



# TIGERS Mannheim



# Open Source/Hardware

Release 2015





# Released Content (New)

- Central software (Saumatra)
  - Incl. Filtering, AI, skills, robot communication
- Modified grSim simulator
- Robot Firmware
  - Fast nRF24L01+ protocol
  - Sensor Fusion (vision, encoders, gyro, accelerometer)
  - Trajectory Generation
  - Various tools (usable in any C/C++ program)



# Released Content (old)

- Mechanical
  - full CAD files for Autodesk Inventor
  - drawings
- Electrical
  - robot mainboard
  - Kickerboard
  - Mediaboard
  - base station (2x nRF24L01+ and Ethernet connection)



# Sumatra

- Includes detailed How-To to start Sumatra and to kick off a match in grSim!
- → Demo!
- Auto referee, just let the game go and watch.
- Modified ball model





# Robot Firmware - Wireless

- High-Speed wireless protocol implementation, 2MBit/s (raw), ~240kBit/s (user data) → 30B every milliseconds (RX/TX)
- TDMA access pattern for up to 12 robots on one channel (RX/TX)
- Using „Consistent Overhead Byte Stuffing“ (COBS) for reliable packet start identification
- Recognizes RX/TX loss by sequence number → link quality indication

→ `lib/common/src/f4xx/util/nrf24_ex.h/.c`

- Reliable communication layer with re-transmission on top of nRF24L01+ library

→ `app/main/src/network.h`



# Robot Firmware – Sensor Fusion

- Advanced onboard filtering algorithms combining:
    - Vision (global position)
    - Encoders (local velocity)
    - Gyroscope (angular velocity)
    - Accelerometers (local velocity)
- Kalman Filter
- `app/main/src/ctrl/ctrl_tigga.h`
- `lib/common/src/util/kf.h`





# Robot Firmware Trajectory Generation

- Bang-Bang trajectory generation 1D
  - With different acceleration and deceleration values
  - Arbitrary input position and velocity. Speed == 0 at final position.
- Bang-Bang trajectory generation 2D
  - Synchronizes two 1D trajectories with a binary search → fixed runtime
- Orientation trajectory generation
  - Uses a 2D trajectory as a limit function
    - Fast passages on XY → none/slow rotation
    - Slow passages on XY → fast rotation
    - Remaining energy is properly used without exceeding limits
- Full 2D+Orient trajectory generation in under 50us on  $\mu\text{C}$   
→ `lib/common/src/util/trajectory*.h`



# Robot Firmware Tools

- arm\_math variants for
  - Pseudo-inverse (Moore-Penrose Inverse)
  - Singular Value Decomposition
  - Cholesky Decomposition
  - Fast 2x2, 3x3 matrix inverse
  - Fast LUT-based atan2→ lib/common/src/util/arm\_mat\_util\_f32.h
- Consistent Overhead Byte Stuffing  
→ lib/common/src/util/cobs.h







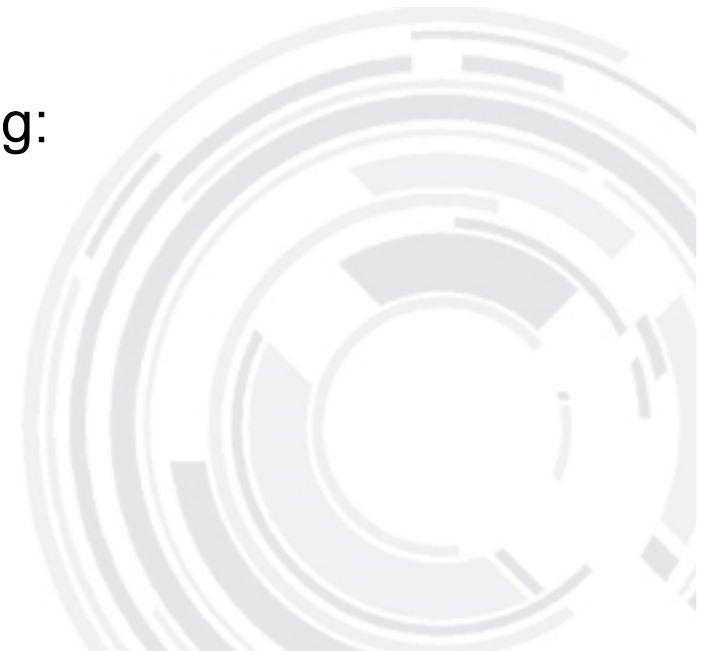
# Robot Firmware Tools

- Filtering and Sensor Fusion:
  - Kalman Filter (linear)  
→ lib/common/src/util/kf.h
  - Unscented Kalman Filter (non-linear)  
→ lib/common/src/util/ukf.h
- Control:
  - Luenberger Observer  
→ lib/common/src/luenberger.h
  - PID (fixed-time, with input/output limits)  
→ lib/common/src/util/pid.h
  - Lag element (PT1, fixed-time, tustin-transformed)  
→ lib/common/src/util/lag\_element.h



# Robot Firmware Tools

- Advanced FIFO structures:
  - Linear FIFO, data is always linearly addressable  
→ `lib/common/src/util/fifo_lin.h`
  - Block FIFO, buffering entity with equally spaced blocks  
→ `lib/common/src/util/fifo_block.h`
- Fast UDP/IP stack for STM32F4 including:
  - ARP
  - IPv4
  - ICMP
  - UDP
  - IGMPv3 (multicast)





# Download

<https://tigers-mannheim.de/download/release2015/>

