

## Testübersicht: OSMP Kommunikations- und Synchronisationstests

Diese Datei gibt einen strukturierten Überblick über alle implementierten Tests im OSMP-Projekt. Sie erklärt, was jeder Test macht, welches Ziel er verfolgt und wann der Test als **erfolgreich** bzw. **fehlgeschlagen** gilt.

---

### ✅ testSendRec.c – Minimaler Funktionstest (2 Prozesse)

- **Was macht der Test?**
    - Prozess 0 sendet eine Zeichenkette ("ABCDE...") an Prozess 1
    - Prozess 1 empfängt und gibt die Nachricht aus
  - **Ziel:**
    - Grundfunktionalität von OSMP\_Send() und OSMP\_Recv()
    - Kommunikation mit genau 2 Prozessen
  - **Richtig:** Nachricht empfangen, Ausgabe korrekt
  - **Fehler:** Keine Ausgabe, Abbruch bei Init, Send oder Recv
- 

### 🔄 testSendRecMulti.c – Ring-Kommunikation (mehrere Prozesse)

- **Was macht der Test?**
    - Jeder Prozess sendet 5 Nachrichten an den nächsten Prozess  $(\text{rank} + 1) \% \text{size}$
    - Jeder empfängt 5 Nachrichten vom Vorgänger  $(\text{rank} - 1 + \text{size}) \% \text{size}$
  - **Ziel:**
    - Gleichzeitiges Senden und Empfangen
    - Slot- und Mailboxnutzung testen
  - **Richtig:** Jeder Prozess sendet und empfängt 5 Nachrichten
  - **Fehler:** Hängen bei Send, fehlende Recv-Nachrichten
- 

### 🔔 testSendRecBroadcast.c – Broadcast-Test

- **Was macht der Test?**
  - Rang 0 sendet eine Nachricht an **alle anderen Prozesse**

- Alle anderen empfangen genau eine Nachricht
  - **Ziel:**
    - Funktionierender Einzelversand von Rang 0 an n Empfänger
    - Belastungstest für OSMP\_Send()-Schleife
  - **Richtig:** Alle Ränge > 0 geben exakt eine Nachricht aus
  - **Fehler:** Nicht alle Prozesse empfangen; doppelte Nachrichten
- 

### testSendRecBlock.c – Blockierungstest

- **Was macht der Test?**
    - Rang 0 empfängt verspätet (schläft 5 Sekunden)
    - Alle anderen Prozesse senden 5 Nachrichten **an Rang 0**
  - **Ziel:**
    - Überprüfung: Blockiert OSMP\_Send(), wenn die Mailbox voll ist?
    - Testet korrekte Semaphorennutzung für Slots und Puffer
  - **Richtig:** Sender warten korrekt, Empfänger holt alle Nachrichten ab
  - **Fehler:** Abbruch in Sendern (z. B. Send schlägt fehl), Slotverlust
- 

### testSendRecRace.c – Race-Condition-Test

- **Was macht der Test?**
  - Jeder Prozess startet **zwei Threads**:
    - Einer sendet 5 Nachrichten
    - Einer empfängt 5 Nachrichten
- **Ziel:**
  - Gleichzeitiger Zugriff auf alle internen OSMP-Strukturen
  - Testet Robustheit der Synchronisation (z. B. Slotverwaltung, Mailboxmutex)
- **Richtig:** Jeder Thread sendet/empfängt ohne Absturz, Daten korrekt
- **Fehler:** Crash, Race-Warnung (z. B. mit ASAN oder Valgrind), falsche Nachrichtenreihenfolge

---

## Zusammenfassung

Testname	Fokus	Fehlerindikatoren
testSendRec	Basisfunktion	Kein Empfang, Exit-Code $\neq 0$
testSendRecMulti	Gleichzeitiges Senden/Empfangen	Hänger, doppelte/fehlende Nachrichten
testSendRecBroadcast	Mehrfach-Senden	Prozesse empfangen nichts/mehrfach
testSendRecBlock	Blockierverhalten	Sender brechen ab oder hängen ewig
testSendRecRace	Paralleler Zugriff	ASAN/Valgrind-Warnung, Slotverlust, Crash

---

Diese Tests decken gemeinsam **alle OSMP-Kernaspekte** ab: Kommunikation, Blockierung, Synchronisation und Fehlerszenarien.  
Ergänzungen möglich durch Gather- oder Barrier-Tests.