*e*Prox™

Design Specification

For evaluation by prospective licensees

This document is made available to prospective licensees of HID *e*Prox™ technology under the terms of the HID Non Disclosure agreement. The information it contains may be used for no other purpose than the evaluation of the technology with a view to entering into a licensing agreement with HID.

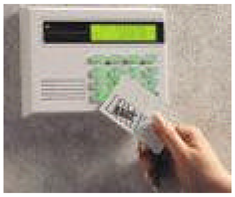
The information contained in this document is preliminary and subject to revision.

Rev 06 June 7, 1999

Table of Contents

1. Overview

*e*Prox™ is HID Embedded Proximity Technology. It permits the integration of a proximity card reader into alarm system keypads in a very cost effective, space efficient way.



By building eProx™ into an alarm system keypad, manufacturers can provide arming, disarming and other functions as a result of a HID proximity card or key-fob being presented at the keypad. This can replace or supplement arming and disarming by PIN, resulting in a significant reduction in false alarms.

eProx™ can substantially reduce false alarms as well as making the alarm system accessible to users who might otherwise have been excluded through age (young and old) or disability. It will be especially useful for infrequent users of systems, such as maids and visitors. In commercial environments it will facilitate the administration of users of the alarm system and enhance security.

eProx™ can be configured to count presentations of the card, permitting additional functionality (configured on the OEM side of the process). This can be used for such functions as distinguishing Arm Away from Arm Home or permitting activation of keypad macros.

eProx™ is available in three versions, each customizable to suit the licensee's particular situation.

- eProx Y consists of the hardware and software specifications that will permit a licensee to fully integrate an eProx card reader in an alarm system keypad - using the keypad's own processor and memory. Licensees will be able to achieve a very low incremental cost, involving only a few hardware components and the antenna.
- eProx D consists of a daughter board and antenna supplied by HID.
- eProx A consists of an HID supplied ASIC for incorporation on to the licensees circuit board and an antenna, supplied by HID or made to HID specification.

A variety of cards are available for eProx systems

eProx™ can be designed to work with all HID compatible cards and tokens, or can be restricted to particular card encoding. Licensees will thus be able to maintain control over the sale of cards and tokens that function on the system should they choose.

ProxKey fobs are ideal for eProx systems.

Note: The eProx™ technology licensee permits the incorporation of eProx into alarm system keypads only. Licensees interested in other applications should discuss these with HID. HID is interested in any application of the technology that is consistent with its strategic plan and does not conflict with its obligations to its licensees and business partners. The eProx™ Technology Licensing Program is not intended to permit licensees to manufacture stand-alone proximity readers.

2. Passive Radio Frequency Identification Device Overview

Radio Frequency Identification Devices (herein known as 'RFID') use radio frequency to identify, locate and track people, assets, and animals. Passive RFID systems are composed of three components, an interrogator (herein known as 'Reader'), a passive tag (herein known as 'Tag'), and a host processor. The Tag is composed of an antenna coil and a silicon chip that includes basic modulation circuitry and non-volatile memory. The Tag is energized by a time-varying electromagnetic radio frequency wave (herein known as 'RF') that is transmitted by the Reader. This RF signal is called a carrier signal. When the RF field passes through an antenna coil, there is an AC voltage generated across the coil. This voltage is rectified to supply power to the tag. The information stored in the tag is transmitted back to the reader. This is often called backscattering. By detecting the backscattering signal, the information stored in the tag can be fully identified.

2.1 Reader

Usually a microcontroller based unit with a wound output coil, peak detector hardware, comparators, and firmware designed to transmit energy to a tag and read information back from it by detecting the backscatter modulation.

2.2 Tag

An RFID device incorporating a silicon memory chip, a wound or printed coil, and a tuning capacitor.

2.3 Carrier

A RF sine wave generated by the Reader to transmit energy to the Tag and retrieve data from the Tag. HID uses 125KHz; higher frequencies are used for RFID tagging but the communication methods are somewhat different. 125KHz utilize transformer-type electromagnetic coupling.

2.3.1 Modulation

Periodic fluctuations in the amplitude of the carrier, used to transmit data back from the Tag to the Reader. Systems incorporating passive RFID Tags operate in ways that may seem unusual to anyone who already understands RF or microwave systems. There is only one transmitter, the Tag is not a transmitter or transponder in the purest definition of the term, yet bi-directional communication is taking place. The RF field generated by a Reader (the energy transmitter) has three purposes:

- **Induce enough power into the Tag coil to energize the Tag.** Passive Tags have no battery or other power source; they must derive all power for operation from the Reader field. 125KHz Tag designs must operate over a vast dynamic range of carrier input, from the very near field to the maximum read distance.
- **Provide a synchronized clock source to the Tag.** HID Tags divide the carrier frequency down to generate an on-board clock for state machines, counters, etc., and to derive the data transmission bit rate for data returned to the Reader.
- **Act as a carrier for return data from the Tag.** Backscatter modulation requires the Reader to peak-detect the Tag's modulation of the Reader's own carrier.

2.3.2 System Handshake

Typical handshake of a Tag and Reader is as follows:

- The HID Reader continuously generates a RF carrier sine wave (when the field is turned on), watching always for modulation to occur. Detected modulation of the field would indicate the presence of a Tag.

- A Tag enters the RF field generated by the Reader. Once the Tag has received sufficient energy to operate correctly, it divides down the carrier and begins clocking its data to an output transistor, which is normally connected across the coil inputs.
- The Tag's output transistor shunts the coil, sequentially corresponding to the data which is being clocked out of the memory array.
- Shunting the coil causes a momentary fluctuation (dampening) of the carrier wave, which is seen as a slight change in amplitude of the carrier.
- The Reader peak-detects the amplitude-modulated data and processes the resulting bit-stream according to the encoding and data modulation methods used.

2.3.3 Backscatter Modulation

This terminology refers to the communication method used by a passive RFID Tag to send data back to the Reader. By repeatedly shunting the Tag coil through a transistor, the Tag can cause slight fluctuations in the Reader's RF carrier amplitude. The RF link behaves essentially as a transformer; as the secondary winding (Tag coil) is shunted momentarily, the primary winding (Reader coil) experiences a momentary voltage drop. The Reader must peak-detect this data at about 60dB down (about 100mV riding on a 100V sine wave). This amplitude modulation loading of the Reader's transmitted field provides a communication path back to the Reader. The data bits can then be encoded or further modulated in a number of ways.

2.3.4 Data Encoding

Data encoding refers to processing or altering the data bit-stream in between the time it is retrieved from the RFID chip's data array and its transmission back to the Reader. The various encoding algorithms affect error recovery, cost of implementation, bandwidth, synchronization capability, and other aspects of the system design.

- **Differential Biphase**

  The bit-stream being clocked out of the data array is modified so that a transition always occurs on every clock edge, and 1's and 0's are distinguished by the transitions within the middle of the clock period. This method is used to embed clocking information to help synchronize the Reader to the bit-stream; and because it always has a transition at a clock edge, it inherently provides some error correction capability. Any clock edge that does not contain a transition in the data stream is in error and can be used to reconstruct the data.

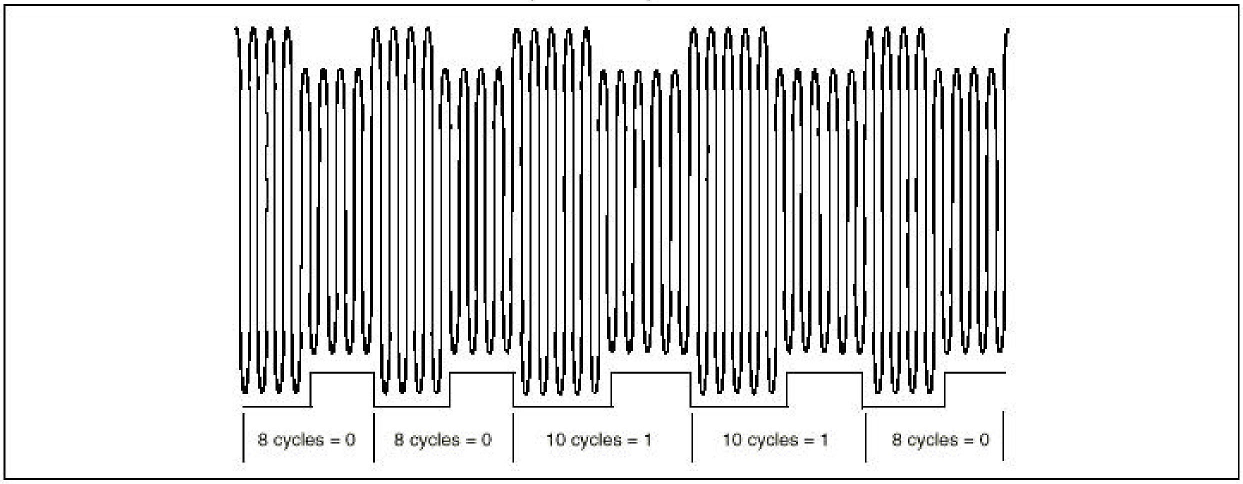
2.3.5 Data Modulation

Although all the data is transferred to the host by amplitude-modulating the carrier (backscatter modulation), the actual modulation of 1's and 0's is accomplished with an additional modulation method.

- **FSK (Frequency Shift Keying)**

  This form of modulation uses two different frequencies for data transfer; the most common FSK mode is Fc/8/10. In other words, a '0' is transmitted as an amplitude-modulated clock cycle with period corresponding to the carrier frequency divided by 8, and a '1' is transmitted as an amplitude-modulated clock cycle period corresponding to the carrier frequency divided by 10. The amplitude modulation of the carrier thus switches from Fc/8 to Fc/10 corresponding to 0's and 1's in the bit-stream, and the reader has only to count cycles between the peak-detected clock edges to decode the data. FSK allows for a simple reader design, provides very strong noise immunity, but suffers from a lower data rate than some other forms of data modulation.

- **Figure for FSK Modulation**



2.4 Antenna Design Notes

Obtaining good read-range performance from an antenna requires careful attention to several factors. All of these factors are affected by the near proximity of electrically conducting (metal) surfaces, which are almost always present in electronic systems. The most important characteristics of the antenna are:

- **Antenna Circuit Inductance**

  The antenna must be reasonably close to an inductance, which will resonate at 125KHz with the series 2400ρF capacitor that is present on the Proximity Reader circuit. The effective inductance seen by the circuit is changed by the proximity of the antenna to metal. For some antenna geometry's, this effect can easily be as much as 30% of total inductance.

- **Eddy Current Field Cancellation**

  When the antenna is located near metal, the effective range of the antenna may be reduced even after the inductance of the antenna is adjusted to obtain resonance at 125KHz. The range reduction is due to the eddy currents present in the metal, which create a magnetic field opposing the antenna magnetic field. The effect is more pronounced when the majority of the field produced by the antenna is perpendicular to the nearby metal surface. This means a large reduction of range for loop antenna placed parallel to a metal surface, as much of the magnetic field is canceled by the field produced by eddy currents in the metal.

- **Antenna Circuit Q**

  The resistive load caused by eddy currents in the housing may reduce the Q of the antenna circuit. Ferrous materials are much worse in this respect than non-ferrous materials. Antenna Q is also dependent on antenna wire diameter, and whether or not a ferrite core is used in the antenna construction. A high Q antenna circuit is desirable.

- **Antenna-Transponder Mutual Inductance**

  The antenna must be a size, which will have good mutual inductive coupling to HID transponders. If the antenna is too large, it will not couple well to keyfobs. If it is too small, it will not couple well to ProxCard II transponders.

2.4.1 Example Antenna Configuration and Design Hints

This example utilizes a simple freestanding, air-core coil and orients the antenna so that most of the magnetic field produced runs perpendicular to the metal surface. This type antenna costs very little, and can be fit into low profile housing. However, it is the type most adversely affected by the metal surface parallel to the antenna. The minimum recommended distance from coil to metal surface is 0.40 inches for a 1 inch (inside diameter) coil.

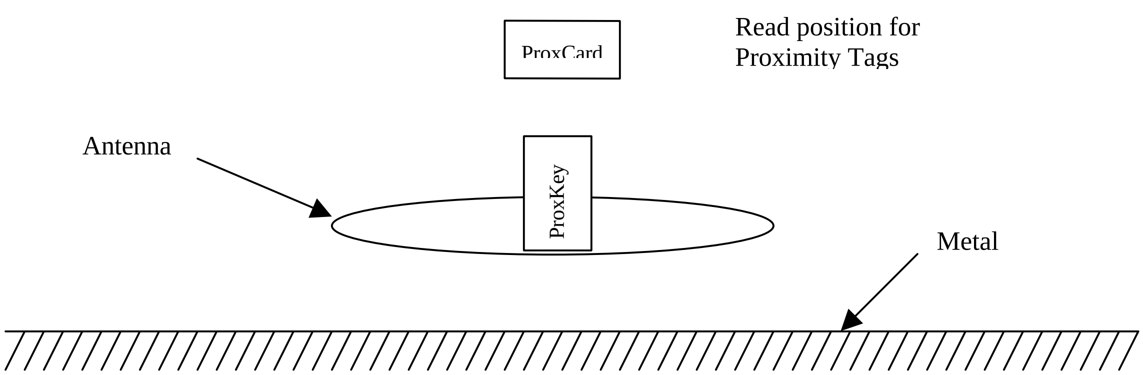
- **Material**
  - Magnet wire, 32 AWG approximately 140 turns x 5"/turn

- **Antenna Dimensions**

  For a circular antenna winding, a coil of approximately 1.2" diameter gives acceptable results. Larger antenna will give more distance with ProxCard, but may not couple well to ProxKey (fobs). Smaller antenna diameters may not couple strongly enough to the ProxCard II antenna.

- **Antenna Winding**

  Adjust total number of turns to 675µH @ 125KHz while antenna is in proximity to the Reader housing, in the same relative position and orientation, which will apply in the production unit. Note that the housing material composition, thickness, and shape will all affect the measured inductance to some extent. To obtain best results, perform the experiments carefully to faithfully reproduce the desired product environment.



3. eProx™ Y - Specifications for full Integration

The precise cost, part count, code space and memory requirements for eProx Y will depend on the particular implementation and the target processor. The following information is therefore given only as an indication to permit preliminary evaluation.

eProx Y provides all the technical information that the alarm system manufacturer requires to fully integrating HID proximity card reading into the keypad.

For software, HID provides the software code specification and flow chart. Sample code is available in Microchip assembly language.

For hardware, HID provides the specification of the exciter drive and receiver circuitry along with the parts list, schematic and theory of operations of its own implementation. A total of four I/O lines are required on the microprocessor.

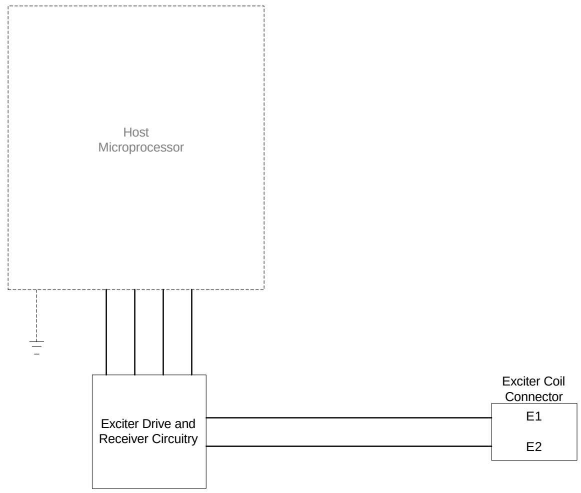
3.1 eProx Y Hardware

3.1.1 Description

Assuming the keypad microprocessor has the necessary capabilities and that a suitably regulated +5VDC power supply is available, the following hardware will be required.

- EEPROM device, maybe internal to the microcontroller
- Exciter drive and Receiver circuitry
- Connector for the exciter coil

3.1.2 Functional Block Diagram



Four connections are required to I/O lines on the microprocessor.

HID

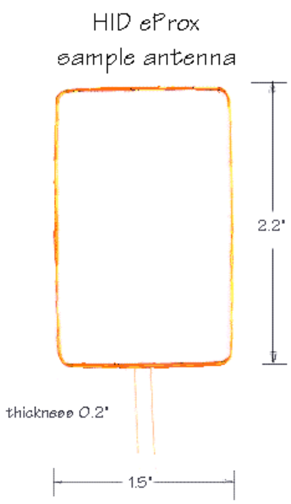
4. eProx ä D - Proximity technology on a daughter board

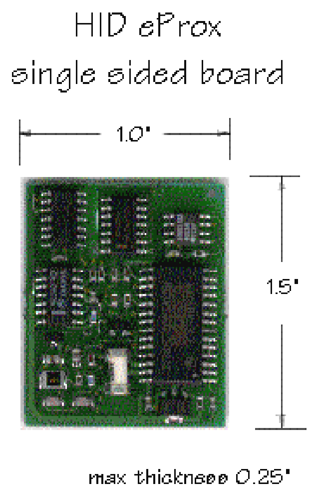
4.1 General technical information

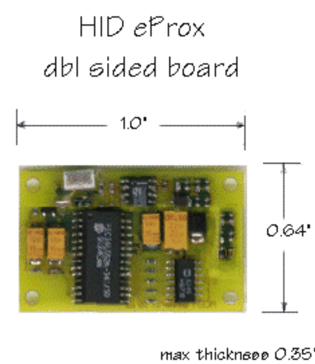
eProx D provides all the componentry required for proximity card reading on a compact daughter board. This is designed to be mounted on the key-pad circuit board, or to the key-pad enclosure, and connected to the key-pad circuit board by a data connection, which in its simplest form can be a single wire plus ground connection.

The daughter board is available as a single or double-sided SMT board as shown below. Typical dimensions are shown in the diagrams, though it should be noted that smaller configurations could be achieved if required.

The antenna is mounted in a suitable location inside the enclosure and is normally soldered permanently to the daughter board. To suit particular physical arrangements it can be connected by a suitable plug (minimum pin spacing will be specified).







The antenna is available in multiple configurations to suit different applications. Typical is a wire wound, rectangular configuration as shown. It is also possible to use a wire wound ferrite rod, which is more compact, but brings more orientation challenges.

4.1.1 eProx Reader Functionality

The eProx unit has the ability to read HID formatted cards. Optionally, it may read only cards from a particular series.

eProx unit output will use a protocol adapted to the alarm system. It is envisaged that the eProx unit may either pass data to the keypad processor on a local data bus or directly to the control unit processor on the alarm system keypad data bus. The choice here will largely be dictated by the alarm system architecture, available inputs on keypad processors and other system level considerations.

4.1.2 Operating Cycle

The ***e***Prox unit will operate in two states:

- Idle/Ready
- Busy/Transmitting

The Idle/Ready State is the normal state waiting for a tag to be presented. When a tag is presented and is being read, the eProx unit is in the Busy/Transmitting State.

4.1.3 Accuracy

The unit will not have more than 1 miss-reads per 10 million.

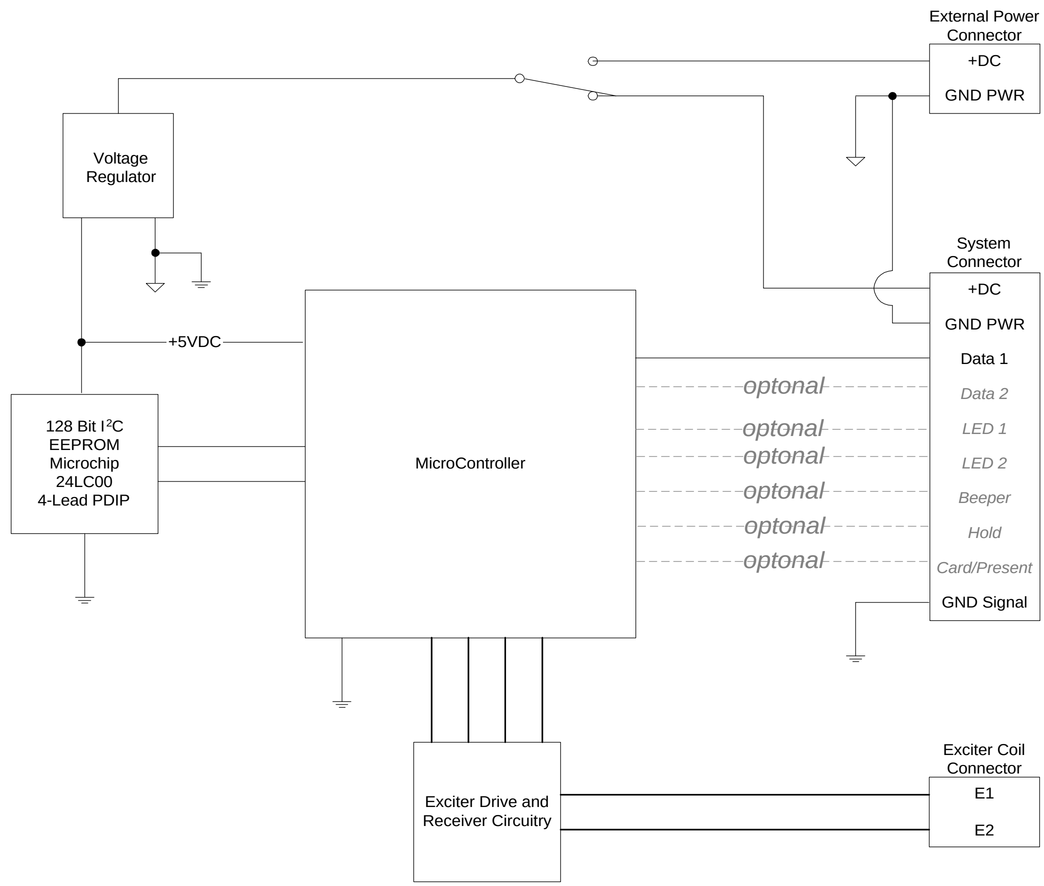
4.2 eProx D Hardware Section

4.2.1 Description

The daughter board hardware will consist of the following components:

- Microcontroller
- Voltage regulator for the board power control, +5VDC (optional)
- EEPROM device, maybe internal to the microcontroller (optional)
- External power connection w/switching or jumper capability
- Connector for the data interface and (optional) indicating devices
- Exciter drive and Receiver circuitry
- Connector for the exciter coil (optional)

4.2.2 Functional Block Diagram



4.3 eProx D Physical Characteristics

4.3.1 Circuit Board

The circuit board is a single or double-sided SMT board. It is available conformal coated if required. It can also be supplied fully potted.

4.3.2 Dimensions

Typical dimensions of the daughter board are 1.5" x 1.0" x 0.25" (single sided) and 1.0" x 0.64" x 0.35" (double sided). Smaller configurations and polygonal configurations can be provided.

4.3.3 Mounting

The board must be securely mounted to the host circuit board or to the keypad enclosure. 4 mounting holes are provided. One mounting hole can provide a ground connection. Alternative configurations can be designed if required.

4.3.4 Connectors

The following connectors are typically used:

Power	Flying leads
Ground	Flying lead or by metal mounting screw
Data	Suitable plug
Antenna	soldered or suitable plug

4.3.5 Antenna

The typical eProx D antenna is a copper wound coil, 2.2" x 1.5" x 0.2", with two flying leads for connection to the circuit board. The antenna may be configured in many different ways to suit particular physical arrangements inside the keypad. An alternative ferrite rod antenna can be used.

4.3.6 Product Shipping Preparation

eProx D units can be packaged individually or in quantity in a shipping box or in a suitable tray for the customer's manufacturing process. Antennae and circuit boards can be packaged separately to facilitate incorporation at different points of the manufacturing process.

4.4 eProx D Reader Specifications

4.4.1 Environmental Characteristics

Operating Temperature Range	-30^{o}C to 65^{o}C (-22^{o}F to 150^{o}F)
Storage Temperature Range	-40^{o}C to 85^{o}C (-40^{o}F to 185^{o}F)
Operating Humidity Range	5% to 95% non-condensing
Operating Vibration Limit	0.04g^2/Hz 20-2000Hz
Operating Shock Limit	30g, 11mSec, Half Sine

4.4.2 Power Requirements (Linear Supply Recommended)

Power Supply	Linear
Operating Voltage Range	5.0VDC -14.0VDC
Absolute Maximum (DC+ non-operating)	16.0VDC
Peak Current 12V (maximum)	60mA
Average Current 12V (maximum)	35mA

4.4.3 Operating Parameters

Excitation Frequency	125KHz
Read and Report Speed (Clock and Data)	506mSec
Card read discrimination interval	180mSec

4.4.4 Read Distance

ProxCard III Proximity Card	2.0”
DuoProx II Proximity Card	1.5”
ProxKey III Proximity Key Fob	0.75”
IsoProx II Proximity Card	1.5”

4.5 eProx Data Output

eProx D can be provided with a variety of outputs to suit the target application. The output can be provided as serial data in a number of protocols or as parallel data. The following list is not exhaustive and HID is ready to work with customers to define custom protocols that optimize the data communication for the target processor. This can include truncation of the card ID, translation of the card ID to user number and counting the number of presentations. With all of the following data communications options, the eProx D can also optionally drive 2 LED's to indicate valid read and invalid read, as well as a beeper. There is also a line that may be used as an interrupt if the data is to be read by the microprocessor without a UART and a line that can be used to signal "ready to receive".

- One wire plus ground asynchronous serial data in a custom format.
- Asynchronous data in RS232 format at 9600 baud on two wires (data plus ground).
- Asynchronous data in RS232 format at 9600 baud on two wires (send, receive and ground).
- I^2C format asynchronous serial data
- Parallel data on eight wires plus ground plus controls.

In its basic configuration, the eProx D generates and sends its own clock pulses for formats that require a separate clock signal. However, the clock can also be supplied by the host processor in response to a signal on the interrupt line and the eProx D will use the supplied clock and treat its presence as a "Ready to Receive" signal.

Designers may wish to consider using an additional one, two (or more) wires on their key-pad connection to the control unit, and transmitting data directly from the eProx D to the control, rather than the key-pad processor, using I^2C or EIA RS232 or RS422. Note that this involves additional protection against static discharge and RFI.

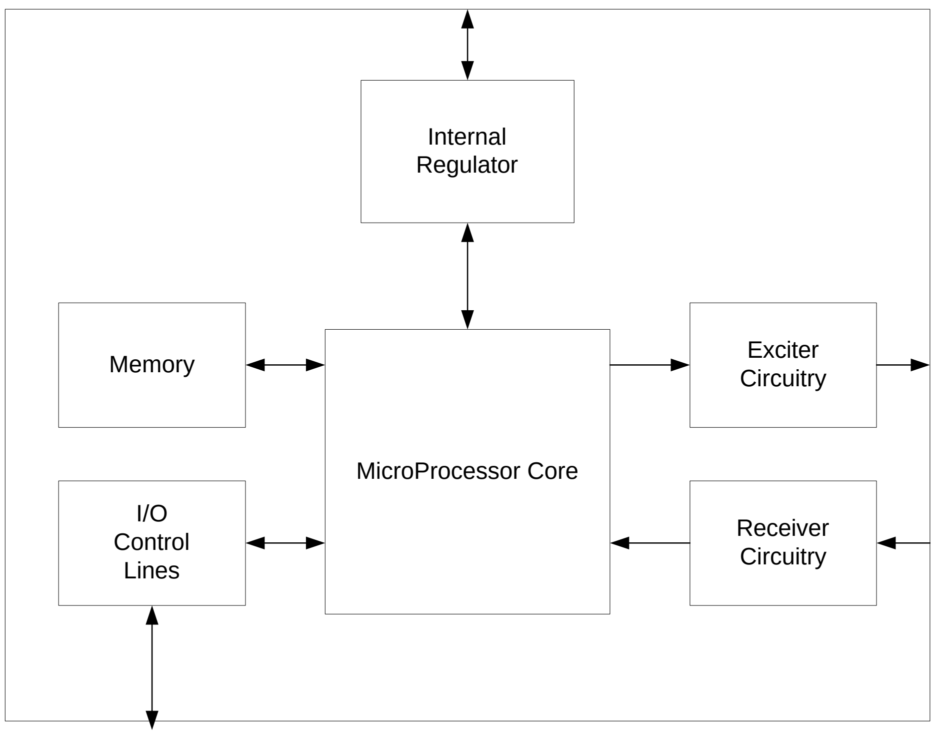
5. eProx™ A - Proximity Technology ASIC

The eProx ASIC puts the functionality into a single integrated circuit. This part of the overall eProx project is still in the specification stage and the final implementation of the ASIC can include specific functionality required by licensees. HID intends to use the input gathered during the implementation of eProx D to finalize the specification of eProx A.

Conceptually, all that has been written above for eProx D can apply to eProx A, including the option of the reader communicating directly with the control processor rather than the key-pad processor. HID is keen to explore with alarm system designers the various options here in order to optimize the product for licensee's planned alarm system architectures.

5.1 eProx ASIC Conceptual Specification

5.1.1 Block Diagram



6. HID Reader Theory and Helpful Specifications

6.1 RFID Card and Card Reader Description

The HID tag reader (herein know as ‘Reader’) generates a 125KHz AC magnetic field with the use of a magnet wire coil (referred to as an “exciter”), located inside the reader enclosure. The coil is connected to a 125KHz square wave signal source through a capacitor. The coil/capacitor combination is designed to form a series resonant network that is tuned to 125KHz. The resonance of the coil/capacitor network turns the square wave drive signal into a sine wave AC current. The magnetic AC field produced by the current in the coil propagates into space and is intercepted by the remote RFID tag (cards and fobs), several inches from the coil. Another coil inside the RFID tag collects some the intercepted magnetic field and converts it into a 125KHz AC voltage. A capacitor connected in parallel with the coil forms a parallel resonant circuit. The parallel resonant circuit improves the coil’s efficiency in converting the AC magnetic field into a voltage. The voltage across the tag coil is routed to a RFID microchip, also inside the tag. The analog front-end of the RFID chip converts the majority of the energy from the AC voltage produced by the tag coil into a DC voltage that is used to supply power to the microchip. In addition, the 125KHz AC voltage is used by the microchip to form a digital clock signal. The microchip uses the clock signal to produce a serial data message from a non-volatile memory, within the microchip. The digital message signal controls an electronic switch that connects and disconnects a resistor to the tag coil. The resistor forms a load across the RFID tag coil and causes the AC voltage across the coil to be reduced. As the digital message emerges from the RFID microchip and switches the resistor across the coil, the AC voltage waveform across the RFID coil is amplitude modulated or frequency modulated. When the tag is placed in front of the Reader, some of the modulated signal from the RFID tag is coupled to the Reader’s coil and appears as weak amplitude modulated signal. As the tag is moved closer to the Reader coil, the level of modulation seen across the Reader coil increases. The modulated voltage across the Reader’s coil is processed by the Reader’s electronic circuit and is ultimately shaped into the original digital message that emerged from the tag’s RFID microchip.

The laws of physics dictate that the strength of the magnetic field projected outward from an open coil will decrease according an inverted cube relationship. In other words, if the separation distance between a tag and a Reader is increased by a factor of two, the magnetic field strength will decrease by a factor of eight. To maintain the same signal level at the RFID tag, the current in the Reader’s coil would then have to be increased by a factor of eight. At a given desired tag read distance, the Reader’s excitation magnetic field needs to be strong enough to activate the RFID tag. The Reader’s receiver circuit also has to be sensitive enough to detect the weak signal produced by the distant RFID tag. If the tag’s circuit is being fully activated by the Reader’s magnetic field but the Reader is not able to detect the tag’s return signal, the system is said to be “receive limited”. The tag would then have to be moved closer to the Reader to be read, shortening the read distance. In other cases the Reader’s receiver circuit may be sensitive enough, but the magnetic field produced by the Reader is insufficient to excite the tag. Such systems are said to be “exciter limited”. Again, the tag would have to be moved closer to the Reader to be read. The goal in a properly designed RFID system is to generate a sufficient magnetic field to insure a tag will be excited at a specific distance and to have a receiver circuit sensitive enough to insure the tag’s return signal could be detected at that same distance.

6.2 125KHz Receiver Information

The standard HID FSK RFID card’s modulated signal is composed of two separate frequencies. The digital message causes the modulation frequency to shift between 12.5KHz and 15.625KHz. The rate, at which the data information is sent, is around 1250 BPS. The minimum number of modulation frequency cycles per bit cell is about 7.

The receiver section of an RFID reader must convert sideband FSK frequencies to baseband frequencies for proper processing. Issues pertinent to RFID readers are briefly discussed below.

6.2.1 Baseband Frequencies, FSK Cards

- Two modulation frequencies = 12.5KHz (125KHz ÷ 10) and 15.625KHz (125KHz ÷ 8)
- Difference between two frequencies = 3.125KHz.
- Center between two modulation frequencies = 14KHz.
- Maximum practical baseband band-pass filter Q = 4.5.

6.2.2 Lower Sideband Frequencies, FSK Cards

- The two tag modulation frequencies produce two lower sidebands -- F1 = 109.37KHz / F2 = 112.5KHz
- Difference between two sideband frequencies = 3.15KHz.
- Center frequency between two sidebands = 111KHz
- Maximum practical 111KHz sideband band-pass filter Q = 37

6.2.3 FSK Band-pass Filters

If a band-pass filter were used with a 111KHz-center frequency, its bandwidth should not be less than ±1.5KHz and should have a Q less than 37. A Q greater than 37 would begin to attenuate the two sideband frequencies. However, if non-adjustable and therefore less accurate components were used, a more practical bandwidth would be about ±2.5KHz with a Q of about 20. With a Q of 20, a bandpass filter centered at 111KHz, would attenuate the 125KHz-carrier signal by about 14db (÷5).

6.2.4 125KHz Notch Filters

A high Q 125KHz-notch filter could be used without influencing the sideband frequencies. For example, a 125KHz notch filter with a Q of 40 would provide about 20db (÷10) of 125KHz attenuation without altering the sidebands. If three accurate pre-tuned Q=40 passive 125KHz notch filters were used, before any active stages, a total of 60db (÷1000) of 125KHz attenuation could be achieved. To receive both FSK and AM modulation frequencies, the notch filters should not contain any additional frequency selection networks. Notch filters using toroidal or potcore style inductors have been built with higher Qs and have tested to provide up to 90db of 125KHz filtering action. However, such filters would require manual adjustments. Notch filters designed to be connected directly across the 125KHz exciter coil will need to be able to handle voltages in excess of 250v peak to peak and will need to have high input impedance.

6.2.5 Receiver Techniques

A popular demodulation technique for receivers in RFID readers is referred as envelope detection and consists of diode detector. The main advantage of the diode detector technique is that the frequency of the signals of interest are three octaves removed from the 125KHz carrier frequency instead of only 10% removed, as in the sideband receiver approach (no demodulation is used). The receiver filter circuits would therefore be easier to achieve high gain as well as high selectivity without the use of expensive manually adjustable networks.

The signals observed across the exciter coil, would be a small card return signal that is superimposed on a very strong 125KHz signal. The exciter coil voltage could be in excess of 250v peak to peak. A single diode detector network would be used to convert the upper and lower sideband RFID card signals into baseband signals. The signal following the diode detector would be a 125v DC level containing the very weak 15.65KHz and 12.5KHz baseband signals of interest and some 125KHz carrier signal. The rather high voltages produced by the diode detection technique do need to be considered when selecting the circuit components. Also, any filters following the diode detector will need to have high impedance, to prevent them from loading down the exciter signal. 125KHz carrier on/off duty cycle control schemes, often used to reduce the average 125KHz power levels, also need to be considered. Each time the 125KHz carrier is turned on or off a 125v level shift would occur. The on and off exciter carrier switching produce very strong low frequency components that would need be removed before the signal was fed to any active filters. Fortunately, the duty cycle frequency is often only about 10Hz and could therefore be easily filtered with some passive components

The low pass filters that form the diode detector circuit would need to be designed carefully. To prevent the circuit from loading down the exciter signal across the coil, the filters will need to use high value resistors and low value capacitors. For an exciter coil voltage of about 250v peak to peak, the filter resistors should be greater than 400KΩ. A three pole passive low pass filter with a knee set at 16KHz and a slope of 18db per octave would attenuate the 125KHz-carrier signal by a factor of 54db (÷500). With such a filter, the rectified 125KHz 125v peak to peak signal would be reduced to about 0.25v peak to peak. However, if the RFID card signal were only about 100µV peak to peak, the ratio between the signals of interest and the 125KHz noise would still be about 68db. Additional filter and gain stages would be needed to produce useful RFID card signal amplitude. If we assume a minimum processed signal level of 50mV peak to peak, then an overall circuit gain of 500 would be needed. If the front-end low pass filter was connected to three bandpass filter stages with a Q of 4 and an overall gain of 1000 the 100µV RFID card signal would be increased to 0.1 volts and the 125KHz carrier would be reduced to nearly nothing.

6.2.6 Exciter Coil Considerations

The card reader exciter coil is designed to generate an AC magnetic field. The magnet field is proportional to the AC current in the coil and the number of turns in the coil. To achieve a practical efficiency, a capacitor is placed between the coil and the coil drive signal. The coil and capacitor form a series resonant circuit. At the resonant frequency, the impedance of the coil/capacitor network drops to a low level. Only the AC resistance of the network limits the coil current. The dominant source of the AC resistance is the DC resistance of the magnet wire used in the coil. The coil wire needs to be selected carefully to achieve the desired coil inductance and the desired coil resistance. If the coil resistance is too low the drive current will be higher than desired. The coil diameter and wire gauge also need to be picked, based on the space available inside the card reader.

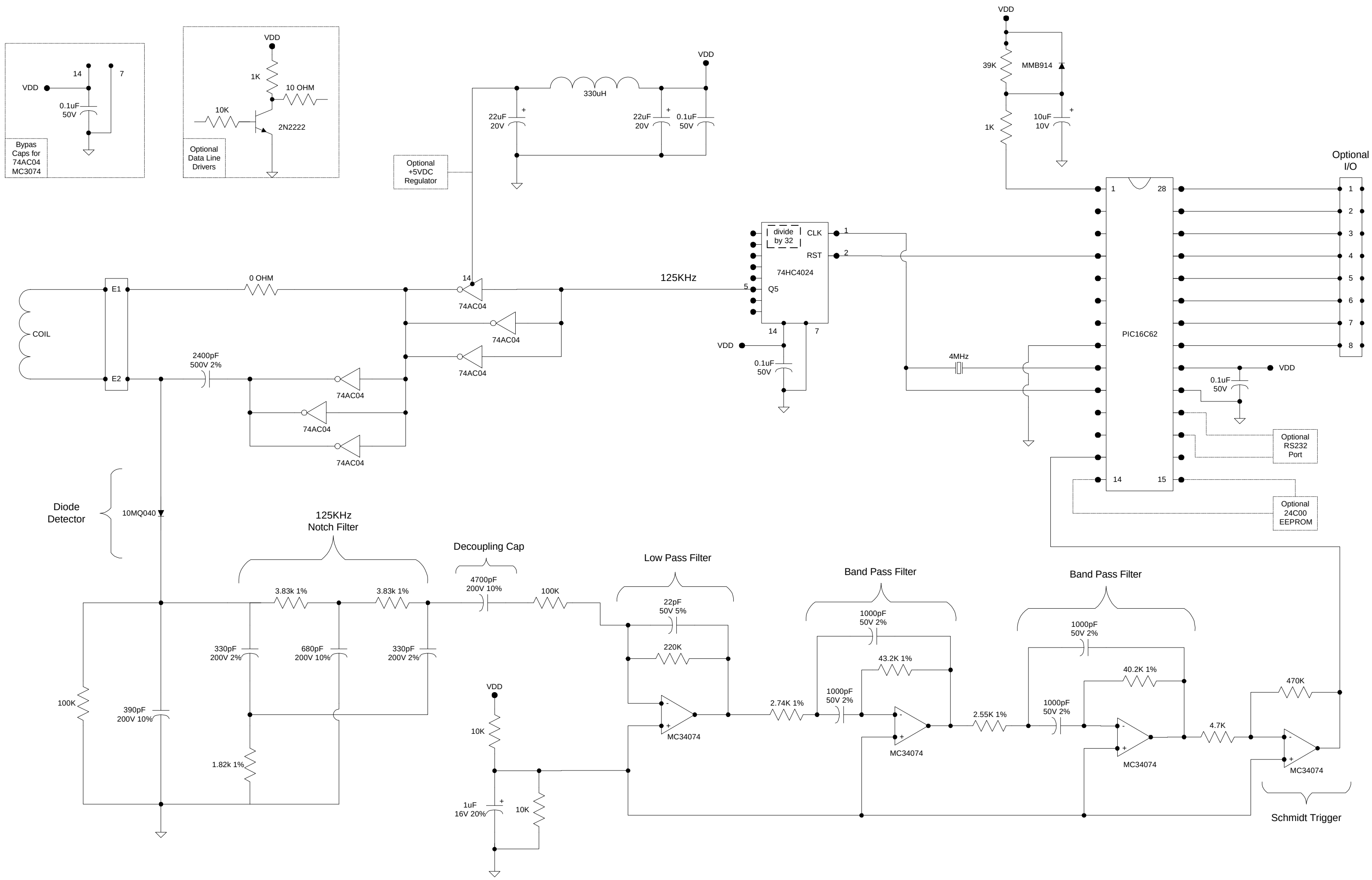
To collect as much tag return signal as possible, the exciter coil should have as many turns as possible. However, to maximize the magnetic field produced, the coil may need to have fewer turns. A compromise must therefore be reached between these two exciter coil needs. For a given peak to peak drive signal applied to the exciter coil, there is large family of possible coil/capacitor combinations that will resonate at 125KHz. But, the goal is to have a coil with the highest practical Q and the right resistance to limit the drive current. By carefully selecting the magnet wire size such a goal can be achieved.

The coil's Q is determined by the ratio of its inductive reactance and its series AC resistance. A high Q will insure that frequencies higher than 125KHz will be attenuated. However, if the exciter coil is to also serve as a receiver for the tag's return signal, the maximum practical Q should be less than about 40. Qs higher than 40 will cause the upper and lower sidebands of the tag signal to be attenuated. These two needs conflict with each other.

If a design calls for a series AC resistance of 15Ω, the inductive reactance of the coil will therefore need to be less than 600Ω to insure the Q is less than 40. A coil with an inductance of 750µH has about the right reactance.

If the coil network is to be driven from lower peak to peak voltages, the coil design will need to change to maintain the same magnetic field. In other words, the amp-turns will need to be the same if the unit is to excite the distant tag properly. The easiest way to achieve the voltage transformation is with the use of a transformer. The lower peak to peak voltage would be connected to the primary of the transformer while the exciter coil would be connected to the transformer secondary.

7. Reader Schematic



7.1 Parts list and target costing

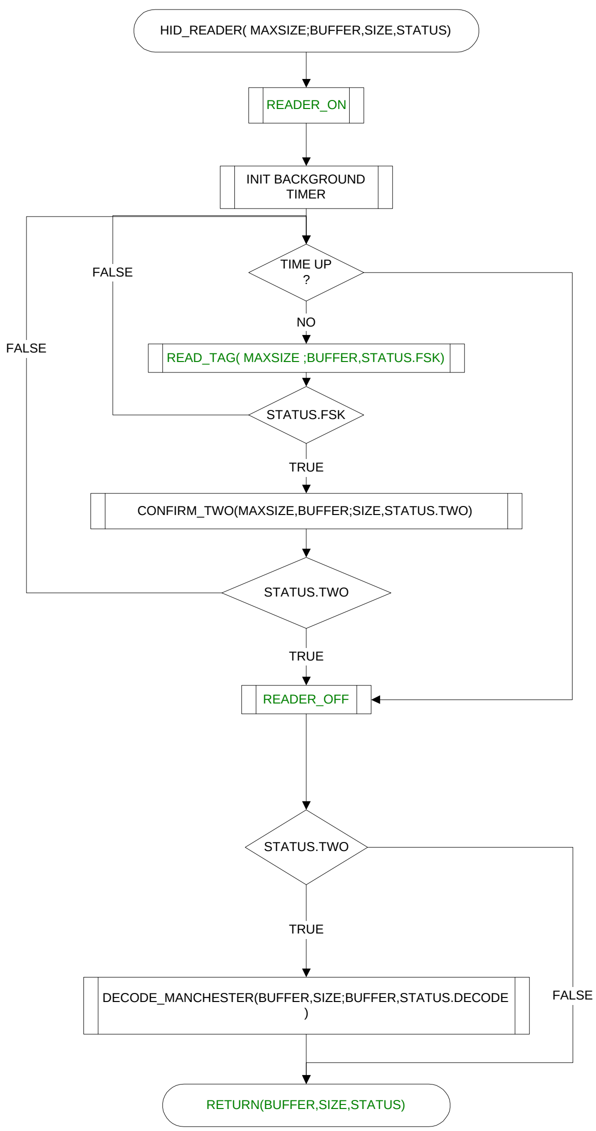
HID Part #	Vendor Number	Description	QTY	Each
298-5104-11	Murata GRM39-25U104M050BD	CAP Z5U, .1UF 50V 20% 0603	5	0.03
296-0220-00	Murata GRM39-X7R220J050BD	CAP X7R, 22PF 50V 5% 0805	1	0.04
290-0005-03	AVX TAJD105M016R	CAP TANT 1UF 16V 20% 7343	1	0.23
290-0005-08	AVX TAJD226M020R	CAP TANT 22UF 20V 20% 7343	2	0.23
290-0005-12	AVX TAJD106M010R	CAP TANT 10UF 10V 20% 7343	1	0.23
21-0005-03	AVX 18127A242GAT	CAP COG 2400PF 500V 2% 1812	1	0.80
297-1102-14	Murata GRM40-COG102G050BD	CAP COG 1000PF 50V 2% 0805	4	0.12
296-0331-30	Murata GRM40-X7R331K200BD	CAP X7R 330PF 200V 10% 0805	2	0.09
296-0391-30	Murata GRM40-X7R391K200BD	CAP X7R 390PF 200V 10% 0805	1	0.09
296-0472-30	Murata GRM40-X7R472K200BD	CAP X7R 4700PF 200V 10% 0805	1	0.09
296-0681-30	Murata GRM40-X7R681K200BD	CAP X7R 680PF 200V 10% 0805	1	0.09
23-0008-01	IR 10MQ040	DIO SCHOT 40V 10MQ040 DO-214AC	1	0.25
164-0003-01	Motorola MMBD914LT1	DIO SW 70V MMBD914LT1 TO-236AX	1	0.05
160-0002-01	Motorola MMBT2222AL	XSTR NPN 40V MMBT2222AL TO-236AX	3	0.06
308-0103-50	Dale CRCWD603103JT	RES FILM 10K .063W 5% 0603	2(3)	0.01
308-0472-50	Dale CRCW0603472JT	RES FILM 4.7K .063W 5% 0603	1	0.01
308-0104-20	Dale CRCW1210104JT	RES FILM 100K .250W 5% 1210	2	0.01
308-0102-50	Dale CRCW0603102JT	RES FILM 1K .063W 5% 0603	1(3)	0.01
308-0100-50	Dale CRCW0603100JT	RES FILM 10 .063W 5% 0603	3	0.01
308-4322-51	Dale CRCW06034322FT	RES FILM 43.2K .063W 1% 0603	1	0.02
308-1821-51	Dale CRCW06031821FT	RES FILM 1.82K .063W 1% 0603	1	0.02
308-3831-51	Dale CRCW06033831FT	RES FILM 3.83K .063W 1% 0603	2	0.02
308-0000-10	Dale CRCW0805000JT	RES FILM 0.0 .100W 5% 0805	1	0.01
308-4022-51	Dale CRCW06034022FT	RES FILM 40.2K .063W 1% 0603	1	0.02
308-0474-50	Dale CRCW0603474JT	RES FILM 470K .063W 5% 0603	1	0.01
308-2741-51	Dale CRCW06032741FT	RES FILM 2.74K .063W 1% 0603	1	0.02
308-2551-51	Dale CRCW06032551FT	RES FILM 2.55K .063W 1% 0603	1	0.02
308-0393-50	Dale CRCW0603393JT	RES FILM 39K .063W 5% 0603	1	0.01
308-0224-50	Dale CRCW0603224JT	RES FILM 220K .063W 5% 0603	1	0.01
101-0003-15	RCD MSI330uHKT	INDUCTOR, SMD, FIXED, 330UH	1	0.20
152-0004-01	Motorola MC74AC04D	IC, INVERTER AC04 MS-012AB	1	0.22
11-0014-01	Motorola MC34074D	IC, OP AMP MC34074, MS-012AB	1	0.40
157-4024-01	Motorola MC74HC4024D	IC, COUNTER-7BIT, HC4024, MS-012AB	1	0.30
11-0009-01	Motorola LM2931CD	IC, VOLT REG-ADJ, LM2931, MS-012AA	1	0.49
5365-360-01	Microchip PIC 16C62	MICROPROCESSOR, PROGRAMMED	1	3.00
14-0013-01	Microchip 24C00-I/OT	IC, MEM-E2PROM, 24C00	1	0.34
24-0010-01	General Semi SMAJ33CA	DIO TVS 33V SMAJ33CA DO-214AC	1	0.16
24-0011-01	General Semi SMAJ33A	DIO TVS 33V SMAJ33A DO-214AC	3	0.18
36-0002-01	Murata CSAC400MGCM0	RESONATOR, CERAMIC, 4.00MHZ THRU	1	0.44
	AMP 640455-2	POLARIZED HEADER, .100", 2-PIN	1	0.26
	AMP 640443-2	RECEPTACLE, .100", 2-PIN	1	0.26
	AMP 770666-1	CST-100 CRIMP CONTACT	2	0.03
	CUSTOM	COIL, ANTENNA	1	1.50

Note: Items in RED are optional components needed for special features if customer requires them.

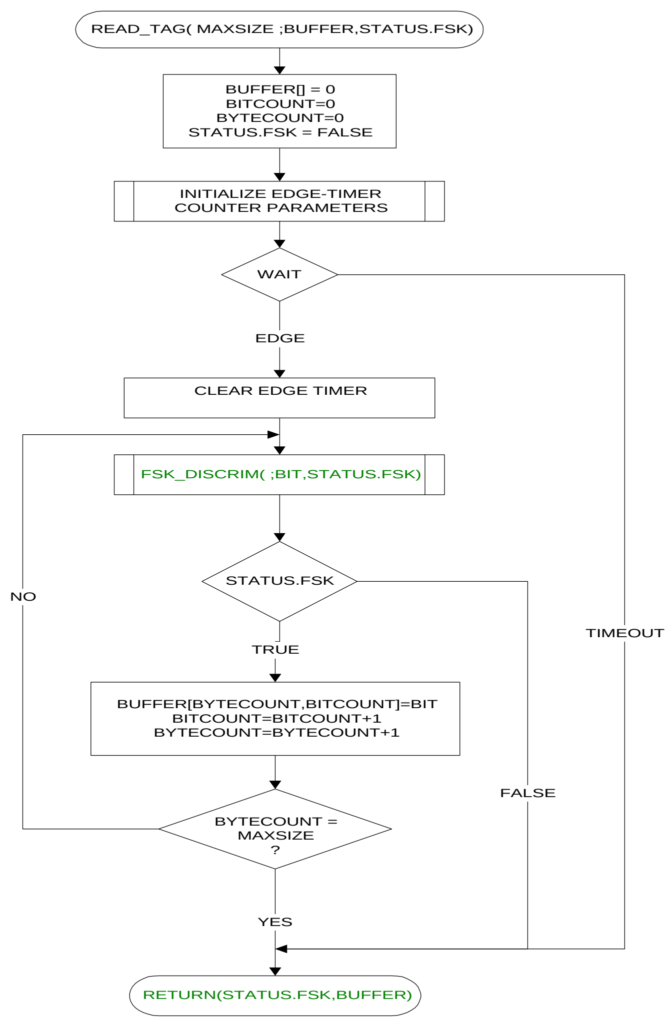
7.2 Software Flowchart

Following is the system flowchart for the card reading functionality. Sample code is available in assembly language and documented later in this specification.

Read Sequence



Read Algorithm



7.2.1 Constraints

The following constraints must be taken into account in assessing whether or not this functionality can be implemented in the host processor.

- Reader presently coded in assembly code for a PIC series of microprocessors running at 4.0MHz and has not yet been tested on other types of microprocessors and frequencies.

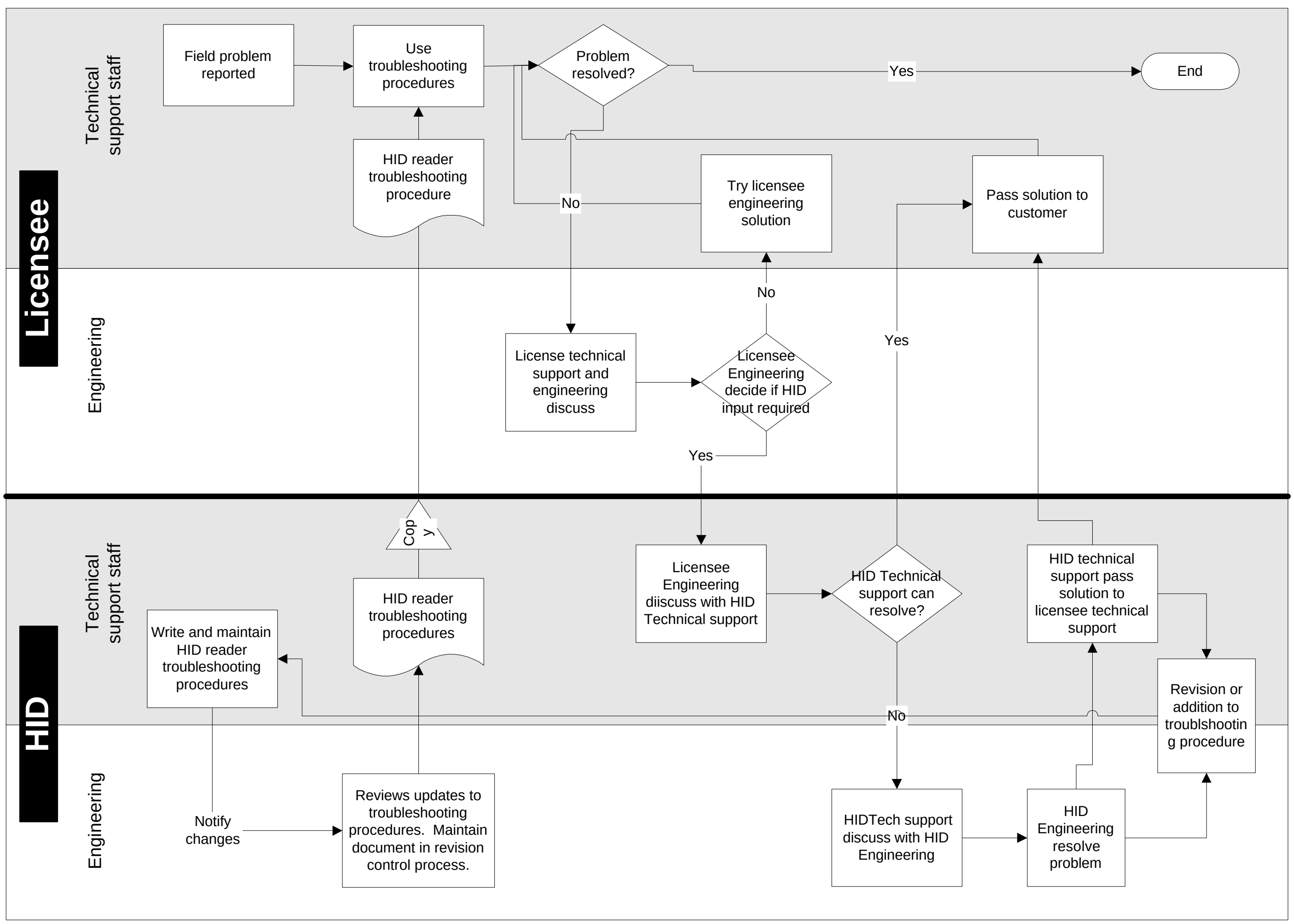
8. Technical Support

During the design stage, a designated HID engineer will be available to the licensee design team to advise on any aspect of the implementation. In addition, HID engineering will review the suitability of the hardware and enclosure physical arrangement for antenna positioning. They will advise on the best antenna location and mounting and carry out tests to establish the nominal read range for each type of HID card.

Upon product release by the licensee, responsibility for technical support of end users and installers will remain with the licensee's technical support staff. A troubleshooting manual will be supplied by HID technical support to provide the methodologies necessary for troubleshooting the eProx units. The licensee technical support staff will have access to HID technical support staff in the event of unresolved problems.

Technical Support Process

For use when field problem is reported to licensee technical support staff



9. Guidelines for Regulatory Approvals

When a modular transmitter or transmitter components are integrated into an electronic system which did not previously contain a transmitter, the system becomes a transmitter. Worldwide, the new system now requires Certification or Type Approval and no longer can be approved under EMC, unintentional radiator, or self-declared test results kept in a file. There is presently no harmonization between countries for 125KHz RFID transmitters; for the most part, each country has its own requirements.

The only country that has a specific standard for electrical safety for RFID Readers as part of an Access Control System is the US. Canada requires a CSA or cUL Mark that is not Access Control System specific. Europe has no specific standard and because HID RF Readers are powered by less than 75VDC, they do not fall under the requirements of the European Low Voltage Directive.

9.1 UNITED STATES

9.1.1 Radio Transmitter FCC Certification

Grant of FCC Certification is a legal requirement to market and sell transmitter products in the US.

Standard: Code of Federal Regulations (CFR) Title 47, Part 15, Subpart C, "Radio Frequency Devices – Intentional Radiators"

The OEM system containing a RFID Reader will require testing at a FCC approved laboratory. Either the lab or the OEM is required to submit an application, test report, technical report, and fees to the FCC. Since the FCC requires this information for an application, the OEM is asked to provide HID with the FCC ID number and application confirmation number; HID will then submit the technical information directly to the FCC laboratory.

The FCC fee for the application is $940 and the confidentiality fee is $135. If a paper submission is made to the FCC, the present time to grant of Certification is 3-4 months; if an electronic submission is made, the time is 2-3 months. FCC approved laboratory test and report fees are additional.

Upon grant of Certification, the OEM may then label his product with FCC ID number and the required FCC one paragraph disclaimer.

9.1.2 Electrical Safety

Electrical safety is the responsibility of the OEM, UL Listing is not a legal requirement in the US.

The 4065/8A ProxPoint OEM Module and the 4035/8C ProxGuts board level Readers are UL Recognized Components.

HID has been securing UL Listing and UL Recognition by submitting products for investigation and testing to UL, Santa Clara.

9.2 CANADA

9.2.1 Radio Transmitter Type Approval

The FCC test report is acceptable and the application will cost $425.

9.3 EUROPE

An EMC Certificate and Type Approval must be granted in one European Country before submissions can be made to subsequent countries.

The OEM will need to author a Declaration of Conformity stating that his product meets the requirements to the European Community Council Directive 89/336/EEC for Access Control Products and cite the standards to which conformity is declared.

9.3.1 CE Mark

Standard: ETS 300 683, "Radio Equipment and Systems (RES); Electro-Magnetic Compatibility (EMC) standard for Short Range Devices (SRD) operating on frequencies between 9KHz and 25GHz"

Testing to ETS 300 683 by an accredited Test House in conjunction with testing to the below Radio Transmitter standards will allow the OEM to apply the CE Mark to his product containing a RFID transmitter after obtaining an EMC Certificate from a European Notified Body.

9.3.2 Germany Radio Transmitter Type Approval

ETS 300 330, "Radio Equipment and Systems (RES); Short Range Devices (SRD) Technical characteristics and test methods for radio equipment in the frequency range 9KHz to 25MHz and inductive loop systems in the frequency range 9KHz to 30MHz"

BAPT 222 ZV 122, "Licensing Regulation for Inductive Transmitting Devices for non-public Radio Control Applications"

An accredited Test House or Notified Body can do testing to the above standards. The application and test report is then submitted to a German Notified Body who issues the Type Approval Certificate. The OEM is then able to apply a product label with the TA number and three-character designation of the Notified Body.

The German Notified Body will invoice the applicant for DM1, 750. Test House fees is additional.

When HID secures Modular Transmitter Type Approval for a board level Reader, that Reader can be integrated into an OEM product and the OEM need only have EMC testing done. The Product Type Approval process described above is unnecessary.

9.3.3 UK Radio Transmitter Type Approval

ETS 300 330, "Radio Equipment and Systems (RES); Short Range Devices (SRD) Technical characteristics and test methods for radio equipment in the frequency range 9KHz to 25MHz and inductive loop systems in the frequency range 9KHz to 30MHz"

MPT 1337, "Performance Specification For low power transmitters and receivers for use in the frequency bands 9KHz to 185KHz and 240KHz to 315KHz allocated for use with low power induction communications and control systems"

An accredited Test House or Notified Body can test to the above standards. The application, EMC Certificate copy, test report, supporting documentation, and application fee of £400 are submitted to the UK Radio-Communications Agency (RA).

After Type Approval is granted, the OEM may then apply the MPT 1337 label to his product.

When HID secures Modular Transmitter Type Approval for a board level Reader, that Reader can be integrated into an OEM product and the OEM need only have EMC testing done. The Product Type Approval process described above is unnecessary.

9.3.4 Test Houses

When the OEM wishes to engage the services of a Test House, it is important that the Test House is accredited in the country of interest and is accredited to test to the above mentioned standards. Most Test Houses have experience with EMC testing for electronic equipment. Fewer have experience with transmitters and fewer still have experience with 125KHz transmitters. Very few are accredited to test 125KHz transmitters to the above standards. Most of these are in Europe and only one we know of is in the US (see below).

HID has been using the services of CKC laboratories as a radio transmitter approval “one stop shop”. CKC is a US lab that is accredited in Germany, the UK, and several other European countries for testing to the above mentioned standards. They can test for the FCC, Canada, Germany, and the UK at the same time for one product. They generate and submit test reports and applications to each country.

CKC Laboratories, Inc.
5473A Clouds Rest
Mariposa, CA 95338
209-966-5420

Appendix I - Parametric Test Procedure

- **Equipment Required**

1) Variable Power Supply, 4.75V and 16V @ 1A DC
2) Oscilloscope, Tektronix 465B or equivalent
3) One 10X Oscilloscope Probe, Tektronix P6119B or equivalent
4) DVM, Fluke 87 or equivalent
5) Standard Wiegand Card
6) Ruler, Inch (Wood or Plastic)

- **Setup**

1) Locate Unit Under Test (UUT) at least four inches away from nearby metallic surfaces such as metal bench tops or metal framework.
2) With the DC power supply adjusted for maximum specified DC voltage and the output disabled, connect the positive lead of the DC power supply in line with the ammeter to VDD. Connect DC power supply ground to GROUND.
3) Set oscilloscope time base for 20µS/division, channel 1 to 0.5 V/division and triggering set to channel 1.

- **Test**

1) On the first power up after manufacturing. Monitor the quiescent current after power up, quiescent current will be less than **30mA** DC.
2) Using oscilloscope channel 1 set to 5 volts/division, measure the coil voltage at E2. All measurements are relative to DC ground. Measure are record the coil voltage and UUT current drain. @4.75V-16V – Voltage will be 0 volts p-p and UUT current drain less than **20mA** DC.
3) Cycle the power off and then back on. This places UUT into standard operation.
4) Adjust oscilloscope channel 1 to 20 volts/division, measure and record the coil voltage on the UUT. @4.75V-16V – Voltage will be greater than 40 volts p-p and UUT current drain less than **30mA** DC.
5) Measure and record the following read distances:

Power Supply Voltage	**Read Distance**	
	Minimum	Nominal
4.75V	2.75”	3.00”
12V	2.75”	3.00”
16V	2.75”	3.00”

PARAMETRIC TEST DATA SHEET

DATE:________________

SERIAL NUMBER	LOW POWER		STANDARD POWER		CHECK READ DISTANCE		CONFIG OPTION NUMBER
	COIL 0 Vp-p (√)	(I) 20 mADC (√)	COIL 40 Vp-p (√)	(I) 30 mADC (√)	NORMAL MODE 3.0 INCH (√)	METAL BACKING 2.0 INCH (√)	

Appendix II - HID PIC Assembly Source Code

```
;*****************************************************************************
; PROJECT     eProx
; ROUTINE     HID_Reader
; FILE        HIDREADX.ASM
; INPUT       maxsize byte: sets maximum size of raw tag data in bytes
;             (4x tag data stream output size)
;
; OUTPUT      buffer array: data collection and manipulation area
;             size byte:  exits with size of decoded data stream in bytes
;             read_stat byte: set of flag bits indicating status of tag read
;
;             Output Format: data is packed so MSB of lowest address byte is first
;                            bit received (Manchester violation is zeroed out)
;
;               buffer+0 buffer+1 buffer+2 buffer+3 buffer+4 buffer+5
;               0000cccc cccxxxxx xxxxxxxx xxxxxxxx xxxxxxxx xxxxxxxx
;
;              c = customer code bit
;              x = data bit
;
; VARIABLES:
; CONSTANTS   See constant.inc for details
; Functions:  controls exciter/receiver internal timers and external hardware
;             calibrates internal RC oscillator to external 500KHz clock reads FSK
;             transponder data auto-detects frame size from 2 to 7 data blocks,
;             detects sync word, decodes Manchester data, and justifies data into
;             buffer area
;
;*****************************************************************************

HID_Reader:
       ;*********************************************************************
       ;**** clear status, turn reader power on, initialize timeout
       ;*********************************************************************
       clrf    read_stat
       call    Reader_On
       call    Init_Timeout
       clrf    cust_code

HID_Loop:
       ;*********************************************************************
       ;**** test for timeout, try to read a tag
       ;*********************************************************************
       btfsc   READER_OVERRUN_STATUS,OVF_FLAG
       goto    HID_Quit
       call    WATCH_DOG
       call    Read_Tag
       btfss   read_stat,FSK
       goto    HID_Loop

       ;*********************************************************************
       ;**** load byte count, confirm two frames
       ;*********************************************************************
       movlw   NOMINAL_BLOCKS*BLOCKSIZ*2
       addlw   BLOCKSIZ
       call    Confirm_Twice
       btfss   STATUS,Z

       ;*********************************************************************
       ;**** mismatch occurred - buffer data not two identical frames
       ;*********************************************************************
       goto    HID_Loop

       ;*********************************************************************
       ;**** found two matching frames - save frame size, turn reader off
```

```
        ;********************************************************************
        movwf   size
        bsf     read_stat,TWICE
        call    Reader_Off

        ;********************************************************************
        ;**** decode the buffer data
        ;********************************************************************
        call    WATCH_DOG
        call    Decode_Manchester

        ;********************************************************************
        ;**** prepare to rotate buffer 4x
        ;********************************************************************
        movlw   0x04
        movwf   temp2

HID_Loop_fixup:
        ;********************************************************************
        ;**** Rotate right to match output
        ;********************************************************************
        movf    size,W
        call    Rotate_Buffer_Right
        decfsz  temp2,F
        goto    HID_Loop_fixup
        call    Init_TMR1
        return

HID_Quit:
        ;********************************************************************
        ;**** Trun reader off and clear reader status
        ;********************************************************************
        call    Reader_Off
        clrf    read_stat
        call    Init_TMR1
        return

;****************************************************************************
;**** End ROUTINE
;****************************************************************************

;****************************************************************************
; PROJECT       eProx
; ROUTINE       Read_Tag
; FILE          HIDREADX.ASM
; FUNCTION
; INPUT
;
; OUTPUT
; VARIABLES:
; CONSTANTS     See constant.inc for details
;
;****************************************************************************
Read_Tag:
        ;********************************************************************
        ;**** Prepare to initialize the tag BUFFER, point to buffer beginning
        ;**** determine buffer area to initialize, initialize the fill data
        ;********************************************************************
        movlw   buffer
        movwf   FSR
        movlw   MAXBLOCKS*BLOCKSIZ*2
        movwf   byte_count
        movlw   FILLDATA

Read_Clear_Loop:
        ;********************************************************************
        ;**** fill buffer area with fill data
        ;********************************************************************
        movwf   INDF
        incf    FSR,F
        decfsz  byte_count,F
```

```
        goto    Read_Clear_Loop

        ;*********************************************************************
        ;**** clear bit count, init byte count, init buffer ptr, clear bit
        ;*********************************************************************
        clrf    bit_count
        movf    maxsize,W
        movwf   byte_count
        movlw   buffer
        movwf   FSR
        bcf     read_stat,FSK

        clrf    prev_time
        clrf    curr_time
        call    INIT_CCP
        bsf     T1CON,TMR1ON

Read_Tag_Loop:
FSK_Discrim:
        ;*********************************************************************
        ;**** Clear both frequency bit counters
        ;*********************************************************************
        clrf    f1_count
        clrf    f0_count

FSK_Discrim_Loop:
FSK_Wait_Low:
FSK_Wait_Edge:
        ;*********************************************************************
        ;**** check for timeout and poll for next positive edge
        ;*********************************************************************
        btfsc   READER_OVERRUN_STATUS,OVF_FLAG
        goto    FSK_Discrim_Abort
        btfss   PIR1,CCP1IF
        goto    FSK_Wait_Edge
        bcf     PIR1,CCP1IF

FSK_Edge:
        ;*********************************************************************
        ;**** capture current time, calculate delta
        ;*********************************************************************
        movf    CCPR1L,W
        movwf   curr_time
        movf    prev_time,W
        subwf   curr_time,W
        movwf   delta_t
        movf    curr_time,W
        movwf   prev_time

        movf    delta_t,W
        ;*********************************************************************
        ;**** subtract hi freq limit from edge timer value and evaluate diff
        ;*********************************************************************
        subwf   t_short,W
        btfsc   STATUS,C

        ;*********************************************************************
        ;**** diff <= hi freq limit - must be hi freq
        ;*********************************************************************
        goto    FSK_Hi_Freq

        ;*********************************************************************
        ;**** diff > hi freq limit - may be lo freq - subtract lo freq limit
        ;**** from timer value and evaluate diff
        ;*********************************************************************
        movf    delta_t,W
        subwf   t_long,W
        btfss   STATUS,C

        ;*********************************************************************
        ;**** diff > lo freq limit - must be low freq
```

```
        ;*******************************************************************
        goto    FSK_Low_Freq

FSK_Transition:
        ;*******************************************************************
        ;**** diff < lo freq limit and diff > hi freq limit - possibly a
        ;**** transition between the two frequencies - count both
        ;*******************************************************************
        ;DB2ON
        incf    f1_count,F
        incf    f0_count,F

        ;*******************************************************************
        ;**** evaluate high freq bit count
        ;*******************************************************************
        movf    f0_count,W
        sublw   0x6
        ;DB2OFF
        btfss   STATUS,Z
        goto    FSK_Trans_1
        goto    FSK_Hi_Done

FSK_Trans_1:
        ;*******************************************************************
        ;**** evaluate low freq bit count
        ;*******************************************************************
        movf    f1_count,W
        sublw   0x5
        btfss   STATUS,Z
        goto    FSK_Discrim_Loop
        goto    FSK_Low_Done

FSK_Low_Freq:
        ;*******************************************************************
        ;**** service lo freq count, clear hi freq count, evaluate lo count
        ;*******************************************************************
        ;DB1ON
        clrf    f0_count
        incf    f1_count,F
        movf    f1_count,W
        sublw   0x5
        ;DB1OFF
        btfss   STATUS,Z
        goto    FSK_Discrim_Loop

FSK_Low_Done:
        ;*******************************************************************
        ;**** store a 1 in buff
        ;*******************************************************************
        ;DB4ON
        bsf     STATUS,C
        rlf     INDF,F
        bsf     read_stat,FSK
        ;DB4OFF
        goto    Read_Tag_A

FSK_Discrim_Abort:
        ;*******************************************************************
        ;**** Timeout while acquiring bits
        ;*******************************************************************
        bcf     read_stat,FSK
        goto    Read_Tag_A

FSK_Hi_Freq:
        ;*******************************************************************
        ;**** service hi freq count, clear lo freq count, evaluate hi count
        ;*******************************************************************
        ;DB0ON
        clrf    f1_count
        incf    f0_count,F
        movf    f0_count,W
```

```
        sublw   0x6
        ;DB0OFF
        btfss   STATUS,Z
        goto    FSK_Discrim_Loop

FSK_Hi_Done:
        ;*********************************************************************
        ;**** store a 0 in buff
        ;*********************************************************************
        ;DB3ON
        bcf     STATUS,C
        rlf     INDF,F
        bsf     read_stat,FSK
        ;DB3OFF

Read_Tag_A:
        ;jump back point for FSK_Discrim block
        ;*********************************************************************
        ;**** evaluate FSK bit in read status
        ;*********************************************************************
        btfss   read_stat,FSK

        ;*********************************************************************
        ;**** FSK bit is clear - abort - must be an error
        ;*********************************************************************
        ; need to evaluate this FSK bit in more detail
        goto    Read_Tag_Abort

        ;*********************************************************************
        ;**** FSK bit is set-service bit count-check for byte full
        ;*********************************************************************
        incf    bit_count,F
        btfss   bit_count,3

        ;*********************************************************************
        ;**** byte not full - go back for more bits
        ;*********************************************************************
        goto    Read_Tag_Loop

        ;*********************************************************************
        ;**** byte is full - move ptr to next byte, clear bit count
        ;*********************************************************************
        incf    FSR,F
        clrf    bit_count

Read_Tag_1:
        ;*********************************************************************
        ;**** service byte count and determine if buffer full or not
        ;*********************************************************************
        decfsz  byte_count,F

        ;*********************************************************************
        ;**** buffer not full - go back for more
        ;*********************************************************************
        goto    Read_Tag_Loop

        ;*********************************************************************
        ;**** buffer is full - ALL done - set FSK bit in read status & return
        ;*********************************************************************
        bsf     read_stat,FSK
        return

Read_Tag_Abort:
        ;*********************************************************************
        ;**** timeout or error - clear FSK bit in read status & return
        ;*********************************************************************
        bcf     read_stat,FSK
        return

;*****************************************************************************
```

```
; PROJECT      eProx
; ROUTINE      Decode_Manchester
; FILE         HIDREADX.ASM
; FUNCTION
; INPUT
;
; OUTPUT
; VARIABLES:
; CONSTANTS    See constant.inc for details
;
;*****************************************************************************
;inputs:       buffer: buffered FSK data (Manchester coded)
;              size:  max size of FSK ProxCard data in bytes
;outputs:      buffer: buffered data (decoded)
;              size:  size of decoded card message in bytes
;              read_stat,COHERENT set if tag passes sync and manchester tests

Decode_Manchester:
       ;*********************************************************************
       ;****
       ;*********************************************************************
       movlw  8
       movwf  bit_count
Decode_Sync_Loop:                     ;find first sync word
       ;*********************************************************************
       ;****
       ;*********************************************************************
       movf   size,W
       movwf  byte_count
       movlw  buffer
       movwf  FSR
Decode_Sync_1:
       ;*********************************************************************
       ;****
       ;*********************************************************************
       movf   INDF,W                  ;test all bytes for SYNC
       sublw  SYNC_PATTERN
       btfsc  STATUS,Z
       goto   Decode_Justify
       incf   FSR,F
       decfsz byte_count,F
       goto   Decode_Sync_1
       movf   size,W                  ;get size of twice-confirmed data
       call   Rotate_Buffer_Left      ;rotate buffer 1 bit to left
       decfsz bit_count,F
       goto   Decode_Sync_Loop
       goto   Decode_Fail
              ;Sync_pattern is now on a byte boundary
              ;  FSR points to SYNC word
              ;  byte_count is size - (FSR-buffer) = (number of bytes
              ;    to end of first buffered tag) including SYNC

Decode_Justify:

       ;*********************************************************************
       ;****
       ;*********************************************************************
       movlw  buffer          ;Justify data to beginning of (second half of) buffer.
       addwf  size,W          ;
       movwf  temp2           ;destination pointer

       ;*********************************************************************
       ;****
       ;*********************************************************************
       movf   FSR,W
       movwf  temp1           ;source pointer (points to location of word after sync word)
       incf   temp1,F

       ;*********************************************************************
       ;****
       ;*********************************************************************
```

```
        decfsz  byte_count,F    ;by checking left-over byte_count from sync search
        goto    Decode_Just_0
        movlw   buffer
        movwf   temp1

Decode_Just_0:
        ;*******************************************************************
        ;****
        ;*******************************************************************
        movf    size,W
        movwf   temp3           ;initialize counter

Decode_Just_1:                  ;BEGIN LOOP
        ;*******************************************************************
        ;****
        ;*******************************************************************
        movf    temp1,W         ;       get source byte
        movwf   FSR
        movf    INDF,W
        movwf   temp4           ;       save somewhere

        ;*******************************************************************
        ;****
        ;*******************************************************************
        movf    temp2,W
        movwf   FSR
        movf    temp4,W
        movwf   INDF            ;       put in destination

        ;*******************************************************************
        ;****
        ;*******************************************************************
        incf    temp1,F         ;       increment source pointer
        decfsz  byte_count,F    ;
        goto    Decode_Just_2   ;        (wrap pointer around end of tag data)
        movlw   buffer
        movwf   temp1

Decode_Just_2:
        ;*******************************************************************
        ;****
        ;*******************************************************************
        incf    temp2,F         ;       increment destination pointer
        decfsz  temp3,F
        goto    Decode_Just_1   ;END LOOP
                ;Test for 'small tags' which fit multiple copies into 'size'

Decode_Test_Small:
        ;*******************************************************************
        ;****
        ;*******************************************************************
        movf    size,W          ;size of tag data area to seek smaller frames
        movwf   temp6           ;(save offset of beginning of second frame)
        call    Confirm_Twice   ;outputs size of smaller tag if found
        btfss   STATUS,Z
        goto    Decode_Large

Decode_Small:
        ;*******************************************************************
        ;****
        ;*******************************************************************
        movwf   size            ;store new found size of (smaller) tag

Decode_Large:
Decode_Pack:
        ;*******************************************************************
        ;****
        ;*******************************************************************
        movlw   buffer          ;Decode justified Manchester to beginning of buffer
        addwf   temp6,W         ;temp1=source pointer, at beginning of second frame
        movwf   temp1
```

```
        addwf   size,W          ;replace SYNC (at end of second frame) w\ 'manchester zero'
        movwf   FSR
        decf    FSR,F           ;
        movlw   0x55
        movwf   INDF

        ;*********************************************************************
        ;****
        ;*********************************************************************
        movlw   buffer          ;temp2=destination pointer
        movwf   temp2

        ;*********************************************************************
        ;****
        ;*********************************************************************
        rrf     size,W          ;size of decoded data = 1/2 size Manchester encoded data
        andlw   0x7F
        movwf   byte_count
        movwf   size

Decode_Pack_Loop
        ;*********************************************************************
        ;****
        ;*********************************************************************
        movf    temp1,W         ;get data byte from source area
        movwf   FSR
        movf    INDF,W
        movwf   temp3           ;get first data byte
        incf    FSR,F
        movf    INDF,W
        movwf   temp4           ;get second data byte
        incf    FSR,W
        movwf   temp1           ;save pointer
        clrf    temp5
        clrf    temp6
        bcf     STATUS,C

        ;*********************************************************************
        ;****
        ;*********************************************************************
        rlf     temp3,F         ;split odd data byte odd bit-weights even bit-weights
        rlf     temp5,F
        rlf     temp3,F
        rlf     temp6,F

        ;*********************************************************************
        ;****
        ;*********************************************************************
        rlf     temp3,F
        rlf     temp5,F
        rlf     temp3,F
        rlf     temp6,F

        ;*********************************************************************
        ;****
        ;*********************************************************************
        rlf     temp3,F
        rlf     temp5,F
        rlf     temp3,F
        rlf     temp6,F

        ;*********************************************************************
        ;****
        ;*********************************************************************
        rlf     temp3,F
        rlf     temp5,F
        rlf     temp3,F
        rlf     temp6,F

        ;*********************************************************************
        ;****
```

```
	;*********************************************************************
	rlf	temp4,F		;split second data byte odd bit-weights even bit-weights
	rlf	temp5,F
	rlf	temp4,F
	rlf	temp6,F

	;*********************************************************************
	;****
	;*********************************************************************
	rlf	temp4,F
	rlf	temp5,F
	rlf	temp4,F
	rlf	temp6,F

	;*********************************************************************
	;****
	;*********************************************************************
	rlf	temp4,F
	rlf	temp5,F
	rlf	temp4,F
	rlf	temp6,F

	;*********************************************************************
	;****
	;*********************************************************************
	rlf	temp4,F
	rlf	temp5,F
	rlf	temp4,F
	rlf	temp6,F

	;*********************************************************************
	;****
	;*********************************************************************
	comf	temp6,W
	xorwf	temp5,W		; If Manchester OK, temp5 = complement temp6
	btfss	STATUS,Z
	goto	Decode_Fail

	;*********************************************************************
	;****
	;*********************************************************************
	movf	temp2,W		;store data byte
	movwf	FSR
	movf	temp5,W
	movwf	INDF
	incf	temp2,F
	decfsz	byte_count,F
	goto	Decode_Pack_Loop

Decode_Success:
	;*********************************************************************
	;****
	;*********************************************************************
	bsf	read_stat,COHERENT
	return
Decode_Fail:
	;*********************************************************************
	;****
	;*********************************************************************
	bcf	read_stat,COHERENT
	return
;*****************************************************************************
;**** End ROUTINE
;*****************************************************************************
```