REVISIONS

REV #	ECN #	Description of Change	Engineer	Approved	Date
A	NDI	New Document	EPF		

Description

Software Requirements for Relay Box Controller

1.0 INTRODUCTION AND OVERVIEW

1.1 Purpose

This Software Requirements Document of the Relay Box Controller is written to provide details required to write the system software for the embedded microcontroller

1.2 Responsibility

The Design Engineer responsible for the product is responsible to keep this document accurate and make sure that all engineers working on various aspects of the product always have the latest copy to work from.

1.3 Definitions and Acronyms

CMS	Cabin Management System
PA	Public Address
A/V	Audio/Video
LL	Low-Latency
TBD	To Be Defined
DSP	Digital Signal Processor
LI	Logical Input
HLI	High (+28VDC) logical input
NO	Normally open
NC	Normally close
Relay	Electrically operated switch
optocoupler	A light-emitting diode (LED) coupled with a photo transistor
µC	Microcontroller

1.4 References

Product Specification for Relay Controller, DWG 107800PS

1.5 Product Overview

The Relay Box controller is an independent module that controls the cabin lights, audio, and other equipment. It receive commands from the CSS keypads modules.

Provide a replacement unit for a legacy Pac Sys 1054-1-1 component. Plus, it forms the basis of a forward fit controller with provisions for future databus control.

1.6 Functional Block Diagram

The Figure 1.1 shows the functional block diagram of the Relay Box Controller with the different subsystems it's composed and how are they connected.

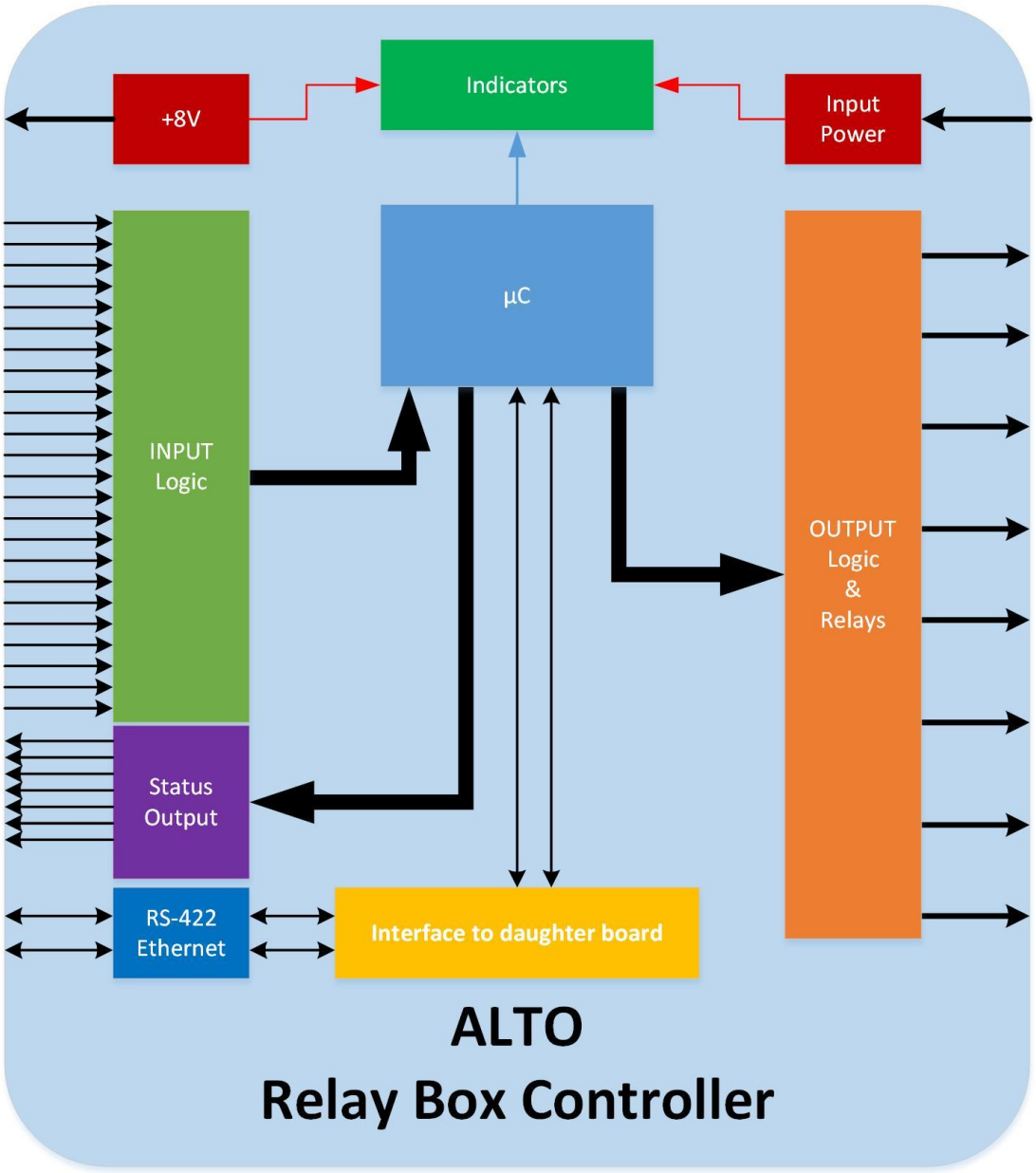

Figure 1.1

1.6.1 Circuit Block Description

Referring to Figure 1.1, the block diagram shows the basic functionality of the product.

1.6.2 Functional Blocks and Connection Definition

1.6.2.1 Output Logic & Relays

The output logic & relay block on the diagram represent the output relays and the logic needed to drive the relays. It receive commands from the µController and activate the corresponding relay. Every Relay Box Controller incorporate 8 relays. The output connector includes the NO (normally open) and NC (normally close) connections per relay. Every relay shall handle 5Amp. Figure 1.2 shows how the µController commands a relay through a driver.

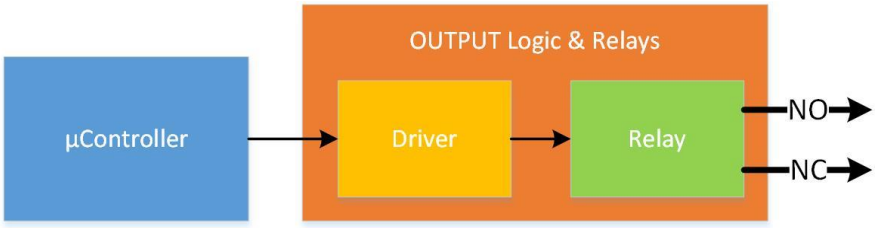

Figure 1.2

1.6.2.2 Input Power

The relay box controller can be supply by 3 different input power pins. The Input Power block on the diagram represent the electronic circuits that will handle the power supply. These three inputs will be combine and generate 3 voltages for the internal circuitry.

- **+28V/+5V**. This is voltage will be used to supply the input logic pull-up resistors. The microcontroller will select either +28V or +5V for all the pull-up resistors.
- **+8V**. This voltage will be used to supply external modules.
- **+3.3V**. This is going to be the main power supply for the electronics on the board.

It also will include input voltage monitors connected to the microcontroller so the microcontroller can select which outputs can be active depending on the input power supply.

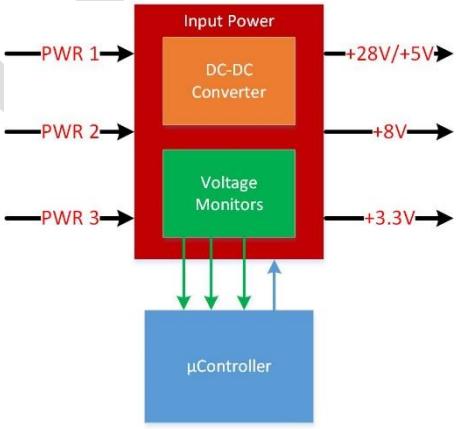

Figure 1.3

1.6.2.3 Input Logic

The input logic block on the diagram accept up to 27 input signals. Every input can be configure to be pull-down or pull up independently. All the pull up resistors are connected to the +5V power supply. There are 8 special inputs can be selected to have 28V pullups.

Figure 1.4 shows the logical connection between the input logic and the microcontroller. The input logic is composed by 27 input channels and every input pin is connected to their corresponding input channel. Every input channel implement a Configurable Logic Unit (or CLU). The function of the CLU is to configure every pin independently as pull down or up with the corresponding voltages and isolate optically the inputs from the microcontroller side as shown in Figure 1.5. The microcontroller is responsible of configure every pin independently to function as the particular application requires.

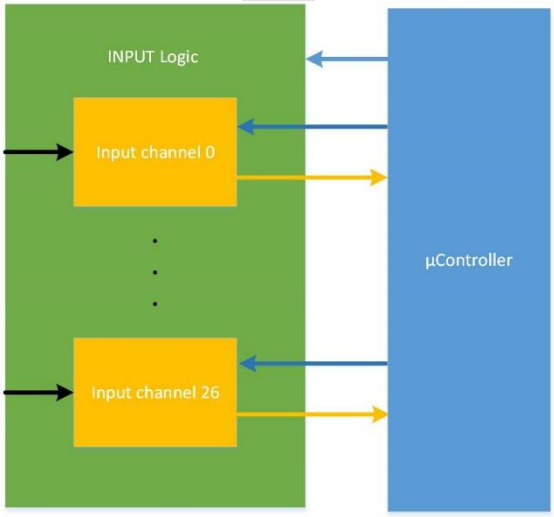

Figure 1.4

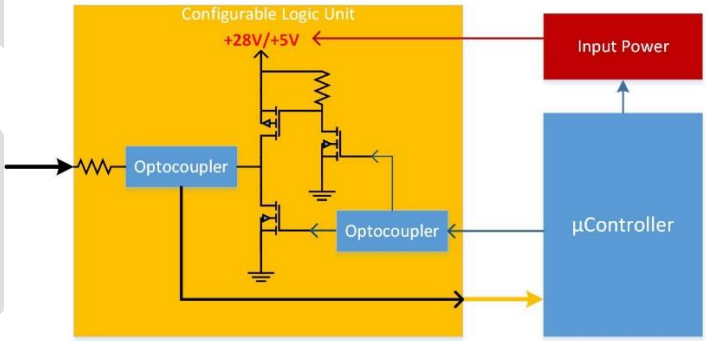

Figure 1.5

1.6.2.4 Status Output

This block covers the electronic that handles the status indicators on an external module. The relay box controller can handle up to 8 indicators and can be configure by software. Every indicator output has an open collector interface so the output module requires to have a pull up resistor.

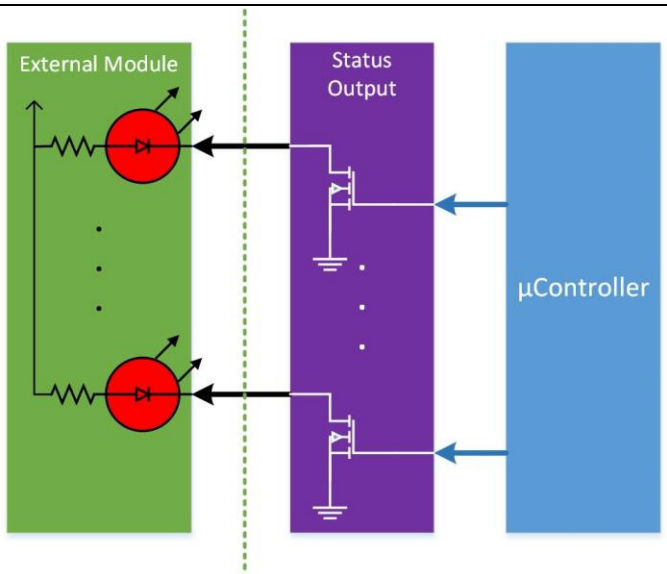

Figure 1.6

1.6.2.5 +8V Output

The relay box controller provides +8V output on the input connector to power external modules when needed.

1.6.2.6 RS-422 & Ethernet

The RS-422 & Ethernet block on the diagram represent the serial interface of the Relay Box. The RS-422 can be used to send commands to the relay box, read status information and configure the unit. The Ethernet interface is connected to the interface to a daughter board for future implementation.

1.6.2.7 Interface to daughter board

The interface to daughter board is included to add new functionalities in the future.

1.6.2.8 Indicators

The indicators will be used as a way to show the status of the Relay Box Controller.

- **+3.3V**. Green - Indicates there is power from the 3.3V regulator running the uCom.
- **Fault**. Red - blinks to indicate faults according to codes.
- **Activity**. Blue - blinks when activity on input or output happens (.25 sec ON)

1.6.2.9 µController

The microcontroller is the brain of the Relay Box Controller. It receives input from all the input signals, and the power supply monitors and act on the relays according to the configuration programmed. Optocouplers are used to isolate control and controlled circuits.

2.0 SOFTWARE REQUIREMENTS

2.1 Software functionality.

The core of the design is a PIC 32MX460F512 micro controller. The software design is based around the proven architecture used already in the Alto DA and DX amplifier series. The primary input and output to the unit is performed using the 27 discrete logic input, 8 relay outputs, and 8 status output. The microcontroller can be configured in various modes to take input from the discrete inputs and trigger the relays and status outputs according to various functional algorithms.

- Discrete logic inputs have selectable pull up (or pull down) either ground or 28VDC active. Each input can be set for active high or low by software. (28VDC active required for Pac Sys replacement)
- LED outputs will be ground active, open collector. The LED state generally will follow the action of the relay. When the NO/COMM are shorted, the LED output will be active.

2.2 PIC Pin assignments.

2.2.1 Output Status indicators (active low triggers light)

PIC port IND_OUTX RB0 to RB7 (8 outputs) O

2.2.2 Relay Output (active low triggers relay)

PIC port RELAY_OUTX RB8 to RB15 (8 outputs) O

2.2.3 Input Discrete matrix. Input pins are address selected, then the input port is read. Inputs are in 4 banks of 8 inputs each for bank A,B,C, and 3 inputs for D

PIC port SEL_INX discrete input address RE0 to RE2 (select input 0 - 7) O
PIC port SEL_INX input bank address RE3 to RE4 (select bank A - D) O
PIC port DISCRETE_IN discrete input data RE9 (read discrete input pin status) I

2.2.4 Input Discrete pullup/dn latches. Pullup latches are address selected, then the latch is written.

PIC port SEL_INX pullup/dn address RE0 to RE2 (LSBs bank latch X0 – X7) O
PIC port PULL_BIT pullup/dn set bit RE7 O
PIC port PULL_RESET pullup/dn reset bit RE8 O (pull low to reset all latches)
PIC port PULL_LD0 pullup/dn Bank 0 Load bit RG13 O (low to hi to latch)
PIC port PULL_LD0 pullup/dn Bank 1 Load bit RG12 O (low to hi to latch)
PIC port PULL_LD0 pullup/dn Bank 2 Load bit RG15 O (low to hi to latch)
PIC port PULL_LD0 pullup/dn Bank 3 Load bit RG14 O (low to hi to latch)
PIC port PULL_+5/28 Select 5 or 28V RD7 O (hi = 5V)

2.2.5 Aux Indicator LEDs. These are 4 red LEDS used to show something???

PIC port AUX_IND0 LED indicator (High for light) RA3 O
PIC port AUX_IND1 LED indicator (High for light) RA2 O
PIC port AUX_IND2 LED indicator (High for light) RG2 O
PIC port AUX_IND3 LED indicator (High for light) RG3 O

2.2.6 UARTs

PIC port UART1_RX (RF2)
PIC port UART1_TX (RF8)
PIC port UART2_RX (RF4)
PIC port UART2_TX (RF5)

2.2.7 SPI Ports

PIC port SPI1_MISO (RC4)
PIC port SPI1_MOSI (RD0)
PIC port SPI1_SCK (RD10)
PIC port SPI_FLASH_CS (RC2)
PIC port SPI_EEPROM_CS (RC1)
PIC port SPI2_MISO (RC4) (unused)
PIC port SPI2_MOSI (RD0) (unused)
PIC port SPI2_SCK (RD10) (unused)
PIC port SPI2_CS (RC2) (unused)

2.2.8 Voltage monitoring

PIC port MADDR0 (RD11) O (LSBs)
PIC port MADDR1 (RD12) O
PIC port MADDR2 (RD13) O
PIC port MADDR3 (RD14) O (bank1)
PIC port MADDR4 (RD15) O (bank1)
PIC port MINT (RD9) (interrupt when over or under V happens) I
PIC port MONITOR (RD8) (hi if voltage OK) I

2.3 Configuration Table

The configuration table contains functional definitions for each of the 27 Input pins, and some additional general settings. Config table is queried and written to over Altonet. Sample as follows

Index	Parameter	Value	Low-Lim	Hi-Lim	Default	DataType
0	Function_Input1	1	1	8	1	DISCRETE
1	Target_Input1_Relay	1	1	8	1	TARGET
2	Target_Input1_Status	1	1	8	1	TARGET
3	HoldDel_Input1	0	0	255	0	DELAY
4	Param_Input1	0	0	255	0	PARAM
5	Pullup_Input1	0	0	1	0	ON/OFF
…						
162	Power1_Mask	0	0	255	0	MUTE
163	Power2_Mask	0	0	255	0	MUTE
164	Power3_Mask	0	0	255	0	MUTE
165	Pullup5or28	0	0	1	0	ON/OFF

2.3.1 Config type definitions

2.3.1.1 Function_InputX

2.3.1.1.1 1= Toggle. Single press turns on the target relay on, second press off, etc.

2.3.1.1.2 2=Momentary. Detect a transition to on and the target relay is turned on, transition to off turns target relay off.

2.3.1.1.3 3=Set-ON. Single press latches relay on. Stays on until Set-OFF issued or another input toggle, or master Off.

2.3.1.1.4 4=Set-OFF. Single press clears any latched on relay.

2.3.1.1.5 5=Sequence. Single press turns on first relay in sequence. Further press turns off current relay and on next relay. Repeat until relay limit parameter reached. Cycles around to first after limit.

2.3.1.1.6 6=Binary Count. Single press turns on target relay. Further presses triggers a binary bit count using relays of higher number until count limit is reached. Press after count limit starts over at initial setting (0)

2.3.1.1.7 7=OneShot. Single press latches relay on for duration specified in parameter field.

2.3.1.1.8 8=Master Off Reset. Single press clears any on-latched relays. Even a momentary switched relay gets cleared.

2.3.1.1.9 9=Hotcup. Single press latches relay for duration specified (similar to OneShot), with differences: Companion output status indicator blinks off once every 5 seconds, When remaining count is less than 10 sec then blink goes to once every 0.5 sec. Press & hold more than 1 sec and timer/relay is reset. A single press received during an already running Hotcup countdown will pause the count down and shut off the relay. During this time the LED will blink twice within .5 sec then off for 1.5 sec. A second single switch press resumes countdown and re-engages relay.

2.3.1.2 Target_InputX_Relay Target is a bit mask allowing targeting more than 1 relay. Bits 0-7 selects relay(s) as the target of the corresponding input action.

2.3.1.3 Target_InputX_Status Target is a bit mask allowing targeting more than 1 status line. Bits 0-7 selects Status line(s) as the target of the corresponding input action.

2.3.1.4 HoldDel_InputX

2.3.1.4.1 0-255 specifies required press and hold time of an input before it is considered triggered. Time is in 20's of mS, so a value of 20 = 400mS.

2.3.1.5 Param_InputX

2.3.1.5.1 For Toggle, Momentary, Set-ON, Set-Off: N/A

2.3.1.5.2 For Sequence: Low nibble specifies the first relay in the sequence, high nibble specifies last relay in sequence.

2.3.1.5.3 For Binary Count: Specifies the upper count limit (max 255)

2.3.1.5.4 For OneShot: Specifies the duration of the one shot hold, in 100's of mS (so value of 25 = 2.5sec.)

2.3.1.5.5 For Hotcup: Specifies the duration of the timer, in 10's of seconds (so value of 25 = 250 seconds – 4 min 10 sec).

2.3.1.6 Pullup_InputX

2.3.1.6.1 0 Sets the corresponding input to pull-down – requiring active high input

2.3.1.6.2 1 Sets the corresponding input to 5V pull-up – requiring active low input

2.3.1.7 PowerX_Mask

2.3.1.7.1 Bit mask to inhibit relays based on the presence of voltage on the power1,2,3 lines. If a relay# is included in the bit mask then it will not activate if the power is not detected on the specified power line.

2.3.1.8 Pullup5or28

2.3.1.8.1 0 indicates a 5V pullup for the pull-up on the dual level inputs (Inputs A, pins 16-19, 34-37).

2.3.1.8.2 1 indicates a 28V pullup for the pull-up on the dual level inputs (Inputs A, pins 16-19, 34-37).

2.4 Altonet commands for operation

Commands that apply to this unit include:

2.4.1 Prepare for Restart

2.4.2 Restart

2.4.3 Device Detailed Status (re-defined)

2.4.4 Config Data Parameter

2.4.5 Manufacturing Info

2.4.6 RelayControlSequence (added to).

2.5 Operation

2.5.1 Startup, initialize, read Config table

2.5.2 For all input lines, select pullup register address, output PULL_BIT based on config setting, Toggle corresponding PULL_LDX line to latch.

2.5.3 Set PULL_+5/28 based on config Pullup5or28.

2.5.4 Do any setup prep based on the Function settings of each input.

2.5.5 Loop task polling all Input lines.

2.5.6 Monitor the Voltage settings.

2.5.7 If a line transition is detected, check HoldDelay setting, set timer to insure Input is held through hold delay time.

2.5.8 Apply function specified by Input settings to relays (considering power mask settings

2.5.9 Likewise, if AltoNet command comes in process it and apply action specified.

2.6 Contention

It is understood that incorrect configuration could cause conflict situations, such as when 1 relay is the target of more than 1 input actions. The configuration of this device is under control of Alto, therefore the Alto engineer needs to insure that it is properly configured without possibility of adverse conflicts. The software can be written without any specific contention handling hindsight.