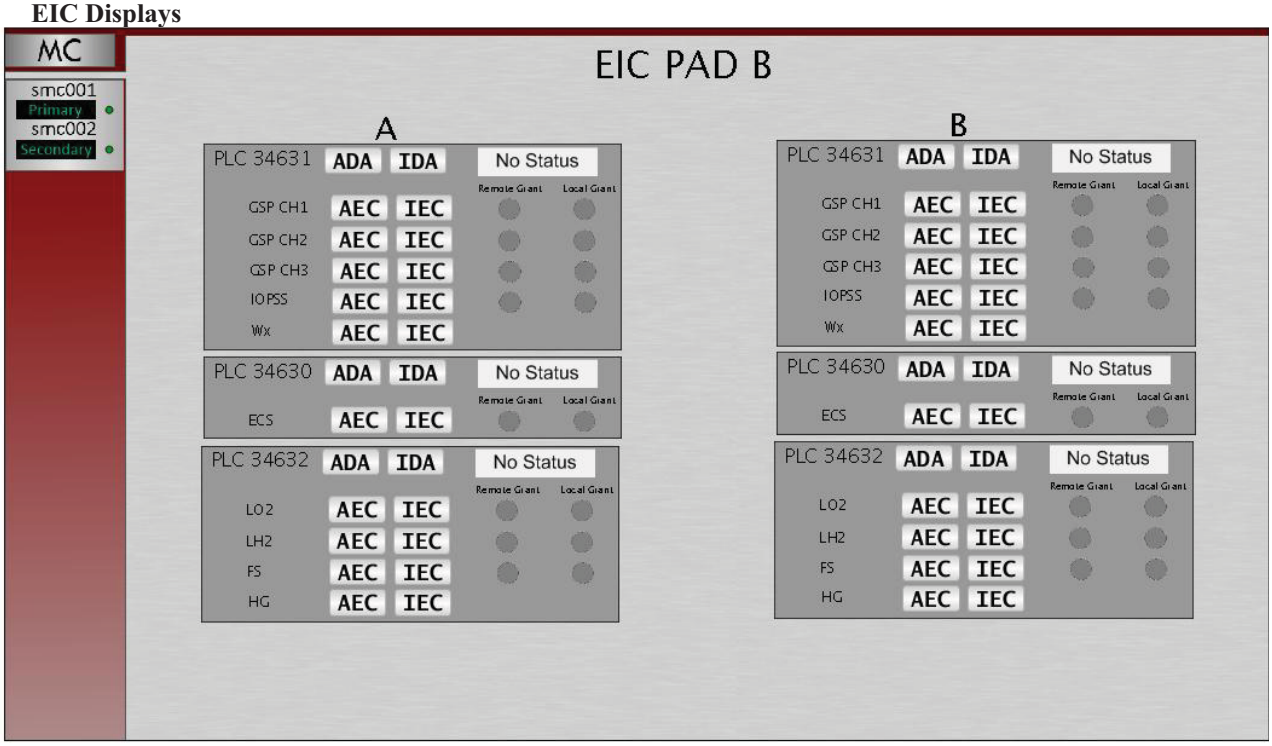


Figure 5 EIC Display First Prototype

This is the first prototype done for the EIC display based on documentation found and email with Katherine Stresau. It has two sets of PLCs and each with its own set of arbiters. It gives the MCO the option to Activate data acquisition as well as enable remote commanding to a certain arbiter.

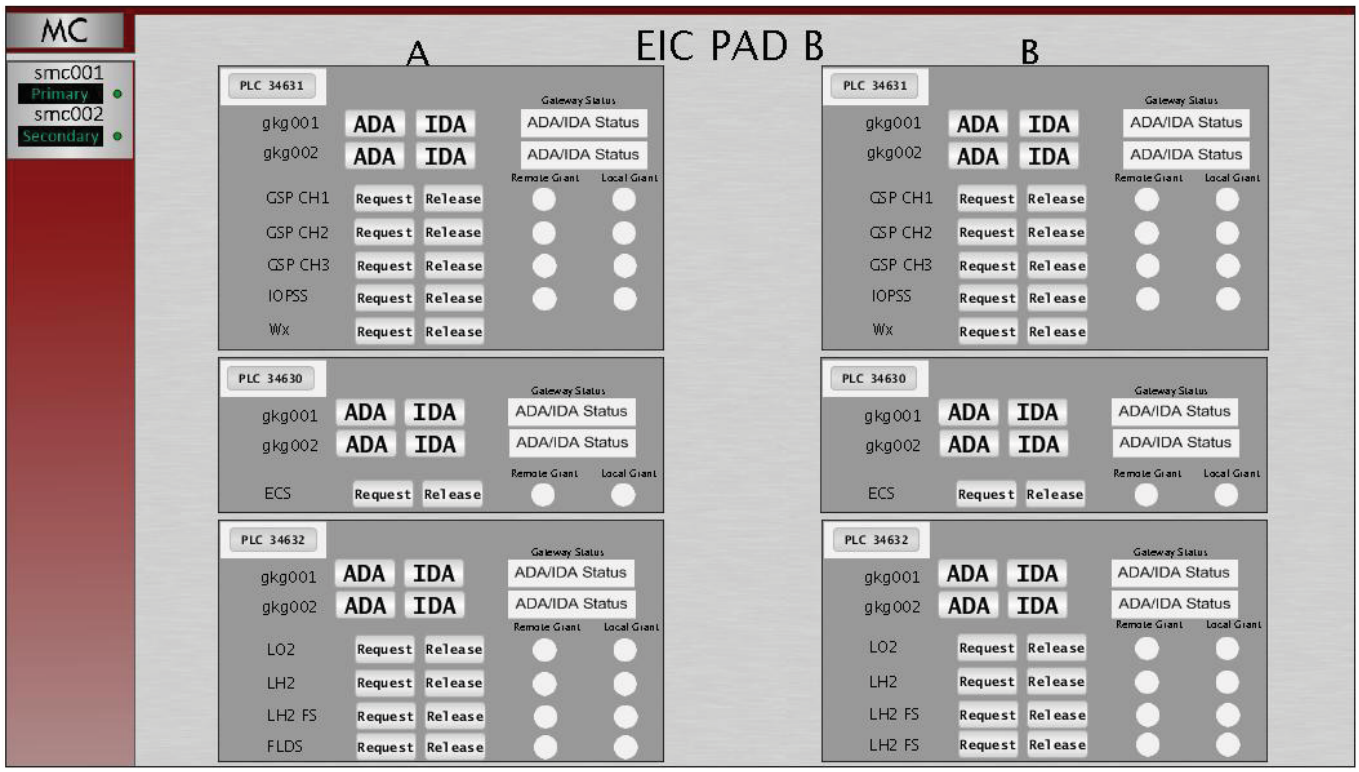


Figure 6 Second EIC display prototype

On this display PLC H&S is displayed like on the KGCS PLC roll up Health display. There is now an ADA and IDA for each gateway and the status. The AEC and IEC have been changed to request and release which is what KGCS uses.

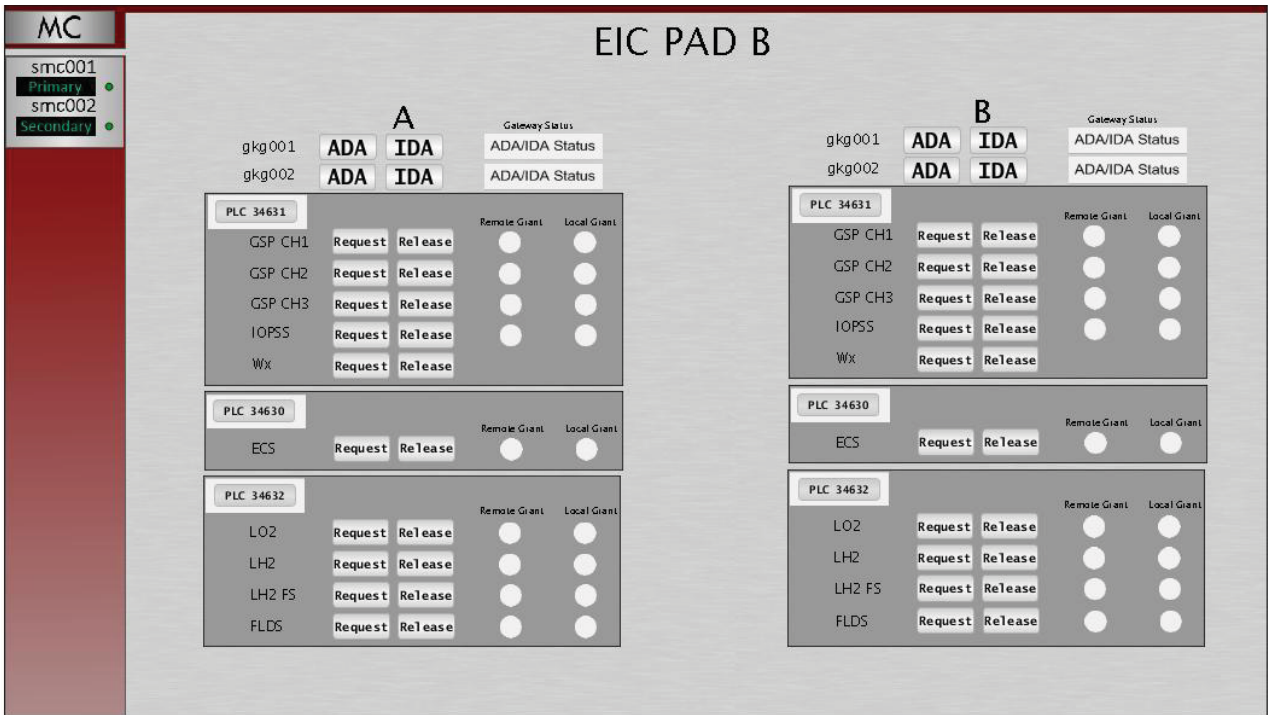


Figure 7 Third prototype for EIC display

This display is like the previous mentioned model but the ADA and IDA are taken out of each PLC and you have one general for each set.

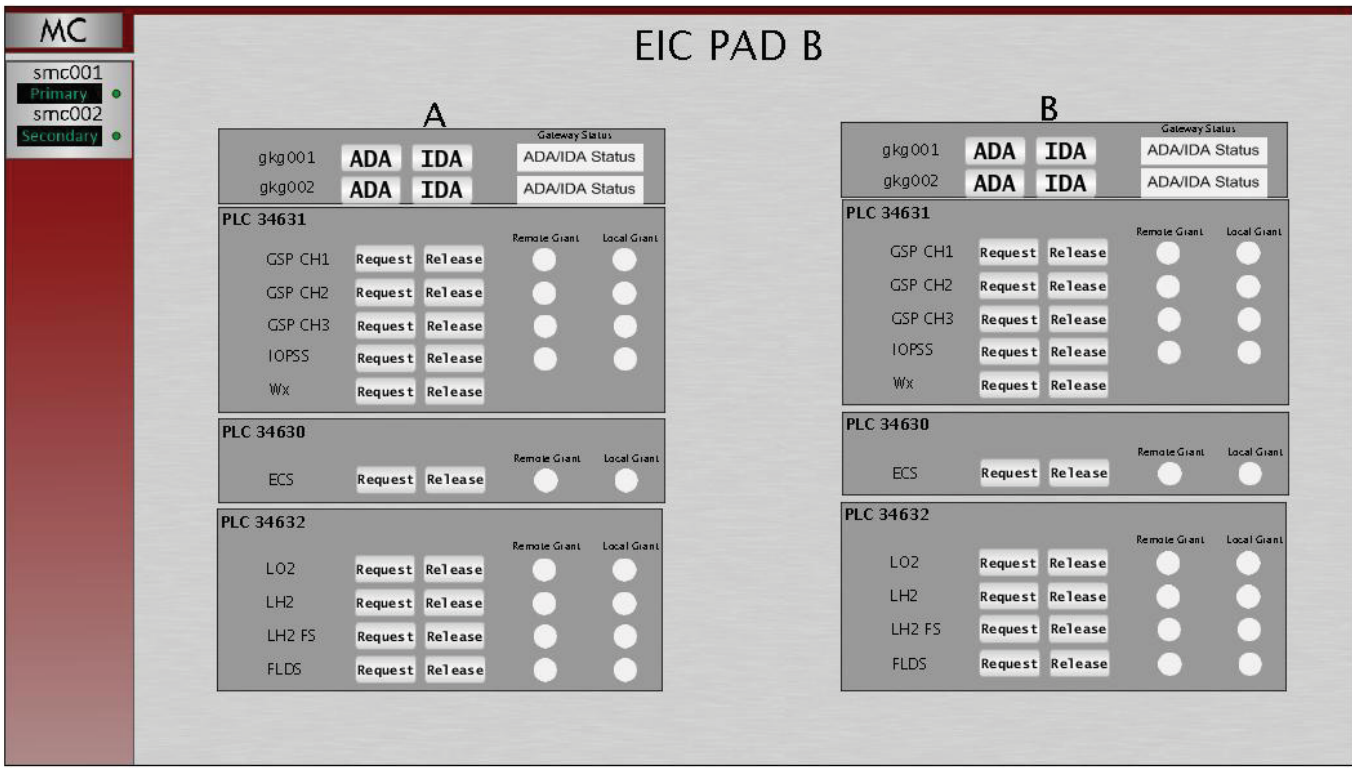
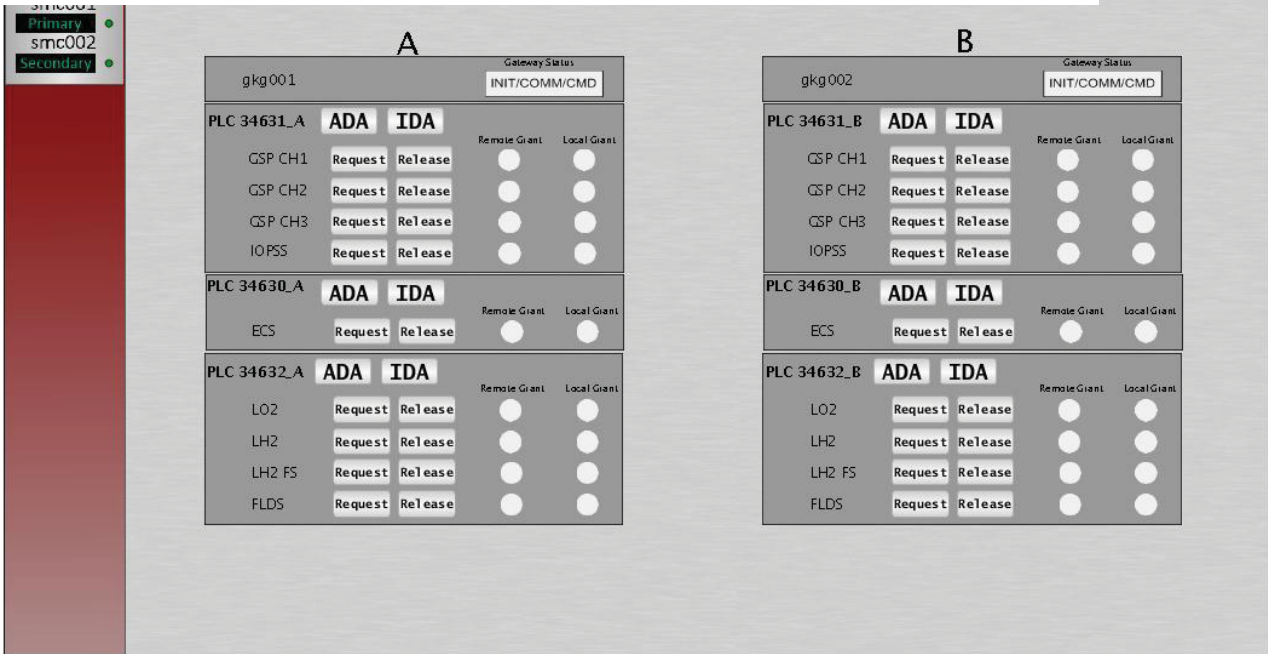


Figure 8 Fourth Prototype of the EIC display

This display is like the previous but to improve the visibility of the gateways ADA and IDA section a box was incorporated into it.

Figure 9 Fifth Prototype EIC display



This is the latest prototype and the one that best addresses all the requirements. There are two sets of PLCs the A and B set; each set is in communication through a gateway. The A set uses GW 1 to communicate and the B set uses GW 2 to communicate, it also shows the gateway status whether it is INIT/COMM/CMD. Each PLC has its own ADA/IDA because they can be activated separately. The weather arbiter has been removed because at the moment it does not have a CUI and it is not clear if it will be included. This display also shows the PLC number along with the letter that tells to which set it belongs. The rest is similar to previous prototypes.

References

Miller, D.T., "System Monitoring and Control Computer Software Configuration Item," *NASA* GOP 507091, 2011.

Hall, D., "Spaceport Command and Control System (SCCS),"NASA,K000143172-PLN , 2013.