N8970UGS4A

Bt8970EVM

Evaluation Module
Software Manual

Rockwell Semiconductor Systems

Print date: February 1998

PRINTED IN THE UNITED STATES OF AMERICA

Table of Contents

List of Figures

List of Tables

1.0 Description

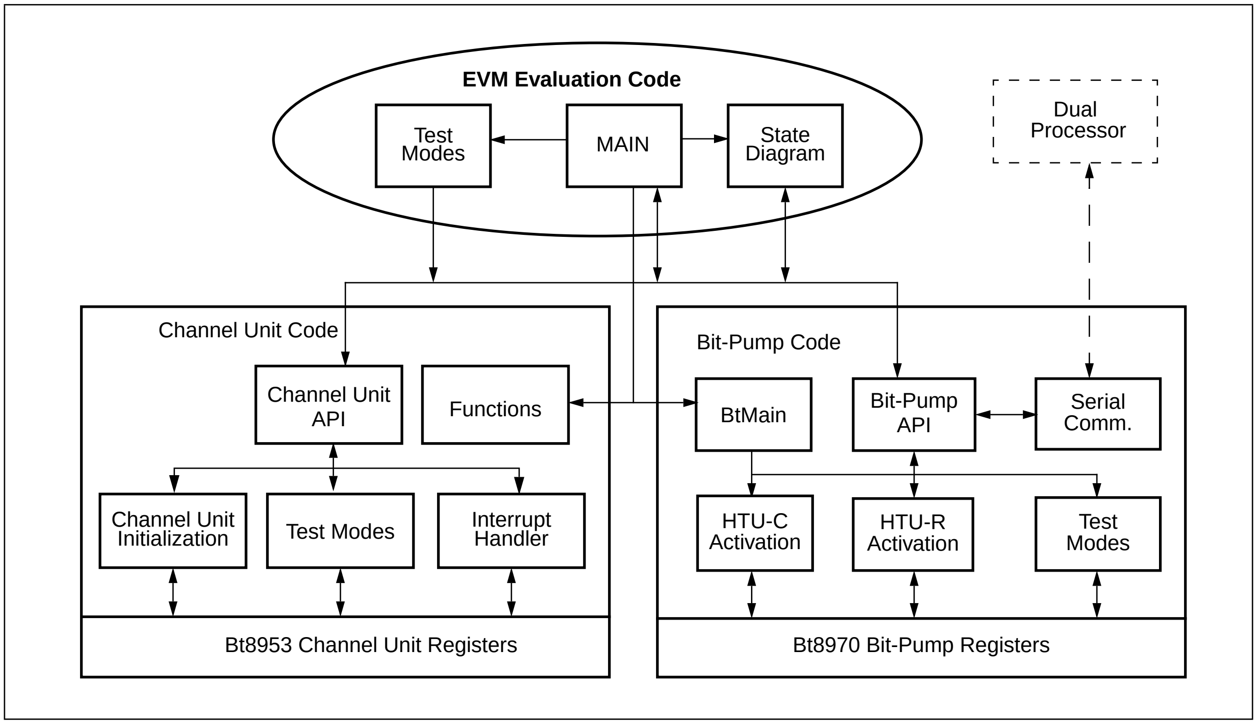
1.1 Introduction

This document describes the EVM application code examples distributed with the Bt8970EVM. The EVM application code, along with the Bt8970 bit-pump code and Bt8953 channel unit code, provide a complete Bt8970 application. Please see the *Bt8970 Software Users Guide* for complete details on the bit-pump code.

Three separate examples are provided in this document: single processor, dual processor, and Echo Return Loss Enhancement (ERLE). For the single processor configuration, an Activation State Diagram is implemented based on the ETSI and ANSI HDSL specifications. For the dual processor configuration, the minimum set of code required to initialize and handle the bit-pump and serial interface commands is provided. The ERLE configuration is a diagnostic tool that verifies the integrity of the hardware's analog front-end.

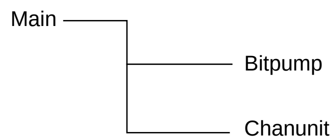
Figure 1-1 illustrates the major software block of the EVM single processor application code example in relation to the Bt8970 bit-pump code.

Figure 1-1. Example of EVM Single Processor Application Code



1.2 Directory Structure

The directory structure is partitioned into three directories: the main (application-specific) directory as the root, the bit-pump code as a subdirectory labeled BITPUMP, and the channel unit code as a subdirectory labeled CHANUNIT.



The application examples, hex files, build script files, etc., are located in the main directory. All of the bit-pump source code and header files are in the BITPUMP subdirectory. All of the channel unit source code and header files are in the CHANUNIT subdirectory.

NOTE: It is recommended that the application have separate sub-directories for the bit-pump and channel unit code. In this way, it is easier to manage the software and handle future upgrades. However, the final implementation of the directory and file structure is up to the user.

1.3 File Names and Dependencies

Table 1-1 lists the application files that can be compiled and linked in with the Bt8970 bit-pump and Bt8953 channel unit code to generate an executable application code.

Table 1-1. File Names and Dependencies

Application Example	Module	File Depends on...
Single Processor	bt8970.c, timer.c	bt8970.h,timer.h, bitpump and channel unit header files
ERLE	erlemain.c	bitpump and channel unit header files
Dual Processor	dualproc.c	btmain.h, init51.h, intbug.h

1.4 Compiling/Linking the Application Examples

The SCRIPT.BLD file in the root directory contains information to generate the different application examples. The SCRIPT.BLD file is an Intersolv (R) Configuration Builder 5.0 script file (makefile).The HEX files listed in Table 1-2 can be built.

NOTE: It is assumed the user knows how to compile and link the source code with the necessary compiler/linker flags.

Table 1-2. HEX Files

Application Example	HEX Files
Single Processor	bt8970cr.hex, tdebug.hex, bt8970c.hex, and bt8970r.hex
ERLE	erle.hex
Dual Processor	dualproc.hex
Note: bt8970cr.hex, tdebug.hex, and erle.hex are sent along with the distribution disk. The bt8970c.hex and bt8970r.hex are similar to the bt8970cr.hex except they only generate code for the HTU-C or HTU-R, respectively. The tdebug.hex is the same as the bt8970cr.hex except it outputs 'printf()' characters through the serial port; the UIP will not work.	

1.4.1 Linker Flags

The LFLAGS variable specifies the linker flags. All of the HEX files use the same linker flags:

```
XDATA(8000) RAM(256)
```

XDATA(8000) specifies the base address location of the external RAM memory mapping, which is at address 0x8000 on the Bt8952-001 motherboard. RAM(256) specifies that the microprocessor contains 256 bytes of internal RAM.

1.4.2 Compiler Flags

The CFLAGS variable specifies the compiler flags. The compiler flags are unique to each application. There are two aspects to the CFLAGS list: 8032 Compiler options and Bt8970 Bit-pump Compiler Switches [denoted by the DF(...)]. The definitions for each bit-pump switch are defined in the Bt8970 Software User's Guide.

The CHANNEL_UNIT flag found in SCRIPT.BLD determines if the channel unit code is compiled into the application. The default is set to 1; however, set this flag to 0 to not compile in the channel unit code.

1.5 Software Development Tools

1.5.1 Compilers

C51	Keil Elektronik GmbH	Phone: ++4989/456040-0 Fax: (214) 735-8055 www.keil.com
C51	Franklin Software, Inc.	Phone: (800) 348-8051

1.5.2 In-Circuit Emulators

EMUL51	Nohau Corporation	(408) 866-1820 www.nohau.com

1.5.3 In-Circuit Debuggers

ChipView	Chip Tools Corporation	Phone: (905) 274-6244 Fax: (905) 891-2715 www.chiptools.com

1.5.4 Make Utility

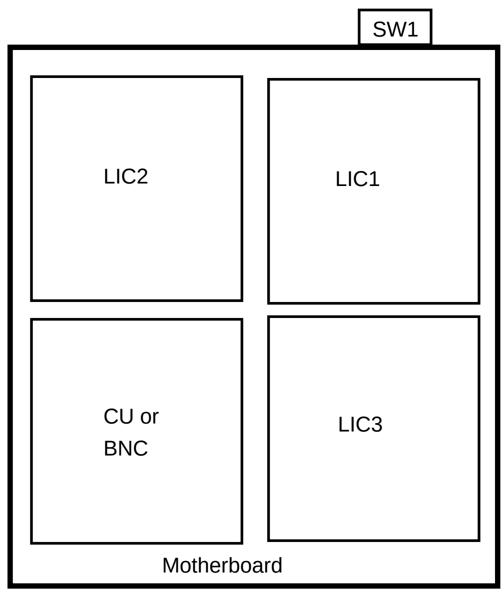
Configuration Builder	Intersolv	(800) 547-4000 www.intersolv.com

1.6 BT8970 EVM Hardware

The modular BT8970 EVM Hardware is designed to accommodate the standard 2T1, 2E1, 3E1, and single-pair (bit-pump with BNC card) configurations. An BT8970 HDSL System is comprised of two BT8970 Terminal Units (HTUs), a Central Office (HTU-C), and a Remote Terminal (HTU-R). In the ETSI standards, HTU-C is synonymous with LTU (Line Terminal Unit) and HTU-R with NTU (Network Terminal Unit). As shown in Figure 1-2, each HTU is comprised of the following:

- Motherboard
- 1, 2, or 3 Line Interface Cards (LIC)
- Channel Unit Card (CU) or BNC Connector Card

Figure 1-2. Bt8970EVM Terminal Unit



1.6.1 Channel Unit Card Present

When the Channel Unit is present, the Line Interface Card(s) can be placed in any of the LIC slots on the motherboard. The Channel Unit EVM board must always be placed in the CU Slot on the motherboard.

When there are less than three Line Interface Cards, i.e. 2T1 or 2E1; the Line Interface Cards can be placed in any two of the three LIC Slots on the motherboard. The software automatically determines which slot(s) the Line Interface Card(s) are located in and only programs the corresponding Channel Unit Loop specific registers. The program then follows the Application Example 1A: HDSL Activation State Diagram.

For example, in a 2E1 system, the 2 Line Interface Cards can be placed in motherboard slots #2 and #3. The Channel Unit would then program the Transmit & Receive Loops #2 and #3, Route/Combine Tables, and other appropriate registers to interface with the Line Interface Cards found in slots #2 and slot #3. The Master Loop would be assigned to the Line Interface Card found in slot #2, since it was the first Line Interface Card found. It is quite possible to have the HTU-C Line Interface Cards found in slots #2 & #3 and to have the HTU-R Line Interface Cards found in slots #1 and #3. However, it is common practice to place the Line Interface Cards in sequential order on the motherboard slots.

The reasons for allowing the Line Interface Cards to be located in any slot are two-fold. First, in case a motherboard slot becomes non-functional, the other two slots could be used. Second, it demonstrates the flexibility in the Channel Unit programming capabilities.

1.6.1.1 Determine Channel Unit Configuration

The _InitChannelUnitEvmBoard() function determines how to configure the Bitpump and Channel Unit. The Framer on the Channel Unit EVM Board is queried to determine its type. Currently only the Bt8510 (E1) and Bt8360 (T1) framers are shipped on the various Channel Unit EVM Boards. In addition, the number of bitpumps present are used to determine how to configure the EVM. Table 1-3 outlines how the different configurations are determined.

Table 1-3. EVM Configurations

# of Bitpumps	Framer Type	Channel Unit Configuration	Bitpump Symbol Rate
1	don't care	2 pair E1	584 k
2	Bt8360	2 pair T1	392 k
2	Bt8510	2 pair E1	584 k
3	Bt8510	3 pair E1	392 k

NOTE: If an invalid configuration is detected, the Channel Unit Error LED is illuminated, and the Channel Unit is no longer supported. The bit-pump configuration is then determined by the SW1 DIP switch on the motherboard (please see Section 1.7) and the program will then follow the Application Example 1B: Bit-pump Activation State Diagram (see Section 1.11).

1.6.2 BNC Connector Card Present

When the BNC Connector Card is present, the Line Interface Card(s) can be placed in any motherboard slots. The external clock and data must then be connected to the appropriate BNC group on the BNC connector card. The program then follows the Application Example 1B: Bit-pump Only Activation State Diagram (see Section 1.11 on page 26).

1.7 DIP Switch

DIP Switch (SW1) on the Bt8952-001 motherboard determines the program flow. The SW1 switch is accessed using the P1 port. The bt8970cr.hex, tdebug.hex, bt8970c.hex, and bt8970r.hex application have a different meaning than the erle.hex. The SW1 switch is not used by the dualproc.hex application. The DIP switch definitions for the bt8970cr.hex, tdebug.hex, bt8970c.hex, and bt8970r.hex application (Activation State Machine examples) are shown below in Figure 1-3. The DIP switch definitions for the ERLE application are shown in the ERLE example.

NOTE: The DIP Switch numbers on the motherboard are transposed as compared to the P1 port, that is, DipSw#1 is connected to P1#7, DipSw#2 is connected to P1#6, etc. The Dip Switch is labeled 1–8 while the P1 port is 0–7.

1.7.1 bt8970cr.hex, tdebug.hex, bt8970c.hex, and bt8970r.hex

DIP switches are used to configure the mode of operation. The modes are outlined in Figure 1-3 with switch 1 being on the left as the user faces the DIP switches.

NOTE: If an HTU-C or HTU-R only version of code is compiled, switch 1 has no function.

Figure 1-3. DIP Switch Mode Configuration

Evaluation Motherboard **SW1 (DIP Switches #1–8)**

8	7	6	5	4	3	2	1

Operating Mode	**#8**
Stand Alone Mode	0
UIP Mode	1

#7–1: Only valid in Stand Alone Mode

Data Rate (Supported Ranges)	**#6**	**#7**
1168 kbps (968 to 1368 kbps)	0	0
784 kbps (656 to 968 kbps)	0	1
528 kbps (160 to 656 kbps)	1	0
1552 kbps (above 1368 kbps)	1	1

Test Mode	**#3**	**#4**	**#5**
Normal Operation	0	0	0
BP Isolated +3	0	0	1
BP Continuous 4 Level	0	1	0
BP External Analog Loopback	0	1	1
BP Digital Loopback	1	0	0
Channel Unit PRA to PRA Loopback[(1)]	1	0	1
LIU Line Loopback[(1)]	1	1	0
Normal Operation–Unframed[(1)]	1	1	1

Reserved (should be kept in the 0 [down] position)	**#2**

Terminal Type	**#1**
Local (HTU-C)	0
Remote (HTU-R)	1

Notes: (1). Requires Channel Unit/Framer Card
2. 0 indicates the switch is down, 1 indicates the switch is up.

1.8 Application Code

The Application Code can operate in two modes:

- PC Control
- Reset

The PC Control Mode is also referred to Dual-Processor or UIP (User Interface Program) mode. The Reset Mode is also referred to Single Processor or Stand-alone mode.

1.8.1 PC Control Mode

When operating in PC Control mode, the Application Code performs the following:

- Initializes the Bitpump
- Initializes the Intel 8051 Microprocessor
- Initializes the Channel Unit Addresses
- Determines the locations of the Line Interface Cards
- Calls BtMain() and _CuHandleFlags() forever

The location of the Line Interface Cards is needed for the Channel Unit software since there is no Channel Unit API command defined for this function. The Bitpump and Channel Unit are then configured and controlled using the Serial Interface API commands, whether it is done from the User Interface Program or a customer's own diagnostic software tool.

1.8.2 Reset Mode

When operating in Reset mode, the software queries the hardware to determine the proper configuration. The Application Code will behave differently depending on whether or not the Channel Unit or BNC card is present, as described in Example 1A and 1B (see Section 1.10 and Section 1.11).

1.9 General Purpose Timer

A general purpose timer is implemented using the 80C32 timer #0 (timer #1 is used as a baud rate generator). The timer structure is implemented as a two-dimensional array to index multiple bit-pumps and general purpose timers. Currently, only one general purpose timer per bit-pump is used for LOSWT (loss of sync word timer).

```
/* Found in TIMER.H */
/* Timer indexes */
#define _NO_GEN_PURPOSE_TIMERS      1
#define PENDING_DEACTIVATE_TIMER    0

/* Timer Structures */
typedef union
{
    unsigned char reg;
    struct
        {
        unsigned char state:1;
        unsigned char complete:1;
        unsigned char reserved:6;
        } bits;
} gen_purpose_timer_status;

typedef struct
{
    unsigned int counter_value;
    gen_purpose_timer_status status;
} GEN_PURPOSE_TIMER;

/* Found in TIMER.C */
static GEN_PURPOSE_TIMER gen_timer[_NO_OF_LOOPS][_NO_GEN_PURPOSE_TIMERS];
```

NOTE: To add more timers, increase the _NO_GEN_PURPOSE_TIMERS definition, and add the appropriate timer index definitions.

NOTE: The general purpose times functions are found in TIMER.C.

Table 1-4. General Purpose Timer Functions and Descriptions

Functions	Description
_InitGenPurposeTimer()	Initializes the general purpose timer. Only needs to be called once at the beginning of the program.
_LoadGenPurposeTimerInterval()	Loads the Timer #0 counter for a 50.0-ms period.
_EnableGenPurposeTimer()	Enables the Interrupt Handler and loads the specified general purpose counter. To specify a 2-second timer, load the general purpose timer to 40 (40 * 50 ms = 2 sec)
_DisableGeneralPurposeTimer()	Disables the specified general purpose timer. If no other general purpose timers are active, the Interrupt Handler is disabled.
_GetGenPurposeTimerStatus()	Returns the status of the specified general purpose timer. See the *gen_purpose_timer_status* structure for status indicator bits (Table 1-5).
_Timer0_ISR()	Interrupt handler that decrements any active general purpose timer. If the counter values reaches 0, the corresponding *complete* bit is set.

1.9.1 Functions: Parameter & Return Value Descriptions

void _EnableGenPurposeTimer (BP_U_8BIT bp, BP_U_8BIT timer, BP_U_16BIT value)

Parameters:

- *bp:* which bit-pump; _BIT_PUMP0 - _NO_OF_LOOPS
- *timer:* which timer; currently only accepts _PENDING_DEACTIVATE_TIMER
- *value:* number of 50mS intervals to count.

Return Value: none

void _DisableGenPurposeTimer (BP_U_8BIT bp, BP_U_8BIT timer)

Parameters:

- *bp:* which bit-pump; _BIT_PUMP0 - _NO_OF_LOOPS
- *timer:* which timer; currently only accepts _PENDING_DEACTIVATE_TIMER

Return Value: none

BP_U_8BIT _GetGenPurposeTimerStatus (BP_U_8BIT bp, BP_U_8BIT timer)

Parameters:

- *bp:* which bit-pump; _BIT_PUMP0 - _NO_OF_LOOPS
- *timer:* which timer; currently only accepts _PENDING_DEACTIVATE_TIMER

Return Value: See table below

Table 1-5. General Purpose Status Bit Definitions

Bit	Description	0 Value	1 Value
0	State	Disabled	Enabled
1	Complete	Not Complete	Complete
2—7	Reserved		

1.10 Application Example 1A: HDSL Activation State Diagram

This section describes how the Activation State Diagram is implemented using the Bt8970 bit-pump and Bt8953 channel unit code. The Activation State Diagram is based on the ETSI ETR-152 and ANSI T1E1.4 HDSL specifications. Refer to the *bt8970.c* module for details.

For each state, the responsibilities of the application code vs. the Bt8970 bit-pump code vs. the Bt8953 channel unit code are distinguished. The application code accesses the Bt8970 bit-pump code through the *_BtControl()* and *_BtStatus()* function calls. The application code accesses the Bt8953 channel unit code through the *_BtControl()* and *_BtStatus()* function calls (which call the *_CuControl()* and *_CuStatus()* functions) as well as some direct function calls.

The application code only needs to issue three API commands to implement the Bt8970 portion of the Activation State Diagram (this is assuming the bit-pump has been properly configured using the appropriate API commands). By repeatedly calling the *_BtMain*() function, the Bt8970 bit-pump code will automatically progress through the bit-pump activation state diagram. The bit-pump API commands and their description are as follows:

- _STARTUP_STATUS—Monitors the bit-pump activation status, which allows the application code to determine what state the bit-pump is in and to detect any error conditions.
- _ACTIVATE—Initiates the bit-pump activation.
- _DEACTIVATE—Deactivates the bit-pump.

The application code implements the Channel Unit portion of the Activation State Diagram by issuing the following API commands and functions: API commands are all capitals, and functions are denoted by the parentheses ().

- _CU_SYNC—Monitors the channel unit sync status.
- _CU_FORCE_SCR_ONES — Configures the channel unit to transmit Framed Scrambled Ones; both the overhead and payload are set to all 1’s.
- _CuConfigureTransmitS1() — Configures the channel unit for S1 data; overhead enabled but the payload data is still all 1’s.
- _CuSetRtrInd() — Set the RTR (ready to receive) indicator bit.
- _CuForceOnes() — Enables the payload data (by disabling the force payload 1’s).
- _CU_TRANSMIT_PAYLOAD—Complete channel unit configuration for normal operation.

NOTE: The Bt8970 + Bt8953A chip set can be compliant to the HDSL standards activation state machine. The bit-pump is responsible for the transceiver functionality, and the channel unit is responsible for the framing functionality. The customer would not need to modify the bit-pump startup algorithms to be compliant (the only changes to the bit-pump would be at the API layer). The customer would be responsible for modifying the channel unit to be compliant. The channel unit EVM code demonstrates proper mapping and initialization; however, there are such issues as EOC, loop reversal, and switching master loop that are not complete. In addition, the customer is ultimately responsible for the high level application code to fit the final application.

1.10.1 Initialization State

The initialization state applies to both the HTU-C and HTU-R.

1.10.1.1 Initialization

At power ON, the software is initialized by calling the following routines:

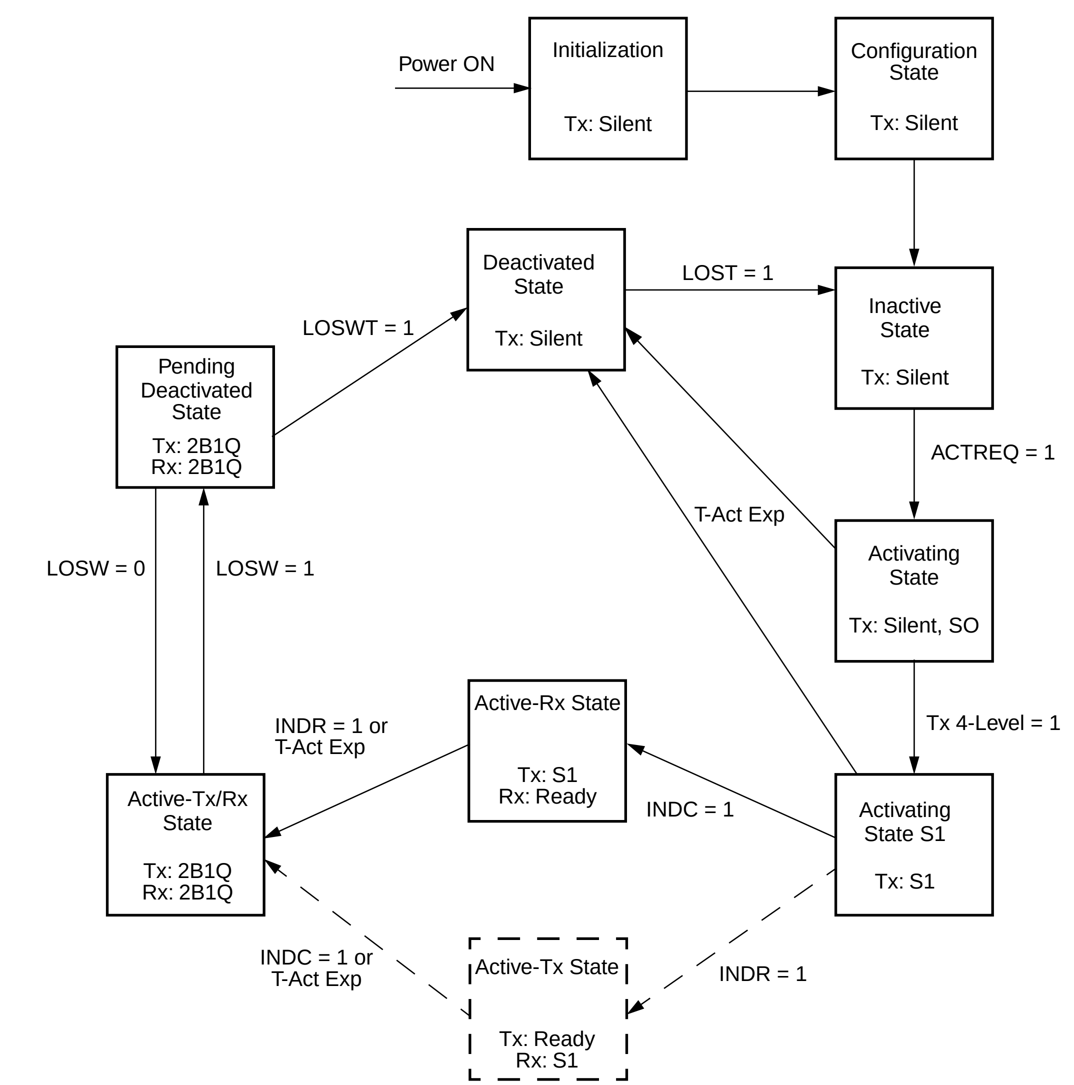
- _BtSwPowerUp()
- _MaskBtHomerInt()
- _Init8051() – Assuming Intel 8051 Family processor
- _InitGenPurposeTimers()
- _CuInitAddress()

NOTE: The activation state is then set to the inactive state after the initialization state according to the HDSL standards.

1.10.2 HTU-C (LTU) Activation

HTU-C and LTU are synonymous. Figure 1-4 shows the state diagram for the activation of the HTU-C.

Figure 1-4. HTU-C Activation State Diagram



Note: ACTREQ = Activation Request (application code)
T-ACT Exp = Activation timer expired (bit-pump code)
Tx 4-Level = Bit-pump is transmitting a 4-level signal (bit-pump code)
INDC = HTU-C Indicator (channel unit code)
INDR = HTU-R Indicator (channel unit code)
LOSW - Loss of Sync Word (channel unit code)
LOSWT - Loss of Sync Word Timer (application code)
LOST - Loss of Signal Timer (bit-pump code)

1.10.3 Configuration State

Application Code: In the configuration state, the bit-pump is configured by issuing the API commands listed in Table 1-6. The channel unit is configured by issuing the API commands listed in Table 1-7.

NOTE: The bit-pump on/off and symbol rate commands must be called first and second respectively. The rest of the API commands can be called in any order.

Table 1-6. HTU-C Configuration Commands, Bit-Pump Configuration

Command Name	Parameter	Remarks
Bit-pump ON/OFF	_PRESENT	Turn bit-pump #0 ON
Symbol Rate	Symbol rate is set according to SW1 (see Figure 1-3)	API value = (Symbol Rate / 4096) i.e., 96 = 392 k / 4096 (784 kbps data rate)
Terminal Type	_HTUC	Central Office Terminal
Analog AGC Configuration	_SIX_LEVEL_AGC	6-level AGC is implemented
Start-up Sequence Source	_EXTERNAL	Use externally generated scrambled data from channel unit during activation
Transmit Scrambler	_BYPASS	Bypass bit-pump scrambler(1)
Receive Descrambler	_BYPASS	Bypass bit-pump descrambler(1)
Framer Format	_SERIAL_SWAP	Sign bit followed by Magnitude bit
Other Side Bt8970	_NO_BT	Other terminal is unknown
LOST Time Period	10	Set LOST = 1 second
Note: (1). Channel Unit provides necessary scrambled data.		

Table 1-7. HTU-C Configuration Commands, Channel Unit Configuration

Command Name	Parameter	Remarks
Terminal Type	_HTUC	Central Office Terminal
Configure Channel Unit	_2T1, _2E1, or _3E1.	Configuration is determined by querying the number of bit-pumps and the framer type.
Configure Framer	_BT8360 (T1) or _BT8510 (E1)	Configuration is determined by querying the framer type.
Configure LIU	_ELS_E1_PCM30 or _ELS_T1_0_110	Configuration is determined by querying the framer type.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump code processes the API commands.

Bt8953 Channel Unit Code: The Bt8953 channel unit code processes the API commands. The _CU_CONFIGURE initializes all the channel unit registers for the specified configuration. The _FRAMER_CONFIGURE and _LIU_CONFIGURE initialize the framer and LIU for transparent operation.

1.10.4 Deactivated State

Application Code: In the deactivated state, the application code issues the bit-pump _DEACTIVATE API command (this will turn off the bit-pump transmitter) and channel unit _CuConfigureBeginStartup() function (this will mask the Tx/Rx 6mS interrupts). The application code will then wait for LOST = 1. When LOST = 1 becomes true, the activation state is then changed to the inactive state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump interrupt handler starts the LOS Timer (LOST) when the _DEACTIVATE API command is received and when the LOS flag is TRUE (LOS = 1). The Bt8970 bit-pump interrupt handler will set the LOST flag (LOST = 1) when the LOST expires.

Bt8953 Channel Unit Code: The Bt8953 Channel Unit performs the same tasks as before it entered this state.

1.10.5 Inactive State

Application Code: In the inactive state, the _ACTIVATE and _CU_FORCE_SCR_ONES API commands are issued to initiate the activation process. The transmitter is initially silent. The activation request signal is always considered to be true (ACTREQ = 1). The activation state is then changed to the activating state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine does nothing during the inactive state. The bit-pump just waits for the _ACTIVATE API command. When _ACTIVATE is detected, the activation state machine will initialize the bit-pump for the activation process.

Bt8953 Channel Unit Code: The Bt8953 channel unit code waits for the _CU_FORCE_SCR_ONES API command. When the command is received, the channel unit configures the HDSL transmitter to Framed Scrambled Ones; where both the overhead and payload data are all ones. The Tx/Rx 6 ms and interrupts are masked and if the master loop, the DPLL is set to Open Mode.[See *_CuConfigureBeginStartup()*]

1.10.6 Activating State

Application Code: In the activating state, the application code monitors the bit-pump activation status. If the bit-pump T-Act timer expires, then the activation state is changed to the deactivated state. If the bit-pump Tx 4-Level flag is detected, the channel unit will enable the S1 signal (enable overhead). The activation state is changed to the activating S1 state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump code starts transmitting the S0 signal and monitors the received signal for S0 (LOS = 0). When the signal is detected, the bit-pump will perform the optimize phase search, adapt filters, etc. The bit-pump code will then start transmitting the S1 signal.

Bt8953 Channel Unit Code: When the Tx 4-Level is enabled, the channel unit enables the Tx/Rx 6 ms interrupts and the overhead data.[See *_CuConfigureTransmitS1()*]

1.10.7 Activating State S1

This state is not part of the ETR-152 HDSL standards, but is required for the implementation of the bit-pump and channel unit devices.

Application Code: In the activating state, the application code monitors the bit-pump activation status and channel unit status bits. If the bit-pump T-Act timer expires, the activation state is changed to the deactivated state.

If the channel unit INDC (valid Sync Word) flag is detected, the RTR indicator bit is set; the activation state is changed to the active Rx state.

If the channel unit RTR flag is detected, the payload data is enabled; the activation state is changed to the active Tx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump is transmitting the S1 and monitors the received signal for S1. When the S1 signal is detected, the bit-pump code will perform some final adaptation.

Bt8953 Channel Unit Code: The Bt8953 channel unit code monitors the sync word and indicator bits. When the Sync Word is detected, the channel unit sync status is set to CU_IN_SYNC. If the RTR indicator bit is detected, then the INDR bit is set. After a status is detected, then the channel unit code will set the appropriate configuration bits as determined by the application code.

1.10.8 Active Rx State

Application Code: In the active Rx state, the application code monitors the bit-pump activation and channel unit status. If the bit-pump T-Act timer expires or the channel unit INDR bit is detected, then the payload data is enabled. The activation state is changed to the active Tx/Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will finalize its activation process.

Bt8953 Channel Unit Code: The Bt8953 channel unit code monitors the indicator bits. If the RTR indicator bit is detected, then the INDR bit is set. When the enable payload command is issued, the following occurs: the payload data is enabled, the pair ID (ETSI) is set, the Tx/Rx FIFOs are reset, and if the master loop, the DPLL is closed and the DPLL interrupt is enabled. [See *_CuConfigureStartupComplete()*]

1.10.9 Active Tx State

The active Tx state path is typically not taken; therefore it is shown as dashed lines.

Application Code: In the active Tx state, the application code monitors the bit-pump activation and channel unit status. If the bit-pump T-Act timer expires or the channel unit INDC bit is detected, t the payload data is enabled. The activation state is changed to the active Tx/Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will finalize its activation process.

Bt8953 Channel Unit Code: The Bt8953 channel unit code monitors the sync status. If the In Sync bit is detected, then the INDC bit is set. When the enable payload command is issued the following occurs: the payload data is enabled, the pair ID (ETSI) is set, the Tx/Rx FIFOs are reset, and if the master loop, the DPLL is closed and the DPLL interrupt is enabled. [See *_CuConfigureStartupComplete()*]

1.10.10 Active Tx/Rx State

Application Code: In the active-Tx/Rx state, the application code monitors the channel unit sync status. If the LOSW = 1 flag is detected, the activation state is changed to the pending deactivated state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump handles any temperature/environmental changes. Different status responses are continuously monitored and can be probed by issuing the corresponding status request commands.

Bt8953 Channel Unit Code: The Bt8953 channel unit code processes the Tx/Rx 6 ms interrupts. The appropriate status bits are set according to the status registers.

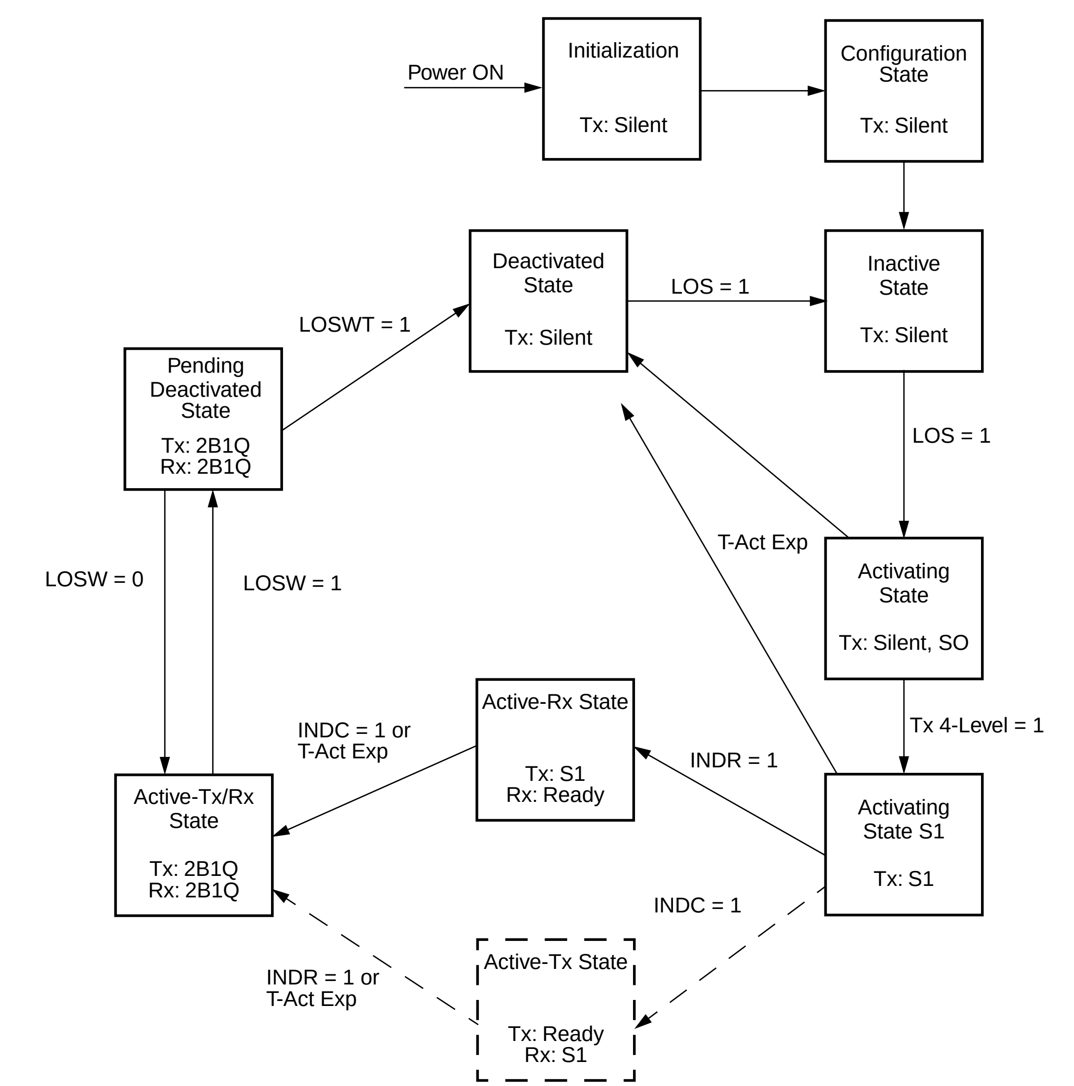
1.10.11 Pending Deactivated-State

Application Code:	In the pending deactivated state, the application code monitors the channel unit status. If the LOSW = 0 (return of sync word) flag becomes valid again within 2 seconds, then the activation state is changed back to the active Tx/Rx state. If the LOSWT = 1 (loss of sync word timer) is detected, then the activation state is changed to the deactivated state.
Bt8970 Bit-Pump Code:	The Bt8970 bit-pump activation state machine behaves similarly to the active Tx/Rx state.
Bt8953 Channel Unit Code:	The Bt8953 channel unit code behaves similarly to the active Tx/Rx state.

1.10.12 HTU-R (NTU) Activation

HTU-R and NTU are synonymous. Figure 1-5 shows the state diagram for the activation of the HTU-R.

Figure 1-5. HTU-R Activation State Diagram



Note: ACTREQ = Activation Request (application code)
T-ACT Exp = Activation timer expired (bit-pump code)
Tx 4-Level = Bit-pump is transmitting a 4-level signal (bit-pump code)
INDC = HTU-C Indicator (channel unit code)
INDR = HTU-R Indicator (channel unit code)
LOSW - Loss of Sync Word (channel unit code)
LOSWT - Loss of Sync Word Timer (application code)
LOST - Loss of Signal Timer (bit-pump code)

1.10.13 Configuration State

Application Code: In the configuration state, the bit-pump is configured by issuing the API commands listed in Table 1-8. The channel unit is configured by issuing the API commands listed in Table 1-9.

NOTE: The bit-pump on/off and symbol rate commands must be called first and second, respectively. The rest of the API commands can be called in any order.

Table 1-8. HTU-R Configuration Commands, Bit-Pump Configuration

Command Name	Parameter	Remarks
Bit-pump ON/OFF	_PRESENT	Turn bit-pump #0 ON
Symbol Rate	Symbol rate is set according to SW1 (see Figure 1-3)	API value = (Symbol Rate / 4096) i.e., 96 = 392k / 4096 (784 kbps data rate)
Terminal Type	_HTUR	Remote Terminal
Analog AGC Configuration	_SIX_LEVEL_AGC	6-level AGC is implemented
Start-up Sequence Source	_EXTERNAL	Use externally generated scrambled data from channel unit during activation
Transmit Scrambler	_BYPASS	Bypass bit-pump scrambler (1)
Receive Descrambler	_BYPASS	Bypass bit-pump descrambler (1)
Framer Format	_SERIAL_SWAP	Sign bit followed by Magnitude bit
Other Side Bt8970	_NO_BT	Other terminal is unknown
LOST Time Period	10	Set LOST = 1 second
Notes: (1). Channel Unit provides necessary scrambled data.		

Table 1-9. HTU-R Configuration Commands, Channel Unit Configuration

Command Name	Parameter	Remarks
Terminal Type	_HTUR	Remote Terminal
Configure Channel Unit	_2T1, _2E1, or _3E1	Configuration is determined by querying the number of bit-pumps and the framer type.
Configure Framer	_BT8360 (T1) or _BT8510 (E1)	Configuration is determined by querying the framer type.
Configure LIU	_ELS_E1_PCM30 or _ELS_T1_0_110	Configuration is determined by querying the framer type.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump code processes the API commands.

Bt8953 Channel Unit Code: The Bt8953 channel unit code processes the API commands. The _CU_CONFIGURE will initialize all the channel unit registers for the specified configuration. The _FRAMER_CONFIGURE and _LIU_CONFIGURE initialize the framer and LIU for transparent operation.

1.10.14 Deactivated State

Application Code: In the deactivated state, the application code issues the _DEACTIVATE API command (this will turn off the bit-pump transmitter) and channel unit _CuConfigureBeginStartup() function (this will mask the Tx/Rx 6 ms interrupts). The application code will then wait for LOS = 1. When LOS = 1 becomes true, the activation state is changed to the inactive state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump interrupt handler sets the LOS = 1 when the loss of signal is detected.

Bt8953 Channel Unit Code: The Bt8953 Channel Unit performs the same tasks as before it entered this state.

1.10.15 Inactive State

Application Code: In the inactive state, the _ACTIVATE and _CU_FORCE_SCR_ONES API commands are issued to initiate the activation process. The transmitter is initially silent. If the bit-pump LOS = 1 is detected, the activation state is then changed to the activating state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine does nothing during the inactive state; the bit-pump just waits for the _ACTIVATE API command. When _ACTIVATE is detected, the activation state machine will initialize the bit-pump for the activation process; this includes turning off the transmitter. The bit-pump code then waits until an S0 signal is detected (LOS = 0).

Bt8953 Channel Unit Code: The Bt8953 channel unit code waits for the _CU_FORCE_SCR_ONES API command. When the command is received, the channel unit configures the HDSL transmitter to Framed Scrambled Ones; where both the overhead and payload data are all ones. The Tx/Rx 6 ms and interrupts are masked and if the master loop, the DPLL is set to Open Mode.[See *_CuConfigureBeginStartup()*]

1.10.16 Activating State

Application Code: In the activating state, the application code monitors the bit-pump activation status. If the bit-pump T-Act timer expires, the activation state is changed to the deactivated state. If the bit-pump Tx 4-Level flag is detected, the channel unit will enable the S1 signal (enable overhead). The activation state is changed to the activating S1 state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will wait until an S0 signal is detected (LOS = 0). When the signal is detected, the bit-pump will perform frequency lock, optimize phase search, adapt filters, etc. The bit-pump code will then begin transmitting the S0 signal and monitor the received signal for S1. When the S1 signal is detected, the bit-pump code will perform some adaptation and will start transmitting the S1 signal; the Tx 4-Level flags is set.

Bt8953 Channel Unit Code: When the Tx 4-Level is enabled, the channel unit enables the Tx/Rx 6 ms interrupts and the overhead data.[See *_CuConfigureTransmitS1()*]

1.10.17 Activating State S1

This state is not part of the ETR-152 HDSL standards, but is required for the implementation of the bit-pump and channel unit devices.

Application Code: In the activating state, the application code monitors the bit-pump activation status and channel unit status bits. If the bit-pump T-Act timer expires, the activation state is changed to the deactivated state.

If the channel unit INDR (valid Sync Word) flag is detected, the RTR indicator bit is set; the activation state is changed to the activate Rx state.

If the channel unit RTR flag is detected, the payload data is enabled; the activation state is changed to the active Tx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump is transmitting the S1 and monitors the received signal for S1. When the S1 signal is detected, the bit-pump code will perform some final adaptation.

Bt8953 Channel Unit Code: The Bt8953 channel unit code monitors the sync word and indicator bits. When the Sync Word is detected, the channel unit sync status is set to CU_IN_SYNC. If the RTR indicator bit is detected, the INDC bit is set. After a status is detected, then the channel unit code will set the appropriate configuration bits as determined by the application code.

1.10.18 Active Rx State

Application Code: In the active Rx state, the application code monitors the bit-pump activation and channel unit status. If the bit-pump T-Act timer expires or the channel unit INDC bit is detected, the payload data is enabled. The activation state is changed to the active Tx/Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will finalize its activation process.

Bt8953 Channel Unit Code: The Bt8953 channel unit code monitors the indicator bits. If the RTR indicator bit is detected, then the INDC bit is set. When the enable payload command is issued the following occurs: the payload data is enabled, the pair ID (ETSI) is set, the Tx/Rx FIFOs are reset, and if the master loop, then the DPLL is closed and the DPLL interrupt is enabled. [See *_CuConfigureStartupComplete()*]

1.10.19 Active Tx State

The active Tx state path is typically not taken; therefore it is shown as dashed lines.

Application Code: In the active Tx state, the application code monitors the bit-pump activation and channel unit status. If the bit-pump T-Act timer expires or the channel unit INDR bit is detected, the payload data is enabled. The activation state is changed to the active Tx/Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will finalize its activation process.

Bt8953 Channel Unit Code: The Bt8953 channel unit code monitors the sync status. If the In Sync bit is detected, the INDR bit is set. When the enable payload command is issued the following occurs: the payload data is enabled, the pair ID (ETSI) is set, the Tx/Rx FIFOs are reset, and if the master loop, the DPLL is closed and the DPLL interrupt is enabled. [See *_CuConfigureStartupComplete()*]

1.10.20 Active Tx/Rx State

Application Code: In the active-Tx/Rx state, the application code monitors the channel unit sync status. If the LOSW = 1 flag is detected, the activation state is changed to the pending deactivated state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump handles any temperature/environmental changes. Different status responses are continuously monitored and may be probed by issuing the corresponding status request commands.

Bt8953 Channel Unit Code: The Bt8953 channel unit code processes the Tx/Rx 6 ms interrupts. The appropriate status bits are set according to the status registers.

1.10.21 Pending Deactivated-State

Application Code:	In the pending deactivated state, the application code monitors the channel unit status. If the LOSW = 0 (return of sync word) flag becomes valid again within 2 seconds, then the activation state is changed back to the active Tx/Rx state. If the LOSWT = 1 (loss of sync word timer) is detected, the activation state is changed to the deactivated state
Bt8970 Bit-Pump Code:	The Bt8970 bit-pump activation state machine behaves similarly to the active Tx/Rx state.
Bt8953 Channel Unit Code:	The Bt8953 channel unit code behaves similarly to the active Tx/Rx state.

1.11 Application Example 1B: Bit-pump Only Activation State Diagram

This section describes how the Activation State Diagram is implemented when only using the Bt8970 bit-pump code. The Bt8970 Activation State Diagram is based on the ETSI ETR-152 and ANSI T1E1.4 HDSL specifications. Refer to the *bt8970.c* module for details.

For each state, the responsibilities of the application code vs. the Bt8970 bit-pump code are distinguished. The application code accesses the Bt8970 bit-pump code through the *_BtControl()* and *_BtStatus()* function calls.

The application code only needs to issue four API commands to implement the Activation State Diagram (this is assuming the bit-pump has been properly configured using the appropriate API command). By repeatedly calling the *_BtMain*() function, the Bt8970 bit-pump code will automatically progress through the Activation State Diagram. The API commands and their functions are as follows:

- _ACTIVATE—Initiates the Activation State Diagram.
- _STARTUP_STATUS—Monitors the bit-pump activation status, which allows the application code to determine what state the bit-pump is in and to detect any error conditions.
- _TRANSMIT_EXT_DATA—Transmits externally supplied payload data.
- _DEACTIVATE—Deactivates the bit-pump.

1.11.1 Initialization State

The initialization state applies to both the HTU-C and HTU-R.

1.11.1.1 Initialization

At power ON, the bit-pump software is initialized by calling the following routines:

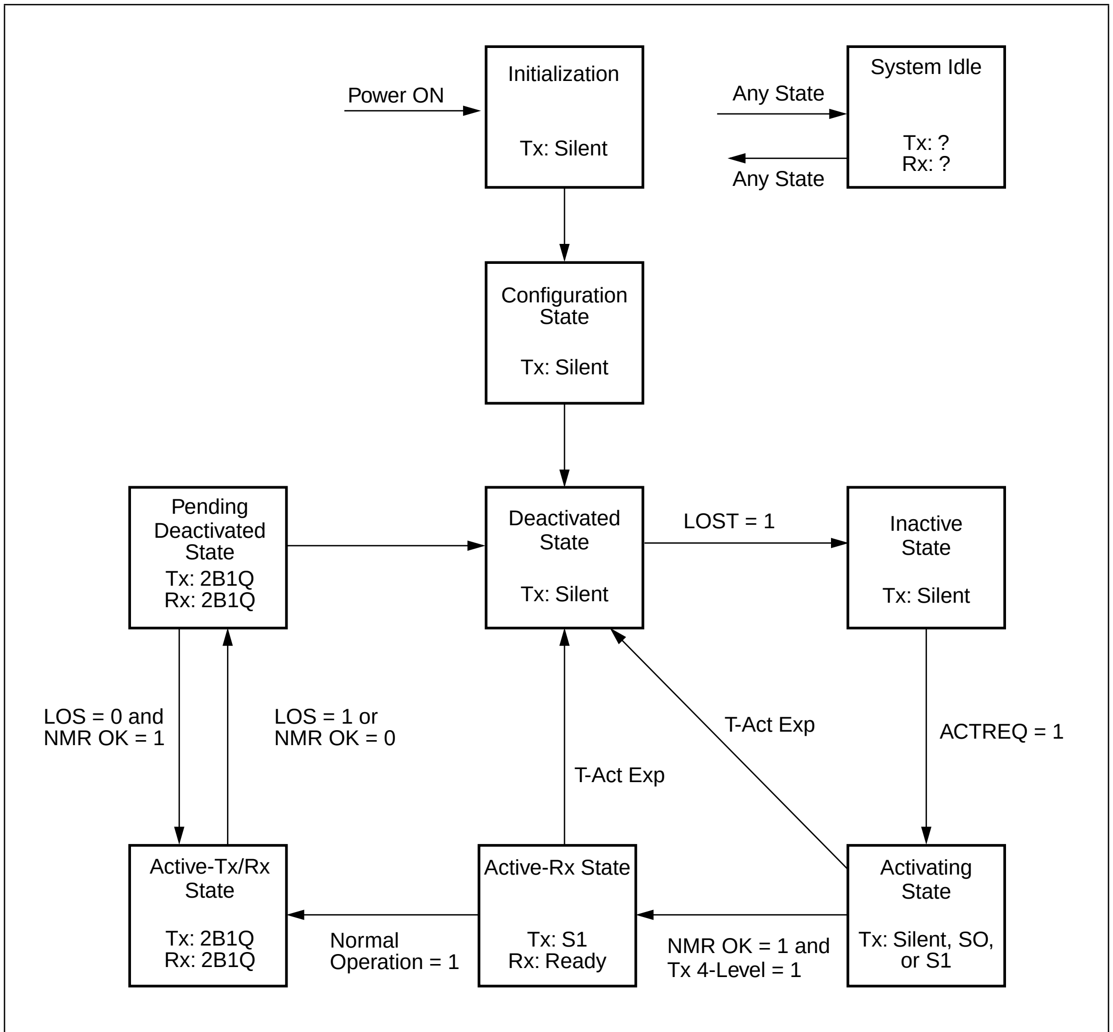
- _BtSwPowerUp()
- _MaskBtHomerInt()
- _Init8051() – Assuming Intel 8051 Family processor
- _InitGenPurposeTimers()

NOTE: The activation state is then set to the inactive state after the initialization state according to the HDSL standards.

1.11.2 HTU-C (LTU) Activation

HTU-C and LTU are synonymous. Figure 1-6 shows the state diagram for the activation of the HTU-C.

Figure 1-6. HTU-C Activation State Diagram



Note: LOS = Loss of Signal.
LOST = Loss of Signal Timer—LOS for more than 1 second.
T-ACT Exp = Activation timer expired.
ACTREQ = Activation Request.
NMR OK = Noise Margin greater than –5 dB.
Tx 4-Level = Bit-pump is transmitting a 4-level signal.
Normal Operation = Bit-pump has completed the activation process.

1.11.3 Configuration State

Application Code: In the configuration state, the bit-pump is configured by issuing the API commands listed in Table 1-10.

NOTE: The bit-pump on/off and symbol rate commands must be called first and second respectively. The rest of the API commands can be called in any order.

Table 1-10. HTU-C Configuration Commands, Bit-Pump Only Configuration

Command Name	Parameter	Remarks
Bit-pump ON/OFF	_PRESENT	Turn bit-pump #0 ON
Symbol Rate	Symbol rate is set according to SW1 (see Figure 1-3)	API value = (Symbol Rate / 4096) i.e. 96 = 392k / 4096 (784 kbps data rate)
Terminal Type	_HTUC	Central Office Terminal
Analog AGC Configuration	_SIX_LEVEL_AGC	6-level AGC is implemented
Start-up Sequence Source	_INTERNAL	Use internally generated scrambled 1s during activation
Transmit Scrambler	_ACTIVATE_SCR	Use bit-pump scrambler
Receive Descrambler	_ACTIVATE_DESCR	Use bit-pump descrambler
Framer Format	_SERIAL_SWAP	Sign bit followed by Magnitude bit
Other Side Bt8970	_BT	Other terminal is known to be Bt8970 based
LOST Time Period	10	Set LOST = 1 second

Bt8970 Bit-Pump Code: The Bt8970 bit-pump code processes the API commands.

1.11.4 Deactivated State

Application Code: In the deactivated state, the application code issues the _DEACTIVATE API command; this will turn off its transmitter. The application code will then wait for LOST = 1. When LOST = 1 becomes true, the activation state is changed to the inactive state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump interrupt handler starts the LOS Timer (LOST) when the _DEACTIVATE API command is received and when the LOS flag is TRUE (LOS = 1). The Bt8970 bit-pump interrupt handler will set the LOST flag (LOST = 1) when the LOST expires.

1.11.5 Inactive State

Application Code: In the inactive state, the _ACTIVATE API command is issued to initiate the activation process. The transmitter is initially silent. The activation request signal is always considered to be true (ACTREQ = 1). The activation state is then changed to the activating state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine does nothing during the inactive state; the bit-pump just waits for the _ACTIVATE API command. When _ACTIVATE is detected, the activation state machine will initialize the bit-pump for the activation process.

1.11.6 Activating State

Application Code: In the activating state, the application code monitors the bit-pump activation status. If the T-Act timer expires, the activation state is changed to the deactivated state. If the Noise Margin OK (NMR OK = 1) and Transmit 4-Level Indicator (Tx 4-Level = 1) flags are detected, the activation state is changed to the active Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump code starts transmitting the S0 signal and monitors the received signal for S0 (LOS = 0). When the signal is detected, the bit-pump will perform the optimize phase search, adapt filters, etc. The bit-pump code will then start transmitting the S1 signal and monitor the received signal for S1. When the S1 signal is detected, the bit-pump code will perform some final adaptation. At this point, the NMR OK and Tx 4-Level flags should become valid.

1.11.7 Active Rx State

Application Code: In the active Rx state, the application code monitors the bit-pump activation status. If the T-Act timer expires, the activation state is changed to the deactivated state. If the normal operation flag is detected, the _TRANSMIT_EXT_DATA API command is called and the activation state is changed to the active Tx/Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will finalize its activation process. The normal operation flag will be set when the bit-pump reaches normal operation.

1.11.8 Active Tx/Rx State

Application Code: In the active-Tx/Rx state, the application code monitors the bit-pump activation status. If the LOS = 1 or NMR OK = 0 (noise margin < -5 dB) flag is detected, the activation state is then changed to the pending deactivated state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump handles any temperature/environmental changes. Different status responses are continuously monitored and can be probed by issuing the corresponding status request commands.

NOTE: The developer should use the framer's sync word (or CRC) to detect the LOS or bad Noise Margin. In certain loop conditions, the bit-pump cannot guarantee 100% reliability in detecting a bad link.

1.11.9 Pending Deactivated-State

Application Code: In the pending deactivated state, the application code monitors the bit-pump activation status. If both the LOS = 0 (return of signal) and NMR OK = 1 (noise margin > –5 dB) flags become valid again within 2 seconds, the activation state is changed back to the active Tx/Rx state. If the LOS = 1 or NMR OK = 0 flag is still set after the 2 seconds, the activation state is changed to the deactivated state

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine behaves similarly to the active Tx/Rx state.

1.11.10 System Idle State

Application Code: In the system idle state, the application code does nothing. This state is useful when controlling the bit-pumps from the serial interface, i.e., UIP or diagnostic program.

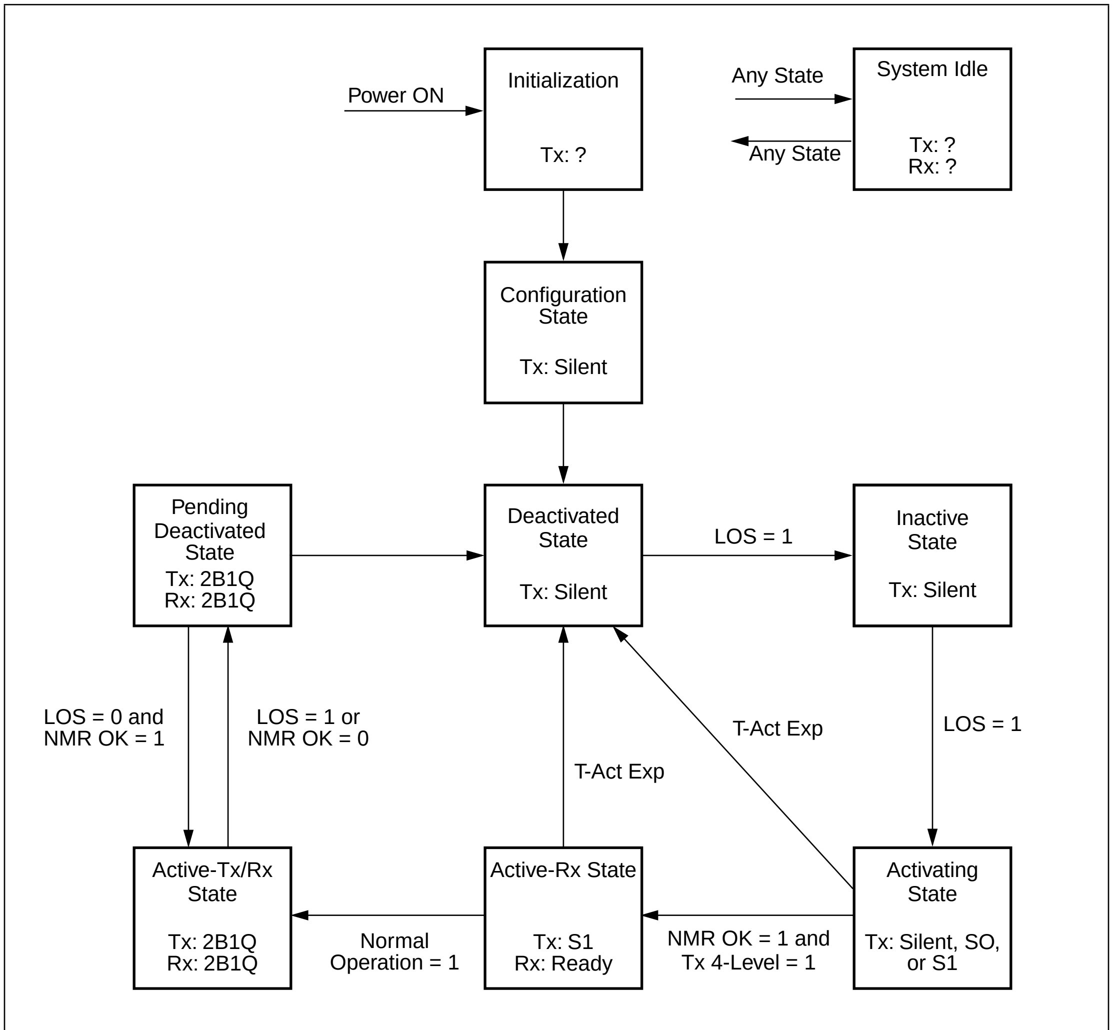
Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine performs any action specified from the serial interface, this could include performing the bit-pump activation or test modes. All instructions are received from the serial interface.

NOTE: In essence, the system idle state is acting like the dual processor mode.

1.11.11 HTU-R (NTU) Activation

HTU-R and NTU are synonymous. Figure 1-7 shows the state diagram for the activation of the HTU-R.

Figure 1-7. HTU-R Activation State Diagram



Note: LOS = Loss of Signal.
LOST = Loss of Signal Timer - LOS for more than 1 second.
T-ACT Exp = Activation timer expired.
NMR OK = Noise Margin greater than –5 dB.
Tx 4-Level = Bit-pump is transmitting a 4-level signal.
Normal Operation = Bit-pump has completed the activation process.

1.11.12 Configuration State

Application Code: In the configuration state, the bit-pump is configured by issuing the API commands listed in. Table 1-8.

NOTE: The bit-pump on/off and symbol rate commands must be called first and second, respectively. The remaining API commands can be called in any order.

Table 1-11. HTU-R Configuration Commands, Bit-Pump Only Configuration

Command Name	Parameter	Remarks
Bit-pump ON/OFF	_PRESENT	Turn bit-pump #0 ON.
Symbol Rate	Symbol rate is set according to SW1 (see Figure 1-3)	API value = (Symbol Rate / 4096) i.e. 96 = 392 k / 4096 (784 kbps data rate)
Terminal Type	_HTUR	Remote site terminal.
Analog AGC Configuration	_SIX_LEVEL_AGC	6-level AGC is implemented.
Start-up Sequence Source	_INTERNAL	Use internally generated scrambled 1s during activation.
Transmit Scrambler	_ACTIVATE_SCR	Use bit-pump scrambler.
Receive Descrambler	_ACTIVATE_DESCR	Use bit-pump descrambler.
Framer Format	_SERIAL_SWAP	Sign-bit followed by Magnitude bit.
Other Side Bt8970	_BT	Other terminal is known to be Bt8970-based.
LOST Time Period	10	Set LOST = 1 second.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump code processes the API commands.

1.11.13 Deactivated State

Application Code: In the deactivated state, the application code issues the _DEACTIVATE API command; this will turn off its transmitter. The activation state is then changed to the inactive state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine does nothing during the deactivate state. The Bt8970 bit-pump code monitors incoming API commands to determine what to do.

1.11.14 Inactive State

Application Code: In the inactive state, the _ACTIVATE API command is issued to initiate the activation process. The transmitter is initially silent. The activation state is then changed to the activating state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine does nothing during the inactive state, the bit-pump just waits for the _ACTIVATE API command. When _ACTIVATE is detected, the activation state machine will initialize the bit-pump for the activation process; this includes turning off the transmitter. The bit-pump code then waits until an S0 signal is detected (LOS = 0).

1.11.15 Activating State

Application Code: In the activating state, the application code monitors the bit-pump activation status. If the T-Act timer expires, the activation state is changed to the deactivated state. If the Noise Margin OK (NMR OK = 1) and Transmit 4-Level Indicator Flag (Tx 4-Level = 1) are detected, the activation state is changed to the active Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will wait until an S0 signal is detected (LOS = 0). When the signal is detected, the bit-pump will perform frequency lock, optimize phase search, adapt filters, etc. The bit-pump code will then begin transmitting the S0 signal and monitor the received signal for S1. When the S1 signal is detected, the bit-pump code will perform some adaptation and will start transmitting the S1 signal. At this point, the NMR OK and Tx 4-Level flags should become valid.

1.11.16 Active Rx State

Application Code: In the active Rx state, the application code monitors the bit-pump activation status. If the T-Act timer expires, the activation state is changed to the deactivated state. If the normal operation flag is detected, the _TRANSMIT_EXT_DATA API command is called and the activation state is changed to the active Tx/Rx state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine will finalize its activation process. The normal operation flag will be set when the bit-pump reaches normal operation.

1.11.17 Active Tx/Rx State

Application Code: In the active Tx/Rx state, the application code monitors the bit-pump activation status. If the LOS = 1 or NMR OK = 0 (noise margin < -5 dB) flag is detected, the activation state is then changed to the deactivated state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump handles any temperature/environmental changes. Different status responses are continuously monitored and can be probed by issuing the corresponding status request commands.

NOTE: The developer should use the framer's sync word (or CRC) to detect the LOS or bad Noise Margin. In certain loop conditions, the bit-pump cannot guarantee 100% reliability in detecting a bad link.

1.11.18 Pending Deactivated State

Application Code: In the pending deactivated state, the application code monitors the bit-pump activation status. If both the LOS = 0 (return of signal) and NMR OK = 1 (noise margin > –5 dB) flags become valid again within 2 seconds, the activation state is changed back to the active Tx/Rx state. If the LOS = 1 or NMR OK = 0 flag is still set after the 2 seconds, the activation state is changed to the deactivated state.

Bt8970 Bit-Pump Code: The Bt8970 bit-pump activation state machine behaves similarly to the active Tx/Rx state.

1.12 Application Example 2: Dual Processor Mode

The DUALPROC.C module contains the minimum code requirements to implement a dual processor mode bit-pump version as described in the “Software Integration” chapter, “Power-up Operations Sequence” section in the Bt8970 Software User’s Guide.

1.13 Application Example 3: ERLE Mode

The ERLEMAIN.C module contains the code to implement the Echo Return Loss Enhancement (ERLE) diagnostic version as described in Appendix C in the Bt8970 Software User’s Guide.

1.13.1 ERLE Dip Switches

DIP switches are used to configure the mode of operation. The modes are outlined in Figure 1-8 with switch 1 being on the left as the user faces the DIP switches.

NOTE: The AAGC is typically set to 12dB and the NLEC is turn OFF. The Symbol Rate range must match the board.

Figure 1-8. ERLE Dip Switch Settings

Evaluation Motherboard

SW1 (DIP Switches #1–8)

8	7	6	5	4	3	2	1

Reserved	**#8**

#7–1: Only valid in Stand Alone Mode

Data Rate (Supported Ranges)	**#6**	**#7**
1168 kbps (968 to 1368 kbps)	0	0
784 kbps (656 to 968 kbps)	0	1
528 kbps (160 to 656 kbps)	1	0
1552 kbps (above 1368 kbps)	1	1

AAGC Setting	**#3**	**#4**	**#5**
0 dB	0	0	0
3 dB	0	0	1
6 dB	0	1	0
9 dB	0	1	1
12 dB	1	0	0
15 dB	1	0	1
Reserved	1	1	0
Reserved	1	1	1

Reserved (should be kept in the 0 [down] position)	**#2**

Non-Linear EC	**#1**
Don't Adapt NLEC	0
Adapt NLEC	1

Note: 0 indicates the switch is down, 1 indicates the switch is up.

1.13.2 How to Run ERLE

The following items are required to run ERLE:

- ERLE EPROM
- 150 Ω Resistor
- RS232 Cable
- Terminal Emulator (i.e., QMODEM, KERMIT, Hyper Terminal, etc.)

The following four steps outline the process for running ERLE.

1. Insert ERLE EPROM—With the power OFF, insert the ERLE EPROM into the system.
2. Connect 150 Ω Resistor—Connect the 150 Ω resistor at the line side of the transformer. The 150 Ω resistor provides an impedance mismatch which in turn creates an echo.

NOTE: A simulated loop, or a real loop, can also be connected to determine the ERLE performance of the Bt8970EVM for that loop. When using a real loop, the far-end must be terminated with a 135 Ω resistor.

3. Connect RS232 Cable—Connect the RS232 cable from the EVM to the computer and run a Terminal Emulator (9600 baud, 8 bits, no parity, 1 stop bit).
4. Power ON—Turn the power ON to start the ERLE test. The results will be displayed in a 'printf()' fashion on the terminal emulator. To re-test, just press the reset button.

1.13.3 Expected Values

The ERLE results for different data rates are found on our FTP Server in the following directory:

```
ftp.brooktree.com/pub/hdsl/bt8970/erle
```

These results should be used to compare ERLE results from the customer's system.

Headquarters
Rockwell Semiconductor
Systems Inc.
4311 Jamboree Road,
P.O. Box C
Newport Beach, CA
92658-8902
Phone: (714) 221-4600
Fax: (714) 221-6375

US Northwest
Phone: (408) 249-9696
Fax: (408) 249-6518

US Los Angeles
Phone: (805) 376-0559
Fax: (805) 376-8180

US Southwest
Phone: (714) 222-9119
Fax: (714) 222-0620

US North Central
Phone: (630) 773-3454
Fax: (630) 773-3907

US South Central
Phone: (972) 479-9310
Fax: (972) 479-9317

US Northeast
Phone: (508) 692-7660
Fax: (508) 692-8185

US Southeast
Phone: (770) 246-8283
Fax: (770) 246-0018

US Florida/South America
Phone: (813) 799-8406
Fax: (813) 799-8306

US Mid-Atlantic
Phone: (609) 219-7462
Fax: (609) 895-2666

European Headquarters
Rockwell Semiconductor
Systems S.A.S.
Les Taissounieres B1
1680 Route des Dolines BP 283
06905 Sophia Antipolis Cedex
France
Phone: 33.4.93.00.33.35
Fax: 33.4.93.00.33.03

Europe Central
Phone: (49-89) 829-1320
Fax: (49-89) 834-2734

Europe Mediterranean
Phone: (39-2) 93179911
Fax: (39-2) 93179913

Europe North
Phone: 44(0) 1344 486444
Fax: 44(0) 1344 486555

Europe South
Phone: (33-1) 49 06 39 80
Fax: (33-1) 49 06 39 90

Asia Pacific Headquarters
Rockwell International
Manufacturing Pte. Ltd.
1 Kim Seng Promenade
#09-01 East Tower
Great World City
Singapore, 237994
Phone: (65) 737-7355
Fax: (65) 737-9077

Australia
Phone: (61-2) 9869-4088
Direct Fax: (61-2) 9869-4077

Hong Kong
Phone: (852) 2 827-0181
Fax: (852) 2 827-6488

India
Phone: (91-11) 692-4780
Fax: (91-11) 692-4712

Korea
Phone: (82-2) 565-2880
Fax: (82-2) 565-1440

Japan Headquarters
Rockwell International
Japan Company Limited
Shimomoto Building
1-46-3 Hatsudai, Shibuya-ku
Tokyo, 151 Japan
Phone: (81-3) 5371 1551
Fax: (81-3) 5371 1501

Taiwan Headquarters
Rockwell International
Taiwan Company Limited
Room 2808
International Trade Building
333, Keelung Road, Section 1
Taipei, Taiwan, 10548 ROC
Phone: (886-2) 720-0282
Fax: (886-2) 757-6760

China
Phone: 86-21-6361-2515
Fax: 86-21-6361-2516

Web:
www.rss.rockwell.com

E-mail Address
literature@rss.rockwell.com

For more information:
Call 1-800-854-8099

International Information:
Call 1-714-221-6996

N8970UGS04A
Printed in USA

Rockwell Semiconductor Systems

Please mail or fax this form to:
Manager, Technical Publications Dept.
Rockwell Semiconductor Systems
9868 Scranton Road
San Diego, CA 92121
Fax: (619) 597–4338

READER RESPONSE PAGE

We welcome your evaluation of this publication. Your comments and suggestions will help us make improvements to all current and future documents. Please rate the following characteristics of the publication you received:

1. Overall usefulness of the publication.
Poor Excellent
❍ ❍ ❍ ❍ ❍ ❍
2. Organization of the publication.
❍ ❍ ❍ ❍ ❍ ❍
3. Completeness and thoroughness of the material.
❍ ❍ ❍ ❍ ❍ ❍
4. Clarity and accuracy of the material.
❍ ❍ ❍ ❍ ❍ ❍
5. Usefulness of the diagrams and illustrations.
❍ ❍ ❍ ❍ ❍ ❍
6. Quantity of diagrams and Illustrations.
❍ ❍ ❍ ❍ ❍ ❍
7. Ease of finding specific information.
❍ ❍ ❍ ❍ ❍ ❍
8. Publication page layout and format.
❍ ❍ ❍ ❍ ❍ ❍
9. I received the publication(s): (Please mark one).
❑ Within one week after ordering it
❑ Within two weeks after ordering it
❑ Within three or more weeks after ordering it
10. The publication delivery time was: (Please mark one).
❑ Earlier than I needed for my needs
❑ Just in time for my needs
❑ Too late for my needs
11. In what format(s) would you prefer to receive future documents? (Check all that apply.)
❑ Printed books
❑ Internet/PDF delivery
❑ CD-ROM
❑ Other: ______________________________
12. How would you rank the importance of the following publication features? (1=Very important)
____ Publication design and layout
____ Accuracy of content
____ Detailed diagrams and illustrations
____ Receiving the publication in a timely manner
13. We would like to hear your thoughts on how we could improve this publication. Please write your comments or suggestions below:

____

____

____

Thank you for taking the time to evaluate this publication.

Document title:___________________________

Revision No.: ____________________________

Your Name: ______________________________

Title: __________________________________

Company: _______________________________

E-mail: _________________________________

Phone: _________________________________