



Everything for Filament Winding

Supplemental Training Manual  
for Unidirectional Wind Module

Version 1.1 – December 7, 2004

Contact Information:

McClellan Anderson  
300 Ross Avenue  
Schofield, WI 54476  
USA

Phone: +1 (715) 355 3006

Fax: +1 (715) 359 0900

On the Web: [www.mcclellananderson.com](http://www.mcclellananderson.com)

Email: [info@mcclellananderson.com](mailto:info@mcclellananderson.com)

© 2004 McClellan Anderson Inc., All Rights Reserved. All information contained in this document is believed accurate at the time of printing. All trademarks belong to their respective owners. **Important note:** this manual contains confidential technical information on McClellan Anderson software and control systems and is only intended for use by McClellan Anderson customers for the operation of our machines. Disclosure of this information to third parties is prohibited without express written consent from McClellan Anderson.

## Table of Contents

|      |   |    |
|------|---|----|
| 1.   | Unidirectional Wind Module .....              | 4  |
| 1.1. | Wind Parameters .....                         | 4  |
| 1.2. | Wind Motion Description .....                 | 9  |
| 1.3. | Notes and Tips for Unidirectional Winds ..... | 10 |

# 1. Unidirectional Wind Module

Beginning with Composite Designer v3.1, a new module is available which generates wind programs capable of laying fibers at the same angle in both the forward and reverse stroke of the carriage (as opposed to typical winds where the forward and reverse strokes result in complementary angles – e.g. +/-45degrees). To accomplish this, the mandrel must reverse direction at each end of the stroke. Furthermore, such winds require a means to hold fibers in place while the mandrel is reversed – typically pin rings.

The steps involved with generating such winds are broadly similar to those for other wind types – in particular they are like zero degree winds in that motion is directly generated without an intermediate motion generation screen or the use of a path file (no path file is generated because at present these require unidirectional mandrel rotation).

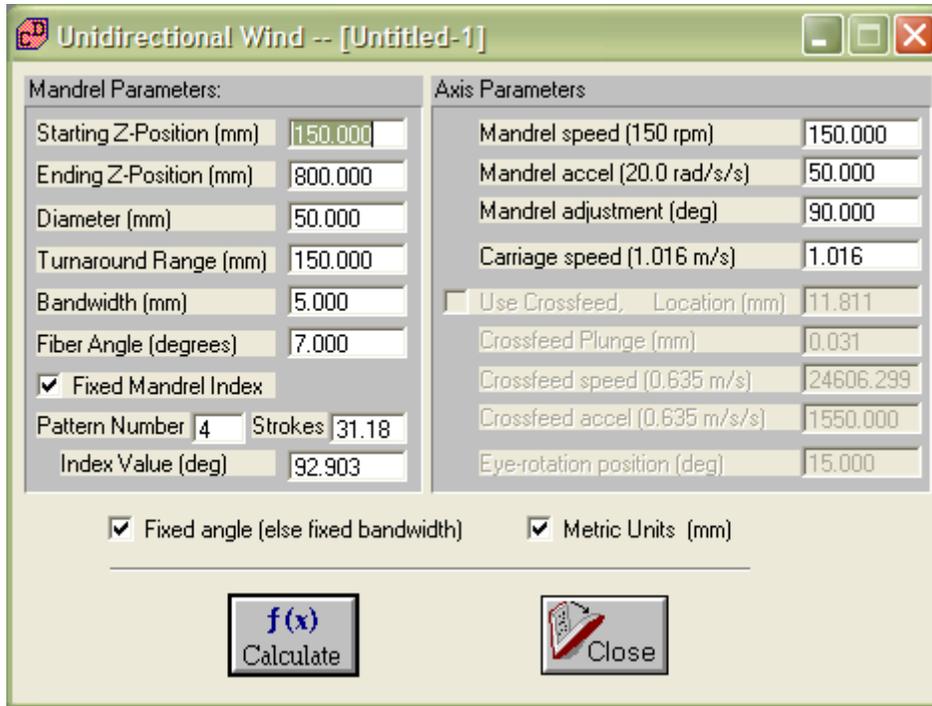
In this section, we will cover the first steps of generating a unidirectional wind. The main manual covers software installation and machine operation. Once Composite Designer (CD) is opened and properly initialized as described in the main manual, the user can begin to edit a wind.

(NOTE: The motion described in this section is still preliminary – some additional tweaking may be needed which may also result in a slightly different parameters and a slightly different wind).

## 1.1. Wind Parameters

To begin, the user selects the File | New | Unidirectional Wind. This brings up the dialog box shown in Figure 1. This dialog box allows the user to specify the various relevant parameters for the wind. Each parameter is covered in turn in the following paragraphs. In broad terms, the left side of the dialog box describes the wind while the right side describes the motion parameters of the axes involved.

The first two parameters (**Starting Z-Position** and **Ending Z-Position**) describe the locations at which the carriage will dwell and turn-around. This describes the bounds of the part – however, the useful part length will be less than this because it must include a sacrificial region beyond the pin-rings in which the fiber will turn around. With this wind, there is no additional motion generation screen – so these values describe absolute coordinates of the carriage (with the carriage's origin located such that the fiber path touches the chuck face). Note that the two entries can be reversed – if the starting position is greater than the ending position then the carriage will start nearer the tailstock and move towards the headstock on the first stroke. Also note that a unidirectional wind is stroke based – rather than circuit. If the resulting motion has an odd number of strokes then the carriage will end at the ending Z-position, if it is even then it ends back at the starting position.



**Figure 1** – The unidirectional wind dialog box.

The next parameter (**Diameter**) is fairly self-explanatory. It is needed to determine the number of strokes required for coverage at the given bandwidth.

**Turnaround Range** describes the distance over which the carriage and mandrel will accelerate/decelerate during their coordinated motion at the target angle (see section 1.2 below for a more detailed description of the motion sequence generated for a unidirectional wind). If the machine has a crossfeed, any plunge motion also takes place in this region. Typically this would be set to coincide with the location of pin rings (e.g. if the desired part runs from 0.5m to 2.5m, the pin rings are located at 0.4m and 2.6m, and the machine will reach 0.2m and 2.8m during turnaround, then the start and end Z-positions would be 200mm and 2800mm, while the turnaround would be 200mm. See section 1.3 for notes on other factors to consider – especially to generate higher speed winds.

**Bandwidth** describes the width of the fiber band. This is common to all winds. If the fixed angle box is unchecked, then this will be the actual bandwidth of the wind; otherwise the software will calculate the closest bandwidth which meets the exact fiber angle.

**Fiber Angle** is the target winding angle. Unlike other wind types, both positive and negative values can be specified – in theory values are limited to between +/-90 degrees, However, the combination of bandwidth and diameter restrict how high the angle can go (at some point the wind degenerates to a circumferential wind with a single stroke and beyond this the effective bandwidth decreases) – generally it is unlikely to be above 80 degrees. If the angle is so high that the resulting wind would have a single stroke, the software will indicate an error. If these parameters are desired, the user should generate a circumferential wind instead.

Fiber angle is defined following normal geometric orientations. This means that 0 is the mandrel axis; a positive angle results in counter-clock-wise fiber "rotation" as one moves along the mandrel axis and looks forwards (it has the same orientation as a backwards or left-handed screw-thread); a negative angle is, of course, the opposite (the same as a normal, right-handed screw-thread, clock-wise rotation when looking forwards down the part).

The **Fixed Mandrel Index** checkbox allows the programmer to alter how the mandrel indexes between strokes. If this box is not selected, then the mandrel will index in opposite directions and the index is approximately +/-180 degrees. See text below for more detailed information on how the index is calculated. If this box is selected, then the remaining three items become available and the programmer can directly specify the index amount.

The next item is the **Pattern Number**. Unlike most filament winding, the unidirectional wind does not involve any weaving over cylindrical region – so there is no pattern comparable to other winds. However, if the programmer desires the fibers to build up around the perimeter of the part in a particular fashion, then the pattern number can be for this purpose. In essence, the pattern number divides the mandrel into a number of regions and attempts to distribute fiber evenly among them. For example, a pattern of 4 attempts to find a mandrel index of around 90 degrees ( $360/4$ ).

The next parameter is **Strokes**. This is the number of strokes to complete a particular wind. The programmer can override the number calculated by the software by typing in a new number here (however, whenever the **Pattern Number** is changed, the **Strokes** value is also updated). The need for the stroke value is described in the following paragraphs. Note that the calculated stroke value is typically a non-integer amount (e.g. 13.45), this is to give the user some indication of the true number of bands required to cover the part and give some guidance when altering this value (e.g. for 12.45, a better alternate entry would be 13 rather than 11). The actual number of strokes will be rounded to the nearest integer when motion is generated.

The final parameter in this section is the **Index Value**. This is the amount the mandrel will rotate between consecutive strokes (note that the **Mandrel Adjustment** parameter described later also applies to fixed index winds). This value is calculated by the software based on **Pattern Number** and **Stroke** value, however, the user may change this value to force a particular index. The software will not validate the user's entry – it will simply carry out the motion as requested.

Whenever the user specifies a pattern number, the software attempts to establish the number of strokes and the resulting angle (**Index Value**). These are immediately updated. If the user then updates the number of **Strokes**, this also causes the **Index Value** to be updated. If no solution exists (e.g. due to incorrect values elsewhere), then no update occurs. The user may notice that the resulting index value is not the closest slot value to the desired pattern.

For example, if a set of wind parameters results in 12 strokes, and the user selects a 4 pattern, the software will calculate an index value of 150 degrees. This is because 150 degrees is the

closest index which forms a closed pattern for 12 strokes. To understand this, a closer analysis is needed:

In this example, the user might expect an index of 90 degrees. With 12 strokes, the wind must have an index value divisible by 30 degrees – ( $=360/12$ ). Each of these 30-degree sections is called a slot. Since 3 slots would equal 90 degrees exactly, it would appear that there is an exact solution. However, if this wind were executed, the following slots would be filled: 0, 3, 6, 9, 12 (=0), 15 (=3)... – in other words, the pattern is degenerate – only slots divisible by 3 are visited. This occurs whenever the desired number of slots and the number of strokes share a common divisor.

The software suggests the slot closest to the desired slot which closes properly – in this case 150 degrees or 5 slots (the pattern is: 0, 5, 10, 15 or 3, then 8, 13 or 1, 6, 11, 16 or 4, 9, 14 or 2, 7, and 12 or 0 – so on the 13<sup>th</sup> stroke we are back to the beginning and have visited all slots) – because 12 has so many divisors, 2 and 4 slot index values are also degenerate.

Because the programmer may want a value closer to the desired pattern, the number of strokes can be modified. In the above case, if the number of strokes was given as 12.45, the programmer might choose 13. Now a 4 pattern results in 83 degrees – quite close to 90 (similarly, if the number of strokes was 11.6, then a stroke value of 11 would have a closed pattern at 98 degrees).

Finally, note that pattern as applied here is not directly comparable to patterns as typically defined in filament winding. There, if a particular wind has a 4 pattern, then after 4 circuits, a fiber will be laid down next to the initial fiber. This is not necessarily the case with the methodology used here. If the user desires a specific index to generate a specific build-up sequence, then more information about how patterns can be calculated is available in the appendix of the main manual.

Turning to the right side of the dialog box – the first three parameters deal with mandrel motion. They are in turn:

**Mandrel speed** – this is the maximum speed for the mandrel. The software will not honor requests above the machine's capability – the top attained speed will depend on angle, carriage speed, and the rate at which the program is executed.

**Mandrel accel** – this is the acceleration applied to the mandrel. This value will be attained while the mandrel indexes at the ends of the part (if the part is run at 100%). Actual acceleration while laying fiber at the desired angle will depend on the angle and carriage accelerations (see section 1.2 for a more detailed break-down of the motion components which make up this wind).

**Mandrel adjustment** – this is an additional term to be added to mandrel motion during each index maneuver at the ends of the part. It is used to adjust motion such that the entire part is covered with no overlap. Because the fiber pays out on different sides of the part depending on the direction of the mandrel, there is some discrepancy in where the fiber lays relative to the "slots" used to calculate the wind. For example, on a 2 stroke part, the machine would wind to one end, then rotate the mandrel back 180 degrees, and wind back towards the start.

However, due to differences in how the fiber pays out in different directions, the actual wind may result in the two strokes being only 160 degrees apart. The programmer would add an adjustment of 20 degrees to align the two strokes to their respective slots. In general, when first running a type of part, the programmer will program and wind a full part and establish if there is any gap at the end. If so, the gap is measure (in degrees), and this is used as an adjustment term. Parts with similar characteristics (e.g. same diameter & angle) are likely to require the same adjustment amount.

The next parameter deals with carriage motion – **Carriage speed**. It describes the maximum permissible carriage speed during the wind. Actual carriage speed may not attain this value, depending on other parameters (angle, diameter and mandrel speed). In general, however, for many low-angle winds, carriage speed will be dominant. Note that the machine has a recommended carriage acceleration limit as well, although this is not presented because the given turnaround range and target speed fix the acceleration. The user is warned if the resulting acceleration exceeds the recommended limit, but the software will still generate motion – in general, excessive acceleration will become noticeable as the program is run near full speed.

The next four parameters deal with crossfeed motion. If the machine does not have a crossfeed, this section is grayed out. If the machine has a crossfeed, then the **Use Crossfeed** checkbox establishes if there will be any plunge motion of the crossfeed. Next to it, the user can enter a fixed crossfeed **Location** (which must be greater than part radius). The crossfeed will operate at this location for most (or if there is no plunge, then for all) of the wind. This location should be such that the crossfeed will clear the part and any pin-rings. Note that any tooling length must be added to this position since the crossfeed's location is normally configured to be the tooling flange. So if the part has a radius of 100mm, the tooling has a length of 150mm, and the pin rings have a height of 25mm, then a reasonable value might be 290 – for a clearance of 15mm ( $290 - 25 - 150 - 100 = 15$ ).

If **Use Crossfeed** is checked, then the user can also enter the plunge parameters:

- **Crossfeed Plunge** – the distance to plunge in on both sides of the part – starting from the crossfeed's **Location** parameter – over the course of the turn-around range.
- **Crossfeed speed** – the maximum speed at which the crossfeed will operate (this speed may not be reached depending on distance and acceleration, and it may not exceed the crossfeed's speed limit)
- **Crossfeed accel** – the acceleration to apply to the crossfeed during the plunge. If this exceeds the recommended limit, the user is warned, but motion can still be generated.

Finally, if the machine has an eye-rotation axis, a fixed axis position may be entered under **Eye-rotation position**. The axis will not move during the wind.

At the base of the screen there are two additional checkboxes. The **Fixed angle** checkbox allows the user to select between applying a fixed angle and finding the closest bandwidth to solve for motion (if checked), or to fix the bandwidth and find the closest angle. For a typical, low-angle wind with many circuits, fixing the angle tends to result in a wind closer to

both parameters than fixing the bandwidth (e.g. the bandwidth error may be a couple percent while the angle error may be a couple degrees).

Next to this is the **Metric Units** checkbox. If checked, then the user can enter metric units for the linear axes (mm, m/s, and m/s/s), otherwise standard units are used (in, ft/min, and in/s/s). On rotary axes, both systems use degrees, rpm, and rad/s/s.

Once parameters are entered, click on **Calculate** to proceed. At this point the entered data will be parsed. Any errors or warnings will be displayed. If an error occurs, the user cannot proceed with motion generation. Once any problems are corrected (or if only warnings are displayed and the user proceeds), a confirmation screen appears. If the user clicks OK, motion is generated. If the file has not yet been saved, the user will first be prompted to enter a filename. Finally, once the unidirectional wind and motion (*.uni* and *.mmt*) files are saved, the user will also receive a motion filter screen (see motion filtering chapter in main manual). Note that filtering ignores any speed and acceleration parameters from the first dialog box and instead enforces whatever limits are entered by the user in its dialog box. In general, filtering is not very important with this type of wind because motion is already fixed by the motion limit parameters entered in the initial dialog.

## 1.2. Wind Motion Description

The unidirectional wind has somewhat unique motion. To assist in understanding the various parameters and their affect on the wind, what follows is a basic description of the wind.

Upon executing a find path, the carriage would move to the starting Z-position, the crossfeed would move to the plunged position (which is the crossfeed's **Location** parameter minus the **Crossfeed Plunge** parameter). By default, the mandrel will rotate to its 0 degree location. If the mandrel should start at the current arbitrary position, then the Mandrel Offset flag must be set by loading the motion file (*.mmt*) and selecting **Edit Axis Motion | Segment Flags**, and then checking **Enable Mandrel Offset**.

At the beginning of the wind (and the beginning of each circuit), the mandrel will rotate through its first index motion. There are two index motions, which over the course of the wind will cause the mandrel to be evenly covered with fiber. An index motion occurs at both ends of the part. Index motion depends on whether or not the "**Fixed Mandrel Index**" box is selected. If selected, then the mandrel's index maneuvers are fixed by the **Index Value** and the **Mandrel Adjustment** term. The adjustment term is added to the first index and then subtracted from the second causing the terms to cancel out over the course of two strokes.

If the "**Fixed Mandrel Index**" box is unselected, mandrel indexing proceeds as follows:

The index motion is in opposite directions at opposite ends of the part – once in the forward direction and once in the reverse.

Given  $x$  number of strokes (which the software calculates based on diameter, angle, and bandwidth), the index distances are as follows:

On a wind with an odd number of strokes, the first index is  $180 \cdot (1 + 1/x)$  degrees, the second index is  $180 \cdot (1 - 1/x)$  degrees. For example, given a 3-stroke part, the mandrel will move forwards 240 degrees, then back 120 degrees and this will deposit fiber in all 3 slots (240, 120, then 360 degrees or slots 2, 1, and 0).

On a wind with an even number of strokes, the first index is  $180 \cdot (1 + 2/x)$  degrees, while the second index is  $-180$  degrees. In the special case of exactly two strokes, the forward index is 180 degrees. For example, given a 4-stroke part, the mandrel will move forwards 270 degrees, then back 180 (the sequence would be 270, 90, 360, 180 or slots 3, 1, 0, 2).

In all cases, any adjustment term is added to the first index and subtracted from the second (e.g. a 4 stroke wind with 20 degrees of adjustment would move +290 degrees, then  $-200$  degrees). These winds result in fiber being placed in all slots. The sequences can be repeated any multiple of times to deposit multiple layers.

After the mandrel has executed its first index motion, it then accelerates in tandem with the carriage at a rate fixed by the fiber angle. This acceleration occurs over the turnaround range. At the same time, if a crossfeed axis is present and a plunge motion has been defined, then the crossfeed will retract during this acceleration maneuver. Depending on which is faster (the crossfeed retraction or the acceleration of mandrel and carriage), one or the other maneuver is delayed so that both actions complete when the carriage has covered the turnaround range.

After the acceleration maneuver, both mandrel and carriage run at a constant speed fixed by the fiber angle, until they reach the end of the linear, useful part distance. They then both decelerate, and any crossfeed plunge occurs. At this point, the mandrel executes its second index maneuver – followed by the same acceleration, constant speed, and deceleration segments (with crossfeed retract & plunge maneuvers). This second set of motions occurs with the mandrel and carriage both running in the opposite direction from the initial stroke, resulting in the same fiber angle for both strokes – hence a unidirectional wind.

### 1.3. Notes and Tips for Unidirectional Winds

Most of the parameters for generating a unidirectional wind are quite straightforward. Like the zero-degree wind, it has a few peculiarities:

Because of reversing mandrel direction, it does not generate a path file. This has little effect on a typical wind; however, it implies that the wind cannot be visualized in the software. It also means that the wind cannot be joined to a chain file using a path-based transition. However, the other chain-file options do work, and typically, a user wishing to generate a chain wind would select the "Transition Motion Files" option (or possibly the "Do Not Transition" option if the file ends align well).

When selecting parameters, it is fairly common to get warnings regarding carriage acceleration. There are several possible resolutions:

- First the user could continue regardless. Provided the turnaround range is a reasonable length, the operator can choose to run the part slower than 100% and decide what acceleration is too high. As a mathematical basis – the acceleration reached is  $v^2/(2*x)$  where v is carriage speed and x is turnaround range. For example, given a top speed of 1m/s and a turnaround range of 0.1m, the resulting acceleration is 5m/s/s or about 1/2g. This is quite high. Depending on the size of the machine, carriage acceleration rates between 1m/s/s and 2.5m/s/s are considered acceptable. Note that if the same program is run at 50%, then the acceleration rate drops to 1.25m/s/s – which should be acceptable on all but the largest carriages (e.g. with slave creels).
- Lowering the carriage speed has the same effect as the operator running the machine slower, however it only affects the carriage – so the mandrel index motions can run at higher speeds/accelerations.
- Increasing the turnaround – this is a straightforward solution, and generally, with larger parts and larger machines, the turnaround will also grow, implying a greater overshoot of the pin-rings and resulting in reduced acceleration requirements.
- Increasing the turnaround without making the part larger – in this case, the machine would already begin to decelerate before the pin-rings are reached. If there is no crossfeed, then it poses no problem. However, if the crossfeed is plunging in, then caution is advised – the programmer should carefully examine the resulting program to ensure that the pins are not clipped by the crossfeed (lowering the crossfeed's acceleration and/or adjusting other plunge parameters can help prevent the crossfeed striking the pins).

Other errors /warnings include:

**Error - Starting [Ending] position outside carriage's range** – the part is larger than the machine can produce. Check that appropriate units are selected (i.e. check the Metric Units box) – if correct, then the starting and ending position must be adjusted. Consider reducing the turnaround range.

**Error - Bandwidth not appropriate for part** – occurs if the bandwidth is set so large that it would exceed covering ½ the circumference of the part in a single stroke. Generally, a significant bandwidth is required because other checks are more restrictive.

**Error - Angle not set appropriately** – only occurs if the angle is set beyond the +/-89 degree limits – even in this range, the angle may still be too large, in which case the following error occurs:

**Error - these parameters don't result in at least 2 strokes** – in this case, the combination of bandwidth, diameter, and angle result in a wind with less than 2 strokes. If a single stroke is desired, consider using the circumferential wind instead.

**Error – turnaround range not set appropriately** – the turnaround range must be positive and cannot exceed half the total stroke length. Reduce the turnaround range or increase the part length.

**Error – Crossfeed location outside crossfeed's range** – only occurs if crossfeed is present. The crossfeed's base location is outside the range of the axis. Typically, this parameter must be decreased which may require selection of shorter tooling to clear the part.

**Error – Plunged crossfeed position outside crossfeed's range** – only occurs if crossfeed is present and its location while plunged is outside the range of the axis. Typically, this parameter must be increased which may require selection of longer tooling to achieve the desired clearance.

**Error – crossfeed location smaller than part radius** – this state is not permitted, the crossfeed location must be increased.

**Warning – plunged crossfeed position smaller than part radius** – this is only a warning. However, if this warning appears then the programmer should not proceed unless the mandrel is not a cylindrical mandrel (i.e. it has a taper such that the plunged crossfeed will not strike the mandrel).

**NOTE:** due to tooling length – crossfeed location should be greater than the part radius plus the tooling length – even if no error is generated, it is still possible for the tooling to strike the mandrel.

Once the user has generated a suitable wind program, it can be filtered and modified as described in the main training manual. It can also be chained with other files to produce more complex winds (remember the limitation that it cannot be chained using path-based motion). Finally, it can be executed using Omniwind. For more information on all these steps, refer to the main manual.

One note on filtering – because the user can specify most of the motion parameters for the original wind, it typically has little benefit – however, it can cause a somewhat rare problem: if the wind has an odd number of strokes, then the machine software must stop the wind partway through a complete cycle. To establish this point, it currently uses one half the time of the cycle. If a mandrel adjustment term is used, then the software will automatically stretch the motion so that both mandrel index maneuvers require the same amount of time and thus the wind can be properly stopped at the ½-way point. However, if the resulting motion is filtered, the two mandrel index maneuvers will no longer require the same amount of time, and the ½-way point in distance will no longer coincide with the ½-way point in time. This can cause problems when executing the wind, but more substantial problems may occur when chaining multiple layers together. Again, this is only likely to occur on winds with an odd number of strokes and becomes more noticeable the larger the mandrel adjustment term becomes relative to other maneuver sections.