The wait logic

The Z80 allows wait states to be inserted between the T2 and T3 cycles. Processors provide wait states in case the memory chip is too slow for the CPU. By adding wait states, the memory has additional time to perform its operation. During this time, the CPU just waits. This allows cheaper memory to be used in a system.

For modern CPUs, the difference between CPU and memory speed is much more dramatic than a couple wait states. For instance, Intel gives the approximate cost of a DRAM access as ~100 ns.
https://software.intel.com/sites/products/collateral/hpc/vtune/performance_analysis_guide.pdf
At 4.4 GHz, this is about 440 cycles cost if the CPU needs to access DRAM. This explains why modern CPUs use multiple levels of caches to avoid most memory accesses.

The WAIT pin allows wait states to be generated externally. This pin is active-low, so a low value indicates a wait state is required. This pin is sampled at the falling edge of the clock during T2 (and any later wait states). If it is low, a wait state is added.

The circuit to do this can be seen below. The input pin is clamped with a diode (like the other input pins) to ensure it doesn't fall below ground. It goes through a couple inverters to buffer it, and then is fed into a latch. While the clock is high, the latch reads the input value, and when the clock is low it keeps its previously latched value. Thus, the latch holds whatever value the pin had at the falling edge of the clock. (But while the clock is high, the output of the latch can change if the input on the WAIT pin changes.)

The next latch reads its input while the clock is low, and holds the latched value while the clock is high, the opposite of the previous latch. Note that this latch will ignore any changes from the previous latch while the clock is high.) Thus, the latch will read the stable value from the previous latch at the falling edge of the clock, continue reading while the clock is low, and then latch the value when the clock is high.

The result of these two latches is the input value of the pin is sampled at the falling edge of the clock and appears with a stable value at the output until the next falling edge of the clock.

As discussed in the timing chapter, this wait signal blocks the entry into T3 from T2. Instead, a latch keeps track of the pending T3 state until the wait signal becomes inactive.

Like the second wait latch, this latch reads a value while the clock is low and latches while the clock is high. It is fed from a pass transistors that is active while the clock is high. Thus, this circuit operates in two phases and remains stable without oscillating. Specifically, while the clock is high, the latch's inputs are set up. As soon as the clock goes low, the latch's inputs are kept from changing (by the pass transistor), the latch grabs its inputs, and the latch's outputs can change. The effect is that the latch grabs the value on the falling edge of the clock. However, it is implemented by two level-sensitive components, not an edge-sensitive component. This prevents any difficulties due to the clock edge reaching different parts of the system at different times.

To understand how the T3-generation logic works, the first gates can be considered:
WANT_T3_NEXT = (IN_T2 or WANT_T3) and not IN_T3

That is, a T3 state is wanted if the current state is T2, or a T3 state was wanted last cycle, as long as the processor isn't in a T3 state.

The W174 new_t_state signal indicates that it is time to move to a new T state. This is fed into the T3 latch when the clock is low. The output of the T3 latch will be blocked if the wait signal is high. In this case, a wait state happens, and the WANT_T3 latch holds the desired T3 state until next time. The T2 state will be exited, so the processor will be in neither T2 nor T3 - this is what a wait state looks like internally.

The other way a wait state can be generated, besides the WAIT pin, is through W186, which indicates an internally-generated wait state. This signal comes from three paths:

1. interrupt activated in M1T2
2. interrupt activated in M1T2, delayed by one clock through the latch
3. insert_inout_wait in T2

insert_inout_wait is triggered for :
M2 for pla21 (inx/inxr)
M3 for pla20 (outx/outxr)
M4 for pla27 (in/out) or pla37 (out *,a/a,*)

Thus, this circuit provides the two wait states for an interrupt and the one wait state for an I/O.

The wait state circuitry is fairly self-contained. According to the timing diagrams, wait states are inserted between T2 and T3. Internally, this is pretty much what happens - the processor leaves T2 but doesn't enter T3 while the wait state is happening. The wait state can either be triggered externally by the WAIT pin, or internally as part of interrupt processing or I/O.