CML Microcircuits

COMMUNICATION SEMICONDUCTORS

Application Note
Security Industry Protocols with the CMX865A

1   Introduction

Security alarm panels are used around the world as a cost-effective means of protecting people and property. The Security Industry Association (SIA)[1], a trade organization that represents security market companies, has developed protocols that describe the communications link used by these panels. DTMF tones, pulsed tones, and frequency-shift keying (FSK) are common signaling methods used throughout these SIA protocols.

The CMX865A low-power wireline modem can be easily configured to perform these SIA protocols making it attractive for security panel applications. The purpose of this document is to describe how the CMX865A can be made to perform SIA communication protocols.

This application note is written on the presumption that the reader is already familiar with SIA protocols. The information contained in this document is not meant to replace the SIA standards, but instead, to highlight specific portions of the SIA protocols that are relevant to the CMX865A. For more detailed information on SIA protocols, the reader is encouraged to contact the SIA (contact information provided at the bottom of this page).

The CMX865A datasheet should be consulted while reviewing this application note.

---

[1] Security Industry Association, 635 Slaters Lane, Suite 110, Alexandria VA 22314, telephone 703-683-2075, www.siaonline.org.

2   The CMX865A Telecom Signaling Device

The CMX865A is a low-power modem IC designed for cost-sensitive applications such as security alarm panels. The CMX865A provides the following functions:

- Bell 202 and V.23 1200bps modem
- Bell 103 and V.21 300bps modem
- Standard DTMF encoder
- Low voice-falsing DTMF decoder
- Programmable tone encoder
- Programmable tone decoder

These features provide the CMX865A with the capabilities of serving the most common SIA communications protocols.

This application note is written from the perspective of a CMX865A placed in a "transmitter" unit (e.g. an alarm panel placed at a customer's residence).

3   General Notes

Where applicable, V_{DD}=3V for purposes of calculations in this document.

Interface details, line level specifications etc. are not specified because these are dependent on the application and the intended country of operation. For this information please contact your local PTT or refer to the relevant specifications.

The following syntax is used to describe the CMX865A C-BUS registers:
1.   *General Control* ($E0) - The register name is given in italics followed by the hexadecimal address of the register in brackets.
2.   b2 = 1 - The register bit, bit 2, is set to a 1.
3.   b5..b2 = 1001 - The register bits, bit 5 through bit 2, are set to 1001 respectively.

All figures in this document assume that the CMX865A start-up procedure has been followed:
1.   The CMX865A is powered up and a General Reset command has been issued (C-BUS address $01, no data required)
2.   Write to the *General Control* ($E0) register
              b7 = 1 (reset internal circuitry)
              b8 = 1 (select normal operating mode)
3.   Wait 20ms
4.   Write to the *General Control* ($E0) register
              b8 = 0 (powersave mode selected)
5.   The CMX865A will now be in powersave mode and can only be woken by the host microprocessor.

Status ($E6) IRQ bit, b15, should always be used to qualify IRQs.

Status ($E6) Energy Detect bit, b10, should be used to qualify call progress tones by monitoring its cadence. This same bit should be used to qualify all pattern detects and the validity of demodulated data when Rx Modem Modes are selected.

Timers and error handling are not always explicitly covered. Please consult local regulatory information for further information.

4   CMX865A Programmable Tone Encoder/Decoder

The CMX865A contains a flexible encoder/decoder that can be programmed for a wide range of tones. This feature helps the CMX865A support the SIA formats because it can be tailored for many different handshake tones, data carrier frequencies, and kissoff tones. The CMX865A datasheet provides full details on this versatile function block.

4.1   Programmable Tone Encoder

The programmable tone encoder consists of four transmit tone pairs, TA, TB, TC, and TD. Each of these 'tone pairs' can be configured for two different tones. For example, tone pair TA can be configured to transmit 1400Hz and 2300Hz, if desired, while the three other tone pairs can be configured for different tones. Single tones are programmed by setting both level and frequency values to zero for one of the tones in the pair.

In order to program a tone for transmission, the desired tone frequency and amplitude must be loaded into the CMX865A Programming register ($E8). Seventeen C-BUS transactions are required to fully load eight different tones into the CMX865A's four tone pairs.

A C-BUS "transaction" is made up of the following events:
- CSN (Chip Select) pulled low.
- CMX865A device register address loaded on Command Data line.
- Register contents, in hex, are loaded on Command Data line.
- CSN pulled high.
- Repeat as necessary for each word to be loaded.

More information on the C-BUS interface can be found in the CMX865A datasheet and in the application notes available on the CML Microcircuits website.

A typical operation with the programmable tone encoder would be as follows:
- Load the Programming register ($E8) with values for the desired tones.
- Write to the Tx Mode register ($E1):
  - Enable the programmable tone encoder by setting b15..12 = 0001
  - Select "programmed tone pair" by setting b4 = 0
  - Select the desired tone pair with b3..0

The tone will be transmitted. Now write to the Tx Mode register ($E1):
- b3..0 = 0000 to turn off the transmitted tone(s)
- b3..0 with the desired tone to turn it back on
- To change the transmitted tone pair, write to the Tx Mode ($E1) register with the desired value for b3..0.

4.2 Programmable Tone Decoder

The programmable tone decoder can be configured to detect two different tones. Coefficients are required to configure the decoder for a particular tone. These coefficients can be generated with third-party software tools, but they can also be easily generated with the CML Coefficient Generator Spreadsheet (discussed in the next section).

The programmable tone decode function is enabled by setting Rx Mode ($E2) b15..12 = 0001. In order for the CMX865A to issue an interrupt when a programmed tone is detected, the programmable tone detector IRQs must be unmasked by setting General Control ($E0) register b1..0=11.

The detection bandwidth of the CMX865A programmable tone detector stays effectively constant until the input signal amplitude falls below 50mVrms. Consequently, the design should be crafted so that the input signal amplitude of the CMX865A's programmable tone detector does not fall below 50mVrms.

5   CML Coefficient Generator Spreadsheet

Many SIA protocols require the detection of special tones. The CMX865A contains a programmable tone detector that can easily detect these tones, but the detector must first be properly configured.

The process of configuring this detector involves determining coefficients for the infinite impulse response (IIR) filters contained within the detector. CML has developed a Microsoft Excel® spreadsheet that quickly calculates the required coefficients for a desired response. This spreadsheet can be found in the "Wireline Telecoms" section in the "Application Notes" area of the CML Microcircuits website, www.cmlmicro.com. (**The application note for the coefficient generator spreadsheet is applicable to many CML products that contain a programmable tone detector.**)

Please note that all examples provided in this application note should be considered as a starting point for a design. The coefficient generator spreadsheet provides the designer with the ability to tweak values for a particular application. Since each application has specific requirements, the designer is encouraged to experiment with the coefficient generator spreadsheet and optimize their own design.

The following example illustrates how this spreadsheet can be used to determine the CMX865A programmable tone detector coefficients for two common SIA tones:

<div align="center">

Tone Frequency 1: 1400Hz +/-5%
Tone Frequency 2: 2300Hz +/-5%

</div>

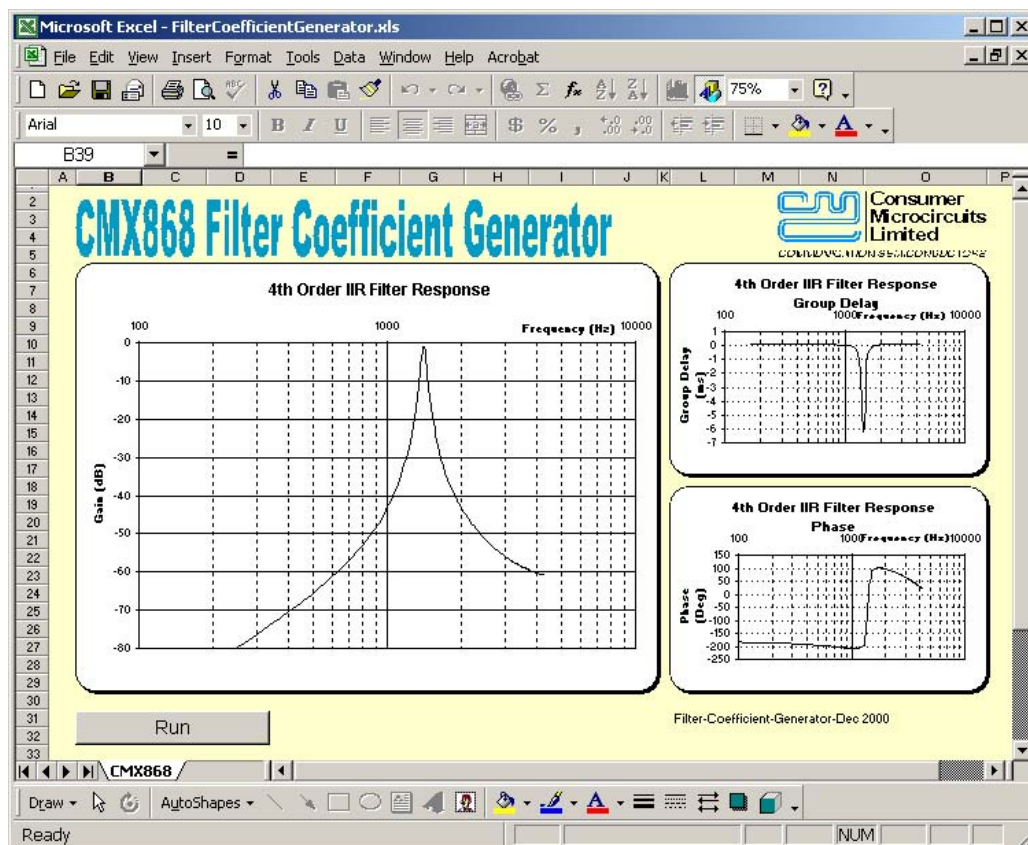When the coefficient generator spreadsheet is started, the opening screen appears like Figure 1.

Figure 1: CMX868 Filter Coefficient Generator main window.

In order to get started with the calculation process, the "Run" button in the bottom left corner should be clicked. A window similar to Figure 2 will appear:

Figure 2: Coefficient Generator Window

The main areas of interest in this window are:
- "Pole Frequency": the center frequency of interest, e.g. 1400 or 2300 in this example.
- "Ncycles": the number of cycles of the frequency of interest used to perform the detection. This parameter effectively specifies the minimum detection time.
- "Low Freq": lowest edge of detection window.
- "High Freq": highest edge of detection window.

To detect 1400Hz +/-5% in 20ms, the following parameters should be entered into the appropriate boxes:

Pole Frequency = 1400 (this value must be entered into both "Biquad A" and "Biquad B" windows)
Ncycles = 28 (1400Hz period = 0.71ms, and (20ms / 0.71ms) = 28)
Low Freq = 1330 (1400 * 0.95 = 1330)
High Freq = 1470 (1400 * 1.05 = 1470)

After these values have been entered in the appropriate spaces, click the "OK" button. The coefficients are calculated as shown in Figure 3.

Figure 3: Coefficients generated for a nominal 1400Hz detection

The values required to program the CMX865A to detect 1400Hz +/-5%, in 20ms, are contained in the "CMX868 2's Complement Coefficients" window. Since there are two programmable detectors (i.e. filters) in the CMX865A, the information in this window can be programmed for "Filter #1" (register words 2-14) or "Filter #2" (register words 15-27). In this example, the values for 1400Hz will be loaded into Filter #1.

The next step involves the calculation of coefficients for 2300Hz +/-5% detection in 20ms:
 Pole Frequency = 2300 (this value must be entered into both Biquad windows)
 Ncycles = 46 (2300Hz period = 0.43ms, and (20ms / 0.43ms) = 46)
 Low Freq = 2185 (2300 * 0.95 = 2185)
 High Freq = 2415 (2300 * 1.05 = 2415)

When these values are loaded into the spreadsheet, a click on the "OK" button will yield coefficients as shown in Figure 3.

Figure 4: Coefficients generated for a nominal 2300Hz detection

At this point, all required coefficients for 1400Hz and 2300Hz detection have been calculated. The coefficients must now be loaded into the CMX865A's "Programming Register". The CMX865A datasheet describes how the Programming Register should be loaded.

The host microcontroller should load the hexadecimal coefficients over the "C-BUS" serial interface, one coefficient at a time, into the CMX865A Programming Register as illustrated in the following table:

Word	Decimal	Hex	Comments
1	32769	0x8001	Index pointer
2	0	0x0000	
3	378	0x017A	
4	32389	0x7E85	
5	7769	0x1E59	
6	23051	0x5A0B	
7	0	0x0000	Coefficients for Filter #1 (1400Hz)
8	378	0x017A	
9	32389	0x7E85	
10	7769	0x1E59	
11	23051	0x5A0B	
12	28	0x001C	
13	182	0x00B6	
14	202	0x00CA	
15	0	0x0000	
16	384	0x0180	
17	32383	0x7E7F	
18	7664	0x1DF0	
19	31731	0x7BF3	
20	0	0x0000	Coefficients for Filter #2 (2300Hz)
21	384	0x0180	
22	32383	0x7E7F	
23	7664	0x1DF0	
24	31731	0x7BF3	
25	46	0x002E	
26	182	0x00B6	
27	202	0x00CA	

Table 1: Example Settings for CMX865A Programmable Tone Detector

Twenty-seven C-BUS transactions are required in order to program the CMX865A for two tone detection.

The process of spreadsheet manipulation and register loading can be repeated as necessary for any desired tone(s).

Once loaded, the coefficients will be retained until the power is removed or the Reset bit is toggled, so programming coefficients can often be handled as part of the start-up code. Note that the coefficients will be overwritten when the CMX865A is brought out of powersave mode.

6 SIA Communications Protocols and the CMX865A

Alarm panels adhering to the SIA communication protocols convey information with DTMF tones, pulsed tones, or FSK data bursts. The CMX865A can perform all of these functions. Some of the most commonly specified SIA protocols include:

- DTMF Format D1
- Pulse Format P1
- Pulse Format P2
- Pulse Format P3
- Pulse Format P4
- SIA Format
- Ademco® Contact ID

A typical security panel transaction begins when the alarm panel ("transmitter") detects an alarm condition and calls the monitoring station ("receiver") to report the condition. The receiver sends handshake information to the transmitter, and the alarm data is passed from transmitter to receiver thereafter. Some form of data verification is performed, and the receiver indicates correct reception by sending an acknowledgement (e.g. "kissoff" tone) to the transmitting alarm panel. The transmitter then terminates the call.

This application note focuses on the proper configuration of the CMX865A for a given SIA protocol. Specific information relating to data format and content is not covered in this document, and the reader is encouraged to consult the applicable SIA standard for such information. For further information regarding SIA standards, the reader may contact the SIA at:

<div align="center">

Security Industry Association
635 Slaters Lane
Suite 110
Alexandria VA 22314
703-683-2075
www.siaonline.org

</div>

6.1   DTMF Format D1
The DTMF D1 format uses DTMF signals to transfer alarm data from the transmitter to the receiver. This format is represented in the following figure.
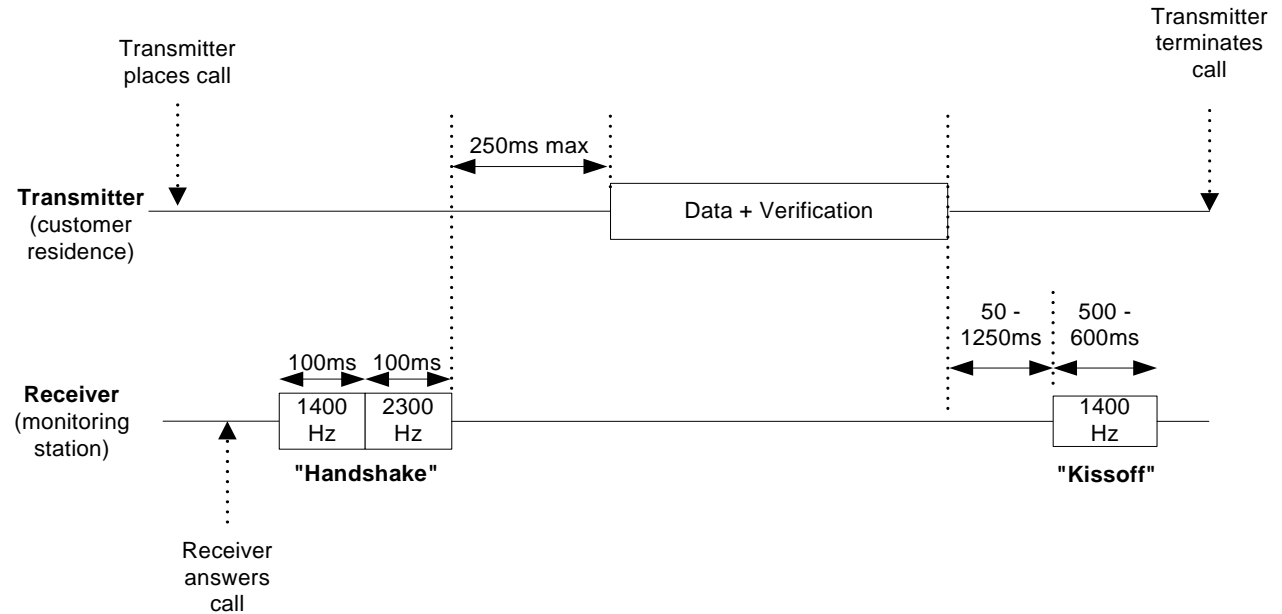
Figure 5: DTMF D1 Format (not drawn to scale)

As can be seen from this figure, a transmitter-based CMX865A should be configured to do the following:
- Transmitter places call:
    - Seize the telephone line and detect dial tone (call progress Rx).
    - Dial the DTMF digits for the monitoring station's telephone number (DTMF Tx).
    - Determine if ringback or busy tone is present (call progress Rx).
- Detect the incoming 'handshake' tone (programmed tone Rx).
- Transmit data with DTMF digits (DTMF Tx).
- Detect the incoming 'kissoff' tone (programmed tone Rx).
- Release the telephone line.

The following flowchart, Figure 6, describes the necessary steps to configure the CMX865A to perform the DTMF format. Please note that some of the timer settings (e.g. wait for dialtone timer setting) have been arbitrarily chosen and can be adjusted in the final application. Additionally, no details are provided regarding the data format because those details are provided in the relevant SIA standard.

Bring out of Powersave mode
General Reset ($01)
General Control ($E0)
    b8..7 = 11 (Device powered and reset)
Wait 20ms

Go off-hook, search for dial tone
General Control ($E0)
    b7 = 0 (Normal operation)
    b6 = 1 (IRQN pin enabled)
    b2 = 1 (Call Progress IRQ unmasked)
Rx Mode ($E2)
    b15..12 = 0001 (Tones detect mode selected)
    b11..9 = *user defined* (Rx level)
    b2..0 = 011 (Select call progress detect)

Start 5s timer for dialtone detection
(timer value arbitrarily chosen)

Wait for IRQ to indicate presence of dialtone.
On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.

If dialtone not present, release telephone line (go on hook).

Enter DTMF transmission mode and dial telephone number for monitoring station
Tx Mode ($E1)
    b15..12 = 0001 (DTMF/Tones)
    b11..9 = *user defined* (Tx level)
    b7..5 = *user defined* (DTMF twist)
    b4 = 1 (Enable DTMF transmit mode)
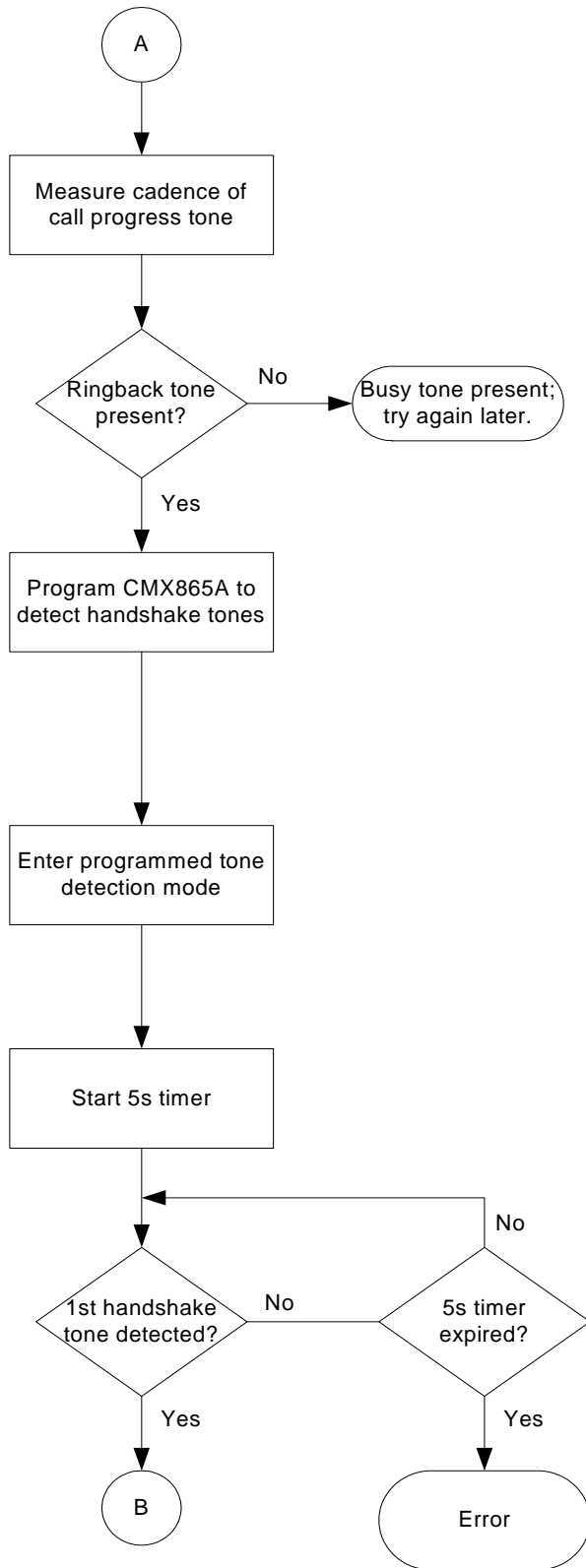   b3..0 = *user defined* (Select DTMF tone)
NOTE:
Set b4..0 = 1xxxx to turn on tone
Set b4..0 = 00000 to turn off tone

Start 5s timer for ringback/busy tone detection
(timer value arbitrarily chosen).
On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.

Figure 6: DTMF D1 Format with CMX865A (continued overleaf)

Measure cadence of call progress tone
Status ($E6) b10 = 1 indicates call progress energy detected.
Note that cadences are country specific

If busytone present, release line and repeat procedure at a later time.

Program CMX865A to detect handshake tones.
Check that *Status Register* ($E6) b13 = 1 (Programming Flag).
Load first word into *Programming Register* ($E8)
Wait for Programming Flag in *Status* Register ($E6) b13 =1.
Repeat process until all required coefficients are loaded.
NOTE: the Programming Register can be loaded at any time after powerup so long as the CMX865A is configured before tone detection is required.

Enter programmed tone detection mode.
General Control ($E0)
  b1..0 = 11 (1st and 2nd programmed tone IRQs unmasked)
Rx Mode ($E2)
  b2..0 = 100 (Enable programmble tone pair detect)
Rx mode and levels previously set

Start a 5s timer for handshake tone detection
(timer value arbitrarily chosen).

Search for 1400Hz & 2300Hz 'handshake' tones.
Assumes first programmable tone detector = 1400Hz and second programmable tone detector = 2300Hz.

Status Register ($E6):
  b6 = 1 (First programmed tone detected)
  b7 = 1 (Second programmed tone detected)

Figure 6: DTMF D1 Format with CMX865A (continued overleaf)

Flowchart content:

B → 2nd handshake tone detected?

2nd handshake tone detected? — No → 5s timer expired?
2nd handshake tone detected? — Yes → Start a 250ms timer

5s timer expired? — No → (loop back to B path)
5s timer expired? — Yes → Error

Start a 250ms timer → Timer expired?
Timer expired? — No → (loop back)
Timer expired? — Yes → Transmit data via DTMF

D → (joins into Yes path)

Transmit data via DTMF → Transmit verification via DTMF → Start a 50-1250ms timer → C

After both handshake tones are detected, start a 250ms timer.

(NOTE: 250ms is maximum allowed delay and there is no specified minimum value, so actual timer value can be less than this.)

When timer expires, transmit data via DTMF.
CMX865A should already be in DTMF Tx mode.
Tx Mode ($E1)
   b4..0 = *1xxxx* (Select DTMF tone)
NOTE:
Set b4..0 = 1xxxx to turn on tone
Set b4..0 = 00000 to turn off tone

When data transmission complete, transmit verification via DTMF.

Start a 50-1250ms timer for the kissoff tone.

Figure 6: DTMF D1 Format with CMX865A (continued overleaf)
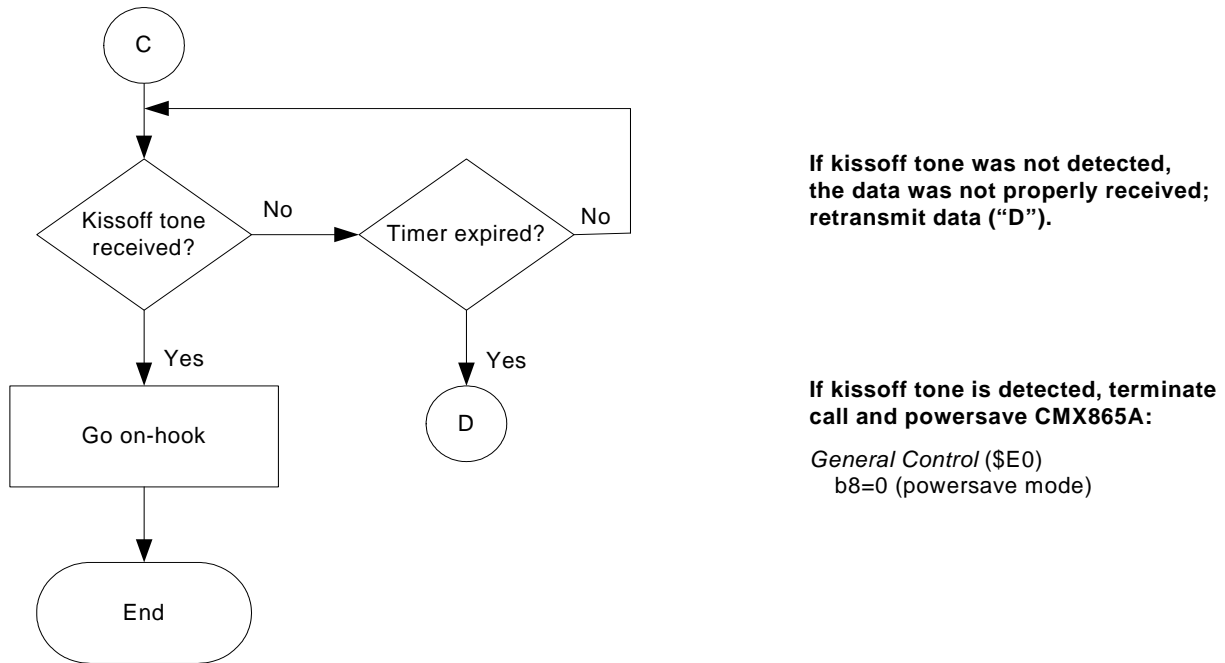
Figure 6: DTMF D1 Format with CMX865A

The DTMF format requires that the DTMF digits be pulsed on and off. It is possible to turn off the CMX865A programmable tone generator by setting Tx Mode ($E1) b14..12=000, but this will disable the CMX865A transmitting amplifier and prevent the line interface circuit from presenting the correct impedance to the line. An alternate plan is to select "No tone" as the transmitted tone by selecting *Tx Mode* ($E1) b4..0=00000. This will allow the carrier to be temporarily secured while ensuring that the correct impedance is presented to the telephone network.

For more information on the DTMF D1 Format, please review the SIA standard:
Digital Communications Technical Report – Generic Overview – of Security Industry Communicator Formats, SIA DC-02-1992.02(R2000.05).

6.2   Pulse Formats

The four SIA "pulse formats" convey information by cycling a carrier on and off. Each pulse format has different values for handshake tone, carrier frequency, data modulation rate, etc. The general flowpath for a pulse format transaction is depicted in the following figure.
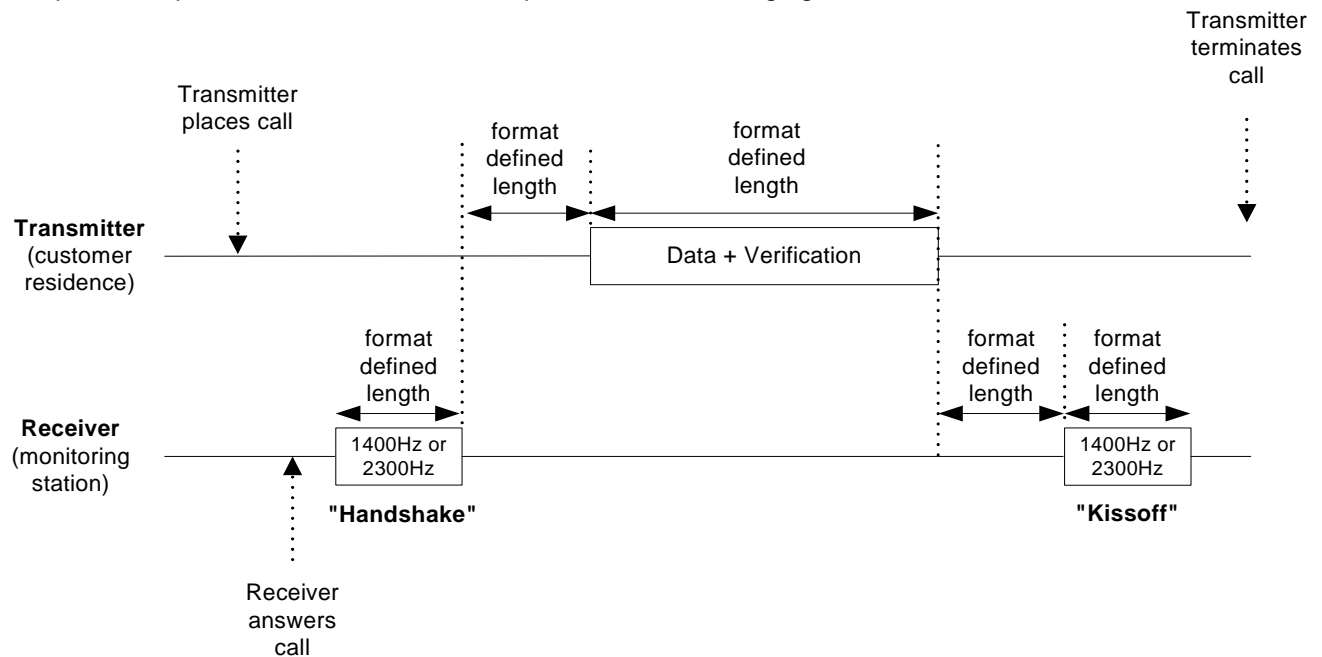
Figure 7: Pulse Format (not drawn to scale)

The CMX865A must be configured as follows to support the SIA pulse formats:
- Transmitter places call:
  - Seize the telephone line and detect dial tone (call progress Rx).
  - Dial the DTMF digits for the monitoring station's telephone number (DTMF Tx).
  - Determine if ringback or busy tone is present (call progress Rx).
- Detect the incoming 'handshake' tone (programmed tone Rx).
- Transmit data by turning the carrier frequency on and off (programmed tone Tx).
- Detect the incoming 'kissoff' tone (programmed tone Rx).
- Release the telephone line.

The programmable tone encoder/decoder of the CMX865A plays a pivotal role in this application. The programmable tone decoder is used for both the 'handshake' and 'kissoff' tones, while the programmable tone encoder is used to generate the carrier for data transmission.

Generally speaking, the 'handshake' and 'kissoff' tones are the same for a particular pulse format. For example, pulse format P1 uses 1400Hz +/- 5% for both the handshake and kissoff tones. Consequently, once the CMX865A programmable tone decoder has been programmed for the handshake tone, it doesn't need to be reprogrammed for the kissoff tone.

Three different frequencies are used for handshake and kissoff tones in the four SIA pulse formats. Each pulse format only requires one of these three tones, however, and the corresponding CMX865A Programming Register settings for tone detection are provided in the following table:

Word #	1400Hz +/- 2%		1400Hz +/- 5%		2300Hz +/- 5%	
	Dec	Hex	Dec	Hex	Dec	Hex
1	32769	**8001**	32769	**8001**	32769	**8001**
2	0	**0000**	0	**0000**	0	**0000**
3	378	**017A**	378	**017A**	384	**0180**
4	32389	**7E85**	32389	**7E85**	32383	**7E7F**
5	7769	**1E59**	7769	**1E59**	7664	**1DF0**
6	23051	**5A0B**	23051	**5A0B**	31731	**7BF3**
7	0	**0000**	0	**0000**	0	**0000**
8	378	**017A**	378	**017A**	384	**0180**
9	32389	**7E85**	32389	**7E85**	32383	**7E7F**
10	7769	**1E59**	7769	**1E59**	7664	**1DF0**
11	23051	**5A0B**	23051	**5A0B**	31731	**7BF3**
12	28	**001C**	28	**001C**	46	**002E**
13	188	**00BC**	182	**00B6**	182	**00B6**
14	195	**00C3**	202	**00CA**	202	**00CA**

Table 2: CMX865A Programmable Tone Detector Settings for Pulse Formats

There are three different carrier frequencies used to transmit information for the four SIA pulse formats (one carrier frequency is used for two different pulse formats). The required Programming Register (tone encoder) settings for these frequencies are shown in the following table:

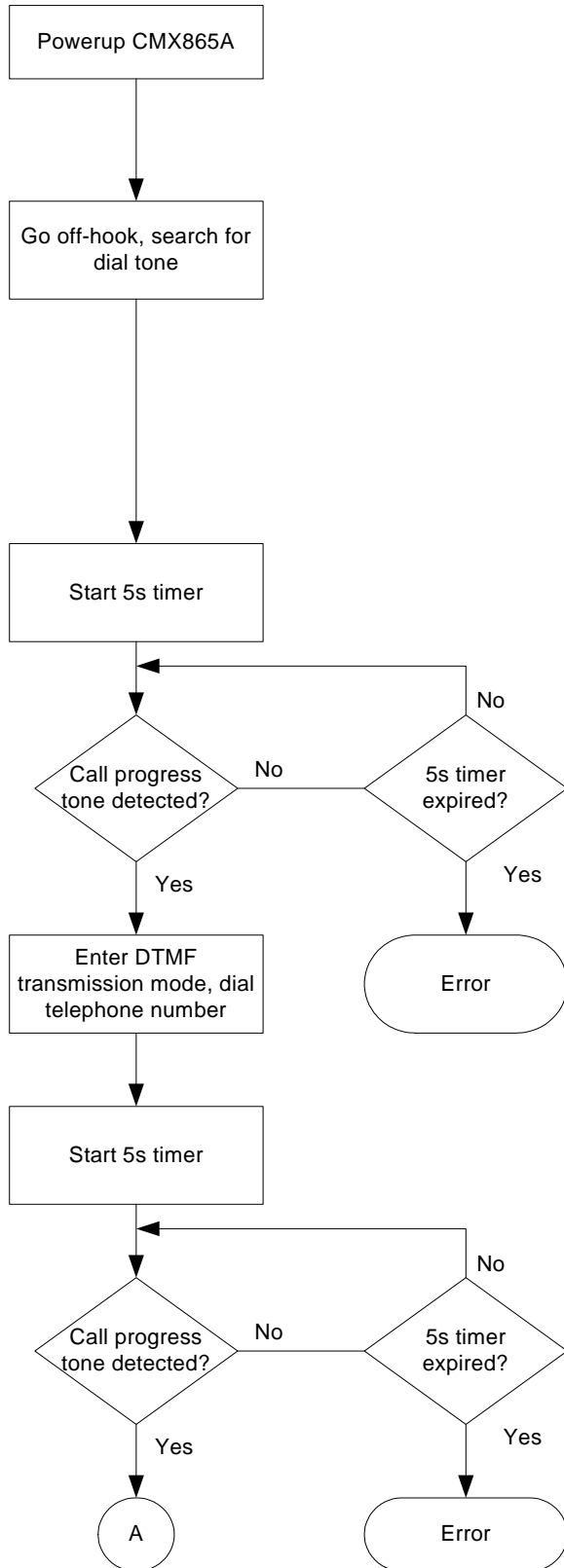| Frequency Information		
	Dec	Hex
1800Hz	6145	**1801**
1850Hz	6316	**18AB**
1900Hz	6487	**1956**
Amplitude Information		
	Dec	Hex
0.5Vrms	15630	**3D0E**

Table 3: CMX865A Programmable Tone Encoder Settings for Pulse Formats (V_{DD}=3V)

The 0.5Vrms tone amplitude mentioned above was selected arbitrarily. The tone amplitude can be adjusted with register settings derived from the following formula:

$$\text{Value to be entered} = \text{Hex value of } [\text{desired } V_{RMS} * (93780 / V_{DD})]$$

Data is conveyed in the pulse formats by turning a carrier on and off. It is possible to turn off the CMX865A programmable tone generator by setting *Tx Mode* ($E1) b15..12=0000, but this will disable the CMX865A transmitting amplifier and prevent the line interface circuit from presenting the correct impedance to the line. An alternate plan is to select "No tone" as the transmitted tone by selecting Tx Mode ($E1) b4..0=00000. This will allow the carrier to be temporarily secured while ensuring that the correct impedance is presented to the telephone network.

The general flow of the four SIA pulse formats is depicted in Figure 8.

Powerup CMX865A	**Bring out of Powersave mode** *General Reset ($01)* *General Control* ($E0) b8..7 = 11 (Device powered and reset) Wait 20ms

Go off-hook, search for dial tone	**Go off-hook, search for dial tone**
	General Control ($E0)
	    b7 = 0 (Normal operation)
	    b6 = 1 (IRQN pin enabled)
	    b2 = 1 (Call Progress IRQ unmasked)
	Rx Mode ($E2)
	    b15..12 = 0001 (Tones detect mode selected)
	    b11..9 = *user defined* (Rx level)
	    b2..0 = 011 (Select call progress detect)

Start 5s timer	**Start 5s timer for dialtone detection** (timer value arbitrarily chosen)
Call progress tone detected? — No — 5s timer expired?	**Wait for IRQ to indicate presence of dialtone.** On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected. If dialtone not present, release telephone line (go on hook).

Enter DTMF transmission mode, dial telephone number / Error	**Enter DTMF transmission mode and dial telephone number for monitoring station**
	Tx Mode ($E1)
	    b15..12 = 0001 (DTMF/Tones)
	    b11..9 = *user defined* (Tx level)
	    b7..5 = *user defined* (DTMF twist)
	    b4 = 1 (Enable DTMF transmit mode)
	    b3..0 = *user defined* (Select DTMF tone)
	NOTE:
	Set b4..0 = 1xxxx to turn on tone
	Set b4..0 = 00000 to turn off tone

Start 5s timer	
Call progress tone detected? — No — 5s timer expired?	**Start 5s timer for ringback/busy tone detection** (timer value arbitrarily chosen). On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.
A / Error	

Figure 8: Pulse Formats with the CMX865A (continued overleaf)

```
        ( A )
          |
          v
┌──────────────────┐
│ Measure cadence of│
│ call progress tone│
└──────────────────┘
          |
          v
      ╱ Ringback ╲   No    ╭──────────────────╮
     ╱   tone     ╲──────> │ Busy tone present;│
     ╲  present?  ╱        │ try again later.  │
      ╲          ╱         ╰──────────────────╯
          | Yes
          v
┌──────────────────┐
│ Program CMX865A to│
│ detect handshake  │
│ tones             │
└──────────────────┘
          |
          v
┌──────────────────┐
│ Enter programmed  │
│ tone detection    │
│ mode              │
└──────────────────┘
          |
          v
┌──────────────────┐
│   Start 5s timer  │
└──────────────────┘
          |
          v<─────────────────────── No
      ╱ 1st       ╲  No    ╱  5s timer ╲
     ╱ handshake   ╲──────╱   expired?  ╲
     ╲ tone        ╱       ╲            ╱
      ╲ detected? ╱         ╲          ╱
          | Yes                 | Yes
          v                     v
        ( B )             ╭──────────╮
                          │  Error   │
                          ╰──────────╯
```

Measure cadence of call progress tone
Status ($E6) b10 = 1 indicates call progress energy detected.
Note that cadences are country specific

If busytone present, release line and repeat procedure at a later time.

Program CMX865A to detect handshake tones.
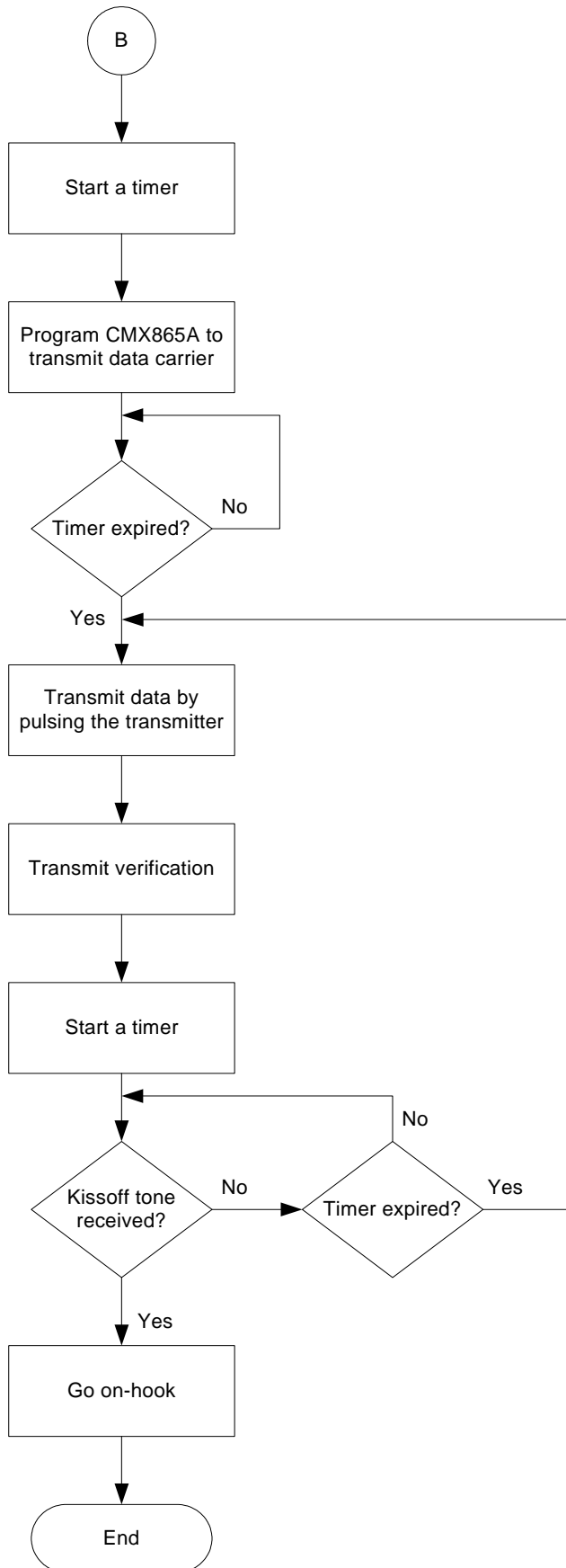Check that *Status Register* ($E6) b13 = 1 (Programming Flag).

Load first word into *Programming Register* ($E8)
Wait for Programming Flag in *Status* Register ($E6) b13 =1.
Repeat process until all required coefficients are loaded.
NOTE: the Programming Register can be loaded at any time after powerup so long as the CMX865A is configured before tone detection is required.

Enter programmed tone detection mode.
General Control ($E0)
  b1..0 = 11 (1st and 2nd programmed tone IRQs unmasked)
Rx Mode ($E2)
  b2..0 = 100 (Enable programmble tone pair detect)
Rx mode and levels previously set

Start a 5s timer for handshake tone detection
(timer value arbitrarily chosen).

Search for 1400Hz & 2300Hz 'handshake' tones.
Assumes first programmable tone detector = 1400Hz and second programmable tone detector = 2300Hz.
Status Register ($E6):
  b6 = 1 (First programmed tone detected)
  b7 = 1 (Second programmed tone detected)

Figure 8: Pulse Formats with the CMX865A (continued overleaf)

B

Start a timer

After handshake tone detected, start a timer.
(NOTE: Timer duration varies between formats, and the specific value is defined in the SIA standard.)

Program CMX865A to transmit data carrier

Program CMX865A to transmit data carrier.
Check that *Status* b13 = 1 (Programming Flag).
Load first word into Programming Register.
Wait for Programming Flag = 1.
Repeat process until all required values are loaded.
NOTE: the Programming Register can be loaded at any time after powerup so long as the CMX865A is configured before tone generation is required.

Timer expired? — No

Yes

Transmit data by pulsing the transmitter

When timer expires, transmit data by cycling the transmitter.

CMX865A should already be in DTMF Tx mode.
Tx Mode ($E1)
 b4..0 = 1xxxx (Select desired DTMF tone)
NOTE:
Set b4..0 = 1xxxx to turn on tone
Set b4..0 = 00000 to turn off tone

Transmit verification

When data transmission is complete, transmit verification.

Consult SIA standard for method of verification as this varies from one format to another.

Start a timer

After verification has been transmitted, start a timer.
(NOTE: Timer duration varies between formats, and the specific value is defined in the SIA standard.)

Kissoff tone received? — No → **Timer expired?** No / Yes

**If kissoff tone not detected, the data was not properly received; retransmit data.**

If kissoff tone is detected, terminate call and powerdown CMX865A.
General Control ($E0)
 b8=0 (powersave mode)

Yes

Go on-hook

End

Figure 8: Pulse Formats with the CMX865A (continued overleaf)

For more information on the SIA pulse formats, please review the SIA standard:
Digital Communications Technical Report – Generic Overview – of Security Industry Communicator Formats, SIA DC-02-1992.02(R2000.05).

6.3   SIA Format

The SIA Format uses Bell 103 signaling at 300bps to convey information between transmitter and receiver. The general flowpath of the SIA Format is depicted in the following figure.
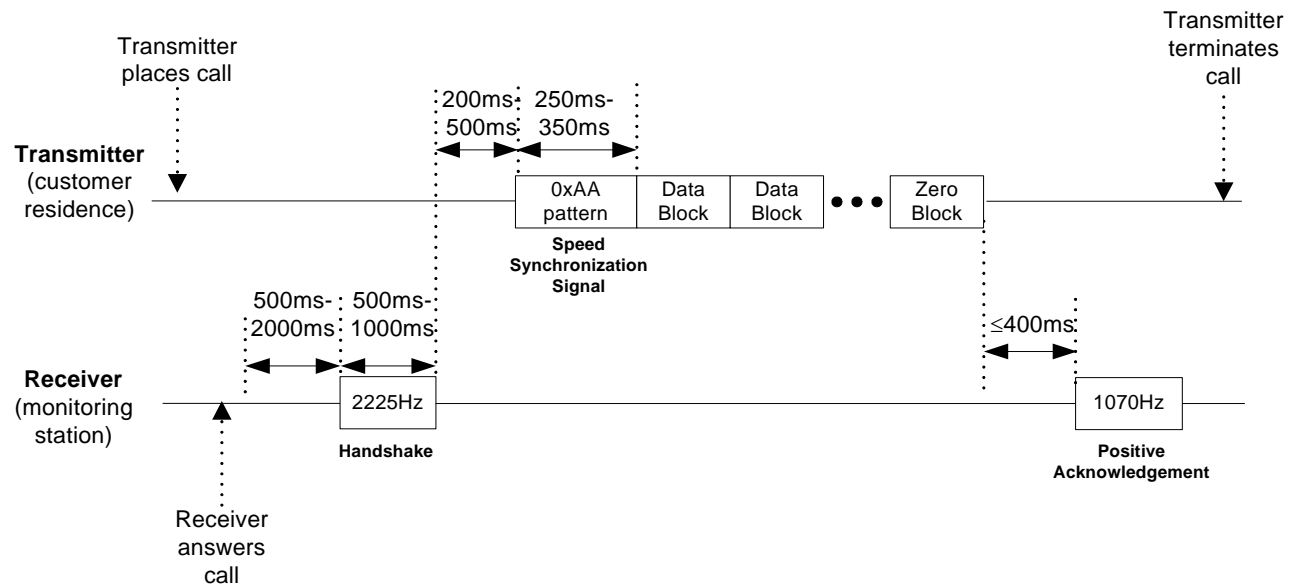
Figure 9: Simplified SIA Format (not drawn to scale)

The SIA format also allows the exchange of data blocks from receiver to transmitter, but this has been omitted from the figure for simplicity.

This format requires half-duplex data exchange in both directions. (Note: the "slow speed" of 110 baud specified in the SIA Format standard is not supported in the CMX865A.) The byte protocol for the SIA format is:
- 1 start bit (at sender's space frequency)
- 8 data bits, LSB first (logic zero is at the sender's space frequency)
- 1 odd parity bit
- 2 stop bits (at sender's mark frequency)

The SIA format is rather complex and a detailed discussion of its possible variations is beyond the scope of this application note, but from the standpoint of the CMX865A, the following functions are germane:
- Call progress detection (dial tone, busy tone, ringback tone)
- Answer tone detection (2225Hz "answer tone" used for handshake tone)
- Bell 103 300bps modem mode (for speed synchronization signal, data transfer, flags and acknowledgements)

The following flowchart, Figure 10, illustrates one possible example of an SIA format transaction, along with the corresponding CMX865A configurations.

Powerup CMX865A

Bring out of Powersave mode
General Reset ($01)
General Control ($E0)
    b8..7 = 11 (Device powered and reset)
Wait 20ms

Go off-hook, search for dial tone

Go off-hook, search for dial tone
General Control ($E0)
    b7 = 0 (Normal operation)
    b6 = 1 (IRQN pin enabled)
    b2 = 1 (Call Progress IRQ unmasked)
Rx Mode ($E2)
    b15..12 = 0001 (Tones detect mode selected)
    b11..9 = *user defined* (Rx level)
    b2..0 = 011 (Select call progress detect)

Start 5s timer

Start 5s timer for dialtone detection
(timer value arbitrarily chosen)

Call progress tone detected?

5s timer expired?

No

No

Yes

Yes

Wait for IRQ to indicate presence of dialtone.
On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.

If dialtone not present, release telephone line (go on hook).

Enter DTMF transmission mode, dial telephone number

Error

Enter DTMF transmission mode and dial telephone number for monitoring station
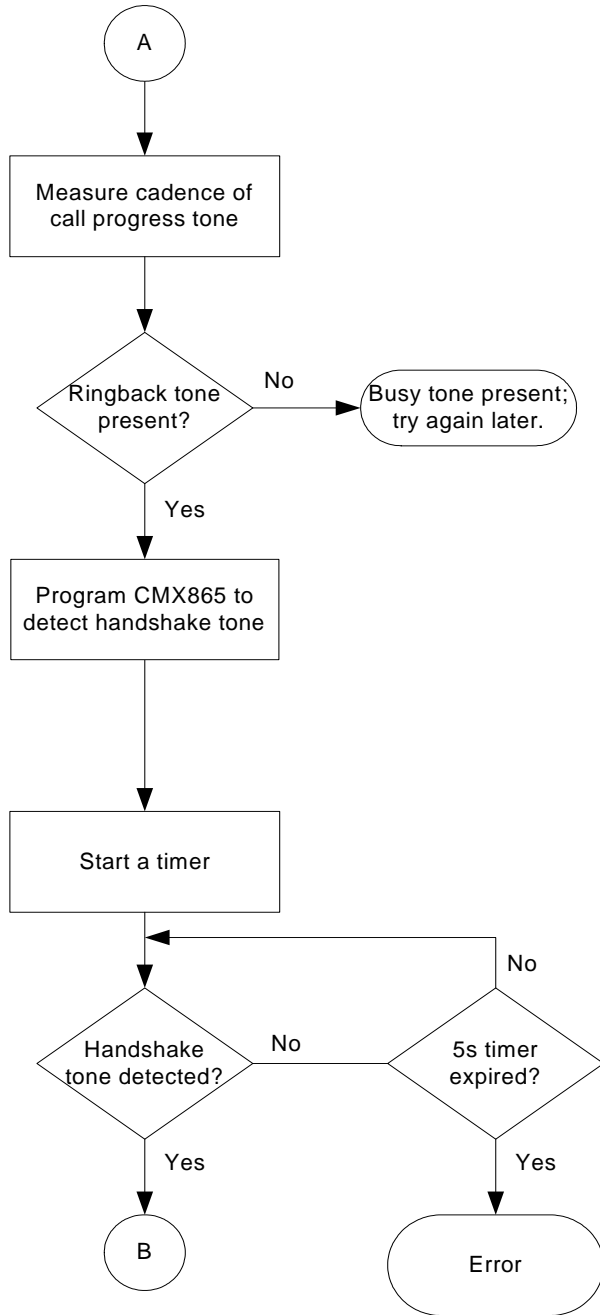Tx Mode ($E1)
    b15..12 = 0001 (DTMF/Tones)
    b11..9 = *user defined* (Tx level)
    b7..5 = *user defined* (DTMF twist)
    b4 = 1 (Enable DTMF transmit mode)
    b3..0 = *user defined* (Select DTMF tone)
NOTE:
Set b4..0 = 1xxxx to turn on tone
Set b4..0 = 00000 to turn off tone

Start 5s timer

Call progress tone detected?

5s timer expired?

No

No

Yes

Yes

Start 5s timer for ringback/busy tone detection
(timer value arbitrarily chosen).
On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.

A

Error

Figure 10: SIA Format with the CMX865A (continued overleaf)

A

Measure cadence of
call progress tone

Measure cadence of call progress tone
Status ($E6) b10 = 1 indicates call progress energy
detected.
Note that cadences are country specific

Ringback tone
present? — No → Busy tone present;
try again later.

**If busytone present, release line and repeat
procedure at a later time.**

Yes

Program CMX865 to
detect handshake tone

Program CMX865A to detect handshake tone
The handshake tone is 2225Hz for which the CMX865
provides a standard detector.
General Control ($E0)
   b0 = 1 (2225Hz detect IRQ unmasked)
Rx Mode ($E2)
   b2..0 = 010 (2225Hz answer tone detect)
Rx Mode and Level previously set

Start a timer

Start a 5s timer for handshake tone detection
(timer value arbitrarily chosen).

No

Handshake
tone detected? — No → 5s timer
expired?

Search for handshake tone.
On interrupt, monitor for Status ($E6) b10 = 1,
indicates 2225Hz handshake tone detected.

Yes   Yes

B   Error

Figure 10: SIA Format with the CMX865A (continued overleaf)

B
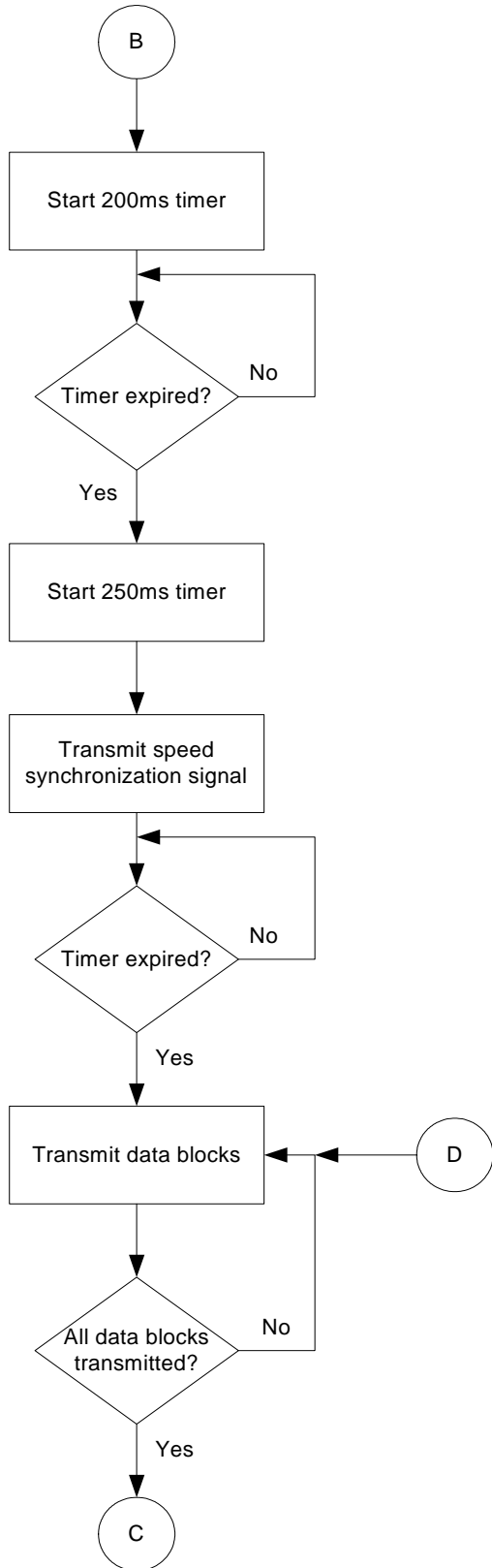
Start a 200ms timer for delay before speed synchronization signal.
Valid range is between 200ms-500ms.

Start 200ms timer

Timer expired? — No

Yes

Start timer for speed synchronization signal.
Valid range is between 250ms-350ms.

Start 250ms timer

Program CMX865A to transmit speed synchronization signal.

General Control ($E0)
   b0 = 0 (mask previously unmasked bits)

Tx Mode ($E1)
   b15..12 = 0110 (Bell 103 300bps low band)
   b11..9 = *user defined* (Tx level)
   b4,3 = 11 (synchronous mode)
   b2..0 = 000 (continuous alternating 1s and 0s)

Transmit speed synchronization signal

Timer expired? — No

Yes

Transmit data blocks ← D

Transmit data blocks and configure to receive acknowledgement.

General Control ($E0)
   b3 = 1 (Tx Data Ready and Underflow IRQs unmasked)
   b0 = 1 (Rx Data Ready and Overflow IRQs unmasked)
Tx Data ($E3) = *user defined* (Load data bytes to transmit)
Tx Mode ($E1)
   b11..9 = *user defined* (Tx level)
   b4,3 = 00 (Start/stop mode with odd parity)
   b2..0 = 111 (8 data bits, 2 stop bits)
Rx Mode ($E2)
   b15..12 = 0111 (Bell 103 high band)
   b11-9 = *user defined* (Rx level)
   b5..3 = 110 (Start/stop mode)
   b2..0 = 111 (8 data bits + parity)

All data blocks transmitted? — No

Yes

C

Figure 10: SIA Format with the CMX865A (continued overleaf)
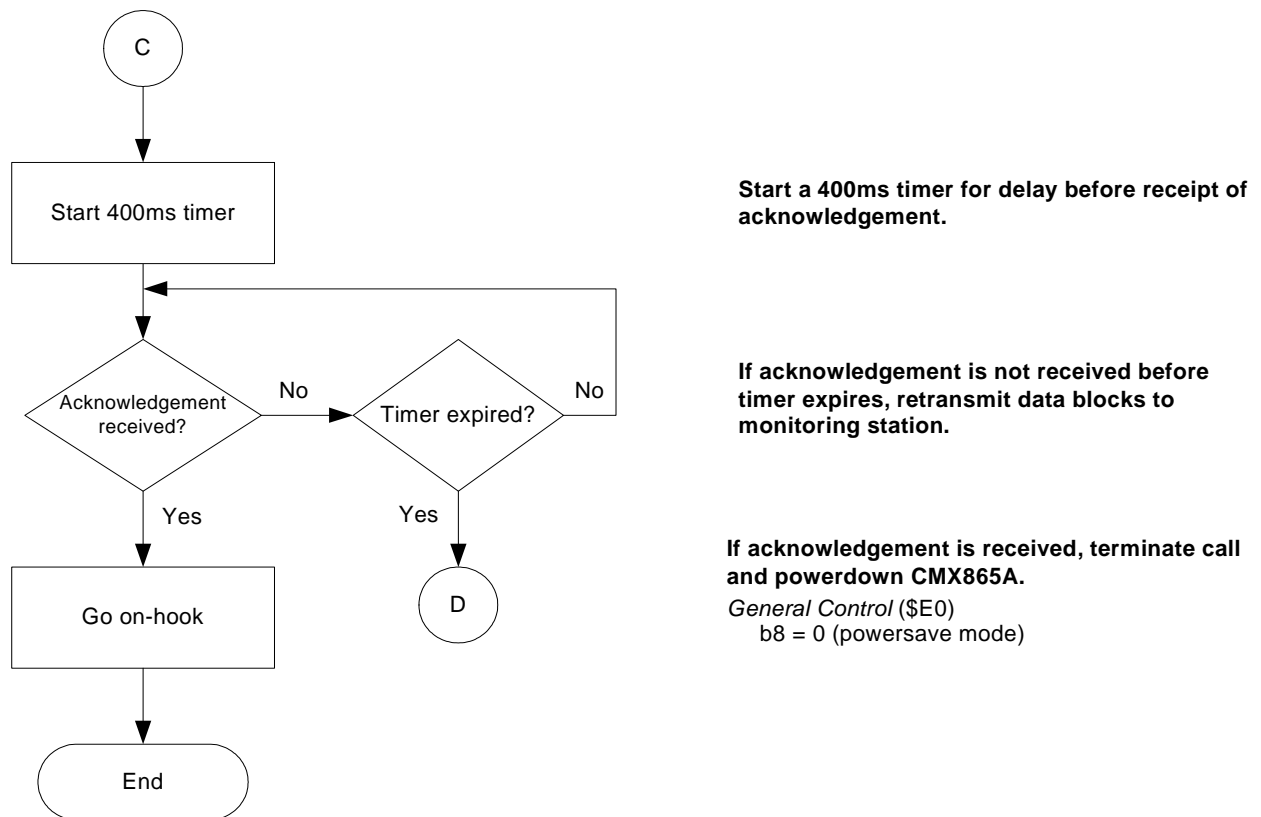
Figure 10: SIA Format with the CMX865A

The SIA Format uses odd parity in its byte protocol:
- Parity bit = 1 if sum of data bits is even.
- Parity bit = 0 if sum of data bits is odd.

The flowchart above includes configuration settings to allow the CMX865A to transmit odd parity information with the data. The SIA Format allows data blocks to be sent from transmitter-to-receiver and from receiver-to-transmitter, so the transmitting CMX865A at the customer's premise must also be able to receive a data block with odd parity.

The CMX865A provides a received data parity indication that can be used in this format. Status Register ($E6) b3 indicates the number of 1s in the received data byte (transmitted parity bit is not included in this calculation):
- $E6 b3=1 if even number of 1s are in received data byte.
- $E6 b3=0 if odd number of 1s are in received data byte.

The transmitted parity bit is not used in the parity calculation internal to the CMX865A. If the application requires that the transmitted parity bit be used in the parity calculation, the receiving CMX865A should be operated in synchronous UART mode so that the parity check can be performed in the host microcontroller.

For more information on the SIA Format, please review the SIA standard:
Security Communications – Digital Communications Standard – "SIA Format" Protocol - for Alarm System Communications, SIA DC-03-1990.01 (R2003.10).

6.4   Ademco®[2] Contact ID Format

The Ademco® Contact ID format uses DTMF signals to transfer alarm data from the transmitter to the receiver. This format is represented in the following figure.
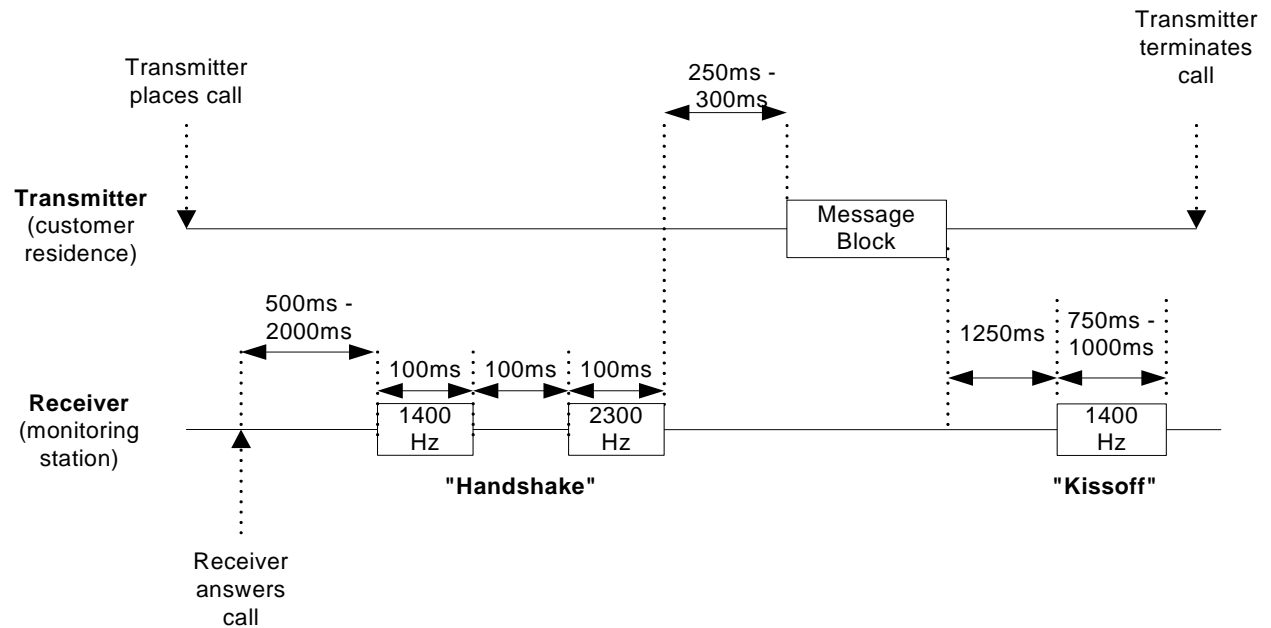

Figure 11: Ademco® Contact ID Format (not drawn to scale)

Only one message block is illustrated in the figure for simplicity. The Ademco® Contact ID format allows multiple message blocks to be transmitted so long as the kissoff tone is received between message blocks. Details concerning the data codes and inter-message time can be found in the SIA standard document.

As can be seen from this figure, a transmitter-based CMX865A must be configured as follows to support the Ademco® Contact ID format:

- Transmitter places call:
  - Go off-hook to seize the telephone line.
  - Enter call progress detection mode to search for dial tone.
  - Enter DTMF transmission mode to dial the digits for the receiver (monitoring station).
  - Enter call progress detection mode to search for busy tone or ringback tone.
- Enter programmed tone detection mode to search for handshake tones from receiver (monitoring station).
- Enter DTMF transmission mode to transmit message blocks (data) to the monitoring station.
- Enter programmed tone detection mode to search for "kissoff" tone from receiver.

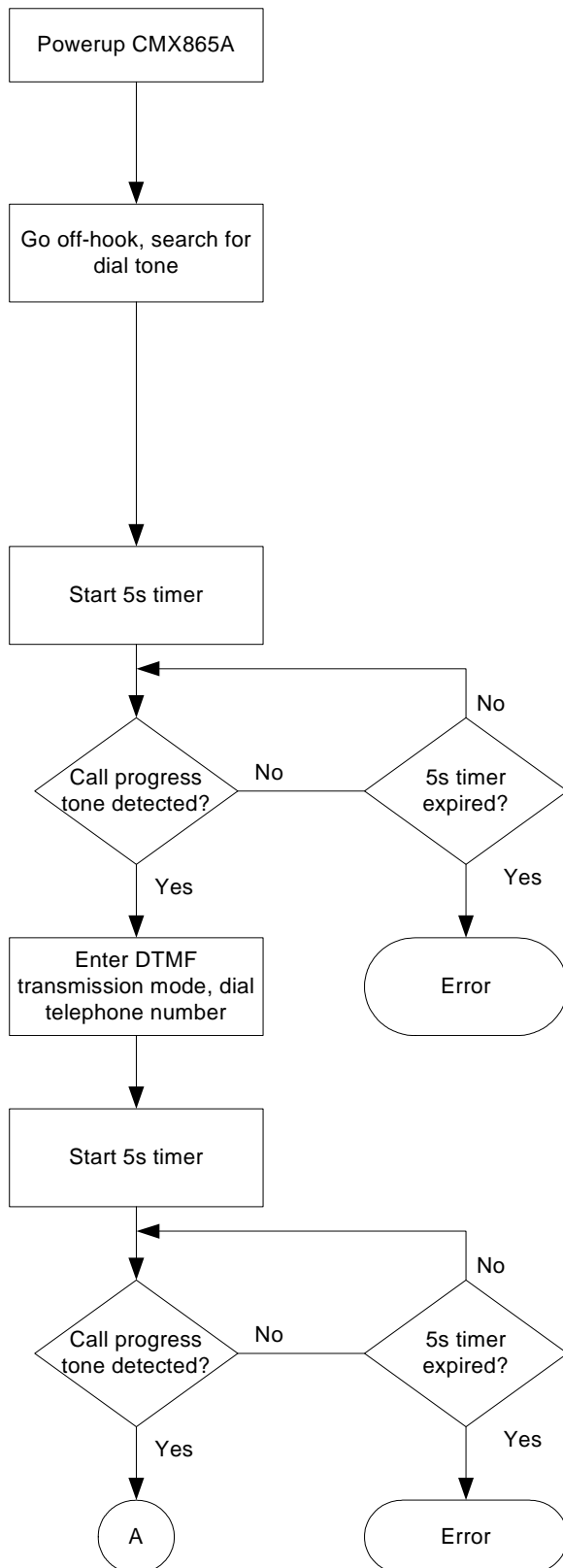[2] Ademco is a trademark of Honeywell Corporation.

There are a few mismatches between the DTMF tone pairs used in the Ademco® Contact ID format and the DTMF tone pairs identified in the CMX865A datasheet. The following table summarizes these differences:

Low Tone (Hz)	High Tone (Hz)	Ademco® Contact ID Assignment	CMX865A Assignment
697	1633	D	A
770	1633	E	B
852	1633	F	C
941	1633	Not used	D

Table 4: Ademco® Contact ID DTMF Mismatches with CMX865A

For example, to transmit a "D" character in the Ademco® Contact ID format, the CMX865A "A" DTMF symbol must be selected by programming the Tx Mode register ($E1), b4..0=11101. Other mismatches should be treated in the same manner.

The following flowchart, Figure 12, describes the necessary steps to configure the CMX865A to perform the Ademco® Contact ID format. Please note that some of the timer settings (e.g. wait for dialtone timer setting) have been arbitrarily chosen and can be adjusted in the final application. Only one message block is illustrated in the flowchart for simplicity, but the Ademco® Contact ID format allows multiple message blocks to be transmitted so long as the kissoff tone is received between message blocks.

Bring out of Powersave mode
General Reset ($01)
General Control ($E0)
    b8..7 = 11 (Device powered and reset)
Wait 20ms

Go off-hook, search for dial tone
General Control ($E0)
    b7 = 0 (Normal operation)
    b6 = 1 (IRQN pin enabled)
    b2 = 1 (Call Progress IRQ unmasked)
Rx Mode ($E2)
    b15..12 = 0001 (Tones detect mode selected)
    b11..9 = *user defined* (Rx level)
    b2..0 = 011 (Select call progress detect)

Start 5s timer for dialtone detection
(timer value arbitrarily chosen)

Wait for IRQ to indicate presence of dialtone.
On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.

If dialtone not present, release telephone line (go on hook).

Enter DTMF transmission mode and dial telephone number for monitoring station
Tx Mode ($E1)
    b15..12 = 0001 (DTMF/Tones)
    b11..9 = *user defined* (Tx level)
    b7..5 = *user defined* (DTMF twist)
    b4 = 1 (Enable DTMF transmit mode)
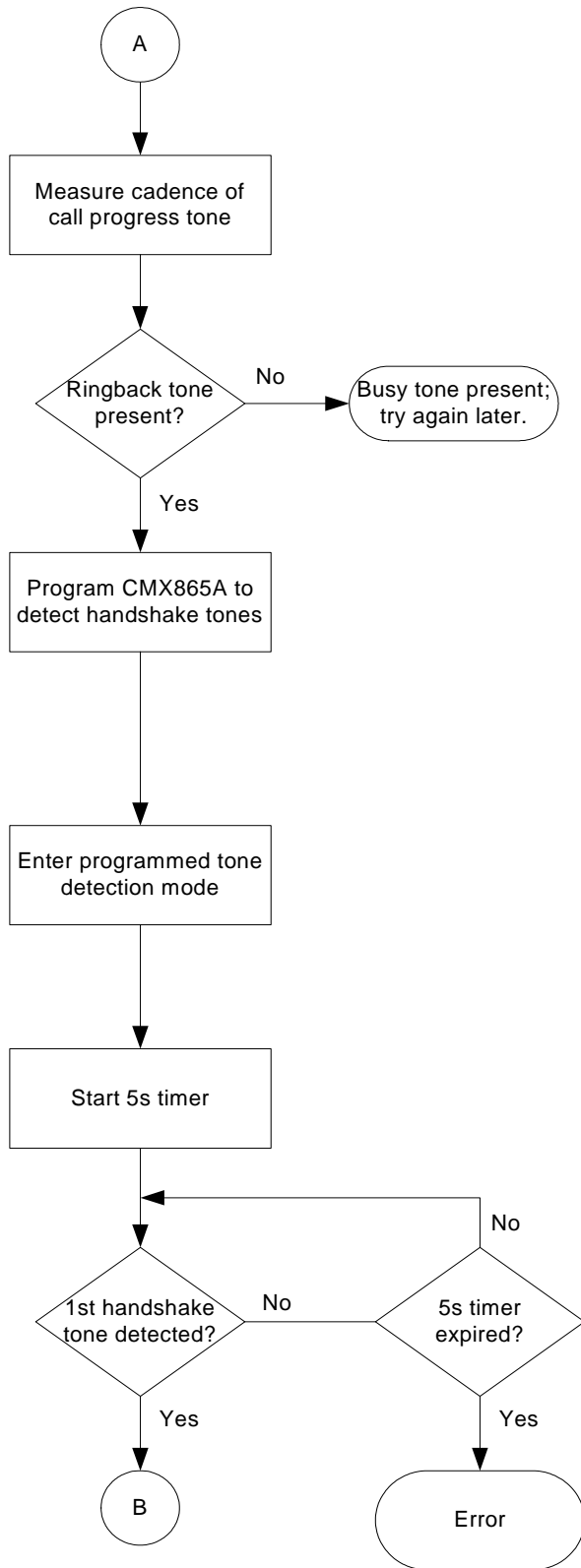   b3..0 = *user defined* (Select DTMF tone)
NOTE:
Set b4..0 = 1xxxx to turn on tone
Set b4..0 = 00000 to turn off tone

Start 5s timer for ringback/busy tone detection
(timer value arbitrarily chosen).
On interrupt, *Status* ($E6) b10 = 1 indicates call progress energy detected.

Figure 12: Ademco® Contact ID Protocol with CMX865A (continued overleaf)

Measure cadence of call progress tone
Status ($E6) b10 = 1 indicates call progress energy detected.
Note that cadences are country specific

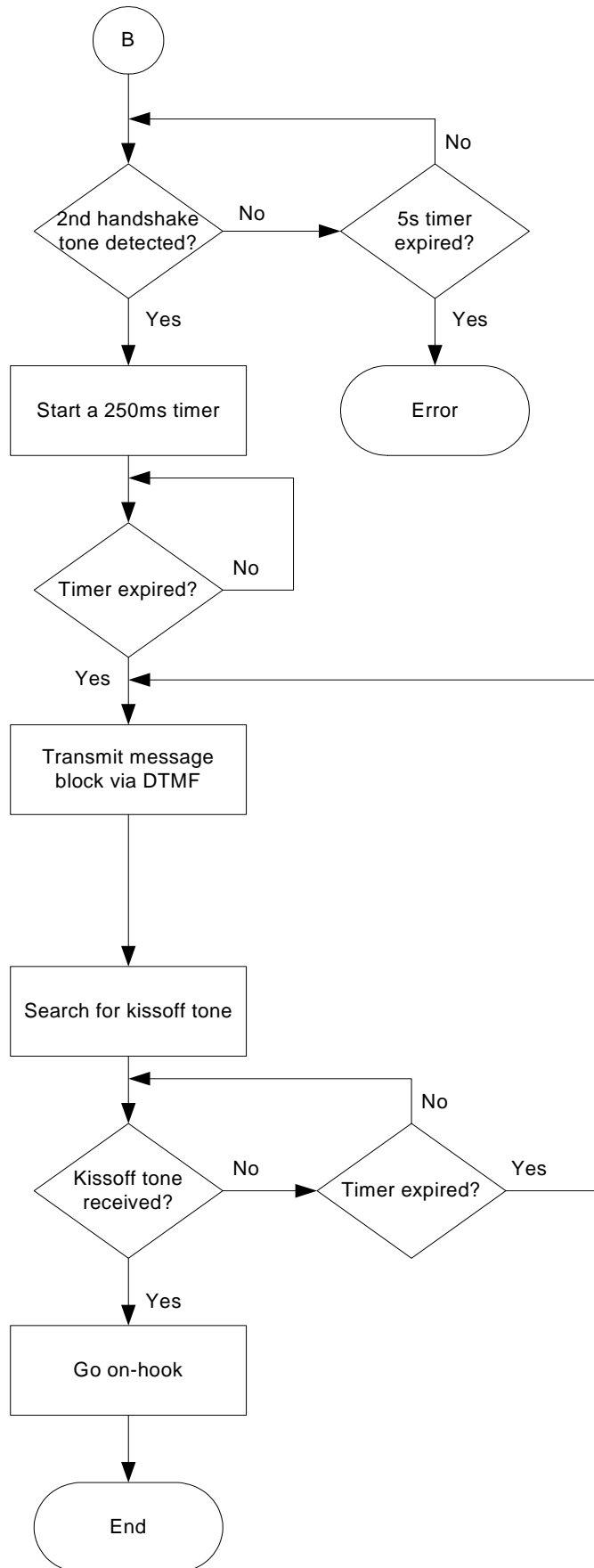**If busytone present, release line and repeat procedure at a later time.**

Program CMX865A to detect handshake tones.
Check that *Status Register* ($E6) b13 = 1 (Programming Flag).

Load first word into *Programming Register* ($E8)
Wait for Programming Flag in *Status* Register ($E6) b13 =1.
Repeat process until all required coefficients are loaded.

NOTE: the Programming Register can be loaded at any time after powerup so long as the CMX865A is configured before tone detection is required.

Enter programmed tone detection mode.
General Control ($E0)
  b1..0 = 11 (1st and 2nd programmed tone IRQs unmasked)
Rx Mode ($E2)
  b2..0 = 100 (Enable programmble tone pair detect)
Rx mode and levels previously set

Start a 5s timer for handshake tone detection
(timer value arbitrarily chosen).

Search for 1400Hz & 2300Hz 'handshake' tones.
Assumes first programmable tone detector = 1400Hz and second programmable tone detector = 2300Hz.

Status Register ($E6):
  b6 = 1 (First programmed tone detected)
  b7 = 1 (Second programmed tone detected)

Figure 12: Ademco® Contact ID Protocol with CMX865A (continued overleaf)

The flowchart (Figure 12) contains the following elements and annotations:

B (connector)

2nd handshake tone detected?
- No → **5s timer expired?**
  - No → (loops back up)
  - Yes → **Error**
- Yes → **Start a 250ms timer**

Status Register ($E6) b7 = 1 (Second programmed tone detected)

After both handshake tones are detected, start a 250ms timer.

(NOTE: 250ms is maximum allowed delay and there is no specified minimum value, so actual timer value can be less than this.)

Timer expired?
- No → (loops back)
- Yes → **Transmit message block via DTMF**

When timer expires, transmit data via DTMF.
CMX865A should already be in DTMF Tx mode.
Tx Mode ($E1)
  b4..0 = *1xxxx* (Select desired DTMF tone)
NOTES:
Set b4..0 = 1xxxx to turn on tone
Set b4..0 = 00000 to turn off tone

See text for a discussion of mismatches betwen Contact ID character mapping and CMX865 DTMF character mapping.

Search for kissoff tone

Start a 2250ms timer for the kissoff tone.
2250ms = 1250ms kissoff tone delay + 1000ms maximum duration of kisoff tone

Search for kissoff tone.
Status ($E6) b1 or b0 = 1 indicates kissoff tone detected.

Kissoff tone received?
- No → **Timer expired?**
  - No → (loops back)
  - Yes → (loops back to transmit)
- Yes → **Go on-hook**

NOTE:
Kissoff tone previously programmed in either first or second CMX865A programmed tone register.

If kissoff tone not detected, the data was not properly received; retransmit data.

If kissoff tone is detected, terminate call and powersave CMX865A.
  General Control ($E0)
    b8=0 (powersave mode)

End

Figure 12: Ademco® Contact ID Protocol with CMX865A

For more information on this format, please review the following SIA standard:
Digital Communication Standard - Ademco® Contact ID Protocol – for Alarm System Communications,
SIA DC-05-1999.09

7   Summary

Security alarm panels are used around the world as a cost-effective means of protecting people and property. The SIA has developed communication protocols using DTMF tones, pulsed tones and FSK signaling to exchange data between alarm panels and monitoring stations.

The CMX865A low-power modem IC can be easily configured to perform these SIA protocols, and its aggressive pricing makes it attractive for security panel applications. This document has described how the CMX865A can be configured to perform common SIA communication protocols. It is hoped that this application note will assist the designer in implementing the CMX865A into SIA-compliant alarm panels.

8   References

1) *Digital Communications Technical Report – Generic Overview – of Security Industry Communicator Formats*, SIA DC-02-1992.02(R2000.05), Security Industry Association standard.
2) *Security Communications – Digital Communications Standard – "SIA Format" Protocol - for Alarm System Communications*, SIA DC-03-1990.01 (R2003.10), Security Industry Association standard.
3) *Digital Communication Standard - Ademco® Contact ID Protocol – for Alarm System Communications*, SIA DC-05-1999.09, Security Industry Association standard.