# TWEET-IE Management Workshop

## Management of a model building lab in support of a wind tunnel

Carlo L. Bottasso, Franz V. Mühle,

Filippo Campagnolo and Simone Tamaro

Technische Universität München

School of Engineering and Design

Wind Energy Institute

Athen, 10. November 2023

# Our model building lab

# TUM Scaled Wind Turbine Family

G0.6 (2017)

G1 (2013)


G2 (2007)

Control hardware (Bachmann M1)
Supervisory control & safety loops
Pitch, torque and yaw control

**All:**

- Real-time individual (G1 & G2) blade pitch, torque & yaw control
- Fully sensorized: shaft and/or blade loads, shaft torque, tower loads, blade pitch and rotor azimuth, nacelle acceleration

# Challenges

Model design
Model control
Manufacturing and assembly of parts
Model sensorization
Calibration and model preparation
Logistics and transportation
Safety
Exchange for research cooperation
Loss of information due to fluctuating University Environment

# Model design


**1:155**


| Parameters | Scaled Model | Samsung S7.0-171 |
|---|---|---|
| Rotor Diameter [m] | 1.1 | 171.2 |
| Hub Height [m] | 0.80 | 110 |
| Rotor Speed [rpm] | 850 | 10.4 |
| Nominal Power [W] | 46 | 7.70e6 |
| Nominal Torque [Nm] | 0.511 | 7.07e6 |
| Reynolds number [-] | 85’000 | 25.0e6 |

Basic design requirement:

- Testing of **wind farm control** strategies

# Model design

Various design tasks can be considered as academic work

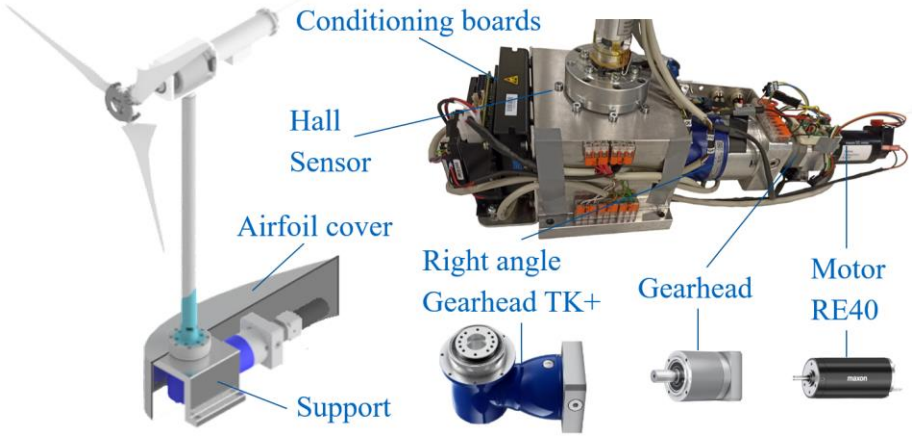
Can be done by students

Research internship

Final thesis

Turbine design is continuously adopted

Turbine design can be divided in several small design tasks



# Model control

Turbine and sensor control

Complex structure
With various applications and
Multiple lines of C+ code

Reduce complexity
- Unify control for whole wind turbine family

Good documentation
Explaining comments

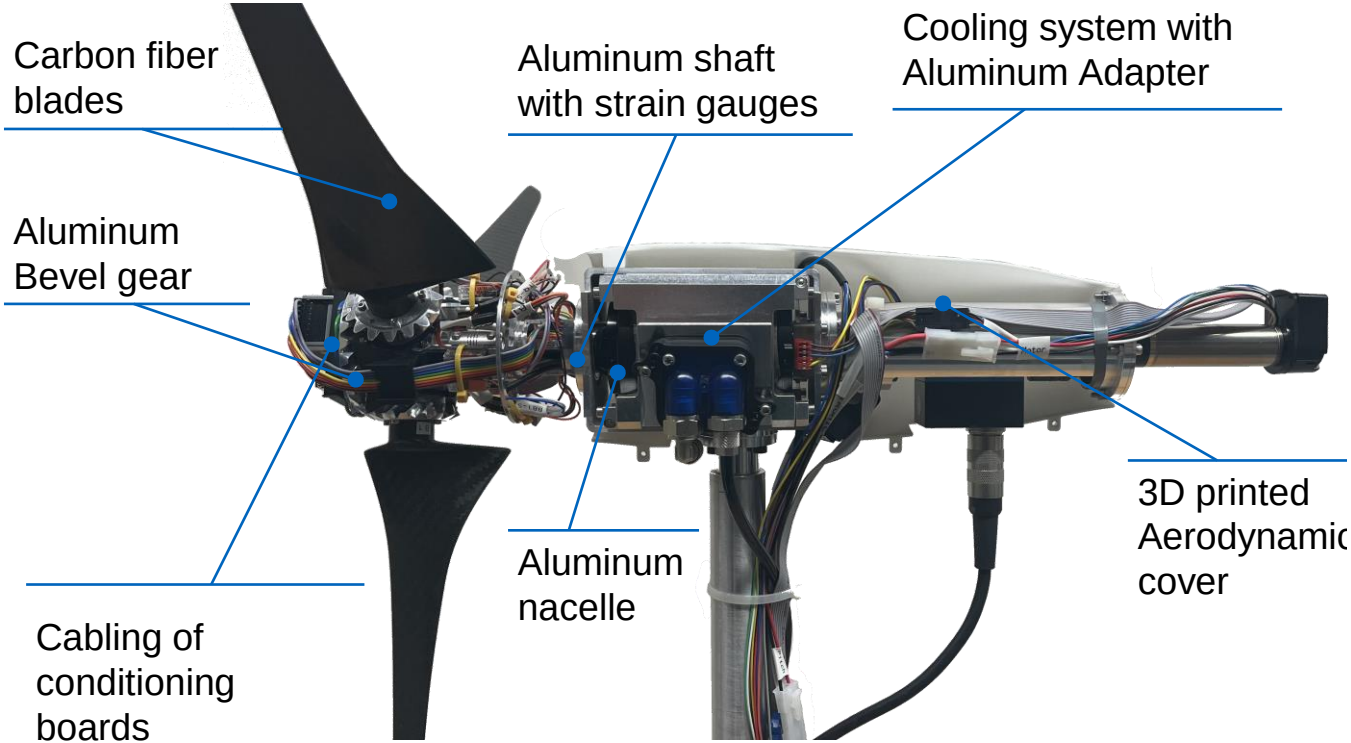
```
425   {"CycleCounter",              SVI_F_INOUT | SVI_F_UINT32, sizeof(UINT32),        (UINT32 *) & CycleCount,
426   {"ModuleVersion",             SVI_F_OUT   | SVI_F_STRING, sizeof(ctrl_Version),  (UINT32 *) ctrl_Version,
427
428   {"CTR/State",                SVI_F_OUT   | SVI_F_UINT32, sizeof(UINT32),        (UINT32*) & CTRL_State,
429   {"CTR/Error",                SVI_F_OUT   | SVI_F_SINT8,  sizeof(SINT),          (UINT32*) & CTRL_Error,
430   {"CTR/Reset",                SVI_F_OUT   | SVI_F_UINT32, sizeof(UINT32),        (UINT32*) & CTRL_Reset,
431   {"CTR/Type_Demand",          SVI_F_INOUT | SVI_F_UINT32, sizeof(UINT32),        (UINT32*) & ControllerParameters.CTRI
432   {"CTR/Type_Active",          SVI_F_OUT   | SVI_F_UINT32, sizeof(UINT32),        (UINT32*) & ControllerParameters.CTRI
433
434   {"Flag/CTRL_Active",          SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.CTRL_Active,
435   {"Flag/Controller",           SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.Controller,
436   {"Flag/StartUp",              SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.StartUp,
437   {"Flag/ShutDown",             SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.ShutDown,
438   {"Flag/Idling",               SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.Idling,
439   {"Flag/GridFail",             SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.GridFail,
440   {"Flag/ShortCircuitFault",    SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.ShortCircuitFault,
441   {"Flag/EmergencyStop",        SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.EmergencyStop,
442   {"Flag/PitchFailure",         SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.PitchFailure,
443   {"Flag/IPC",                  SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.IPC,
444   {"Flag/CounterGridFail",      SVI_F_INOUT | SVI_F_UINT1,  sizeof(UINT32),        (UINT32*) & Flags.CounterGridFail,
445   {"Flag/CounterTypeLaw",       SVI_F_INOUT | SVI_F_UINT8,  sizeof(UINT32),        (UINT32*) & Flags.CounterTypeLaw,
446
447
448   {"Time/Step_s",               SVI_F_OUT   | SVI_F_REAL32, sizeof(REAL32),        (UINT32*) & AppClkTime_s,
449   {"Time/Runtime_s",            SVI_F_OUT   | SVI_F_REAL32, sizeof(REAL32),        (UINT32*) & Runtime_s,
```

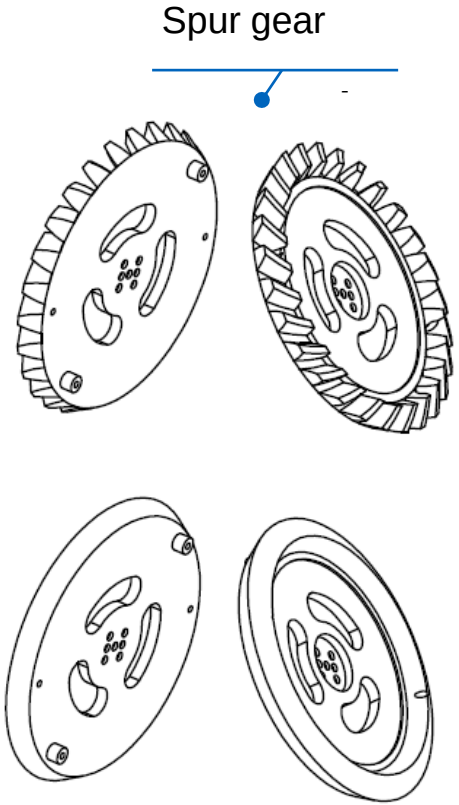
...

```
2229    Offset_rads = Offset_rpm * rpm2rads;
2230    Angle_rad   = (Azimuth_cont_deg + Phase_deg) * deg2rad;
2231
2232    Pitch_Demand_deg[0] = Pitch_Demand_Collective_deg + Amplitude * (REAL32) cos(Frequency_per_rev*Angle_rad );
2233    Pitch_Demand_deg[1] = Pitch_Demand_Collective_deg + Amplitude * (REAL32) cos(Frequency_per_rev*Angle_rad - Offset_rads)
2234    Pitch_Demand_deg[2] = Pitch_Demand_Collective_deg + Amplitude * (REAL32) cos(Frequency_per_rev*Angle_rad + Offset_rads)
2235 }
2236
2237 /*
2238  * This function performs a linear trim of a demand value. (Rate Limiter)
2239  * This means that each cycle, only the Increment value can be added or substracted
2240  * from the previous Output to generate the new output. It is essentially a rate limiter
2241  */
2242 LOCAL VOID Trim_linear( REAL32 *Output, REAL32 Demand, REAL32 Increment )
2243 {
2244     if      ( *Output < Demand )                        // Demand is still higher than previous output
2245     {
2246         *Output += Increment;
2247
2248         if ( *Output >= Demand)    *Output = Demand;    // Limit to Demand
2249     }
2250     else if ( *Output > Demand )                        // Demand is still lower than previous output
2251     {
2252         *Output -= Increment;
2253
```

...

# Manufacturing and assembly of parts

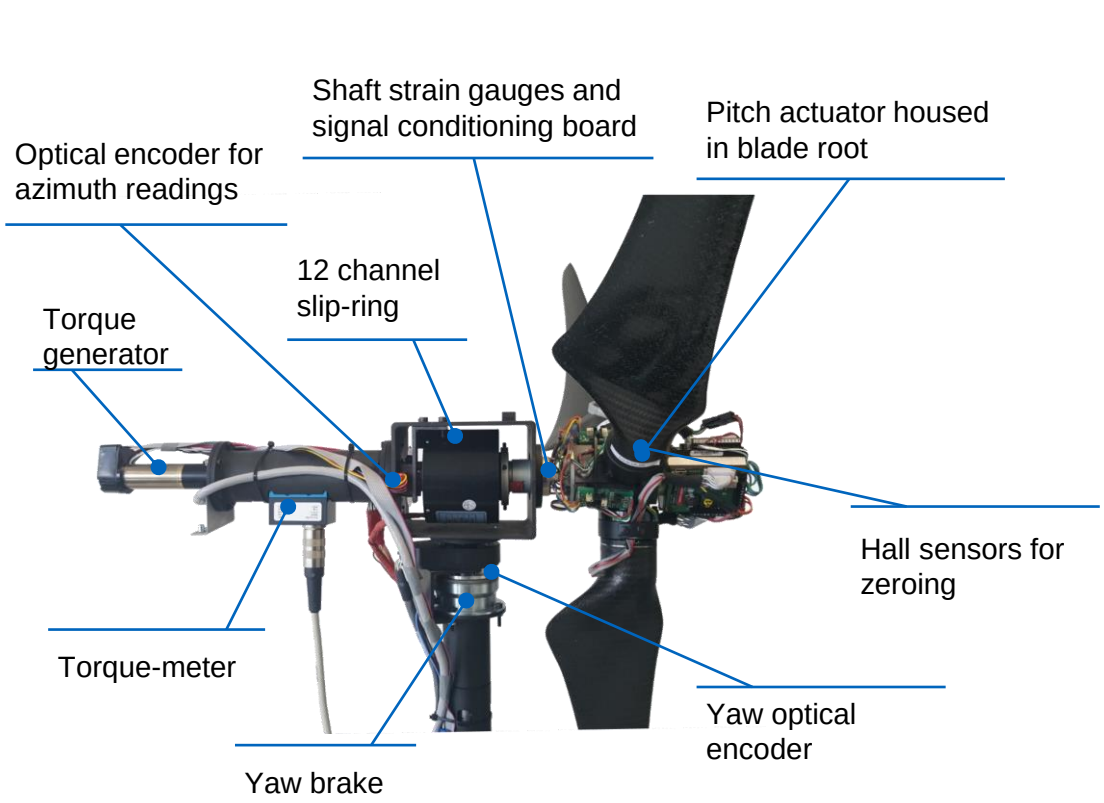


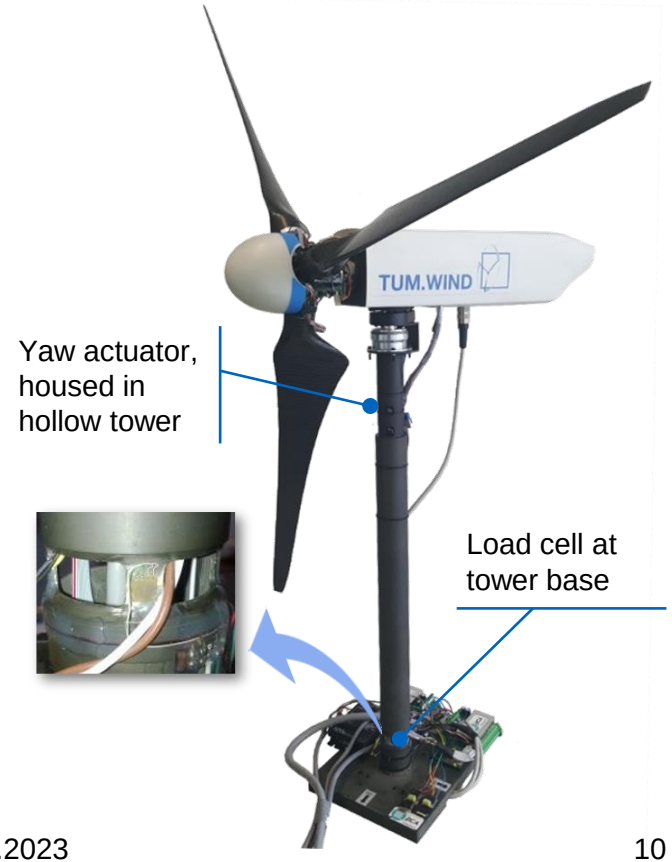



# Manufacturing and assembly of parts

| Internal | vs. | External |
|---|---|---|
| Workshop needed<br>- High investment costs<br>- Technicians needed | | No major infrastructure needed |
| Fast response possible<br>(Can also be problematic) | | Depended on delivery time provided by industrial company |
| Quality can vary | | Guaranteed quality and accuracy |
| Limited to equipment at University | | Possibility to order from various companies and find specialists |
| Mostly availability of non-professional work force, but experts in wind energy | | Technicians are experts in mechanical/electrical work but not wind |
| Organization can be more efficient | | Communication can be demanding |
| Possibly cheaper | | Higher costs |

# Sensorization and actuation



# Sensorization and actuation

Internal vs. External
Communication with turbine software

Strain gauges external for certified quality
Use same technique in whole wind turbine family
Use one manufacturer for the actuators

# Calibration and model preparation

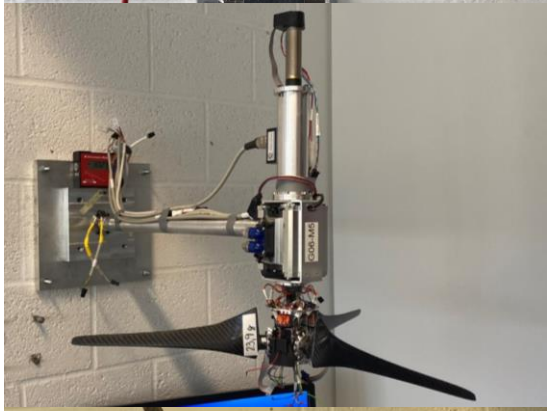
Standardized calibration

- Preferably at institute before shipment
- Conditions can be replicated at different location

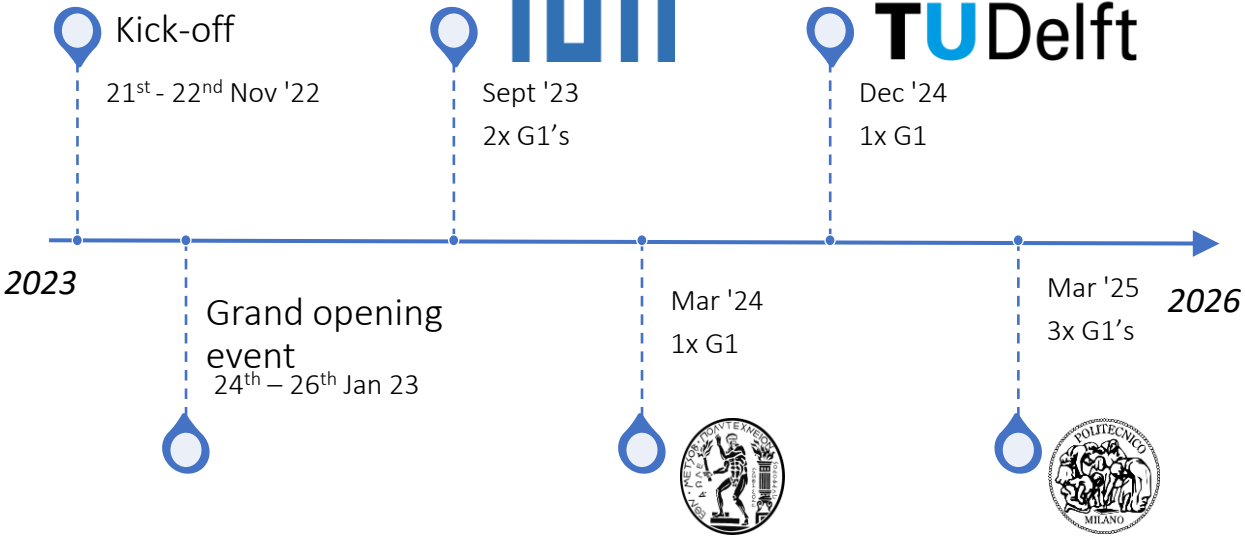
Turbine assembly

- Transport of whole turbine (G06)
- Ensure by detailed labelling for correct connections





# Logistics and Transportation




# Logistics and Transportation

Protect models
Customs (if shipped outside EU)

Prefabricated boxes with foam inlay
Own transport is possible if distance is not too far
problem insurance
Transport companies offers insurance less stressful
Logistic companies help with customs

**Speditionsauftrag** muenchen.preisanfrage@dbschenker.com DB SCHENKER
Revision 003 09/17 ZSF mcd

# Safety

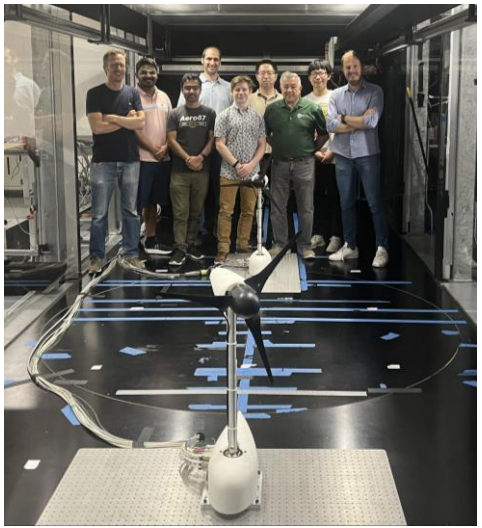
# Exchange for Research cooperation

Turbines are complex machines who need extensive care
Experience is needed to solve problems

We cannot sell the turbines due to University regulations

We can provide the turbines as members of a project
(TWEET-IE)

We are able to rent the turbines
legal department needs to be involved
short term to long term (10 years)



# Loss of information

Researchers with longer contracts
Permanent technicians
Documentation




Wind Energy Institute

# Thank you for your attention!