DALLAS
SEMICONDUCTOR

Application Note 335
DS2141A, DS2151 Programming,
Controlling the FDL

Note: Contact the factory for C code firmware that implements the FDL requirements as per ANSI T1.403 and AT&T TR54016.

Due to expected future changes in the requirements for the Facility Data Link (FDL), the DS2141A and DS2151 T1 Controller makes use of software in the host to support the communications that occur on the FDL. The use of host's software to help control the FDL allows designs to be easily modified as the standards progress. The DS2141A and DS2151 implements in hardware the more basic functions of controlling the FDL such as byte aligning to the FDL data stream. In current T1 networks, there are two common protocols that exist on the FDL in the Extended Superframe (ESF) framing mode. One is defined in the American National Standards Institute (ANSI) document T1.403–1989 and the other is defined in the AT&T publication TR54016. Depending on the carrier used, either one of these protocols (or both) may be required. This Application Note details how a host would interface with the DS2141A and DS2151 to extract information from the FDL and to insert data into the FDL. The user has the option on the DS2141A and DS2151 to either use its onboard RFDL and TFDL registers or to use the RLINK and TLINK pins to extract/insert data from/into the FDL data stream. This Application Note covers the usage of the RFDL and TFDL registers. The specifics of the two FDL protocols will be not be fully covered. Please refer to the two documents mentioned for more details. (Note: all references in this Application Note to specific data values are in Hexadecimal which is denoted by the value enclosed within < >.)

Receive FDL Control

The DS2141A and DS2151 contains an eight–bit register called the RFDL which is automatically loaded with data from the FDL. There is also a set of two match registers called RFDLM1 and RFDLM2 which can be used relieve the host from constantly polling the RFDL to see if any important data has arrived. If either of the values in the match registers corresponds to the current value in the RFDL, then the host will be alerted. Since most of the communications on the FDL follows the LAPD protocol, the DS2141A and DS2151 also contains a zero destuffer. The zero destuffer should always be enabled when using the RFDL to decode the FDL data stream. Also, when decoding the FDL, the DS2141A and DS2151 should have the ZBTSI (RCR2.6 = 0) and SLC–96 (CCR2.1 = 0) modes disabled.

Figure 1 displays a flowchart of how to decode data off of the FDL. The Match Registers are programmed to respond to the opening addresses of either a Performance Report Message (PRM) in the ANSI T1.403 protocol or a request message in the AT&T TR54016 protocol. The DS2141A and DS2151 will only match on the first byte following an opening flag of <7E> and it will automatically byte align to the incoming LAPD stream. The external controller is normally waiting for a match to occur. Once a match has occurred, the external controller will wait for the RFDL register to fill and then read it. The Abort interrupt (SR2.1) takes on a new function once a match has occurred; it will now also report when the closing flag <7E> is received. Hence the external controller can monitor this flag during the extraction of data from the FDL to determine when all of the data has been captured.

If an abort flag is ever received (eight consecutive ones), then the SR2.1 bit will be set to a one. An Abort flag normally represents the beginning of an Unscheduled (bit–oriented) Message in the ANSI T1.403 protocol. The Unscheduled Message in ANSI T1.403 is a repeating 16–bit pattern of the form "...0CCCCCC011111111..." (the abort flag is transmitted first; the C's represent one of 64 possible code words). The Unscheduled Message will be reported in the RFDL as shown below.

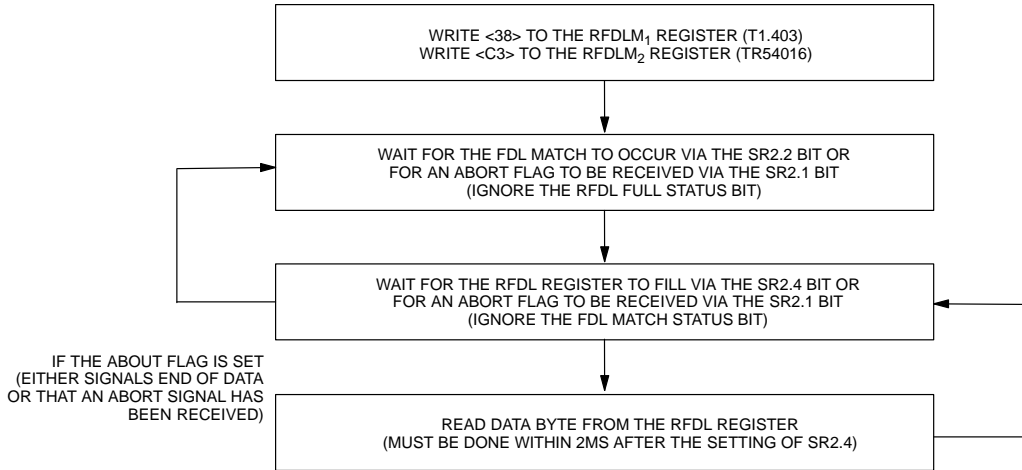|	**(MSB)**							**(LSB)**
RFDL	1	0	C	C	C	C	C	C
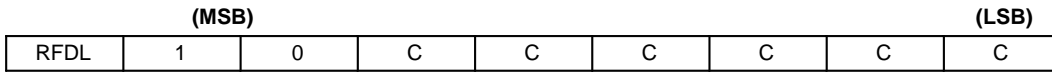
It is possible for an Unscheduled Message to interrupt a PRM or a request message, hence the external controller should monitor the Abort interrupt at all times.

Transmit FDL Control

The DS2141A and DS2151 contains an eight–bit register called the TFDL. Data that is to be transmitted onto the FDL will be loaded into the TFDL register, one byte at a time. Since most communications on the FDL follow the LAPD protocol, the DS2141A and DS2151 contains a zero stuffer to insure that the data between the opening and closing flags does not resemble either an opening or closing flag. The zero stuffer in the DS2141A and DS2151 should only be enabled for the data bytes within the opening and closing flags. If the TFDL register is not updated, it will retransmitted the previous value contained in the register. When using the DS2141A and DS2151 to insert data into the FDL, the device should be set up to source the FDL from the TFDL register (TCR1.2 = 0), and both the ZBTSI (TCR2.5 = 0) and SLC–96

(CCR2.5 = 0) modes should be disabled. Figures 2 and 3 display two different flowcharts for inserting data into the FDL. Figure 2 details how to transmit an Unscheduled (bit–oriented) Message in an ANSI T1.403 environment. The specification calls for Unscheduled Messages to be transmitted at least 10 times (Priority Messages are to be sent for at least one second). Figure 3 details how to transmit either a Performance Report Message (PRM) according to T1.403 or a response message in the AT&T TR54016 protocol. According to ANSI T1.403, PRM's are to be transmitted once a second. The One Second Timer in Status Register 2 can be used to determine when a PRM should be sent. In TR54016, response message are only sent once a request has been received. When the FDL is idle and not sending any PRM's, the flag value of <7E> is to be transmitted.

RECEIVE FDL DECODING Figure 1

WRITE <38> TO THE $RFDLM_1$ REGISTER (T1.403)
WRITE <C3> TO THE $RFDLM_2$ REGISTER (TR54016)

WAIT FOR THE FDL MATCH TO OCCUR VIA THE SR2.2 BIT OR
FOR AN ABORT FLAG TO BE RECEIVED VIA THE SR2.1 BIT
(IGNORE THE RFDL FULL STATUS BIT)

WAIT FOR THE RFDL REGISTER TO FILL VIA THE SR2.4 BIT OR
FOR AN ABORT FLAG TO BE RECEIVED VIA THE SR2.1 BIT
(IGNORE THE FDL MATCH STATUS BIT)

IF THE ABOUT FLAG IS SET
(EITHER SIGNALS END OF DATA
OR THAT AN ABORT SIGNAL HAS
BEEN RECEIVED)

READ DATA BYTE FROM THE RFDL REGISTER
(MUST BE DONE WITHIN 2MS AFTER THE SETTING OF SR2.4)

STATUS BIT(S) USED:
SR2.1 – RECEIVE FDL ABORT
SR2.2 – RECEIVE FDL MATCH OCCURRENCE
SR2.4 – RECEIVE FDL BUFFER FULL
REGISTER(S) USED:
RFDL – RECEIVE FDL REGISTER
RFDLM1 – RECEIVE FDL MATCH REGISTER 1
RFDLM2 – RECEIVE FDL MATCH REGISTER 2

TRANSMIT T1.403 UNSCHEDULED MESSAGE Figure 2

```
┌─────────────────────────────────────────────────────┐
│         WRITE <FF> TO THE TFDL REGISTER              │
│        DISABLE ZERO STUFFER (CCR2.4+ = 0)            │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│            WAIT FOR THE TFDL REGISTER TO             │
│              EMPTY VIA THE SR2.3 BIT                 │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│         WRITE <14> TO THE TFDL REGISTER         (1) │
│   (MUST BE DONE WITHIN 2MS AFTER THE SETTING OF SR2.3) │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│            WAIT FOR THE TFDL REGISTER TO             │
│              EMPTY VIA THE SR2.3 BIT                 │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│         WRITE <FF> TO THE TFDL REGISTER             │
│   (MUST BE DONE WITHIN 2MS AFTER THE SETTING OF SR2.3) │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│              WRITE <7E> TO THE                  (2) │
│                TFDL REGISTER                        │
└─────────────────────────────────────────────────────┘
```

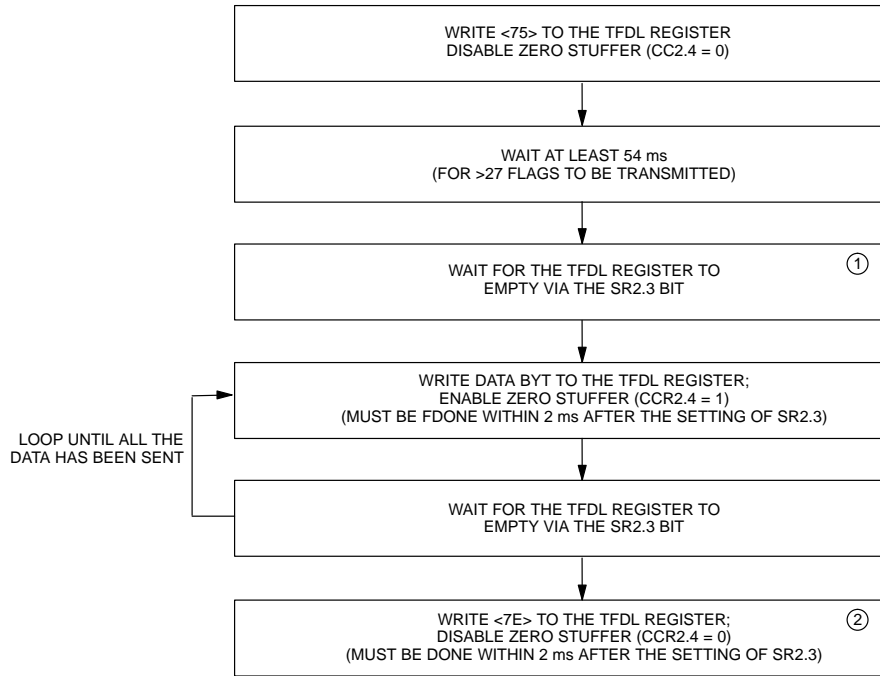LOOP AT LEAST 10 TIMES
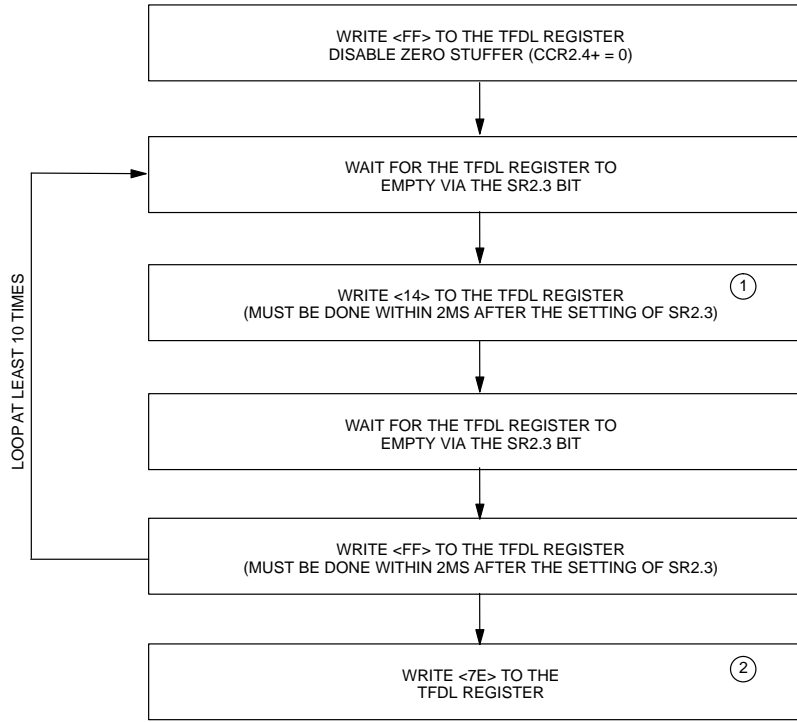
NOTES(S):
1. TRANSMITTING THE UNSCHEDULED MESSAGE FOR PAYLOAD LOOPBACK ACTIVATE IS SHOWN.
2. THIS <7E> WILL BE THE FDL IDLE CODE.

STATUS BIT(S) USED:
SR2.3 – TRANSMIT FDL BUFFER EMPTY
REGISTER(S) USED:
TFDL – TRANSMIT FDL REGISTER
CONTROL BIT(S) USED:
CR2.4 – TRANSMIT ZERO STUFFER ENABLE

TRANSMIT FDL CODING Figure 3

```
┌─────────────────────────────────────────────────────┐
│          WRITE <75> TO THE TFDL REGISTER             │
│          DISABLE ZERO STUFFER (CC2.4 = 0)            │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│              WAIT AT LEAST 54 ms                     │
│         (FOR >27 FLAGS TO BE TRANSMITTED)            │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│          WAIT FOR THE TFDL REGISTER TO        ①      │
│          EMPTY VIA THE SR2.3 BIT                     │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│         WRITE DATA BYT TO THE TFDL REGISTER;         │
│         ENABLE ZERO STUFFER (CCR2.4 = 1)             │
│   (MUST BE FDONE WITHIN 2 ms AFTER THE SETTING OF SR2.3) │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│          WAIT FOR THE TFDL REGISTER TO               │
│          EMPTY VIA THE SR2.3 BIT                     │
└─────────────────────────────────────────────────────┘
                          │
                          ▼
┌─────────────────────────────────────────────────────┐
│         WRITE <7E> TO THE TFDL REGISTER;       ②     │
│         DISABLE ZERO STUFFER (CCR2.4 = 0)            │
│   (MUST BE DONE WITHIN 2 ms AFTER THE SETTING OF SR2.3) │
└─────────────────────────────────────────────────────┘

LOOP UNTIL ALL THE
DATA HAS BEEN SENT
```

NOTES(S):
1. THE CCR2 4 BIT ONLY NEEDS TO BE SET WHEN THE FIRST DATA BYTE IS WRITTEN.
2. THIS <7E> WILL BE THE CLOSING FLAG.

STATUS BIT(S) USED:
SR2.3 – TRANSMIT FDL BUFFER EMPTY
REGISTER(S) USED:
TFDL – TRANSMIT FDL REGISTER
CONTROL BIT(S) USED:
CR2.4 – TRANSMIT ZERO STUFFER ENABLE