QP State transitions

原文：http://www.rdmamojo.com/2012/05/05/qp-state-machine

QP=Queue Pair. (Introduction to RDMA Programming. pp25-26. RDMA access model).

In a QP lifetime, the possible states can be:

* Reset
* Initialize (Init)
* Ready To Receive (RTR)
* Ready To Send (RTS)
* Send Queue Drained (SQD)
* Send Queue Error (SQE)
* Error

A QP can be transitioned from one state to another state by two possible ways:

1. Explicit call to ibv\_modify\_qp()
2. Automatic transition by the device in case of a processing error

A QP is being created in the Reset state.

Any QP can be transitioned to Reset or Error state by calling ibv\_modify\_qp() from any state.

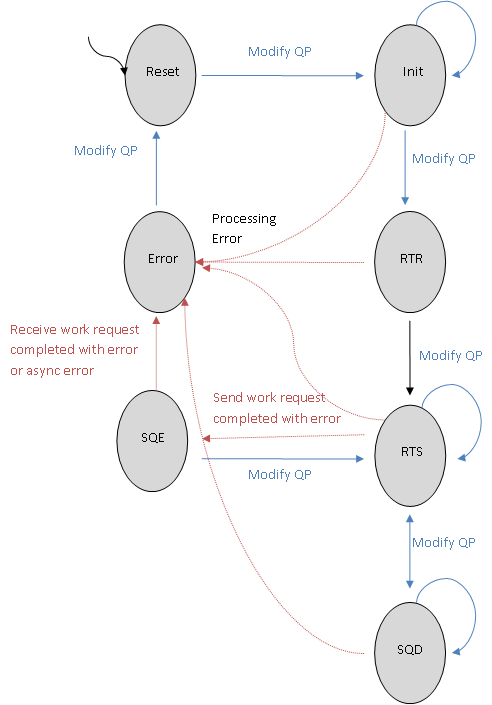


Figure QP State Machine

# Reset state

## Description

A QP is being **created** in the Reset state. In this state, all the needed resources of the QP are already allocated.

In order to reuse a QP, it can be transitioned to Reset state from any state by calling to ibv\_modify\_qp(). If prior to this state transition, there were any Work Requests or completions in the send or receive queues of that QP, they will be **cleared** from the queues.

## Work Requests

Work Requests shouldn't be posted to either send or receive queues. Trying to do so will result in an immediate error.

Work Requests in both send and receive queues won't be processed.

## Packets

Incoming packets which are targeted to this QP will be silently dropped.

No packets will be sent from this QP.

# Init state

## Description

In order to move to this state, one must call ibv\_modify\_qp() from Reset state.

## Work Requests

Work Requests may be posted **only to the receive queues**. Trying to post Work Requests to the send queue will result in an immediate error.

Work Requests in both send and receive queues won't be processed.

## Packets

Incoming packets which are targeted to this QP will be silently dropped.

No packets will be sent from this QP.

# Ready To Receive(RTR) state

## Description

In order to move to this state, one must call ibv\_modify\_qp() from the Init state.

In RTR state, the QP may be used only as a responder.

## Work Requests

Work Requests can be posted only to the receive queues. Trying to post Work Requests to the send queue will result in an immediate error.

Only Work Requests in the receive queue will be processed.

## Packets

Incoming packets to this QP will be served. The QP will send packets from this QP as a responder (i.e: data, acks and nacks).

# Ready To Send(RTS) state

## Description

In order to move to this state, one must call ibv\_modify\_qp() from RTR or SQD states.

In RTS state, the QP may be used as a requester or as a responder.

## Work Requests

Work Requests can be posted to both send and receive queues.

Work Requests in both send and receive queues will be processed.

## Packets

Incoming packets to this QP will be served. The QP will send packets from this QP as a responder (i.e: data, acks and nacks).

The QP may initiate packets as a requester.

# Send Queue Drained(SQD) state

In order to move to this state, one must call ibv\_modify\_qp() from the RTS state.

This state affects only the send queue: Work Requests that their processing have started will be handled until fully completed. However, new Work Requested won't be processed.

This state is separated into two internal states:

**Draining** - there are still Work Requests in the send queue that are being handled

**Drained** - there aren't any Work Requests in the send queue that are being processed

One can the state of that QP to RTS or SQD only if the internal state of the QP is Drained.

## Work Requests

Work Requests can be posted to both send and receive queues.

Work Requests in the receive queue will be processed.

Work Requests in the send queue that their processing have started will be completed, new Work Requests won't be processed.

## Packet

Incoming packets to this QP will be served. The QP will send packets from this QP as a responder (i.e: data, acks and nacks).

The QP may initiate packets as a requester.

# Send Queue Error(SQE) state

## Description

Transition to this state will happen automatically by the device for all QP types, except for RC QPs, that were in the RTS state and processed a Work Request in the Send Queue that ended with completion error. The status of this Work Request is undefined (it is unknown if the Work Request has been partially or fully handled) and the content of the local or the remote buffers (depends on the used opcode) is undefined.

All subsequent Work Request in the send queue will be flushed with error.

One may call ibv\_modify\_qp() to move back to the RTS state to recover the Send Queue.

## Work Requests

Work Requests can be posted to both send and receive queues.

Only Work Requests in the receive queue will be processed.

Work Requests to the send queue will be flushed with error.

## Packets

Incoming packets to this QP will be served.

The QP will send packets from this QP as a responder (i.e: data, acks and nacks).

# Error state

## Description

This is the last state in the state machine of a QP.

A QP may be transitioned to this state automatically by the device, if a Work Request in the send queue was completed with error for an RC QP or if a Work Request in the receive queue was completed with error for any QP type, or by explicitly calling ibv\_modify\_qp() from any state.

If the QP state was transitioned automatically, the status of this Work Request is undefined (it is unknown if the Work Request has been partially or fully handled) and the content of the local or the remote buffers (depends on the used opcode) is undefined.

## Work Requests

Work Requests can be posted in both send and receive queues.

If the QP was transitioned to this state automatically, the first Work Request that completed with error will indicate the reason for the error. All subsequent Work Requests in that queue and all Work Request in the other queue and new Work Requests posted to both of the queues will be flushed with error.

If the QP was transitioned to this state using ibv\_modify\_qp() all outstanding Work Request in both send and receive requests will be flushed with error.

## Packet

Incoming packets which are targeted to this QP will be silently dropped.

No packets will be sent from this QP.

# Summary

In this post, we covered the various states of a QP and the behavior of that QP.

The following table summarizes the behavior of a QP depends on its state:

Table QP State behavior table (part 1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Reset | Init | RTR | RTS |
| Post receive request | Disallowed | Allowed | Allowed | Allowed |
| Post send request | Disallowed | Disallowed | Disallowed | Allowed |
| Receive Request processing | Not processed | Not processed | Processed | Processed |
| Send request processing | Not processed | Not processed | Not processed | Processed |
| Incoming packets | Silently dropped | Silently dropped | Handled | Handled |
| Outgoing packets | None | None | None | Initiated |

Table QP State behavior table (part 2)

|  |  |  |  |
| --- | --- | --- | --- |
|  | SQD | SQE | Error |
| Post receive request | Allowed | Allowed | Allowed |
| Post send request | Allowed | Allowed | Allowed |
| Receive Request processing | Processed | Processed | Flushed with error |
| Send request processing | New WRs aren't  processed | Flushed  with error | Flushed with error |
| Incoming packets | Handled | Handled | Silently dropped |
| Outgoing packets | Initiated | None | None |

In the next post I will describe how to use the various QP states.