BURR-BROWN®
APPLICATION BULLETIN
Mailing Address: PO Box 11400, Tucson, AZ 85734 • Street Address: 6730 S. Tucson Blvd., Tucson, AZ 85706 • Tel: (520) 746-1111
Telex: 066-6491 • FAX (520) 889-1510 • Product Info: (800) 548-6132 • Internet: www.burr-brown.com/ • FAXLine: (800) 548-6133

ADS7809 TAG FEATURE

By Jerry Horn

The ADS7809 is part of a family of capacitive redistribution SAR A/D converters that feature a serial output and a tag pin for cascading multiple converters. Other members of this family include the ADS7806, ADS7807, ADS7808, ADS7824, and ADS7825. Note that the even numbered converters are 12-bit converters and the odd numbered converters are 16-bit converters.

This application bulletin deals only with the tag feature on the ADS7809. While other members of the family are slightly different, they are similar enough that this document can be used as a guide. Some members of the family do not include the SYNC pulse. In the following diagrams, this pulse and its associated clock cycle should be ignored for the ADS7806 and ADS7807. Likewise, the 12-bit converters require only 12 clock cycles for the conversion data instead of 16.

The preferred method of using the tag feature is covered first—a gated clock which is active after the conversion is complete. While this is the best method for the entire family, it is particularly true for the 16-bit converters. If this method is not acceptable, other ways of using the tag feature are described at the end of this document. The first method should be studied to determine if it can be used. If not, the alternatives will then be more understandable as well as the disadvantages that accompany them.

THE BASIC SETUP

Figure 1 shows the digital connections between two ADS7809s that have been set up to use the tag feature. The connections for the two converters are identical except that the DATA pin of converter A is connected to the TAG pin of converter B. Also, the TAG pin of converter A is grounded (why the TAG pin is tied LOW will be explained later). Note that while $\overline{CS}$ (chip select) is shown as being supplied by the user, this document assumes this input will always be LOW unless specifically stated otherwise.

The tag feature only works with an external clock. It is very important to remember that while the serial shift registers inside the converters are being clocked by the external clock, the actual conversion is still being run by the internal clock. For the entire family, the conversion time is set by an internal, laser-trimmed clock and cannot be changed by any external settings.

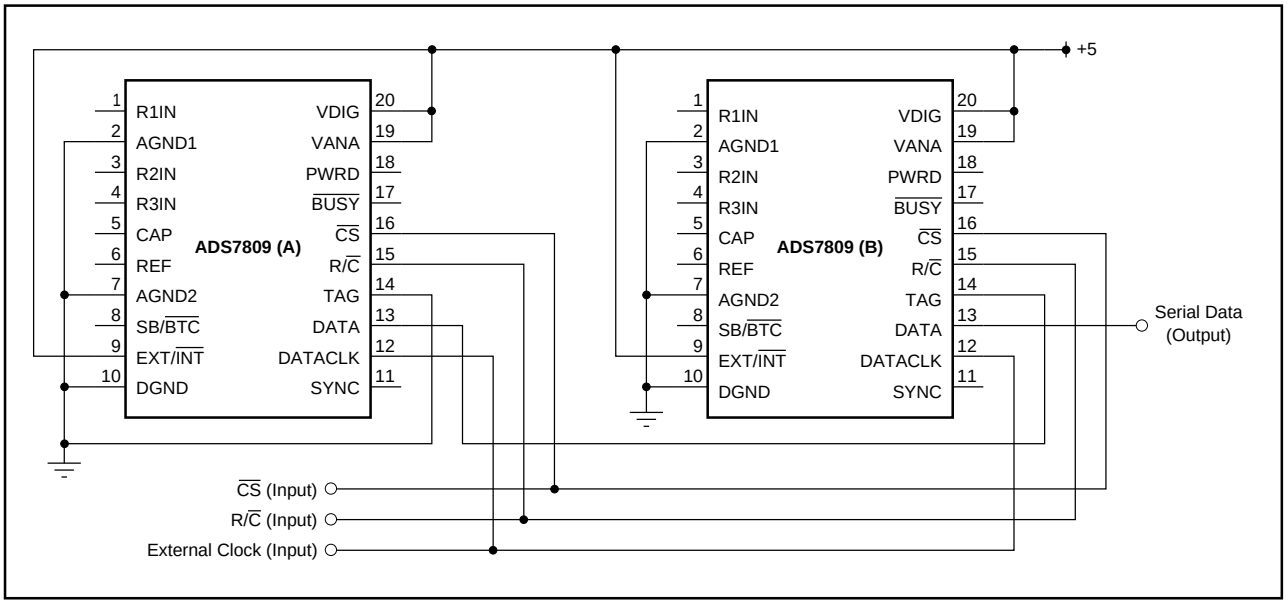

FIGURE 1. Two ADS7809s Utilizing the Tag Feature.

Printed in U.S.A. January, 1997

Figure 2 shows a very simplified diagram of how the serial ports work in conjunction with the TAG pin for the two converters of Figure 1. The important point on this diagram is the extra output flip-flop connected to the DATA output pins (top and bottom flip-flops at the far right of the diagram).

The other flip-flops are the holding registers for the conversion results as well as being the complete serial output shift register. Note that during a conversion, the conversion results are held in a temporary working register. Only at the end of the conversion are the results placed into the output shift register.

Figure 3 shows the timing diagram for the two ADS7809s through a single conversion. This diagram assumes that both converters receive the same R/$\overline{C}$ (read/convert) signal. Also, it assumes that both converters are completely finished with the conversion before the external clock is applied.

Note that the $\overline{BUSY}$ signal shown represents the logical AND of the $\overline{BUSY}$ signals from both converters. That is, $\overline{BUSY}$ is only HIGH when the $\overline{BUSY}$ signals from both converters are HIGH. In some circuits, the logical AND of the two $\overline{BUSY}$ signals will actually have to be generated. In others, this will not be the case and the signal shown in Figure 3 is simply for reference.

For some applications, there will be a lengthy delay between the start of conversion and the start of the external clock (DATACLK). This may be an artifact of the circuit design or the way software has been coded. The ADS7809 guarantees that the conversion will be complete within 8µs of the start of conversion. If the system design guarantees that this minimum delay is met, then the $\overline{BUSY}$ signals from the converters are not needed.

On the other hand, the clock generating the DATACLK signal may be continuously running. DATACLK will then be a gated version of this clock and is enabled by the logical AND of the two $\overline{BUSY}$ signals. A system can also be designed where DATACLK is triggered by the logical AND of the two $\overline{BUSY}$ signals. That is, DATACLK starts running after the $\overline{BUSY}$ signal shown in Figure 3 goes HIGH.

There is an important distinction between triggering and gating the DATACLK signal. Triggering means that the event (in this case, the worst-case $\overline{BUSY}$ going HIGH) causes a clean start of DATACLK. This clock might actually be the output of a microcontroller, a DSP, or a state-machine. Gating means that an external clock (such as a system clock) is continuously running and is routed through an AND gate with the event (again, in this case, the worst-

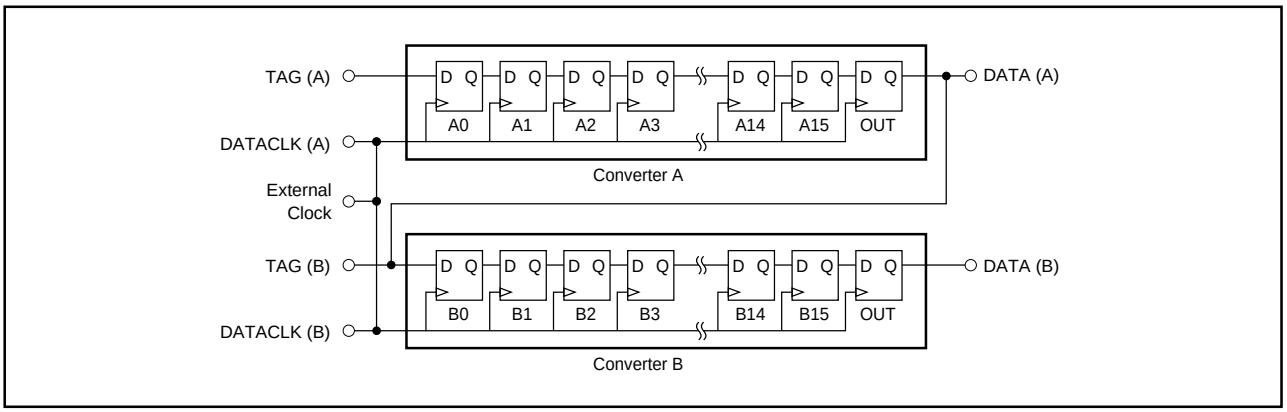

FIGURE 2. Simplified Diagram of the Internal Shift Registers of Two ADS7809s.

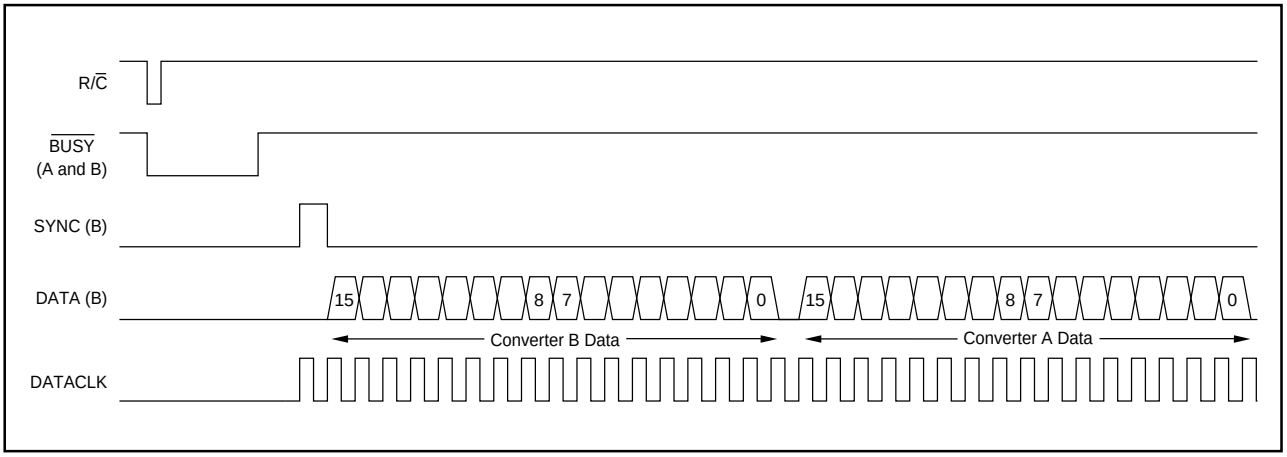

FIGURE 3. Timing of a Single Conversion for the Converters of Figure 1.

case $\overline{\text{BUSY}}$). The output of the AND gate could then be provided to the converters as the DATACLK input. This is shown in Figure 4.

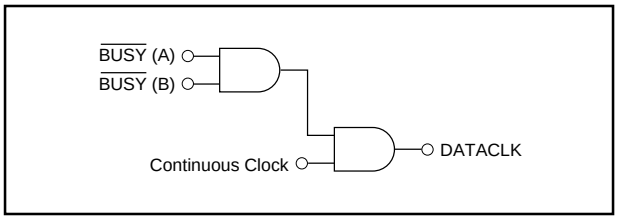

FIGURE 4. Gating a Continuous Clock with a Worst-Case $\overline{\text{BUSY}}$. (Not recommended.)

When a clock is gated in this manner, there is a risk that the event gating the clock may arrive just prior to the clock going LOW. The output of the AND gate will then be a very brief pulse. The risk is that one converter may respond to this clock while the other does not. This would put the two converters "out of sync" and corrupt the serial data.

Figure 5 shows a better method of gating the continuous clock to produce the DATACLK signal. A higher speed system clock is used to synchronize the continuous clock with the worst-case $\overline{\text{BUSY}}$ signal. Note that the system clock can be divided down by circuitry not shown to produce the continuous clock.

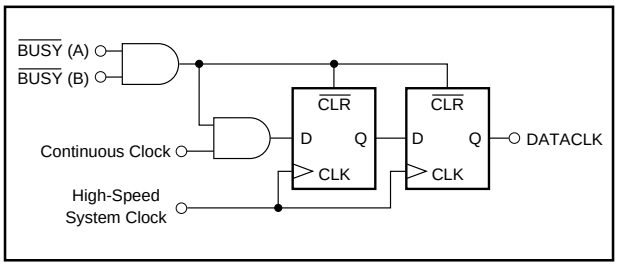

FIGURE 5. Gating a Continuous Clock with a Worst-Case $\overline{\text{BUSY}}$. (Recommended method.)

Figure 5 does not stop the DATACLK signal before the ADS7809s start their next conversion, and it is not a complete circuit as to how DATACLK should be generated from a continuous running clock. A complete circuit would feature a counter and/or state machine that would control the number of DATACLK cycles.

This may be a bit puzzling because once the next conversion starts, the two $\overline{\text{BUSY}}$ signals will go LOW—stopping the DATACLK signal. Unfortunately, this is not a good idea. The reason is that the DATACLK signal could transition just prior to the converters going into the hold mode (at the start of the conversion). If this occurs, the noise generated during the transition of the DATACLK pin could be coupled into the sampling capacitor, resulting in an error in the held voltage.

Figure 5 is NOT a problem if the start of the conversion is synchronous to the DATACLK signal and the time from the last DATACLK transition to the start of the conversion is long enough. This obviously needs further explanation. The first requirement is that the DATACLK signal be synchronous to the start of the conversion. For Figure 3, the conversion is initiated by R/$\overline{\text{C}}$ going LOW. If R/$\overline{\text{C}}$ is generated by the same master clock that generates DATACLK, then the two are synchronous.

For this case, DATACLK will transition at a known time before R/$\overline{\text{C}}$ goes LOW. If this period of time is long enough, then the noise generated by the DATACLK transition will have settled out before the converters go into the hold mode. The time for this to occur is longer for the 16-bit converters than for the 12-bit converters. Also, the time will depend on the layout, impedance of the analog signal source, and bypassing. For a good layout with a fast op-amp driving the analog signal, the time period might be as short as a few hundreds of nanoseconds.

For the 100kHz converters (the ADS7808 and ADS7809) and a reasonable layout, the time should be no longer than 1µs. For the 40kHz converters (the other family members), the time constant of the analog input is longer and so the time period might be as long as 2µs. In both cases, these are extremely conservative numbers.

DISCUSSION OF THE TIMING

Figure 3 shows the basic timing for when each data bit is available from the DATA output pin of converter B. The first rising edge of the external clock (DATACLK) causes the SYNC output on both converter A and B to go HIGH. No shifting of the internal shift register occurs at this point.

The SYNC output can be used with some DSP processors to start reception of serial data. However, after having gone HIGH during the first external clock period, the SYNC output will not return HIGH at the start of the data from converter A. This will limit the usefulness of the SYNC output in conjunction with the tag feature. For the rest of this document, the SYNC output will be ignored.

At this point (the first external clock period), the value of the DATA output pin is LOW. This value arises from the previous conversion and the fact that the TAG pin on converter B is grounded. On the very first conversion after power-up, the DATA output pin will be indeterminate. Also, the timing diagrams assume that the rising edge of the external clock will clock data into the receiving component. If the falling edge is used, then the very last clock is not needed. If this clock is not given, then the DATA output will reflect the value of bit 0 of the last result from converter A.

After the rising edge of the second external clock, the DATA output will reflect bit 15 (the MSB) of the conversion result. This second rising edge of DATACLK causes a shift in the shift register (refer to Figure 2). This also causes the bit 0

register to latch the value on the TAG input pin (connected to the DATA output pin of converter A). Since the TAG pin on converter A is grounded, the DATA output pin will be zero (indeterminate on the first conversion after power-up).

During the second external clock period, the DATA output on converters A and B reflect the MSB of the results of the previous conversion and the bit 0 register in converter A holds zero. The falling edge of the second external clock or the rising edge of the third clock can be used to latch data into the serial port on the receiving component. As mentioned previously, the timing diagram assumes the rising edge is used. If the falling edge is used, the very last clock that is shown is not needed.

The "dead bit" in between the two conversion results is probably the most confusing aspect of the ADS7809 tag timing. This, the SYNC output, and the "falling edge of clock n or the rising edge of clock n+1" can get very confusing. As shown so far, the conversion results from converter B are valid on the 3rd rising edge of the external clock through the 18th rising edge. On the 19th rising edge, the "dead bit" is valid. On the 20th through the 35th clock, the conversion results from converter A are valid. All results are MSB first.

If the component receiving this serial stream clocks on the falling edge, then subtract 1 from the discussion above. The 35th clock is not needed in this case.

MORE THAN TWO CONVERTERS

If this is clear, then the extrapolation to more than two converters should be easy. Daisy chain the DATA outputs and TAG inputs of the converters. After the first conversion result, each subsequent result will be available after one "dead bit" between it and the previous result. Tie the last TAG input LOW or HIGH depending on any preference for the "resting" state of the DATA outputs and internal shift registers. Do not let the last TAG input float.

Figure 6 shows the timing of the conversions and serial clocking in a broader perspective. There are several important items to note in this diagram.

Time t_1 is the conversion time (worst case). Time t_2 is the time it takes to clock out the conversion results. Time t_3 represents the amount of time given to the converters to acquire the input signal for the next conversion. This diagram is not to scale. Time t_2 and t_3 can be overlapped slightly to optimize the conversion throughput.

The maximum throughput (conversions/second) can be determined from Figure 6 and the ADS7809 data sheet. The data sheet guarantees that a conversion will take no longer than 8µs. It also guarantees that the converter can acquire the signal to the precision needed in under 2µs. Finally, the smallest external clock period that can be used for proper operation is 100ns (10MHz).

From Figure 6, a single conversion and data output cycle is the sum of t_1, t_2, and t_3 plus a small amount of time between t_1 and t_2. This time, shown much larger than needed, allows for a small amount of time between $\overline{BUSY}$ going HIGH and the start of the external clock. If an external 10MHz clock and a state machine were used to generate R/$\overline{C}$, the 8µs delay, and the external 10MHz serial clock, then the time between t_1 and t_2 could be zero (because everything is synchronous).

Also, the number of serial clocks needed to clock the data is n•17+1 where n is the number of converters. The time required to clock the data through is obviously the result of this calculation times the clock period. In the example so far, this time would be 3.5µs.

Finally, it is possible to overlap the clocking of data from conversion n with the acquisition of the analog signal for conversion n+1. This would effectively overlap time t_2 and t_3. However, a complete overlap is not recommended. It is possible to affect the held voltage by clocking the converter just prior to going into the hold mode (as was previously noted). The external clock should be turned off at least 500

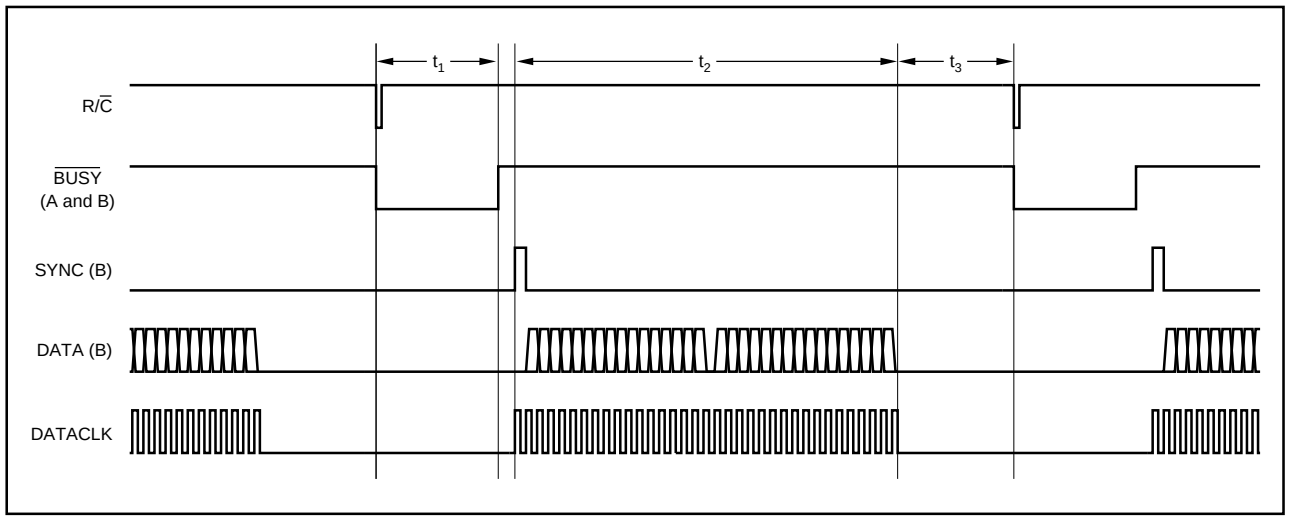

FIGURE 6. Multiple Conversions for the Converters Shown in Figure 1.

nanoseconds prior to R/$\overline{C}$ going LOW. A more conservative number would be 1µs. (Again, this would be longer for the slower family members—those that run at 40kHz instead of 100kHz. For these converters, 1µs is recommended as a minimum and 2µs would be a more conservative number.)

So, the maximum throughput with the gated-clock method of using the tag feature would be:

$$8\mu s+(\text{clock period})\ (n\bullet 17+1)+0.5\mu s$$

For a 10MHz clock and two converters, this would be 12µs (83kHz).

A NOTE ABOUT MULTIPLEXING

As a side note, if the ADS7809s are to be used in conjunction with multiplexers, then the best time to switch the input channels would be just after the end of the conversion (the beginning of time t_2). This is provided that the input channel can be switched and the new channel completely settled when a new conversion begins.

If this is not the case, the second best time is just after the start of time t_1. There is no evidence that a changing analog signal at the input of the ADS7809 while it is converting causes any feedthrough problems. However, switching the channel at the beginning of the conversion insures that this change occurs when the most significant bits are being decided and not the least significant bits (when the converter is more sensitive to external noise).

A NOTE ABOUT LAYOUT

Since the internal conversion clock of the ADS7809 is generated on-board, each converter in a multiple converter system will be converting at its own speed. This creates the potential for one converter to interfere with a nearby one if care is not taken in the layout.

The best layout for a multiple converter system starts with the recommendations for a single converter system: good grounding practices with a ground plane strongly recommended; clean, low-impedance +5V supply; good bypassing; and adequate, low-impedance signal-conditioning prior to the converter input. Multiple converter systems should pay more attention to the grounding scheme and the +5V supply. A solid ground plane which is continuous between one converter and the next is not recommended. Instead a "slot" of the ground plane should be removed between converters or the ground plane should extend "fingers" to each converter and signal conditioning circuitry separately.

For the +5V power, the recommendation for a single ADS7809 is to run both the analog and digital +5V pins from the same +5V supply. For multiple converters it may be best to split the pins between two +5V supplies. This will prevent the digital clocking of one converter from affecting another. However, care must be taken to insure that the digital supply comes up after the analog supply to prevent latch-up, and the digital +5V supply should still be fairly clean. Another possible technique would be to generate +5V for each converter separately and continue to tie both analog and digital supply pins together. For designs where the +5V will be generated by a voltage regulator, this might be an acceptable, simple solution.

OTHER WAYS TO USE THE TAG PINS

Another method of using the tag pin involves a gated clock that is active just after the conversion has started. This essentially takes time t_2 (see Figure 6) and places it inside time t_1. Clocking the converters in this manner has the advantage of not affecting the maximum throughput of the converters (10µs). The disadvantage is that there is a very good chance of affecting the conversion results due to the noise generated inside the converter by the serial data stream and clock.

It is impossible to predict to what degree the converter will be affected by clocking data through it as it is doing a conversion. Each layout and circuit is unique, and DATACLK will have a different rise and fall time. It is known that the conversion results WILL be different. Rough "guesstimates" predict that the transition noise will be increased, possibly up to many tenths of an LSB (for the 16-bit converters). For some systems with poor layout or very high frequency clocks, this number could be substantially higher. The good news is that the 12-bit members of the family should work acceptably as their "LSB size" is 16 times greater.

Finally, there are two other methods that involve a continuous running clock (see Figure 7). This diagram is similar to that in Figure 6 where the data is clocked after the conversion is complete. The other method (not shown) involves a continuous clock which clocks out data just after the start of conversion, as just discussed for the gated clock. Both of these methods enable the transmission of data from the ADS7809 by controlling R/$\overline{C}$ and $\overline{CS}$.

The continuous clock mode and clocking data after the data is complete (as shown in Figure 7) is the most conservative approach after the gated-clock mode of Figure 6. However, this method affects the conversion throughput in exactly the same manner as for the gated clock mode. While the method does not require that DATACLK be gated, it does require a $\overline{CS}$ signal. Similar concerns regarding a short clock period apply if the continuous clock is free-running with respect to the $\overline{CS}$ signal. This means that the $\overline{CS}$ signal must be synchronized with the DATACLK signal. This could be done with a circuit similar to that shown in Figure 5 (but a "synchronized" $\overline{CS}$ signal will be the output of the latches).

The method shown in Figure 7 is better than clocking data during a conversion because only one digital signal is being transitioned. On the other hand, a continuous running clock that clocks the data during a conversion (time t_1) is the worst possible method of using the tag pin. The following is a ranking in order of preference for the various methods outlined in this application bulletin:

- Gated Clock reading the data after the conversion is complete
- Continuous clock reading data after the conversion is complete
- Gated Clock reading the data during the next conversion
- Continuous clock reading the data during the next conversion

Note that for the last two cases, it will help if the data is clocked through the converters as quickly as possible. In this way, the DATA signal will transition to a continuous LOW signal (remember, the TAG pin on converter A is tied to ground). Also, there is a very important consideration regarding the last two methods. For all the converters, if $\overline{CS}$ is LOW, R/$\overline{C}$ HIGH, and DATACLK is HIGH when a conversion is finished, then the data from that conversion is lost. A conversion finishes just prior to $\overline{BUSY}$ rising.

For the ADS7806 and ADS7807, a conversion may finish between 12µs and 20µs after the conversion is initiated. Actually, the time is closer to 12µs for the ADS7806 (a 12-bit converter) and closer to 20µs for the ADS7807 (the same converter, but with 16-bits of resolution). The 12µs to 20µs "window" was established so that both parts would operate correctly in the same design. The ADS7824 and ADS7825 have a similar window. The ADS7808 and ADS7809 window is 5µs to 8µs.

This raises a further concern with these last two methods (where the data is clocked out during a conversion). As was calculated previously, the fastest that the data can be clocked out of the circuit in Figure 1 is 3.5µs. So, if there is only 5µs available in which to clock the data, then the circuit of Figure 1 is very close to the limit of cascading multiple converters. In fact, only three ADS7809s can be cascaded under these conditions, and then only if the DATACLK signal runs at 10MHz (the maximum rate). For a 5MHz DATACLK signal, the circuit of Figure 1 would not work.

The situation is better for the ADS7806 and ADS7824 (12-bit, 40kHz converters). However, even for any two of these converters, the DATACLK signal must be faster than 2.16MHz. For the ADS7807 and ADS7824 (16-bit, 40kHz converters) the minimum DATACLK is 2.9MHz.

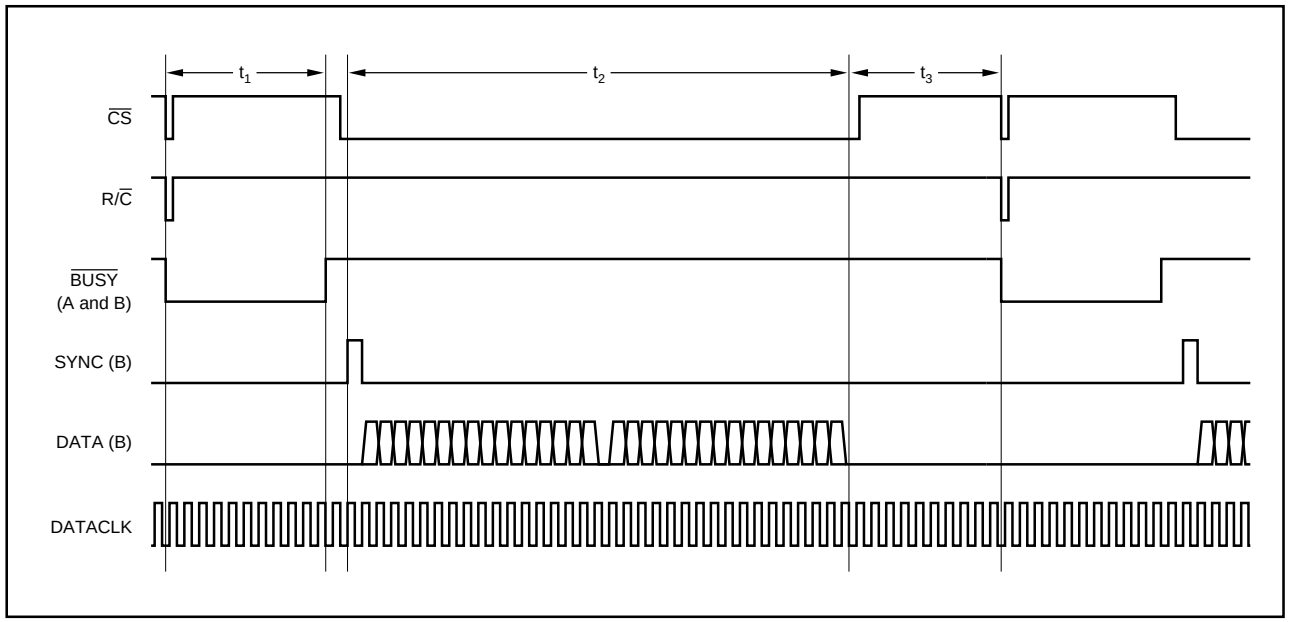

FIGURE 7. Continuous Clocking of the Converters Shown in Figure 1 with $\overline{CS}$ used to Enable Data Transmission.