

**STATE OF FLORIDA  
DEPARTMENT OF TRANSPORTATION**

---



**EFB Mobile  
User's Handbook  
SURVEYING AND MAPPING OFFICE  
TALLAHASSEE, FLORIDA**

---

---



# Contents

<b>Introduction</b>	<b>1</b>
<b>Basic Concepts</b>	<b>2</b>
Point Naming .....	2
Reference Naming .....	3
Planimetric Information .....	4
("G:" field of Setup, HVD Obs and SOR Obs screens) .....	4
Geometry .....	4
Types of curves .....	5
Chains .....	5
Chain lists .....	6
Rules for chain lists .....	6
Vertical Information .....	11
Attribute .....	11
Zone .....	12
Identifying Objects .....	12
Feature Code .....	12
Graphically mixing chains and points .....	13
Feature code organization .....	13
Feature code options: Adding Text .....	13
Available feature codes .....	13
CEFB - Feature Code Input Mode .....	13
<b>The installation of EFB field software</b>	<b>15</b>
Installing Windows Mobile Emulator for Testing EFB on a PC .....	16
Overview .....	16
Files Required .....	16
Installation Steps .....	16
<b>The operation of EFB field software</b>	<b>21</b>
Controlling the System – Switchboard .....	21
Controlling the System - Data Prompts .....	21
Starting EFB .....	22
Files and Data Management .....	24
File Menu .....	24
Ctrl_Pnts Menu .....	27
Help Menu .....	28
The Switchboard .....	29
Data Transfer .....	30
Functions .....	30
Feature Codes .....	32
Overview .....	32

Using Feature Codes.....	32
Program Logic – Point and Feature Code entries.....	35
Attribute Update Prompt (CEFB Input Mode only):.....	36
Header (Switchboard, Option - 1).....	38
Functions Available: Header .....	39
Calibration - (Switchboard, Option - 2) .....	41
Axis Testing.....	42
Level Testing .....	43
Toggles.....	43
Weather.....	44
Comments.....	45
Functions Available: Calibration .....	46
Test - (Switchboard, Option - 3) .....	47
Executing Axis Test .....	48
Functions Available: Axis Test .....	49
Executing Level Test.....	50
Functions Available: Level Test .....	50
Setup - (Switchboard, Option - 4) .....	52
Functions Available: Setup.....	54
Observations .....	56
HVD Observations - (Switchboard, Option - 5).....	56
Other Components of the HVD observation.....	60
Functions Available: HVD Observation.....	63
SOR Observations - (Switchboard, Option - 6).....	66
Components of the SOR Observation .....	67
Functions Available: SOR Observation.....	68
Remarks - (Switchboard, Option - 7) .....	71
Functions Available: Remarks .....	71
Chains - (Switchboard, Option - 8).....	73
Active Edit.....	74
Chain Names .....	74
Required Chain Data.....	74
Stationing .....	75
Automatic Building of Chain Lists .....	75
Chain Error: .....	76
Functions Available: Chains .....	78
Taping (Switchboard, Option - 9) .....	80
Taping Operation .....	81
Functions Available: Taping.....	82
Search - (Switchboard, Option - 0).....	86
Components of the Search Screen.....	87
Control Points.....	88
Control Point Settings.....	88
Functions Available: Control Point Settings.....	89
Adding Control Points .....	89
Functions Available: Control Points.....	90
Editing Control Points .....	92

**APPENDIX A - Standard Abbreviations** **93**

**APPENDIX B - Pocket Guide** **100**





# Introduction

Florida Department of Transportation (FDOT) Electronic Field Book system (EFB) is designed to meet the specialized operational requirements for highway route surveying for FDOT.

The Electronic Field Book is a system comprised of:

- Surveying Instrument(s) (usually Total Stations)
- Software
- Computer Platform(s)

The computer platform for field operations is a rugged, hand-held Personal Computer (PC) operating under Windows Mobile or Windows CE.

The computer platform for office operations is a desk top PC, operating under Windows XP or Windows 7.

The software is designed to be simple and consistent with current field practice with the classical EFB collection system in production since the late 1980's which ran DOS. The software-hardware combinations are also device (field computer, total station) independent, however only a limited number of total station devices are currently supported. The goal of the system is to provide both an automatic and manual field book for the gathering of conventional survey data, in a format that may be processed electronically and used later in the highway design process. The software also uses state-of-the-art data processing techniques to improve the accuracy of results delivered to the Designer or Mapper. The EFB permits uniformity and efficiency in the data gathering process, while not sacrificing the flexibility of the manuscript field book.

This documentation is intended to describe the operation of the EFB on the hand-held computer for field operations. The documentation serves as a description of software operation that is common to all supported hand-held computer hardware and the field operation of the system.

Processing the data collected is explained in separate publications.

# Basic Concepts

As with any computer, the EFB cannot make intuitive decisions about the environment in which it operates. Therefore, the surveyor "guides" the EFB through the survey tasks by structuring the data in screens designed for specific types of data, which are then recorded in chronological order, thus producing the complete record of field operations. The data are placed in the task screens by responding to prompts, and the screens are selected through the use of menus. The menus, which open specific data screens, are the means by which the EFB is guided through a job.

---

## Point Naming

("N:" field of Setup, HVD Obs and SOR Obs screens, and Chain screens)

Point names used by the Department will adhere to the abbreviations found in APPENDIX A - Standard Abbreviations located in this document. The user still has flexibility in naming, albeit within the limitations of the naming system described below.

For EFB a system of alphanumeric point names consisting of two parts was developed:

1. Alphanumeric prefix designated by the surveyor
2. Numeric suffix assigned by the system.

Users select an alphanumeric prefix consisting of up to seven characters in length while the numeric suffix is an integer, and is assigned by the system. The numeric suffix can also be from one to seven digits, depending on the length of prefix chosen by the user. The length of the combined prefix and suffix will not exceed 8 (eight) characters. Many thousands of points of the same alphanumeric prefix name, each differentiated by a unique numeric suffix can be created by such a system.

A point previously observed or occupied may be used at a later time in the survey by simply typing in the complete point name designation of the object (prefix and suffix) as it was earlier designated.

Some examples of point names are: PAVT5, TREE16, FH27, MH2, FL88, RR92, DR16.

Notice that in each case the numeric suffix is attached to the prefix to make the complete point name! A prefix of seven alphanumeric characters is the maximum that is allowed to be assigned by the surveyor. In such a case where seven characters are chosen as a prefix, the point name has only nine possible suffixes (i.e. JOHNSON1 through JOHNSON9). Although using long prefixes is allowed, it is inadvisable to do so since it limits the total number of point names that can be used, and can often be cumbersome to type on a limited field keyboard.

The following will illustrate the magnitude of point names available with such a point designation system:

A1	A9999999
AA1	AA999999
AAA1	AAA99999
AAAA1	AAAA9999
AAAAA1	AAAAA999



AAAAAA1      AAAAAA99

AAAAAAA1 AAAAAA9

'A' represents a possible alphanumeric character (A through Z).

You can see from the previous example that the total number of point names available is very large. Since this also provides flexibility, and the opportunity for meaningful abbreviations, the surveyor should use the simplest prefix in the point naming that is meaningful to him. He would select a prefix that will be easily remembered and will intuitively suggest the objects observed in the field.

---

## Reference Naming

("B:" field of Setup, HVD Obs and SOR Obs screens)

In many instances, it becomes necessary to communicate the *OFFICIAL* designation of an object. This is especially true when using the results of previous surveys and their monumentation, such as the National Geodetic Reference System (NGRS). Points monumented by the United States Coast and Geodetic Survey, National Geodetic Survey, U.S. Army Corps of Engineers, Departments of Transportation, or other organizations are often designated with a name, stamped on the monument itself. The EFB has the capability to capture these designations, but they are used as *reference* to the EFB point name described previously.

The *reference name* is used for communicating the stamping, or "official" designation of a mark or object for others who may subsequently tie to the object. The reference name is the way to associate common points between all other data sources, including past or future surveys. The reference name is also the means by which CAiCE's segment manager knows that points potentially contain control information; coordinates or azimuths. When the processed data is read into CAiCE, it will interrogate the incoming data, looking for reference names. If a reference name is found, CAiCE will attempt to find control coordinate or azimuth data for that point in a control database set up by the user. If the reference name found in the field data does not have control information in the control database, the user will have the opportunity to add some control information to the CAiCE control database for that given name.

An example of a reference name would be the stamping on an NGS horizontal control station. The surveyor notes the designation (stamping) of that mark with a reference name. The surveyor must also assign a point name for use by the EFB for the NGS station. So the point might have an EFB point name like "C13", but also have a reference name like "LEO STATION 1955".

Any monumentation set while collecting data having a particular stamping or designation of importance would also be given a reference name (that of the designation), since others may use that monumentation at some later date, and will need to be able to "reference" back to work performed establishing it (i.e. it's coordinates). This is the manner in which the surveyor may update his control database, once the processing of the data is completed.

The Reference Name should be entered into the B: field the first time that the point is utilized in that particular segment. For CAiCE, if it is a point to be used to establish new Control in the CAiCE control database it will be preceded with a period (.). If it is existing known Control already entered into the data base from a previous segment, enter the Reference Name only without the period.

Reference Names are typically keyed in exactly as they are stamped and may contain special characters such as dashes (-), spaces should be keyed as an under bar, and periods (.). However, they are limited to 16 alphanumeric characters. In some instances, a reference name may exceed 18 characters as stamped on the monument, but the user must abbreviate in some way.

An example of a reference name that might be keyed is:

**LOGAN\_TOWER\_ 1934:** The underbar ( \_ ) is printed when a space is input in the reference name field on the data screens.

**.60-92-A05:** The period (.) indicates this point will be added to the Control Database.

WARNING: It is extremely important for the field surveyor to use the reference name capability for designating which points are Control and their identity, and to carefully key in reference names to avoid misspelling. If the field surveyor is careless in typing, the ability to automatically create **CAICE** control files for the project during processing will be lost, and he may not be able to even perform important processing steps!

---

## Planimetric Information

("G:" field of Setup, HVD Obs and SOR Obs screens)

### Geometry

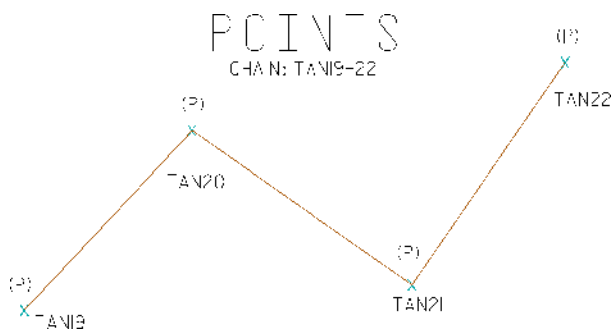
The *Geometry* is the geometric use of a point, when that point is part of a multi-point chain object. Many objects may be designated with a single point, such as a telephone pole, or a valve cover. When larger objects are to be designated, such as the edge of pavement along a length of highway, *many points* are often used. When an observation is made to locate an object, the EFB requires that information about that object be input to indicate the object's *potential* relationship in multiple-point configurations called chains. The information required is what is known as the *geometry* of the point.

For all practical purposes, there are simply three geometric elements that exist that are of significance in field surveying; individual points, lines, and curves. The geometric relationship between points becomes important when we begin to associate many points together, such as our edge of pavement example in the previous paragraph. When we speak in these terms, the geometric elements (more than one point) that concern us are lines and curves, since "a point is a point". The surveyor will attach geometric relationship information to every object (point) he locates using the EFB.

For EFB surveying, all geometry attached to a point is one of two types:

**Point (P)** - a single location for an object wholly defined by one unique set of coordinates, and the intersection point of straight lines (tangents) in a line string. This will become clearer in the examples shown later.

**Curve (C)** - a point on a *circular* arc, or a point on a *smooth* curve (spline) passing through a series of non-linear points. Again, this will become clearer in the examples shown later.



Point geometry, **P**, indicates that the "point" in question will never be used in a curve for line work passing through that point. To explain this a bit further, we need to have an object in mind, such as a tree. A tree will seldom be used as part of a curve, therefore the geometry for the location of that tree should be **P**, for point. Points connected together (a *chain*) may also have been given the geometry type of **P** for point. This is true, for example, when the surveyor connects the points representing poles supporting a power line. The pole locations are given a **P** geometry, since they all lie as points-of-intersection (PI's) of the wire span. See the illustration above, where the

geometry is set to P for the EFB points TAN19, TAN20, TAN21, and TAN22. Also note that points TAN10 through TAN19 must not exist, or the line work would pass through them as well.

A *Curve* geometry, **C**, indicates that the point in question is part of a curve. The **C** geometry would be used to indicate any point lying on a curve. More importantly, the curve geometry is used for points to indicate critical places of curvature, such as the PC's, POC's, PT's, etc.

Identifying the geometry of the points located in the field is therefore minimized to simply **P**, for point, or **C**, for curve. The geometry of points eases construction of complicated chain elements as we will see later.

## Types of curves

To simplify the EFB, only two types of curves are supported in the EFB. The first type of curve defined for the EFB is the *circular arc* which is a curve with a constant radius. When surveying a circular arc, it is only necessary to locate points actually on the curvature of the arc. The radius, and other geometric elements of the arc are solved from the relationship of the points surveyed.

The second type of curve is what is known as a *smooth curve*. A smooth curve *best fits* a spline curve passing through points defining the curve itself. A good example of an object that may be located with the smooth curve is a stream. There is no defined geometry for a stream, but the meandering of the stream can be shown to pass through survey locations modeled by the smooth curve.

To present how the circular arc, and the smooth curve are defined in the field, a few simple rules should be kept in mind when deciding how observations are to be taken:

- Three (3) curve points in sequence define a *circular arc* (PC, POC, PT) their sequence in the chain might look like ...PPP**CCC**PPP. Note that this circular arc may, or may not be tangent to the back and ahead lines connecting this "CCC" arc.
- A special case of a circular arc may be defined in the field by a single curve point, provided the back and ahead tangents are properly defined by P points, such as ...PP**C**PP, where the "P" points define the back and ahead tangents. A tangent circular curve passing through the "C" point is defined.
- A smooth curve is defined by four or more curve points in a sequence. ...PPCCCCCCCCPP...

The EFB user is responsible for indicating the geometry of every point he observes or occupies. As mentioned, there are only two choices:

1. The observed point is a **P**-Point location (i.e. a tree, fire hydrant, PI of two intersecting lines, etc.)
2. The observed point, **C**-Point, is on a circular arc or spline curve (i.e. edge of a meandering stream, a horizontal curve in the road, etc.)

It is possible that the assigned geometry of a point has no meaning. Such is the case when a tree, for instance, is accidentally given a C geometry type. The system does not care whether a single location for an object is P or C. When locations are tied together, it becomes very important that the surveyor correctly define the geometry. The *tying* of points together is how the geometrics of line work will be expressed, which is known as the chain.

## Chains

*Chains* named (chain name) series of points that have been previously located and their geometry (P -or- C) for each point in the series is defined.

Geometry of points is important when points are associated together to describe the shape of objects. The association of points is what is known as a *chain*. A chain is the *ordered* connection of points that define the boundary of an object. An analogy of how a chain is constructed is the child's drawing game *Connect the Dots*. The line work (chain) connecting the dots (points) exactly describes objects containing multiple points. The "chained" association of points with their

corresponding geometry(s) on the individual points in the chain creates circular arcs, smooth curves, and straight lines.

This series of points, the *chain*, is listed by the surveyor in the order that indicates the succession of points along an object. The line work connecting the successive points defines the limits of the object.

Like a point, a chain has a *chain name*. The chain name has a prefix and suffix, and follows the same general rules as point names. A chain name may contain a total of eight characters, where the last numerals are maintained by the EFB.

A chain may be *stationed*, which makes it a baseline. If a chain is stationed, observations may be made to points referencing the stationed chain, which is commonly known to the surveying community as location by station and offset.

The stationing of a chain ("**S**:" field of Chain Screens) is input in standard station notation or as a whole number and decimal fraction. An example is:

10+00 or 1000 for station 10+00.000

<OR>

13+66.273 or 1366.273 for station 13+66.273

The stationing defines the beginning length for the *first point* in the chain listing, and as progression is made down the chain list, the stationing *increases*. The station value is not input for any other point in the chain list, but may be computed in the office by the coordinate geometry package(s) available for the EFB. A chain can represent a three-dimensional object, and is controlled by the XYZ coordinates (or Station, Offset, Elevation) of every point included in the chain listing. If any point in the chain listing does not contain a Z ordinate, the chain will be two-dimensional (2D) only. If the surveyor includes a point in the chain listing where only an elevation or Z ordinate value may be computed, the chain is ignored altogether. A two-dimensional (2D) chain is any sequence of points where not all of the points in the chain listing contain computable Z ordinates.

Chains not only represent topographic features, but may be *design* elements of the route alignment. An example of this is a chain containing the critical points that comprise a highway route, which are the PC's, PT's, and PI's. Note that EFB can only describe lines, circular arcs, and splines (not spirals, but can be approximated with splines).

## Chain lists

Since a chain is simply a listing of points in sequence, the surveyor must identify the ordering of the points in the *chain list*. The EFB allows any point previously observed to be included in a chain list.

Since the surveyor defined the geometry of points (**P**oint location or **C**urve location) at the time of observation, *arcs*, *smooth curves*, and *straight lines* between points will be generated automatically based upon the sequence of points as they occur in the chain list. The surveyor does not have to define the elements of curves (radius, length, degree of curve, etc.).

## Rules for chain lists

Points used to define a chain are listed in the order that the line work (or curve work) connecting the points is to follow (not necessarily the chronological (time) order in which they were surveyed).

**EXAMPLE: ABC1,ABC17,ABC2** Point ABC2 must have been observed prior to ABC17 (due to the lesser point numerical suffix), yet ABC17 can occur earlier in the chain list.

Individual points, or groupings of points in the chain list are separated by commas, which indicates connection will be made between the individual points, or the groups.

**EXAMPLE: XSECT1,BX45,TL1** Indicates that line work will connect point XSECT1 to point BX45, and connect point BX45 to point TL1.

When hand entering chain lists, a group of points to be connected in *ascending* order which share the *same* name prefix, may be listed by the name prefix only, thereby *globally* including all points by that prefix and observed up to that point in timer in the chain. The order runs from the lowest numbered suffix to the highest numbered suffix.

**EXAMPLE: XSECT** If seven (7) XSECT points existed, EFB would determine this, and long-hand the global listing to connect XSECT1 to XSECT2, XSECT2 to XSECT3, and so on until XSECT7 is reached. The following would be created:

Shorthand list: XSECT1-7

Expanded list: SEC1,XSEC2,XSECT3,XSECT4,XSECT5,XSECT6,XSECT7

You can immediately see the convenience of the Shorthand list.

In the case of a global listing, the order of points may be *reversed* by placing a dash (-) in front of the listed global prefix to indicate the points are to be taken in *descending* order.

**EXAMPLE: -XSECT** If seven (7) XSECT Points existed, this global listing connects XSECT7 to XSECT6, XSECT6 to XSECT5, and so on until XSECT1 is reached.

A limited *range* of points may be listed by placing a dash between the beginning point and the ending point of the range desired. The order of the points, ascending or descending, is indicated by the magnitude of the point suffix that occurs earliest in the list.

**EXAMPLE: XSECT1-45** Indicates that point XSECT1 through point XSECT45 are to be chained in ascending order, connecting XSECT1 to XSECT2, XSECT2 to XSECT3, XSECT3 to XSECT4, and so on until XSECT45 is reached.

---

The range may also be listed in descending order, where XSECT45-1 indicates the reversed list of the previous example.

---

A double comma (,,) between individual points, global listings, or ranges in a chain list indicates a *gap* or break between the point before and after the double comma (gap).

**EXAMPLE: XSECT1-15,,18-22** Indicates there is **NOT** to be a connection of line work between point XSECT15 and point XSECT18, however the ranges indicated in the listing (XSECT1-15 and XSECT18-22) are to be connected.

A chain list does not have to be *singular*, which means any number of points with differing point name prefixes may be used in the list.

**EXAMPLE: RD,PAVT1-44,DWY6-14,,15-22,PAVT45-71,,72-108,-LAKE**

The point name prefix is *implied* or carried through the list until a new point name prefix is encountered. A more expanded listing of the previous example would look like: RD,PAVT1-PAVT44,DWY6-DWY14,,DWY15-DWY22,PAVT45-PAVT71,,PAVT72-PAVT108,-LAKE

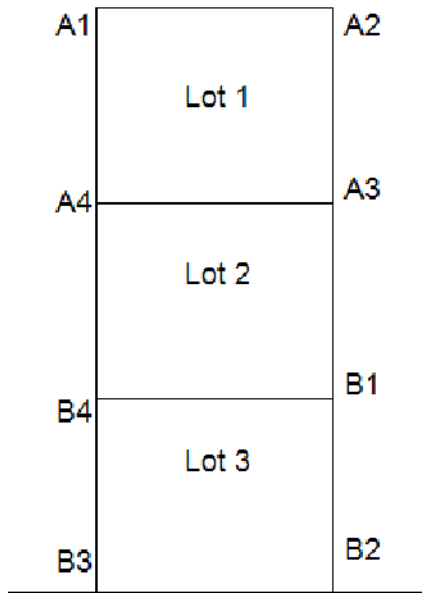
---

The user will probably elect to use the abbreviated methods of listing the chain to reduce the amount of typing in the field.

---

The following example should illustrate how the concepts of *global* listings, *reversed* listings, and *ranges* are used to simplify chain lists. Several examples of valid chain lists are given to describe the lots shown below. The use of the *comma* separator and *gap* is also demonstrated.

The examples



LOT1 could be listed as:

- A1,A2,A3,A4,A1 -or-
- A1-4,A1 -or-
- A1-4,1 -or-
- A,4 -or-
- A4-1,1 etc.

previous

LOT2 and LOT3 could be listed as:

- A4,A3,B1-4,A4,,B1,B4 -or-
- A4,A3,B,1,,B4,A4 -or-
- B,1,A3-4,B4 etc.

LOT1 and LOT3 could be listed as:

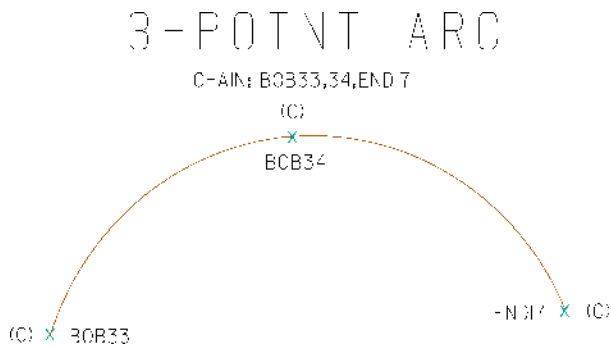
- A1-4,1,,B1-4,1 -or-
- A4-1,4,,B4-1,4 -or-
- A,1,,B,1 -or-
- B,4,,A,1 etc.

demonstrate a variety of ways a *similar* chain can be defined. You can see there are many ways to list a chain and achieve the same results. Now that we have demonstrated how the chain list is assembled, we will look at how the geometry of points in the chain list affects the graphical representation of the chain. Remember that it is the **P** and **C** geometry classification of points that influence the way arcs, smooth curves, and line strings are generated in the EFB chain lists.

The geometrics of the chain are derived from the geometry of the points included in the chain listing. Since points included in the listing will contain either a **P** or **C** (point or curve respectively) as their geometry type, the frequency of the **C** geometry type will determine how lines and curves in the chain are to be constructed.

Curves are determined in three ways:

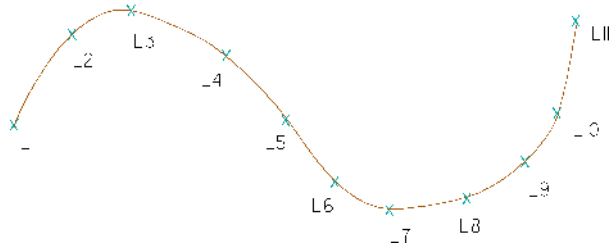
1. If three (3) points with geometry type **C** are encountered sequentially, then a circular arc is generated, where the first point is the **PC**, the second a **POC** and the third the **PT**.



2. If four or more (4 -> N) points with geometry type **C** are encountered sequentially then a smooth curve is fitted through the points. The first point encountered is the beginning of the smooth curve and the last one encountered is the end.

## SMOOTH CURVE=

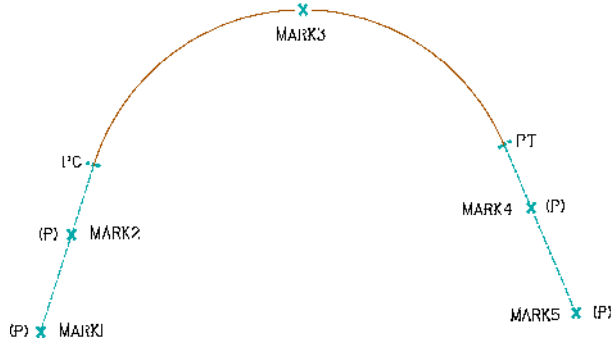
CHAIN: L1 II



3. If a pair (2) of points with the geometry type *P* precede a single point with the geometry type *C*, then are followed by another pair of points with the geometry type *P*, the first pair define the *back tangent* of the curve, and the second pair define the *ahead tangent* of the curve, and the point with the *C* geometry type defines the **POC**. In this case the **PC** and **PT** are computed internally by CADD software interpreting the EFB chain data, such as the SDMS processor, CAiCE, GEOPAK, and Civil3D.

## 1-POINT ARC

CHAIN: MARK1-5  
(C)



If two type *C* points are encountered in sequence and bounded by *P* points, they are treated as *P* points, and will be connected by line segments.

EXAMPLES 1), 2), and 3) reading from top to bottom:

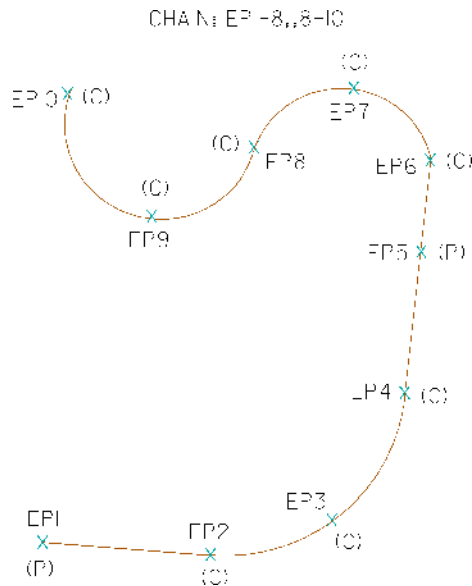
1)	2)	3)
P	P	P (BACK TANGENT)
P	P	P (BACK TANGENT)
C (PC)	C (BEGIN SMOOTH CURVE)	C (POC)
C (POC)	C	P (AHEAD/BACK TANGENT)
C (PT)	C	P (AHEAD/BACK TANGENT)
P	C	C (POC)
C (PC)	CP	P (AHEAD TANGENT)
C (POC)	C (END SMOOTH CURVE)	P (AHEAD TANGENT)
C (PT)	P	
P	P	

Example **3**) would have the **PC** and **PT** points computed automatically by the CADD Systems when the chain is graphically presented. The following expanded example helps explain the concept better for EXAMPLE 3) above:

FROM CHAIN LIST	INTERPRETED AS
P	P
P	P
C (POC)	PC (CREATED/INSERTED BY EFB)
P	C
P	PT
C (POC)	P
P	P
P	PC
	C
	PT
	P
	P

You can then see that merely coding the points with a P or C Geometry at the time of collection (the field surveyor can see if any curvature passes through any given point easily in the field) and creating a chain through that point, tangents, circular arcs, and smooth curves are easily created. More complex geometry, like compound and reverse curves can also be created by inventive use of the GAP (,,) which causes interpretation of the chain list geometry to re-start at the point following the GAP (,,).

The following figure illustrates how the GAP (a double comma, or ",," ) is used in Chain lists. Notice at point EP8, a gap causes the processing to re-start beginning with Point EP8 processing forward to EP9, and EP10, which is interpreted as a 3-point arc. In this case since EP6-8 was also an arc, we end up with a reverse curve (EP6-8,,8-10):



The GAP (,,) may also be used to "pick the pen up" and have the same chain resume drawing in a completely different location than the last point preceding the GAP. This is common where an object represents a single entity, but the line work may not be continuous. Such is the case of a single edge of pavement broken by driveways, or other complex objects. A word of caution however, the GAP should not be used to continue drawing similar objects that are not the same entity. For example it would not be proper to describe two separate objects in the field (i.e. two



different inlet tops with the same chain), because it was convenient just to use a GAP to separate them. That is both lazy and creates difficulties for the Office staff processing and cleaning up the Chain data.

---

## Vertical Information

We have seen the importance of points and chains in the description of the planimetric or *XY* location of objects. We also saw how the geometry of a point affects how a curvilinear object may be represented. We will now focus our attention on the vertical or *Z* location of objects and the points and chains which comprise them.

The ground surface or sub-surface is often located by the surveyor with a variety of techniques. One of the most familiar to most surveyors is *elevation* referenced to a vertical datum.

*Terrain Modeling* software has given us the ability to accurately represent the ground or other super/subterranean surfaces. Cross sections may be derived at any orientation or interval along a desired line that passes through the observation data (*model*). Observations taken as *cross sections*, or *random points* may be included in three dimensional chains (representing *break lines* or otherwise known as terrain discontinuities) are used to create computer models of the surfaces.

The *Triangulated Irregular Network (TIN)* is one method that the computer generates models of the surfaces.

Random points may represent the locations of planimetric objects. If a tree is located by the surveyor, and the Rodman held the base of the target pole at the ground, the elevation computed from that observation may be used in a TIN.

Cross section data may also be used in a TIN model since the observations that represent the cross sections are usually ground observations. A TIN model could be created exclusively from cross section observations, without additional ground observations between the surveyed stations or having additional three-dimensional chains representing break lines. The resultant TIN model would be significantly weaker because there would be no data *along* breaks in the terrain between the cross section station, and the computer will have to interpolate between sections.

The most important data that may be used in a TIN model are the *break lines*. A break line is surveyed by observing points along the breaks of a particular object, then associating those points together as a chain. This describes the profile of the terrain passing through the points that are included in the chain.

## Attribute

An *Attribute* is a sub-classification of a point to inform the processing programs what relationship that point's elevation ordinate has to a surface. The *attribute* of a point is used to tell the EFB how the elevation computed for that point relates to a particular surface.

There are three attribute(s) defined for the EFB:

**G - Ground Point** - A point whose computed elevation is on a given surface that may occur individually (spot shots), or are often connected together in a chain to describe a terrain discontinuity (break line) or the three dimensional geometry of an object along the surface. This constrains the TIN model to the elevations of the break line.

**F - Feature Point** - A point that indicates a *planimetric* object only, whose elevation is NOT to be used in the surface model.

**U - User Point** - A point whose elevation is *generally* not to be used when defining the surface for terrain modeling, but has a special meaning for the user. This elevation classification is similar to the Feature attribute, however EFB allows a distinction so data may be culled or queried in the downstream CADD systems based upon the attribute classification.

## Zone

A *Zone* is a tag to differentiate attributes between differing surfaces for terrain modeling. Sometimes the Zone is used to also distinguish other characteristics about the objects being observed or may be used to differentiate between different vertical surfaces. The user may desire to perform two independent TIN models of the same general vicinity, such as the existing ground before some construction activity and the *as-built* surface after construction. Points and chains may be retrieved from a database using the zone number as a search key.

The zone is designated for an object (point or chain) by its zone number. The zone number is an integer digit from one (1) through nine (9). The user may easily define up to nine different zones for each attribute. The zone number has also been used to define other classifications of data. For example the surveyor might wish to use a particular zone number to represent something other than a surface. For example, he/she may wish to reserve zone numbers to group real property information, and another to represent utility information, and yet another to group survey control information, and so on. Because there are 9 zones allowed in field operations with EFB, there are enough zone numbers to cover the distinguishing of different surfaces, and defining these groupings of data, should the surveyor choose to do so.

---

## Identifying Objects

Until now, we have input into the EFB only how a point or chain is to be used. It is also important to tell the system (and other users of the data) what the object really is, for purposes of graphical representation and information sharing.

## Feature Code

A *Feature Code* is a series of a maximum eight alphanumeric characters that identifies a point or chain as a particular object. The feature code ties the named object as an index to a definition of what that object really represents. A by-product of surveying usually is a scaled map based upon the survey measurements. To do this in an automated fashion, the user must communicate the graphical significance of objects to the computer. This *communication* is accomplished in the EFB by tagging *selected points* and *all chains* with feature codes.

The feature code is information that identifies a point or chain as a particular object. The system will interpret the feature code based upon a table (discussed later) and place symbology or line work for points and chains representing objects. The symbology and lines have varying shapes, color, thickness (weight) and line style for objects identified by their feature codes.

The user instructs the system as to what the points and chains represent by using feature codes. The feature code indicates which symbol to plot for individual points, and what type of line to plot for chains.

A feature code *could* be assigned to each point observed by the surveyor; however those points included in chains generally DO NOT have to have a feature code. *It is not necessary or recommended to assign a feature code to each point observed.* Only points requiring an individual graphic symbol on a map need be assigned feature codes. Each chain must be assigned a feature code in order for it to be plotted with other than the default line style, weight, and color.

If a series of points in a chain have no graphical significance (except for the chain itself), such as points along an edge of pavement, no feature code need be assigned to the individual points when they are observed. The chain that contains those points to describe the edge of pavement WILL be assigned a feature code. The converse is true for a power line, for example, that may span many supporting poles. The poles will be assigned the appropriate feature code because each pole is a planimetric map item. The chain connecting the poles representing the power line must also be assigned the correct feature chain feature code. The poles themselves ARE graphically significant, as well as the chain representing the power line.

## Graphically mixing chains and points

Chains will plot with the graphics associated with the chain feature code. Individual points in the chain will plot with the cells associated with their individual feature codes.

## Feature code organization

*APPENDIX B* of this documentation lists the feature codes used by FDOT. It also contains a brief description of the object(s).

## Feature code options: Adding Text

It is important to minimize the total number of feature codes used, because a long list is cumbersome in the field, and yet retain enough codes to effectively define most objects. Some objects that may be encountered in a field survey may not have been assigned feature codes. User defined text extensions to the feature codes may be used in those situations – this text is called a description.

The *description* is a text annotation that follows the feature code, and separated from it by a dash (-). Since the feature code field is 18 characters long, a surveyor can use short descriptions to place additional information about a point or chain in the EFB. If the feature code itself is eight characters, then nine characters are available for the description (one character is used by the dash(-)). An example is a tree. For example, TREE is the default feature code for any tree in the current FDOT feature table. To describe a tree more fully, the description could be used, as shown below:

### **Example: TREE-48" WHITE OAK**

Thus any size and species of tree is easily accommodated with the typed-in description.

EFB also supports aliases in its feature code list. Entering an alias code will cause the system to automatically substitute the current alpha numeric equivalent feature code from the cross-referenced EFB\_Settings.XML file. This is useful if feature codes are complicated, and a more simple designation would be appropriate for field personnel. This feature code will be substituted

into the **F:** field when entry is terminated by



For descriptions that exceed the 18 character *feature-description* limitation, the comment field (described later) could also be used.

## Available feature codes

In *APPENDIX B* a list of the FDOT feature codes that have been defined for the EFB can be found. The user may modify the default feature table, and create any feature codes desired.

## CEFB - Feature Code Input Mode

Entering Points is optimized by entering a Feature Code first. Point names may be created from the Feature Code or from the pre-assigned point name associated with the Feature Code.

When a Feature Code is entered into the applicable screen, it is searched in the Defaults table (maintained internally by EFB) for a match. This table is generated on-the-fly by EFB and associates Point names to the feature code used previously. When there is no match in the defaults table, the program will go to the EFB\_Settings.xml for the point prefix and parameters associated with the Feature Code entered.

The Feature Code defaults table is beneficial for reusing specific combinations for a given project. When a Feature Code is entered with a new Point Prefix or changes to a parameter is made, the new data set will be updated to the Feature Code Defaults table. The program will only use the default (what is specified in the EFB\_Settings.xml) settings when a new (to the project) Feature

Code is used. All other uses of the Feature Code will be pulled from the Feature Code defaults table.

For added flexibility, users will be prompted with the Attribute Update Prompt for which set of data to use...“as entered”, from the Defaults table or some combination of the two. Attribute Update Prompt will be explained in further detail in the EFB Operations section.

This function in EFB provides the functionality where if a given feature is selected, the other properties can become a default, such as the Attribute, Zone, and even the Point Name prefix. Likewise, if a Point Name Prefix is selected, then the Feature Code, Attribute, and Zone may automatically be selected as a default. This functionality improves consistency of the identifying data observed for objects in the field, helps eliminate misclassification mistakes, and reduces the amount of key entry the field surveyor has to make.

# The installation of EFB field software

1. *EFB* is initially loaded onto the Mobile unit by utilizing the USB cable that is provided by the manufacturer with the unit Windows ActiveSync and basic Windows file management commands.
2. Connect the cable that comes with the Mobile unit to the Mobile unit and then plug the other end into the host computer that has the source program files.
3. In the Mobile unit, use the Windows File Explorer commands to create a folder under the Program Files folder named:

**EFB.Mobile.CE50** for Windows CE operating system

<OR>

**EFB.Mobile.PPC50** for Windows Mobile operating system

4. Copy the applicable two files from the same folder structure of the host PC.

EFB.Mobile.PPC50.exe & EFB\_Settings.xml

<OR>

EFB.Mobile.CE50.exe & EFB\_Settings.xml

The following is required to be up to date on your mobile platform:

SQL CE 3.5

<http://www.microsoft.com/sqlserver/2005/en/us/compact.aspx>

.NET Compact Framework 3.5

<http://www.microsoft.com/download/en/details.aspx?displaylang=en&id=65>

<http://msdn.microsoft.com/en-us/netframework/aa497273>

The links provided give the Web locations of where the utilities may be downloaded, in case your data collector has not been updated in some time.

In addition, EFB mobile can also be run on the Windows desktop in an emulator mode. The following instructions are provided in order to set up emulation mode:

---

# Installing Windows Mobile Emulator for Testing EFB on a PC

## Overview

The following steps will enable the use of an Emulator version of the EFB Mobile unit.

## Files Required

1. Program files for EFB Mobile or EFB Mobile .CE
2. Windows Mobile 6.1.4 Professional Images (380.8MB)  
(<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=1a7a6b52-f89e-4354-84ce-5d19c204498a#Instructions> )
3. .Net Compact 3.5 Framework (EX: NETCTv35.cm.armv4i.cab)  
(<http://www.microsoft.com/downloads/en/details.aspx?FamilyID=e3821449-3c6b-42f1-9fd9-0041345b3385&displaylang=en> )
4. SQLCE3.5 for Windows Mobile (EX: sqlce.wce5.armv4i.cab)  
(<http://www.microsoft.com/downloads/en/details.aspx?displaylang=en&FamilyID=5544c638-c532-48e3-871c-58b49c5d855c> )


## Installation Steps

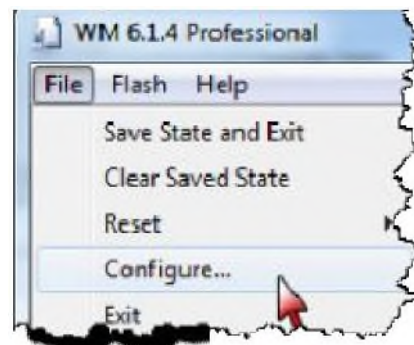
1. Download and Install **Windows Mobile 6.1.4 Professional Images** .
2. Download **.Net Compact 3.5 Framework** (NETCTv35.cm.armv4i.cab) and **SQLCE3.5 for Windows Mobile** (sqlce.wce5.armv4i.cab)
3. Create a folder on the **C:\** drive as a " share" for the emulator to retrieve files. Example path: **C:\Users\Username\My Documents\EFB Share**.
4. Place the **NETCTv35.cm.armv4i.cab** file, the **sqlce.wce5.armv4i.cab** file and the **EFB.Mobile** folder into the newly created *Share* folder.

---

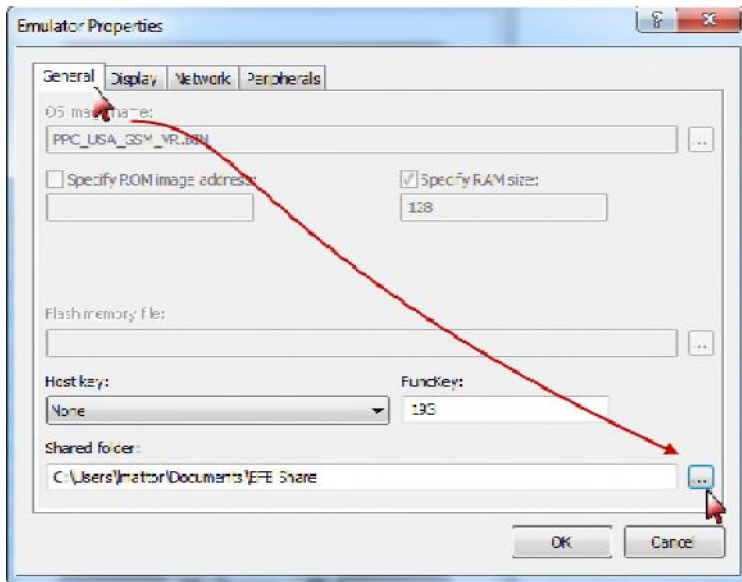
**The EFB.Mobile folder should be located in the Program Files \FDOT.EFB 3.1\Program folder. These may not be the exact file names as they differ from Mobile to Mobile.CE and there may be later versions on the Microsoft website.**

---

5. Launch **Windows Mobile 6.1.4 Professional** by navigating to **All Programs from the Windows** icon  and selecting **Windows Mobile 6 SDK > Standalone Emulator Images > US English > WM 6.1.4 Professional**.
6. Select the **File** menu and click **Configure**. This step is necessary to **identify (configure) the location of the Share folder, where files are stored for access from the Emulator**.

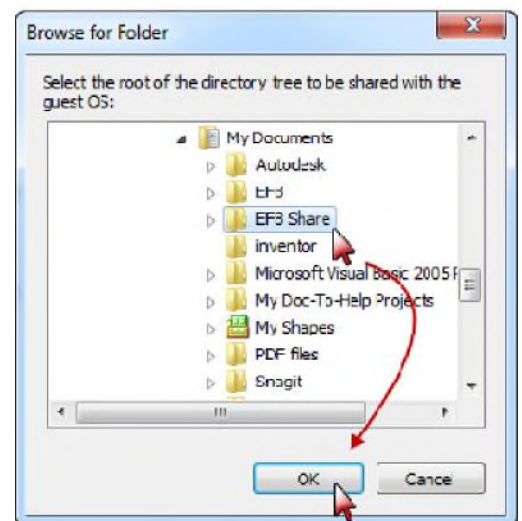


7. Select the **General tab**, then click the button adjacent to the *Shared folder:* **box**



**Navigate to the folder that was created in step 3 above.**  
Example path: [C:\Users\Username\My Documents \EFB Share](#).  
**Select the applicable folder, and then click the OK button** to save and exit.

**This "share" folder will be referred to as the Storage Card** in the Emulator.



**Optionally, you can setup a COM port for connection to a Total Station.**

To map a serial port:

Select the **COM port on the desktop to access via** the emulator.

Select the **Peripherals** tab.

Select the **COM#** from the **Serial port 0:** text box. Click the **OK button to save and exit.**

NAVIGATION HINTS:

**Folder Path Selector (A):** This is a dropdown list that allows the user to browse and select a folder or to switch between the Device and the Storage Card.

**UP button (B):** Moves the folder structure up one level to the current folder's parent folder.

**Virtual Keyboard toggle (C):** Opens and closes the On Screen virtual keyboard.

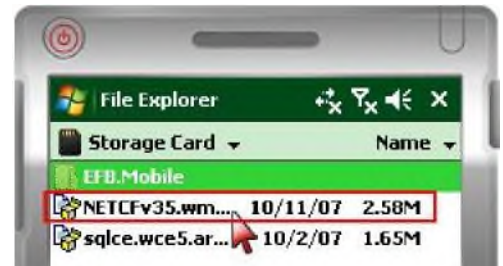
**Menu button (D):** Open the File Explorer menu.



8. Select the down arrow of the **Folder Path Selector**, and then select **Storage Card**.

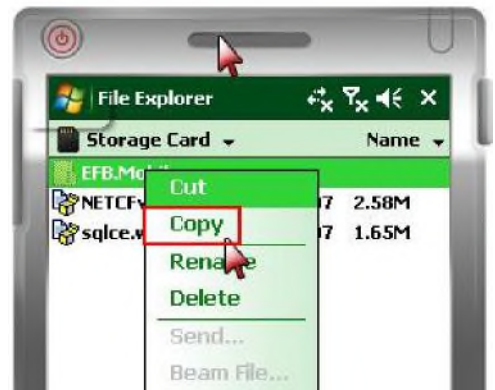
Select the **NETCTv35.cm.armv4i.cab** file in the list. This will start the Install process.

**When completed, close the install dialog box.**



9. Select the **sqlce.wce5.armv4i.cab** file in the list. This will start the Install process. When completed, close the install dialog box.

10. Using the program executable to the device, select the **EFB.Mobile** folder in the **Storage Card** folder and hold the left mouse button down until the **command list** appears, slide the mouse to **Copy** and select.





11. The **EFB.Mobile** folder needs to be located in the *Program Files* folder in the Emulator device. Navigate to this folder by:

1. Click the down arrow of the **Folder Path Selector**, and then select **My Device**

Select **Program Files** in the folder list.

Select the blank area under **Program Files** and hold the **left mouse button down until the command list appears**, slide the mouse to **Paste** and select.



12. Open the **EFB.Mobile** folder by selecting from the list.



13. Select **EFB.Mobile.PPC50.exe** file from the list to load EFB.



14. The EFB program Start window will appear as shown below:



15. Select **File** menu, then **Save State and Exit** command. This will **save the setup process that was just completed.**

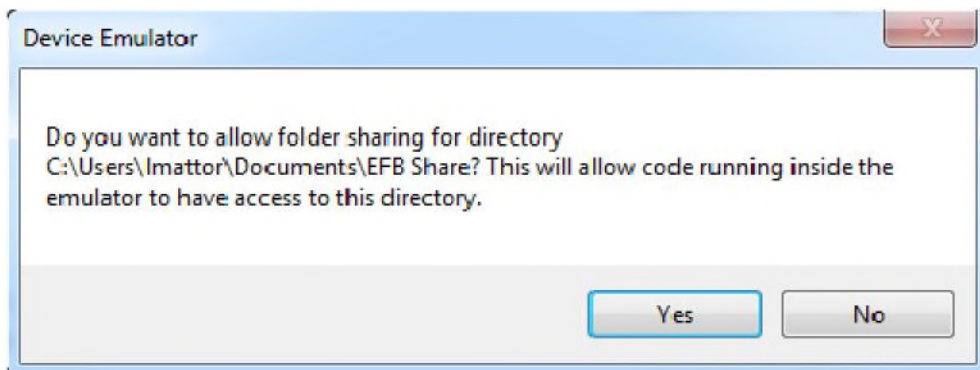


**To reopen the EFB Mobile Emulator, repeat step 5**

<OR> Select **WM 6.1.4 Professional** from the recent programs list in the Start Windows list.



16. Upon opening, click the **Yes button in the Device Emulator dialog box.**



# The operation of EFB field software

EFB is the name used throughout the remainder of this documentation for the computer program operating on the mobile unit used to gather field data. EFB is a single program. The Windows name for the program may be EFB.Mobile.PPC50.exe or EFB.Mobile.CE50.exe, depending on the operating system of the mobile unit.

EFB was designed to help guide the surveyor through the data collection operation. Imbedded in EFB are routines that perform checks for data integrity to help ensure the data gathering process proceeds smoothly and with minimal error. The user is appropriately warned of a variety of potential problem situations.

---


## Controlling the System – Switchboard


Operation of EFB is based on a *Switchboard* screen. A menu item is selected by pressing the appropriate *digit* on the keypad corresponding to the task the surveyor wishes to accomplish. The Switchboard may also be navigated with the use of the stylus that is provided with the mobile unit.

---

## Controlling the System - Data Prompts


Data is entered into a data field by pressing the letter key (*prompt letter*) corresponding to the data field that is to be edited. A *message field* at the bottom of the data screen displays information about the field selected. The cursor is automatically placed in the chosen field, indicating EFB is waiting for input. This assists the user in placing data in the appropriate

locations. When editing of a data field is complete, pressing  returns the cursor to *standby status* in the message field at the bottom of the data screen. The next function may then be evoked by the user.

The use of the  key will move the cursor from one field to the next, allowing the entry of all necessary fields without having to key in the prompt letter or going back to the message field standby status each time.

A system of locks deactivates portions of the keyboard before and during data field editing to help prevent erroneous data being accidentally input.

In a few instances, a data field is limited to a specific set of data; such as, the geometry and attribute data items discussed earlier. In this case, a *toggle* limiting selection to an appropriate option is used. In such situations with limited choices of data entry, pressing the prompt letter key repeatedly will toggle through all of the possible options for that field, without requiring the user to enter the field and edit it.

When using a toggle, pressing  is NOT required to terminate entry -instead it will close the screen, returning the system to the switchboard.

When the focus (cursor) is in the data field with a specific set of data as in a dropdown, selection



(up) and



(down) arrows to toggle through the set. To accept the selection and return

may be made by typing the character or the first character of a full-word option <OR> by using the



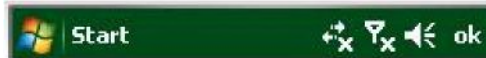
 to the message field, press . Press  to move to the next data field.

Data will remain in a data screen until the user is ready to record the information (selecting the

key) in order to verify that all data fields are correct. A "Saved" message is displayed indicating that the record operation was successful. Corrective action may be taken if an error occurred, or if any mandatory items have been left blank. If erroneous data is not detected, and the record operation was successful, the user has an additional opportunity to correct any mistakes using functions described later.

## Starting EFB

To start up EFB select the **Start** button on the Mobile unit.



Then EFB can be accessed from: **File Explorer > My Device > Program Files** and then selecting the program file.

Windows CE operating system filename: **EFB.Mobile.CE50.exe**

<OR>

Windows Mobile operating system filename: **EFB.Mobile.PPC50.exe**

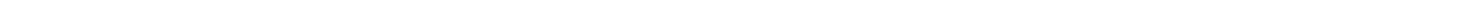
Upon startup of the EFB Mobile program a **Startup Log** is created. The log will provide system information about the *Name and Location of the EFB Mobile executable, Port names and the Operating System Parameters* of the mobile device that is being used.

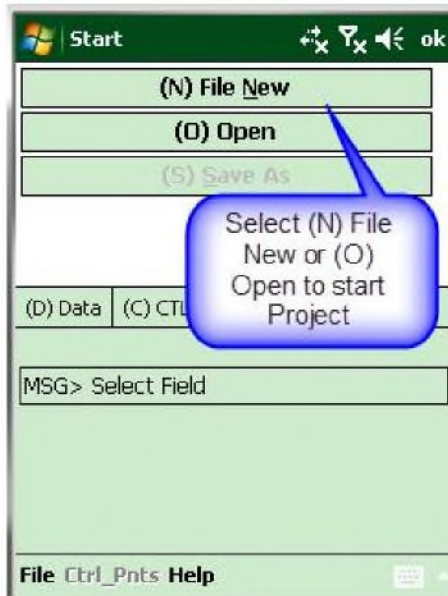
The Windows.Mobile default location for the **Startup\_Log.xml** file would be found in the EFB folder in *My Device\My Documents*.

EXAMPLE: My Device\My Documents\EFB\Startup\_Log.xml.

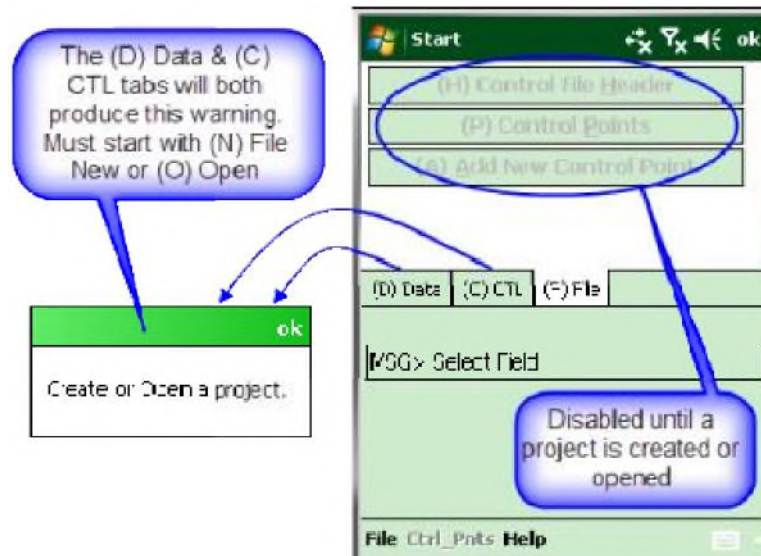
An environment variable (*Environment.SpecialFolder.Personal*) is used to locate the folder. This file may be viewed using a text editor program such as NotePad.

EFB Mobile program opens with the dialog box shown below. Upon execution of the program the user has two choices; open an existing project or create a new one. A project file is stored in a database format with an .SDF file extension.





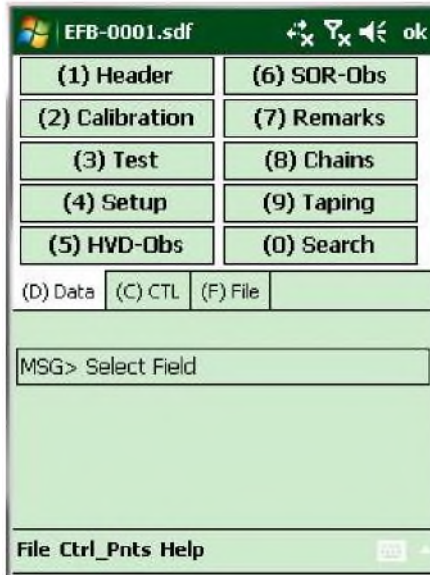
The dialog box has three tabs. The (D) Data and (C) CTL tabs functionality will be disabled until an action has been taken in the (F) File tab. Selecting the (D) Data and (C) CTL tabs will open a dialog message guiding the user back to the (F) File tab selections.



The (F) **File** tab has three options: (N) File New, (O) Open or (S) Save As, as illustrated above.

The (C) **CTL** tab also has three options (H) Control File Header, (P) Control Points or (A) Add New Control Point, which are shown disabled above.

The (D) **Data** tab displays the Switchboard . A view of the switchboard is below:



---

## Files and Data Management

EFB allows the user to gather survey measurements and data by either automatic entry from electronic surveying equipment via the Polling function or by manual key-in using the keyboard. The situation and the type of equipment used will determine how data will be entered.

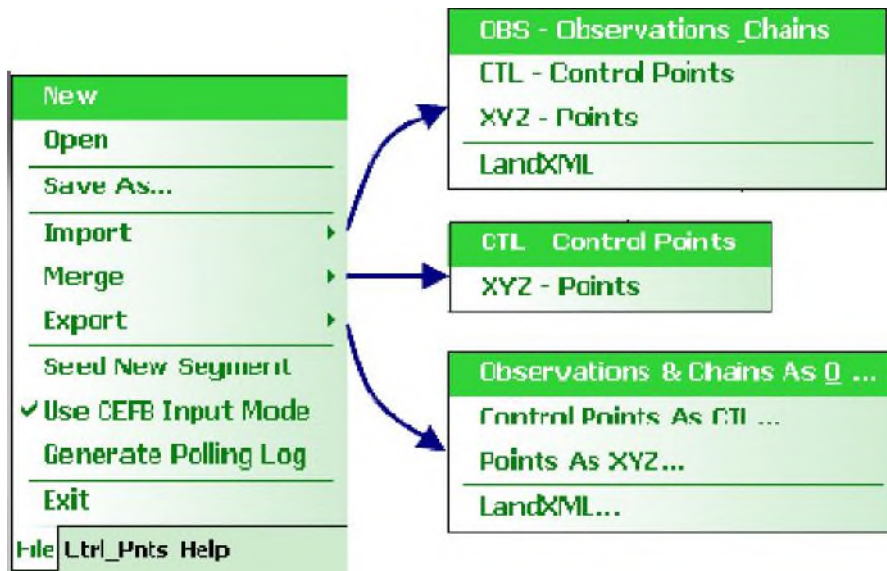
Files are stored in a Windows folder, determined by the surveyor. Files on the field computer are one of five types:

- .**SDF** = Native file format and contains all Observations, Chains and Control Points.
- .**OBS** = Observations and Chains
- .**CTL** = Control Points (known defined points)
- .**XYZ** = Control Points (coordinate values of points)
- .**XML** = LandXML

The first file type "SDF" contains all types of collected data and is the native file format of EFB. The remaining four types (OBS, CTL, XYZ, & XML) provide backward compatibility as well as a way to transfer data between applications.

### File Menu

To access a file select the **File Menu** at the bottom of the screen. The following commands are available and are explained below:



**Detailed explanation of each of the File commands:**

**New:** Opens the **Save As** dialog box, prompting the user to create a new project name, folder, type and location.



**Open:** Allows the user to open an existing SDF project file.





**Save As:** Opens the **Save As** screen prompting the user to create a new project name, folder, type (will default to SDF) and location. It is used to create the filename and storage location for the segment being surveyed. This provides a permanent file on the mobile unit physical drive.

**Import:** Allows the user to import existing segment data (OBS & LandXML), Control Points (CTL or XYZ) to be used with the segment being surveyed. The segment file may be imported as either:

- .**OBS** file which would contain Observations and Chains.

- .**CTL** file which would contain established Control Points.

- .**XYZ** file which would contain coordinate values of Control Points.

- .**XML** file (LandXML) which would contain all information (observations, chains and/or control points) that makes up a segment. These LandXML files provide portability to other applications.

**Merge:** Allows the user to merge existing Control Points (CTL) or Control Points (XYZ) to be used with the segment being surveyed. This differs from import; instead of creating an entire new file, only the new information is brought into the current file. The routine compares the existing data with the merged data and for any new point it will merge all of the information associated with that point. For points that are the same name, the only data that will be merged is for blank fields in the existing data in the current .SDF. The existing information takes precedence.

**Export:** Allows the user to export the current data to OBS, LandXML Control Points (CTL or XYZ) files. The segment file may be exported as either:

- Observations and Chains as .OBS** file extension. It contains established Observations and Chains.

- Control Points as .CTL** file extension. It contains established Control Points.

- Points As XYZ** as .XYZ file extension. It contains coordinate values of Control Points.

- LandXML** with .XML file extension. It contains all information (observations, chains and/or control points) that makes up a segment. These LandXML files provide portability to other applications.

**Seed New Segment:** Allows the user to seed a new segment with the projects existing points and chain information, so the new segment of data can call up the relationships of point names to Feature, Attribute, Zone, etc. data that was observed in the earlier segment.



**Use CEFB Input Mode:** When checked, the CEFB (C representing Cadastral) **Input Mode** feature makes the input mode to feature driven. This feature is "On" or  by default. **checked See Feature Code section for more information.**

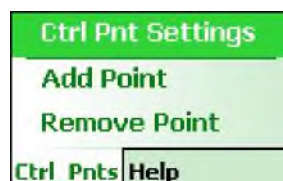
**Generate Polling Log:** When checked, the **Generate Polling Log** feature creates a log to capture the sequence of events while polling information from an instrument. This information is used by the programmers if debugging is necessary to sort out the communications issues between the EFB and the instrument. This feature is "Off" or NOT checked  by default. The log is useful for troubleshooting polling errors. Unlike the Startup\_Log which writes to an XML file, the **Polling Log** writes to the internal .SDF database only, and this SDF file would need to be transmitted to support for technical assistance with polling issues.

**Exit:** Closes the program. In the event that the file has not been saved, the following prompt for a final save.



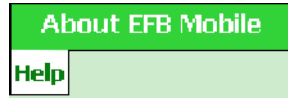
## Ctrl\_Pnts Menu

The **Control Points** menu provides a method to manage the setup of Control Points, by **establishing** the *Control Point Settings* for the project, *Adding* a point or *Removing* existing control points in an EFB file.



## Help Menu

**Help** > **About EFB Mobile** displays the version information.



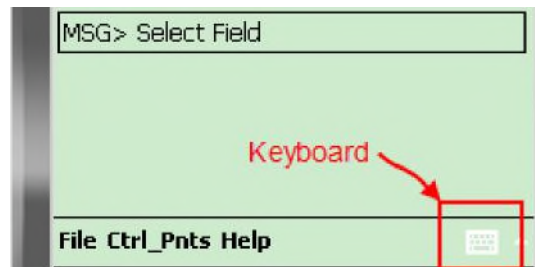
# The Switchboard

Selecting the (D) Data tab the EFB Switchboard is displayed.

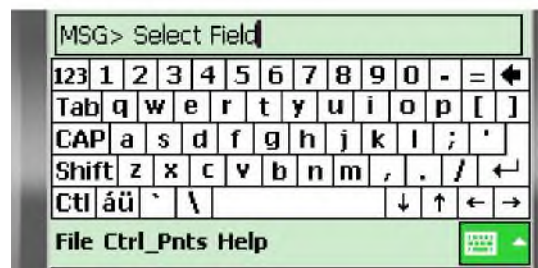
The Switchboard screen as shown below:



To activate the keyboard, select the keyboard image on the lower right hand section of the mobile unit.



Keyboard display:



If your Mobile unit does not have a physical keyboard on a screen, the virtual keyboard can be used with a stylus.

Once EFB is running, the opening menu is titled *EFB 3.1*. This menu contains the primary operations that are performed when surveying with EFB. This screen indicates the available surveying *tasks* supported by EFB. The options will be discussed one at a time and in greater detail, since they are the "meat" to using EFB as a data collection system.

Surveying operations are divided into general categories, or tasks. The tasks currently supported correspond to major field activities (like calibrating the instrument, performing an instrument Setup, taking, HVD or SOR observations, taking, creating Chains and so on). They are in EFB as follows:

**HEADER:** User information about the Segment.

**CALIBRATION:** User information about environment, crew and instrument, selection of instrument, and optional instrument testing.

**TEST:** A test screen for Axis and Level testing of instruments, which data is used during the processing of the data to help resolve systematic errors in the instruments being used.

**SETUP:** Data about the point the instrument occupies.

**HVD-Obs:** Data about the point the target occupies, and measurement data (horizontal direction, zenith angle (direction), and distance (slope and horizontal) combinations) of the target point relative to the current setup.

**SOR-Obs:** Data about the point the target occupies, and measurement data (station, offset, and rod reading combinations) of the target relative to a baseline.

**REMARKS:** General remarks during the data collection process.

**CHAIN:** Construction of Point lists (chains) that define curvilinear geometric objects.

**TAPING:** Remote taping operations independent of the traditional instrument setup.

**SEARCH:** Searches for a data record by setup, reference name, prefix, or point name.

A detailed description of each task will be included in the section about the data screen used to perform that task.

---

## Data Transfer

EFB has NO internal ability to transfer programs and data to and from the host PC computer. The Mobile unit will need either a USB or serial cable which will allow the transfer of files to the host PC. This is usually provided by the manufacturer. Files can be transferred in either direction with the use of Microsoft ActiveSync and standard Windows file management commands.

Users should consult the manuals and documentation from the manufacturer of the particular survey equipment being used for their serial communications options for connecting to total stations.

---


## Functions

It is appropriate to introduce the concept of *functions* at this time, since we will use one of them to record the Header into the segment record of the EFB database. Manipulation of data and data screens are performed also through the use of *functions*. The *function keys* are generally assigned to the numeral keys <<1234567890>> on the field computer, and are active only when the cursor is in standby status in the message field, NOT when editing a data field. All that is required to activate a function is to press the appropriate numerical key, and the function selected is performed.

Example of cursor in message field:

MSC> Select Field|

Cursor in standby status  
within the message field

Pressing  **to complete a function is not required, although some functions that potentially have detrimental effects on the data processing will require the surveyor to confirm his choice of action (usually a confirmation prompt) before the function executes.** The location of each function is mapped to a particular numeral key for individual data screens. Many functions perform similar operations in all data screens as we will see later.

# Feature Codes

## Overview

Feature codes can be setup in advance for the default values of the point / chain attributes. The following attributes may be preset: Point Geometry, Units, Alias, Prompt, DTM Attribute, Zone & Point Prefix.

Feature code "builder" functionality will assist the user in the creation of the descriptor of the feature code by utilizing a *builder\_template* command in the *EFB\_Settings.xml* file. This functionality will provide a more efficient method of data entry and result in standardized naming and attribution conventions.

The Administrator of the EFB software will predefine the default values for the point attributes and nomenclature for any given feature code by editing the *EFB\_Settings.xml* file.

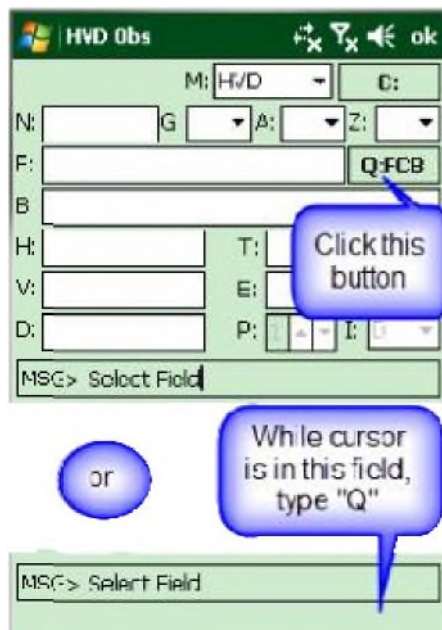
## Using Feature Codes

Feature Code Builder is available in the following screens: *Setup*, *HVD-Obs*, *SOR-Obs*, *Chains*, and *Taping* along with being integrated into the *Control Point* menu.

Feature Code Builder initiation is as follows:

EFB Input Mode (*Data input type in the File Menu, where Use CEFB Input Mode is NOT checked*):

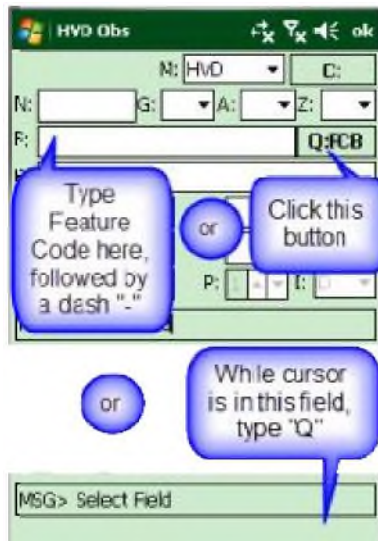
Typically in this scenario a Point or Chain prefix would be entered into the N: text box.



- Click the **Q:FCB** button to the right of the Feature Code field. This will display a list of applicable feature codes for the feature type (point or chain).
- <OR> with the cursor in the MSG> box, type the key Q to initiate the Feature Code Builder.

**CEFB Input Mode** (*The mode selected for data input type in the File Menu, where Use CEFB Input Mode IS checked*):

Typically in this scenario the Point or Chain Prefix is determined by the Feature Code being used, although it is not required as flexibility is built into the program.

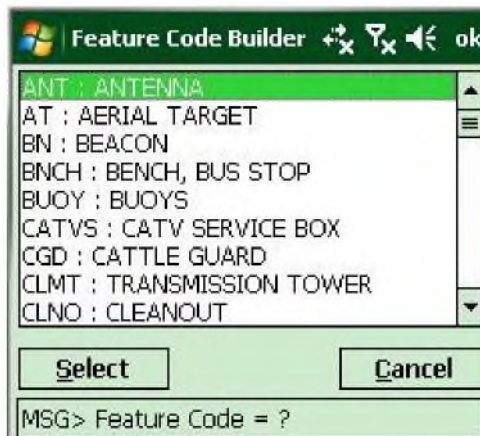


- In the *Feature Code (F:)* field, type a known feature code (EX: ANT) followed by a dash "-". Once the dash is keyed, the **Feature Code Builder** will initiate.
- <OR> Click the **Q:FCB** button to the right of the **Feature Code** field. This will display a list of applicable feature codes for the feature type (point or chain) desired.
- <OR> With the cursor in the **MSG>** box, type the key **Q** to initiate **Feature Code Builder**.

Based on your selection the builder will prompt for the proper settings of the particular feature selected.

EXAMPLE:

1. Select the applicable feature code from the **Feature Code Builder** list.



2. Use the up and down arrows or the stylus and cursor to highlight the required feature code, then click the **Select** button.

---

Only **feature codes that pertain to the geometry type will be presented in this list.**

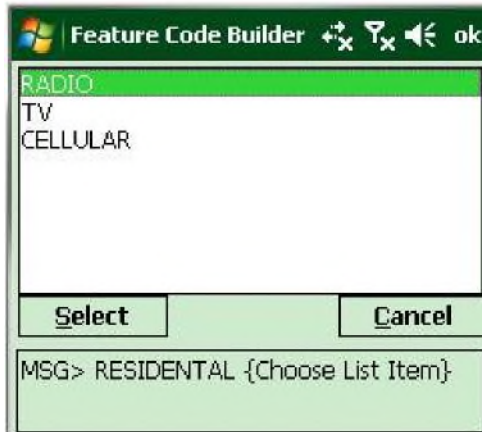
---

3. Upon selecting the feature code for Antenna "ANT" the following dialog box will appear:





- Use the up and down arrows or the mouse and cursor to highlight the applicable item, then click the  button to accept the selection and to open the next dialog box.

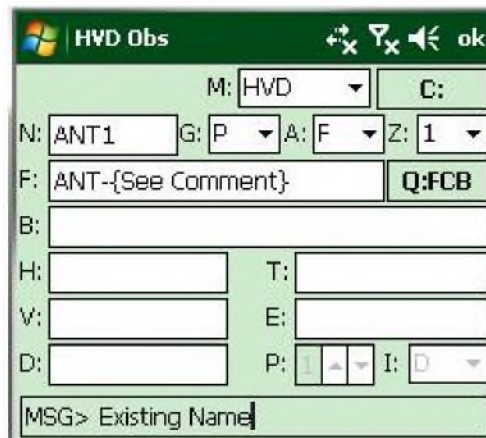


- Click the  button to accept the selection and to open the next dialog box, in this example all the information required was entered and the builder goes back to the original screen. This will build a Feature-description string of "ANT-RESIDENTIAL RADIO ANTENNA". Note that this builder can also embed the literal terms like "ANTENNA" in the description strings it build. This function will help standardize descriptions of objects observed in the field.

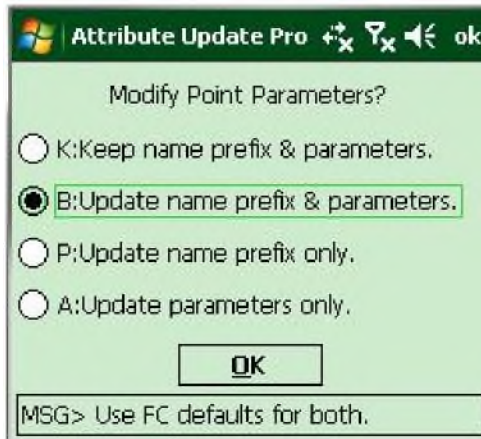
---

This example is in the **EFB Input Mode**

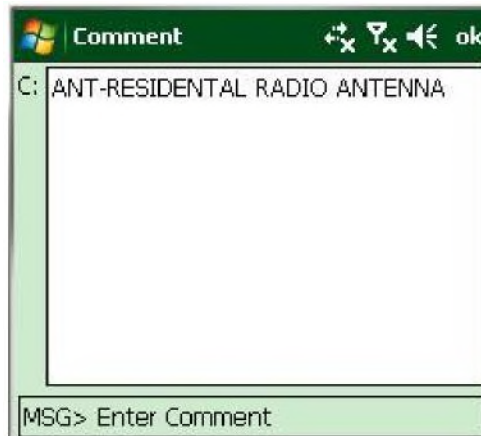
---



- When in the *CEFB Input Mode* and changes have been made to the Feature Code, point prefix or any of the parameters (Point Geometry, DTM Attribute, Zone, and Chain Name), the *Attribute Update Prompt* dialog box will display prior to returning to the original screen.



7. Select the applicable **action** to be performed and then click the **OK** button to return to the original screen. See the *Attribute Update Prompt* section below for further explanation of its functions.
8. When the feature code exceeds the 18 character space limitation, the feature code descriptor will be placed in the *Comments* screen. Click the **C:** button to call up the *Comments* screen:



## Program Logic – Point and Feature Code entries

### When a point is saved: (EFB or CEFB input mode)

- 1) The feature code field is checked to see if a value was entered.
- 2) If a value was entered the Feature Code to Prefix relationship is added to the internal FeatureDefaultsList table in the database along with a time stamp.
- 3) If no value was entered then it is ignored and the relationship to the other attributes is ignored as well.

### When a point is entered: (EFB or CEFB input mode)

- 1) The Feature Defaults List is searched for the most recent use of the point prefix.
- 2) If not found the Setting XML feature codes are searched for a point prefix match.
- 3) Finally if not found the defaults P G 1 are used.

### When a feature is entered: (CEFB input mode only)

- 1) The Feature Defaults List is searched for a feature match.
- 2) If not found the Setting XML feature codes are searched for an alias match.
- 3) If not found the Setting XML feature codes are searched for a feature match.
- 4) Finally if not found the defaults P G 1 are used.

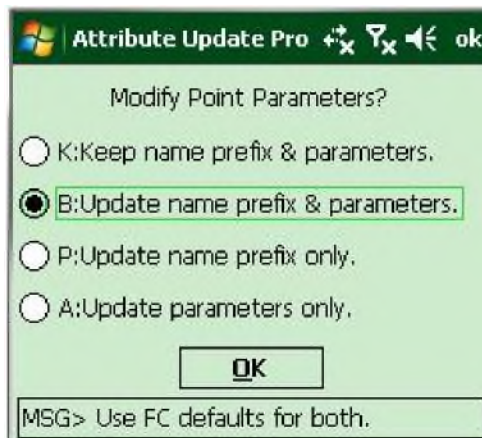
## Attribute Update Prompt (CEFB Input Mode only):

When in the *CEFB Input mode*, the user has options available on how the feature code naming and data will be stored. The *Attribute Update Prompt* dialog box will appear when a different value is entered into any of the following fields: *Feature code*, *Point Geometry*, *DTM Attribute*, *Zone*, or *Chain Name*. The *Attribute Update Prompt* dialog box will prompt the user to choose whether to overwrite any, part of or *all* of the previously entered information.

Name refers to Point Prefix (**N:**) and parameters refer to *Point Geometry*, *DTM Attribute*, *Zone*, and *Chain Name*.

The first time a Feature code is used the program will use the Settings XML values for the Point prefix and the parameters for Point Geometry, Units, Alias, Prompt, DTM Attribute, Zone, and Chain Name.

A Feature Code default table is generated with each new SDF file (project file). As feature codes are entered and saved, the Point name and the parameters are stored in this table.



### **K: Keep name prefix & parameters**

This selection maintains the values that were entered in the previous dialog box (Setup, HVD-Obs, SOR-Obs, Chains or Taping). The point prefix or any of the parameters listed above WILL NOT be overwritten. This is the default action.

### **B: Update name prefix & parameters**

This selection changes the values that were entered in the previous dialog box (Setup, HVD-Obs, SOR-Obs, Chains or Taping). The point prefix or any of the parameters listed above WILL be overwritten by the new values entered. The program compares the values to the Feature Code defaults table. If the Feature code is found, it uses the Point prefix and parameter values in the table to overwrite the values entered in the previous dialog box.

If the Feature code is not yet in the table, the program uses the Feature code Point prefix and parameter values from the *EFB\_Settings.xml* to overwrite the values entered in the previous dialog box. When the record (a new point) is saved to the EFB database, it will also write the new Feature code point prefix and parameters to the Feature code defaults table. The next time that same feature code is used, it uses the values from this table, if B: Update name prefix and parameters is selected. If different values are required, the changes will need to be made in the Observation (or chains or taping) dialog box.

### **P: Update name prefix only**

This selection changes the Point prefix value that was entered in the previous dialog box (Setup, HVD-Obs, SOR-Obs, Chains or Taping). The point prefix WILL be overwritten. The program compares the value to the Feature Code defaults table. If the Feature code is found it uses the Point prefix value in the table to overwrite the value entered in the previous dialog box.


If the Feature code is not yet in the table, the program uses the Feature code Point prefix value from the *EFB\_Settings.xml* to overwrite the value entered in the previous dialog box. When the record (a new point) is saved to the EFB database, it will also write the new Feature code point

prefix to the Feature code defaults table. The next time that same feature code is used, it will use the values from this table, if P: Update name prefix only is selected. If different values are required the changes will need to be made in the HVD-Obs, SOR-Obs, Chains or Taping dialog box.

**A: Update parameters only**

This selection changes the parameter values that were entered in the previous dialog box (Setup, HVD-Obs, SOR-Obs, Chains or Taping). The parameter values WILL be overwritten. The program compares the parameter values to the Feature Code defaults table. If the Feature code is found it uses the parameter values in the table to overwrite the values entered in the previous dialog box.

If the Feature code is not yet in the table, the program uses the Feature code parameter values from the *EFB\_Settings.xml* to overwrite the values entered in the previous dialog box. When the record (a new point) is saved to the EFB database, it will also write the new Feature code parameter values to the Feature code defaults table. The next time that same feature code is used, it will use the values from this table, if A: Update parameters only is selected. If different values are required the changes will need to be made in the HVD-Obs, SOR-Obs, Chains or Taping dialog box.

	<b>Hint</b>	In a saved point record, if a new feature code is entered and <i>K: Keep name prefix &amp; parameters</i> is selected, a point is <i>modified</i> and will need to <i>type 6</i> to modify & save. If <i>B: Update name prefix &amp; parameters</i> is selected, then a new point is <i>created</i> and you will need to <i>type 0</i> to save.
---	-------------	---

---

In all cases, if the Feature Code does not exist in the *EFB\_Settings.xml* an error message will appear in the MSG> text box. EX: Bad Feature Code.

---

# Header (Switchboard, Option - 1)

The *Header* contains the opening pieces of information that go in the EFB file. The Header is the first record in any EFB segment. The Header only occurs once per segment. The Header contains descriptive information about the *nature* of the segment itself. The data items to be placed in the Header include a toggle for units of measurements between English and Metric, a *segment name*, and space for a short paragraph of *remarks*. There is only one Header in an EFB segment. The only data required for the Header is a segment name and the units. The segment name in the Header allows a longer, more descriptive name than the *filename* given during the Windows Save As function.

The header data screen:



Screen title: **Header**

Units of linear measurements



The **U:** field toggle will allow the choice between English or Metric units. Once this header screen has been recorded the units will have been selected for the project and cannot be changed from within the software. The decimal precision of distance inputs will be to the millimeter (.001) for metric units. If English units are selected the precision is set to one hundredth of a foot (.01).



Segment name field

The segment name may be up to 18 characters in length.

Example: I-75 COLUMBIA CO.

Remarks field about nature of the segment

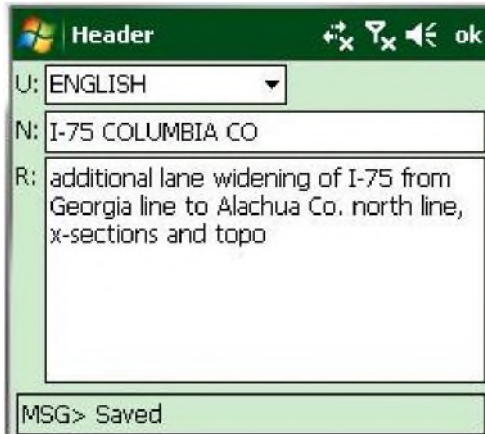
Remarks are used to input a brief description about the work to be performed. The surveyor has space to input up to 98 characters (including spaces) for the description.



Example: Additional lane widening of I-75 from Georgia line to Alachua Co. North line, x-sections and topo


When typing in remarks, or any long listings of data, EFB will automatically wrap the text to the next line to keep all of the text visible on the screen. The field will stop accepting input once it has reached the character limit.

The actual screen the surveyor sees after input of the above example is:



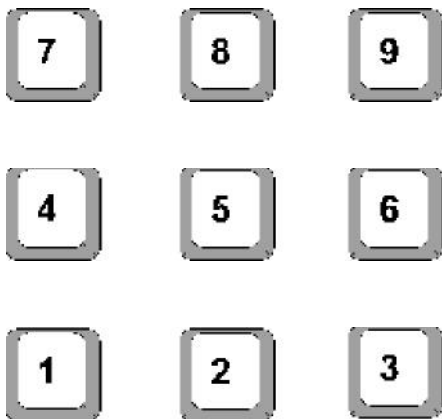
Information is placed in the screen by manipulation of the cursor. This is done by pressing the *prompt* key on the keyboard representing the letter in the field to be edited or by toggling. The cursor is then placed in the selected data field for editing (most cases) or by toggling between the embedded values or strings, as in the case of the **U:** field, and a corresponding message will appear in the message field explaining the action that has been taken. The field may now be edited.

**MSG>** Message block for communicating status of operations to the surveyor.

When editing is complete, pressing  returns the cursor to *standby status* in the message field. With the exception of data *toggles* (described later), which change the contents of data fields without a key-in, editing of most data fields takes place in the manner just described.

When editing any data screen, the information input remains in the *screen manager* until the user decides to record the information to the EFB database. This allows the user to correct mistakes prior to recording the data permanently to the EFB database.

### Functions Available: Header



PAGE BACK PAGE FWRD

**U**  
RECORD



**RECORD** Records the current data screen to the segment file on the field computer. Data is written to the EFB database on the device drive.



**PAGE BACK** Displays each screen in reverse order of recording.



**PAGE FWRD** Displays each screen in order of recording.

## **Record**

The *Record* function is used to record the contents of a data screen to the EFB database .SDF file. When selected, the Record function checks the contents of the data screen to ensure that the required data is input in the right format. If an error is detected, the surveyor is warned with the appropriate message printed in the message field. When an attempt to record the data screen is successful, the message "Saved" is displayed in the message field, indicating the Record function was successful in its operation.



The Record function will ALWAYS be located on  key, for all data screens.

## **Page Backward/Page Forward**

Two functions exist that allow the user to effectively "turn the page" of the field book to any of the previously recorded data screens. The **Page Backward**<sup>1</sup> and **Page Forward**<sup>2</sup> will display with each press of the function key the next/previously recorded screen in sequence depending on whether the direction is forward or back.

The functions will also "revolve" past the end or beginning of the recorded data as the following illustration depicts:

PAGE BACKWARD

(beginning of file)

(end of file)

PAGE FORWARD

This allows the user to page continuously in either direction where the paging does not end with either the beginning or the ending of the recorded data, similar to a Rolodex.

## Calibration - (Switchboard, Option - 2)

The *Calibration* contains information about the environment, the field crew and the instrument being used. The Calibration is both useful and necessary to maintain a complete record of the condition of the environment during the survey. Since the Calibration also contains field crew information, its data can be used to determine who held what responsibility during the survey. The most important function of the Calibration pertains to the surveying instrument. It is at the time of Calibration that the user selects in EFB the electronic total station (if any). EFB will then set up the proper communications protocol for the selected instrument. Systematic error testing is also performed for the instrument at the Calibration, if the testing option is chosen.

A Calibration is a required record after the Header, and must also be performed each time the user re-activates a segment. The axis test of the instrument should also be performed if the observer (instrument man) changes, or if a duration of time passes since the previous test, or if significant environmental change has taken place. A new Calibration is required if the type of instrument has changed.

The systematic error testing implemented in the Calibration is performed for two types of instruments, an angle measuring device (theodolite, total station) and the level.

The Calibration Screen:

The screenshot shows a software interface for a 'Calibration' screen. At the top, there is a green header bar with the title 'Calibration' and several small icons. Below the header, there are several input fields arranged in a grid-like fashion. The fields are labeled as follows: a large empty box with a 'C' button, T: [input], P: [input], W: [input], O: [input], B: [dropdown], N: [input], M: [dropdown], R: [input], E: [input], D: [input], S: [input], I: [input], and A/L: [dropdown]. At the bottom of the screen, there is a status bar that reads 'MSG> Select Field'.

Screen title: **Calibration**



Temperature, input in degrees Fahrenheit.

Atmospheric pressure, input in inches of mercury (Hg).

Weather code (see Weather Code Table).



Observers initials.



Note keeper's initials.



R	Rodman(s) initials (initial groups separated by a space example: R:HEC BMD CBM).
B	Brand of instrument (a <i>toggle between supported brands</i> ).
M	Model of instrument (a <i>toggle between supported models of a brand</i> ).
I	Identification (serial number) of instrument.
E	EDM absolute error value in millimeters (manufacturer's specifications).
D	Distance dependent error in parts per million (ppm - per manufacturer's specifications).
S	Stadia constant for instruments equipped with stadia hairs in their optics (manufacturer's specifications).
A	<div style="display: inline-block; border-left: 1px solid black; border-top: 1px solid black; width: 20px; height: 20px; margin-right: 5px;"></div> Axis testing (collimation, inclination and pointing) / Level testing (peg test) selector - branches to the type of test screen selected. See below and the Test section (Switchboard option -3).
C	Comments button branches to comments screen pertaining to the Calibration.

## Axis Testing

The *Axis Test* in the Calibration is actually three tests designed to measure two systematic errors and one random error that occur when using an angle measuring instrument. The axis test answers the following questions:

- How well does an observer/instrument point at a target? [a random error]
- What is the horizontal axis collimation error of the instrument? [a systematic error]
- What is the vertical axis collimation error of the instrument? [a systematic error]

It is important to note that a test for the observer's ability to read a vernier or scale for manual instruments is not tested! EFB was designed around electronic reading total station operations, where an observer's reading error is not a consideration, however reading error of optical scales and micrometers is an important contributor of error in surveying operations and should be recognized as such.

The Axis Test provides data to the processing program (**EFBP.EXE**) that allow for the appropriate reduction/correction of the raw observations for the determined error values. The results are also displayed to the user in a remarks screen on EFB and stored as a remarks record in the file

immediately following the calibration record, so the user can determine if the instrument is in sufficient adjustment for continued field use.

The Axis Test is comprised of pointing at a fixed target several times in the face 1 (*direct*) telescope orientation, then transiting the telescope and pointing at the same target an equal number of times in the face 2 (*reverse*) telescope orientation. The values of the horizontal and zenith angles to the target are used to perform the computations for the three errors described above.

## Level Testing


The *Level Test* (sometimes referred to as a *Peg Test*) is designed to test if the line-of-sight of a level is truly normal to gravity. EFB prompts the user for rod readings on two rods (A & B) from different setups (SET1 & SET2) of the level, and then instructs him to adjust the cross hairs, if necessary. From Setup 1 (SET1) centered between the two rod locations (A and B) make readings on rods at locations A and B. Then move the level to Setup 2 (SET2) near the rod location A. Again read and enter the two rod location readings from setup 2. The amount to adjust the level cross hair so that the reading at rod B is correct is computed.), and then instructs him to adjust the cross hairs if necessary.

Both axis and level testing are companion to the Calibration and the option to perform the test is selected from within the Calibration data screen.



## Toggles

A *toggle* is used to reduce the number of key-strokes to input the proper data, when the number of options for that data is limited.

Pressing a prompt letter key repeatedly (which is a toggle) will cycle the available data options in

the data field. This process may be repeated until the desired option appears. Pressing  is not required (because the cursor is in standby status, remaining in the message field) after using a toggle.

When the focus (cursor) is in the data field, where there are a specific set of data as in a dropdown, selection may be made by either typing the highlighted character or the first character

of a full-word option <OR> by using the  and  arrows to toggle through the set of available selections. To accept the selection and return to the message field you must press



You may also press  to move to the next data field.

The first toggle we will discuss is found in EFB for the selection of the *Brand* and *Model* of instrument used for the surveying observations. Since EFB supports a limited number of total stations, a toggle is used to aid the surveyor with the selection of the instrument.

### **B**

In Calibration, the **B** key will toggle through all the available supported Brands of electronic total station or levels.

### **B**

When pressing the **B** key, other data fields are also changing, namely the model field, EDM standard error field, and the EDM parts-per-million error field. When a particular brand is selected, model is also a toggle and is limited to the valid instruments of the selected brand. The EDM standard error, parts-per-million error, and stadia factor are automatically input for all instruments currently supported in EFB.

## Weather

Weather information is placed into the Calibration screen by editing the fields designated.

The *Temperature* (T:) is input in degrees Fahrenheit.

The *Pressure* (P:) is input in inches of mercury.

The NGS *Weather Code* (W:) is a six digit number, where any single digit may only be a 0, 1, or 2.

EFB automatically checks the ranges of the data for the weather items. For the NGS weather code, EFB enforces the appropriate digits allowed for input. The weather code table is used to determine the weather code digits required for NGS blue booking of data. Note the NGS weather code does not automatically update to correct values if the temperature (**T:**) is out of range.

WEATHER CODE TABLE:

		CODE DIGITS		
	ITEM	0	1	2
1	PROBLEM INDICATOR	No Problem Encountered	Not Affecting Observations	Possibly Affecting Observations
2	VISIBILITY INDICATOR	Good (Over 15 mi.)	Fair (7 mi. to 15 mi.)	Poor (Under 7 mi.)
3	TEMPERATURE INDICATOR	Normal Range (32°F to 80°F)	Hot (Over 80°F)	Cold (Below 32°F)
4	CLOUD COVER INDICATOR	Clear (Below 20%)	Partial (20% to 70%)	Overcast (Over 70%)
5	WIND INDICATOR	Calm (Under 5 mph)	Moderate (5 mph to 15 mph)	Strong (Over 15 mph)
6	HUMIDITY INDICATOR	Dry (Below 30%)	Normal (30% to 70%)	Muggy (Over 70%)


The weather code has six digits, each digit being a **0**, **1** or **2**. The sequence of digits is read from the left side of the table, where the first digit in the code corresponds to the problem indicator, and the second to the visibility, and so on.

An example weather code for a day where:

- The weather is possibly affecting observations (2)
- fair visibility (1)
- very cold temperature (18 degrees F.) (2)
- clear sky (0)
- strong winds (2)
- moderate humidity (1)

Would be input as: 212021

The Calibration screen is a data screen, therefore the use of the screen manager data input into the remaining data fields is the same as discussed earlier. The data fields are entered by pressing the appropriate key corresponding to the prompt letter.

When editing of the field is complete, pressing  places the cursor into standby status in the message field. A message in the message field will indicate the action taken when a prompt letter key is pressed.