Chapter 8 -- DISPLAY OPERATIONS

There are thirteen commands which affect the display operations and which are most appropriately utilized during the map editing process. The **coord_type** command controls the X,Y units display, while ginput allows the user to specify a graphic input device (**mouse**, **dig1**, **dig2**). The **graphics** command influences what types of data will appear in the map display area TILE, while the **redisp_map** will redraw the display area data which can be displaced or degraded by the display of help text information. Three commands, **step**, **view** and **goto**, assist with movement and display resolution. The **edge_maps** function facilitates edgematching operations. Four functions (**level_clear, level_snap, level_dup, level_copy**) are designed to work with any or all of the 96 levels containing vector data, in conjunction with the current active map. The final command, **dem_view**, is designed specifically to provide a vertical, side-lit view of a digital elevation model (dem) file for analysis of the data quality.

MAPPING COORDINATES
LTPlus, by design, conducts its basic operations on any given map in pixel (row and column) coordinates. However, it provides the user with the opportunity to perceive movement within that same map in as many as 19 different coordinate unit types, as listed below. The current coordinate type selected will be identified in screen TILE 2. Note that the user may define special coordinate types. User-defined coordinate systems may be established and stored in the config file. The allowable units of measurement which may be displayed with the **coord_type** command include:

0 = Raster pixels
1 = N. longitude, W. latitude (0,0), decimal degrees
2 = UTM meters
3 = State Plane feet
4 = Digitizer1, absolute coordinates
5 = Digitizer2, absolute coordinates
6 = Map inches
10 = User-defined units
11 = User-defined units
12 = User-defined units
13 = User-defined units
14 = User-defined units
15 = User-defined units
16 = User-defined units
17 = User-defined units
18 = User-defined units
19 = User-defined units

THE MOUSE AND THE DIGITIZER

A Mouse By Any Other Name...
LTPlus relies upon the user to properly identify the active graphics input device (GIN) where more than one such device is logically available. A default device is established in the **config** file (**ginput=xxx**) and is immediately active upon execution of the **ltpgo** command from the keyboard. Once an active LTPlus session has been initiated, the user may, at will, change the graphics input device with the **ginput** command. And what are the choices?

The LTPlus program allows for the co-existence of three graphics input devices, although the average system configuration includes only two. With Release 2.00 for the Xenix-based systems, the user may identify and use either a Logitech 3-button or MicroSoft 2-button serial mouse. The term serial means that the mouse must be connected to the system through one of the hardware serial ports, and not through the DOS-popular mouse port. In either case, the **mouse_type** and **mouse_adr**[ess] must be recorded in the **config** file prior to usage. A Logitech C7 or C9 mouse has a **mouse_type = ltsmse**, while the MicroSoft **mouse_type = mssmse**. For AT&T UNIX users, the mouse is usually attached to the system as a bus mouse and the **mouse_type = attbmse**. The **mouse_adr = xxx** parameter is defined by the system name for the port into which the mouse is connected.

The second type of graphics input device is the combination of digitizer 1 (primary) and its cursor, and the third device alternative is digitizer 2 (secondary) and its cursor. Both types and addresses must be recorded in the **config** file prior to use (**dig1_adr, dig1_type, dig2_adr, dig2_type**). A typical configuration includes a mouse and one (primary--dig1) digitizer set-up.

What is the difference between a mouse and a cursor? Well, one is small, gray and furry....no, wait....wrong species. A mouse in the world of computers is a handheld device which is capable of sensing and transmitting the magnitude and direction of its movement from a previous to a current position. Some mice do this sensing optically while others detect motion mechanically (the ever-popular trackball or rollerball). The primary distinction is that the mouse operates on any smooth surface and does not require an accompanying digitizer board/tablet. The mouse is usually attached to the computer by a cable connected to either a bus or serial interface. In contrast to the digitizer cursor, the mouse has no viewing apparatus--it tracks relative movement only and is incapable of exact positioning over a physically-viewed target location.

The mouse is used to direct the movement and positioning of a screen cursor (crosshair) image. Many applications rely on the mouse to act as the user's agent in selecting actions and commands from menus or as the drafting tool in computer-assisted drafting (CAD) operations.

In LTPlus, the mouse is empowered to highlight (by positioning over) menu choices (commands or macros) and activate menu selections (click one of the preprogrammed mouse buttons). It may also be used during the graphic input/edit processes as a raster drafting tool. It effectively replaces the arrow key used in LIDES and the trackball used in Line Trace.

The cursor is a less independent little beast..it relies on a symbiotic relationship with a digitizer tablet. Unlike a mouse, the cursor detects and transmits its absolute (not relative) position on the tablet's surface. It is an electronic gizmo whose primary role is to supply a signal to the digitizer system controller which then determines the cursor's location in relation to the fine XY grid of wires enmeshed beneath the tablet's smooth surface. When the digitizer cursor is empowered in relative mode, the digitizer controller is actually sending absolute positioning information. The LTPlus software then calculates the relativity with the cursor's previous position, hence portraying the cursor motion as that of a mouse. If the ginput mode is set to absolute, the cursor positioning is represented by LTPlus as the accurate XY location reported by the digitizer controller.

This is an overly simplified description of what occurs with the electrostatic digitizer systems currently in use with LTPlus. Some digitizer products generate similar data but from a different technology base; i.e., sonics, mechanical motion, or optical definition (it's all done with mirrors, don't you know?).

Life With a Rogue Digitizer

All digitizers produce some form of absolute position information, although there is no standard format for the transmitted data and there are distinct differences with regard to resolution and accuracy specifications. Typically, the default format of the controller-generated data can be preset with dip switches, and the transmission format can often be soft set by the application (i.e., LTPlus can tell the CalComp 9000 Series digitizers exactly what format should be used to send XY and cursor button status information--see the *Appendices* for connection advice).

If you do not have one of the tested digitizer types (see *Appendices* for information about the currently supported graphic input devices), you are not out of luck. You can create a digitizer device file, reference that file in the **config** file, and hook up any digitizer that meets your resolution and accuracy requirements (piece of cake!). The *Appendices* also contain information about the content, format and path for the digitizer device files.

It is important to note that LTPlus allows the user to define the active graphics input device and to further specify, with regard to the digitizer-cursor hardware, whether the positional information is relative or absolute. In actuality, the digitizer controller (non-smart models) always supplies absolute **XY** positional data and LTPlus converts that to a relative strategy, if requested by the user. Yep, Jocko-- it is all done with mirrors!

GRAPHICS INPUT DEVICE

Only one graphics input device may be active at any given time, although up to three devices may be present. The standard three devices are the mouse, digitizer 1 and digitizer 2. Each device type and address should be identified in the **config** file. The primary, default startup device is also set in the **config** file as **ginput** = (**dev**). The user may reset the graphics input device at any time by executing the **ginput** command.

Entering the optional **-a** argument sets the coordinate mode to absolute; the default (no argument) is relative mode. In relative mode, LTPlus is basically treating the incoming **X,Y** positions as relative movement, with no relationship between the screen cursor position and the map location on the digitizer. This is perfectly sufficient for most LTPlus activities. However, absolute mode is desirable when digitizing or editing map data from the digitizer tablet; it provides the correlation between tablet cursor movement on the map surface and the position of the screen cursor on the corresponding map data display.

THE GRAPHICS DISPLAY

The **graphics** command provides the user with full control of the type and size of the map display. The user may designate that the entire screen be used for the graphics display by entering the optional **-f** argument; this may also be acomplished by using the mouse/cursor and depressing the center, or **start**, button over TILE 11, the small box in the lower, right-hand corner of the map display area. The default condition is a normal, TILE 7 (square), map display. Vectors from any set, all objects, attributed objects, or adjacent map vectors may be displayed in any combination as normal or highlighted colors (colors are set as **penN** values in the **config** file).

The display is reconfigured each time this function is executed. If no parameters are specified, you get the square, raster-only display. Note: If the map has not been **assembled** and no adjacent map data has been added, then there will be no vectors to display. All options are independent. If an object fits both the normal **[+n]** and the highlight [+**h**] groups, then it will be displayed highlighted. The order of the arguments makes no difference. Any number of normal [**+-n**] and/or highlight **[+- h**] parameter groups may be included. The **graphics** command is additive in nature. That is, display characteristics may be added and subtracted from the current display set by using the + and - symbols, as indicated in the table below:

(+)	Add the option specified to the display
(-)	Remove the option specified from the display
(+-)f	Full screen width graphics display (no menus)
(+-)r	Display the raster (**-r** = no raster display)
(+-)nXX	Display the vectors in XX (see below) as normal vectors.
(+-)hXX	Display the vectors in XX as highlighted vectors where:
	XX =

0	all map type objects (pts/lns/polys)
1-24	map objects in this set number
25	map objects with 1+ attributes
26	map objects without any attributes
27	map objects with 2+ attributes
28	map objects with less than 2 attributes
29	map objects w/out 2 identical attributes
30	segments w/same polygon on both sides
31	reserved
32	geodetic window (in vector form, following the projected curve of latitude & longitude)
33	geodetic planar map frame
34	geodetic corner
35-95	user imported vector files (DLG, MOSS, etc.)

REFRESHING THE DISPLAY

The raster display may deteriorate during the editing process; screen cursor movement across lighted (*on*) pixels may inadvertently turn them *off*. To recalculate and reconstruct the raster file and update its screen appearance, execute the **redisp_map** command.

CURSOR MOVEMENT

The screen cursor may be re-positioned to a specified location. Note that the current coordinate type units are in effect with regard to the X,Y arguments for this command. The user may display the cursor at a designated X,Y position by entering the **goto** command with type coordinate, then followed by the X and Y coordinate values, relevant to the currently active coordinate type (see **coord_type**). If the current graphics input device is the tablet cursor and it is activated in absolute mode, the viewing window on the screen will be moved to display the X,Y position defined by the tablet cursor button press (i.e., the X and Y coordinates are supplied by the X and Y coordinates transmitted from the tablet cursor).

The **goto** command may also be used to move to a specific record or serial number, by specifying the type as serial or record, followed by a valid record or serial number.

The cursor may be directed to a **log** file location by specifying the type as log, with the argument which follows representing a record number in the log file. If relative movement through the log file is desired, the **goto** command should define the type as log, followed by **a + or -** , for forward or backward movement from the current log file record number.

MAP MOVEMENT

The **step** command provides incremental map display movement in the specified direction. The map may be shifted left, right, up or down. The movement is relative to the current view. For example, to bring a portion of the map which lies to the right of the current view, execute a **step right** command. If the user attempts to **step righ**t while against the right edge of the map, the view will wrap around to the left edge and down. The **step** command operations may be graphically accomplished by moving the screen cursor above or below, or to the left or right of the colored blocks which appear in the elevator-type TILES 8 and 9 and then pressing the middle (or **start**) cursor button.

VIEWING CONTROL

With the **view** command, the map may be displayed at one of five resolution definitions as specified by the user. The view types include:

map	entire raster displayed in an approximately 400 x 400 pixel square area.
block	4:1 (approx. 1600 pixels from top to bottom)
full	1:1 (approx. 400 pixels from top to bottom)
zoom	1:8 (approx. 50 pixels from top to bottom)
last	previously viewed resolution

The map resolution displays the entire map's raster, filling the screen in TILE 7 and generalizing the display to approximately 400 x 400 pixels of resolution.

A **block** resolution places the map, at its defined raster resolution into a raster that is approximately 1600 pixels in height. Maps which have been created with a lower resolution will appear in whole in a portion of this display, while maps created with a higher resolution will appear only partially in the map display area.

A map may be displayed at **full** resolution, meaning that as much of a map as possible will be displayed in a 400-pixel high area. If the map's resolution is much greater, then only a portion of the map will appear, centered around a user-entered X,Y position. If the resolution of the map is much less, then the entire map will appear at the greater resolution (approximately 400-pixels high).

The **zoom** resolution displays an area of the map (identified by the user-inut X,Y center position) at approximately 50-pixels high.

If the resolution type variable is entered as the term **last**, the display will change to the previous view. Often, the **view last** command is mapped to one of the cursor or mouse buttons, allowing for efficient 'swapping' of views for editing work (i.e., between **view full** and **view zoom**).

EDGEMATCHING

The **edge_maps** command is responsible for determining the names of all of the adjacent maps which currently exist along the default **data_path** directory in the **config** file. Names of adjacent maps which are found on the active online storage under the aforementioned directory path are then stored in the header file for the current, active map.

LEVEL COMMANDS

The LTPlus **level_** functions provide the user with the ability to perform edgematching operations and, perhaps more importantly, to aggressively resolve coincident line situations. LTPlus recognizes 97 levels of data (-1 through 95). Some of these levels are reserved by the system for pre-defined data sets:

Level	Description (pertain to active map only)
-1	Raster
0	Vector, all objects
1-24	Vector map objects in sets 1-24
25	Vector map objects w/1 or more attributes
26	Vector map objects w/no attributes
27	Vector map objects w/2 or more attributes
28	Vector map objects w/<2 attributes
29	Vector map objects w/out 2 identical attributes
30	Vector segments w/the same polygon on both sides
31	reserved
32	Vector geodetic window (lat & lon projection)

The remainder of the levels may be defined by the user:

Level	Description
33-95	User-imported vector files (dlg, moss, adjacent maps, etc.)

There are four **level_** commands which work with the aforementioned data levels. The **level_clear** command clears all data from an import/display level. This command is needed when old data on a level needs to be updated with a new import file. It clears all data from a single import/display level. It only works on levels 33-95 (levels [sets] 1-24 are controlled using the **set_** functions).

The **level_icopy** command changes the input mode to **level_icopy**, an interactive (one-line-at-a-time) method to selectively copy lines from import/display levels onto the active map raster. When in this input mode, the **start** command will locate the object from a level (32-95) which is currently displayed in normal color (**n+ graphics** option). The object found will be highlighted, but not moved to the raster yet. After an object has been highlighted, the **continue** command will copy the highlighted object onto the active map raster. This provides the best-possible copy (in the raster world) of concurrent (coincident) lines. Note: after the raster line has been **assembled**, the vertices can be snapped to the vector object from the import level for an even close match (see the **level_snap** command).

The **level_dup** command duplicates all of the data from an import/display level onto the active map raster. Objects and/or attributes may be transferred from any level numbered between 32 and 95 inclusive.

The **level_snap** command will snap all vertices of the objects in a given set (0-24) to the nearest vertex or side of an object on the specified level (32-95) if it is within a given distance. Note: the snap is preferably vertex-to-vertex. If a vertex in the given set cannot be matched to a vertex on the given level within the specified radius, then it will be matched to the nearest point on a side, if it is, again, within the specified radius.

DIGITAL ELEVATION MODEL

The **dem_view** function allows the user to view a digital elevation model(dem) on the screen as if the topography were side-lit. This allows a final check of the dem file for elevation errors, smoothing, etc. The sidelighting source direction is identified by the user as part of the command parameters. For more information about the usage of these commands, please review the *Command Language Reference* section of this manual.