Omega Race AG Converter Board Description

The Game:

The arcade game "Omega Race" is a B&W vector game released by Midway in 1981. There were several cabinet styles such as a single player upright, mini, and cockpit, and a two player cocktail table. The object of the game is to fly your ship around a simple rectangular racing course destroying as many droids as possible without getting hit yourself. More information can be found on the following website www.klov.com .

The Spinner:

The direction of the player's ship is controlled using a spinner. The game board was designed to accept a 6-bit parallel Gray code from the spinner. The spinner assembly, originally, was a mechanical contacting Gray code encoder. The encoder assembly was relatively expensive and wore out quickly. This made it a bad investment for arcade operators. You might imagine that an ideal replacement would be an optical encoder assembly that, although a 6-channel Gray output device would be quite expensive, would be a much longer lived device. Alternatively, a much less expensive 2-channel optical quadrature encoder could have been chosen. The quadrature encoder could have been decoded into 6-bit Gray code by an add-on circuit board much the same as was done in the game Tron. Such a solution would have provided a reliable linear control without any jitter or dead spot issues. This was not the case however. The alternative chosen by Midway was a simple and inexpensive mechanical potentiometer (pot). The pot has no built-in end stops so it can spin freely. You might think that since Midway had so many problems with the first mechanical encoder that they would avoid a second mechanical solution, but this is in fact what they chose. An additional circuit board is required to read the pot position and convert it to a Gray code value. While the first time installation of the pot and the converter board might seem relatively expensive, the converter board would be a one-time purchase since it would not wear out. When a pot wears out or is damaged, only the pot need be replaced. Over the long term, this would prove to be much less expensive than replacing the mechanical Gray encoder assembly several times.

The A/G Converter Board:

Because a potentiometer (pot) is an analog device (simply a variable resistor), an additional circuit is required to measure the resistance of the pot (ship position) and output the appropriate proportional 6-bit Gray code that will be sent to the main game board. The *"Analog to Gray Converter board"* (or AGC) performs this function for two channels (two pots in, two 6-bit codes out). Since the AGC board can convert two pots simultaneously, only one board is required for all versions of the game including the two player cocktail table. After removal of the original mechanical 6-bit encoder, the pot(s) and the board must be installed. The original wiring harness connector that used to plug into the 6-bit Gray encoder now plugs into the output header of new AGC board. The pot(s) and a +5VDC supply are connected to the input port of the new AGC board.

AGC Operation:

The operation of the AGC board is the same for both channels. The circuit consists of a simple timer clock (555) that increments a 6-bit binary counter (comprised of two 4-bit counters 74LS163). The counter is free running and resets to zero each time it overflows making the output value range 0 to 63. The output of the counter is converted to Gray code by five EX-NOR gates (74LS266). The output of these gates (Gray bits 0 thru 4) and binary bit-5 are sent to a 6-bit latch (comprised of two 4-bit latches 74LS174). The output of the 6-bit latch is the Gray code that gets sent to the game. The 6-bit latch must grab and store the counter value the represents the position of the pot. This is done using a 556 dual timer that is set up as a pair of monostable multivibrators (one-shot). Each one-shot timer is reset and triggered each time the counter is reset to zero. The resistance of the pot(s) determines the charge time (pulse length) of the one-shot. As the timing capacitor of the one-shot charges through the pot, the counter is incrementing. When the timing capacitor voltage reaches a fixed threshold, the output pulse ends (changes state) and triggers the latch to capture the present value of the counter. The output of the latch is thus updated with the latest Gray code. This cycle of events repeats once every 64 ticks of the free-running 555 timer. Since it runs at approximately 50 kHz, the sampling rate of the converter is approximately 780 Hz. This is much faster than is actually required to make the motion of the player's ship seem fluid. Even if the player could rotate the spinner knob all the way around in just one tenth of a second, the ship position would be sampled 78 times and the game could render the ship at all of the 64 possible positions during that single spin.

AGC Board Components:

One might wonder why the AGC board uses 555 timers and several counters, gates, latches, and discrete components to generate an output code instead of using an ADC (analog to digital converter) chip (one for each channel). An ADC circuit would produce a more linear output and would require fewer components and a smaller PCB. Today, this would be the case. In fact, these days a very tiny and cheap microcontroller with a built-in ADC could sample the pot voltage and convert it directly into Gray code. However, back in the early 1980's, most ADC chips were very expensive and also required a split (or dual) power supply, a precision voltage reference, and an output buffer amplifier. All things considered, the 555 timers and the other TTL chips were much less expensive and would operate on a single +5VDC supply.

AGC Board Issues:

Dead Spot:

All pots consist of a mechanical wiper that moves around a resistive track. The track has to have a beginning and an end with terminals at each end. Typically, the angular gap between the two ends is at least 30°. Some pots have a gap as wide as 90°. No ship motion would be produced as the pot is turned through this gap. However, the actual ship angle code would be interpreted as position 63 (of 63) since the pot is actually an open circuit throughout this range of motion. The ship angle shown on the screen would depend upon the position of the pot when the level began because the game samples the pot value (Gray code) and uses this value as an offset value throughout the level. In other words, the pot will produce a "control dead range" that is different and random for each level, that is unless you mark the spinner knob and deliberately align it to a favorite position before each level begins. The worst aspect of the dead spot is that a small change in ship angle requires a large change in spinner angle as the spinner is turned over the entire dead spot gap.

Jittery Ship:

As the spinner pot wears out, the wiper can lose contact with the resistive trace or the trace itself can wear out. This can cause intermittent or inconsistent resistance values that will be read by the AGC board as different positions thus making the ship angle appear to jump around.

Missing States:

Because the timing of the 555 one-shots and the counter clock on the board is based on resistor and capacitor values and since the value tolerance of such components is quite wide, each channel must be calibrated by an on-board single-turn trimpot. Correct calibration is achieved when the counter reaches the value 63 when the resistance of the spinner pot is at its maximum (10 kΩ) but is not open circuit (inside the dead spot gap region). Maladjustment of the calibration pot can cause the spinner pot to reach the maximum Gray code value 63 too early or not at all thus making the apparent dead spot (described above) seem even larger than it should be. Failing to reach the maximum value of 63 is the worst of the two possible conditions because missing values (or counts or states) means that the player ship cannot be controlled to the angles represented by those missing values.

Another area of missing states is at the very beginning of the counter capture window. When the spinner pot is turned all the way to the minimum value of its range, the counter capture window is as short as it can be. However, even this short period is longer than a single clock cycle. On one sample board, the period was long enough that the counter incremented to a value of two. This means that states zero and one were not achievable and are therefore missed states. As with missed states at the other end of the timing window, these missed states add to the control dead spot.

Linearity:

The linearity of the spinner pot is the measure of how consistently the resistance changes as the rotational angle changes. Typically, a pot has a specification that indicates that the resistance scale has linear or logarithmic (audio) taper. You definitely would not use anything other than a linear taper. The linear response is supposed to change the resistance by a consistent and repeatable amount for any given measure of angular displacement over the entire range of angular displacement. However, linear pots do tend to bend this response quite a bit near the ends of travel. Thus you might find that the ship angle changes a relatively large amount for a small amount of spinner angle change when it is near the end of travel of the pot.

Non-standard Gray Code Output:

The circuit design of the AGC board uses EX-NOR gates to convert the binary counter value into a Gray code value. EX-NOR gates were used instead of EX-OR gates so the Gray code bits generated by the gates are inverted. The inverting gates may have been used because the *"counter reset"* signal generated by the counter must be inverted to reset the counter chips and the 555 one-shot timers. One of the EX-NOR gates is used as the inverter gate to accomplish this. However, even though there remain two unused EX-NOR gates, neither of them were used to invert the most significant bit (bit 5) of the 6-bit output code. All of this means that while bits 0 thru 4 are inverted, bit 5 is not. To read the Gray output code values 0 thru 63 from the AGC board sequentially, the output can be EX-OR'd with the binary value 011111 (or 1F as a hex value) which effectively inverts bits 0 thru 4 but not bit 5. The game code likely does this. The reason why the Gray code values were generated this way (with some bits inverted and others not) may be that this was simply the way that the original mechanical Gray encoder was already designed. Many mechanical Gray encoders manufactured today do not produce sequential output values. For example, the Bourns ACE128 series Gray encoders produce a completely randomized sequence that requires a lookup table to decode. Apparently, the reason for this is that the sequence dramatically simplifies the design of the conductive tracks and makes the design more compact and more reliable. The disadvantage is that the standard method of converting the values using an EX-OR function (either in hardware or software) is not possible. The spirit of Gray code is maintained however because only 1 bit of the value changes state between adjacent positions.

Calibration:

Calibration is best done using an oscilloscope with 3 channels or with a 2 channel scope and a voltmeter.

1. Connect a voltmeter or scope channel to the pot input lead and turn the pot to its maximum resistance (close to 10 kΩ). It is important that the pot not be within the "dead spot" gap range. The output should be around 2V but this varies depending on the position of the calibration trimpot. Alternatively, you could disconnect the input connector from the AGC board and measure the resistance of the pot directly using an ohmmeter and set it to its maximum (approximately 10 kΩ, not open circuit).
2. Connect a scope probe to pin 6 or 8 or 4 or 10 of the 556 timer chip. This will show the timer (and counter) reset pulse. The falling edge of this pulse is the end limit of the timing window. The rising edge is the beginning of it. Set the scope to trigger on this pulse.

3. Connect the next scope channel to pin 5 for pot channel #1 (or pin 9 for pot channel #2). This shows the counter capture window. The falling edge shows when the counter value is latched.
4. If you have a third scope channel, you may optionally connect it to pin 3 of the 555 astable timer. This shows one pulse for each increment of the counter. The timer reset and counter clear are synchronized to this clock.
5. Turn the trimpot on the board that is above the 556 chip fully counter clockwise. This trimpot will be used to calibrate channel #1. Turn the trimpot on the board that is below the 556 chip fully clockwise. This trimpot will be used to calibrate channel #2.
6. With the input port connected and the board powered up, the falling edge of the counter capture window should appear significantly before the falling edge of the reset pulse. Turn the channel #1 trimpot clockwise (or the channel #2 trimpot counter clockwise) until the falling edge of the counter capture window moves to a position very close to the falling edge of the reset pulse (within about 10 to 12 μs [microseconds]). Do not adjust the trimpot beyond this point. If you do, the spinner pot will reach the maximum code value too soon. If the capture window is too short, not all codes will be achievable. Either condition makes the "dead spot" in the spinner control appear larger although the latter condition is worse because the ship cannot reach certain angles due to the missing codes.

Absolute versus Incremental Encoders:

Why did Midway game developers decide to use an *absolute* type of encoder for the spinner? After all, it requires 6 wires and a much more complex and expensive encoder as well as more pins on the connector headers of the circuit boards and more I/O lines in the input circuitry. Other games such as Atari Tempest, Major Havoc, or Sega Star Trek have much simpler *incremental* 2-channel quadrature encoders for their spinners. Incremental encoders are cheaper, require far fewer wires, and their 2-channel output can be read by the game main processor. Sounds great right? Consider the game Bally-Midway Tron. It too uses a 2-channel incremental encoder but that encoder is connected to a separate small counter board that converts the incremental quadrature encoder signals into an 8-bit wide binary value that is then sent to the main game board. Why?

At first thought, one might assume that the angle of the spinner control could be directly translated to that of the player's ship (or direction of firing as in Tron). Well, in can be, however in the case of Omega Race, the ship angle is always set to a specific value at the start of each level regardless of the actual absolute angle of the spinner control. That said, the spinner control is treated as if it was incremental. The game reads the absolute value of the spinner at the beginning of the level, calculates the difference between that and the desired position, and applies this new offset value for the duration of the level. Okay, so if Omega Race treats the absolute encoder as if it was incremental, then why was the game designed for an absolute encoder?

The answer has to do with timing. An incremental encoder outputs pulses (or ticks) as it turns. The main game processor would have to detect those pulses (logical state changes of the signals) by repeatedly sampling the signals. If the sampling rate is slower than the fastest rate at which a player can produce pulses by turning the spinner, the game will miss some of the pulses. This can result in the turning rate of the ship dividing down intermittently or it can even result in the ship changing directions. However, polling incremental encoder signals fast enough to prevent such problems can be difficult for the main game processor to do especially considering that it has to generate all aspects of the entire game at the same time. In contrast, no matter how fast a player turns an absolute encoder, the output code (position) is absolutely known. Even if the game has missed some of the state changes since the last sampling, the difference between the new position and the old position is absolutely known. In other words, the main game processor can sample the absolute encoder at its own pace without losing sight of what position the encoder is in. This is also the reason why Tron uses an external counter board to convert its quadrature incremental encoder into an absolute 8-bit binary value. Similarly, Midway could have taken the same path as Tron by designing the AGC board to count pulses from a Tron-style incremental 2-channel quadrature optical encoder instead of sampling a pot. This would have resulted in a more linear and jitter-free output with no dead spot and the mechanism would have had a much longer working lifespan.