Radionics, Inc.

D229 Relay Module
External Design Document

Revision History

Version	Author	Date	Remarks
00.00	Paul L. Calinawan	12/28/95	Initial Document
00.01	Paul L. Calinawan	1/15/95	Added Heart Beat & DIP Switch

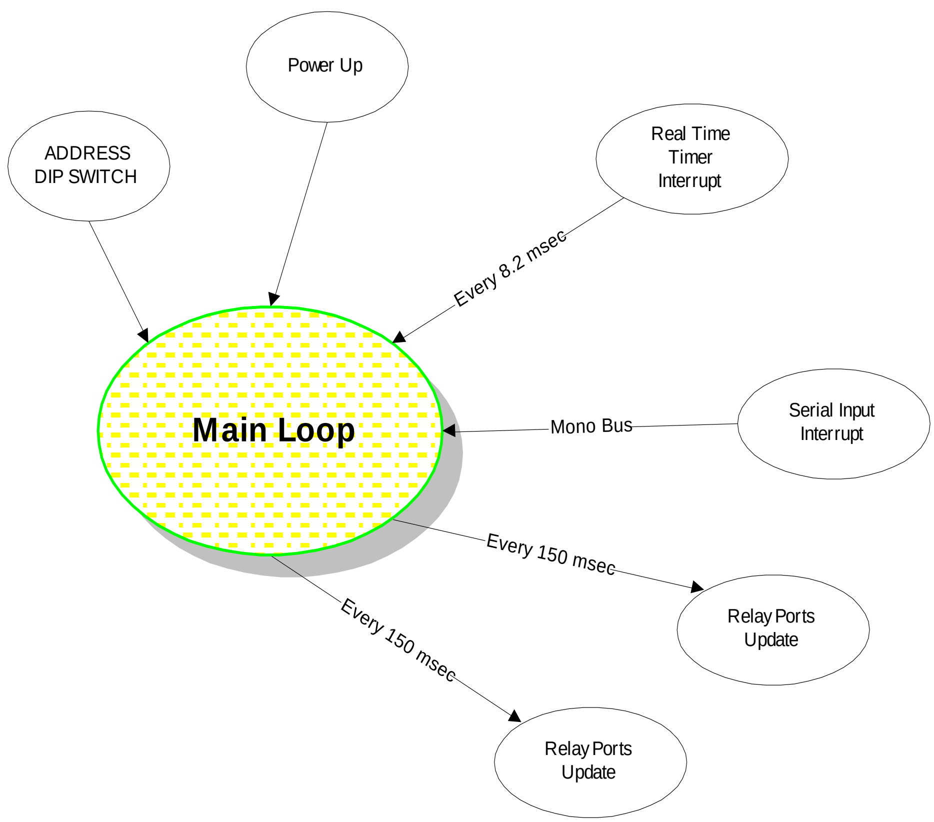
1. Overview

The D229 is an external relay board module designed to work with the D2412 panel @2400 BAUD MONO-BUS speed. This module gives the end user the option of using more than the 2 onboard relays provided by the first release of the D2212 system.

The idea is not new. Octo-relay modules as they are called, are being produced for our mid as well as our high-end products. These modules are generally used for added annunciation capabilities. Others are even being used to provide some form of home automation (sprinkler control) through the use of “automated sked” functions. A module such as the D229 will be a great addition to the D2212 lines of accessories since the D2212 platform will be moving towards the field of home automation.

The general software structure has been based on the D202A. The main loop responds to the MonoBus Communication Interrupts as well as the 8.2 millisecond timer interrupts. Relay states are updated at the rate of 150 msec.

D229 Context Diagram



2. Port Assignments
(Motorola 6805J1A)

Port A

Pin	Direction	Connection
PA0	Output	Relay Number 1
PA1	Output	Relay Number 2
PA2	Output	Relay Number 3
PA3	Output	Relay Number 4
PA4	Output	Relay Number 5
PA5	Output	Relay Number 6
PA6	Output	Relay Number 7
PA7	Output	Relay Number 8

Port B

Pin	Direction	Connection
PB0	Input	Mono Bus Data In
PB1	Output	Mono Bus Data Out
PB2	Input	DIP Switch No 1[1]
PB3	Input	DIP Switch No 2
PB4	Input	DIP Switch No 3
PB5	Output	Heart Beat LED

[1] This switch configures the Address of the Device.

3. Interrupts

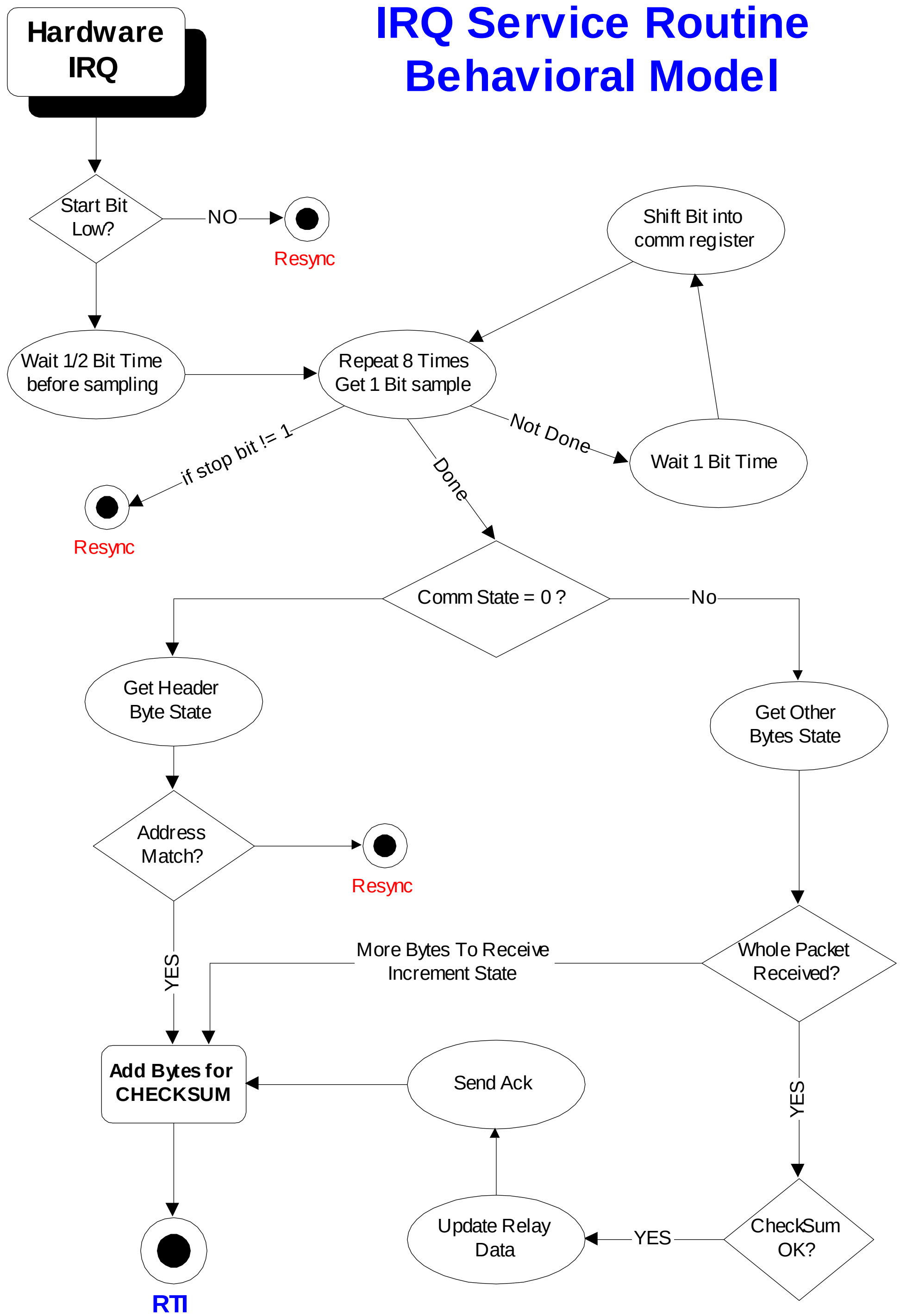
The only interrupts that are available in the MC68HC05J1A are the real-time Interrupt (Timer) and the IRQ Interrupt (Hardware). The RTI can be selected for periods of 8.2 ms, 16.4 ms, 32.8 ms, and 65.6 ms. The D229 keypad uses the 8.2 ms rate. This is the same timer rate as that of the D202A. This timer rate has been primarily selected to allow communication resynchronization while allowing tasks for the main loop to be executed.

The IRQ interrupt is setup to trigger on the negative edge transition of the MonoBus. A software UART is designed for handling the IRQ routine when the start bit is detected (low logic) on PORT B0. It ignores any succeeding interrupts, as it reads in the 8 data bits (closed loop) to form a byte of data. The routine then checks that the stop bit is a logic high before the byte received is considered valid.

The IRQ interrupt routine then checks for the correct ADDRESS BYTE. If the wrong header byte is received, the software will throw away all serial data for 8.2 msec (ReSynchornization). Once a valid ADDRESS is received, it then collects the whole packet and is not finished until the last bit of the byte is sent. The last byte or the CRC BYTE is finally checked for correctness before the packet is considered valid. Further processing is done once a valid packet is received.

To detect a MonoBus communication failure, a counter is incremented every cycle of the main loop (8.2 msec). This counter is cleared when a valid packet is received. If no valid packet is receive within 60 seconds, all outputs of PORTA are cleared thereby turning off any relay that was active.

IRQ Service Routine Behavioral Model



4. Serial Interface

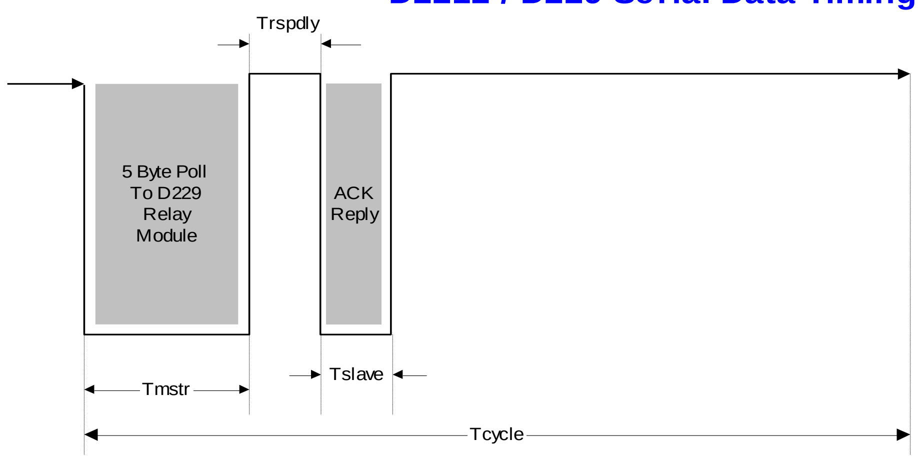
4.1 D2212 / D229 Overall Serial Timing

The D2412 employs a 2400 baud ASYNCHronous serial data to communicate with peripheral devices on the serial bus (MonoBus). Peripherals would include LED keypads, Alpha Text keypads, local programmers, and any future peripherals.

The bus operates in master-slave mode at 2400 baud, in true logic wire-and, i.e., mark - data = 1 - line is high. D229 Relay Modules will be polled at a rate of approximately 450 msec. Any device on the bus can pull the line down to zero. It is the responsibility of the software as well as the hardware to ensure that the bus is released upon completion of the transmission by the peripheral.

The communication protocols for each peripheral will be defined in subsequent Peripheral Protocol Documents.

D2212 / D229 Serial Data Timing

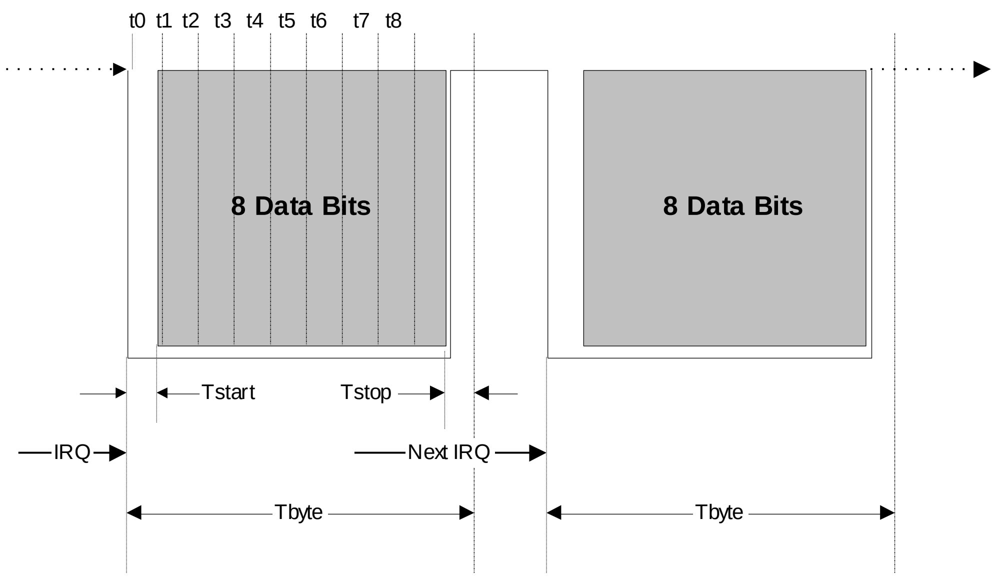


		Time (Milliseconds)
Tcycle	Panel polling rate	450
Tmstr	Panel poll length	41.62
Trspdly	Relay Module response delay	8
Tslave	Relay Module transmission time	8.3
Tmstrlstn	Panel waiting for response	MonoBus Scheduler ?

4.2 D229 Serial Receive Timing

The D229 Relay Board Module employs a software UART to receive data over the MonoBus. The Serial Receive is triggered by the IRQ interrupt activating on a start bit. Once the start bit is detected, the software waits in a delay loop for 1.25 bit times. The software will then samples for the first data bit and then delays for 1 bit time. Following bits are sampled in the same fashion. The eight-bit sample will be formed into a byte of data and the byte data will be saved. The software then checks that the stop bit is logic high. If the byte received is the last byte in the expected packet (CRC BYTE), the message is then processed. Otherwise, the software will return to the non interrupt level and wait for the next IRQ interrupt.

D229 Serial Receive Data Timing
(2400 BAUD)



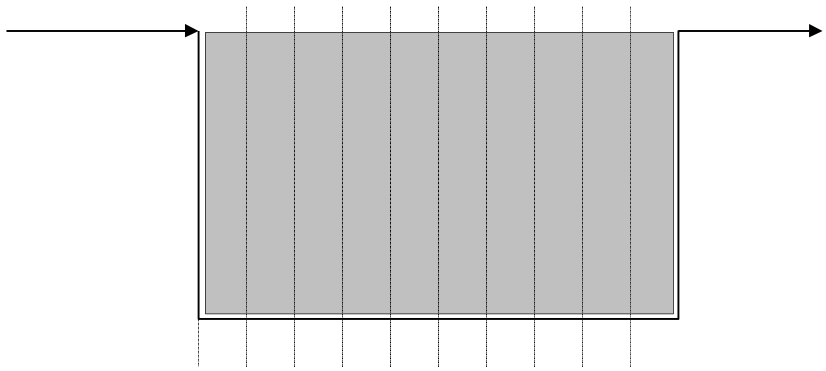
Tbyte	Byte time
Tstart	Start bit time
t0 through t8	Sample times
Tstop	Stop bit time
Tinterbyte	Panel inter-byte time

The table below shows the nominal timing of the interrupt routine when it receives a message byte.

	Time (Milliseconds)	Possible Deviation
Tbyte	4.166	- 2 Microseconds
Tstart	0.4166	- 2 Microseconds
Sample Times	Reference to IRQ Going Low	
t0 (Start bit)	.0675	- 10 Microseconds
t1 (Data Bit 1) [2]	.475	+/- 1 Microsecond
t2 (Data Bit 2)	.8885	+/- 1 Microsecond
t3 (Data Bit 3)	1.302	+/- 1 Microsecond
t4 (Data Bit 4)	1.7155	+/- 1 Microsecond
t5 (Data Bit 5)	2.129	+/- 1 Microsecond
t6 (Data Bit 6)	2.5425	+/- 1 Microsecond
t7 (Data Bit 7)	2.956	+/- 1 Microsecond
t8 (Data Bit 8)	3.3695	+/- 1 Microsecond
Tstop	0.4166	- 2 Microseconds
Tinterbyte	0.0 < Tinterbyte < 0.4166	- 2 Microseconds

[2] t1 - t8 are times when the individual bits are being sampled.

The Ideal Byte Timing :



This table show the ideal timing of an incomming byte @ 2400 BAUD:

	Time in milliseconds
StartBit	0
t1 (Data Bit 1)	.416
t2 (Data Bit 2)	.832
t3 (Data Bit 3)	1.248
t4 (Data Bit 4)	1.664
t5 (Data Bit 5)	2.08
t6 (Data Bit 6)	2.496
t7 (Data Bit 7)	2.912
t8 (Data Bit 8)	3.328
StopBit	3.744

2400 BAUD Calculations :

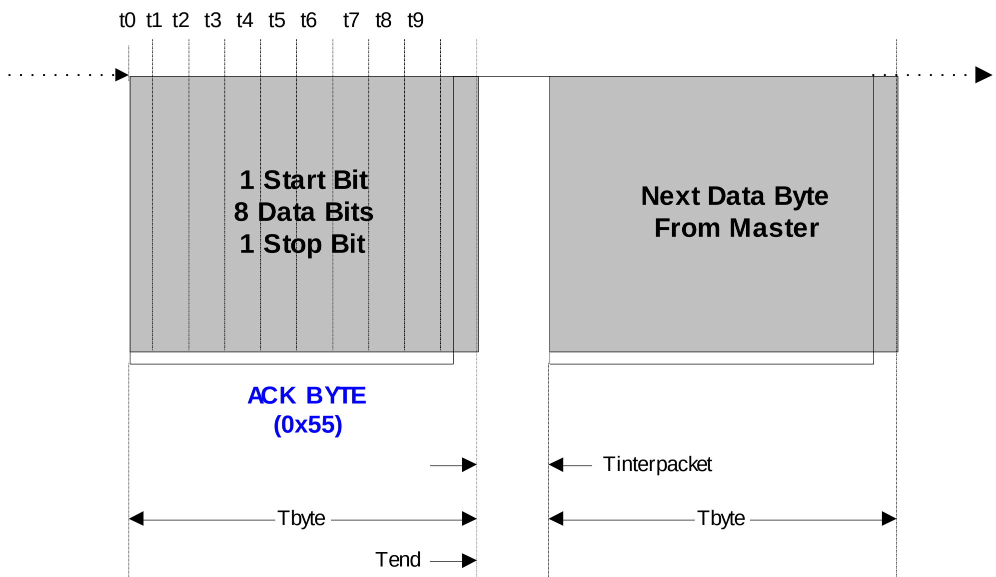
2400 Bits / Second = 1 Bit / 416 μsec

2 Mhz MCU freqency = 0.5 μsec / MCU instruction cycle

4.3 D229 Serial Transmit Timing

The D229 Relay Board Module employs a software UART to transmit data over the Serial Data Bus. The Serial Transmit routine is a non-interrupt machine that is triggered by a valid panel message being received by the D229 Module. The D229 can only transmit an ACK (0x55) message in response to a valid packet with a matching Device Address.

D229 Serial Transmit
Data Timing
(2400 BAUD)



Tbyte	Byte time
t0 through t9	Bit times
Tend	Complete byte sent
Tinterpacket	MonoBus inter-packet time

	Time (Milliseconds)	Possible Deviation
Tbyte	4.166	+/- 40 Microseconds
Bit Times	Reference to Transmit Flag	
t0 (Start bit) [3]	0.000	
t1 (Data Bit 1)	.416	+/- 4 Microsecond
t2 (Data Bit 2)	.833	+/- 4 Microsecond
t3 (Data Bit 3)	1.2495	+/- 4 Microsecond
t4 (Data Bit 4)	1.666	+/- 4 Microsecond
t5 (Data Bit 5)	2.0825	+/- 4 Microsecond
t6 (Data Bit 6)	2.499	+/- 4 Microsecond
t7 (Data Bit 7)	2.9155	+/- 4 Microsecond
t8 (Data Bit 8)	3.332	+/- 4 Microsecond
t9 (Stop bit)	3.7485	+/- 4 Microsecond
Tend	4.165	+/- 40 Microseconds
Tinterbyte	.080 < Tinterbyte < 1.20	+/- 80 Microseconds

[3] The reply is sent after waiting 8 msec. from the last valid message.

4.4 Communication Protocol

The information the Relay Module receives looks like this:

Position	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**
Description	Address	Length	Relay Data 1	Relay Data 2	Checksum
Value	0x0A	0x03	0x00 - 0xff	0x00 - 0xff	cc

The first byte is the device ID, and the second is the number of bytes to follow. The checksum is the value that when all the bytes are added together comes out to 0xFF. The length of this packet is fixed as 5 bytes. Byte 3 contains the data for relays 1-4 while Byte 4 contains the data for relays 5-8.

The Relay Data is packed as follows:

Relay Number	Byte Number	Data Bit Position	Mask Byte
1	3	(MSBit) 0000 00XX (LSBit)	0x03
2	3	0000 XX00	0x0c
3	3	00XX 0000	0x30
4	3	XX00 0000	0xc0
5	4	(MSBit) 0000 00XX (LSBit)	0x03
6	4	0000 XX00	0x0c
7	4	00XX 0000	0x30
8	4	XX00 0000	0xc0

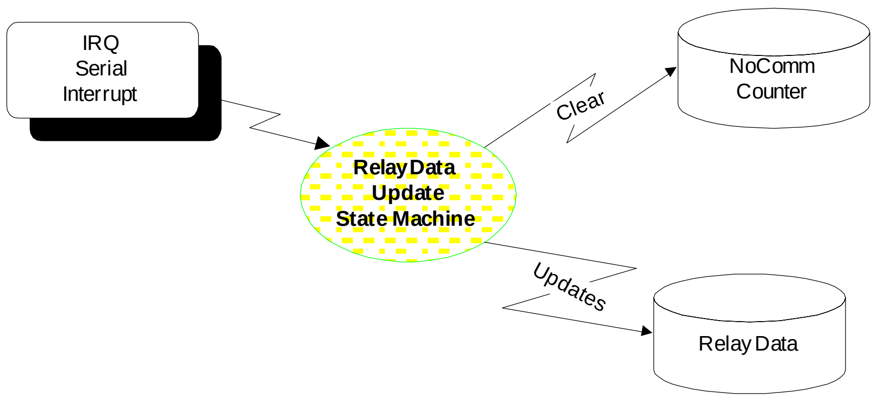
There are four possible Relay States:

Relay Data	Relay State Definition
0x00	Steady
0x01	Pulsed
0x02	Off
0x03	Undefined

5. Software State Machines

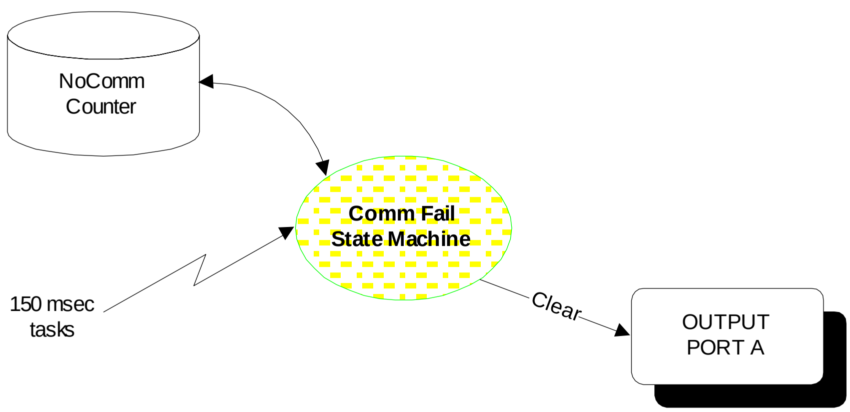
5.1 Relay Data Update Machine

The Relay Data Update state machine along with the IRQ state machine is in charge of refreshing the relay data variables. Every time a valid message is received from the MonoBus, the relay data is extracted from the packet and stored. The Relay Data Update State Machine is in charge of clearing the NoComm Counter everytime a valid packet is received.



5.2 Comm Fail Machine

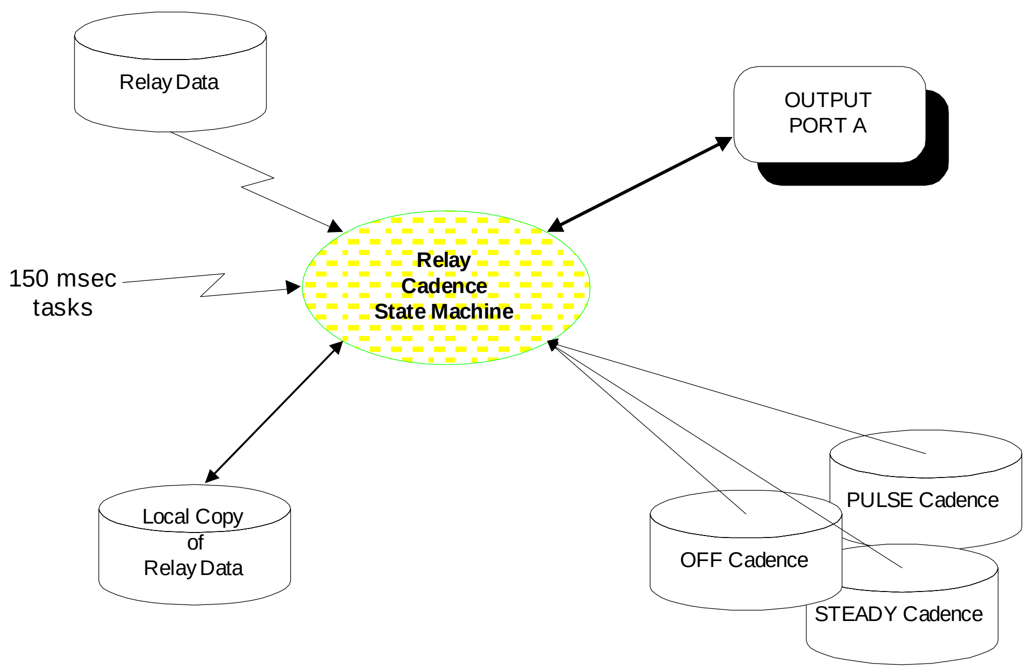
Ideally the device is to be polled on every MonoBus cycle. When the 150 msec tasks are executed, the Comm Fail State machine keeps track of the number of times that the device did not receive a poll or a valid packet from the panel. If the device determined that it has been failing for 25 seconds, it goes into a communication fail state. In which case, all of the relay outputs are deactivated.



5.3 Relay Cadence Machine

The Relay Cadence State Machine is responsible for controlling the output states of each of the relays. This state machine is also executed every cycle of the 150 msec tasks. This state machine retrieves the Relay Data and executes its cadence, i.e., STEADY, PULSE or OFF for each of the relay.

The Relay Cadence State Machine does not retreive the data unless it has completed a cadence cycle. The current design simply stores the cadence as the bit pattern in a constant byte. This later be expanded if more complex cadences were required.



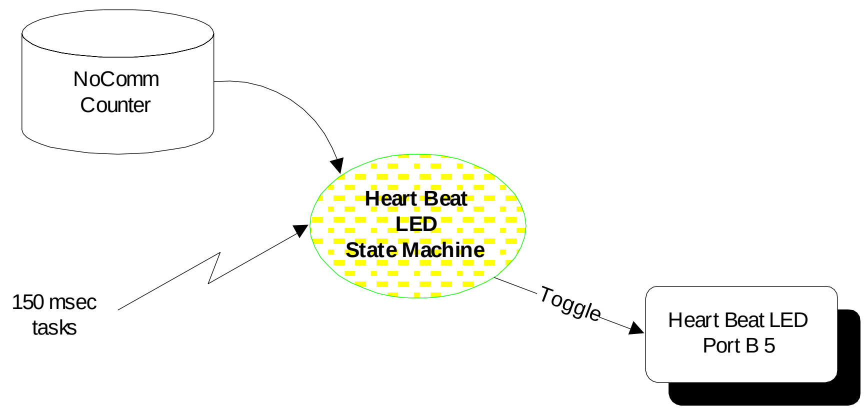
Cadence Data :

Cadence Type	1	2	3	4	5	6	7	8
OFF	0	0	0	0	0	0	0	0
STEADY	1	1	1	1	1	1	1	1
PULSE	1	1	1	1	0	0	0	0

SCALE : 150 msec

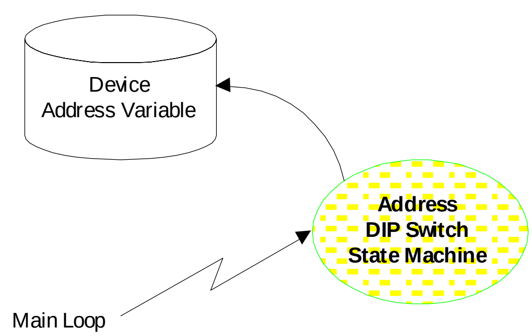
5.4 Heart Beat LED Machine

Every 150 msec interval, the “Heart Beat” LED is toggled. The toggling only stops when the device is not polled for more than 1 second. The heart beat indicates valid messages are being received and the device is working properly.

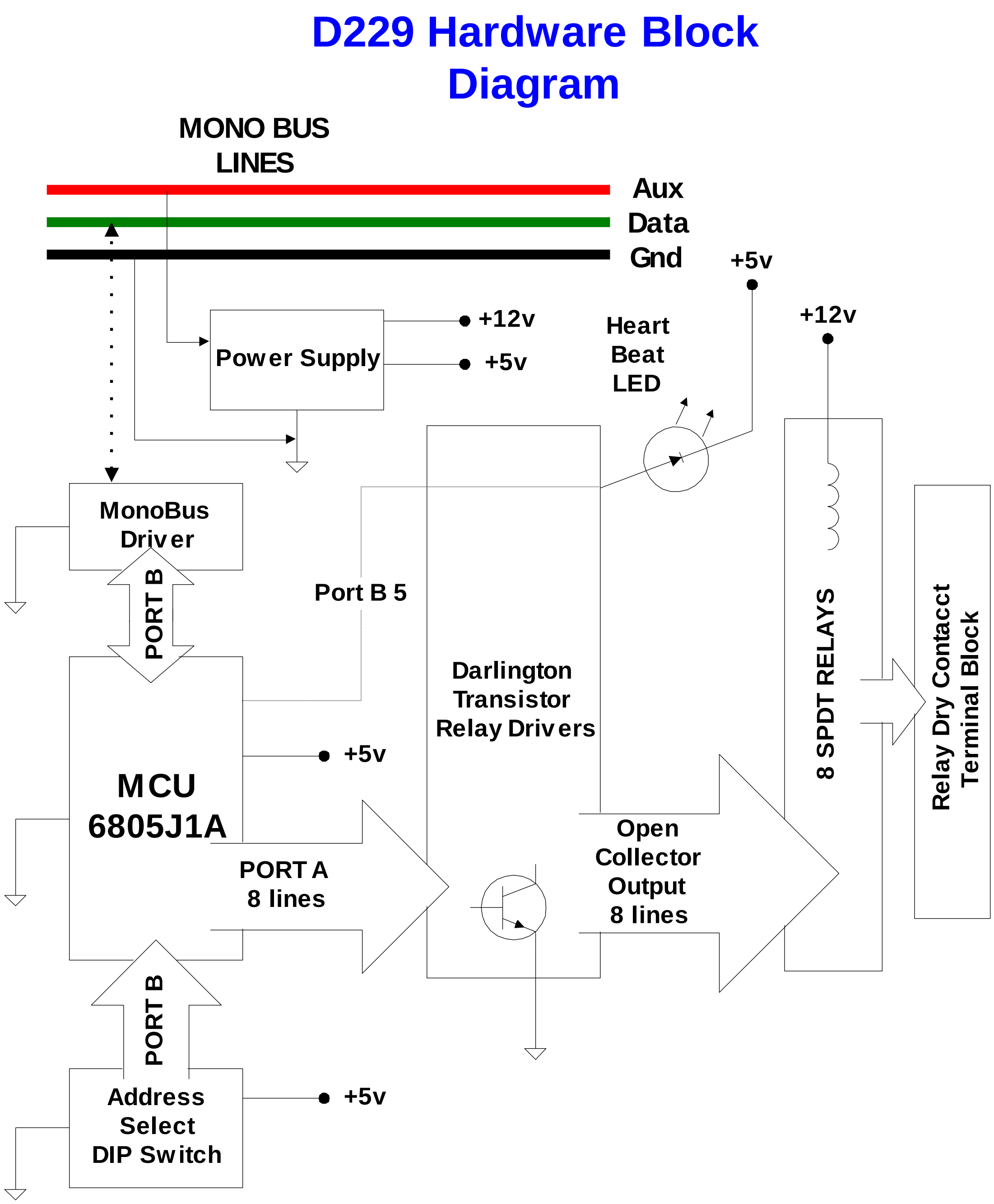


5.5 Address DIP Switch Machine

The address of the device can be selected with the use of a 3-position DIP switch. The address is stored in ram and is updated every cycle of the main loop. Address can be changed on-the-fly. The DIP Switch address calculation is being done within the main loop to avoid latencies in the communication interrupt routine.



6. Hardware Block Diagram



6.1 DIP Switch Configuration

A 3-position DIP switch is provided for device address selection. The table below shows the possible settnings:

DEVICE ADDRESS	DIP SWITCH 1	DIP SWITCH 2	DIP SWITCH 3
10	OFF	OFF	OFF
11	ON	OFF	OFF
12	OFF	ON	OFF
13	ON	ON	OFF
14	OFF	OFF	ON
15	ON	OFF	ON
16	OFF	ON	ON
17	ON	ON	ON

7. Issues