Connected

```
entry/ start m_dataPollTimer(once)
exit/ stop m_dataPollTimer

WIFI_WRITE_REQ
/ while (m_dataOutFifo not empty)
     writeLen = m_dataOutFifo->GetUsedBlockCount(),
     writeLen = MIN(MAX_SEND_LEN, writeLen),
     ES_WIFI_SendData(m_dataOutFifo->GetReadAddr(), writeLen),
     if (send data failed) failed = true, break
     m_dataOutFifo->IncReadIndex(writeLen)
  if (failed) Raise(DISCONNECTED)
  else Raise(WIFI_EMPTY_IND)

DATA_POLL_TIMER
/ totalLen = 0;
  do
    readLen = m_dataInFifo->GetAvailBlockCount(),
    if (readLen)
      ES_WIFI_ReceiveData(m_dataInFifo->GetWriteAddr(), readLen, &recvLen),
      if (receive data failed) failed = true, break
      m_dataInFifo->IncWriteIndex(recvLen);
      totalLen += recvLen;
  while(readLen && (recvLen == readLen))
  if (failed) Raise(DISCONNECTED)
  else if (totalLen) Send(WIFI_DATA_IND, m_client)
  start dataPollTimer(once)
```

WIFI_WRITE_REQ does not require CFM.

To avoid hogging may send reminder before out fifo is emptied.

GetUsedBlockCount() returns the max length of the next contiguous used block.

GetAvailBlockCount() returns the max length of the next contiguous available block

Note - Cannot check "wasEmpty" to skip sending WIFI_DATA_IND since receive data function is not called in crit-sect.

WIFI_READ_MORE_REQ triggers Wifi to read more data when the m_dataInFifo is emptied by the client. It is not used since it uses polling.

dataPollTimer is (re)started "once" each time to avoid event queue from overflowing when ES_WIFI_SendData() blocks for a few seconds.