Software development for automatic steering and implement control of agricultural equipment utilizing the global positioning system and a geographic information system
by Brian Thomas Mosdal

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Agricultural Engineering
Montana State University

Abstract:
This thesis is a report of the software development for automatic steering and implement control of agricultural equipment. The software will use the Global Positioning System for position location and a Geographic Information System to define the input requirements for the machine control functions.

There are several modules contained in the software. These are configuration modules for the geometry of the tractor and hitch, implement, G.P.S. antenna mounting, and field / G.I.S. grid cell interaction. Another module is used to test a field target travel path for continuity and tangency of its travel path segments, before it is used in the main part of the program. The main section of the software deals primarily with determining the steering angle required for a tractor and implement to follow the predrawn, target travel path through the field. This target travel path is constructed using a CADD program or any other method that can produce a DXF output file of the travel path entities.

Provisions have been made for the future development of the software and for incorporating a G.I.S. database, as well as other possibilities. This software development should provide a good basis for future development of automatically controlled agricultural equipment.

SOFTWARE DEVELOPMENT FOR AUTOMATIC STEERING AND IMPLEMENT CONTROL OF AGRICULTURAL EQUIPMENT UTILIZING THE GLOBAL POSITIONING SYSTEM AND A GEOGRAPHIC INFORMATION SYSTEM

by

Brian Thomas Mosdal

A thesis submitted in partial fulfillment
of the requirements for the degree

of

Master of Science

in

Agricultural Engineering

MONTANA STATE UNIVERSITY
Bozeman, Montana

May 1994

APPROVAL

of a thesis submitted by

Brian Thomas Mosdal

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the College of Graduate Studies.

July 29, 1994 — Date

William E. Larsen — Chairperson, Graduate Committee

Approved for the Major Department

29 July 1994 — Date

Head, Major Department

Approved for the College of Graduate Studies

8/23/94 — Date

Graduate Dean

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a master's degree at Montana State University, I agree that the Library shall make it available to borrowers under rules of the Library.

If I have indicated my intention to copyright this thesis by including a copyright notice page, copying is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for permission for extended quotation from or reproduction of this thesis in whole or in parts may be granted only by the copyright holder.

Signature Brian T. Mosdal

Date July 29, 1994

TABLE OF CONTENTS

TABLE OF CONTENTS (CONTINUED)

LIST OF FIGURES

ABSTRACT

This thesis is a report of the software development for automatic steering and implement control of agricultural equipment. The software will use the Global Positioning System for position location and a Geographic Information System to define the input requirements for the machine control functions.

There are several modules contained in the software. These are configuration modules for the geometry of the tractor and hitch, implement, G.P.S. antenna mounting, and field / G.I.S. grid cell interaction. Another module is used to test a field target travel path for continuity and tangency of its travel path segments, before it is used in the main part of the program. The main section of the software deals primarily with determining the steering angle required for a tractor and implement to follow the predrawn, target travel path through the field. This target travel path is constructed using a CADD program or any other method that can produce a DXF output file of the travel path entities.

Provisions have been made for the future development of the software and for incorporating a G.I.S. database, as well as other possibilities. This software development should provide a good basis for future development of automatically controlled agricultural equipment.

CHAPTER 1

INTRODUCTION

Project Overview

Another revolution in farming is taking place — precision farming!

> What is being called 'precision farming,' harnesses such recent space-age developments as global positioning satellites, variable-rate controllers on application machinery, real-time yield monitors, and crop sensors, and powerful computer software to make the science of farming vastly more scientific than it is today. (Gerstner 1994)

This revolution in farming is due primarily to the great advancements in digital computer technology made during the last ten to twenty years. There are many advantages to using high precision farming techniques versus the traditional, manual methods of farming. Some of the major advantages of precision farming will be discussed within this thesis.

Software development for automatic steering and implement control of agricultural machinery utilizing the Global Positioning System (G.P.S.) and a Geographic Information System (G.I.S.) is the subject of this thesis. Major productivity and efficiency gains can be accomplished with an autonomous farm machine – the *ultimate* goal of machine positioning, guidance and control. The purpose of this software development project is to provide a solid framework for farm machine guidance and implement control, not an autonomous control system per se, but some important components of such a system. This software can be adapted, modified and refined to handle the required inputs and outputs of an agricultural machinery control system.

Alternative Navigation Systems

Several different equipment navigation and guidance systems have been proposed over the years for use in agricultural applications. These systems include, but are not limited to the following methods – laser detection units, LORAN-C, radar beacons, inertial, radio frequency, odometry, computer vision, and "dead reckoning", as well as the differential, real-time (kinematic) G.P.S. system. The G.P.S. based system appears to have the best potential, with respect to precision, accuracy, standardization, reliability, and future feasibility, for use in precision farming.

Role of G.P.S. and G.I.S.

Very high precision location determination is possible when using a G.P.S. receiver in a differential correction mode. When the G.P.S. signal is corrected in real-time, it is possible to accurately and precisely determine the kinematics of a ground-based vehicle operating at typical farm field travel speeds. This positioning information can be used for navigating the tractor and machinery through a field. The G.P.S. system was developed by the United States Department of Defense and consists of an array of 24 satellites in orbit around the earth, (currently with 2 backup satellites). The system can now be used on a 24 hour per day basis throughout the world.

Field scale G.I.S. databases can be used to store and retrieve data pertinent to a particular field management grid cell. Grid cells can be made to any size, *within* geometrically, technically and economically feasible limits. A number of G.I.S. databases, with different grid sizes, depending on the data type, would be constructed using various sensors to acquire the required data about a field. Such data would include attributes of the soil, terrain and crop condition. These databases would then be combined to create a single field management implement control application grid.

A potential limit is to make the field management grid cells for a particular field operation, *at least* two times as wide, in both directions, as the width of the particular implement used. This ensures that the machine will pass through each cell at least twice, so that a good representation of that particular cell of the field can be acquired. If the cell was only one implement width wide, for example, the implement center (where the G.P.S. receiver is ideally located) could possibly miss that particular G.I.S. grid cell. This would cause the automatic application of the implement control to be based on the neighboring cells, rather than the correct cell.

Advantages of a G.P.S. Based Guidance System

The benefits to be gained from the use of a G.P.S. based guidance system are many. This system would provide a level of machine control, over an extended period of time, that is not possible by even an experienced human operator. A desired performance level for the guidance system should keep the machinery within 5 centimeters of the planned travel path at all times. This software has not been tested to determine if this can be accomplished, since there was no G.P.S. hardware available to the author to test at the time of writing. Computer simulation (Erbach, Choi and Noh 1991) has shown that a tractor *can* be controlled to within a 5 centimeter position error using one-step-ahead deterministic prediction models. This level of precision would be higher than a human operator could sustain over an entire day, especially at "higher" field speeds.

By using the G.P.S. system for navigation, the machinery could automatically be positioned on the same travel path every time it goes through the field. This level of precision would minimize skips and overlaps when travelling through the field.

A major advantage of this type of guidance and control system is that precision farming can contribute to reduced groundwater pollution. The information obtained from a G.I.S. database can be used to tailor the rate of application of fertilizer and other chemical inputs to the specific requirements of a particular "cell" within the field. Only the amounts of material needed by the soil in a specific cell will be applied to the field, and major skips and / or overlaps will be avoided. This would reduce the chance of groundwater contamination by controlling the application of fertilizer or other chemicals, such as herbicides and pesticides.

The input **and** output parameters stored in the G.I.S. database will allow the user to document what has been done in any part of any field, as well as when it was done and why it was done. All of the information that can be measured, within current technological, economic and other practical limits, *can* be available via the input and output G.I.S. database files. The G.I.S. database(s) could be used to keep track of weeds, insects etc., with the information about a certain problem being available to the user long after *visible* signs of the problem have been eliminated, even in later years (Larsen, Tyler, Nielsen 1991).

Wheel track soil compaction is of critical importance in some areas of the country, and the location of wheel tracks can be controlled by precision farming. By using G.P.S. to navigate through a field, the machinery can be programmed to automatically follow the same travel paths through the field each and every time. This would create a permanent roadway and keep the wheel tracks off the rest of the field. Soil compaction due to tire tracks would be minimized, with no extra effort by the machine operator.

The ultimate goal of precision farming is to obtain better control on all farming practices with the goal of reducing inputs of time and money, therefore maximizing productivity and efficiency while also minimizing negative environmental impact. Precision farming will apply only the required amounts of material at the correct location, and in a timely manner. This will result in a savings over the conventional, broadcast farming methods that used manual steering and implement control.

CHAPTER 2

GUIDANCE SYSTEM

Overview

Hardware and software for a guidance system, as outlined in the following sections, are integrated into a framework that, when fully developed, will provide for automatic steering and implement control. The current software is not capable of being used immediately for machine control, as there is no specific code provided where a hardware input or output is required. This can be developed later, when the hardware is available for coding, testing or use.

Hardware Requirements

A number of hardware items must be combined to form a precision farming system. The main hardware consists of a prime mover (tractor), an implement, a portable differentially corrected real-time G.P.S. receiver (with a base station, to provide the differential correction capabilities), and a portable computer system (with its peripherals – monitor and keyboard). Other required components include a steering angle transducer, a steering correction mechanism (servo or step motor), a hitch "bump stop" sensor to indicate that the hitch is at full swing (if equipped with a swinging drawbar or three-point hitch), and the G.P.S. mast angle indicators. Also required are the necessary communication cables, A/D or

D/A converters and other materials required to connect the hardware components and subsystems to the computer system. A more detailed description of the interaction between the hardware and the software is described in the software module sections of this paper.

Software Overview

The software developed consists of several major sections, or modules, of computer code written in the (ANSI) 'C' language; Borland Turbo C, version 3.0 (DOS) was used in the software development. The main function modules of this computer program are:

TRACTOR

IMPLEMENT / G.P.S.

FIELD STATUS

FIELD TEST

TRAVEL PATH DISPLAY

FARM FIELD

There are also support modules to these main modules, these are:

G.P.S. MAIN

GENERAL INPUT

GENERAL ERROR

G.P.S. HEADER

The actual names of these modules (in the computer program) are abbreviated to the DOS name limit of eight characters.

The desired travel path through a field is defined as the target travel path, and the navigation system is designed to follow the target travel path through a field. The target travel path is constructed of lines and arcs, and has the potential to use points along this path as control points for future program development. This continuous line method is used instead of a series of points with travel from one point to the next. It is easier to construct the travel path from line(s) and arc(s); point(s) can be included along these path segments, if needed. This type of a travel path construction also makes it easier to calculate the turning radii for the path segments. A straight line has an infinite turning radius (with a steering angle of zero), and the turning radius, when the machinery is on an arc segment of the travel path, is constant (with a constant steering angle). Steering angles would be constant only when there is no travel path error, i.e. the machinery followed the travel path perfectly. The method of travel path construction is described in more detail later in the report and suggestions are made for future development and improvement. The program software overview is as follows.

The G.P.S. MAIN module is the main program menu system. It contains the sign on and sign off messages and it sets up the computer display parameters. The G.P.S. HEADER module contains all of the function prototypes, global definitions, type definitions and constants used in this program. The standard libraries required in the program are also called in the header section.

The GENERAL INPUT module contains all program file handling routines for writing to and/or reading configuration files. The appropriate file name extensions are appended to the user specified file name when a file is opened for writing or reading. Other functions, such as the user prompt "Press the <Space Bar> to continue program" and double precision and integer input values are also contained in this module.

The GENERAL ERROR module contains the warnings and errors that are used for most of the program. These error messages are used primarily for the Drawing Exchange (or Interchange) Format (DXF) to Travel Path Data (TPD) file conversion module (TEST FIELD) and for the tractor configuration input module (TRACTOR). If any error is made by the user during the creation of either type of file, this module warns the user of the problem, and will not save the file until the errors have been corrected.

The TRACTOR, IMPLEMENT and FIELD STATUS modules of the program are where the tractor, implement and field parameter configuration files (respectively) can be created, modified or viewed. These configuration files have the file name extensions of .TRC, .IMP and .FSC respectively.

The FIELD TEST module of the program is where a CADD drawing DXF output file is tested for validity and logic and, upon success, converted to a TPD file for use in the FARM FIELD section of the program. The field target travel path is also displayed for the user in the field test module. A separate module called TRAVEL PATH, is used to display a previously tested target travel path.

The FARM FIELD module of the program is the main module of the program in that it is where the inputs of the entire system are brought together, processed, and outputs to the system are produced. A more detailed description of what is done within each of these modules is given in chapter 3.

CHAPTER 3

DESCRIPTION OF SOFTWARE PROGRAM MODULES

Main G.P.S. Module

The module, GPS1.C is the **main** program function. The C code listing for this module is contained in appendix A. The primary function of this module is to provide the main program menu to the user. The main menu prompts the user to enter the operations they want to perform. Their options are to create, modify or view a tractor, implement or field parameter status configuration file. The user can also test a proposed field travel path, created from a CADD DXF output file, display a target travel path or run the machinery through the field according to the required input software file specifications. The video display colors are set and sign on / sign off messages are displayed in this module as well.

G.P.S. Header Module

The module, GPSHEADE.H includes all standard information that is required for the program to function. The C code listing for this module is contained in appendix B. This module includes the standard program libraries, all function prototypes, all data type definitions and all global constants used in this program.

The global constants used by this program are as follows:

The constant pi (3.14159265359) is used in calculations to convert degrees to radians and vice-versa. The constant Standard_Time_Delay (2000 milliseconds) is used to prevent certain output screens from being cleared immediately. The constant Zero (0.0) is used for clarity, since this "constant" could possibly be changed to a "small" number when used as a tolerance variable in the program. The constant GPS_Update_Interval (set to 0.25 seconds) could be changed to accommodate the position update interval of a particular type of G.P.S. receiver. The constant Dist_Tlrnc (set to 0.01 meters) is used as an error tolerance when calculating distances used in this program. The constant Travel_Path_Err_Tol (set to 0.5 meters) is the maximum distance that implement can be physically separated from the starting point of the field travel path and be considered as being "initialized" to the starting point of the field travel path. The constant Width_Tol (set to 0.05 meters) is the tolerance on the travel path to account for the "weaving" of the implement, about the travel path, as it is being pulled through the field. The constant Max_Impl_Width (set to 50.0 meters) is used as an error check on the user's input of the width of an implement. The constant Max_Impl_Hitch_Length (set to 25.0 meters) is also used as an error check on the user's input to the hitch length.

General Input Module

The module, GENINPUT.C is the program function where file configuration input and output is performed. The C code listing for this module is contained in appendix C. Whenever a configuration file(s) (as described in the tractor, implement and field status sections) is created, saved or modified, the general input module function, Open_File, appends the appropriate file name extension to the user specified file name. The correct file is then opened for writing or reading, depending on the user's specification. After the information has been read from or written to the file, the file is closed in the function in which it was created.

Other functions of the general input module are to provide a double precision input function, named Double_Precision, and an integer number input function, named Integer_Input. A small function, Continue_Program, that pauses the program and waits for the user to press the space bar before the program will continue is also provided in this module.

General Error Module

The module, GENERROR.C is where all warnings and errors pertaining to the creation of a TPD file from a DXF file, and the tractor configuration creation process, are handled. The functions are called General_Error and Tractor_Configuration_Logic. The C code listing for this module is contained in appendix D. The incoming codes can either pertain to an error, a warning or a notice to the user. These codes are summarized as follows:

If the incoming error code is a 100 level code, an error has occurred in the DXF to TPD file creation process. An error can occur due to a number of conditions. These include the following:

A DXF output file was a "full" DXF output file when the DXF output file **must** be created using the entity selection mode. This means that the travel path entities have to be selected *individually*, in the order that they are to be followed in the field rather than simply exporting the drawing as an entire (full) DXF file.

The DXF output file is a null file, in other words, no entities were selected for inclusion into a travel path file.

The DXF output file contains no data.

Invalid DXF output file. The output file section of the file must start with a POINT entity type.

Invalid DXF output file. The output file contains only a start POINT, there are no travel path entities in the file; there must be at least one line or arc as a travel path to follow.

An invalid entity type has been detected; valid travel path entities consist solely of POINT(S), LINE(S), and ARC(S).

The error messages coming from the TEST FIELD module pertaining to the creation of a TPD file are due to an invalid DXF output file. If the user does not specify a proposed travel path, from the CADD DXF output file, with a POINT as the starting entity and at least one other valid travel path entity, a LINE or an ARC, the program will return an error message to the user.

If the incoming error code is a 200 level code, a warning is issued to the user during the DXF to TPD file creation process. A warning can occur due to a number of conditions. The following travel path entity / entity interfaces are checked for validity.

POINT / LINE

LINE / POINT

LINE / LINE

POINT / ARC

ARC / POINT

ARC /ARC

LINE / ARC

ARC / LINE

The warnings indicate that the travel path has been constructed invalidly or illogically. This means that either a turning radius was too small for the implement width specified in the implement configuration file (the implement turning radius must be at least the width of the implement) or that the travel paths were specified in a discontinuous order. This means that there were gaps between the consecutive travel path segments (entities). Other warnings include travel path entities not tangent to each other at their start and end points or two POINT entities in succession. The methods used to determine the validity and logic of the travel path are explained in more detail in the TESTFIEL.C module section of this report.

Upon receiving an error message, the program will halt and return control to the user. If a warning occurs, the user is notified of the problem and the program continues until the entire proposed travel path is processed. The DXF output file is not converted to a TPD file if there are warnings or errors present. The user must first correct the problem(s) before attempting to continue the program.

The Tractor_Configuration_Logic function of this program is designed to catch any logical errors in the creation of a tractor configuration file. If there are any logical errors, the error trapping function will alert the user to the problem. There are many logic problems possible when specifying a tractor configuration file. These errors include, but are not limited to, the following items, which are checked in the Tractor_Configuration_Logic function of the program.

The error messages for one specific type of tractor configuration, a front wheel steering tractor with a swinging drawbar, are now detailed. A swinging drawbar cannot be located at the front or center of the tractor. The drawbar swing point cannot be located ahead of or behind the front axle of the tractor (it is allowed ahead of the rear axle on all tractor configurations). The tractor's drawbar swing point must be provided for tractors configured with swinging drawbars and three-point hitches. Note that the swing point for a three-point hitch is the virtual point about which the hitch would pivot if free to swing about that point.

The preceding error messages correspond to one possible type of tractor and hitch type configuration. The user cannot save a tractor configuration file until all error(s) have been corrected. This is due to the fact that the input configuration is looped through the tractor modification function until the user has corrected the problem. Similar errors messages dealing with specific configuration errors can be seen in appendix D.

Tractor Configuration Module

The following explanation describes what the TRACTOR.C program module does and explains each of the functions within the program module. In essence, the module allows for a complete configuration of the geometry of a tractor to be developed, saved, modified and reviewed by the program user.

This is the module, where the user can create a new tractor configuration file, modify an existing file or view an existing file (to check its specifications). There are seven tractor configuration parameters. These parameters are steering type, hitch type, hitch swing point (where applicable), hitch location, wheelbase, drawbar length, hitch swing distance (where applicable), maximum hitch swing angle (where applicable) and the maximum steering angle of the tractor. The geometry of a front wheel steered tractor with a swinging drawbar is presented in figure 1. Figure 2 shows the detailed geometry of the swinging drawbar. The C code listing for this module is contained in appendix E.

The Configure_Tractor function permits the user to enter their choice of creating a new file, modifying or viewing an existing tractor configuration file or exiting from this sub-menu back to the main program menu.

FIGURE 1

FRONT WHEEL STEERING TRACTOR WITH SWINGING DRAWBAR

[TOP VIEW]

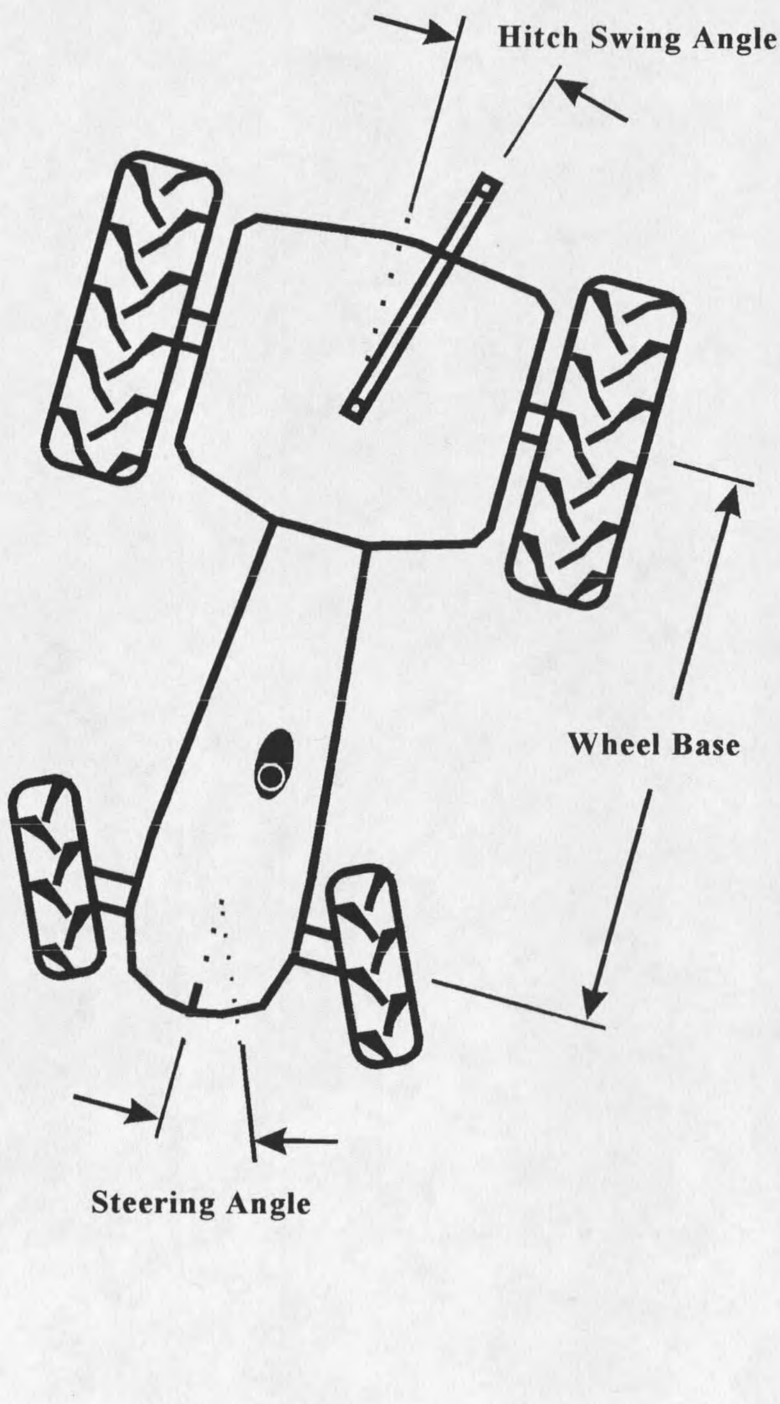
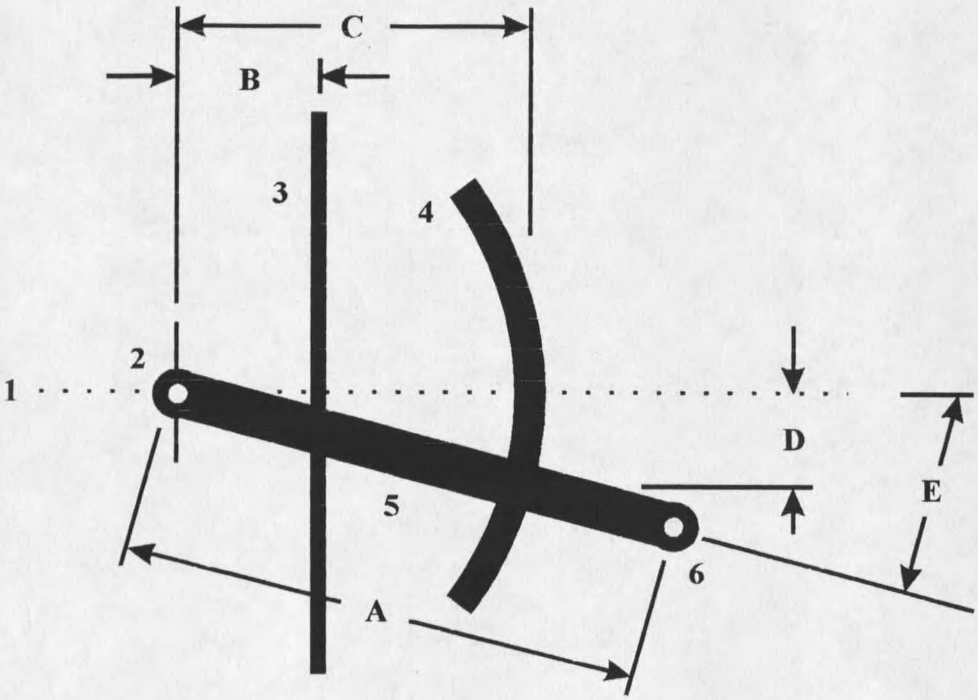


FIGURE 2

SWINGING DRAWBAR TRACTOR HITCH (DETAIL)

[TOP VIEW]



Key to Hitch Components:

1	Tractor centerline	A	Hitch length
2	Hitch swing point	B	Hitch Swing distance
3	Tractor rear axle	C	Hitch reference distance
4	Hitch support frame	D	Hitch swing reference distance
5	Swinging drawbar	E	Hitch swing angle
6	Drawbar hitch pin hole		

If the user chooses to create a new tractor configuration, the function Create_Tractor_Configuration is called. From this function, the steering type, hitch type, hitch location, wheelbase and drawbar length input functions are called. If the hitch type is a swinging drawbar or a three-point hitch, the user is then prompted to enter the hitch swing point, hitch swing distance and the maximum hitch swing angle, otherwise the tractor hitch swing point is set to be not applicable and the hitch swing distance and maximum hitch swing angle are both set to a value of zero. The user is then prompted to input the maximum steering angle of the tractor.

After the values have been input, the new configuration parameters are sent to the Check_Tractor_Configuration function. From this function the parameters are sent to the Tractor_Configuration_Logic function as described in the GENERAL ERROR module section. Once the values for the tractor configuration are correct, they are sent to the Tractor_File_Info function. This function displays the current configuration of the tractor's parameters. The user then has the option of changing the parameter values, saving the file or returning to the previous menu.

If the user chooses to modify a tractor configuration file, the function Modify_Tractor_Configuration is called. The user is then presented with a list of the tractor configuration parameters to choose to modify or the user can return to the previous menu. If the user decides to modify a particular configuration parameter, the appropriate parameter input function is called.

If the user decides to view an existing tractor configuration file, the function Open_Tractor_Config_File is called to prompt the user to input the file name of the tractor configuration that they wish to review. Then, the Tractor_File_Info function is called to display the tractor configuration. When the user presses the space bar, the program returns to the previous menu.

A description of the tractor configuration parameter options follows this list of the tractor configuration parameters:

Steering Type

Hitch Type

Hitch Location

Wheelbase

Hitch Length

Hitch Swing Point

Hitch Swing Distance

Hitch Swing Reference Distance

Hitch Support Distance

Maximum Steering Angle

The variable, Tractor_Steering_Type input in the function Steering_Type, can be front wheel, articulated or frame, skid or controlled differential, all wheel or rear wheel steering.

The variable, Tractor_Hitch_Type input in the function Hitch_Type, can be swinging drawbar, rigid drawbar, three-point hitch or a semi-integral type of hitch.

The variable, Tractor_Hitch_Location, input in the function Hitch_Location, can be specified as being located at the front, center or rear of the tractor. Some types of hitches cannot be applied to certain sections of a tractor, as has been described in the GENERAL ERROR module section of the program description.

The variable, Tractor_Wheelbase, input in the function Wheelbase, is the wheelbase of the tractor. This measurement is from the center of the front axle to the center of the rear axle of the tractor.

The variable, Tractor_Drawbar_Length, input in the function Drawbar_Length, is the critical length of the particular type of hitch. For a rigid drawbar, this length is the distance from the center of the hitch-pin hole to the center of the rear axle. For a swinging drawbar, this length is the distance from the center of the hitch-pin hole to the center of the pivot pin where the other end of the hitch is attached to the tractor. For a three-point hitch, this length is the distance from the center of the mounting pin holes on the lower links to the (actual or virtual) pivot of the three-point hitch. For a semi-integral hitch, this length is the distance from the center of the pivoting mount points to the center of the reference axle on the tractor. The tractor's reference axle is the one which is located closest to the mounted implement.

The variable, Tractor_Hitch_Swing_Point input in the function Hitch_Swing_Point, can be, not applicable to this tractor, located ahead of the front axle, behind the front axle, ahead of the rear axle or behind the rear axle. The difference between, behind the front axle or ahead of the rear axle, is only when the hitch swing point is applied to an articulated steering type tractor. With this type of steering system, the hinge point separates the tractor into two distinct halves – front and back. With all other types of steering systems, there is no distinction between ahead of the rear axle or behind the front axle, so the default should be set appropriately relative to where the hitch is located. For example, if a front wheel steering tractor had a swinging drawbar located at the rear, the hitch swing point would be set ahead of the rear axle.

The variable, Tractor_Hitch_Swing_Distance, input in the function Hitch_Swing_Distance, is the length of the swinging distance between the center of the reference axle on the tractor and the pivoting point(s) on the hitch.

The variable, Tractor_Max_Hitch_Swing_Angle, input in the function Max_Hitch_Swing_Angle, is used for three-point and swinging drawbar tractor hitches. This angle is the angle between the centerline of the tractor and the centerline of the hitch reference member when the hitch is at its maximum swing. The hitch reference member for a swinging drawbar is the swinging drawbar itself, while the reference member for a three point hitch is either one of the lower links. The maximum hitch angle is computed by taking the arc-sine of the variable Tractor_Swing_Ref_Dist divided by the variable Tractor_Hitch_Ref_Dist. The variable Tractor_Hitch_Ref_Dist is the distance from the center of the swinging drawbar pivot point to the centerline of the hitch at the center of the swinging drawbar support frame. For the case of a three-point hitch, this distance is the (real or virtual) distance from the pivot point of the hitch to an arbitrary reference point along the length of one of the lower lift arms. The variable Tractor_Swing_Ref_Distance is the perpendicular distance between the tractor centerline and the contact point at the centerline of the swinging drawbar where it rests on its support. For the case of the three-point hitch, this distance is the perpendicular distance from the centerline of the tractor to the centerline of the lower link arm when it is at its maximum swing angle minus the same point when it is at its normal swing angle (the angle when the implement is in its normal position straight behind the tractor). This variance in the angle is "small" enough to negate any explicit travel error, any implicit error due to this method should be taken into account by the steering feedback and control.

The variable, Tractor_Max_Steering_Angle, input in the function Max_Steering_Angle, is the maximum steering angle for the particular tractor steering type. For a front or rear wheel steer tractor, this angle is the angle between the centerline of the tractor and the centerline of the steering tires when the steering is at its maximum left or right position. Caster action front wheel steering systems are handled the same as a front wheel steer tractor. For an articulated steering tractor, the maximum steering angle is the angle between the centerline of the rearward section of the tractor and the centerline of the forward section of the tractor. The maximum steering angle for an all wheel steer tractor would be the angle between the projected center-lines of the front and rear steered tires when they are steered to their maximum angles. For a skid steered tractor, the maximum steering angle is, in essence, infinite. This steering angle would have to be limited to some maximum amount for practical applications. This angle would vary depending on what type of hitch was being used. For example, any maximum steering angle (up to 180 degrees in either direction) could be used if the implement was mounted on the tractor with a three-point or semi-integral hitch. The maximum steering angle would have to be limited to approximately forty-five degrees if a rigid or swinging drawbar were used. This is to keep the turning radius of the tractor large enough to prevent interference between the tractor and the drawn implement. Similar steering angle limits would have to be placed on other tractor steering methods if the tractor could steer "sharp" enough to cause interference between the tractor and the towed implement.

The function, Open_Tractor_Config_File prompts the user for a tractor configuration file name, appends the appropriate file name extension, .TRC, and opens the file. The file is then either written to or read from depending on the file-mode that was chosen by the user.

Implement Configuration Module

The following explanation details what the IMPLEMEN.C program module does and explains what each of the functions within the program module do. In essence, the module allows for a complete configuration of the geometry of an implement and G.P.S. receiver to be developed by the program user.

The implement configuration module is quite similar to the tractor configuration module in that it accomplishes the same types of functions for the implement. This is the module, where the user can create a new implement and G.P.S. receiver configuration file, modify an existing file or view an existing file (to check its specifications). The C code listing for this module is contained in appendix F.

There are two implement and configuration parameters and five G.P.S. receiver configuration parameters. These parameters are the implement hitch length and working width, the G.P.S. receiver height, and the distance forward or rearward and left or right of the center of the effective working area of the implement. Figures 3 and 4 show the implement and G.P.S. receiver geometry respectively.

During the implement / G.P.S. receiver configuration creation process function, Configure_Implement_GPS, the user is prompted to enter his / her choice of creating, modifying or viewing an implement and G.P.S. receiver configuration file or exiting from this sub-menu back to the main program menu.

If the user decides to create a new file, the function Create_Implement_GPS_Configuration is called. This function prompts the user to input the implement hitch

FIGURE 3
IMPLEMENT GEOMETRY
[TOP VIEW]