CML Semiconductor Products

Application Information

V.22 Full Duplex Call Set-Up on the CMX867

This document provides details on the V.22 wire-line communications protocol and pointers on how the handshaking sequence can be achieved with the CMX867 modem IC.
Also included is additional information on the V.14 and V.25 protocols.

This document is designed to be used in conjunction with the current CMX867 datasheet.
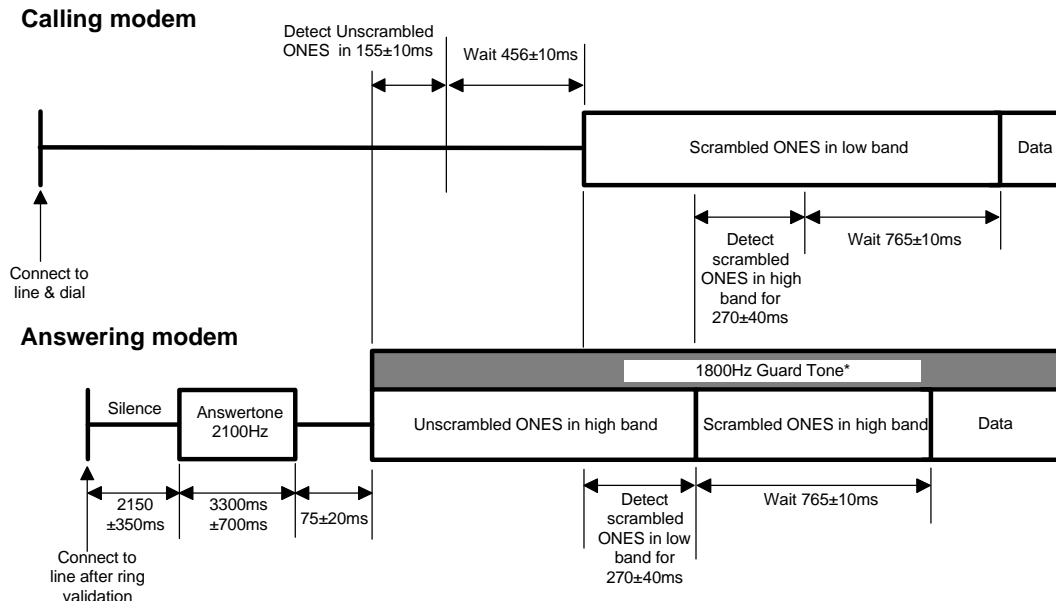
Calling modem

Detect Unscrambled ONES in 155±10ms

Wait 456±10ms

| Scrambled ONES in low band | Data |

Detect scrambled ONES in high band for 270±40ms

Wait 765±10ms

Connect to line & dial

Answering modem

1800Hz Guard Tone*

| Silence | Answertone 2100Hz | | Unscrambled ONES in high band | Scrambled ONES in high band | Data |

2150 ±350ms

3300ms ±700ms

75±20ms

Detect scrambled ONES in low band for 270±40ms

Wait 765±10ms

Connect to line after ring validation

Fig. 1. V.22 Connect Handshake.
Guard tones are country dependent

1. **V.22 Handshake**

Figure 1 above is an adaptation of the V.22 Connect Handshake from Volume VIII - Fascicle VIII.1 of the Blue Book series published by the ITU (formerly CCITT) November 1988. The actions taken by the calling and answering modems are illustrated with flow charts and 'C-Bus' command strings for implementation on the CMX867 – see Figures 2 and 3.

1.1 Points to note

(a) The modem can connect to line by setting the RDRVN output (CMX867 pin 3) to Vss in order to drive a relay or semiconductor switch. The On-Hook level of RDRVN must be Vdd otherwise spurious line seizures could occur on initial power up or reset.

(b) Dial tone will be present typically 300ms after going Off-Hook. If dialling without dial tone detection then a pause of at least 2.7s (max. 8s) should be introduced before dialling commences. If no dial tone is detected within 5 seconds of going Off-Hook then the modem should return On-Hook.

(c) DTMF digits should be typically 100ms long with 100ms silence gaps in between.

(d) Figures 2 and 3 are intended to give general guidance on the use of registers. They do not show software timeout features which avoid lockup on certain loops. The treatment of interrupts (enabling, servicing & disabling) is not fully shown since these are options which may be implemented for a particular design. Guard tones are also not included.

(e) On the General Switched Telephone Network (GSTN), the calling modem should transmit on the low band and receive in the high band.

(f) Error handling has not been included since this aspect is implementation dependent.

(g) The detection of in-band energy is indicated by b10 of the *Status* register ($E6). It is advisable to check this bit in both call set-up and data modes where signals are due to be received since it is possible for noise to occasionally give false indication of bit patterns on a line which does not in fact have a true signal.

(h) The following bits in the *Transmit Mode* ($E1) and *Receive Mode* ($E2) registers need to be set according to the circuit and function requirement:-

Tx Mode	b11…b9	= Tx gain
Tx Mode	b8,b7	= 550Hz/1800Hz Guard Tone enabling
Tx Mode	b4…b0	= USART mode and format
Rx Mode	b11…b9	= Rx gain
Rx Mode	b5…b0	= USART mode and format

(i) Some points are shown on the drawing in Fig. 1 which relate to circuit switching on the host external port (e.g. "Detect scrambled ONES in high band for 270±40ms" is a point at which circuit 109 can be switched.)

(j) Register bit states on the flow charts assume a General Reset ($01) has been issued and the normal start-up procedure has been followed prior to entering the Calling and the Answering procedures. Each state during these two procedures lists only the register bit changes that are required to move from the previous state.

For both Calling and Answering modems:-
After connecting power to the modem:-
Write to the *General Control* ($E0) register
        b7 = 1 (reset internal circuitry)
        b8 = 1 (select normal operating mode)
Wait 20ms
Write to the *General Control* ($E0) register
        b8 = 0 (powersave mode selected)
        b6 = 1 (enable IRQN output)
        b5 = 1 (unmask Ring Detect IRQ)
The modem is now in powersave mode but programmed to interrupt the controlling micro controller when ringing is detected.
Proceed as shown in the flow charts for Calling Modem

For the Auto-Answering Modem when ring interrupt is serviced:-
Maintain the CMX867 in Powersave mode but monitor the *Status* register ($E6) to check the ring detect on/off timing. When valid ringing has been received, proceed as shown in the flow charts for Answering Modem.

2. **V.25 and V.14 implementation on the CMX867**

2.1. V.25 timing has been included within figures 2 and 3. The only point not covered is the optional transmission of 1300Hz by the calling modem in the

2

period after dialling and before Answer tone recognition. The cadence is 0.6s ON, 1.75s OFF (typ.).
For V.22 this tone is NOT permitted, but the following details are included for other applications:

TX Mode ($E1)
      b15..b12 = 0001 (select DTMF / Tones)
      b3..b0 = 1001 (1300Hz Calling tone)

2.2. The V.25 automatic answering sequence shall be transmitted from the answer mode modem on international GSTN connections. The transmission of the V.25 sequence may be omitted on point-to-point leased circuits, or on national connections on the GSTN where permitted by the Administration.

2.3. V.14 specifies stop bit omission / recovery where DTE data speed is different from modem speed and specifies two difference ranges, +1% to –2.5% (basic) and +2.3% to –2.5% (extended). The CMX867 enables stop bit deletion by providing two *Tx Data* registers, $E3 for normal transmission and $E4 for transmission with one stop bit removed from the current format. Hence the host can adjust the removal of stop bits according to V.14 by directing bytes to the appropriate *Tx Data* register.
At the receiving end the CMX867 can tolerate missing stop bits if set to basic or extended over-speed mode.
      Rx Mode ($E2)
            b5…b3 = 101 (for basic range)
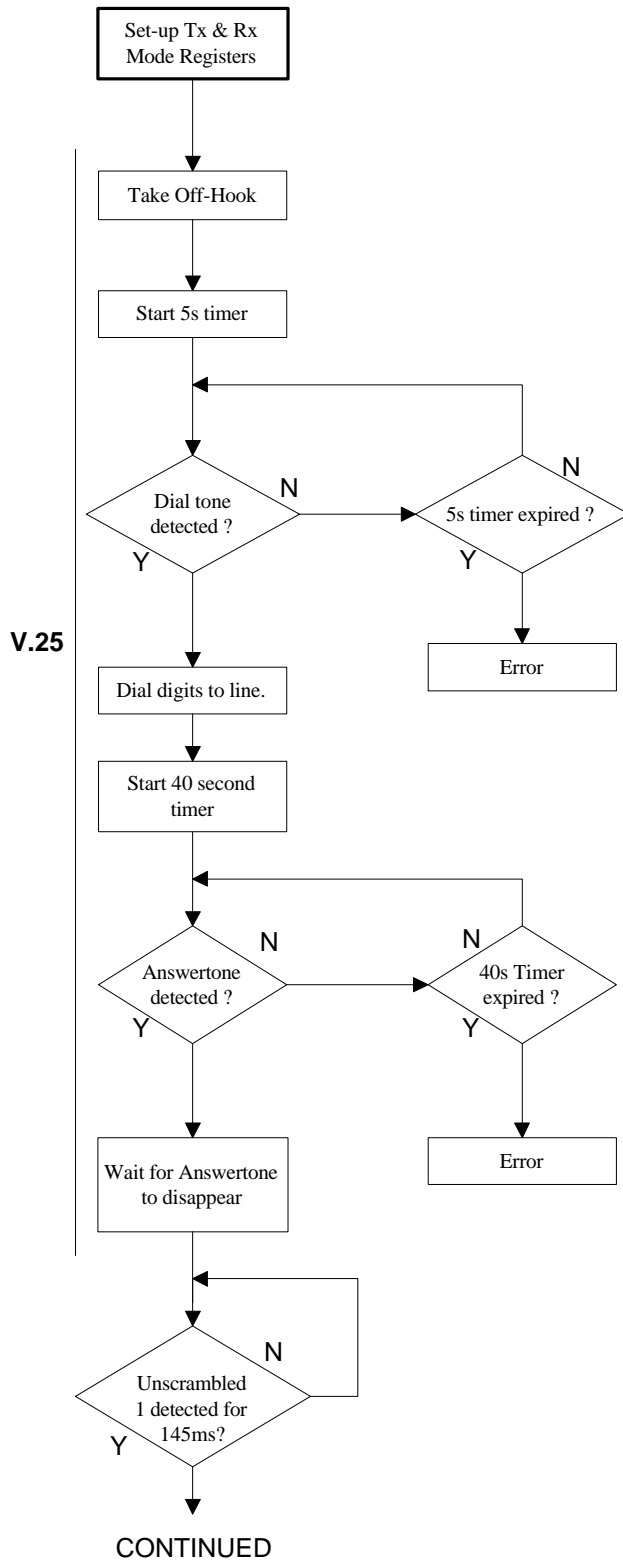   *or*   b5…b3 = 100 (for extended range)
    or   b5…b3 = 110 (for over-speed function disabled)

2.4. The CMX867 passes only 8 bit data over the C-Bus to the remote host controller. The host controller must read the *Status* register ($E6) and check for parity, framing, and overflow errors before reconstituting the character in its original format for onward transmission to the remote DTE.

Figure 2

Calling Modem

V.25

```
Set-up Tx & Rx Mode Registers
        ↓
Take Off-Hook
        ↓
Start 5s timer
        ↓
Dial tone detected ? ──N──→ 5s timer expired ? ──N──┐ (loop back)
        │Y                          │Y
        │                           ↓
        │                        Error
        ↓
Dial digits to line.
        ↓
Start 40 second timer
        ↓
Answertone detected ? ──N──→ 40s Timer expired ? ──N──┐ (loop back)
        │Y                          │Y
        │                           ↓
        │                        Error
        ↓
Wait for Answertone to disappear
        ↓
Unscrambled 1 detected for 145ms? ──N──┐ (loop back)
        │Y
        ↓
    CONTINUED
```

Set-up Registers
Tx Mode ($E1)
    b15..b12 = 0001 (select DTMF / Tones mode)
    b11..b9 = user defined (Tx level)
Rx Mode ($E2)
    b15..b12 = 0001 (select DTMF / Tone mode)
    b11..b9 = user defined (Rx level)

Take Off-Hook
General Control ($E0)
    b9 = 1 (Relay drive pin pulled to Vss)
    b5 = 0 (Mask Ring Detect IRQ)

Note:- If blind dialling then wait 3 seconds before jumping to the section below headed "Dial digits to line (DTMF)".

Start a 5 second timer.

Dial Tone Detect (before 5s timer expires)
At this point only the dial tone (or possibly busy tone) should be heard, (country dependent).
General Control   ($E0)
    b2 = 1 (unmask Call Progress Energy detect IRQ)
Status ($E6)
    On interrupt, monitor b10 for Call Progress Energy. If CP energy detected within 5 secs, check whether dial tone (continuous CP energy) or busy tone (interrupted CP energy) is present.

Dial digits to line (DTMF)
Tone cadence of 100ms on and 100ms off (country dependent) is suggested.
General Control ($E0)
    b2 = 0 (mask Call Progress Energy IRQ)
Tx Mode ($E1)
    b4 = 1 (for DTMF generation)
    b3..b0 = user defined (digits to be dialled)
Tone ON
Tone OFF { b4..b0 = 00000 (select No Tone)

Start 40s timer

Detect Answer Tone (before 40s timer expires)
Rx Mode ($E2)
    b2..b0 = 010 (Answertone detect mode)
General Control   ($E0)
    b1 = 1 (unmask 2100Hz Answertone IRQ)
Status ($E6)
    On interrupt, monitor bit 7 for 2100Hz Answertone.

Wait for Answertone to disappear.
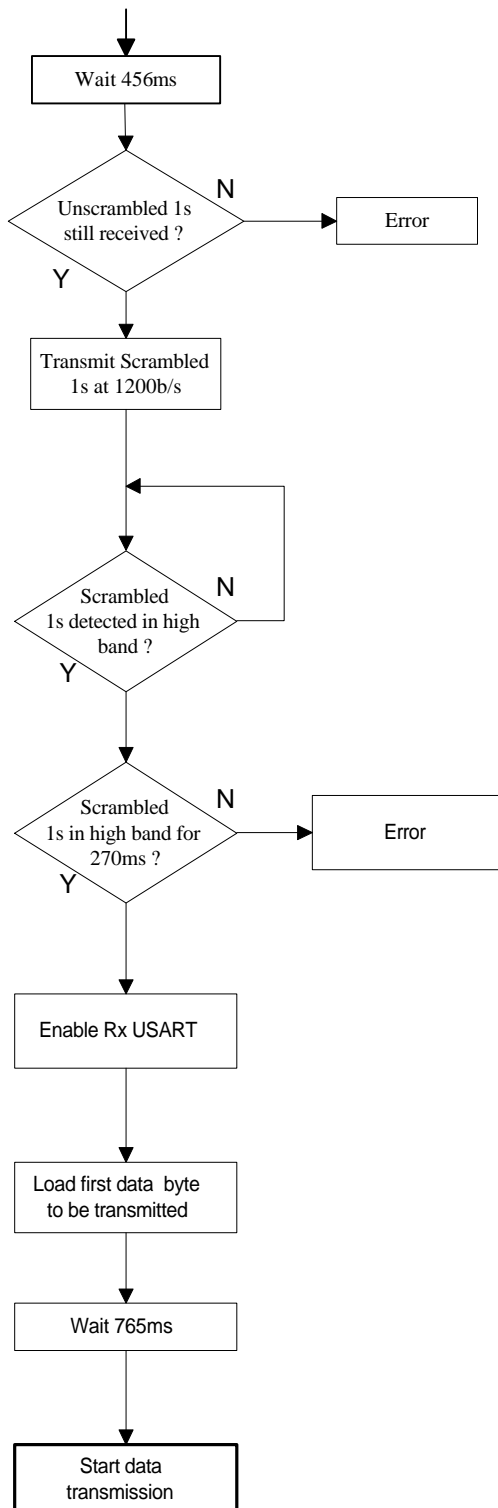
Detect Unscrambled 1s for 145ms
The 2225Hz Answertone detector will also trigger on unscrambled 1s.
General Control   ($E0)
    b1 = 0 (mask 2100Hz Answertone IRQ)
    b0 = 1(unmask 2225Hz Answertone IRQ)
Status ($E6)
    On interrupt, monitor bit 6 for 2225Hz Answertone which shows that unscrambled 1s are being received. Check that unscrambled 1s are received for at least 145ms.

Calling Modem (2)

```
         │
         ▼
┌─────────────────┐
│   Wait 456ms    │
└─────────────────┘
         │
         ▼
    ╱─────────╲           N    ┌──────────┐
   ╱ Unscrambled ╲─────────────▶│  Error   │
   ╲ 1s still received?╱         └──────────┘
    ╲─────────╱
         │ Y
         ▼
┌─────────────────┐
│ Transmit Scrambled │
│  1s at 1200b/s  │
└─────────────────┘
         │
         ▼
    ╱─────────╲
   ╱ Scrambled  ╲    N
   ╱ 1s detected in╲───┐
   ╲ high band?   ╱    │
    ╲─────────╱◀───────┘
         │ Y
         ▼
    ╱─────────╲          N     ┌──────────┐
   ╱ Scrambled  ╲───────────────▶│  Error   │
   ╱ 1s in high band╲            └──────────┘
   ╲ for 270ms?  ╱
    ╲─────────╱
         │ Y
         ▼
┌─────────────────┐
│  Enable Rx USART │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│ Load first data byte │
│  to be transmitted │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│   Wait 765ms    │
└─────────────────┘
         │
         ▼
┌─────────────────┐
│   Start data    │
│  transmission   │
└─────────────────┘
```

Wait for 456ms

Check *Status* ($E6) bit 6 to confirm that unscrambled 1s are still being received.

Transmit Scrambled 1s at 1200b/s
Tx Mode   ($E1)
         b15..b12 = 1100 (select V.22 low band)
         b6,b5 = 11 (scrambler and 64 x 1s detect enabled)
         b4,b3 = 11 (select synchronous mode)
         b2..b0 = 011 (select continous 1s)

Detect scrambled 1s in high band
Rx Mode   ($E2)
         b15..b12 = 1101 (select V.22 high band)
         b7,b6 = 11 (descrambler and 64x1s detect enabled)
General Control   ($E0)
         b1 = 1 (unmask scrambled 1s IRQ)
         b0 = 0 (mask IRQ)
Note that b0 was enabled earlier for 2225Hz Answertone
IRQ, but is disabled here after selecting V.22 high band because no
interrupt is yet required for Rx Data Overflow or Rx Data Ready.

On interrupt, check   *Status* ($E6)
         If b10 = 1 (Rx energy detected)
         b8,b7 = 11 (scrambled 1s found)

Check scrambled 1s are detected for 270ms (nom.) If not then an
error has occured.

Enable Rx USART
NOTE The use of the autoequaliser can improve performance.
Rx Mode   ($E2)
         b8 = user defined (autoequaliser setting)
         b5..b3 = user defined (Rx USART mode)
         b2..b0 = user defined (data bits and parity
                  settings)
General Control   ($E0)
         b0 = 1 (unmask Rx Data overflow and Rx
            Data Ready IRQ)
         b1 = 0 (mask scrambled 1s detection IRQ)

Load Tx Data register ($E3 or $E4) with first byte
Note:- Check that   *Status* ($E6) bit 12 = 1 (Tx Data Ready) before
loading the data.

Wait for 765ms

Start data transmission
Tx Mode   ($E1)
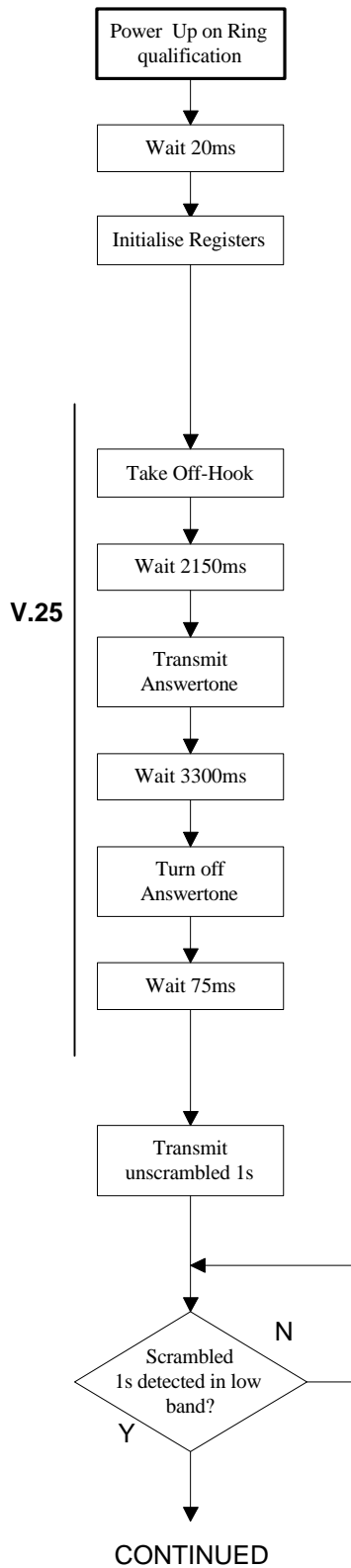   b4,b3 = user defined (Tx USART settings)
         b2..b0 = user defined (data and stop bit settings)
General Control   ($E0)
         b3 = 1 (unmask TX Data underflow and Tx
            Data Ready IRQ)

Figure 3

Answering Modem

```
┌─────────────────────────┐
│  Power Up on Ring        │
│  qualification           │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Wait 20ms               │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Initialise Registers    │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Take Off-Hook           │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Wait 2150ms             │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Transmit                │
│  Answertone              │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Wait 3300ms             │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Turn off                │
│  Answertone              │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Wait 75ms               │
└─────────────────────────┘
            │
            ▼
┌─────────────────────────┐
│  Transmit                │
│  unscrambled 1s          │
└─────────────────────────┘
            │
            ▼
       ◇ Scrambled
       1s detected in low ──N──┐
         band? ◇               │
            │Y                 │
            ▼                  │
       CONTINUED              (loops back)
```

V.25 brackets the steps from Take Off-Hook through Wait 75ms.

On Ring Interrupt
A ring interrupt should wake the host controller to perform a ring qualification routine and power up the CMX868.

Power up
General Control ($E0)
b8 = 1(Device powered normally)
b7 = 1(Reset internal circuitry)

Wait 20ms

Initialise Registers
General Control ($E0)
b7 = 0 (Normal operation)

Tx Mode ($E1)
b15..12 = 0001 (select DTMF/Tones mode)
b11..9 = user defined (Tx level)

Rx Mode ($E2)
b15..12 = 1100 (select V.22 low band)
b11..9 = user defined (Rx level)
b7,b6 = 11 (descrambler enabled)

Take Off-Hook
General Control ($E0)
b9 = 1 (Relay drive pin pulled to Vss)

Wait for 2150ms.

Transmit Answertone
Tx Mode ($E1)
b3..0 = 1010 (2100Hz Answertone)
or   b3..0 = 1011 (2225Hz Answertone)

Wait for 3300ms

Turn off Answertone
Tx Mode ($E1)
b3..0 = 0000 (No Tone)

Wait for 75ms
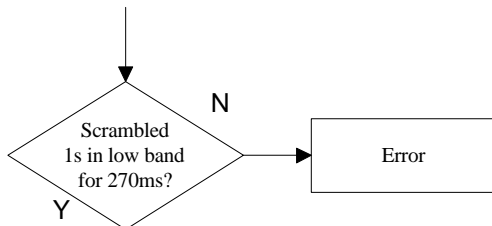
Transmit Unscrambled 1s
Tx Mode ($E1)
b15..12 = 1101 (select V.22 high band)
b8,b7 = user defined (Tx guard tone)
b4,b3 = 11 (select synchronous mode)
b2..b0 = 011 (select continuous 1s)

Detect scrambled 1s
General Control ($E0)
b1 = 1 (unmask scrambled 1s detect IRQ)

On interrupt, check *Status* ($E6)
If b10 = 1 (Rx energy detected)
b8,b7 = 11 (scrambled 1s found)

Answering Modem (2)

```
          ┌─────────────┐
          │  Scrambled  │  N
          │ 1s in low band ├──────────►  ┌──────────┐
          │  for 270ms? │               │  Error   │
          └──────┬──────┘               └──────────┘
               Y │
```

Check that scrambled 1s are detected for 270ms (nom.)
If not then an error has occured.

```
          ┌─────────────────┐
          │ Enable Rx USART │
          └────────┬────────┘
```

Enable Rx USART
NOTE The use of the autoequaliser can improve performance.
Rx Mode ($E2)
        b8 = user defined (autoequaliser setting)
        b5..b3 = user defined (Rx USART mode)
        b2..b0 = user defined (data bits and parity settings)

General Control ($E0)
        b0 = 1 (unmask Rx Data Overflow & Rx Data Ready IRQ)
        b1 = 0 (mask scrambled 1s IRQ)

```
          ┌──────────────────┐
          │ Transmit scrambled │
          │        1s        │
          └────────┬─────────┘
```

Transmit scrambled 1s (for 765ms)
Tx mode ($E1)
        b6,b5 = 11 (scrambler enabled)

```
          ┌──────────────────┐
          │ Load Tx buffer with │
          │  first data byte │
          └────────┬─────────┘
```

Load Tx Data register ($E3 or $E4) with first byte
Note:- Check that *Status* ($E6) bit 12 = 1 (Tx Data Ready) before
loading the data.

```
          ┌──────────────┐
          │  Wait 765ms  │
          └──────┬───────┘
```

Wait 765ms

```
          ┌──────────────┐
          │  Start data  │
          │ transmision  │
          └──────────────┘
```

Start data transmission
Tx Mode ($E1)
        b4,b3 = user defined (Tx USART settings)
        b2..b0 = user defined (data and stop bit settings)

General Control ($E0)
        b3 = 1(Unmask TX Data Underflow and Tx Data Ready IRQ)