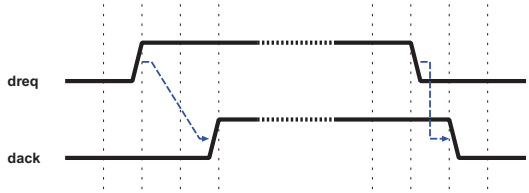
Q11

The timing diagram below is mentioned in the slides.



The following two handshake properties were given:

- following a rising edge on `dreq`, the value of `dreq` remains 1 (i.e. *true*) until it is acknowledged by a rising edge on `dack`
- following a falling edge on `dreq`, the value on `dreq` remains 0 (i.e. *false*) until the value of `dack` is 0

Formalise these two properties as formulae in a suitable temporal logic. You should state what logic you are using and briefly describe why you chose it.

Solution

I will use PSL because it is more readable, though LTL would do. If we interpret the informal text as requiring that a `dack` must always occur, the properties are:

```
always {!dreq;dreq}([dreq until! dack])
always {dreq;!dreq}([!dreq until! !dack])
```

This is a strong until ([`dreg` **U** `dack`] in LTL). If we don't require an acknowlegement to actually happen, then a weak until would be used:

```
always {!dreq;dreq}([dreq until dack])
always {dreq;!dreq}([!dreq until !dack])
```

This is [`dreg` **W** `dack`] in LTL.

Note that the timing diagram, but not the informal descriptions given as bullet points, suggests that `dack` should be low during the rising edge and high during the falling edge. This could be reflected, for the strong interpretation, by:

```
always {{!ack;!ack}&&{!dreq;dreq}}([dreq until! dack])
always {{ack;ack}&&{dreq;!dreq}}([!dreq until! !dack])
```