Raster/Cycloid Retrace Waveform Generation mdt 12/21/04

The purpose of this paper is to describe how to construct an efficient and accurate raster/cycloid waveform. The contention of this paper is this waveform is superior in its ability to move the scanner through a linear sweep and retrace with the smoothest waveform possible, thus minimizing resonance effects.

All physical things have naturally occurring resonances that when excited cause unacceptable behavior from the system. It is crucial when moving the scanner through very accurate positioning that the excitation of these resonances be avoided. Most systems that have high Q resonances employ either bandwidth limiting or band “notching” within the system to attempt to attenuate these resonances. Another method is to control the bandwidth of the input function (command signal). From basic Control Theory, the actual output response of any system is a combination of both the input function and the natural response of the system. This paper addresses how to construct a waveform that properly bandwidth limits the input function when “flying back” after a linear ramp has been performed.

The linear ramp is one of the most commonly used functions when forming an image or obtaining imaging information. The image is written or the information obtained while the scanner is sweeping one direction only thus minimizing the errors (referred to as scan to rescan errors) associated with scanning in both directions. Thus, the command signal must “fly back” to the starting point to start the next sweep as quickly as possible. This linear sweep with a fast retrace is called a “raster.” Several design rules should be followed in order to create the most efficient waveform possible.

1.) Accept all input parameters from the user/customer and attempt to perform the linear sweep at the correct place in space, at the correct time, and to then retrace back to the starting point as quickly as possible without exciting the higher order resonances of the system.
2.) The linear sweep must be as smooth as possible so the scanner is settled as much as possible for the entire sweep. Thus each step must be exactly the same size and occur with constant period or small perturbations in the sweep velocity will cause “banding” to occur.
3.) The retrace must begin and end as smoothly as possible and be as short as possible so as to maximize efficiency of the overall waveform. It also cannot be too short or either resonances will be excited or power limits will be reached. The non-linear circuitry that will then be employed on most modern servos will then destroy the elegant nature of the cycloid retrace. Care must be taken and sometimes iteration is necessary to choose the correct waveform. Experience helps greatly in choosing the exact parameters for any given system.
4.) Lastly, because of the finite bandwidth of the system, the command signal is attenuated. This means that what the command signal tells the scanner to do is not what the scanner actually did. It is similar to a truck turning a corner with the rear wheels cutting the corner early. This “attenuation factor” of the system must be taken into account in order to start and end the linear sweep accurately.

Linear Sweep Portion of the Raster/Cycloid Retrace Waveform

The linear sweep portion of the waveform is the working part. This is portion that the customer wants to be accurately positioned both in space and time. All other parts of the waveform are there just to make this part the most accurate. For example, when forming of an image, this is the part of the waveform when the beam is on.

First we need to define where the linear sweep starts, how long it is, how fast it will sweep, and how often will it do so. There are many ways of defining these terms. Shown below is the waveform derived both in general terms and for a specific case as an example.

1.) Define the starting position in degrees as: **A0** = -10 degrees
    a. A0 is the point at which the scanner is finally settled on the linear ramp after the retrace and settle portion.
    b. A0 is the position that is the actual start of the linear ramp for the position signal.
2.) Define the magnitude and sign of the linear ramp in degrees as: **A** = +20 degrees
    a. A is the length and direction for the settled portion of the scanner's position signal.
    b. This is not the total length of the ramp for the command signal. This will be explained later.
3.) Define the time spent performing the linear ramp as: **Tr** = 20msec
    a. TR is the time the scanner spends performing the linear ramp.
    b. The scanner's position signal should be settled for the entire Tr.
4.) Define the slope of the ramp in degrees per second as: **W1**
    a. W1 = A/Tr = 1000deg/sec and is the slope of both the command signal and the position signal for the ramp.
    b. The scanner will have this slope only when it is settled. This is described more fully below.
5.) Define the overall frequency of the waveform in Hz as: **F** = 40Hz
    a. F is the frequency at which the overall waveform will perform.
    b. This allows us to define the total period as: **T** = 1/F = 1/40Hz = 25msec
6.) Define the overall efficiency of the waveform in % as: **EFF**
    a. If we know both the total period, T, and the time spent on the ramp, Tr, then the efficiency of the waveform, EFF can be defined as:
       EFF = Tr/T * 100% = 20/25 * 100% = 80%
    b. Some customers prefer to define the efficiency of the waveform and then let the frequency be what it needs to be in order to achieve that efficiency. In that case: F = EFF*100/Tr
7.) Define the step response time of the scanner as: **Ts** = 1msec
    a. Ts is the 0 – 99% settling time for a critically damped system to a step input.
    b. Ts is a very important parameter for many of the calculations below. It is necessary for the customer to know this in order to create a waveform that is accurate for the entire ramp period.

The linear portion of the waveform takes the form of a straight line, that is:

$$y(x) = mx + b$$

Let x = time, t and y(x) = s(t) where s(t) is the position of the scanner in degrees with respect to time, and m = W1, the slope of the ramp. Thus,

$$s(t) = W1 \cdot t + A0 \quad \text{where } 0 \le t \le Tr$$

or

$$s(t) = 1000\text{deg/sec} \cdot t - 10\text{deg} \quad \text{where } 0 \le t \le 20\text{msec}$$

would create a command signal that performs the linear ramp we would like the scanner to perform. However the finite bandwidth of the system causes the scanner to lag behind the command signal. There are two lagging effects. In order for the ramp to be accurate, we must account for both effects.

From the moment that any command signal changes slope, the scanner responds in a very predictable way. The servo responds to the changing input as long as it keeps changing. Once it has stopped changing, or has stopped changing slope (such as a constant slope ramp) then the scanner settles onto that new position or slope. It will be settled onto the new position or ramp in precisely one step response time, Ts. Thus, the command signal has to be started sooner then when we need the scanner to be settled onto the ramp signal after a retrace. Additionally, once the scanner is settled on a linear ramp, the scanner will always lag behind the command signal in time by one half of a step response or ½ * Ts. Referring to figure #1, we can see both of these effects.

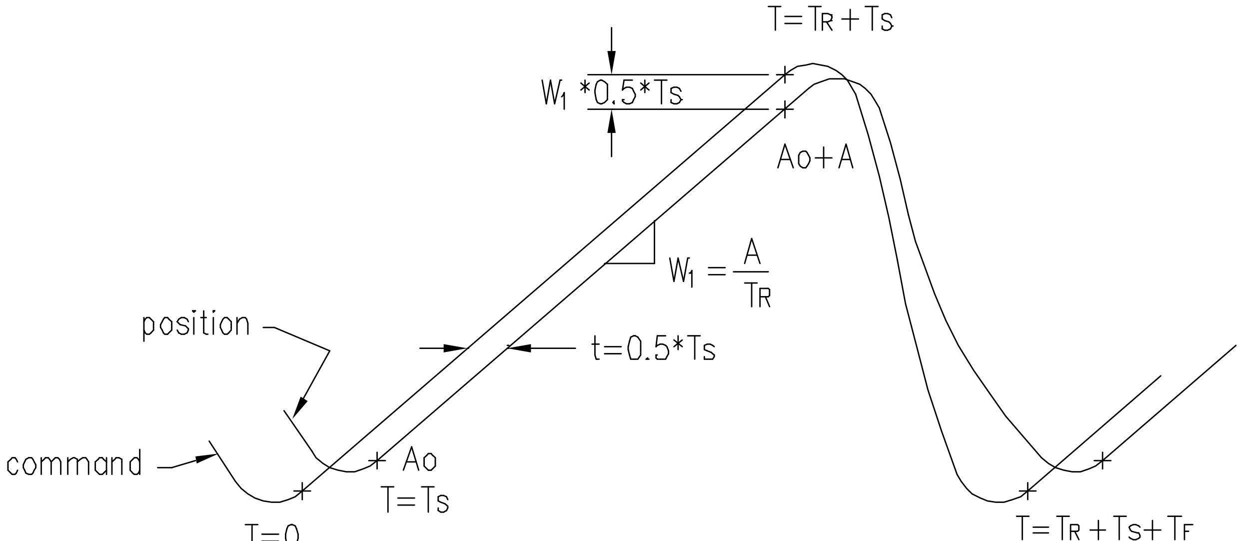

Figure #1.

Note even though the command signal is started Ts seconds before the time the scanner needs to be settled at A0, we only have to back up the start of that ramp in distance by 0.5 * Ts * W1 to ensure proper settling. In other words, we must back up the start of the command signal by this amount to give the scanner the proper amount of time to get up to the correct velocity at the correct point in time and space.

Also note that since we started the command signal one whole step response time behind in time, but only 0.5* Ts in distance, we have already accounted for the lag factor at the end of the ramp. Thus, no further elongation of the command signal is necessary since it was completely accounted for at the beginning. Again refer to figure #1.

These changes cause the equation for the linear ramp to take on the new form:

$$s(t) = W1 \cdot t + (A0 + 0.5 \cdot Ts \cdot W1) \quad \text{where} \quad (0 - Ts) \le t \le Tr$$

or

$$s(t) = 1000\text{deg/sec} \cdot t - 9.5\,\text{deg} \quad \text{where} \quad \text{-1msec} \le t \le 20\text{msec}$$

We will have trouble computing something in negative time, so we must shift the waveform again so that the entire waveform is running in positive time. To do this we add 'Ts' to both ends of the inequality above, and subtract it from 't' in the equation. Doing this and combing terms results in the equation shown below.

$$\underline{s(t) = W1 \cdot t + (A0 - 0.5 \cdot Ts \cdot W1) \ \text{where} \ \ 0 \le t \le (Tr + Ts)}$$

or

$$\underline{s(t) = 1000\text{deg/sec} \cdot t - 10.5\,\text{deg} \quad \text{where}\ 0 \le t \le 21\text{msec}}$$

But, because of the time shift, now that scanner is settled on A0 not at time t = 0, but at time t = Ts = 1msec. Please note that when programming the system triggering of the laser or other data acquisition system.

The settled portion of the ramp for the scanner is positioned such that it begins now at A0 degrees at Ts seconds, and it ends at A0 + A degrees at Ts + Tr seconds. It will have slope W1 degrees/second. Thus, all of the original design parameters will be met.

Discretizing the Waveform

In order to program this waveform into a computer, it must first be discretized.

1.) Instead of using the independent variable for time, t, we define a step index, **n**.
    a.) 'n' is the particular step number at any point in the waveform. For the very first step in the waveform, n = 0, then n = 1, etc.
2.) Instead of using the dependent variable for position as a function of time in degrees, s(t), we define a variable that is DAC counts as a function of n, **S(n)**.
    a.) S(n) is the position in DAC counts at any step, n.
3.) A scale factor, **SC**, that relates DAC counts to degrees is defined in terms of the systems Command Input Scale Factor (refer to the system's instruction manual for a description of this), the DAC's maximum voltage output, and the DAC's size in bits. We define this scale factor in the following manner. Note that B is the size of the DAC in bits.

$$SC = \frac{\text{Max DAC Voltage Range in Volts}}{\text{Command Input Scale Factor in deg/volt}} \cdot \frac{\pi\text{, radians}}{180\text{ deg}} \cdot \frac{1}{2^B - 1}\text{, in radians per DAC count}$$

For a CTI system that has:

a. Max DAC Voltage Range = 20 volts
b. Command Input Scale Factor = 0.5volts/degree
c. A 16-bit bipolar DAC
d. Yields:

$$SC = \frac{20\text{volts}}{0.5\text{volts/deg}} \cdot \frac{\pi\text{, radians}}{180\text{ deg}} \cdot \frac{1}{2^{16} - 1} \Rightarrow$$
$$SC = 10.65281 \cdot 10^{-6}\text{ radians/dac count} = 10.65281\mu\text{radians/dac count}$$

Note: Because later calculations demand it, all discretized calculations must be done in radians, not degrees.

4.) Using SC, we can calculate some of the ramp parameters. Note that since CTI DACs are unipolar, an offset of 32768 is added:

a. Starting point for the linear ramp in DAC counts, **$A0_D$**:

$$A0_D = A0 \cdot \frac{\pi}{180} \cdot \frac{1}{SC} + 2^{B-1}$$
or
$$A0_D = -10 \cdot \frac{\pi}{180} \cdot \frac{1}{10.65281 \cdot 10^{-6}} + 32768 \Rightarrow$$
$$A0_D = 16384.25 \cong \underline{16384\text{dac counts}}\ \text{(No fractions allowed!)}$$

b. The length of the ramp in DAC counts, **A_D**:

$$A_D = A \cdot \frac{\pi}{180} \cdot \frac{1}{SC} + 2^{B-1}$$
or
$$A_D = 20 \cdot \frac{\pi}{180} \cdot \frac{1}{10.65281 \cdot 10^{-6}} \Rightarrow$$
$$A_D = 32767.5 \cong \underline{32768}\text{ dac counts (No fractions allowed!)}$$

5.) A time increment between each step, **Tinc** that relates the continuous variable time, t to the discrete step variable, n. This term is very important and some time should be spent discussing this, as it will affect the overall system performance very strongly. The scanning system will perform that smoothest ramp if both the time interval between steps <u>and</u> the step size for each step are constant for every step.

The step size must be chosen so that the size is an integer multiple of a single DAC count. If the slope of the ramp is computed such that a round off occurs, then eventually the step size will change by at least one LSB. This will cause a temporary change in the velocity of the scanner right in the middle of a scan. The will cause unacceptable "banding" to occur in most images. Thus, the ramp must be constructed of some number of steps that must divide evenly with no remainder. Choosing a ramp size that is some prime number of LSBs (least significant bits) is then not possible. Selecting the proper time interval so that slope is maintained can accommodate slight changes in the length of the ramp.

Conversely, the time increment chosen also affects the size of the steps. The scanner's bandwidth can sometimes be very high and choosing a step size and time increment that allows the scanner to much time to settle between each step will cause high velocity jitter and poor positioning will result. Here is one method to guide you in your choice of Tinc:

a. Arbitrarily choose an interim Tinc so that it is somewhere between 0.05 * Ts and 0.01 * Ts. – Let's choose for our example: Tinc = 0.04 * 1msec = <u>40µseconds</u>.

b. Calculate an interim **Nr**, the total number of steps in the settled ramp:

$$Nr = \frac{Tr}{Tinc} \quad \text{For our example: } Nr = \left(\frac{20\text{msec}}{40\mu\text{sec}}\right) = 500 \text{ steps}$$

c. Calculate an interim step size, **Sz**:

$$Sz = \frac{A_D}{Nr} \quad \text{For our example: } Sz = \frac{32768}{500} = 65.536\frac{\text{dac counts}}{\text{per step}}$$

d. To solve the fraction problem (fractional step sizes will cause round offs to occur), let's change Tinc just slightly so that Sz is an integer and so that Sz divides evenly into Nr. Let's choose in the above example, Sz = 64 dac counts per step. Thus, the new Nr and Tinc are:

$$Nr = \frac{A_D}{Sz} = \frac{32768}{64} = \underline{512 \text{ steps}} \Rightarrow$$

$$Tinc = \frac{20\text{msec}}{512\text{steps}} = \underline{39.0625\mu\text{sec}}$$

e. It is very important that the time between steps now be held as close to that Tinc as possible or else the slope will not match the original customer's requirements. In the above example, more steps that were each smaller could have been chosen, but then the requirements on how finely the time is divided becomes even harder. Ultimately it is up to the Engineer designing the hardware to determine what these numbers must be to meet the customer's requirements.
f. If we instead picked a Tinc, then an Sz, we could have let Nr be what changes which would have caused the end of the ramp to be different. For most applications the customer decides what the slope of the ramp should be, not the exact end point of the linear ramp. However, the

finite resolution of both time and magnitude to be quantized determines how closely the waveform matches the customer's original requirements.

6.) Calculate the number of steps during the speed up portion at the beginning of the ramp, **Ns**:

$$Ns = \frac{Ts}{Tinc} \quad \text{For our example: } Ns = \frac{1\text{msec}}{39.0625\mu\text{sec}} = 25.6 \text{ steps} \cong 26 \text{ steps}$$

Here it is OK if the ramp is just slightly longer than necessary. What it cannot be is shorter since the scanner can always settle slower than this time, but not faster. But this also means the linear ramp start time may have to be shifted slightly back in time or the scanner may not be at exactly the correct point in space at the exact correct time. These are some of the trade offs that must be accounted for when discretizing both time and angle.

7.) The angular velocity during the ramp in dac counts per step, **$W1_D$,** is now calculated as:

$$W1_D = \frac{Sz}{\text{step}} = Sz \quad \text{For our example: } W1_D = 64 \text{ dac counts/step}$$

8.) Finally, all of the terms above can be put into the linear equation form to create the discrete time version of the original ramp function:

$$\underline{S(n) = W1 \cdot n + \left(A0_D - 0.5 \cdot Ns \cdot W1_D\right) \quad \text{where} \quad 0 \le n \le \left(Nr + Ns\right)}$$

For our example:

$$\underline{S(n) = 64 \frac{\text{dac counts}}{\text{step}} \cdot n + 15552 \text{ dac counts} \quad \text{where } 0 \le n \le 538 \text{ steps}}$$

Cycloid Flyback Portion of the Raster/Cycloid Waveform

The “cycloid” portion is a waveform that was found originally in a cam designer’s handbook. It is used to transport a cam follower from one position to another while controlling the first and second derivatives to keep from damaging the follower and/or the cam. This waveform has been adapted for our purposes in order to connect the end of the ramp to the beginning of the next ramp in the same manner that the cam transported the cam follower.

The standard cycloid itself is made up of a linear ramp with positive slope and a negative sine wave each with amplitude such that the position, velocity, and acceleration are all continuous functions at the beginning and ending transitions of the waveform. This is why this waveform minimizes exciting the mechanical resonances of the system. Thus, the purpose of this section of this paper is to guide the reader to construct such a cycloid waveform as to complete the raster as smoothly and as quickly as possible.

The equation of motion for a standard cycloid that would transport a cam follower from starting point A0 to ending point A0 + A in T time is the following:

$$s(t) = \frac{A}{2\pi} \cdot \left[\omega \cdot t - \sin(\omega \cdot t)\right] \quad \text{where} \quad \omega = \frac{2\pi}{T}, \text{and } 0 \le t \le \frac{2\pi}{\omega}$$

Before we go on, we need to define some additional terms:

1.) The flyback time in seconds: **Tf**
   a. Tf = T – Tr – Ts where T = the total period (1/F), Tr = the ramp time, and Ts = the settling time of the scanner as defined above in the linear portion of this paper
   b. The flyback time is set according to the customer’s needs, but also within the limits of what the system can do without overly exciting the resonances or without causing the servo to go non-linear.
   c. A good rule of thumb for the flyback time is to start off with it being as long as possible, and then shortening it as the system allows. Once it is so fast that the resonances become excited or the voltage or current limits on the servo’s output stage are beginning to saturate, it is now too fast. Again experience will be the main guide for this term.
   d. For our example: Tf = 25msec – 20msec – 1msec = <u>4msec</u>

2.) The circular velocity of the sinusoidal component of the above waveform: **W2**
   a. W2 = 2*π/Tf and its units are in radians/second.
   b. Note: The <u>circular</u> velocity of W2 is not to be confused with the <u>angular</u> velocity of W1. W2 is measured in radians as a measure of its distance around the unit circle. Please do not confuse the two.
   c. For our example: W2 = 2*π/4msec = <u>1570.8 radians/sec</u>

Using these new terms, the cycloid flyback becomes:

$$s(t)=\frac{A}{2\cdot\pi}\cdot\left[W2\cdot t-\sin(W2\cdot \mathrm{t})\right]\ \text{ where }\ 0\le t\le\frac{2\cdot\pi}{W2}$$

For our example:

$$s(t)=3.1831\frac{\text{deg}}{\text{radians}}\cdot\left[1570.8\frac{\text{radians}}{\text{sec}}\cdot t-\sin\left(1570.8\frac{\text{radians}}{\text{sec}}\cdot t\right)\text{radians}\right]$$

where $0\le t\le 4\text{msec}$

Note: The radians after the sin component above was necessary to make the units agree. Radians have this ability without violating any dimensional analysis when it is referring to circular velocity.

But this cycloid starts and stops with no initial or final velocities, so although the start and stop position terms may be coincident with the original linear ramp's, the velocity term is not. What is missing is a velocity term that matches our W1 calculated earlier for the linear ramp. To correct this, we add the W1 * t term to the above equation resulting in:

$$s(t)=\frac{A}{2\cdot\pi}\cdot\left[W2\cdot t-\sin(W2\cdot t)\right]+W1\cdot t\quad\text{where }\ 0\le t\le Tf$$

For our example:

$$s(t)=6.3662\frac{\text{deg}}{\text{radian}}\cdot\left[1570.8\frac{\text{radians}}{\text{sec}}\cdot t-\sin\left(1570.8\frac{\text{radians}}{\text{sec}}\cdot t\right)\text{radians}\right]$$

$$+1000\frac{\text{radians}}{\text{sec}}\cdot t\ \text{ where }\ 0\le t\le 4\text{msec}$$

This equation now has the correct starting and stopping ending velocities, but the W1 * t term made the waveform not traverse all of A degrees back. Also there is no direction shown above, and it is likely the flyback would occur in the wrong direction. Thus a new definition:

1.) The amplitude in degrees of the flyback waveform: **A2**
   a. A2 needs to account for the the "W1 * t" term for one whole cycle or for Tf seconds. Thus, a "W1 * Tf" term needs to be added to the amplitude.
   b. A2 also needs to account for the small settling portion at the beginning of the linear ramp. This is being run for Ts seconds. Thus, a "W1 * Ts" needs to be added to the amplitude.
   c. The sign of A2 must always be opposite that of A in order to reverse direction and go back to the beginning.
   d. Thus,

$$A2 = -\left(A + W1 \cdot \left(Tf + Ts\right)\right)$$

For our example:

$$A2 = -\left(20\text{deg} + 1000\text{deg/sec} \cdot \left(.004\text{sec} + .001\text{sec}\right)\right) = \underline{-25\text{degrees}}$$

2.) The cycloidal discussed thus far does not start at where the ramp ended. Thus we define another term: **Astart**
   a. At the end of the settled portion of the linear ramp the command signal is 0.5 * Ts * W1 ahead of the position signal.
   b. At that same time, (Tr + Ts), the position signal is at A0 + A.
   c. Thus, Astart = A0 + A + 0.5 * Ts * W1
   d. For our example:

$$Astart = -10\text{deg} + 20\text{deg} + 0.5 \cdot 0.001\text{sec} \cdot 1000\text{deg/sec} = \underline{10.5\text{deg}}$$

3.) Lastly, in order for the waveform to be positioned at the correct place in time, it must be shifted Tr + Ts later than shown above. To accomplish this, the limits listed in the inequality are shifted by the Tr + Ts seconds. To account for this within the equation, the shift is subtracted back out.

4.) Thus, combining all terms:

$$s(t) = \frac{A2}{2 \cdot \pi} \cdot \left[W2 \cdot \left(t - Tr - Ts\right) - \sin\left(W2 \cdot \left(t - Tr - Ts\right)\right)\right] + W1 \cdot \left(t - Tr - Ts\right) + Astart$$
$$\text{where } \left(Tr + Ts\right) \le t \le \left(Tr + Tf + Ts\right)$$

For our example:

$$\underline{s(t) = -3.9789\frac{\text{deg}}{\text{radian}} \cdot \left[1570.8\frac{\text{radians}}{\text{sec}} \cdot \left(t - 21\text{msec}\right) - \sin\left(1570.8\frac{\text{radians}}{\text{sec}} \cdot \left(t - 21\text{msec}\right)\right)\text{radians}\right]}$$

$$\underline{+1000\frac{\text{radians}}{\text{sec}} \cdot \left(t - 21\text{msec}\right) + 10.5\text{deg} \quad \text{where } 21\text{msec} \le t \le 25\text{msec}}$$

Discretizing the waveform:

The last step in this process is to discretize the waveform. We will use the terms calculated in the previous discretizing section, and a few more. The most important thing to remember is that we keep the same Tinc as before, and we don't worry about round off errors because during this section of the waveform, the scanner is never settled. Thus, any velocity perturbations created here will be settled out before starting the ramp. Thus,

1.) Define the number of steps for the flyback as: **Nf**

$$Nf = \frac{Tf}{Tinc}$$

For our example:

$$Nf = \frac{4\text{msec}}{39.0625\mu\text{seconds/step}} = 102.4\text{steps} \cong \underline{102\text{steps}}$$

2.) Define the circular velocity of the flyback in radians/step as: **W2$_D$**

$$W2_D = \frac{2 \cdot \pi}{Nf}$$

For our example:

$$W2_D = \frac{2 \cdot \pi}{102} \underline{\frac{\text{radians}}{\text{step}}}$$

3.) Define the size of the flyback in dac counts as: **A2$_D$**

$$A2_D = \left[- A_D - W1_D \cdot (N_F + N_S)\right]$$

For our example:

$$A2_D = (32768 - 64 \cdot (102 + 26) = \underline{-40960 \text{ dac counts}}$$

4.) Define the starting position for the flyback in dac counts: **Astart$_D$**

$$Astart_D = A0_D + A_D + 0.5 \cdot W1_D \cdot N_S$$

For our example:

$$Astart_D = 16384 + 32768 + 0.5 \cdot 64 \cdot 26 = \underline{49984 \text{ dac counts}}$$

5.) Combining all these new terms together with what we already had yields the following discretized flyback:

$$S(n) = \frac{A2_D}{2\cdot\pi}\cdot\left[W2_D\cdot(n - Nr - Ns) - \sin\left(W2_D\cdot(n - Nr - Ns)\right)\right] + W1_D\cdot(n - Nr - Ns) + Astart_D$$

where $(Nr + Ns) \le n \le (Nr + Nf + Ns)$

For our example :

$$S(n) = -\frac{40960}{2\cdot\pi}\frac{\text{dac counts}}{\text{radian}}\cdot$$

$$\left[\frac{2\cdot\pi}{102}\frac{\text{radians}}{\text{step}}\cdot(n - 538\,\text{steps}) - \sin\left(\frac{2\cdot\pi}{102}\frac{\text{radians}}{\text{step}}\cdot(n - 538\,\text{steps})\right)\text{radians}\right]$$

$$+\,64\,\frac{\text{dac counts}}{\text{step}}\cdot(n - 538\,\text{steps}) + 49984\,\text{dac counts}$$

where $538\text{steps} \le n \le 640\,\text{steps}$ and $s(n)$ is an integer and must be rounded off to the nearest dac count