# Passer

* Check
  + Interruptions
  + Shutdowns
* Check
  + to see if another thread hasn’t already claimed the passer spot
* Claim
  + the passer spot
* Indict
  + the passer is ready

## Handshake

1. If the receiver is already ready > don't worry about the passer’s timeout
   1. set the message
   2. let the receiver know that the message has been set
   3. since the handshake has begun, indict that the receiver isn’t ready to prevent **this passer** from being interrupted and have their spot being stolen
   4. notify the receiver that the passer is ready
   5. wait until the receiver has received the message
      1. check for interruptions and shutdowns
2. If the passer doesn’t have a timeout
   1. wait forever until the receiver is ready
      1. check for interruptions and shutdowns
   2. receiver is ready now, set the message
   3. let the receiver know that the message has been set
   4. since the handshake has begun, indict that the receiver isn’t ready to prevent **this passer** from being interrupted and have their spot being stolen
   5. notify the receiver that the passer is ready
   6. wait until the receiver has received the message
      1. check for interruptions and shutdowns
3. If the passer has a timeout
   1. initialize the max time and the timeout
   2. wait until the receiver is ready and that time hasn’t run out
      1. check for interruptions and shutdowns
      2. recalculate the remaining time to end time minus the time already spent
   3. If the receiver is ready
      1. set the message
      2. let the receiver know that the message has been set
      3. since the handshake has begun, indict that the receiver isn’t ready to prevent **this passer** from being interrupted and have their spot being stolen
      4. notify the receiver that the passer is ready
      5. wait until the receiver has received the message
         1. check for interruptions and shutdowns
   4. If the receiver isn’t ready
      1. exit because the max time has been reached, timeout
4. Regardless if the handoff is complete or not
   1. Reset all the fields set by this method

# Receiver

* Check
  + Interruptions
  + Shutdowns
* Check
  + to see if another thread hasn’t already claimed the receiver spot
* Create
  + an instance to hold the message
* Claim
  + the receiver spot
* Indict
  + the receiver is ready

## Handshake

1. If the passer is already ready > don’t worry about the passer’s timeout
   1. notify the passer that the receiver is ready
   2. wait until the message is set by the passer
      1. check for interruptions and shutdowns
   3. since the handshake has begun, indict the passer isn’t ready to prevent **this receiver** from being interrupted and have their spot being stolen
   4. set the instance equal to the message
   5. let the passer know that the message has been received
   6. notify all the waiting threads that the handshake has been completed
2. If the receiver doesn’t have a timeout
   1. wait forever until the passer is ready
      1. check for interruptions and shutdown
   2. since the handshake has begun, indict the passer isn’t ready to prevent **this receiver** from being interrupted and have their spot being stolen
   3. notify the passer that it can set the message
   4. wait until the message is set by the passer
      1. check for interruptions and shutdowns
   5. set the instance equal to the message
   6. let the passer know that the message has been received
   7. notify all the waiting threads that the handshake has been completed
3. If the receiver doesn’t have a timeout
   1. wait forever until the passer is ready
      1. check for interruptions and shutdown
   2. notify the passer that the receiver is ready
   3. wait until the message is set by the passer
      1. check for interruptions and shutdowns
   4. since the handshake has begun, indict the passer isn’t ready to prevent **this receiver** from being interrupted and have their spot being stolen
   5. set the instance equal to the message
   6. let the passer know that the message has been received
   7. notify all the waiting threads that the handshake has been completed
4. If the receiver has a timeout
   1. initialize the max time and the timeout
   2. wait until the passer is ready and that time hasn’t run out
      1. check for interruptions and shutdowns
      2. recalculate the remaining time to end time minus the time already spent
   3. If the passer is ready
      1. since the handshake has begun, indict the passer isn’t ready to prevent **this receiver** from being interrupted and have their spot being stolen
      2. notify the passer that it can set the message
      3. wait until the message is set by the passer
         1. check for interruptions and shutdown
      4. set the instance equal to the message
      5. let the passer know that the message has been received
      6. notify all the waiting threads that the handshake has been completed
5. If the receiver has a timeout
   1. initialize the max time and the timeout
   2. wait until the passer is ready and that time hasn’t run out
      1. check for interruptions and shutdowns
      2. recalculate the remaining time to end time minus the time already spent
   3. If the passer is ready
      1. wait until the message is set by the passer
         1. check for interruptions and shutdowns
      2. since the handshake has begun, indict the passer isn’t ready to prevent **this receiver** from being interrupted and have their spot being stolen
      3. set the instance equal to the message
      4. let the passer know that the message has been received
      5. notify all the waiting threads that the handshake has been completed
   4. If the passer isn’t ready
      1. exit because the max time has been reached, timeout
6. Regardless if the handoff is complete or not
   1. Reset all the fields set by this method
7. Return the instance