

REVISION			
LTR	DESCRIPTION	DATE	APPROVED
<b>A</b>	<b>Released per ECO 581-00</b>	12/6/02	MAA

## 1.0 SCOPE

This document controls RTC Protocol Web Page.

## 2.0 Installation and User Instructions

See attached Instruction page 1-17.

REQUIRED SIGNATURES	SIGNATURE	DATE	<b>LiteTouch Inc. - 3400 So. West Temple Salt Lake City, Utah (801) 486-8500</b>		
Technical Support Manager	MARK ALAN ASHBY	12/6/02	DWG SIZE A	TITLE.  <b>RTC Protocol Web Page</b>	REV.  <b>A</b>
Technical Support	JASON B. FREEH	12/6/02	SCALE  N/A	DWG NO.  <b>10116-001-A.DOC</b>	
This document contains information proprietary to LiteTouch And may not be reproduced, published, or distributed in any Form or disclosed in whole or in part without the authorization of LiteTouch.			SHEET (S) 1 OF 1		
FORM FILE: ITEM-1.DOC			REVISED: 1 MAY 2000		

## Real Time Control / Diagnostic Commands and Responses

### Modification: December 05, 2002

Added functions for dynamic load group creation, modification, and information.

The real time control and diagnostics protocol is a subset of the 5000LC serial protocol (The remainder of the protocol is proprietary). It is based on ASCII character commands using a comma as a delimiter. The carriage return signals the end of a message. Each command has a 2-second timeout between characters, which will reset the command parser and require the entire command to be re-entered.

The 5000LC programming is based on response to system events like switch presses, holds, and releases or timer activation and controlling groups of loads. By definition, a load group may contain one or more loads. The load groups must be present in the system programming to access them through the real time control protocol. Each load in a load group has the following parameters:

Preset level - used in scenes (i.e. - SetLoadLevels command) and as a minimum "turn on" value.

Maximum level - maximum ramp level for dimming.

Minimum level - minimum ramp level for dimming.

Fade on - time in seconds for fading to the on level.

Fade off - time in seconds for fading off

The load group as a whole also has a ramp time parameter used in the StartRamp/StopRamp command sequence.

### Notify Setup Commands

#### **Command: Internal Event Notify**

**Format:** R,SIEVN,[0|1|2|3|4|5|6|7]

0 - Turn off notification

1 - Enable Internal (User (Combination and Startup) and Timer Event) notification

2 - Enable Internal, Switch, and LED Update notification (Maintain compatibility.)

3 - Enable Switch (Press / Hold / Release) notification

4 - Enable LED Update notification

5 - Enable Module Update notification

6 - Reserved

7 - Enable all implemented notifications (Combination/Timer, Trigger, LED, Module, etc.)

**Return:** none

**Description:** Enable notification of events such as combinations, timers, and button presses, LED updates, Module Updates, etc. These settings are mutually exclusive – i.e. you can't have both 3 and 5.

**Example:** R,SIEVN,7r (This will turn on notification for all events)

#### **Command: Station Notify**

**Format:** R,SSTNN,<xxx>,[n]

xxx - Station Address

n = 0 - Turn off notification

n = 1 - Enable Switch press/hold/release notification

n = 2 - Enable LED update notification

n = 3 - Enable both Switch press/hold/release and LED update notification

**Return:** none

**Description:** Enable notification of station events and/or LED updates for a station. This does not enable notification of combinations and timers.

**Example:** R,SSTNN,010,2r (This will turn on LED notification for Station 10)

## LITE TOUCH RTC PROTOCOL WEB PAGE

---

**Command:** Module Notify

**Format:** R,SMODN,<xxx>,[n]  
xxx - Module Address  
n = 0 - Turn off notification  
n = 1 - Enable Load State and Level notification

**Return:** none

**Description:** Enable notification of state or level change events for a module.

**Example:** R,SMODN,032,1r (This will turn on Load State and Level notification for Module 32)

### Notify Responses

**Response:** LED Update Notification

**Format:** R,RLEDU,xxx,bbbbbbbbbbbbbbbb

**Description:** Notifies that the LEDs for a station have just been updated. The first parameter is the station address and the second is the LED state map. The first 1 or 0 in the map corresponds to the first switch.

**Example:** R,RLEDU,010,0010110000000000r (This indicates that LEDs for button 3,5, and 6 are ON for station 10)

**Response:** Module Update Notification

**Format:** R,RMODU,<xxx>,<map>,level1,...,level8

**Description:** Notifies that the loads on a module have just been updated  
where: xxx is the module address in hex  
map is a hex map of loads on a module that changed. (1 = changed) (In this implementation, all loads are reported, so the mask is always FF).  
level1,...,level8 are the levels of the loads. -1 indicates no change. 0 indicates load was turned off. 1 – 100 indicates a load is on at the given level.  
Note: Unprogrammed loads are included.

**Example:** R,RMODU,0032,FF,90,-1,0,-1,50,0,30,-1r (This indicates that on module 32, loads 1,5, and 7 were turned on to 90, 50, and 30 respectively. Loads 3 and 6 were turned off. Loads 2, 4, and 8 did not change level or state.)

**Response:** Event Notification

**Format:** R,REVNT,<[SWP|SWH|SWR|TMB|TME|USR>,<ssso|val>  
[SWP],<ssso> - sss is the station ID and o is the button number.  
[SWH],<ssso> - sss is the station ID and o is the button number.  
[SWR],<ssso> - sss is the station ID and o is the button number.  
[TME],<val> - val is the Timer ID number.  
[TMB],<val> - val is the Timer ID number.  
[USR],<val> - val is the User Event (combination) ID number.

**Description:** Event Notification for Timers, User Events, and Station I/O events.

## **Diagnostic Commands**

### **Command: Full Station Test**

**Format:** R,DFSTS

**Return:** R,RTRES,<result>,<set>,<port>,<xxxx...>

*result* – is either 2, 3, or 4.

- 2 indicates an ACK (acknowledgment) from the station
- 3 indicates that more than one station responded (duplicate)
- 4 indicates the station responded with invalid data.

*set* – (unknown)

*port* – lists the port that the station was tested on. (zero based)<sup>1</sup>

*xxxx* – Bitfield which indicates which stations the result code applies to.<sup>i</sup>

There will be a total of 12 responses to this command for the C2000 cards.

**Description:** Runs a general station diagnostic test on ALL stations.

This command is hard to interpret. It is recommended that you use the CP5000 program to run this diagnostic command.

### **Command: Station Test**

**Format:** R,DSTST

**Return:** R,RTRES,<result>,<set>,<port>,<xxxx...> see Full Station Test for details.

**Description:** Runs a general station diagnostic test on all FOUND stations.

This command is hard to interpret. It is recommended that you use the CP5000 program to run this diagnostic command.

### **Command: Full Module Test**

**Format:** R,DFMTS

**Return:** R,RTRES,<result>,<set>,<port>,<xxxx...>

*result* – is either 2, 3, or 4.

- 2 indicates an ACK from the module
- 3 indicates that more than one module responded (duplicate)
- 4 indicates the module responded with invalid data.

*set* – (unknown)

*port* – lists the port that the module was tested on. (zero based)<sup>1</sup>

*xxxx* – Bitfield which indicates which module the result code applies to.<sup>2</sup>

There will be a total of 12 responses to this command for the C2000 cards.

**Description:** Runs a general module diagnostic test on ALL modules.

This command is hard to interpret. It is recommended that you use the CP5000 program to run this diagnostic command.

### **Command: Module Test**

**Format:** R,DMTST

**Return:** R,RTRES,<result>,<set>,<port>,<xxxx...> see Full Module Test for details.

**Description:** Runs a general module diagnostic test on FOUND modules.

This command is hard to interpret. It is recommended that you use the CP5000 program to run this diagnostic command.

### **Command: Get Clock**

**Format:** R,DGCLK

**Return:** R,RQRES,DGCLK,yyyymmddhhmmss

*yyyy* – year     *mm* – month     *dd* – day     *hh* – hours

*mm* – minutes     *ss* – seconds

**Description:** Requests the date-time.

# LITETOUCH

## RTC PROTOCOL WEB PAGE

---

**Command: Set Clock**

**Format:** R,DSCLK,yyyymmddhhmmss see Get Clock for details

**Return:** none

**Description:** Sets the clock with the specified date-time.

**Command: Get Sunrise**

**Format:** R,CGTSR

**Return:** R,RCACK,CGTSR,Sunrise at [HHMM] Astro Time is [HHMM] Local Time is [HHMM]

**Description:** Returns the time sunrise will occur for this day.

**Command: Get Sunset**

**Format:** R,CGTSS

**Return:** R,RQRES,CGTSS,Sunset at [HHMM] Astro Time is [HHMM] Local Time is [HHMM]

**Description:** Returns the time sunset will occur for this day.

**Command: Get Module Levels**

**Format:** R,DGMLV,<mmm> where: *mmm* is the module address in hex<sup>1</sup>

**Return:** R,RQRES,DGMLV,<map>,<level1>,<level2>,<level3>,...,<leveln>

where: *map* is a hexadecimal bitmap of the states (on or off) of the loads on the module.

*level1...leveln* are the load levels from 0 to 100% of the lowest (1) to the highest (n) loads on the module. Currently, a maximum of 8 levels is returned. A level of -1 indicates that the load is not a programmed load (i.e. not present in the uploaded \*.prg file).

**Description:** Returns the levels of all loads on a module.

**Command: Set Module Levels**

**Format:** R,DSMLV,<mmm>,<map>,<time>,<level1>,<level2>,<level3>,...,<leveln>

where: *mmm* is the module address in hex<sup>1</sup>

*map* is a bitmap (in hexadecimal) of the outputs to be changed

*time* is the time in seconds over which dimmable loads should transition to the new levels.

*level1...leveln* are the load levels from 0 to 100% of the lowest (1) to the highest (n) loads on the module. Currently, up to 8 levels are accepted, but not required. Setting a level to zero turns the load off.

**Return:** R,RDACK,DSMLV

**Description:** Sets the levels of selected loads on a module.

**Command: Memory Monitor Test**

**Format:** R,DMMTS,[1|2|3],[0|1]

1 - CPU

2 - C2000

3 - Modem

0 - Off

1 - On

**Return:** none

**Description:** Starts a memory monitor test that sends information out the serial port every 1 second. This command will be implemented with the CP5000 program in a future release.

## **Function Commands**

**Command:** **Get Load State**

**Format:** R,CGLST,<mmmo> where: *mmm* is module and *o* is output (zero based)<sup>1</sup>

**Return:** R,RQRES,CGLST,<*b*> where: *b* = 0 means off and *b* = 1 means on

**Description:** Returns the current state of a specific load.

**Command:** **Get Load Level**

**Format:** R,CGLLV,<mmmo> where: *mmm* is module and *o* is output (zero based)<sup>1</sup>

**Return:** R,RQRES,CGLLV,<*level*> where: *level* is 0 to 100%

**Description:** Returns the current level of a specific load. (This value only has meaning for dimmers modules)

**Command:** **Set Loads On**

**Format:** R,CSLON,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CSLON

**Description:** Turns ON the loads in the specified load group.

**Command:** **Set Loads Off**

**Format:** R,CSLOF,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CSLOF

**Description:** Fades OFF the loads in the specified load group using the fade off times specified in the load group parameters except on relay modules.

**Command:** **Set Load Levels**

**Format:** R,CSLLV,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CSLLV

**Description:** Fades ON the loads in the specified load group to their preset levels using the fade on times for the loads in the load group parameters.

**Command:** **Set Previous Load States**

**Format:** R,CSPLS,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CSPLS

**Description:** Restores the previous load state and level for all loads in the specified group.

**Command:** **Get Preset Levels (formerly Get Current Levels)**

**Format:** R,CGCLV,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CGCLV

**Description:** Copies the current load levels to the preset value for each load in the load group.

**Command:** **Get Minimum Levels**

**Format:** R,CGMIN,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CGMIN

**Description:** Copies the current load levels to the minimum dim level setting for each load in the load group.

**Command:** **Get Maximum Levels**

**Format:** R,CGMAX,<*load group*> where *load group* is a load group number.

**Return:** R,RCACK,CGMAX

**Description:** Copies the current load levels to the maximum dim level setting for each load in the load group.

## LITETOUCH RTC PROTOCOL WEB PAGE

---

**Command: Get Load Value**

**Format:** R,CGLVA,<mmmo>,<load group>  
where: *mmm* is module and *o* is the output (zero based) <sup>1</sup>  
<load group> specifies the load group number.

**Return:** R,RQRES,CGLVA,<level> where *level* is 0 to 100%

**Description:** Returns the “level” of the specified load in a load group. This value is used as either the minimum turn on “level” or scene preset “level”.

**Command: Set Preset Value (formerly SetLoadValue)**

**Format:** R,CSLVA,<mmmo>,<load group>,<value>  
where: *mmm* is module and *o* is the output (zero based) <sup>1</sup>  
<load group> specifies the load group number.  
<value> is 0 to 100%

**Return:** R,RCACK,CSLVA

**Description:** Sets the preset level of the specified load in a load group to value.

**Command: Get LED State**

**Format:** R,CGLED,<ssso> where: *sss* is the station address and *o* is the button position number (zero based) <sup>1</sup>

**Return:** R,RQRES,CGLED,<bool> where *bool* is 0 (off) or 1 (on)

**Description:** Returns the current state of a specific LED.

**Command: Get LED States**

**Format:** R,CGLES,<sss> where: *sss* is the station address (zero based) <sup>1</sup>

**Return:** R,RQRES,CGLES,<xx>  
*xx* - Is the decimal value of the binary bitmap of LED states.

**Description:** Returns the current state of all LEDs on a station

**Command: Get Valid Switches**

**Format:** R,CGVSW,<sss> where: *sss* is the station address (zero based) <sup>1</sup>

**Return:** R,RQRES,CGVSW,<ssss>,<xx>  
*ssss* – Station address in Hex.  
*xx* - Is the decimal value of the binary bitmap of valid switches.

**Description:** Returns a bitmap that describes the valid switches for this station

**Command: Set LED On**

**Format:** R,CLDON,<ssso> where: *sss* is the station address and *o* is the button position number (zero based) <sup>1</sup>

**Return:** R,RCACK,CLDON

**Description:** Sets a specific LED on.

**Command: Set LED Off**

**Format:** R,CLDOF,<ssso> where: *sss* is the station address and *o* is the button position number (zero based) <sup>1</sup>

**Return:** R,RCACK,CLDOF

**Description:** Sets a specific LED off.

## LITETOUCH RTC PROTOCOL WEB PAGE

---

**Command:** Open Loads

**Format:** R,COPNL,<load group> where *load group* is a load group number.

**Return:** R,RCACK,COPNL

**Description:** Opens the motion control loads in the specified load group.

**Command:** Close Loads

**Format:** R,CCLSL, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CCLSL

**Description:** Closes the motion control loads in the specified load group.

**Command:** Stop Loads

**Format:** R,CSTPL, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CSTPL

**Description:** Stops the loads in motion in the specified load group.

**Command:** Press Switch

**Format:** R,CPRSW,<ssso> where: *sss* is the station address and *o* is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CPRSW

**Description:** Generates a switch press.

**Command:** Hold Switch

**Format:** R,CHDSW,<ssso> where: *sss* is the station address and *o* is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CHDSW

**Description:** Generates a switch hold.

**Command:** Release Switch

**Format:** R,CRLSW,<ssso> where: *sss* is the station address and *o* is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CRLSW

**Description:** Generates a switch release.

**Command:** Toggle Switch

**Format:** R,CTGSW,<ssso> where: *sss* is the station address and *o* is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CTGSW

**Description:** Generates a switch press followed by a switch release.

**Command:** Press Hold Switch

**Format:** R,CPHSW,<ssso> where: *sss* is the station address and *o* is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CPHSW

**Description:** Generates a switch press then 0.4 seconds later a hold event.



## LITETOUCH RTC PROTOCOL WEB PAGE

---

**Command:** Toggle Loads On

**Format:** R,CTLON, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CTLON

**Description:** Toggles the loads in the specified load group. If the specified group consists of multiple loads at indeterminate states, all loads in the group will first be turned on. The next Toggle Loads On command will turn the loads off.

**Command:** Toggle Loads Off

**Format:** R,CTLOF, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CTLOF

**Description:** Toggles the loads in the specified load group off. If the specified group consists of multiple loads at indeterminate states, all loads in the group will first be turned off. The next ToggleLoadsOff command will turn the loads on.

**Command:** Start Ramp

**Format:** R,CSTRP, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CSTRP

**Description:** Starts ramping the loads in the specified load group.

**Command:** Stop Ramp

**Format:** R,CSPRP, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CSPRP

**Description:** Stops ramping the loads in the specified load group and leaves them on at the present levels.

**Command:** Start Ramp To Min

**Format:** R,CSRMN, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CSRMN

**Description:** Starts ramping the loads in the specified load group down to the min level

**Command:** Start Ramp To Max

**Format:** R,CSRMX, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CSRMX

**Description:** Starts ramping the loads in the specified load group up to the max level

**Command:** Lock Loads

**Format:** R,CLCKL, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CLCKL

**Description:** Locks the loads in the specified load group. This makes this load group inoperable from any source until it is unlocked.

**Command:** Unlock Loads

**Format:** R,CUNLL, <load group> where *load group* is a load group number.

**Return:** R,RCACK,CUNLL

**Description:** Unlocks the loads in the specified load group. This releases control of the load group, making it operable from all sources

## LITETOUCH RTC PROTOCOL WEB PAGE

---

**Command:** Lock Switch

**Format:** R,CLCKS, <ssso> where: sss is the station address and o is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CLCKS

**Description:** Locks the specified switch making it inoperable to press, hold, or release commands until unlocked.

**Command:** Unlock Switch

**Format:** R,CUNLS, <ssso> where: sss is the station address and o is the button position number (zero based)<sup>1</sup>

**Return:** R,RCACK,CUNLS

**Description:** Unlocks the specified switch.

**Command:** Lock Timer

**Format:** R,CLCKT,<val> where: val is the timer identification number.

**Return:** R,RCACK,CLCKT

**Description:** Locks the timer making it inoperable until unlocked.

**Command:** Unlock Timer

**Format:** R,CUNLT,<val> where: val is the timer identification number.

**Return:** R,RCACK,CUNLT

**Description:** Unlocks the specified timer.

**Command:** Set Global

**Format:** R,CSETG,address,value

**Return:** R,RCACK,CSETG

**Description:** Set Global variable at *address* with *value*

**Command:** Get Global

**Format:** R,CGETG,address

**Return:** R,RQRES,CGETG,<value>

**Description:** Get Global variable at a specific global *address*.

**Command:** IncrementLoadLevels

**Format:** R,CUPLL,<loadgroupid>,<value>  
where: load group is the load group number / ID

**Return:** R,RCACK,CUPLL

**Description:** Increments (Ups) each load group member's current level by the value

**Command:** DecrementLoadLevels

**Format:** R,CDNLL,<loadgroupid>,<value>  
where: load group is the load group number / ID

**Return:** R,RCACK,CDNLL

**Description:** Decrements (Downs) each load group member's current level by the value.

## LITETOUCH RTC PROTOCOL WEB PAGE

---

**Command:** InitializeLoadLevels

**Format:** R,CINLL,<loadgroupid>,<value>  
where: load group is the load group number / ID

**Return:** R,RCACK,CINLL

**Description:** Initializes (Sets) each load group member's current level to the value

<sup>1</sup> In this Document (zero based) relates to the numbering scheme used for module and station ports, button positions, and module outputs. By using a zero based numbering system, a port, button position, or output #1 becomes #0, #2 becomes #1, etc. Therefore Station 53 button 1 would appear as 0530.

<sup>2</sup> Continuous string of byte values in hex. A single hex value could be 0-9,A-F. Each hex digit may be represented as a 4 digit bit map like this:

0 = 0000	4 = 0010	8 = 1000	C = 1100
1 = 0001	5 = 0101	9 = 1001	D = 1101
2 = 0010	6 = 0110	A = 1010	E = 1110
3 = 0011	7 = 0111	B = 1011	F = 1111

Each pair of hex digits represents 8 stations where the first pair represent stations 7-0 like this  
AA equates to 1010 1010 and would indicate stations 7,5,3 and 1 are val

---