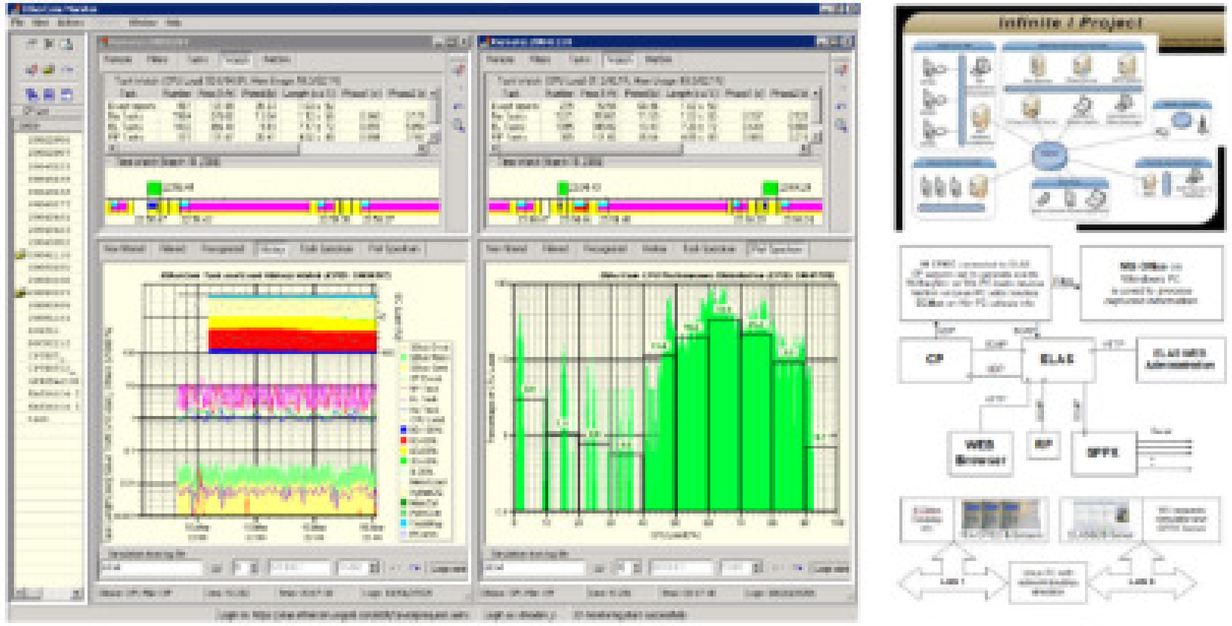
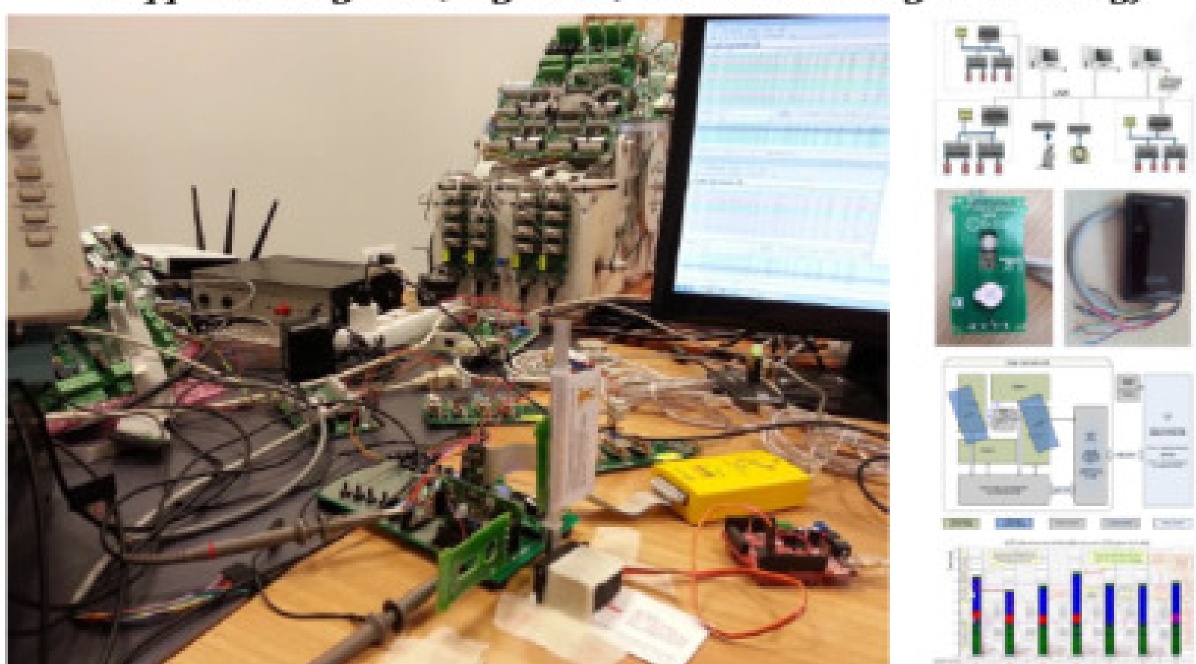
Electronics Line 3000: Internet enabled security system (2003 - 2008)
Introduced: Long term, high load, functional testing methodology

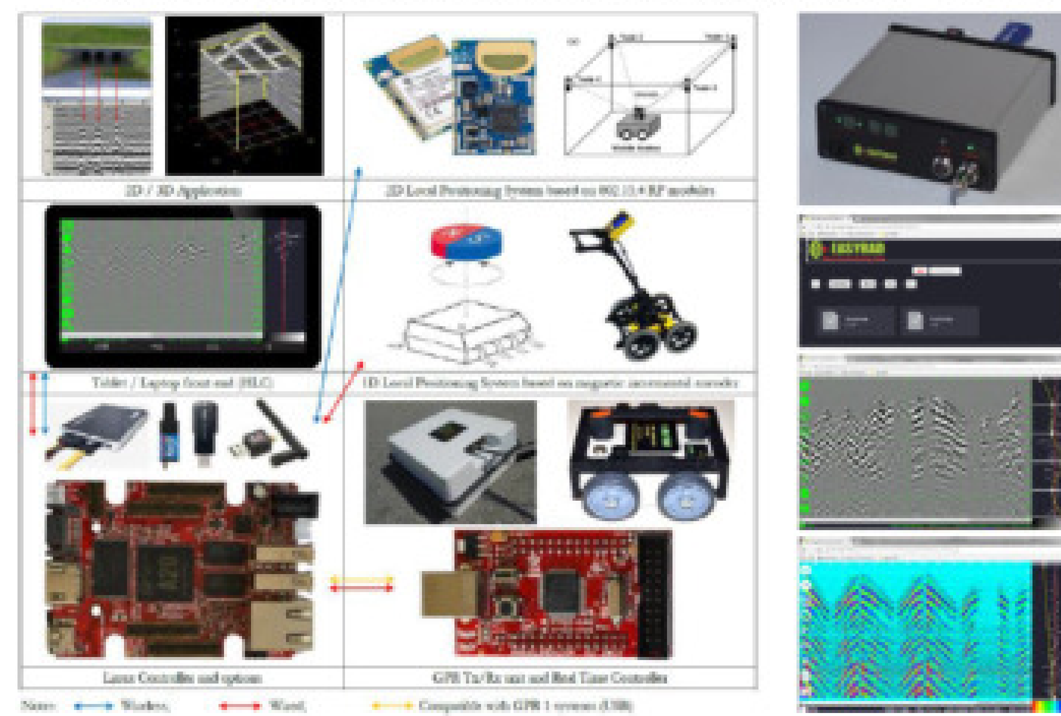


Get connected and distributed (2008 - 2012)
Student's bachelor and master degree thesis

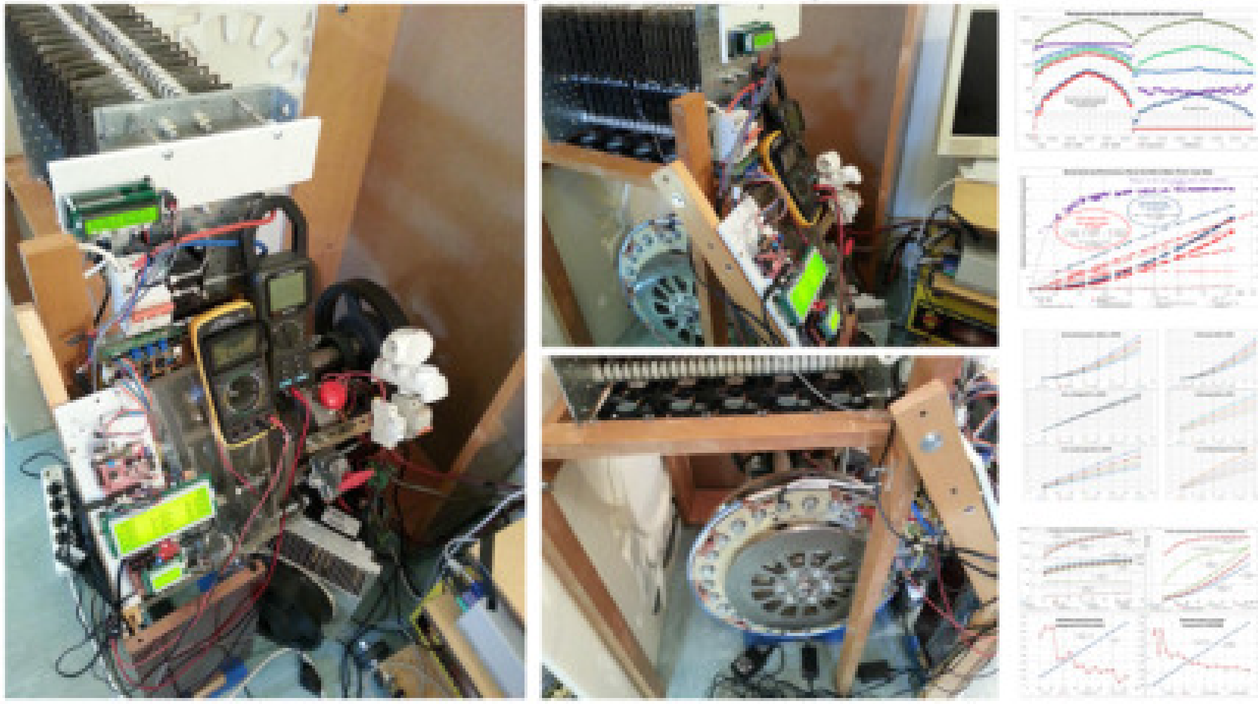
RISCO: RFID reader for Access Control System (2011 - 2014)
Reapplied: Long term, high load, functional testing methodology



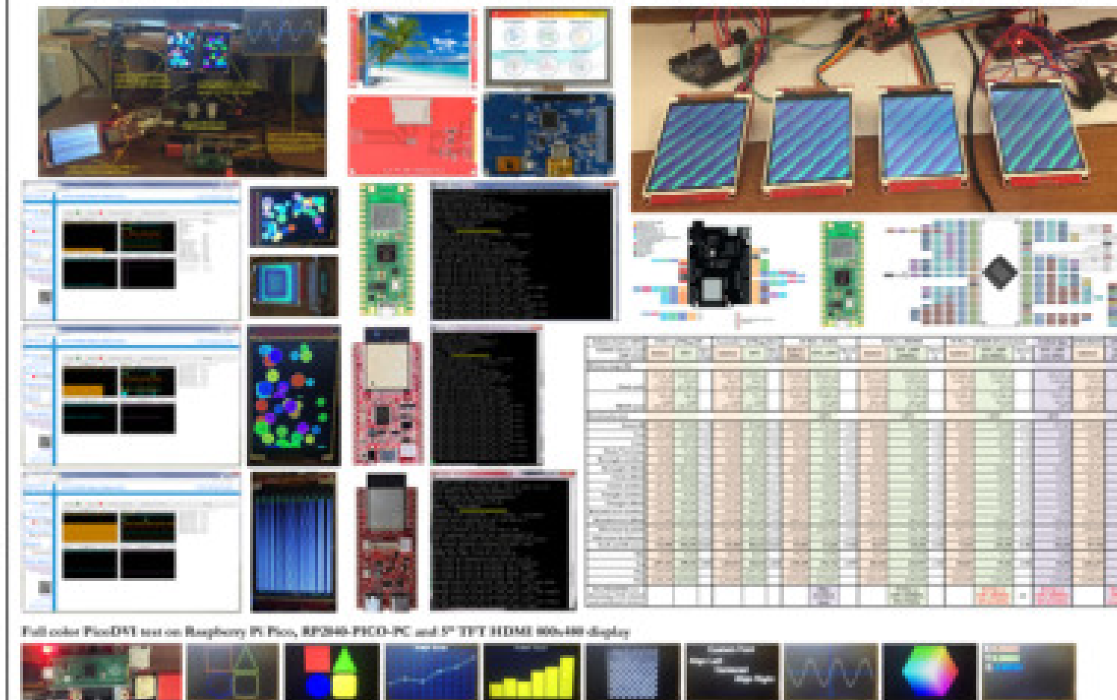
Own project: Easy Ground Penetrating Radar (2015 - 2018)



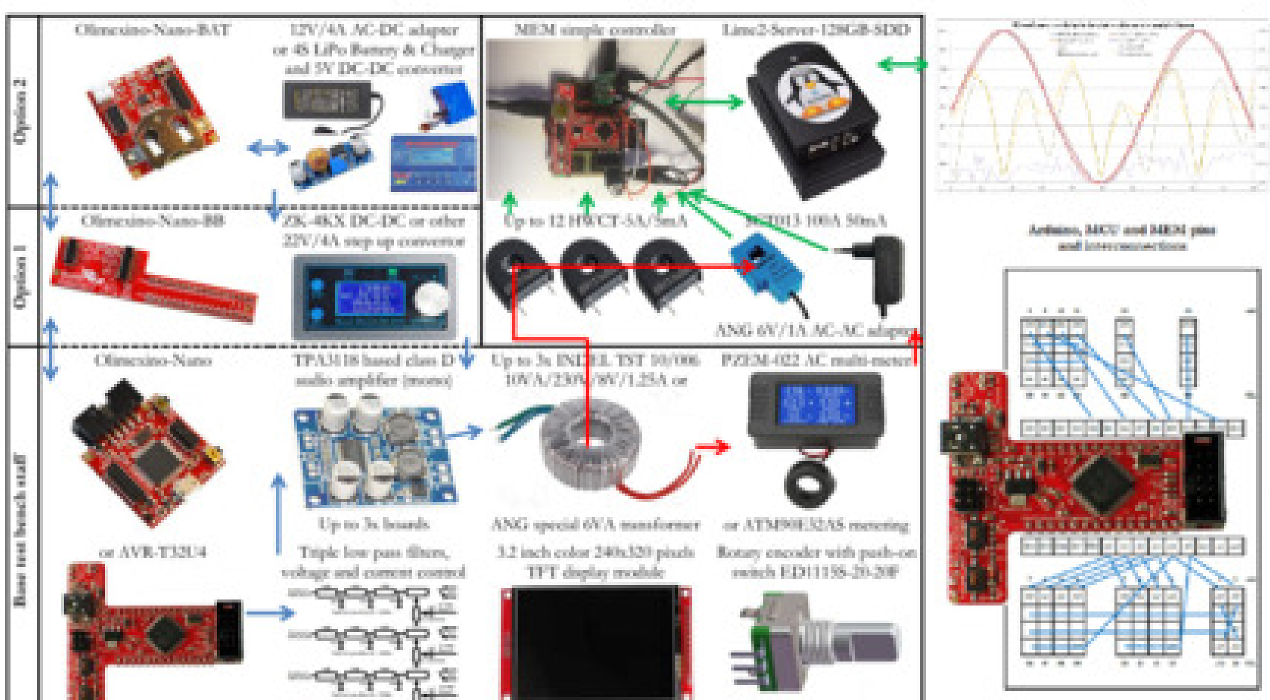
5kW Wind Generator, test bench and measured data (2013)



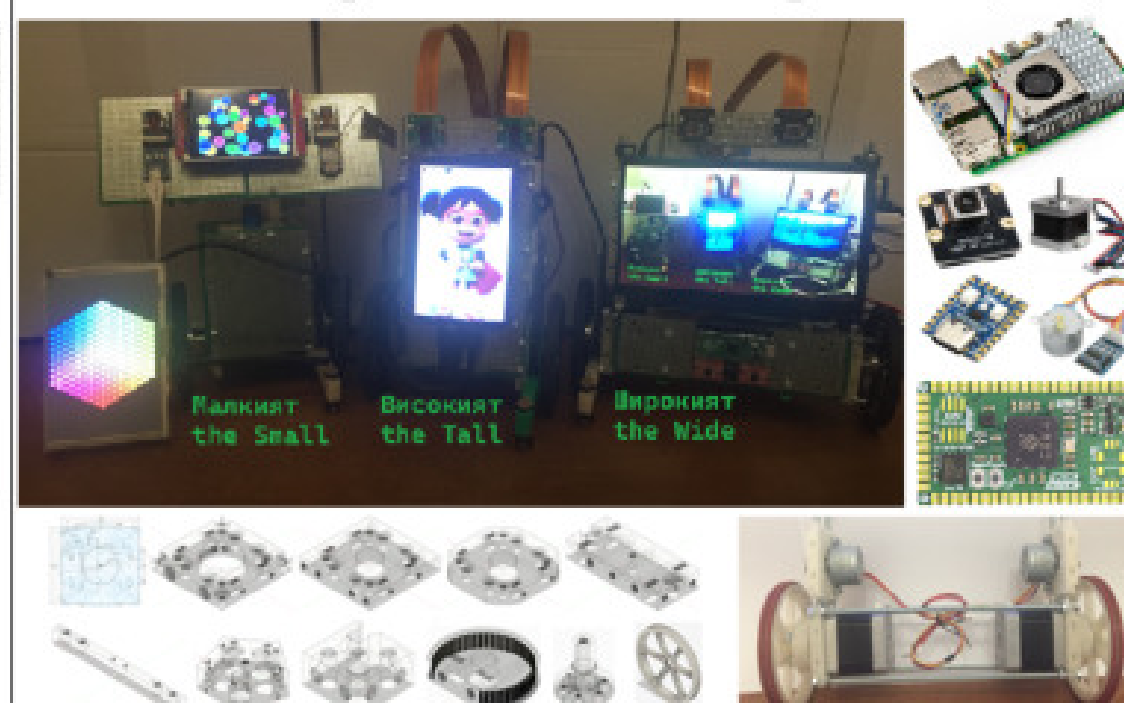
Unified Multicore Low Power IoT Platform (2023)



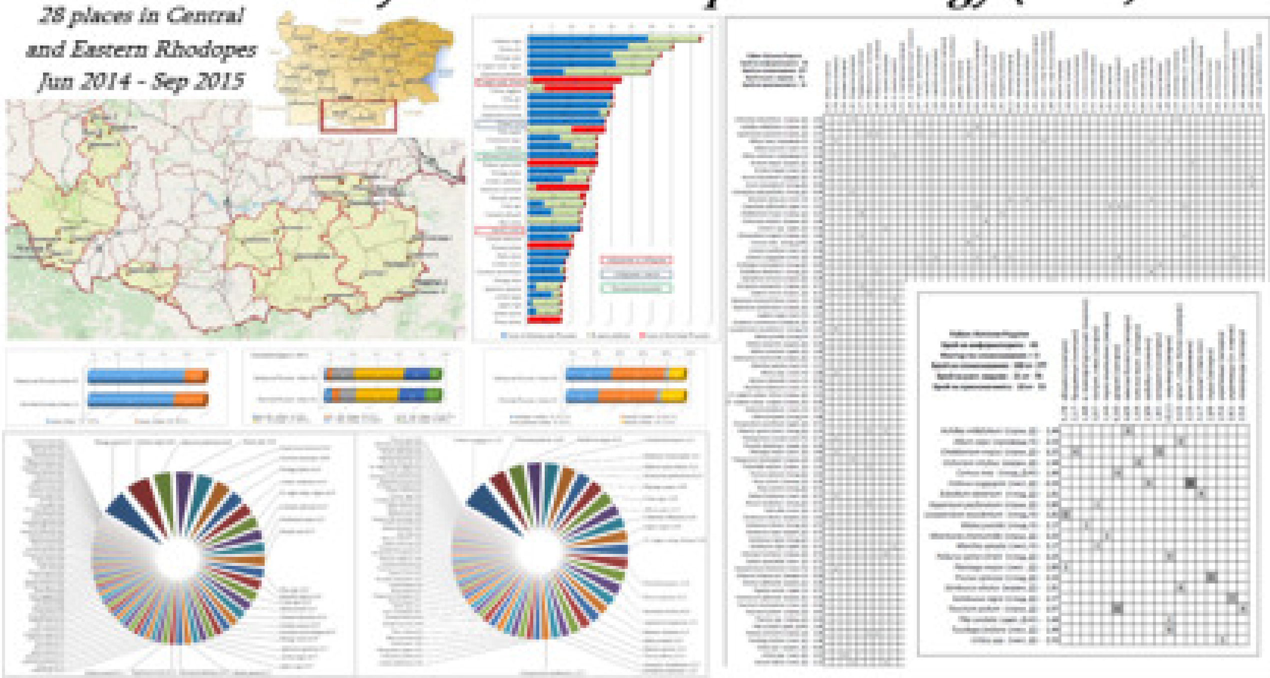
Multichannel Energy Metering system and test bench (2023)



Adroid - the open S.T.E.A.M. robot platform (2024)

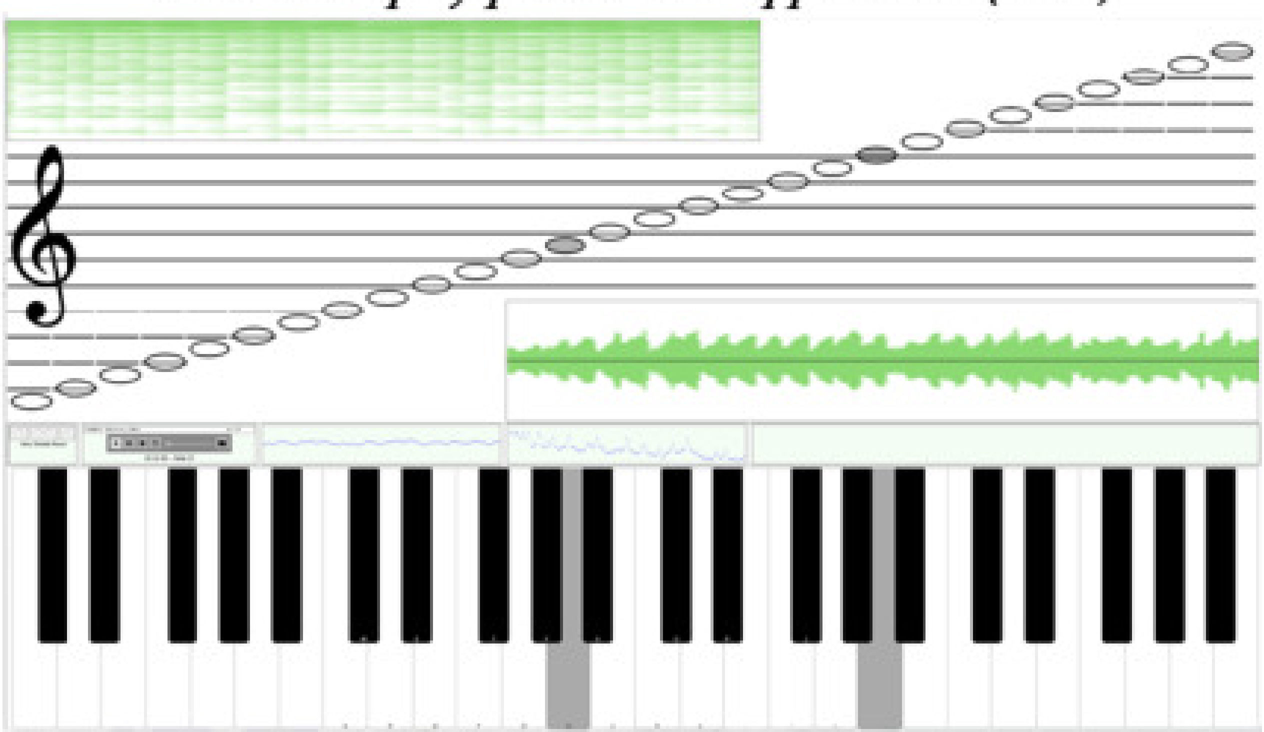


Irena Mincheva's doctoral dissertation in the field of the Ethnobotany and the Ethnopharmacology (2019)

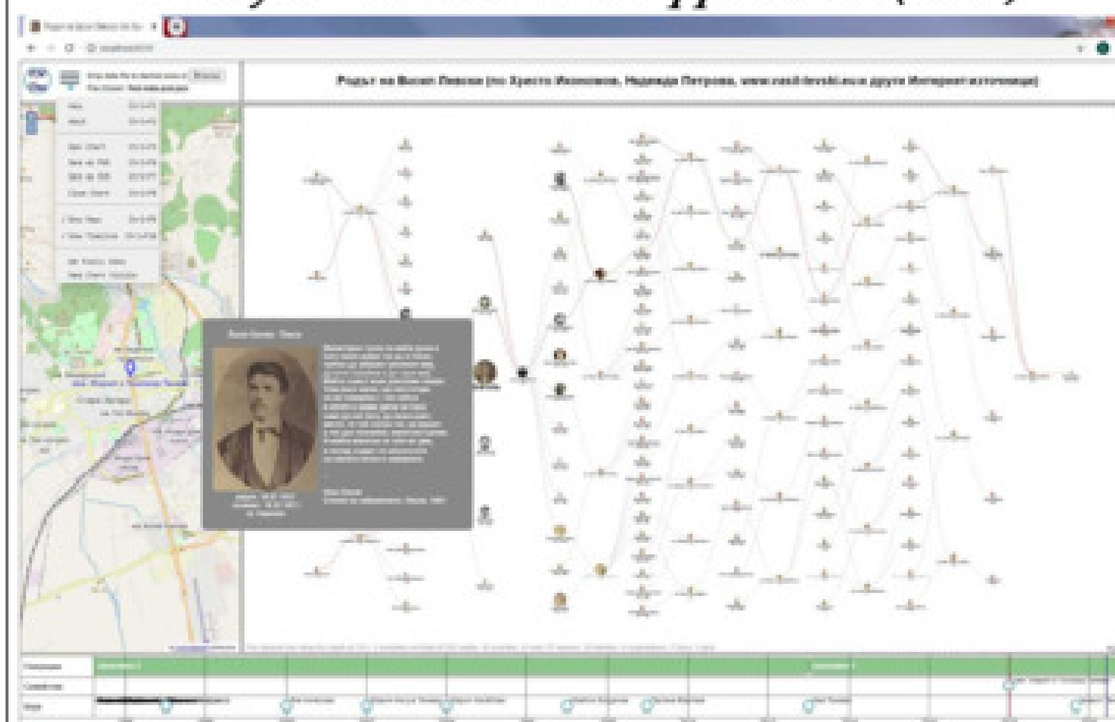


Weather and green energy production monitoring web application (2022)

Listen and play piano - web application (2021)

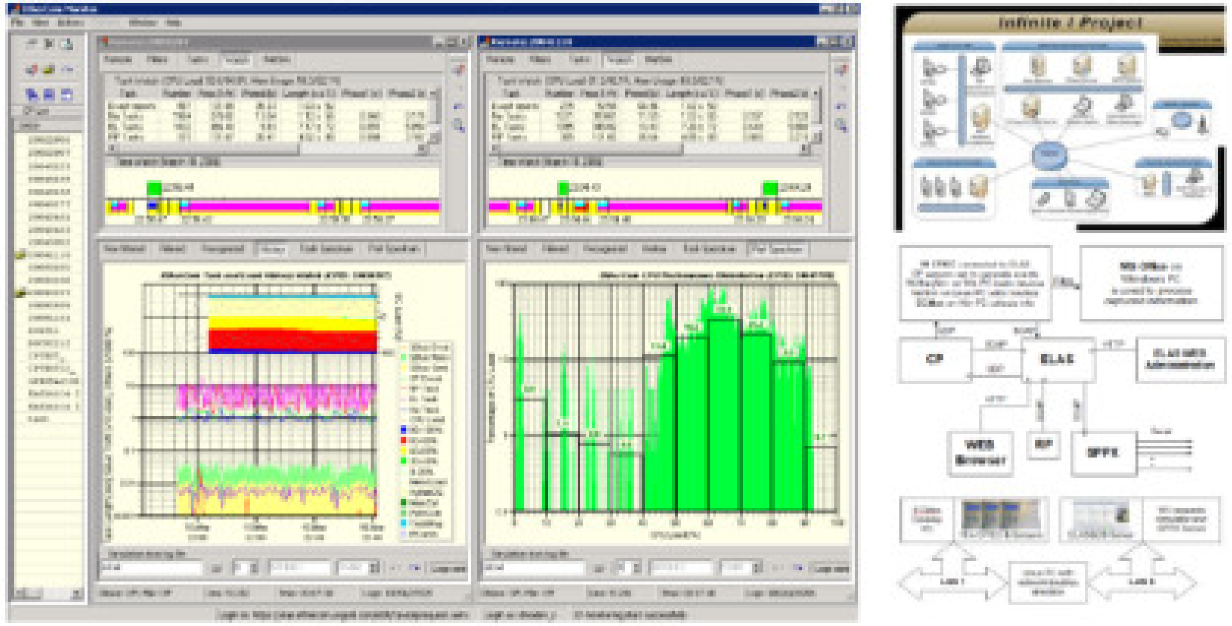


Family Multi Tree - web application (2020)



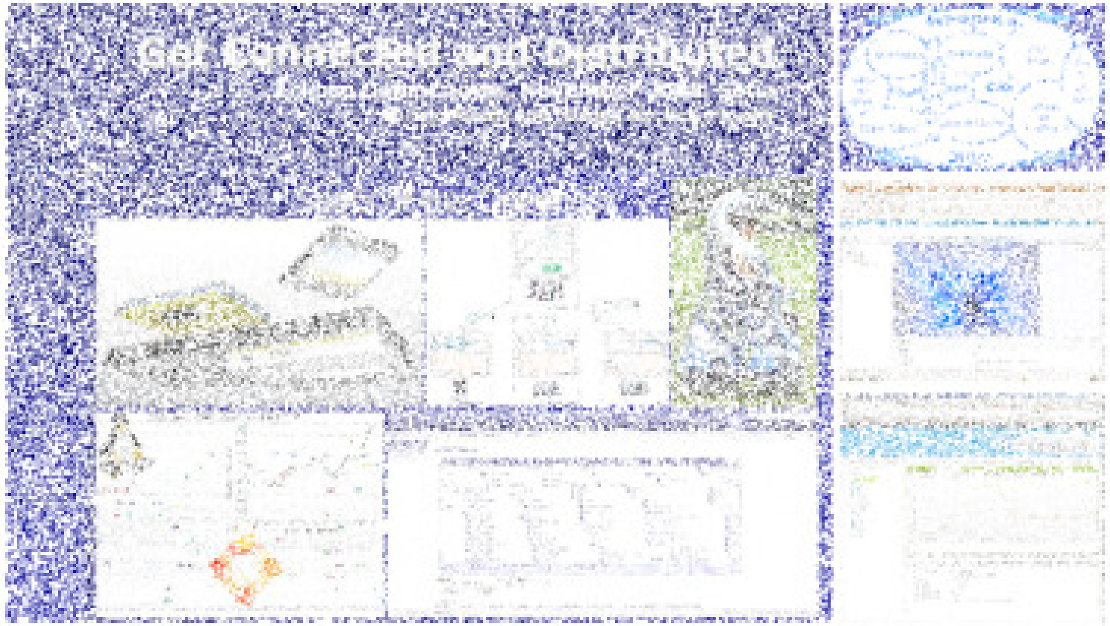
Electronics Line 3000: Internet enabled security system (2003 - 2008)

Introduced: Long term, high load, functional testing methodology



Get connected and distributed (2008 - 2013)

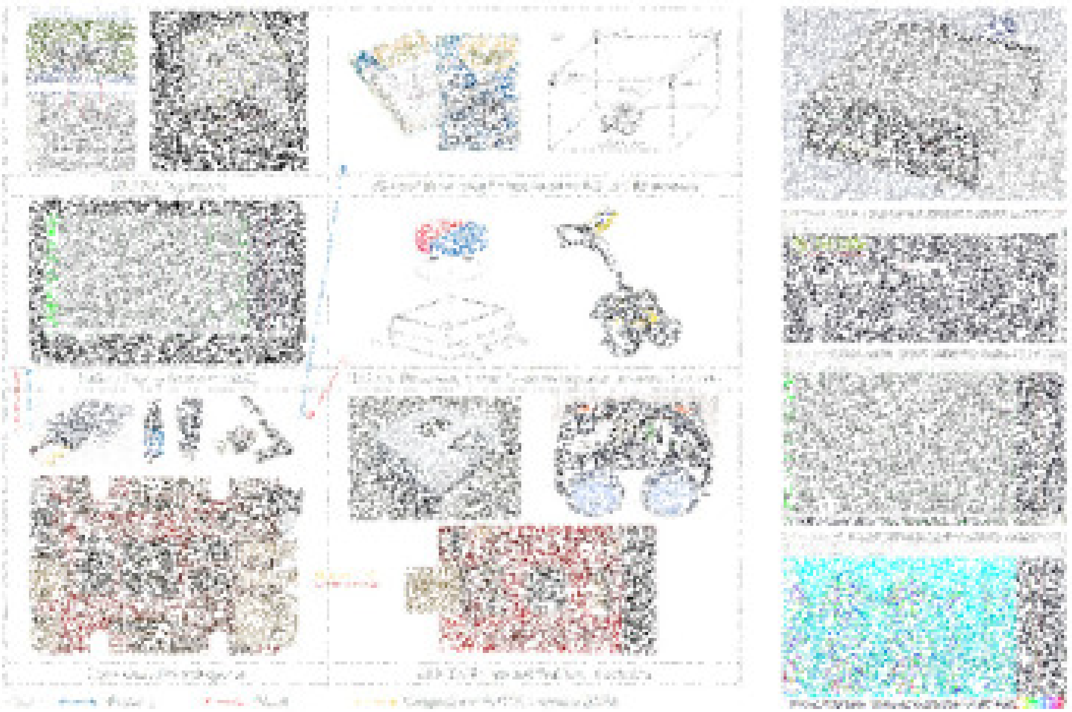
Student's bachelor and master degree thesis



RISCO: RFID reader for Access Control System (2011 - 2014)

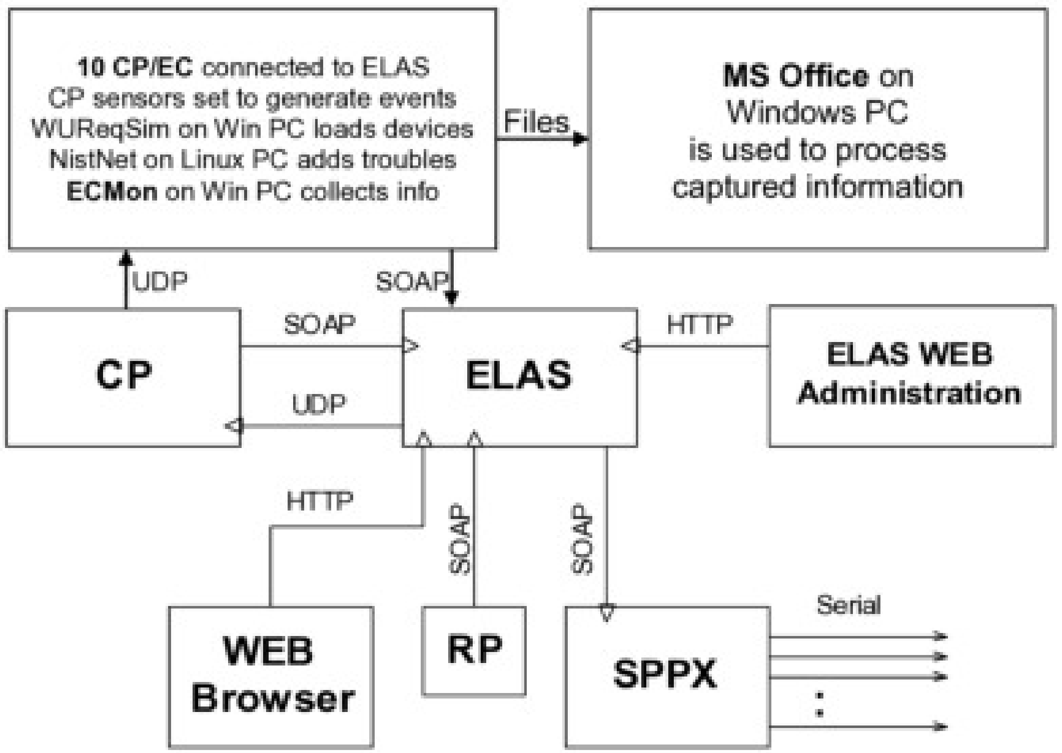
Reapplied: Long term, high load, functional testing methodology

Own project: Easy Ground Penetrating Radar (2015 - 2018)



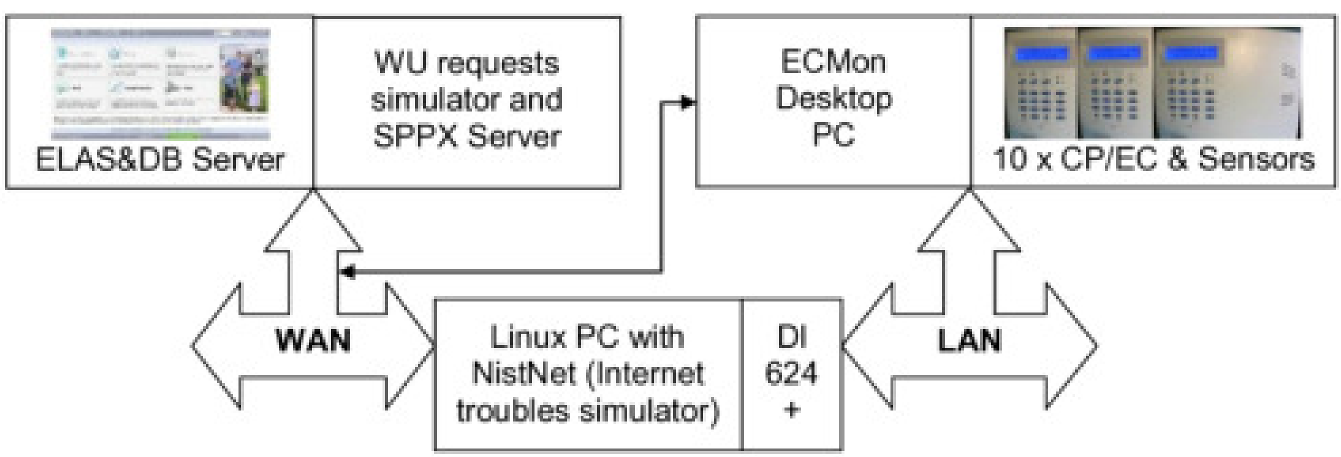
Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

Internet enabled security system (ELAS&SPPX and CP/EC) and long time, extremely high load testing methodology

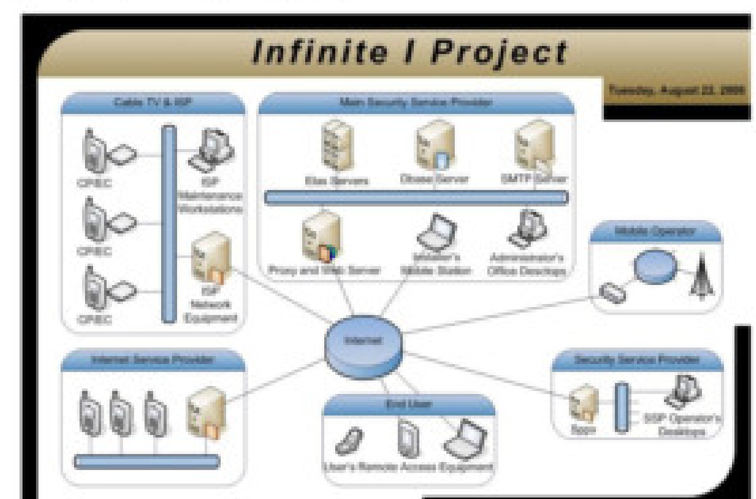


ELAS&SPPX, CP/EC and ECMon tool architecture

ECMon monitoring tool in action (snapshot)



Test suit block diagram



Security system (Infinity-I) architecture

Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

Test conditions (WU requests by testCPProxy):

- All CP/ECs are requested continuously and alternatively to perform SetCPState and GetCPLog with 0.5sec pause between. Periodically (when user access log records exceed 64) CP log is cleared.

Test conditions (RP requests by LogCollector):

- UploadCPLog: every 5min;
- UploadECLog: every 4min;
- UploadData: every 3min.

Test conditions (Event reporting – all CPs):

- All CPs are set with the same parameters (close to defaults);
- All CPs are set not to report via PSTN, no report cycles;
- All CPs have 3 normal PIR sensors;
- All CPs have 2 sensors generation events every minute.

Test conditions (Elas settings):

- Elas is set not to send events to Elpx/Sims or to user via SMS;
- Elas is set to use indirect CP notification;
- Alive cycle is set to 30 sec for both armed and disarmed CP state.

Test conditions (general):

- All ECs are connected to ECMon for monitoring;
- In some cases CP and/or EC have been restarted depending on test case.

Test conditions (infrastructure):

- All CPs are installed behind router (DLink DI-624+ or Edimax);
- All CPs are connected to the router via Ethernet HUB;
- One of the interfaces of the monitoring station is connected to HUB;
- Elas is connected to router's WAN via Linux router;
- NestNet simulator is installed on the Linux router;
- L2TP and PPP services are installed on Linux on VMWare box;
- Separate Elas' CPWS is installed for independent control.

Test conditions (NistNet simulator when on):

- Effects are set to change traffic between Elas and VMWare box;
- Equal effects are set in both directions;
- Packet delay is set to 2ms with 1ms delta and 0 correlation;
- Packet drop ratio is set to 1% with 0 correlation;
- Packet duplication ratio is set to 1% with 0 correlation.

Test conditions (Router settings):

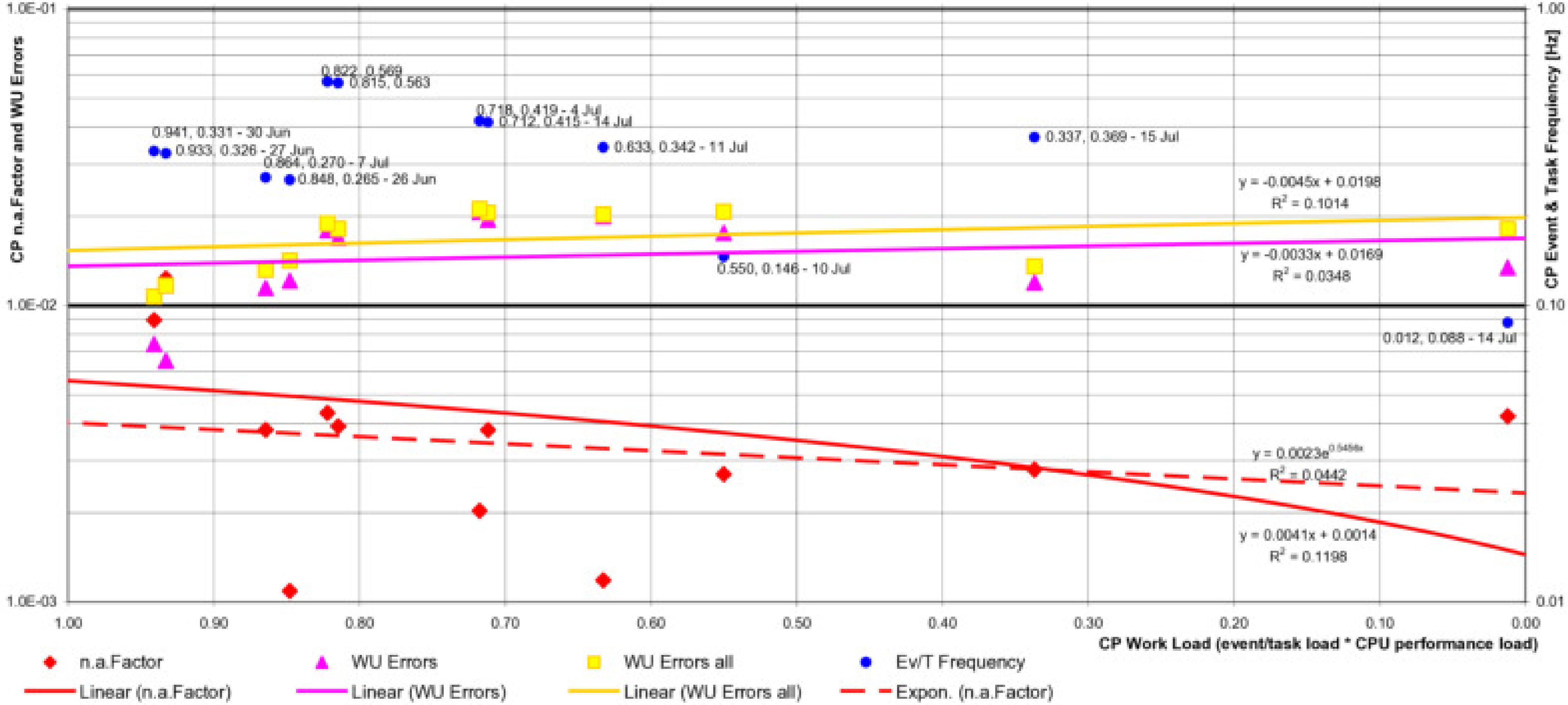
- WAN is set to use static IP address;
- WAN is set to use L2TP on Edimax as alternaive;
- LAN is set to have one and the same IP (used as DGW);
- DHCP is set to use one and the same IP range;
- Rousers have been exchanged or reset depending on test case.

Availability and Reliability Test Results (table show)

Test Date	Work Load	n.a.Factor	WU Errors	WU Errors all	Ev&Task Freq.	TaskLoad	CPULoad	Num of CPs	Test Time [h]	CP * hours	EC IP settings	IIS & cl. certs	L2TP (WAN)	NistNet	WU (sleep)
(start date)	*[%]*	*[%]*	*[%]*	*[%]*	*[Hz]*	*[%]*	*[%]*	*[pc.]*	*[h]*	*[CP*h]*			*[bytes]*	*[ms,%]*	*[sec.]*
15-Jul-08	33.7%	0.28%	1.2%	1.4%	0.369	66.4%	50.8%	8	19.0	152	DHCP	ignore	1492	2/1,1,1	4.5-11
14-Jul-08	1.2%	0.42%	1.3%	1.8%	0.088	10.3%	11.8%	2	65.0	130	DHCP	ignore	1492	2/1,1,1	18-54
14-Jul-08	71.2%	0.38%	1.9%	2.1%	0.415	102.4%	69.5%	8	69.0	552	DHCP	ignore	1492	4/2,2,1	0.5
11-Jul-08	63.3%	0.12%	2.0%	2.0%	0.342	103.0%	61.4%	8	22.0	176	DHCP	ignore	1192	2/1,1,1	0.5
10-Jul-08	55.0%	0.27%	1.8%	2.1%	0.146	107.3%	51.3%	8	22.0	176	DHCP	require	1192	off	0.5
8-Jul-08	81.5%	0.39%	1.7%	1.8%	0.563	100.8%	80.8%	8	24.5	196	DHCP	ignore	Static	2/1,1,1	0.5
7-Jul-08	86.4%	0.38%	1.1%	1.3%	0.270	104.4%	82.8%	8	72.0	576	DHCP	require	1492	2/1,1,1	0.5
4-Jul-08	71.8%	0.20%	2.1%	2.1%	0.419	102.3%	70.1%	8	20.0	160	DHCP	ignore	1492	2/1,1,1	0.5
3-Jul-08	82.2%	0.43%	1.8%	1.9%	0.569	99.8%	82.4%	8	24.0	192	Static	ignore	Static	off	0.5
30-Jun-08	94.1%	0.89%	0.7%	1.1%	0.331	102.8%	91.5%	10	66.0	660	DHCP	require	Static	off	0.5
27-Jun-08	93.3%	1.23%	0.7%	1.2%	0.326	102.2%	91.3%	8	18.0	144	DHCP	require	Static	off	0.5
26-Jun-08	84.8%	0.11%	1.2%	1.4%	0.265	104.0%	81.5%	8	17.5	140	DHCP	require	1392	2/1,1,1	0.5
Min:	*1.22%*	*0.11%*	*0.65%*	*1.07%*	*8.75%*	*10.33%*	*11.81%*	*2*	*17.5*	*130*					
Max:	*94.09%*	*1.23%*	*2.06%*	*2.12%*	*56.92%*	*107.30%*	*91.52%*	*10*	*72.0*	*660*					
Average:		*0.43%*	*1.46%*	*1.68%*			*Summary:*	*92*	*439*	*3,254*					
Std. Dev.:		*0.3%*	*0.5%*	*0.4%*											

Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

Availability and Reliability Test Results (graph show)



Test results and conclusions:

- Web user access errors are 1.46% ± 0.5% and less than 2.06% for all test cases;
- There is no significant influence of CP/EC work conditions on Web user access errors;
- CP Non Availability as function of CP Work Load is like y=0.004x+0.0014 (with R^2=0.1198);
- The influence of CP/EC work conditions on CP Non Availability is small and could be ignored in first approximation;
- CP Non Availability (in time measure in case of CP event and task concurrency) is 0.43% ± 0.3% in first approximation and less than 1.23% for all test cases;
- CP Non Availability is mainly influenced by CP/EC deadlocks causing 3 or 5 min unavailability time window for both RP and WU access while CP reports normally;
- CP/EC deadlock causing 5 min unavailability time window is more rare and could be seen at tests with largest CP Work Load (more than 80%);
- There is no bug (any of the four main issues) manifestation for all CPs and the time they are under the testing (3,254 CP*hours).

Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

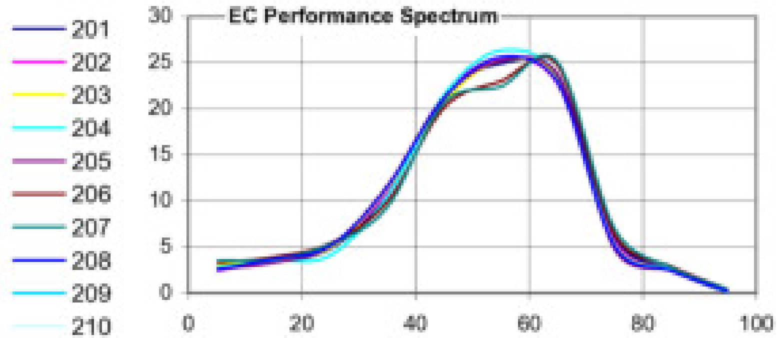
8 CPs * 19h (152 cp*h) at: DHCP, w/o IIS req. certs, with L2TP (1492b), with NistNet (2/1,1,1), WU load (4.5-11s)

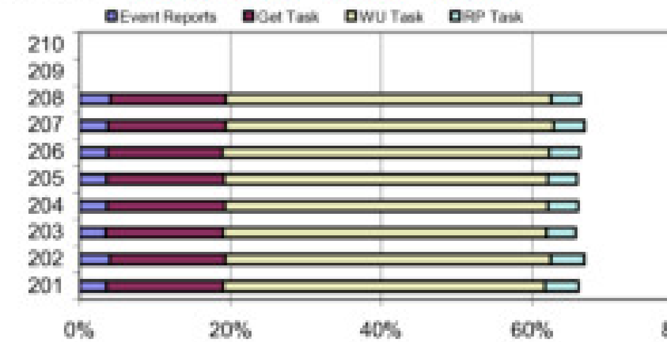
CP Even & Task Frequency [Hz]: 0.369

14-15.07.2008

Task Length [s]: 1.335 0.844 2.974 3.064

3M	5M	1M	n.a.F	WU Errors	
1	0	1	0.31%	1.41%	1.22%
1	0	0	0.26%	1.41%	1.22%
2	0	0	0.53%	1.72%	1.35%
0	0	1	0.04%	1.21%	1.21%
1	0	2	0.35%	1.18%	1.00%
1	0	3	0.39%	1.28%	1.10%
1	0	1	0.31%	1.33%	1.14%
0	0	1	0.04%	1.32%	1.32%





n.a.Factor: 0.28% 1.36% 1.20% ***Work Load: 33.6%*** ***EC CPU Load: 50.8%*** ***CP Even & Task Load: 66.2%***

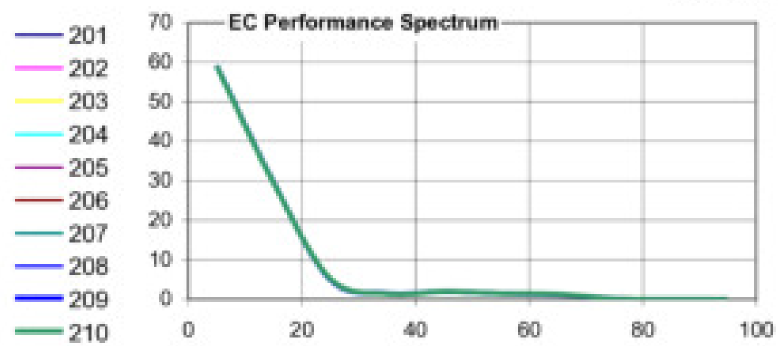
2 CPs * 65h (130 cp*h) at: DHCP, w/o IIS req. certs, with L2TP (1492b), with NistNet (2/1,1,1), WU load (18-54s)

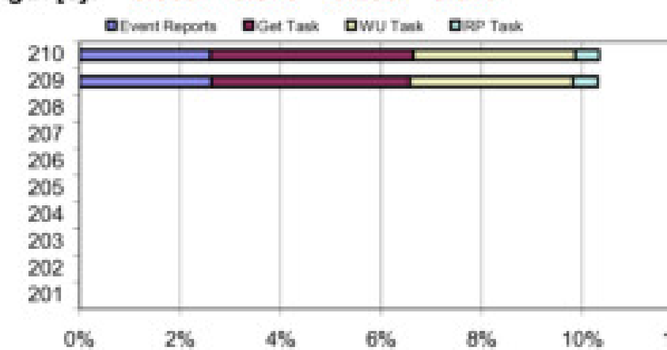
CP Even & Task Frequency [Hz]: 0.088

11-14.07.2008

Task Length [s]: 0.865 0.870 3.205 3.640

3M	5M	1M	n.a.F	WU Errors	
7	0	0	0.54%	1.95%	1.36%
4	0	0	0.31%	1.67%	1.33%





n.a.Factor: 0.42% 1.81% 1.34% ***Work Load: 1.2%*** ***EC CPU Load: 11.8%*** ***CP Even & Task Load: 10.3%***

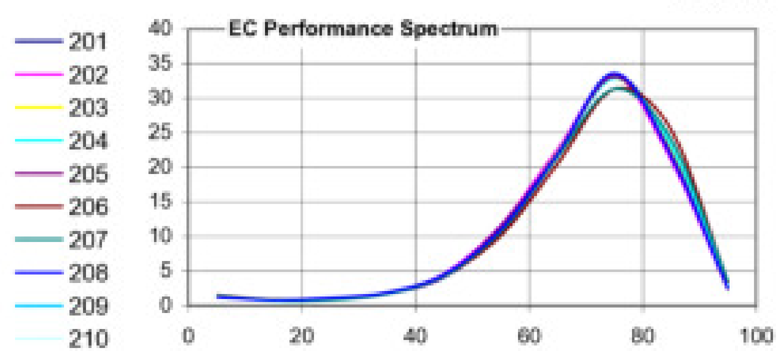
8 CPs * 69h (552 cp*h) at: DHCP, w/o IIS req. certs, with L2TP (1492b), with NistNet (2/1,1,1)

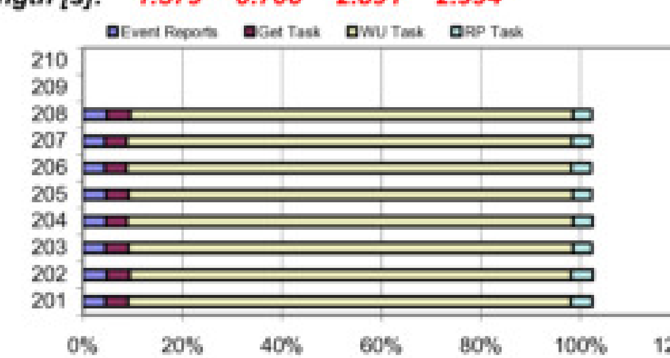
CP Even & Task Frequency [Hz]: 0.415

11-14.07.2008

Task Length [s]: 1.579 0.700 2.891 2.994

3M	5M	1M	n.a.F	WU Errors	
5	0	7	0.45%	2.13%	1.98%
5	0	4	0.41%	2.16%	2.01%
4	0	9	0.40%	2.12%	2.00%
0	0	11	0.13%	1.96%	1.96%
5	0	13	0.52%	2.17%	2.02%
2	0	9	0.25%	1.84%	1.78%
1	0	12	0.22%	1.77%	1.74%
7	0	13	0.66%	2.29%	2.08%





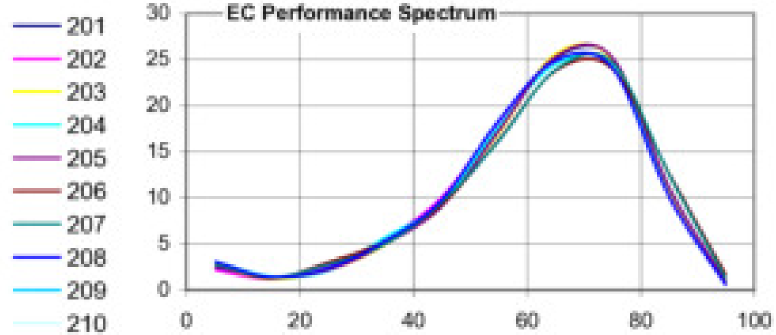
n.a.Factor: 0.38% 2.05% 1.95% ***Work Load: 71.2%*** ***EC CPU Load: 69.5%*** ***CP Even & Task Load: 102.3%***

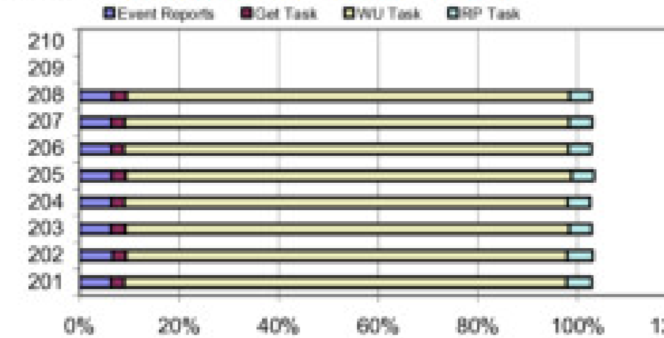
Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

8 CPs * 22h (176 cp*h) at: DHCP, w/o IIS req. certs, with L2TP (1192b), with NistNet (4/2,2,1)
10-11.07.2008

CP Even & Task Frequency [Hz]: 0.342
Task Length [s]: 2.230 0.774 3.368 3.548

3M	5M	1M	n.a.F	WU Errors	
1	0	0	0.23%	2.26%	2.15%
0	0	0	0.00%	2.20%	2.20%
0	0	3	0.11%	1.95%	1.95%
0	0	3	0.11%	1.99%	1.99%
0	0	5	0.19%	1.93%	1.93%
1	0	0	0.23%	2.03%	1.93%
0	0	0	0.00%	1.94%	1.94%
0	0	2	0.08%	1.91%	1.91%



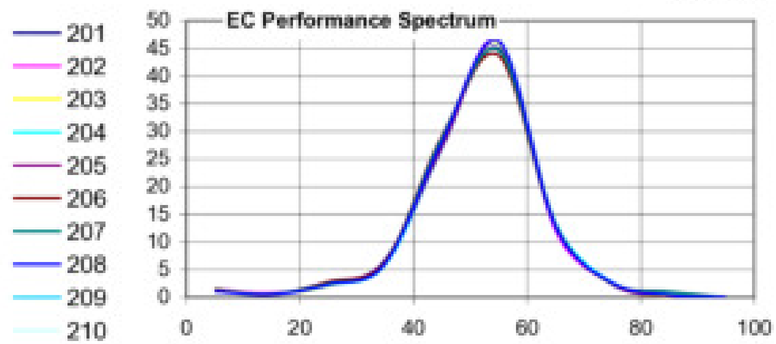


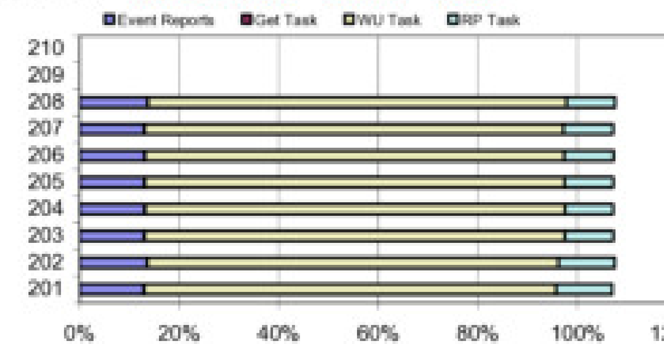
n.a.Factor: 0.12% 2.03% 2.00% **Work Load: 63.3%** **EC CPU Load: 61.4%** **CP Even & Task Load: 103.0%**

8 CPs * 22h (176 cp*h) at: DHCP, with IIS req. certs, with L2TP (1192b), with NistNet (2/1,1,1)
09-10.07.2008

CP Even & Task Frequency [Hz]: 0.146
Task Length [s]: 4.524 2.389 8.075 7.940

3M	5M	1M	n.a.F	WU Errors	
1	0	1	0.27%	2.32%	2.04%
2	0	0	0.45%	2.51%	1.94%
1	0	0	0.23%	2.14%	1.87%
1	0	2	0.30%	1.98%	1.70%
2	0	0	0.45%	2.07%	1.52%
0	0	0	0.00%	1.59%	1.59%
0	0	0	0.00%	1.63%	1.63%
2	0	0	0.45%	2.33%	1.78%



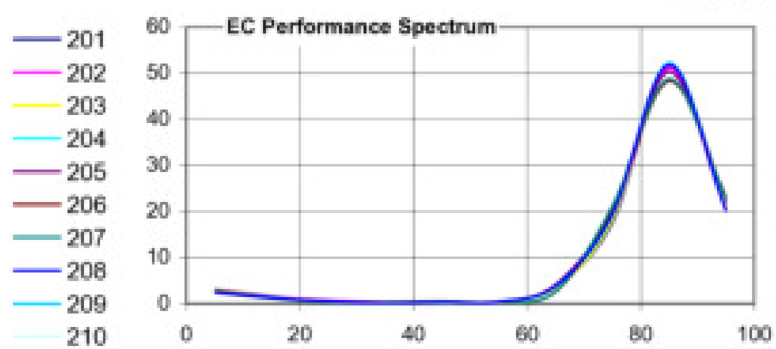


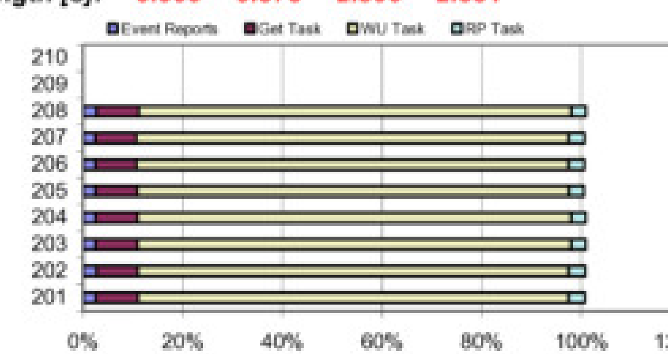
n.a.Factor: 0.27% 2.07% 1.76% **Work Load: 55.0%** **EC CPU Load: 51.3%** **CP Even & Task Load: 107.3%**

8 CPs * 24.5h (196 cp*h) at: DHCP, w/o IIS req. certs, w/o L2TP, w/o NistNet
07-08.07.2008

CP Even & Task Frequency [Hz]: 0.563
Task Length [s]: 0.909 0.579 2.300 2.331

3M	5M	1M	n.a.F	WU Errors	
1	0	1	0.24%	1.80%	1.73%
3	0	3	0.71%	2.07%	1.86%
2	0	1	0.44%	1.88%	1.74%
1	0	0	0.20%	1.73%	1.66%
1	0	1	0.24%	1.77%	1.70%
3	0	1	0.65%	1.74%	1.53%
1	0	0	0.20%	1.51%	1.44%
2	0	1	0.44%	1.99%	1.86%





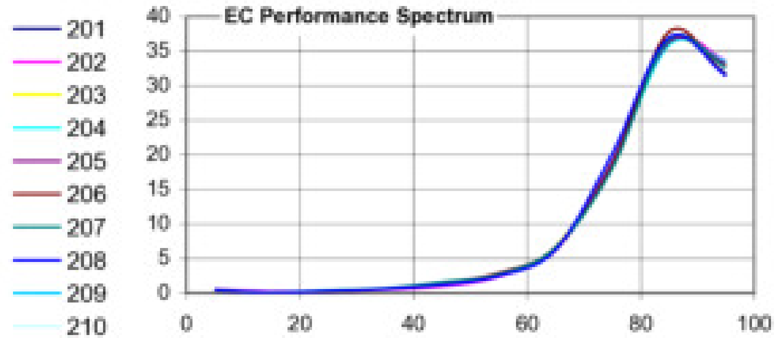
n.a.Factor: 0.39% 1.81% 1.69% **Work Load: 81.4%** **EC CPU Load: 80.8%** **CP Even & Task Load: 100.8%**

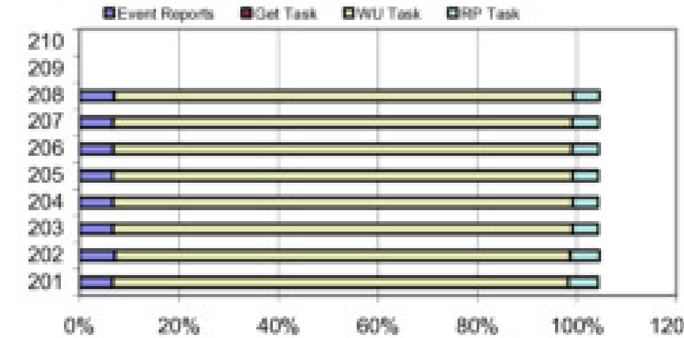
Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

8 CPs * 72h (576 cp*h) at: DHCP, with IIS req. certs, with L2TP (1492b), with NistNet (2/1,1,1) — ***CP Even & Task Frequency [Hz]:*** 0.270

04-07.07.2008 — ***Task Length [s]:*** 2.388 1.150 4.036 4.101

3M	5M	1M	n.a.F	WU Errors	
3	0	3	0.24%	1.13%	1.01%
1	0	6	0.14%	1.15%	1.11%
6	0	2	0.44%	1.40%	1.17%
8	1	7	0.75%	1.51%	1.21%
6	0	4	0.46%	1.38%	1.15%
2	0	2	0.16%	1.22%	1.14%
5	0	10	0.46%	1.34%	1.14%
5	0	3	0.38%	1.42%	1.22%



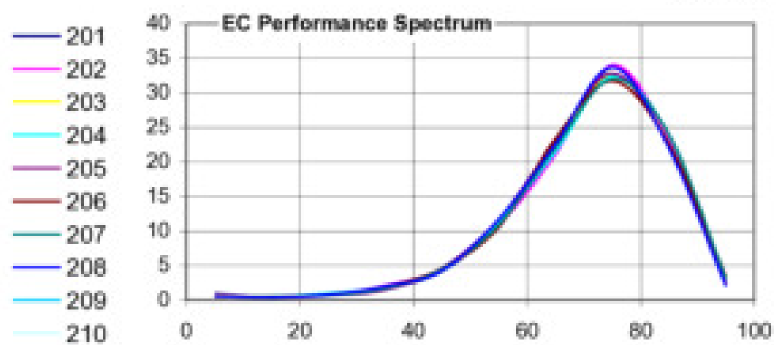


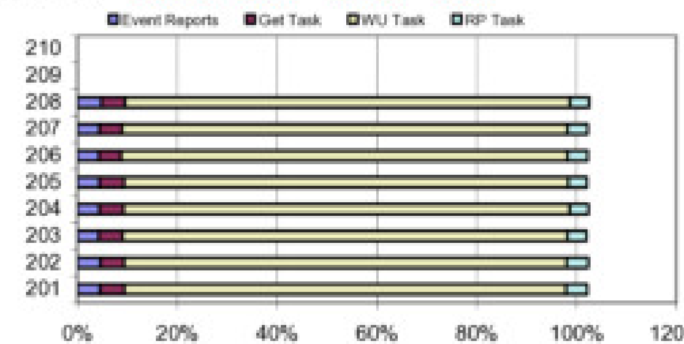
n.a.Factor: 0.38% 1.32% 1.15% ***Work Load:*** 86.4% ***EC CPU Load:*** 82.8% ***CP Even & Task Load:*** 104.4%

8 CPs * 20h (160 cp*h) at: DHCP, w/o IIS req. certs, with L2TP (1492b), with NistNet (2/1,1,1) — ***CP Even & Task Frequency [Hz]:*** 0.419

03-04.07.2008 — ***Task Length [s]:*** 1.585 0.679 2.891 2.981

3M	5M	1M	n.a.F	WU Errors	
1	0	1	0.29%	2.25%	2.15%
0	0	3	0.13%	1.97%	1.97%
1	0	2	0.33%	2.20%	2.10%
1	0	3	0.38%	2.20%	2.10%
0	0	2	0.08%	2.02%	2.02%
1	0	2	0.33%	2.07%	1.96%
0	0	1	0.04%	1.89%	1.89%
0	0	1	0.04%	2.33%	2.33%



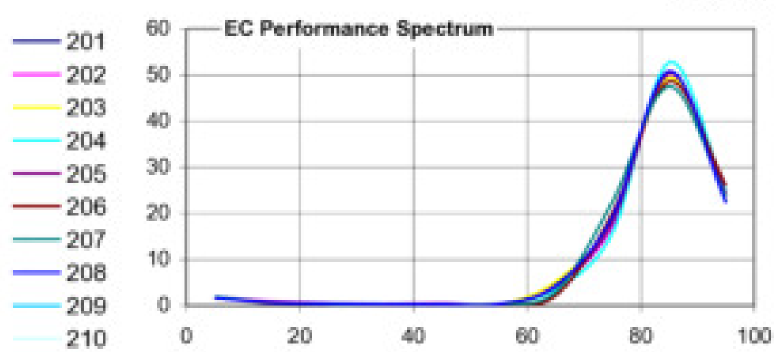


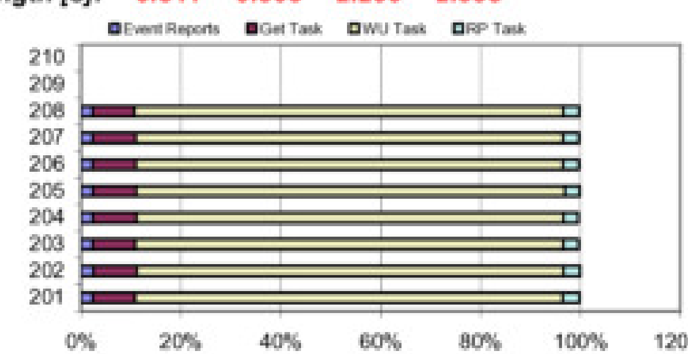
n.a.Factor: 0.20% 2.12% 2.06% ***Work Load:*** 71.7% ***EC CPU Load:*** 70.1% ***CP Even & Task Load:*** 102.3%

8 CPs * 24h (192 cp*h) at: static IPs, w/o any extra load — ***CP Even & Task Frequency [Hz]:*** 0.569

02-03.07.2008 — ***Task Length [s]:*** 0.841 0.565 2.280 2.335

3M	5M	1M	n.a.F	WU Errors	
0	0	2	0.07%	1.75%	1.75%
3	0	2	0.69%	2.23%	2.02%
1	0	2	0.28%	2.04%	1.96%
1	1	2	0.63%	1.98%	1.91%
2	0	2	0.49%	2.13%	1.99%
1	0	2	0.28%	1.46%	1.39%
1	0	7	0.45%	1.50%	1.43%
2	0	5	0.59%	2.04%	1.90%





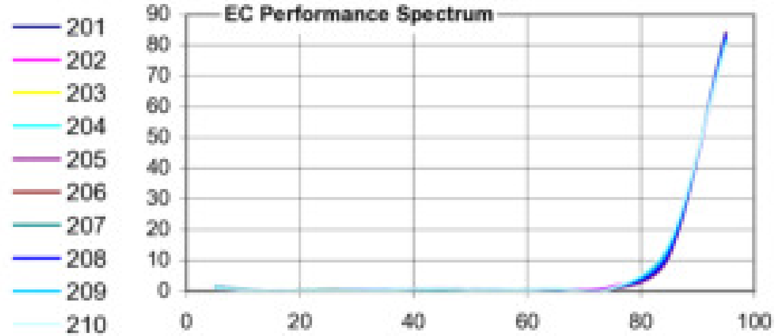
n.a.Factor: 0.43% 1.89% 1.79% ***Work Load:*** 82.2% ***EC CPU Load:*** 82.4% ***CP Even & Task Load:*** 99.7%

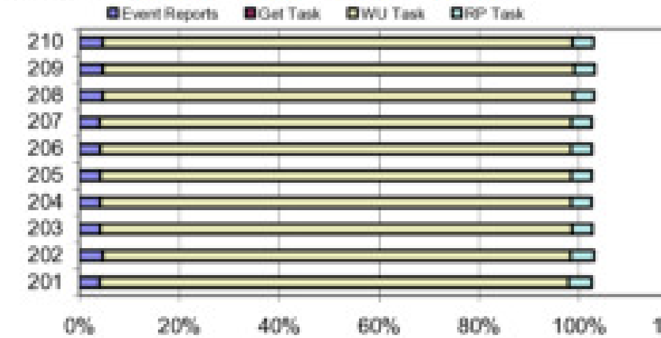
Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

*10 CPs * 66h (660 cp*h) at: DHCP, with IIS req. certs, w/o L2TP and NistNet*
27-30.06.2008

CP Even & Task Frequency [Hz]: 0.331
Task Length [s]: 1.613 1.017 3.236 3.234

3M	5M	1M	n.a.F	WU Errors	
9	0	4	0.73%	1.00%	0.70%
10	2	8	1.11%	1.13%	0.79%
5	0	3	0.42%	0.88%	0.71%
4	1	5	0.49%	0.78%	0.64%
15	1	4	1.31%	1.23%	0.73%
11	1	3	1.00%	1.08%	0.70%
9	0	5	0.74%	1.00%	0.70%
12	1	4	1.09%	1.14%	0.74%
14	1	3	1.22%	1.35%	0.88%
10	0	4	0.81%	1.14%	0.80%
		n.a.Factor:	0.89%	1.07%	0.74%



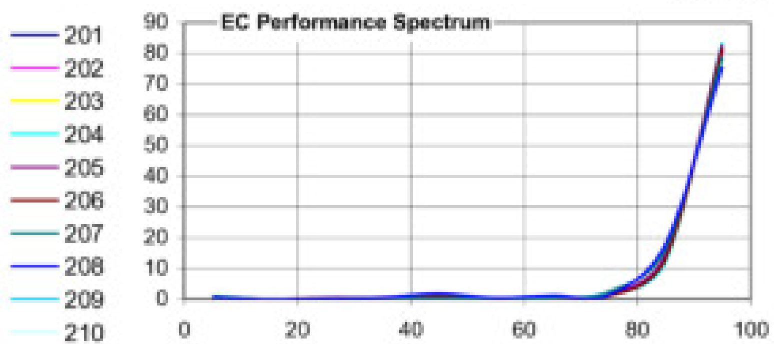


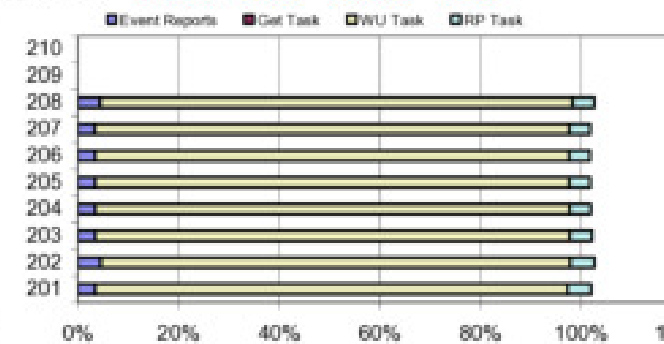
Work Load: 94.1% *EC CPU Load:* 91.5% *CP Even & Task Load:* 102.8%

*8 CPs * 18h (144 cp*h) at: DHCP, with IIS req. certs, w/o L2TP and NistNet*
26-27.06.2008

CP Even & Task Frequency [Hz]: 0.326
Task Length [s]: 1.483 0.983 3.273 3.240

3M	5M	1M	n.a.F	WU Errors	
5	0	0	1.39%	1.24%	0.64%
4	0	0	1.11%	1.21%	0.73%
4	1	0	1.57%	1.09%	0.61%
2	0	3	0.69%	0.87%	0.63%
0	0	0	0.00%	0.62%	0.53%
6	0	0	1.67%	1.36%	0.64%
3	0	0	0.83%	0.97%	0.60%
9	0	2	2.59%	1.97%	0.83%
		n.a.Factor:	1.23%	1.17%	0.65%



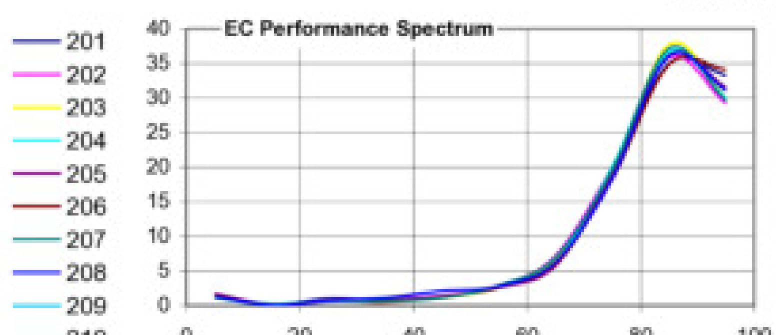


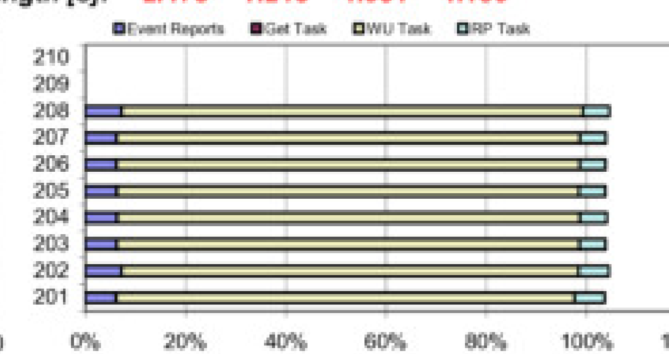
Work Load: 93.3% *EC CPU Load:* 91.3% *CP Even & Task Load:* 102.1%

*8 CPs * 17.5h (140 cp*h) at: DHCP, with IIS req. certs, with L2TP (1392b), with NistNet (2/1,1,1)*
25-26.06.2008

CP Even & Task Frequency [Hz]: 0.265
Task Length [s]: 2.478 1.248 4.081 4.130

3M	5M	1M	n.a.F	WU Errors	
0	0	0	0.00%	1.17%	1.17%
3	0	1	0.24%	1.67%	1.17%
0	0	1	0.01%	1.16%	1.16%
0	0	1	0.01%	1.26%	1.26%
1	0	2	0.10%	1.39%	1.23%
1	0	2	0.10%	1.40%	1.24%
0	0	1	0.01%	1.14%	1.14%
5	0	1	0.39%	2.13%	1.32%
		n.a.Factor:	0.11%	1.42%	1.21%





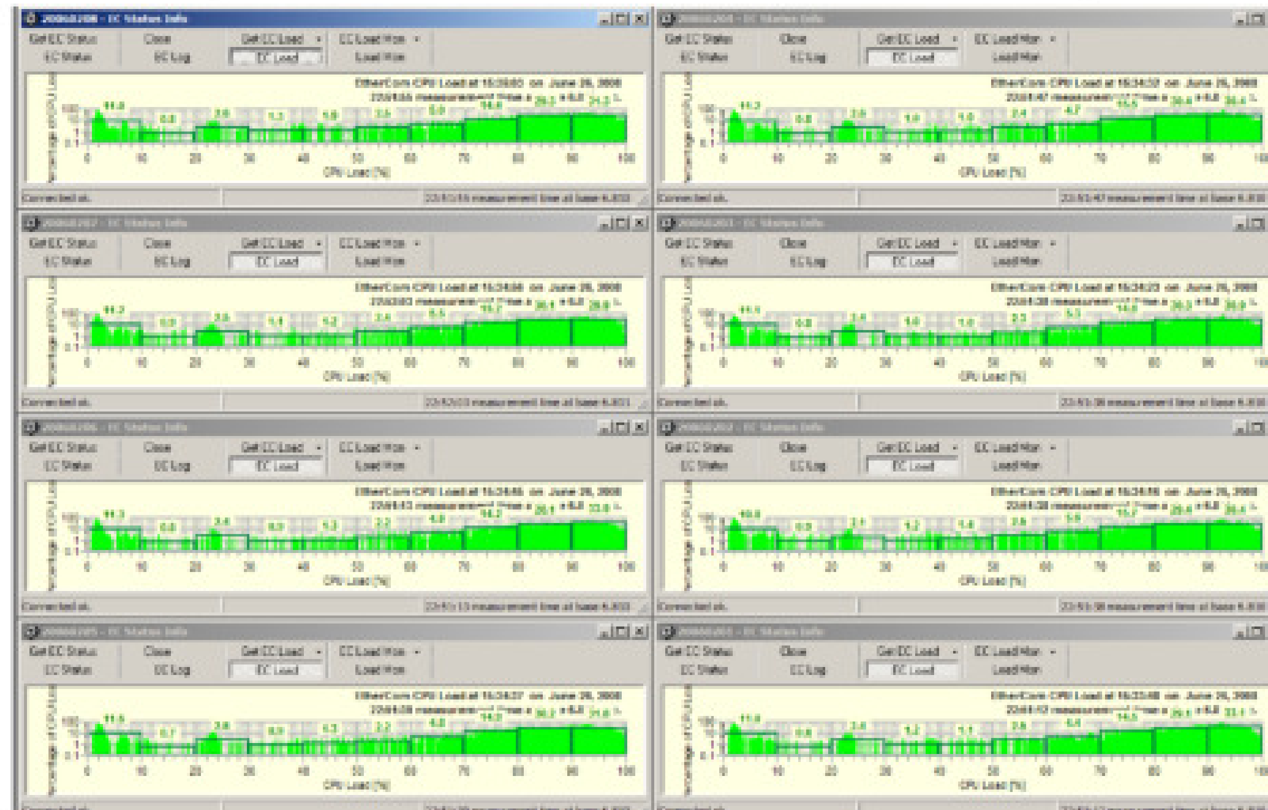
Work Load: 84.8% *EC CPU Load:* 81.5% *CP Even & Task Load:* 104.0%

Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

Main EC issues test made at 26.06.2008 under standard test conditions and the following actions:

- Test was start before 11:00 after first night one (25-26.06.2008) without restarting of all 8 CP/EC and using the same test conditions;
- Elas' CNWS was updates between 11:00 and 11:30;
- At 11:52 and 11:57 Linux server (where L2TP and PPP servers are installed) was rebooted – it takes approximately 2 min to come ready;
- At 12:02 Linux server (where main routing and NistNet are installed) was rebooted – it takes approximately 2 min to come ready;
- At 12:05 Edimax router was rebooted – it takes approximately 2 min to come ready. All CPs gets new IPs between 12:27 and 12.28;
- At 12:30 Edimax router was disconnected – it takes approximately 45 sec all CPs to show LAN Trouble;
- At 12:30 DLink router was connected – it takes approximately 30 sec all CPs to continue and 40sec to drop LAN Trouble. All CPs get new IPs at 12:42;
- DLink's LAN cable was disconnected for time of ½, ¾ and the time when DHCP lease expired. 4 of the CPs get IPs at 13:28 while other 4 – at 13:45;
- DLink's router was changed with Edimax at 13:50 – it takes 2-3 min all CPs to continue normal work;
- Edimax' LAN was disconnected for 6 minutes at 13:54 and 14:09 when DHCP lease events was expected;
- Edimax' LAN was in connected condition at 14:27 and 14:42 – all CPs gets new IPs as expected (4 of them at 14:27 and 4 others at 14:42);
- Edimax router was changed with DLink's at 14:58 – All 8 CPs get new IPs at 15:00;
- Test was finished at 15:15.

Snapshoot of all ECs CPU load for the test was shown on the picture below



The following standard conditions are used at test:

- IIS was set to require client certificates from ECs;
- Elas was set to use 30 sec alive times;
- Edimax router was set to use L2TP (1392b) WAN settings;
- Edimax' DHCP server was set to offer 30 min lease time;
- DLink router was set to use static WAN settings;
- DLink's DHCP server was set to offer 60 min lease time;
- Both routers' DHCP servers use the same IP range;
- NistNet was set at 2/1,1,1 conditions;
- LogCollector was set as usual test case (5,4,3 min);
- WU access simulator was set to reach maximal CP task load;
- All CPs have 2 sensors generation events every 1 min;
- All CPs have 3 more PIR sensors.

EC load parameters are the following (like 25-26.06.2008 test):

- EC CPU performance load – 81,5%;
- CP Event & Task Load – 104.0%;
- CP Work Load – 84.8%.

Notes:

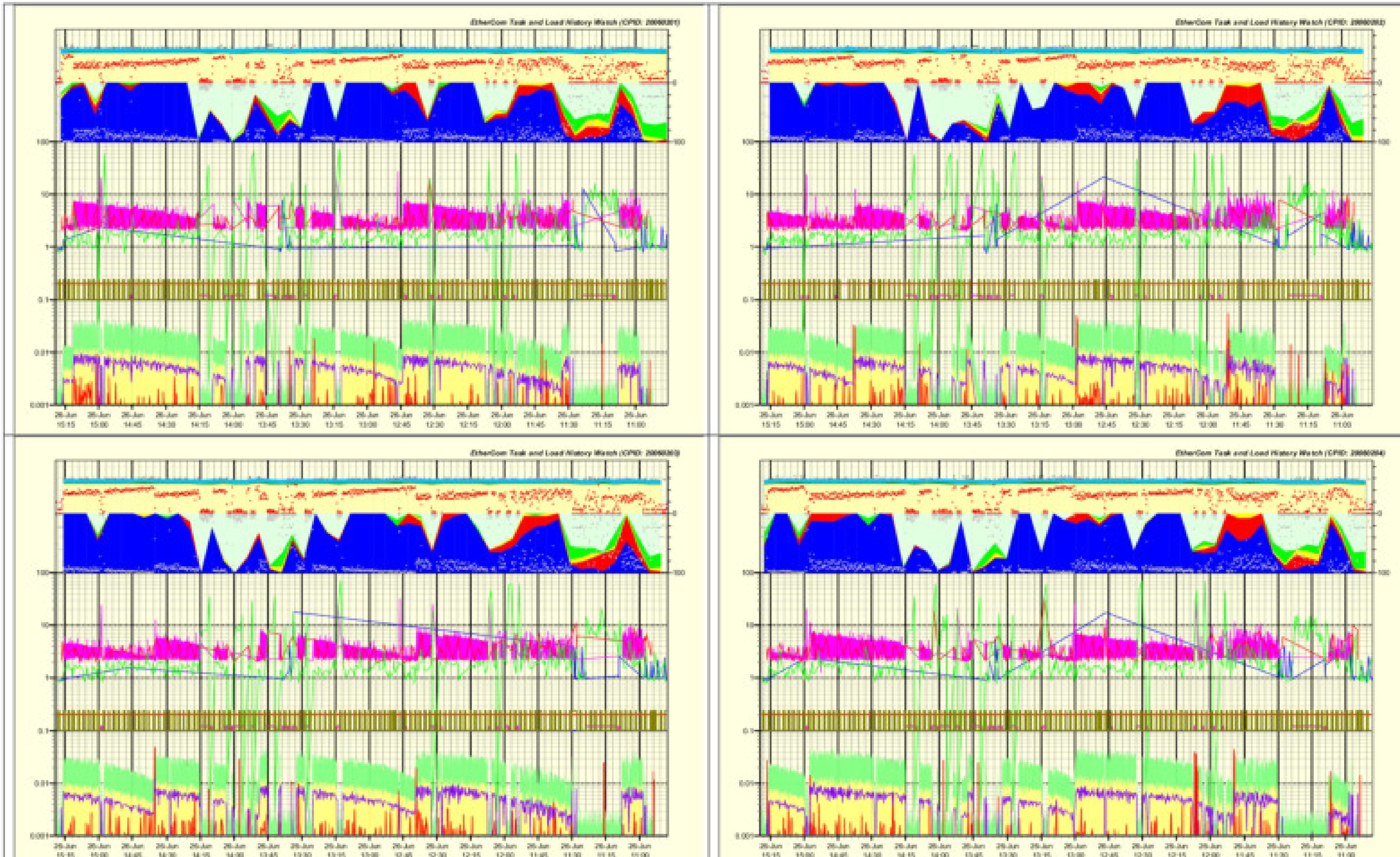
- n.a.Factor and WU Errors was not monitored;
- CP/EC load parameters ware not calculated;
- High resolution history and packed capturing is used as well.

Conclusion:

- All ECs worked as expected without any bugs manifestation;
- EC firmware is quite responsive and adequate at all changes.

Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

Snapshots of all EC History records for the whole test on 26.06.2008



Long Time Extremely High Load Tests of Broadband Firmware CP ver. 0.49 and EC ver. 1.08 (25.06-15.07.2008)

Snapshots of all EC History records for the whole test on 26.06.2008 (continued)

Test conditions (WU requests by testCPProxy):

- All CP/ECs are requested continuously and alternatively to perform SetCPState and GetCPLog with 0.5sec pause between. Periodically (when user access log records exceed 64) CP log is cleared.

Test conditions (RP requests by LogCollector):

- UploadCPLog: 3min;
- UploadECLog: 2min;
- UploadData: 1min.

Test conditions (general):

- Load assessment for 9 CP/ECs tested for more than 500 CP * hours total;
- Elas is set (for CP 20060201 … 20060204 only) up to not send events to Elpx/…/Sims;
- All ECs are with Release firmware build based on HyNetOS release ver. 2.4;
- 20060201, 20060202and 20060206 (old EC hardware).

Test conditions (EC settings):

- All ECs are connected to ECMon for monitoring;
- All ECs was restarted (because of firmware update) before the test.

Problems under current investigation (summary):

- ***CP/EC inaccessibility: see table below***

CPID	*EC restarts (takes 2-5min)*	*EC Task Loss (takes 10min)*	*CP Timeouts (takes ~3min)*	*Inaccessibility time (~min)*
20060201		4	14	***82***
20060202		5	12	***86***
20060203		3	5	***45***
20060204		3	9	***57***
20060205		3	10	***60***
20060206		3	14	***72***
20060207		3	9	***57***
20060208	1	1	7	***34***
20060209			4	***12***
All CP/EC	**1**	**25**	**84**	**505**
Total [%]	**0.01%**	**0.82%**	**0.83%**	**1.66%**

Note: CP/EC inaccessibility percentage calculation was based on total observation time 506.6 CP * hours (for all panels under on-line monitoring).

Tests details summary:

- Test was started after 21:00 on 31.03.2005;
- Test was ended at ~9:00 on 03.04.2005;
- Mean Elas: CPU load was ~70%, Memory usage raised up to 2.3GB;
- Full test restart (without CP/ECs) was done 27 hours after the start;
- Off-line processing is made to summarize results;
- All values in this document have to be used only for investigation purposes and not referred to other (incl. normal work) conditions.

Test conditions (CP settings):

- All CPs are set with the same parameters (based on defaults);
- All CPs are set not to report via PSTN, no report cycles;
- All CPs have 2 detectors generating events automatically every 1min;
- All CPs was restarted (because of firmware update) before the test.

Test conditions (Elas settings):

- Alive cycle: 300sec (armed);
- Alive cycle: 600sec (disarmed);
- Service Providers were disabled for CP 20060201 … 20060204;
- Service Providers were set for CP 20060205 … 20060209.

Other problems summary (including once described previously):

- ***Event Reporting***
  - Elas say "***SPPXNotAvailable***" after 2:20 on 02.04.2006 for CP/EC 20060205 … 20060208 (Elpx previously available);
  - Delays in Event reporting chain (Elas/Elpx/…/Sims) could exceed 30sec forcing EC to timeout HTTP session every 30sec.
- ***Task Performing***
  - Elas has to send NoTask first after EC start/restart;
  - Elas say "***ASCPLostConnection***" after daylight time change – it takes up to next CP/EC alive connection (from tests on 27-27.03).
- ***CP/EC Communication***
  - Internal CP Serial Bus is bandwidth overloaded and collisions reached high levels for long time intervals at Get/Upload Log, Download All parameters and at CP Image transfer to EC (at EC start-up and CP communication parameters changes);
  - Incorrect CP parameters changes (causes CP loss) and EC performance degradation (causes EC loss) at Download All CP parameters (from tests on 28.02 and 27-28.03);
  - High memory circulation in EC at GetCPLog after its clearing.

Problems solved until now (thanks to EC monitoring technology):

- ***CP/EC:***
  - Event report and RP/WU task concurrency broaden (up to highest level possible in current architecture and implementation);
  - Expanding CP/EC reliability and stability making them operational for long time periods at maximal possible Event/Task intensities;
  - Decreasing CP/EC inaccessibility and respective errors to lowest possible in current system architecture levels;
  - Many bugs in CP firmware (including once coming from original Infinite code) caused restarts, memory faults, deadlocks etc.;
  - Many bugs in EC firmware (including once coming from SND's HyNetOS) caused restarts, memory faults, deadlocks etc.;
  - CP/EC interconnection issues and inter-device deadlocks.
- ***Elas:***
  - CPWS – locking all event reporting chains;
  - CPWS – RP/WU task queue blocking;
  - RPWS/WUWS – task response loosing;
  - WUWS – performance issue at CP log processing.
- ***Elpx:***
  - Service malfunction at automatic pool maintenance;
  - Unavailability at restarting with empty/full queue;

Note: Only major problems are listed above.

Notes:

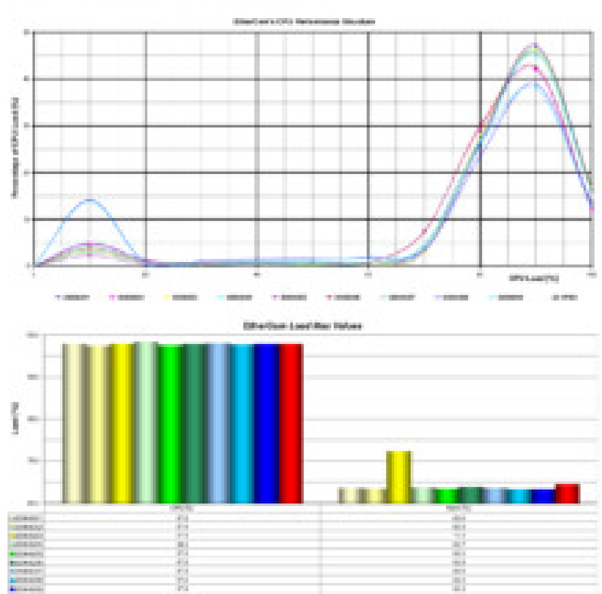
- Currently made and described tests and analyses ware only targeted to CP/EC problems investigation/solving, stability/reliability increasing and finalizing CP/EC release versions.
- Problems described above could be observed rarely at extremely high load and whole system monitoring.
- There are number of potential problem sources (like complete Event reporting chain incl. PSTN backup) not investigated at all.
- Some of the problems on system level ware only observed but not investigated deeper for finding out their sources and solutions.

Conclusions:

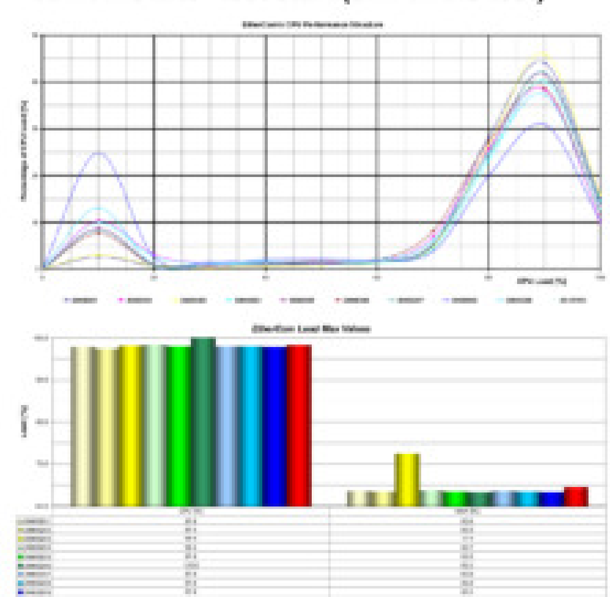
- ✓ All critical bugs in CP/EC firmware were fixed.
- ✓ CP/EC is function as close to requirements as possible.
- ✓ CP/EC firmware is proven to work in required environment.
- ✓ CP/EC is proven to work at peak load for long time periods.
- ✓ CP/EC firmware is ready to be released and used for production.
- ✓ As side results many problems on architectural and system level ware found and some of them solved.

EtherCom CPU and Memory Usage Observation

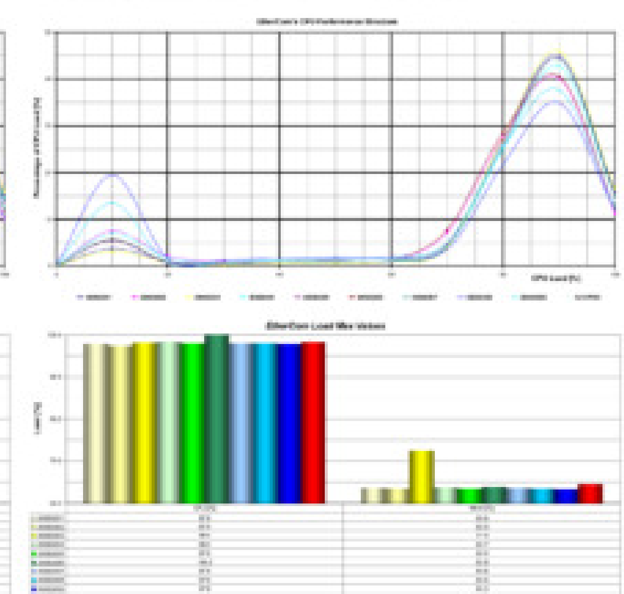
~27 hours after test start



~60 hours after test start (end of the test)



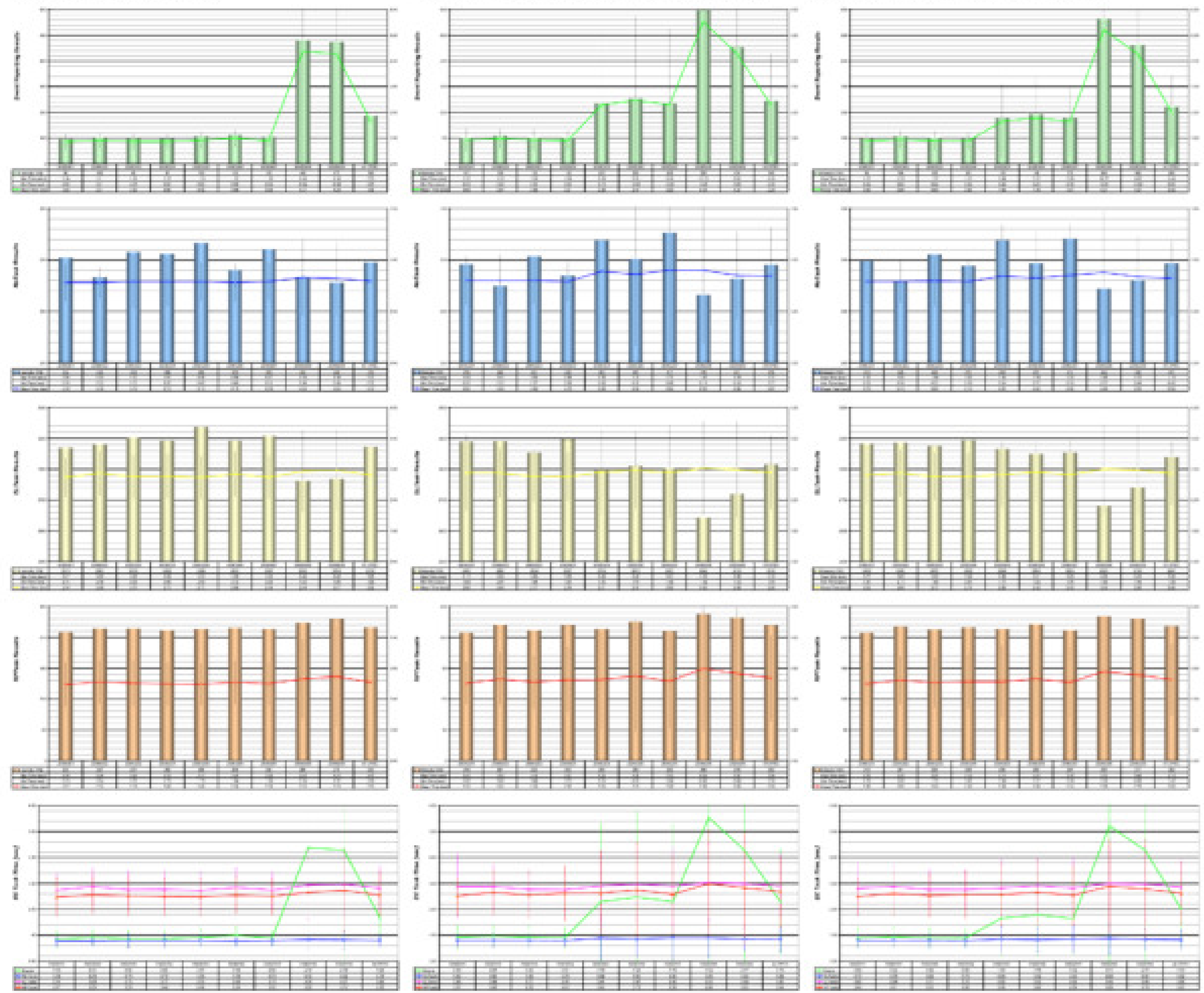
Result summarized for whole test



EtherCom On- Line Statistics Results Processing

Task Intensities and Lengths Observation

~27 hours after test start | **~60 hours after test start (end of the test)** | **Result summarized for whole test**



Min and Max values in above charts was calculated on standard deviation captured at on-line monitoring.

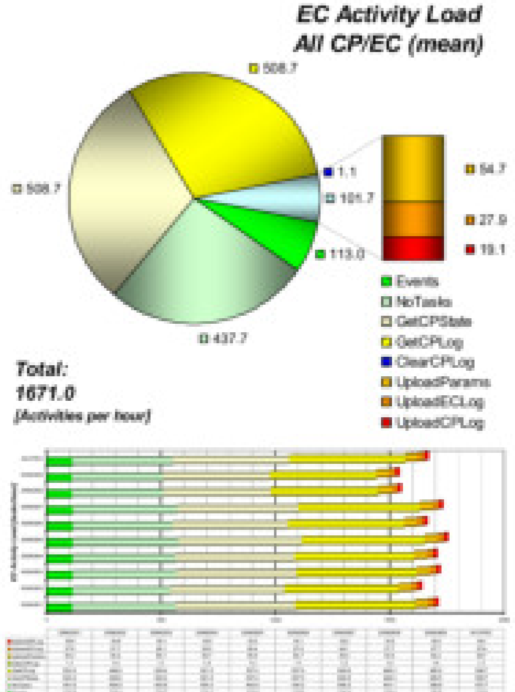
EtherCom Off-Line Statistics Results Processing

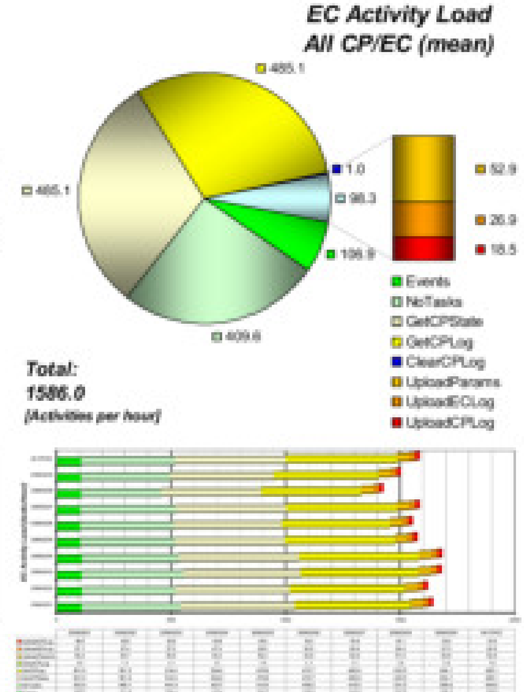
EtherCom Load Assessment Snapshots

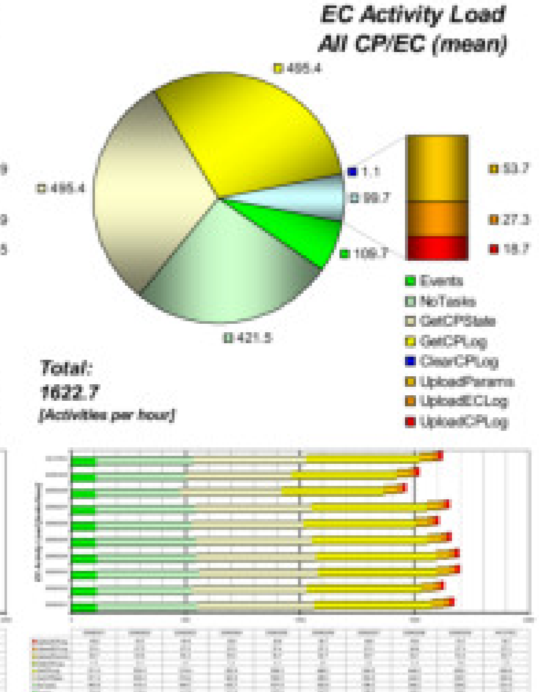
~27 hours after test start

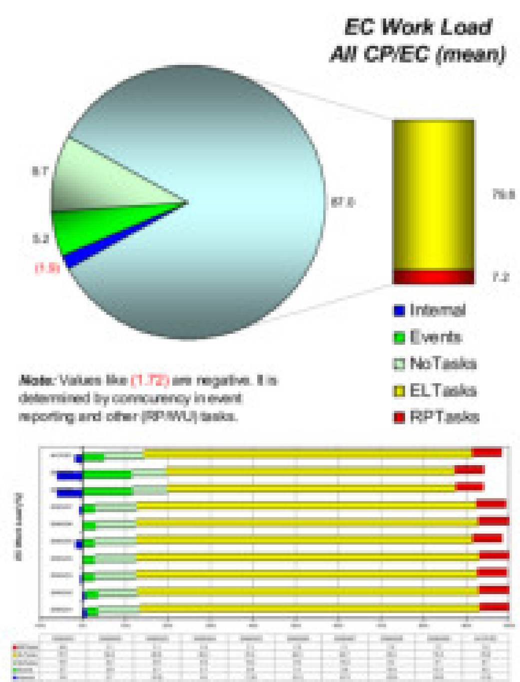
~60 hours after test start (end of the test)

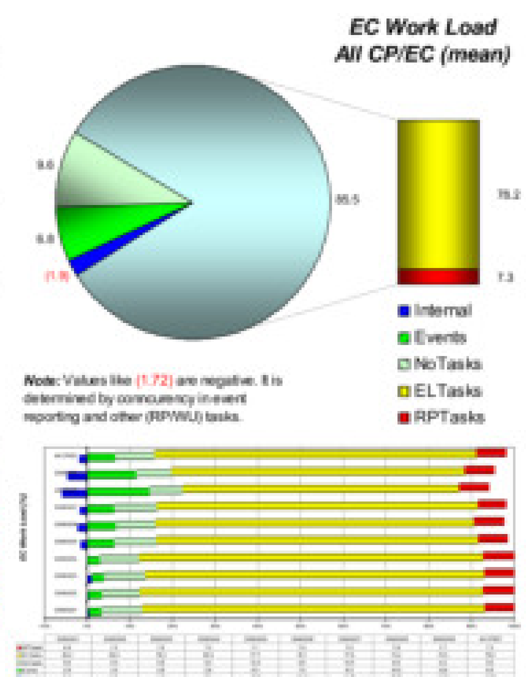
Result summarized for whole test

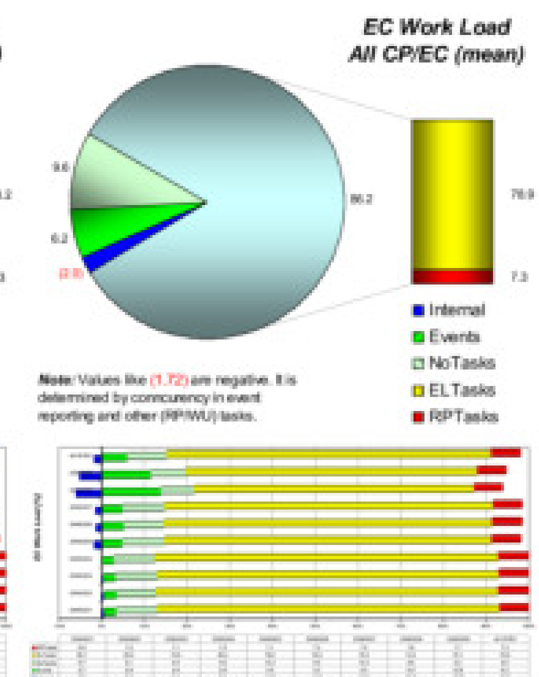




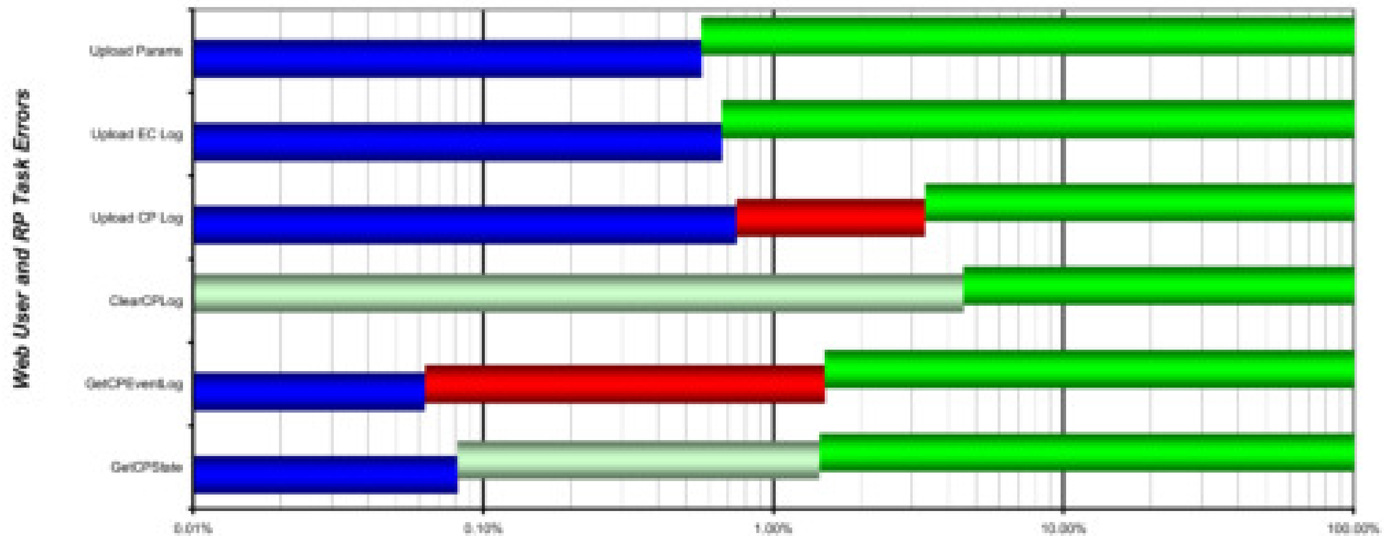






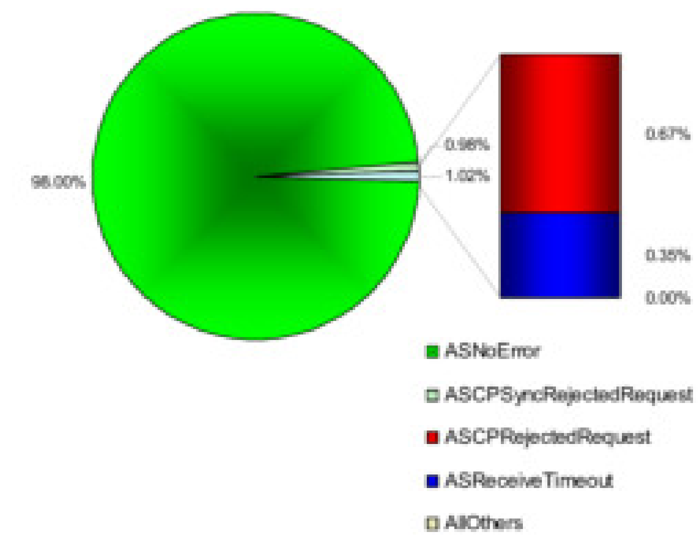


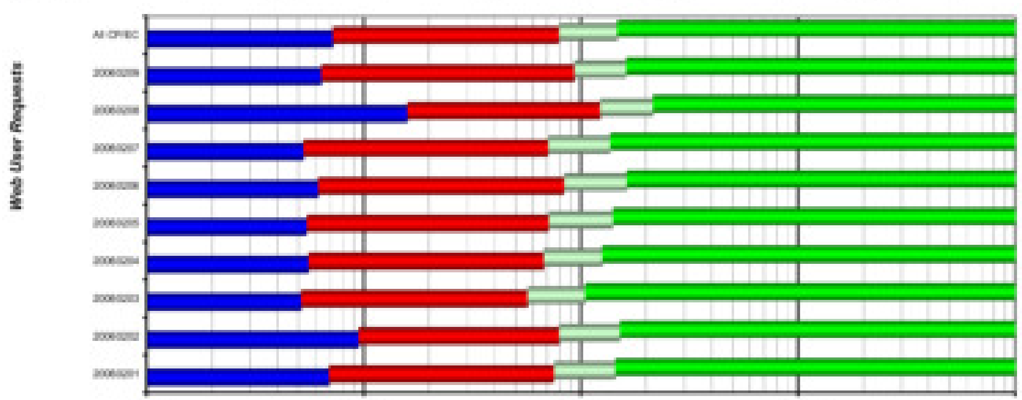
Final Statistics Results



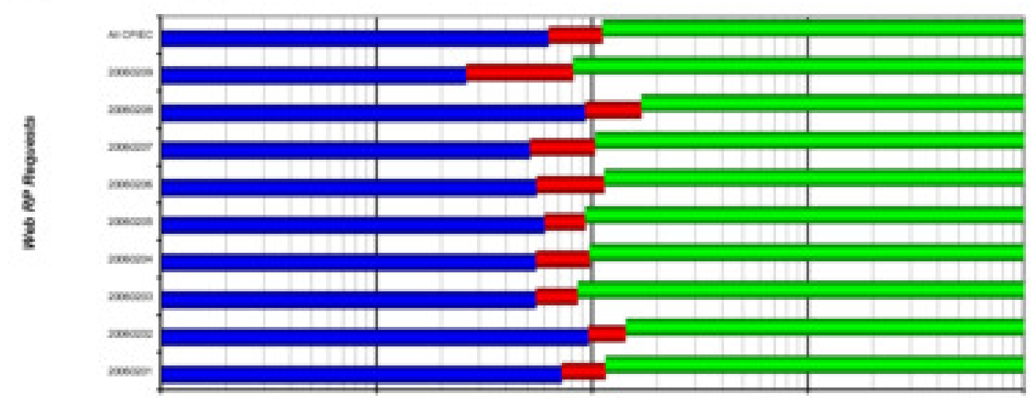
	GetCPState	GetCPEventLog	ClearCPLog	Upload CP Log	Upload EC Log	Upload Params
ASNoError	98.56%	98.50%	95.48%	96.67%	99.34%	99.43%
ASCPSyncRejectedRequest	1.36%	0.00%	4.52%	0.00%	0.00%	0.00%
ASCPRejectedRequest	0.00%	1.44%	0.00%	2.59%	0.00%	0.00%
ASReceiveTimeout	0.08%	0.06%	0.00%	0.74%	0.66%	0.57%
AllOthers	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

Task Errors Summary



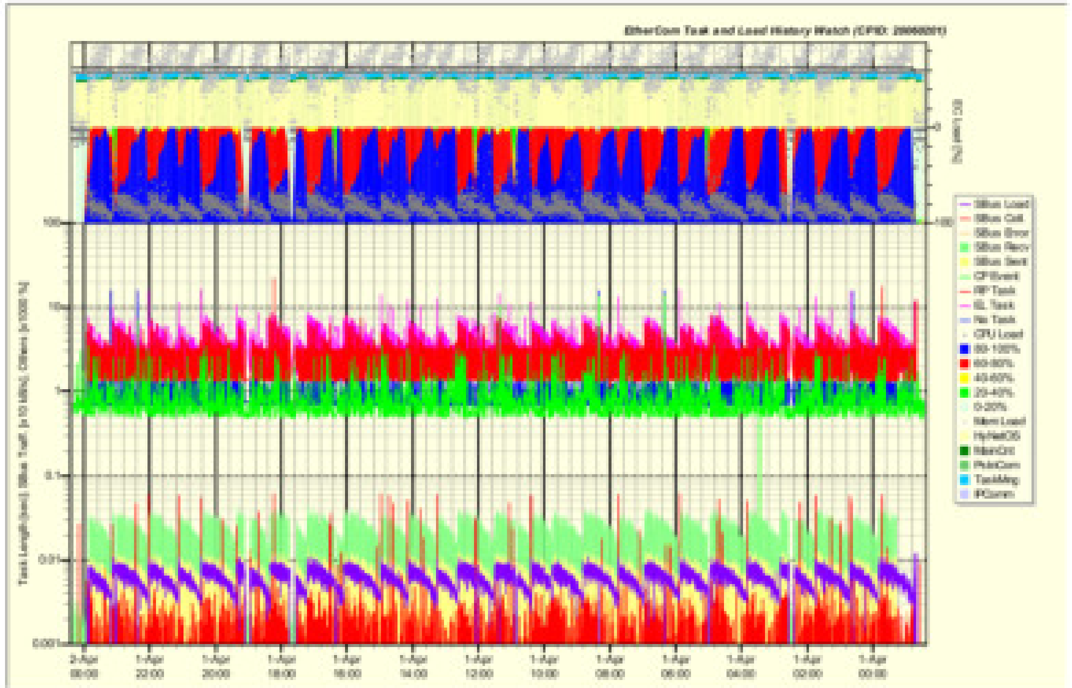


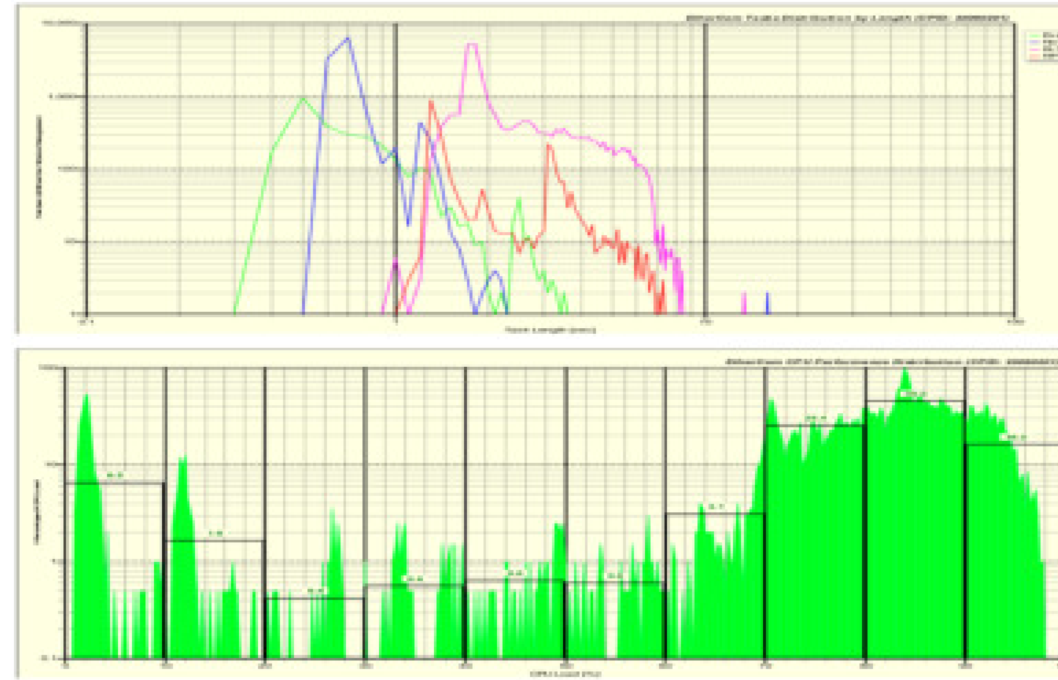
	20060201	20060202	20060203	20060204	20060205	20060206	20060207	20060208	20060209	All CP/EC
ASNoError	98.33%	98.48%	98.95%	98.73%	98.60%	98.37%	98.63%	97.85%	98.39%	98.52%
ASCPSyncRejectedRequest	0.70%	0.72%	0.48%	0.58%	0.69%	0.79%	0.66%	0.92%	0.68%	0.68%
ASCPRejectedRequest	0.88%	0.70%	0.52%	0.62%	0.66%	0.78%	0.65%	1.07%	0.87%	0.72%
ASReceiveTimeout	0.07%	0.09%	0.05%	0.05%	0.05%	0.08%	0.05%	0.16%	0.08%	0.07%
AllOthers	0.00%	0.00%	0.00%	0.00%	0.00%	0.01%	0.00%	0.00%	0.00%	0.00%

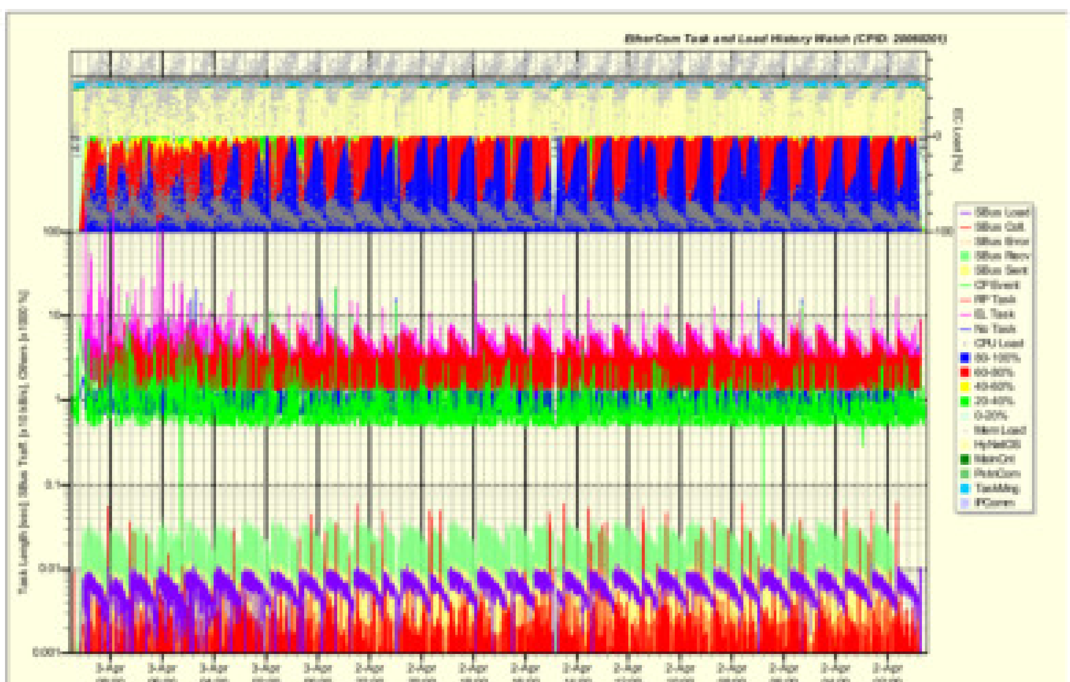


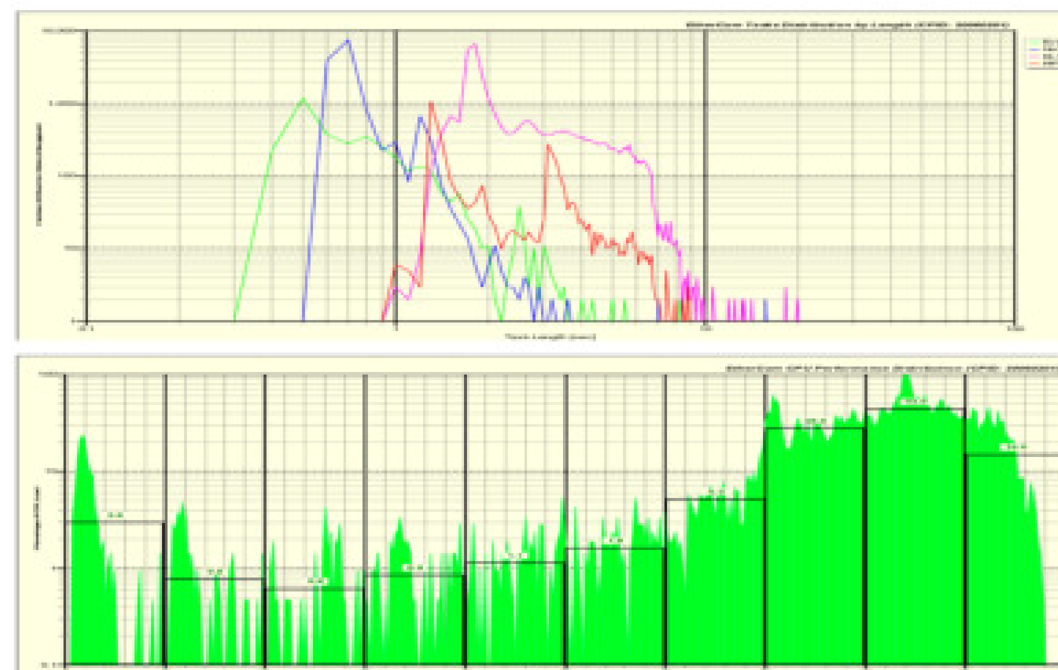
	20060201	20060202	20060203	20060204	20060205	20060206	20060207	20060208	20060209	All CP/EC
ASNoError	98.85%	98.57%	99.14%	99.02%	99.07%	98.86%	98.97%	98.30%	99.18%	98.89%
ASCPSyncRejectedRequest	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
ASCPRejectedRequest	0.43%	0.46%	0.31%	0.43%	0.33%	0.58%	0.52%	0.77%	0.56%	0.49%
ASReceiveTimeout	0.72%	0.96%	0.55%	0.55%	0.60%	0.55%	0.52%	0.93%	0.26%	0.63%
AllOthers	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%

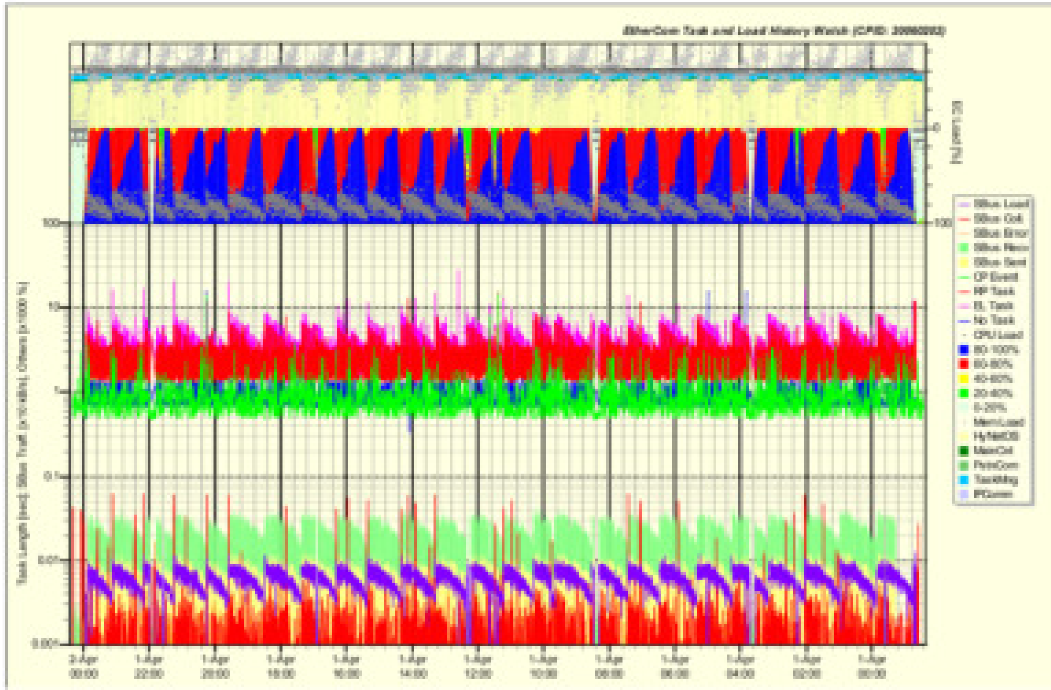
EtherCom Task and Load History Observation

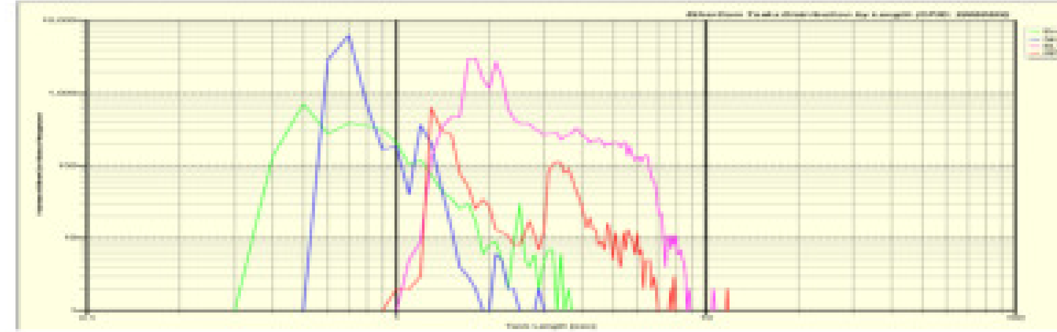


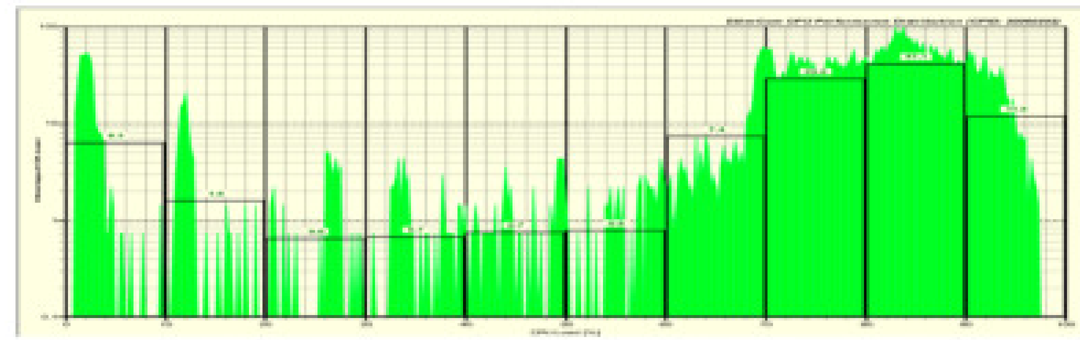


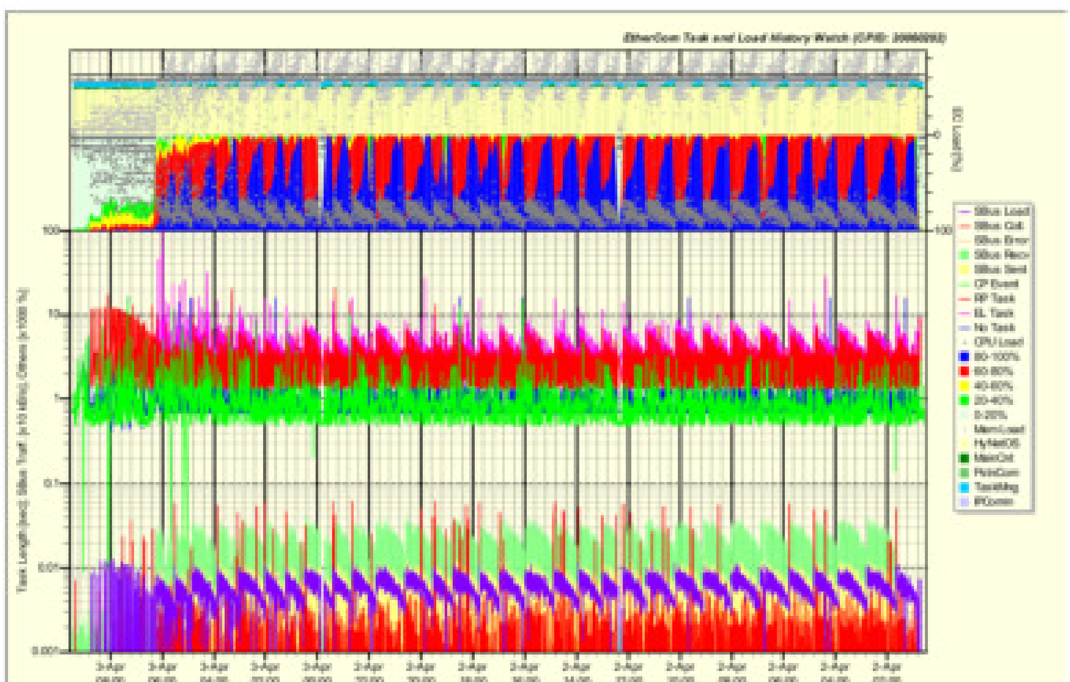


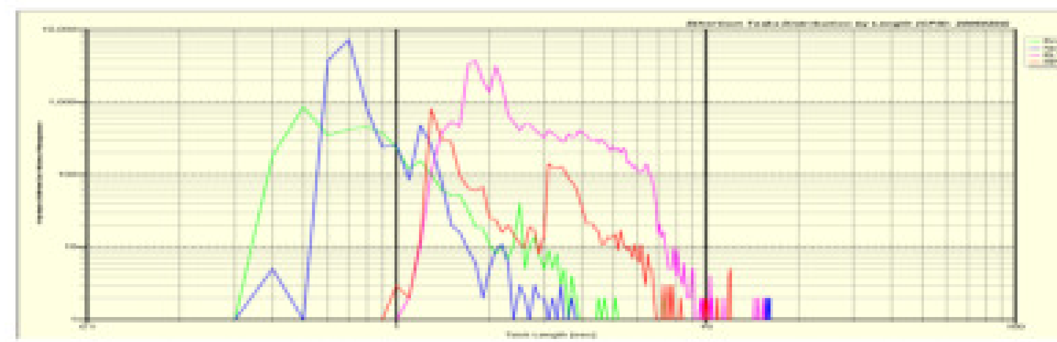


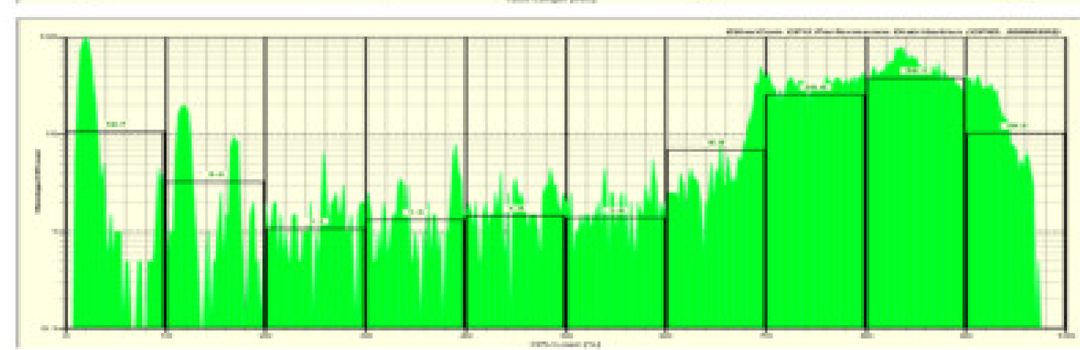

SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCtrl
PstnCom
TaskMng
IPComm

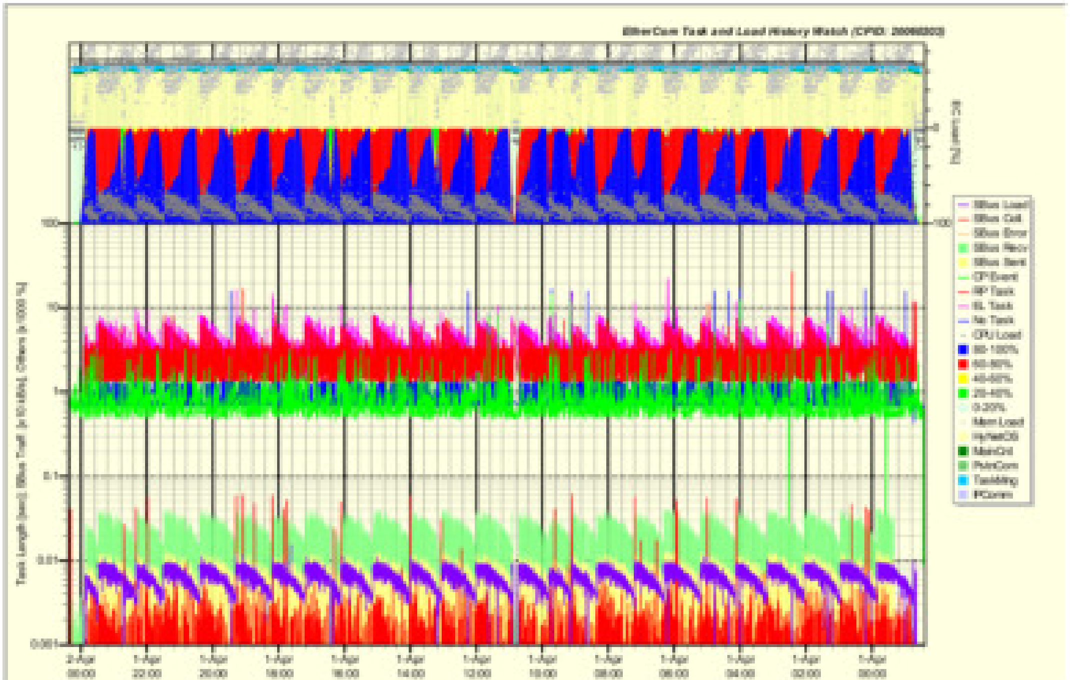


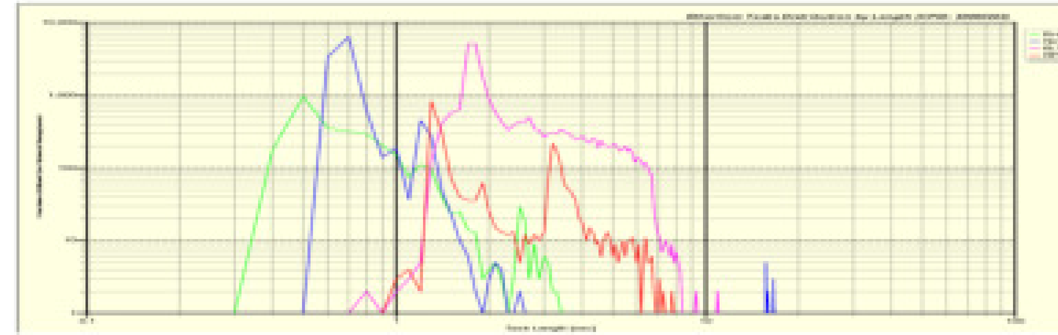


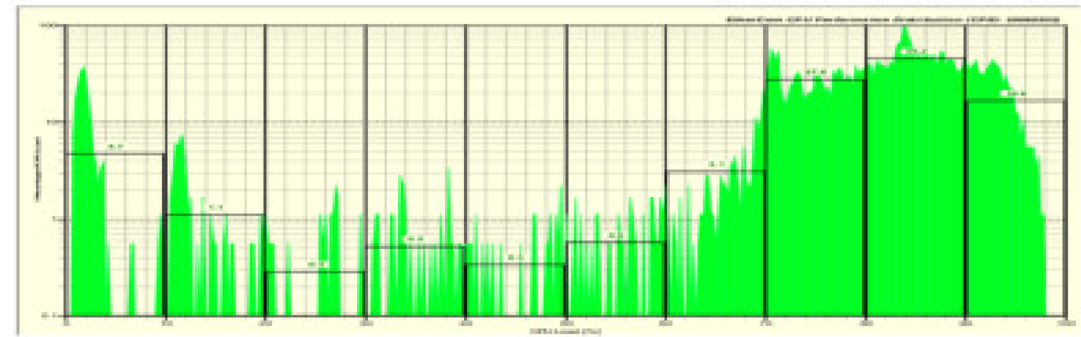

SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCtrl
PstnCom
TaskMng
IPComm

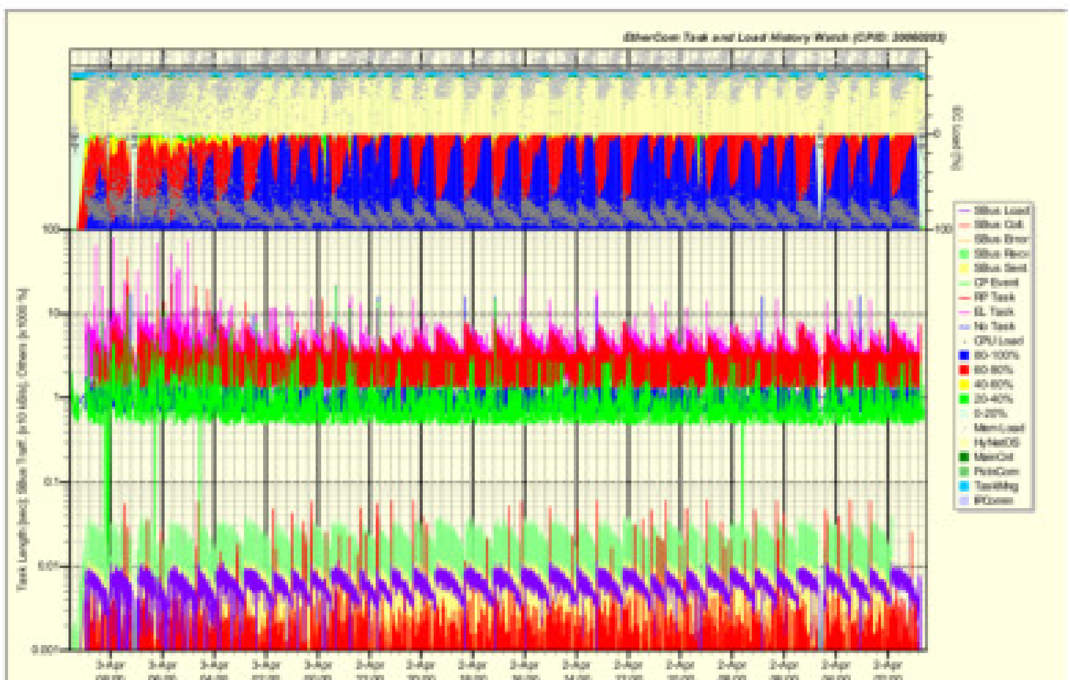


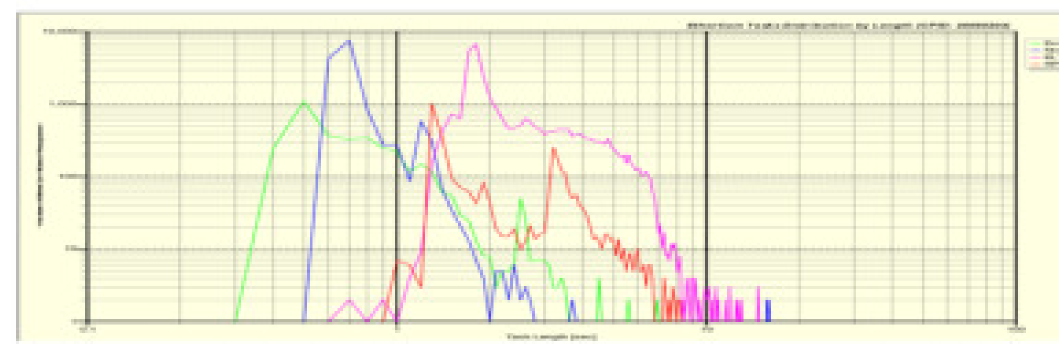


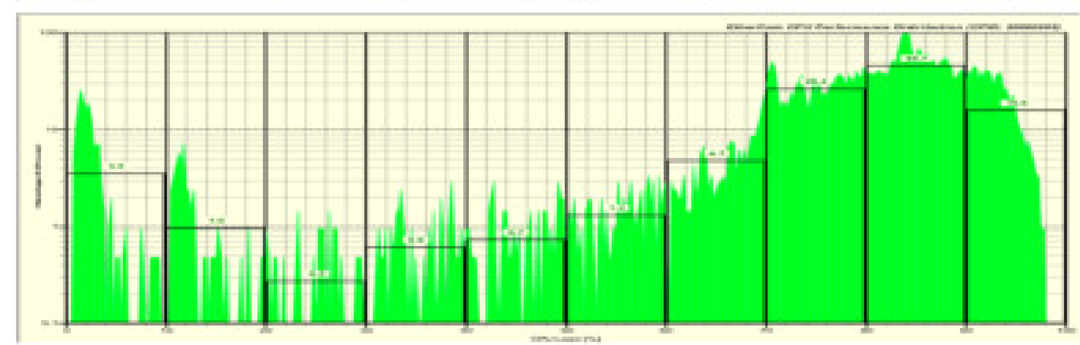

EtherCom Task and Load History Watch
SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCtrl
PrimCom
TaskMng
IPComm

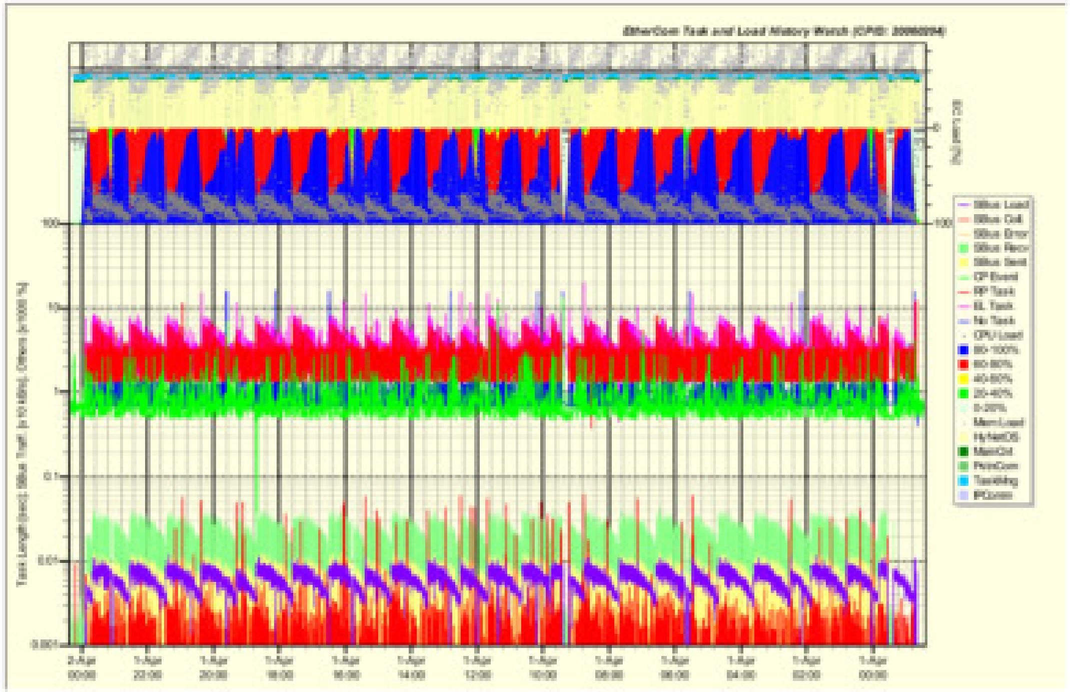


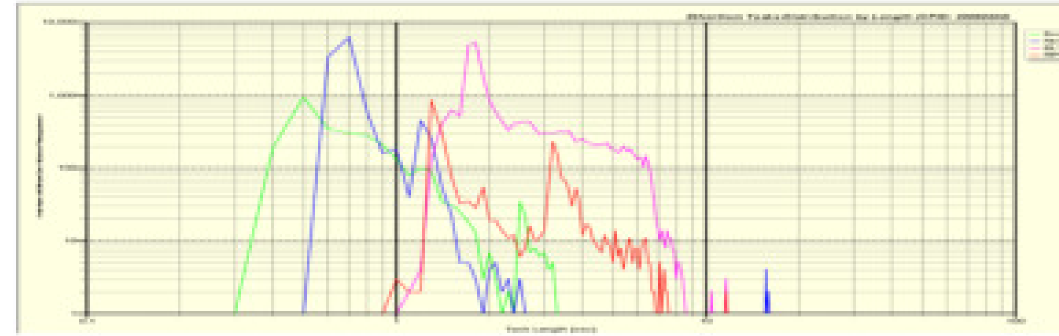


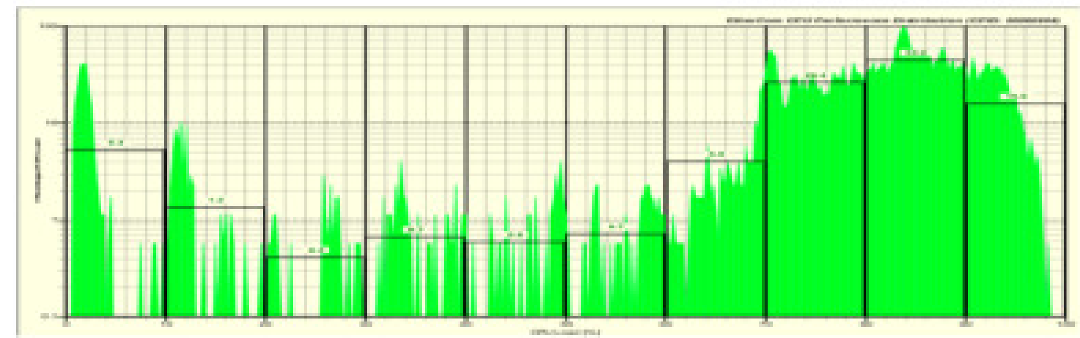

EtherCom Task and Load History Watch
SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCtrl
PrimCom
TaskMng
IPComm

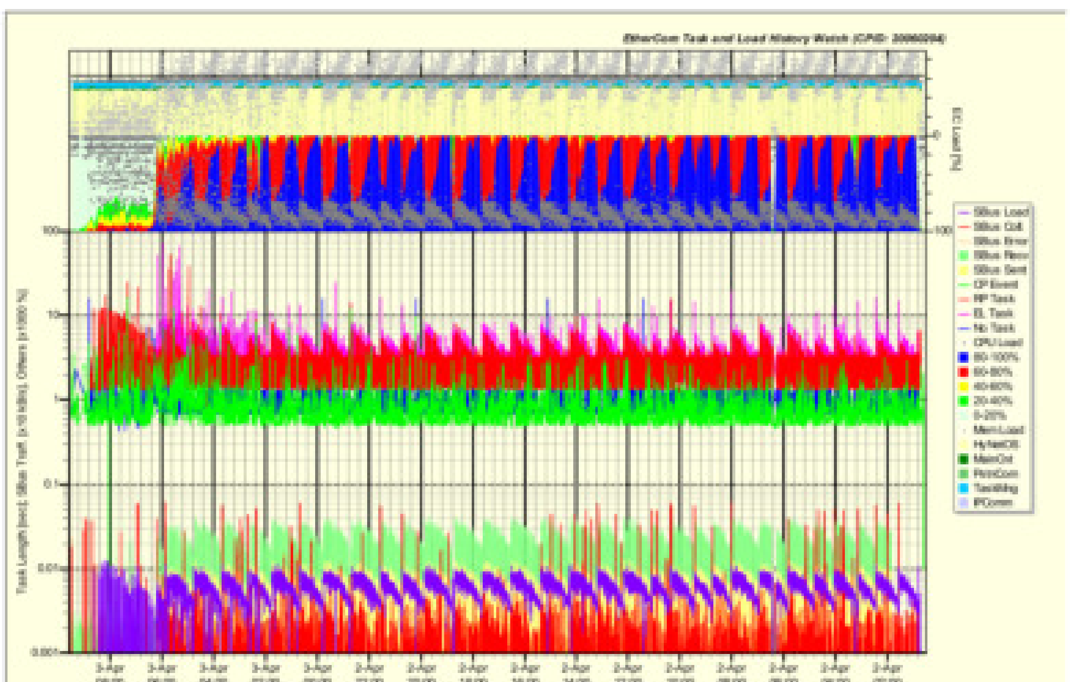


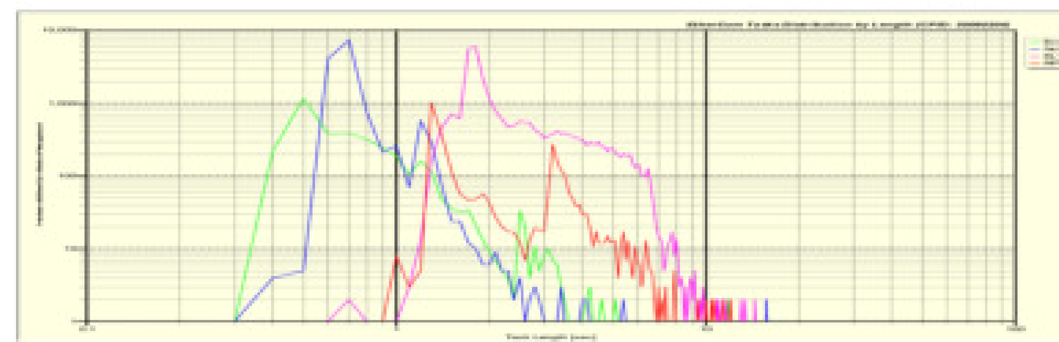


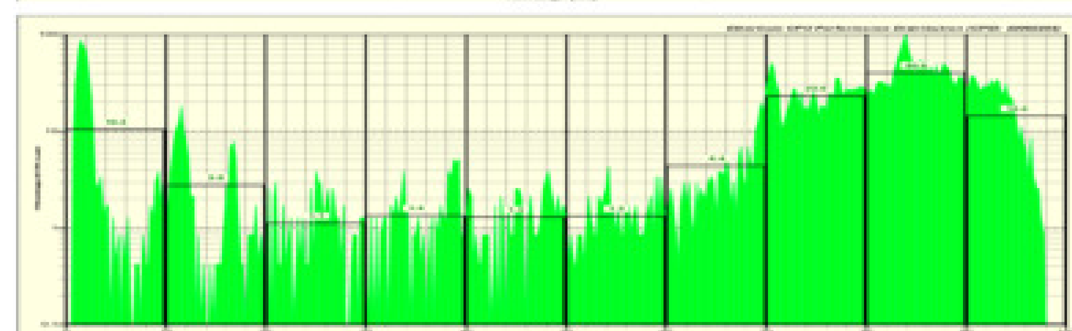

EtherCom Task and Load History Watch
SBus Load
SBus Coll
SBus Error
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
TaskMng

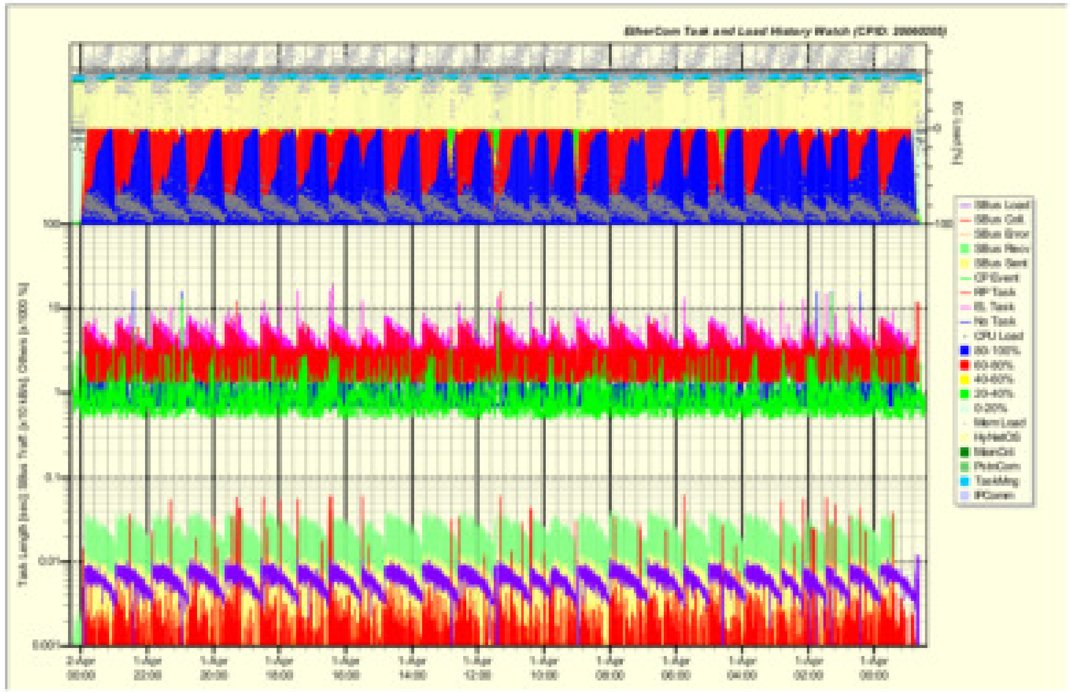


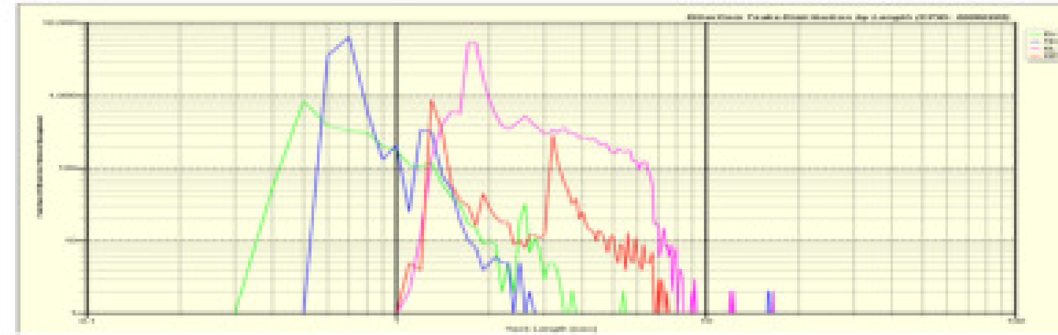


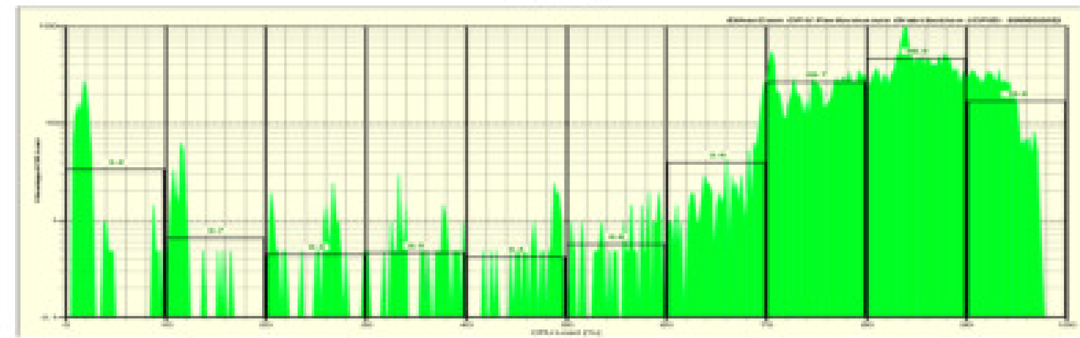

EtherCom Task and Load History Watch
SBus Load
SBus Coll
SBus Error
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
TaskMng

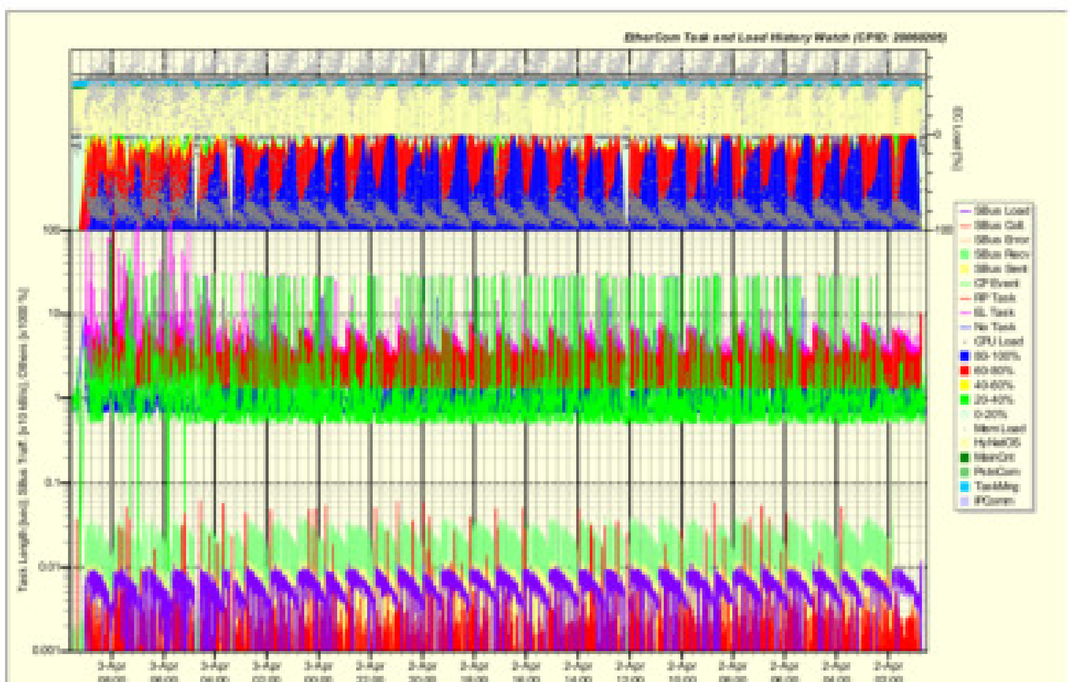


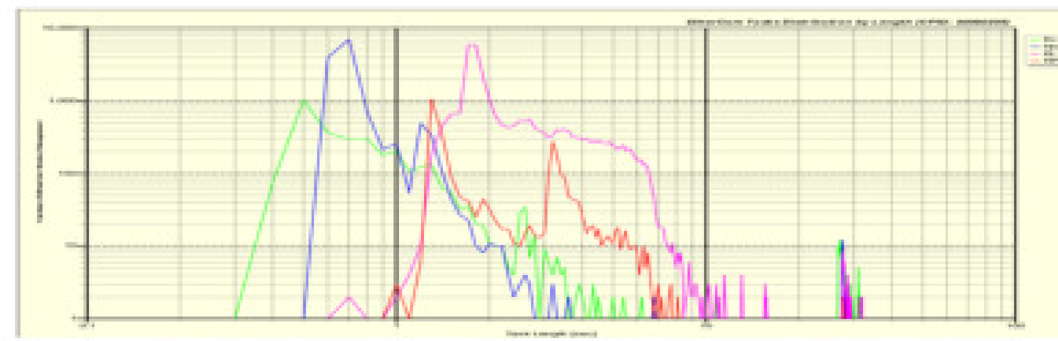


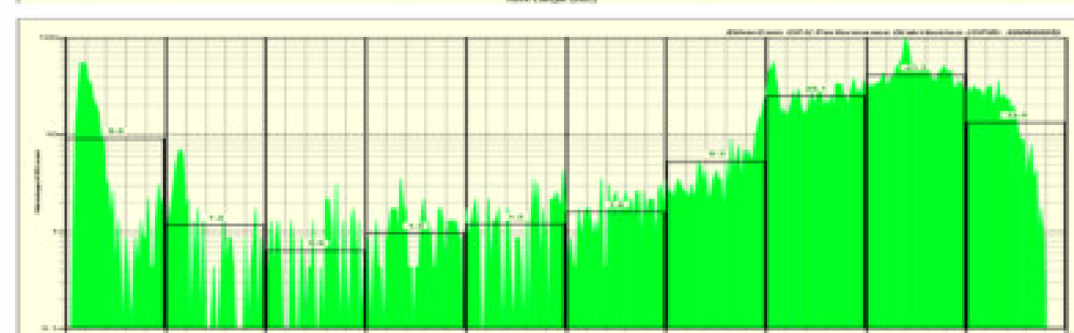

EtherCom Task and Load History Watch (CPID: 20060205)
SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CPEvent
RP Task
SL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCtrl
ProtoCom
TaskMng
IPComm

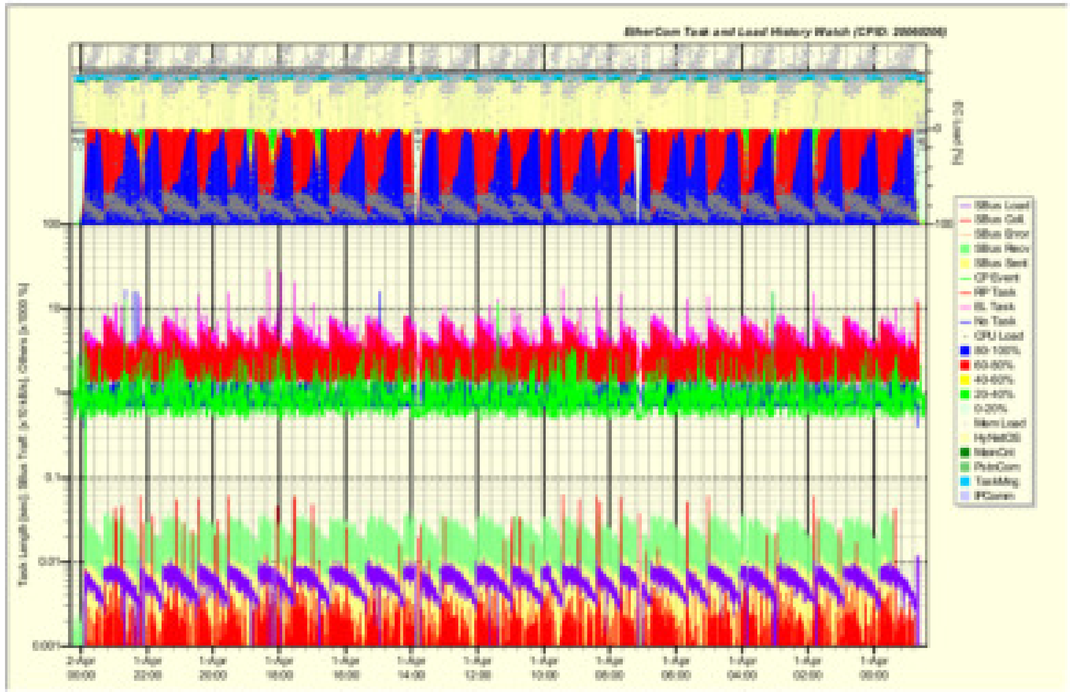


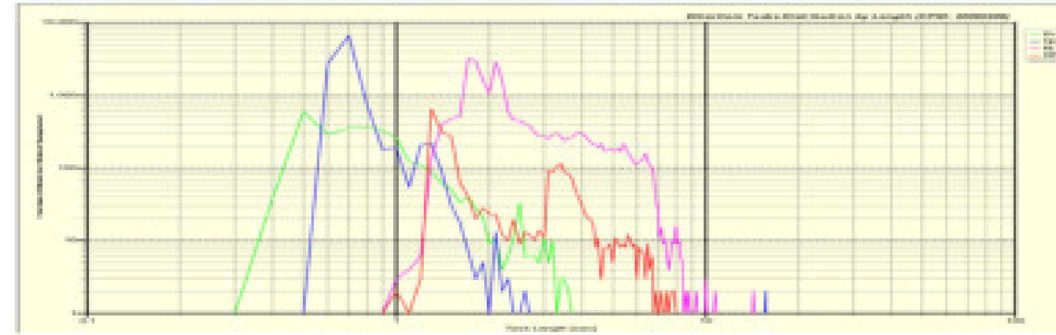


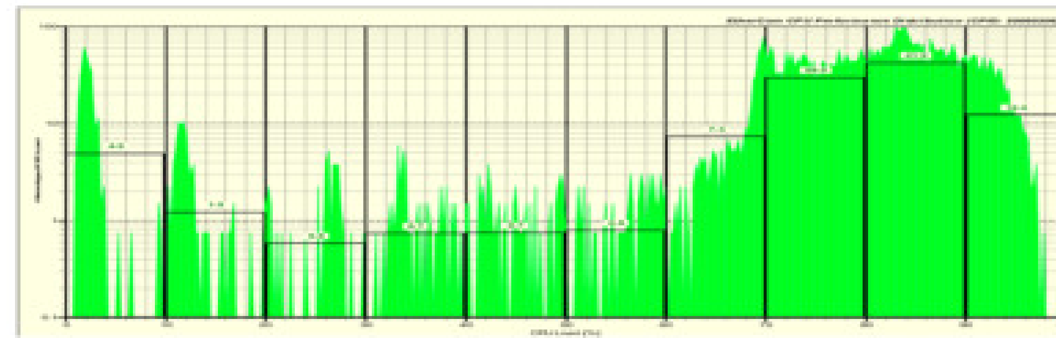

EtherCom Task and Load History Watch (CPID: 20060205)
SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CPEvent
RP Task
SL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCtrl
ProtoCom
TaskMng
IPComm

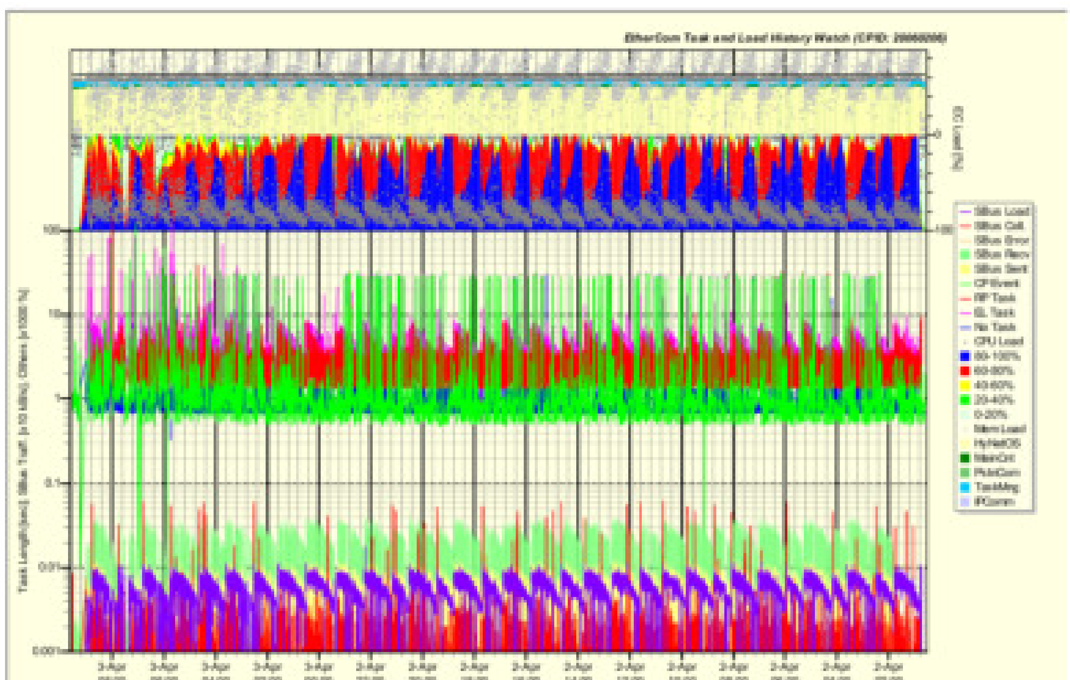


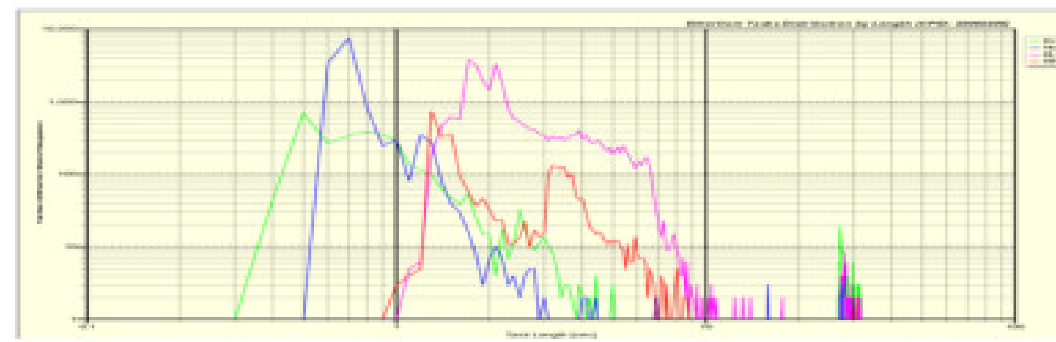


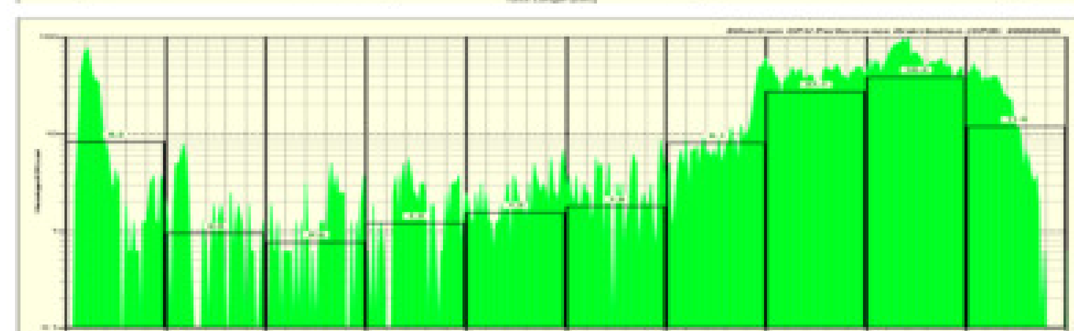

EtherCom Task and Load History Watch (CPID: 20060006)
SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
SL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCnt
PubCom
TaskMng
IPComm

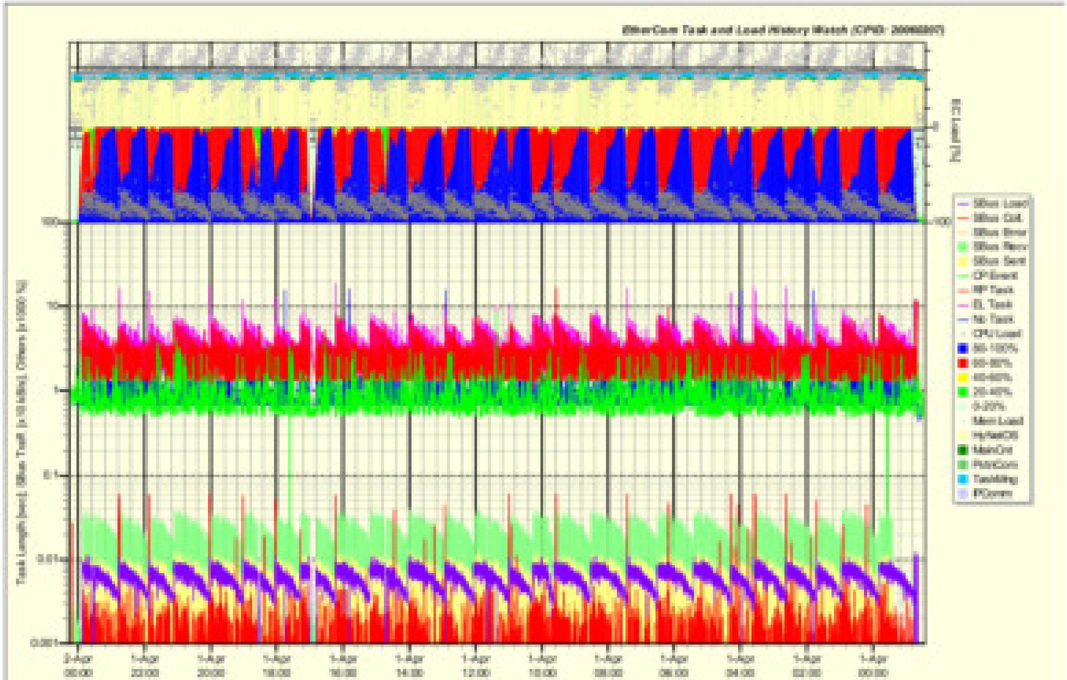


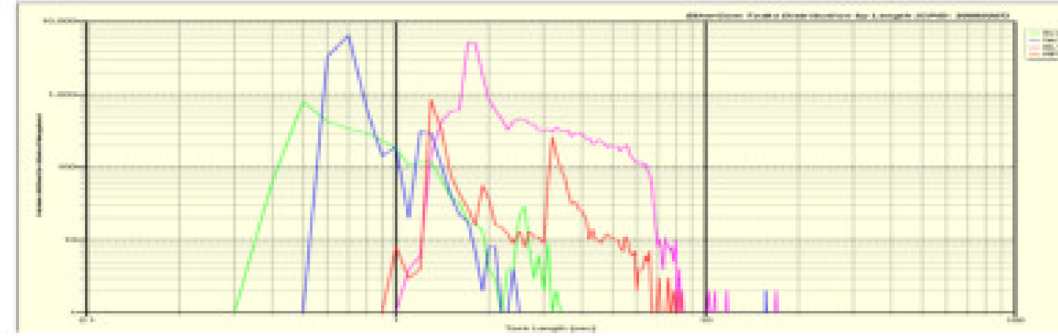


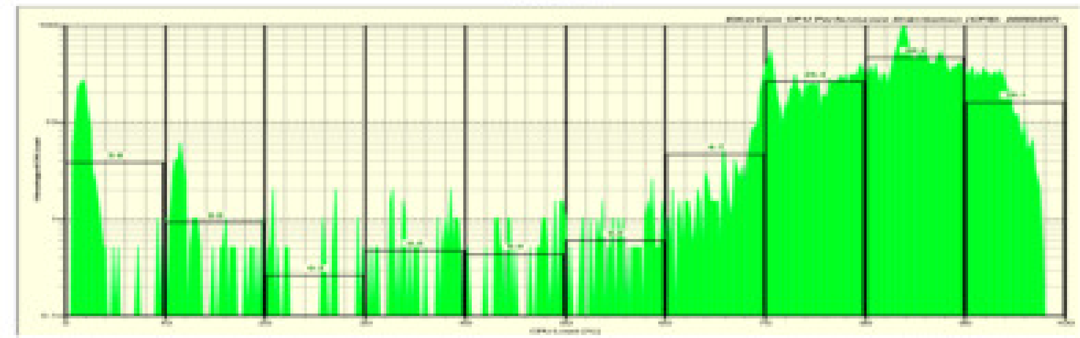

EtherCom Task and Load History Watch (CPID: 20060006)
SBus Load
SBus Coll
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
SL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCnt
PubCom
TaskMng
IPComm

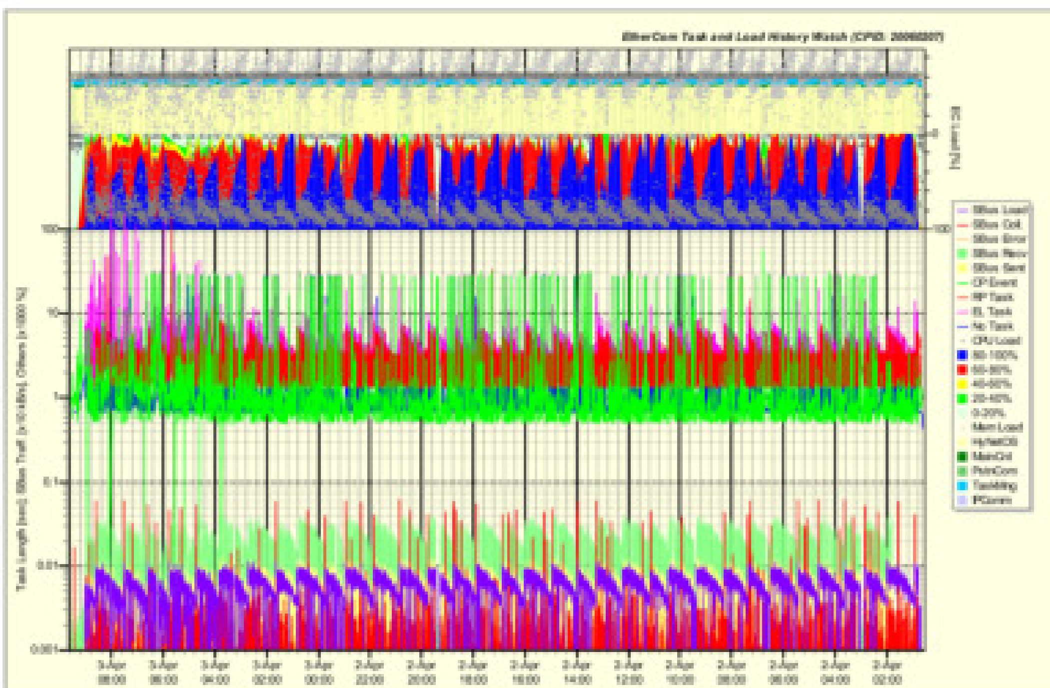


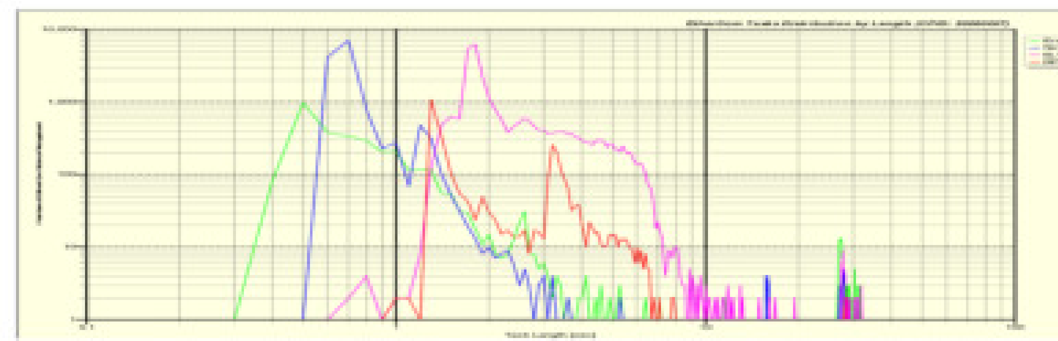


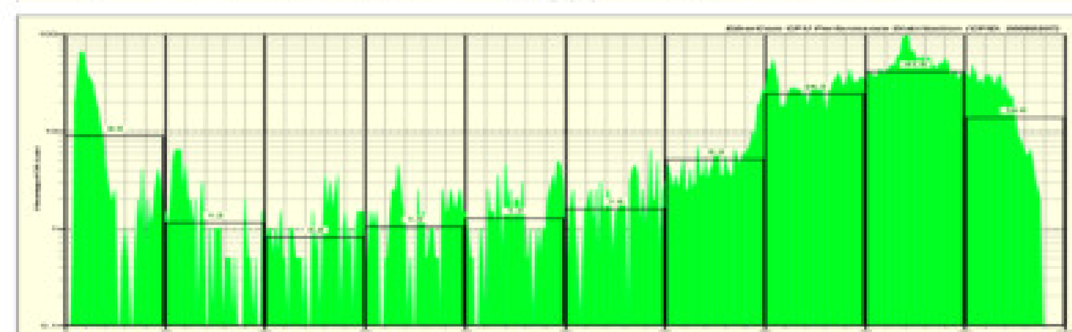


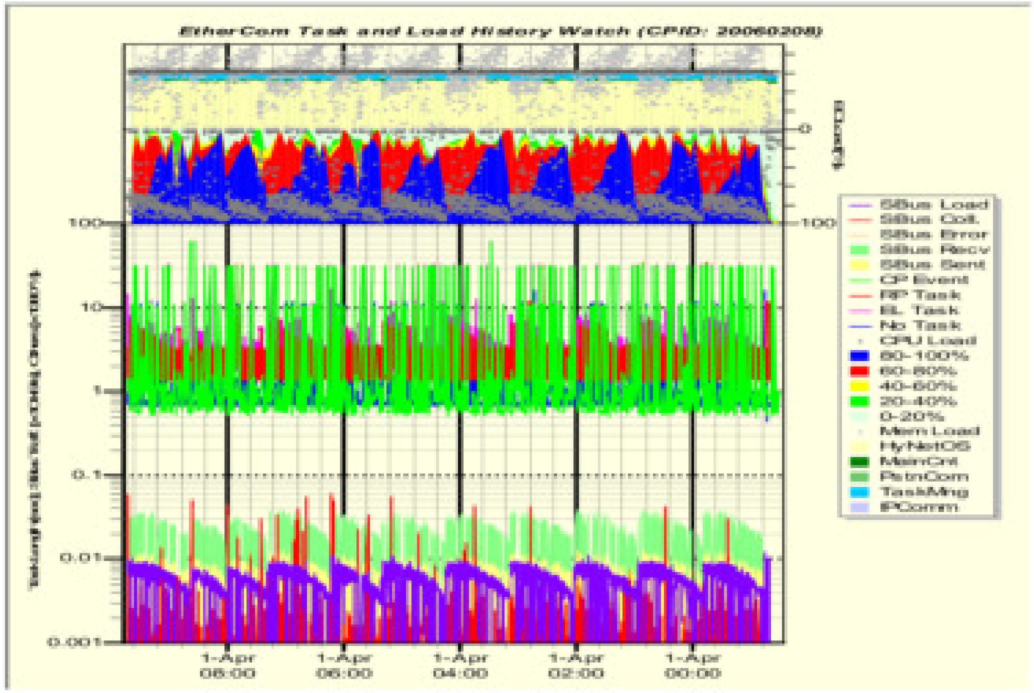


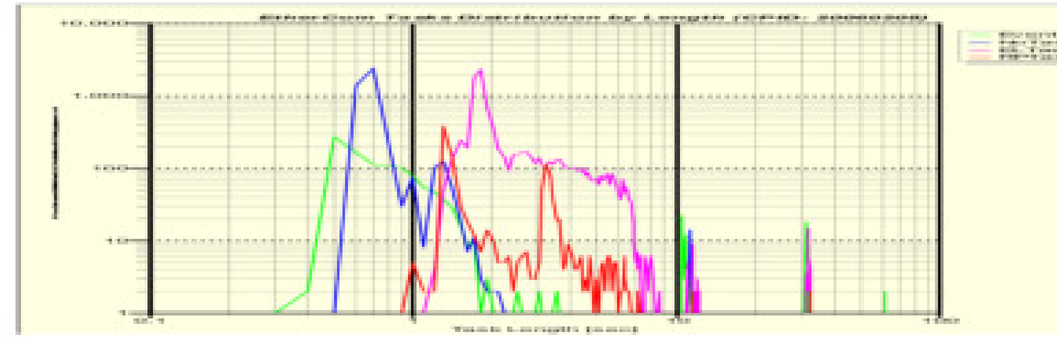


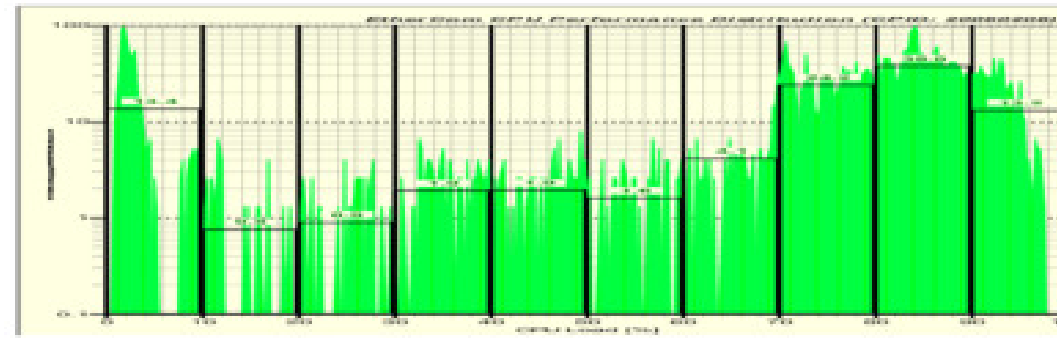


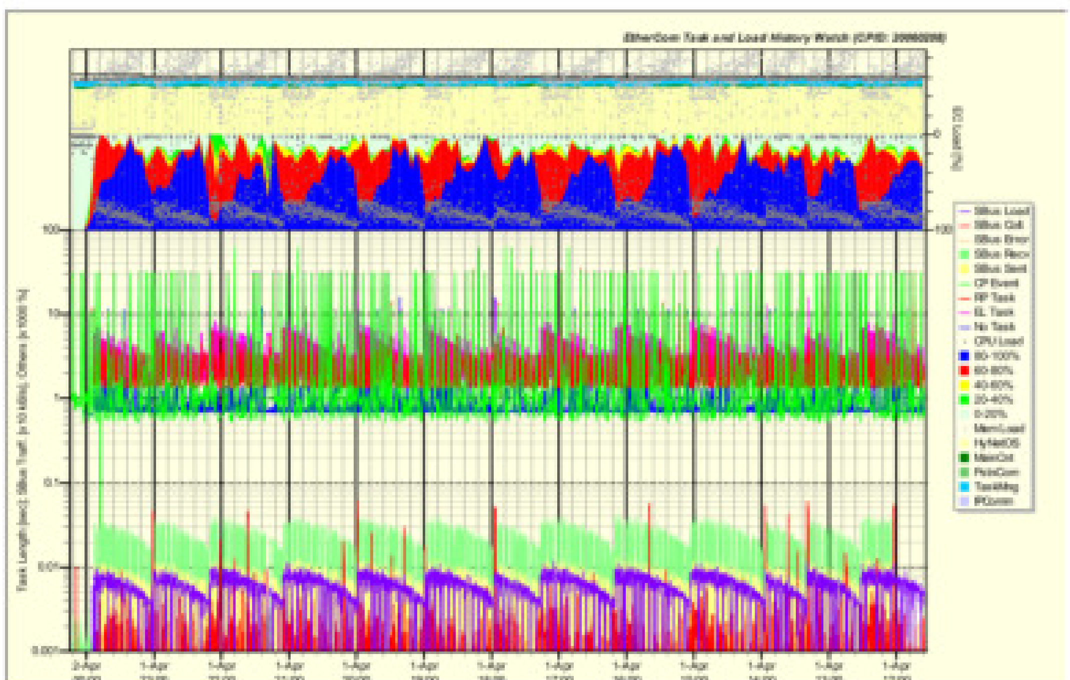


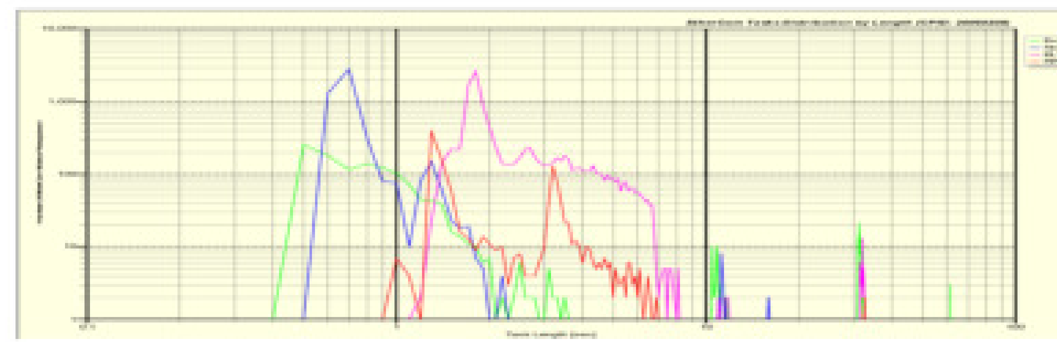


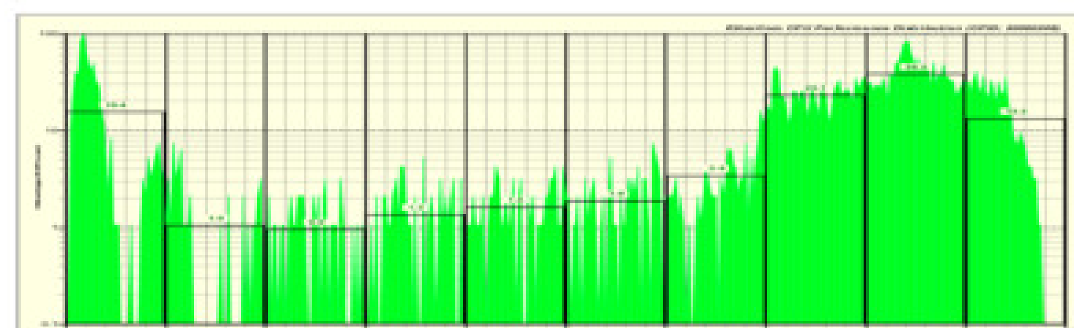

EtherCom Task and Load History Watch (CPID: 20060208)
SBus Load
SBus Coll.
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCnt
PstnCom
TaskMng
IPComm
1-Apr 08:00
1-Apr 06:00
1-Apr 04:00
1-Apr 02:00
1-Apr 00:00

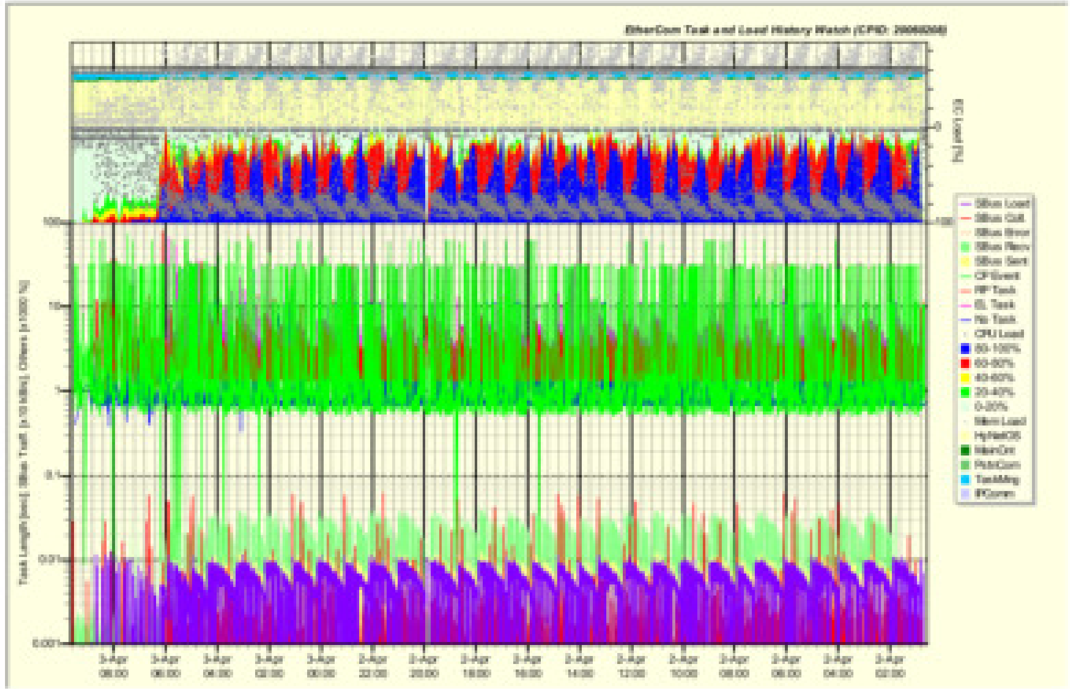

Task Length [sec]

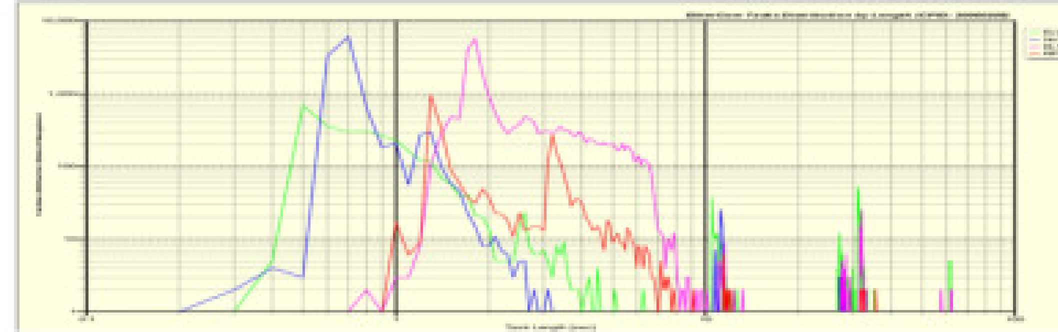

CPU Load [%]

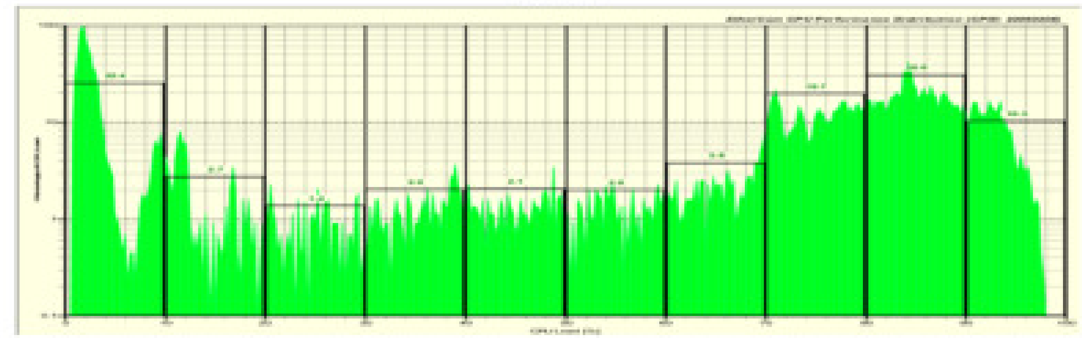

EtherCom Task and Load History Watch (CPID: 20060208)
SBus Load
SBus Coll.
SBus Error
SBus Recv
SBus Sent
CP Event
RP Task
EL Task
No Task
CPU Load
80-100%
60-80%
40-60%
20-40%
0-20%
Mem Load
HyNetOS
MainCnt
PstnCom
TaskMng
IPComm

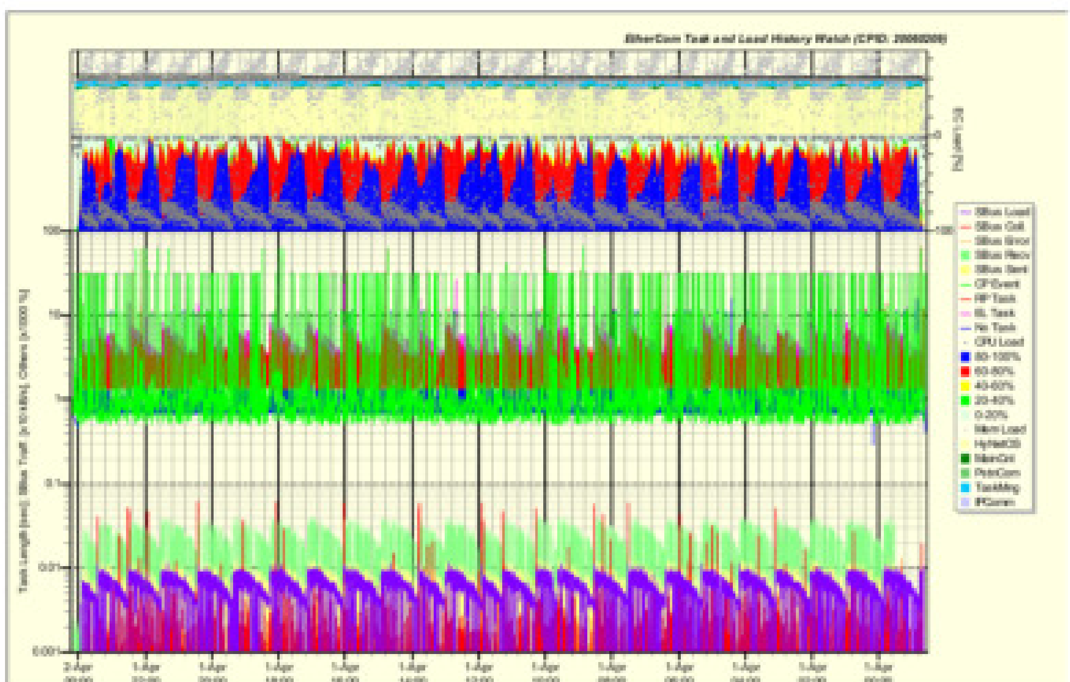


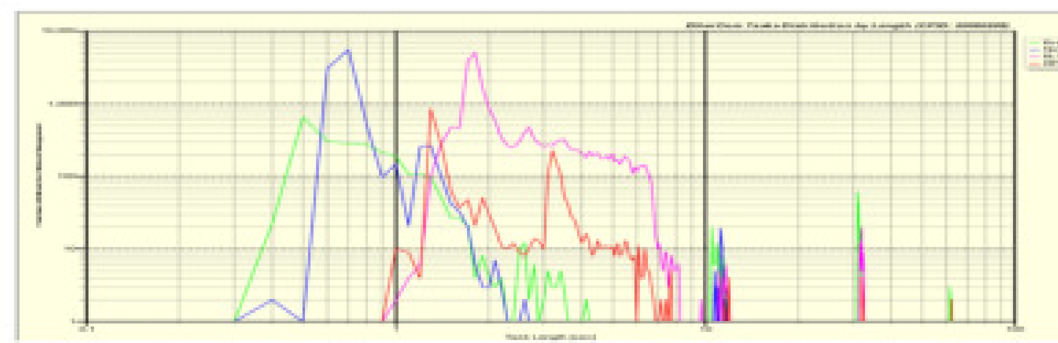


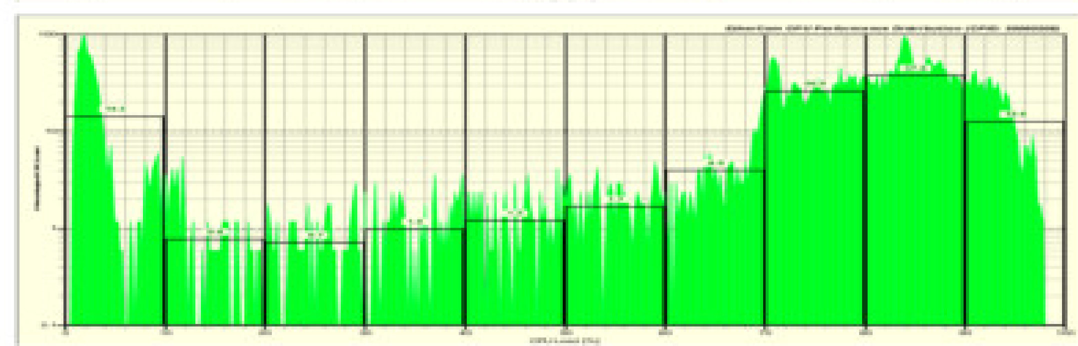


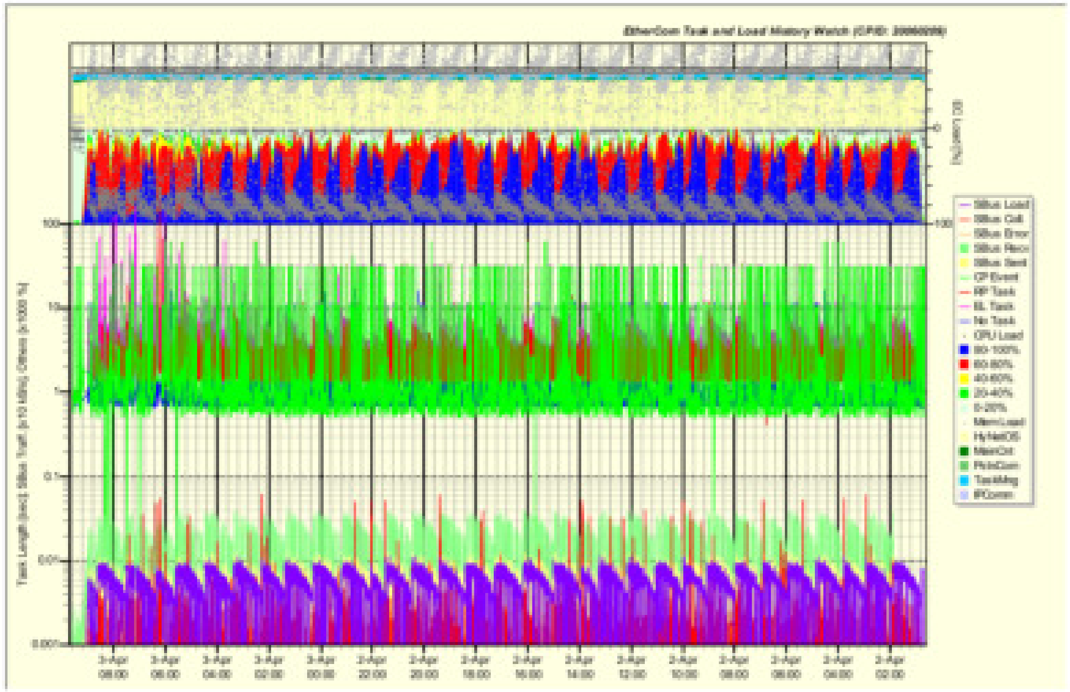




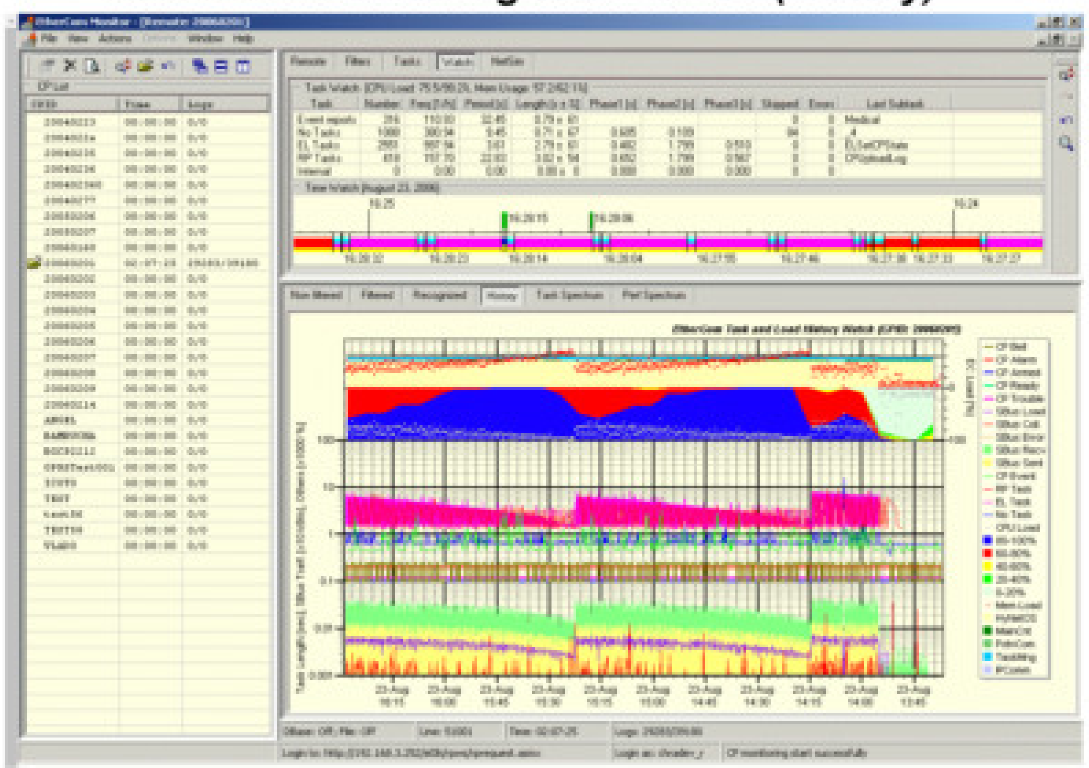


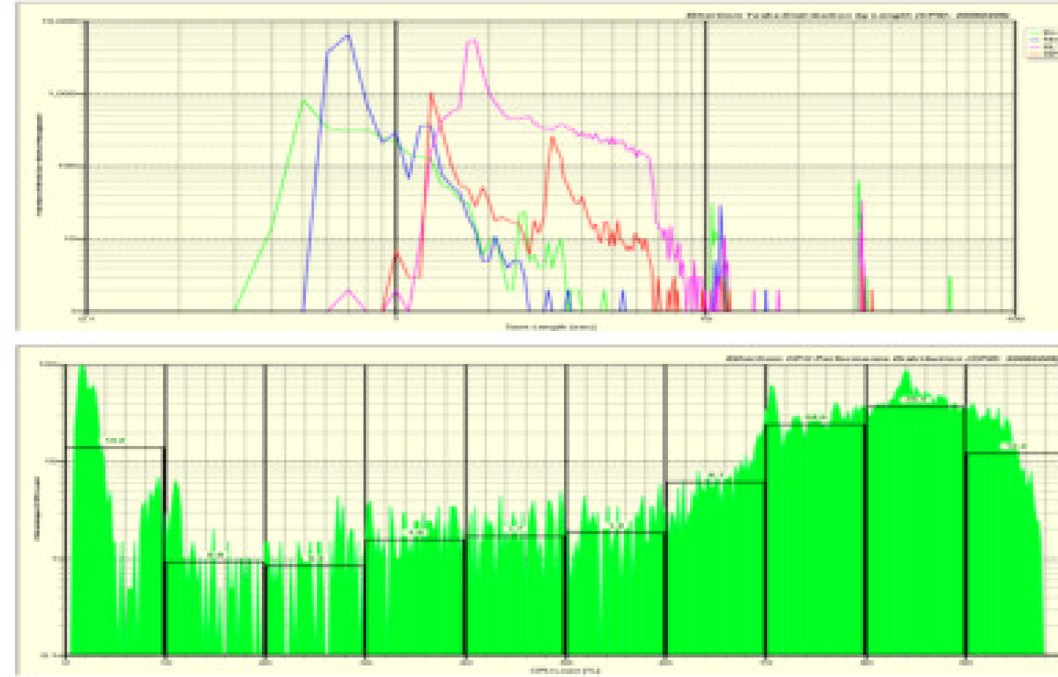




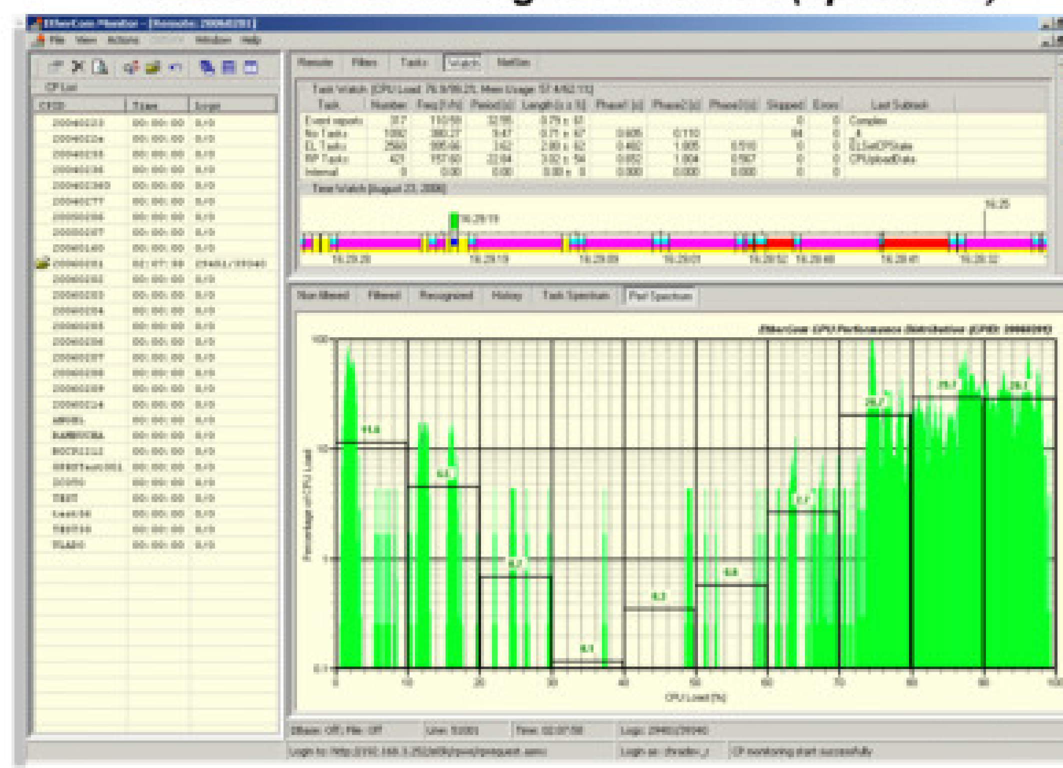


Ethercom monitoring tool RCMon (history)

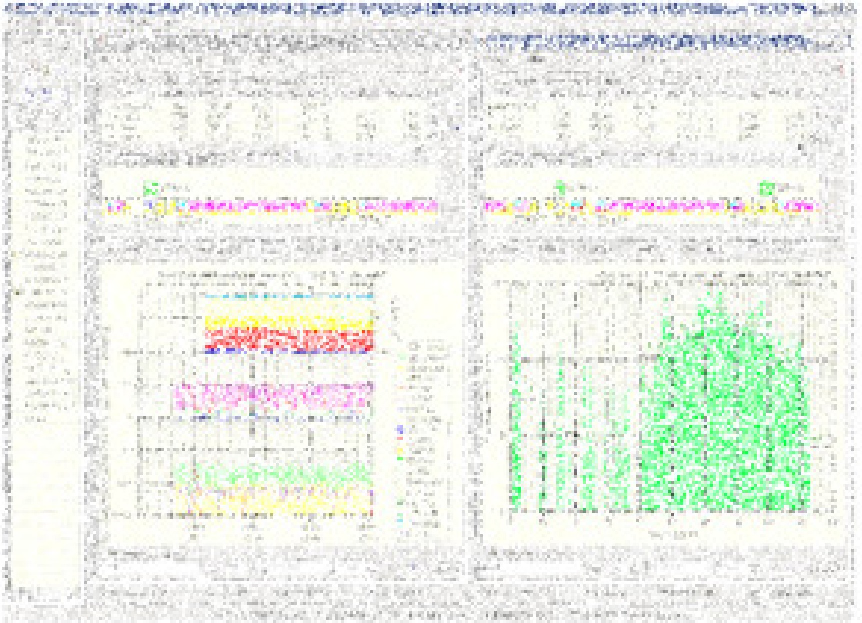


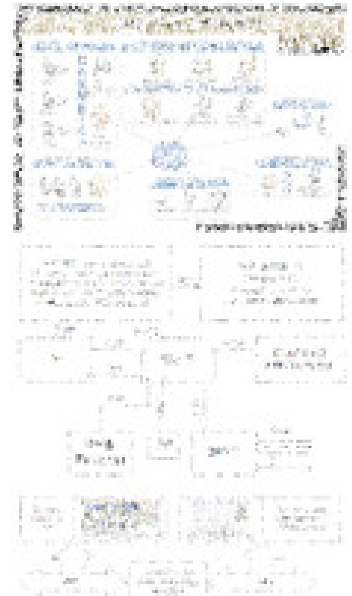


Ethercom monitoring tool RCMon (spectrum)

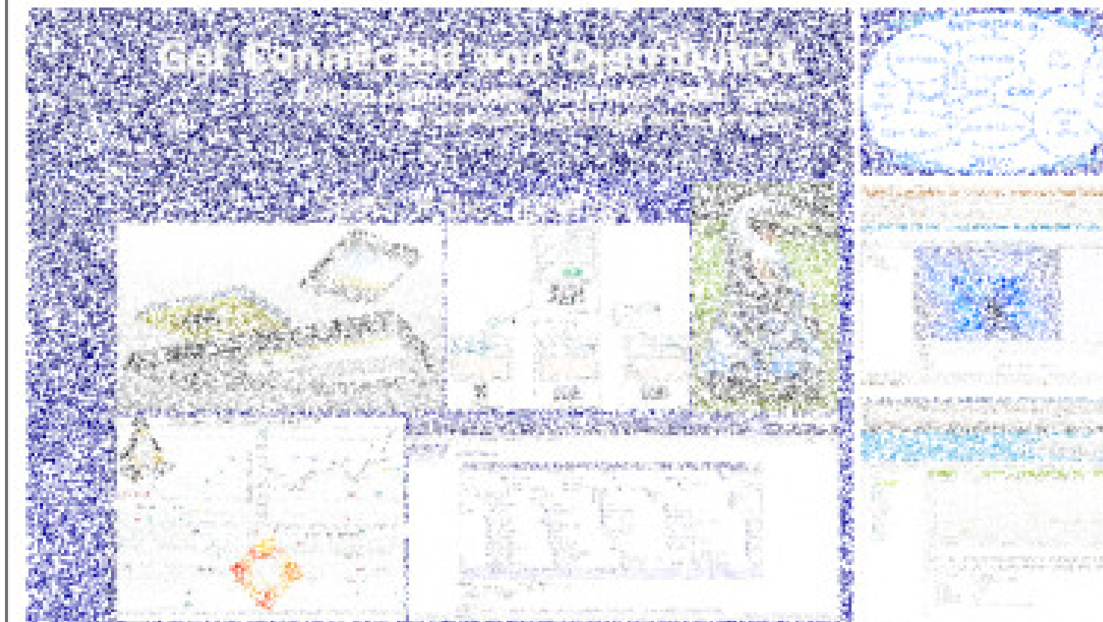


Electronics Line 3000: Internet enabled security system (2003 - 2008)
Introduced: Long term, high load, functional testing methodology

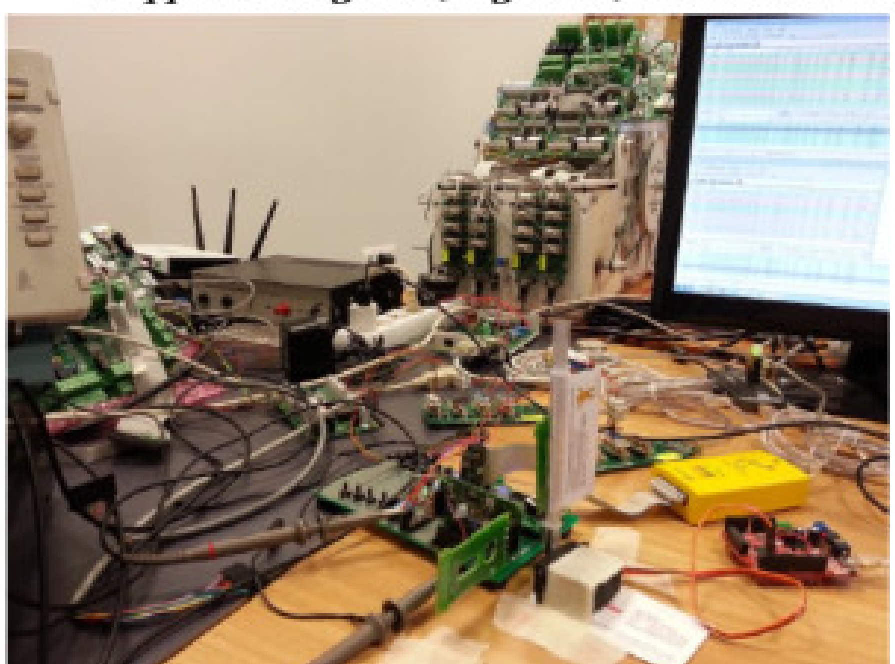


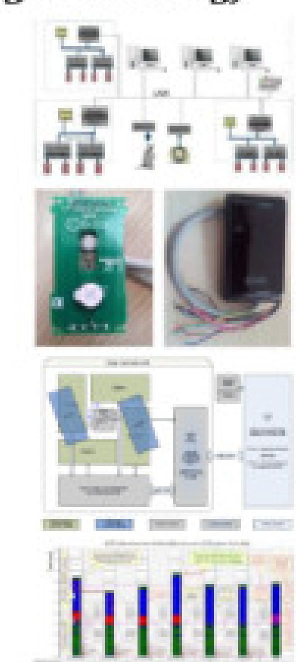


Get connected and distributed (2008 - 2013)
Student's bachelor and master degree thesis

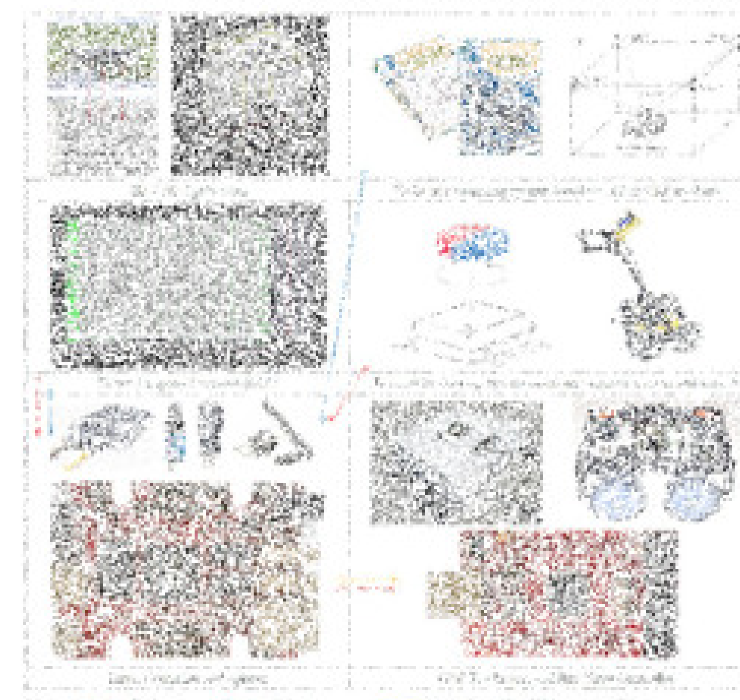


RISCO: RFID reader for Access Control System (2011 - 2014)
Reapplied: Long term, high load, functional testing methodology

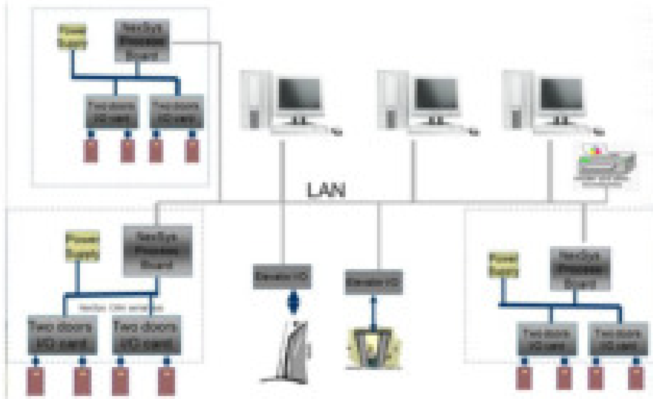




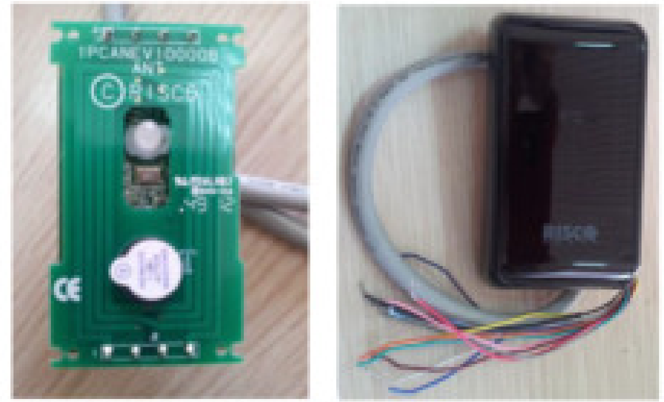
Own project: Easy Ground Penetrating Radar (2015 - 2018)



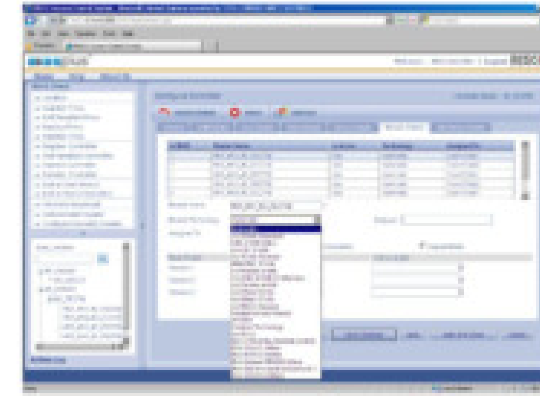
RISCO AxesPlus Access Control System, EV1 RFID Reader and Test Suit for long term, high load functional testing



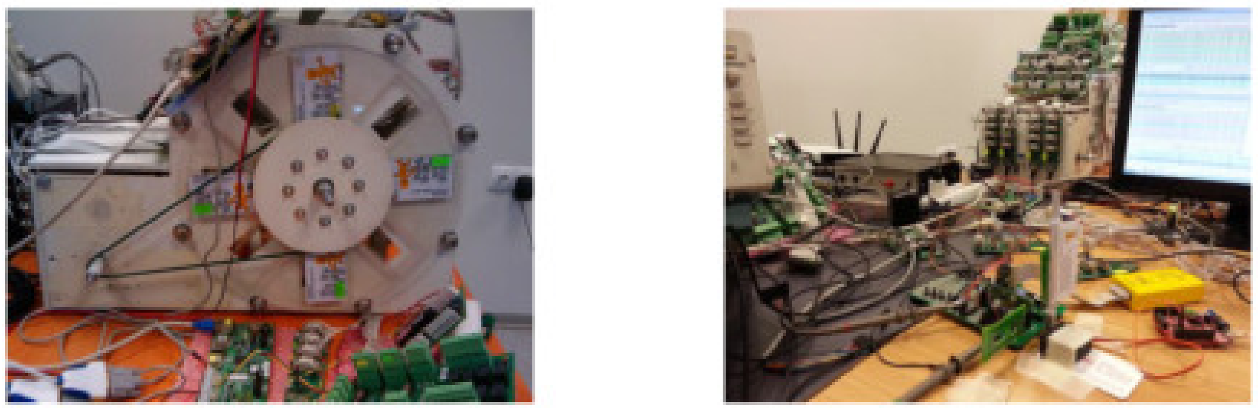
RISCO AxesPlus Architecture (3-rd party SW)



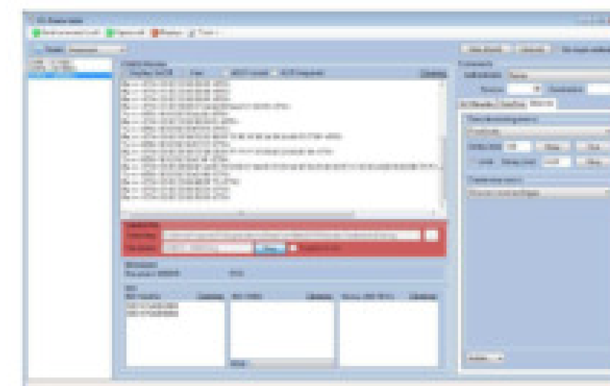
RISCO EV1 RFID Reader (FW development and tetsing)



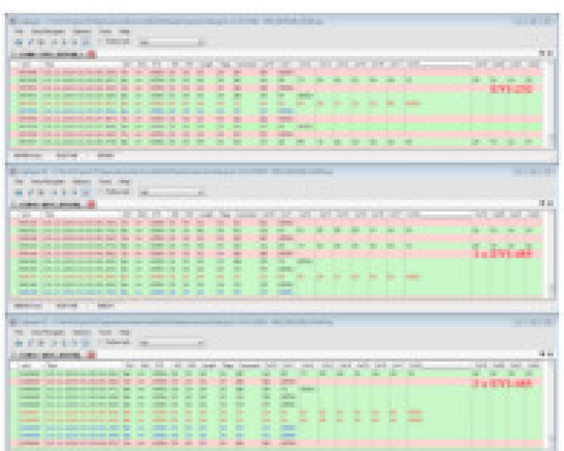
AxesPlus Web Interface (HW setup section)



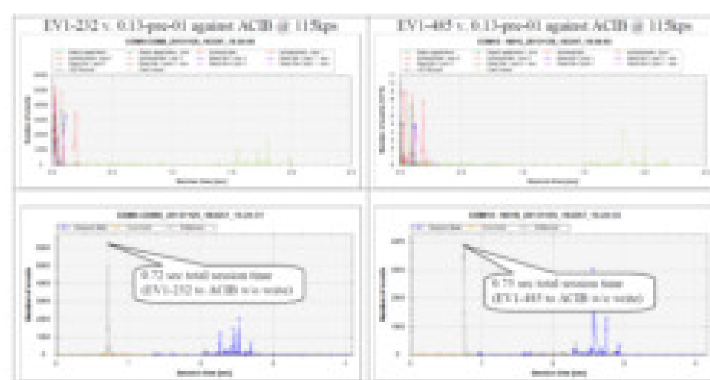
Test suit with controllable rotational speed, up to 16 EV1 Readers with RS232/RS485 interface, up to 16 RFID Desfire Cards, ACIO/HIO interface and ACCB control boards, AxesPlus Application and DB Servers on VMWare Workstation



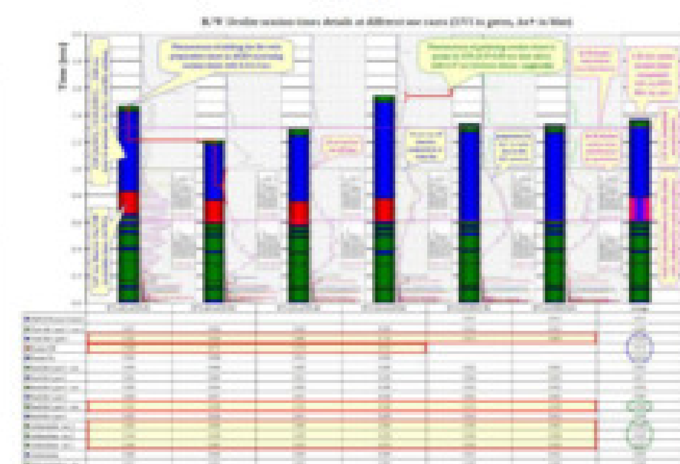
Capturing Application (via RS232/RS485 to USB adapters) written specially for testing purposes of EV1 Reader's FW



Other tools used to capture EV1 functionality



Performance tests on EV1 RS232/485 unified FW



Long term, high load functional tests on EV1 unified FW

EV1 unified FW ver. 0.13-RC2 – long term intensive in-system test results (2014-03-31)

Test conditions:

- 4 x EV1-232 and 2 x EV1-485 HW rev. C with unified FW ver. 0.13-RC2;
- Ax+ (single server) ver. 2.0.7 with ACCB, ACIB and 2 x HIO (with EV1-232 at 57kbps);
- All EV1 readers are set in factory default mode using Ax+ "R/W RISCO Desfire" technology;
- EV1 Reader Tester utility is used to capture and analyze reader behavior and test results.

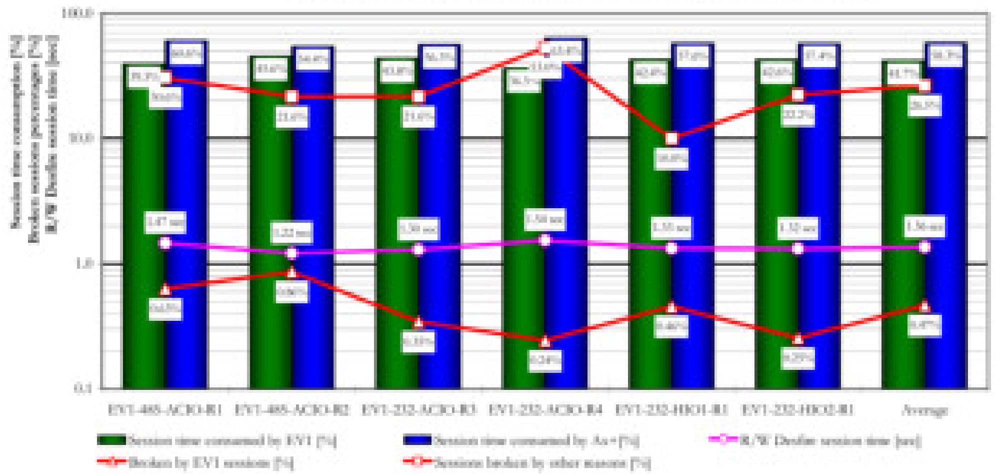
Notes:

- HW latency (RF off time of 0.045 sec mean, 0.09 sec maximal) will be added to the total session times below;
- Items below marked in red have to be analyzed with care for potential problems and possible improvements.

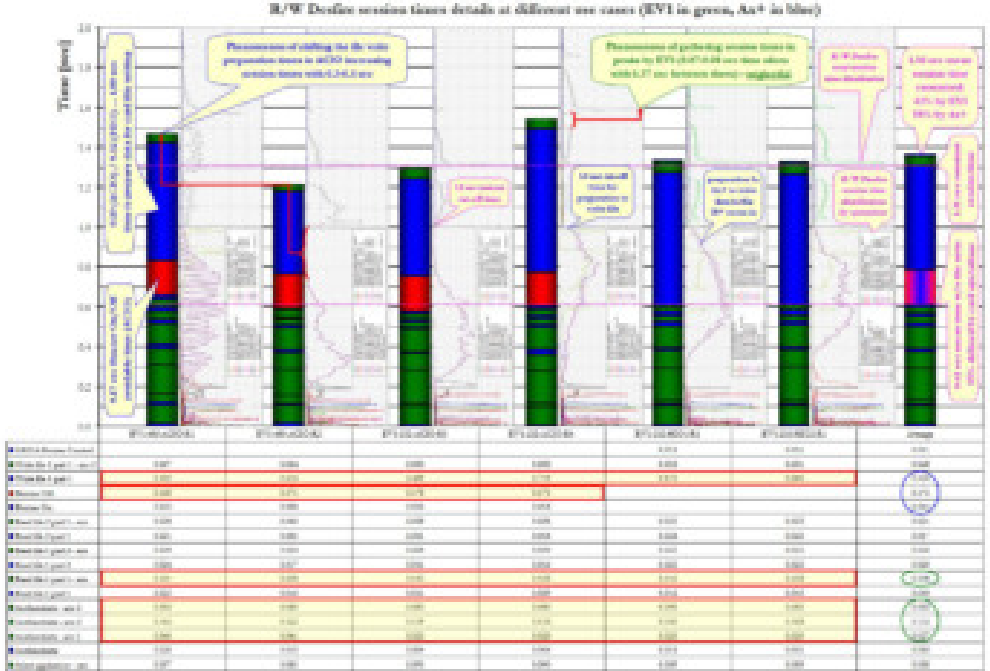
Conclusions:

- There are no unrecoverable cases observed for more than 450h on 6 EV1 readers continuously tested in Ax+ system at 8 events/min loads;
- Inaccessibility rates and lost events in EV1 unified FW ver. 0.13 and their reasons are comparable with once observed at ver. 0.12 for EV1-232;
- The phenomenon of R/W Desfire sessions breaking by EV1 was not identified as reasons but it was limited to 0.47% of sessions.
- Performance improvements (incl. HW latency limited by RF off time) do not influence on EV1 efficiency (incl. thermal);
- 1.0 sec R/W Desfire session time with EV1 Reader is reachable after Ax+ optimization while 0.5 sec – is unfeasible;
- More tests could to be done for identifying the other reasons (some unknown) for breaking 26% of R/W Desfire sessions;
- No room for more improvements in current EV1 HW/FW framework.

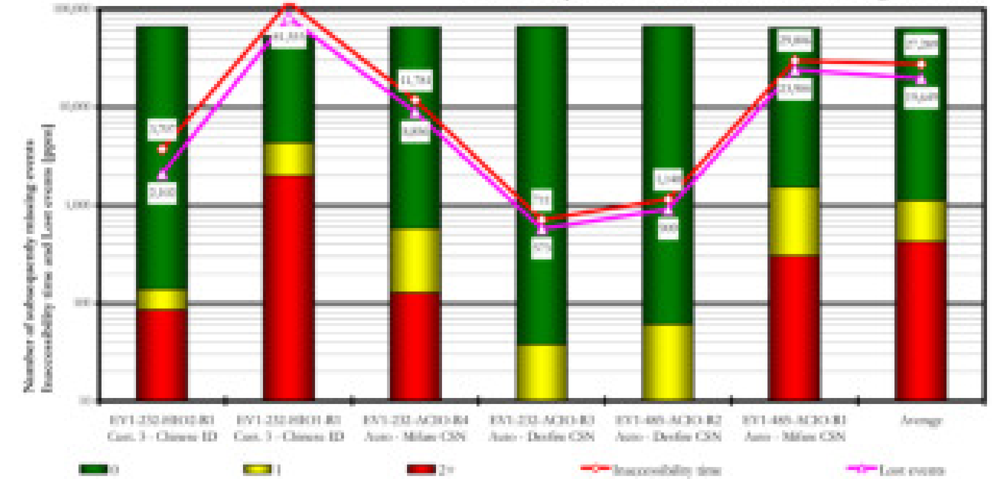
Session time, session consumption and broken sessions percentages



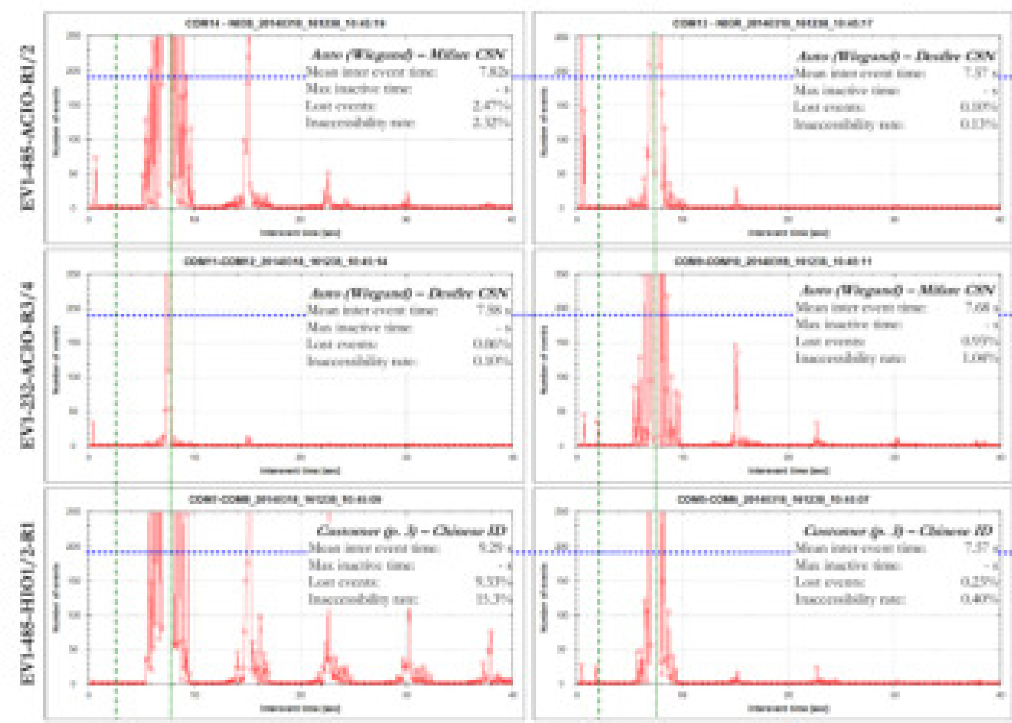
R/W Desfire session times details at different use cases (EV1 in green, Ax+ in blue)



Inaccessibility time and lost events at different technologies and cards



Details of irregular events at different technologies and cards



Legend

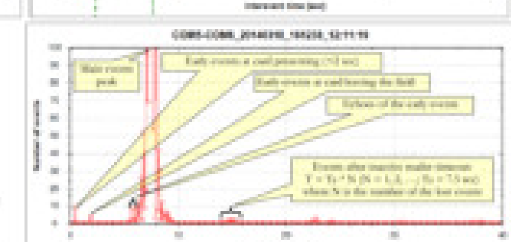
Card presence period (~7.5 sec)

Card presence time (~2.5 sec)

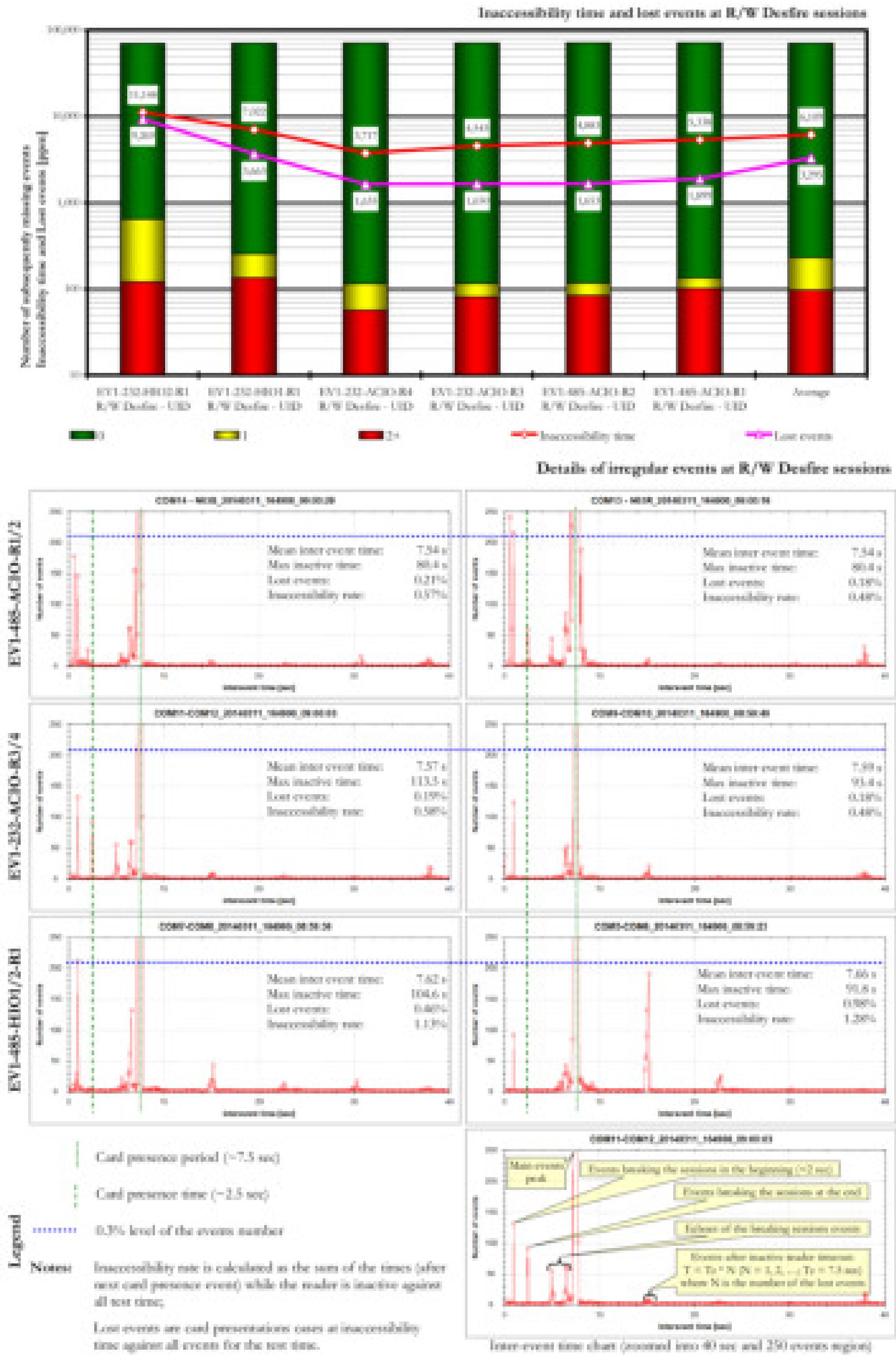
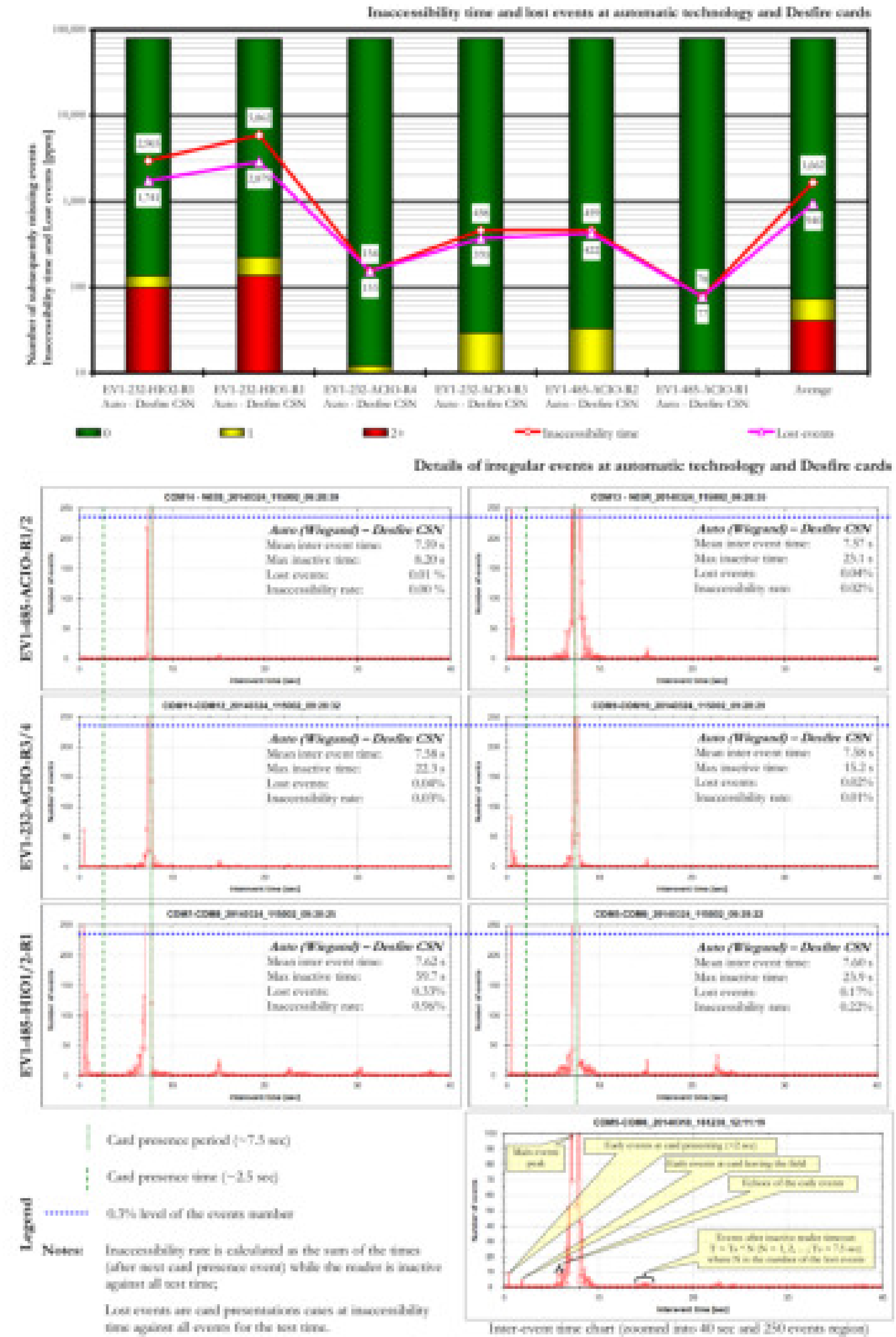
0.3% level of the events number

Notes: Inaccessibility rate is calculated as the sum of the times (after next card presence event) while the reader is inactive against all test time;

Lost events are card presentations cases at inaccessibility time against all events for the test time.



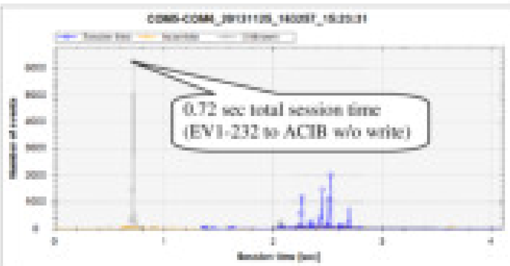
Inter-event time chart (zoomed into 40 sec and 250 events region)

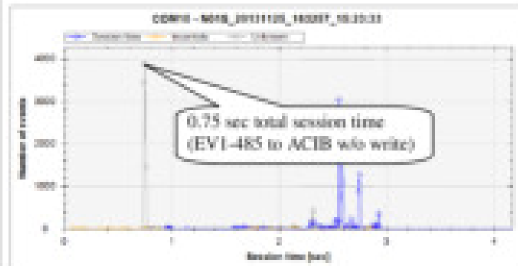

Inaccessibility time and lost events at R/W Desfire sessions
Inaccessibility time
Lost events
Details of irregular events at R/W Desfire sessions
EV1-485-ACIO-R1/2
EV1-232-ACIO-R3/4
EV1-485-HIO1/2-R1
Mean inter event time:
Max inactive time:
Lost events:
Inaccessibility rate:
Main events peak
Card presence period (~7.5 sec)
Card presence time (~2.5 sec)
0.3% level of the events number
Legend
Notes:
Inaccessibility rate is calculated as the sum of the times (after next card presence event) while the reader is inactive against all test time;
Lost events are card presentations cases at inaccessibility time against all events for the test time.
Inter-event time chart (zoomed into 40 sec and 250 events region)

Inaccessibility time and lost events at automatic technology and Desfire cards
Inaccessibility time
Lost events
Details of irregular events at automatic technology and Desfire cards
EV1-485-ACIO-R1/2
EV1-232-ACIO-R3/4
EV1-485-HIO1/2-R1
Auto (Wiegand) – Desfire CSN
Mean inter event time:
Max inactive time:
Lost events:
Inaccessibility rate:
Card presence period (~7.5 sec)
Card presence time (~2.5 sec)
0.3% level of the events number
Legend
Notes:
Inaccessibility rate is calculated as the sum of the times (after next card presence event) while the reader is inactive against all test time;
Lost events are card presentations cases at inaccessibility time against all events for the test time.
Inter-event time chart (zoomed into 40 sec and 250 events region)

Performance tests on EV1 unified FW ver. 0.13-pre-01 report (2013-11-26)

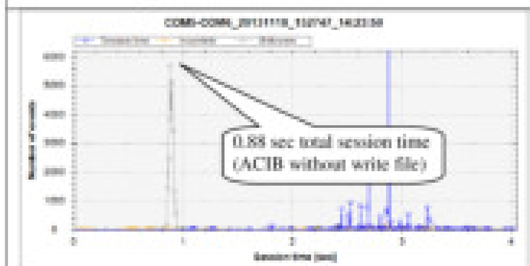
EV1-232 v. 0.13-pre-01 against ACIB @ 115kps

EV1-485 v. 0.13-pre-01 against ACIB @ 115kps

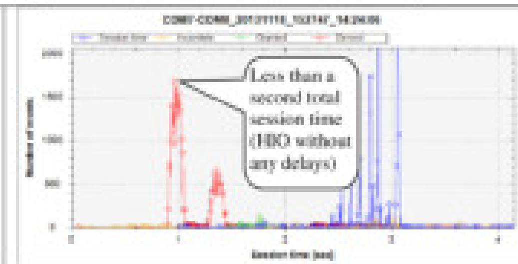




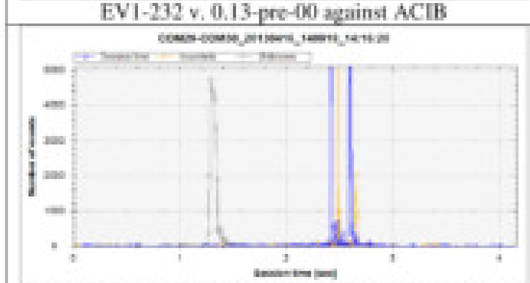
Comparison to older EV1 FW versions: ver. 0.13-pre-00 (unified) and ver. 0.12 (RS232 only)



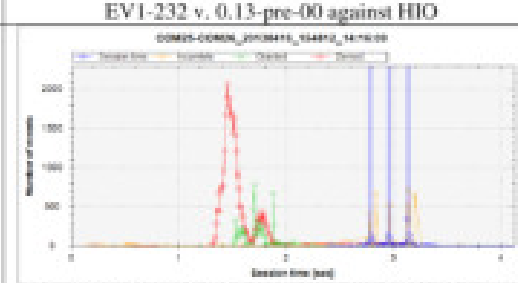
EV1-232 v. 0.13-pre-00 against ACIB



EV1-232 v. 0.13-pre-00 against HIO

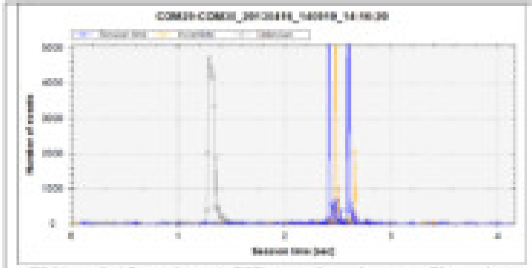


EV1 v. 0.12 against ACIB (session time no file write)

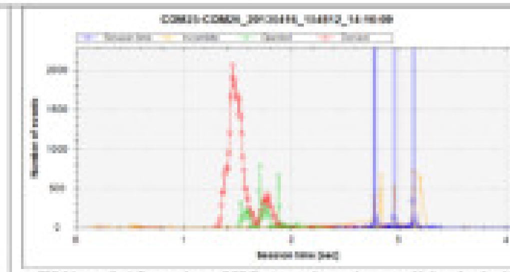


EV1 v. 0.12 against HIO (session times all included)

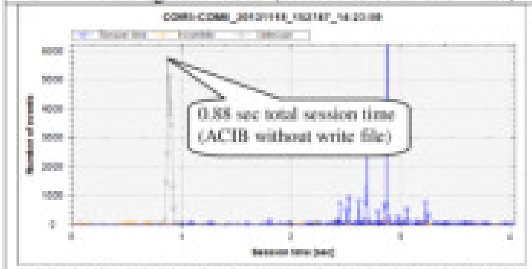
Performance tests on EV1 unified FW ver. 0.13-pre-00 report (2013-11-19)



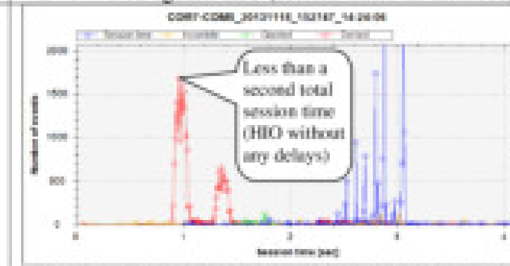
EV1 v. 0.12 against ACIB (session time no file write)



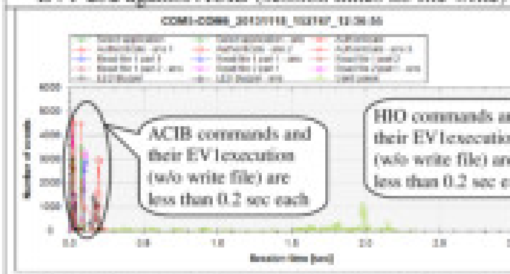
EV1 v. 0.12 against HIO (session times all included)



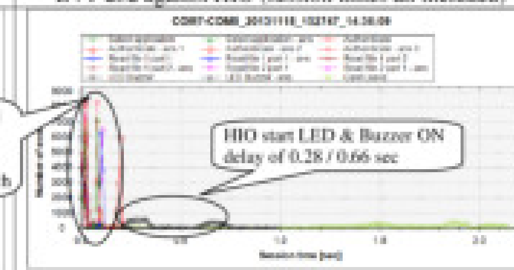
EV1-232 against ACIB (session times no file write)



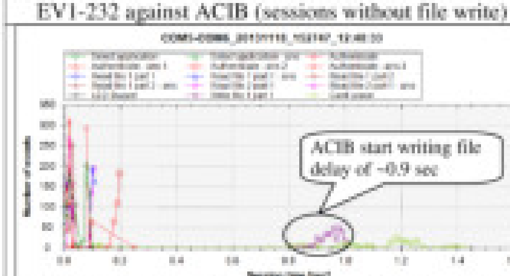
EV1-232 against HIO (session times all included)



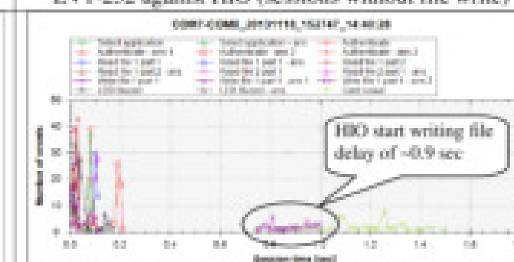
EV1-232 against ACIB (sessions without file write)



EV1-232 against HIO (sessions without file write)



EV1-232 against ACIB (sessions with file write)



EV1-232 against HIO (sessions with file write)

Notes:

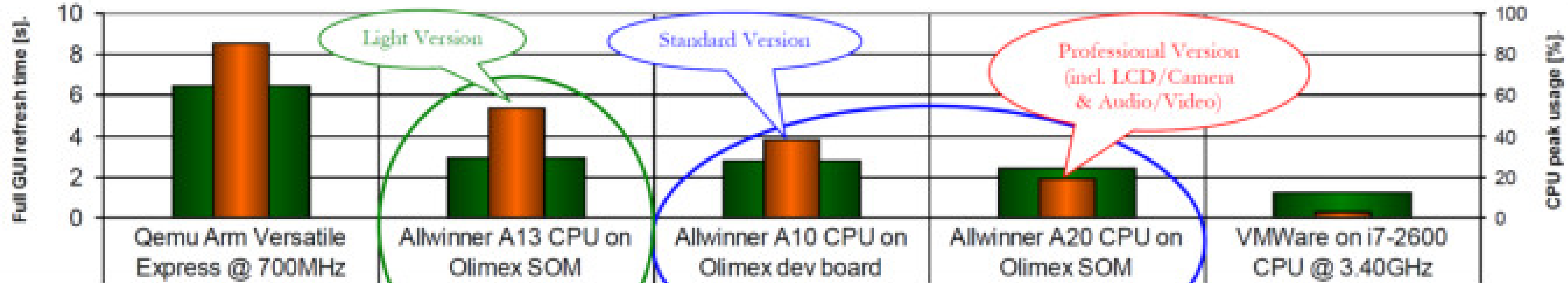
- Session times at ACIB case cannot be recognized because ACIB do not send LED ON/OFF command;
- Session times with file write at ACIB case cannot be recognized because command mismatch;
- There are some limitations of PC EV1 Reader Test utility but presented figures are representative;
- Detailed analyses are not presented in current report because of complexity.

Conclusions:

- Delay (~0.9 sec) in starting write file operation double card access time at both ACIB and HIO cases;
- There is unaccountable delay (~0.66 sec) of LED and Buzzer ON command start at HIO case;
- There is significant performance increase in EV1 unified FW ver. 0.13-pre-00 correlated to ver. 0.12;
- Biggest EV1 reader delays at file opening and authentication are card related (visible in detailed analysis).

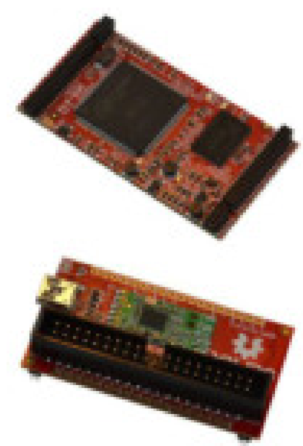
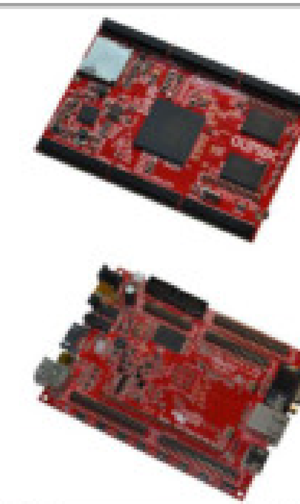
Application service and Web client interface framework for RC & RR products (preliminary performance assessment on Allwinner SoC series)

Application Service and Web Interface performance framework (preliminary performance figures)



	Qemu Arm Versatile Express @ 700MHz	Allwinner A13 CPU on Olimex SOM	Allwinner A10 CPU on Olimex dev board	Allwinner A20 CPU on Olimex SOM	VMWare on i7-2600 CPU @ 3.40GHz
Full GUI refresh time [s]	6.50	2.94	2.83	2.43	1.25
CPU peak usage [%]	85.0	53.7	38.5	18.8	3.0

Tested platforms

			Debian 7 (wheezy) with Linux kernel 3.2.51-1 Versatile Express HW (700MHz) emulated by Qemu-arm Qemu 1.3 running on Ubuntu LTS 2014-04 Ubuntu running on VMware Workstation 8 VMware Workstation running on MS Windows 7 Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz
CPU Board A13-SOM-512 Size: 61 x 33 mm; **Periphery Board** A13-SOM-WIFI-4GB Size: 61 x 33 mm **CPU** A13, **RAM** 512, **NAND** 4096, **uSD** 4096, **LAN** 100 MBit USB More: RTL8188CU 802.11n **USB Host** 0, **USB OTG** 1, **SATA** no, **LCD** yes, **HDMI** no, **Audio** no, **Cam** no	**CPU Board** A10-OLinuXino-LIME Size: 84 x 60 mm Consumption: 380/240; **USB 802.11n WiFi, 2.5" SATA HDD** SATA Performance: 246/61/68 MB/s Consumption: 510/380, 1110/870 Note: Consumption is given in booting/booted pairs according to http://hardware-libre.fr/2014/06/raspberry-vs-banana-vs-a10-olinuxino-powering-and-sata-performance/ **CPU** A10, **RAM** 512, **NAND** 0, **uSD** 4096, **LAN** 100 MBit native More: Battery charger, USB RTL8188CU 802.11n, 2.5" SATA HDD **USB Host** 2, **USB OTG** 1, **SATA** yes, **LCD** yes, **HDMI** yes, **Audio** no, **Cam** no	**CPU Board** A20-SOM-4GB Size: 81 x 56 mm; **Periphery Board** A20-SOM-EVB Size: 135 x 103 mm **CPU** A20, **RAM** 1024, **NAND** 4096, **uSD** 4096, **LAN** Gigabit native More: SD-MMC, Battery charger **USB Host** 2, **USB OTG** 1, **SATA** yes, **LCD** yes, **HDMI** yes, **Audio** yes, **Cam** 2 Mpix	Versatile Express HW (700MHz) emulated by Qemu-arm Intel(R) Core(TM) i7-2600 1-core CPU @ 3.40GHz on VMware Workstation
root@a13-OLinuXino-Micro-SOM:~# cat /proc/cpuinfo Processor : ARMv7 Processor rev 2 (v7l) BogoMIPS : 1001.88 Features : swp half thumb fastmult vfp edsp neon vfpv3 tls CPU implementer : 0x41 CPU architecture: 7	root@a10Lime:~# cat /proc/cpuinfo Processor : ARMv7 Processor rev 2 (v7l) BogoMIPS : 1001.88 Features : swp half thumb fastmult vfp edsp neon vfpv3 tls CPU implementer : 0x41 CPU architecture: 7 CPU variant : 0x3	root@a20-Lime2-SOM:~# cat /proc/cpuinfo Processor : ARMv7 Processor rev 4 (v7l) processor : 0 BogoMIPS : 2011.05 processor : 1 BogoMIPS : 2011.05 Features : swp half thumb fastmult vfp edsp neon vfpv3 tls	root@debian-armhf:~# cat /proc/cpuinfo Processor : ARMv7 Processor rev 0 (v7l) processor : 0 BogoMIPS : 680.75 Features : swp half thumb fastmult vfp edsp neon vfpv3 tls CPU implementer : 0x41

```
CPU variant     : 0x3
CPU part        : 0xc08
CPU revision    : 2
Hardware        : sun5i
Revision        : 0000
Serial          : 0000000000000000

root@a13-OLinuXino-Micro-SOM:~# uname -a
Linux a13-OLinuXino-Micro-SOM 3.4.90+ #11 PREEMPT
Thu Jun 5 16:40:24 EEST 2014 armv7l GNU/Linux

root@a13-OLinuXino-Micro-SOM:~# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 7 (wheezy)" …

root@a13-OLinuXino-Micro-SOM:~# cat /proc/meminfo
MemTotal:        395388 kB
MemFree:          74096 kB
Buffers:           9920 kB
Cached:          138804 kB
SwapCached:           0 kB
SwapCached:           0 kB
Active:          219780 kB
Inactive:         99028 kB
Active(anon):    131432 kB
Inactive(anon):     120 kB
Active(file):     88348 kB
Inactive(file):   98908 kB …
```

```
CPU part        : 0xc08
CPU revision    : 2
Hardware        : sun4i
Revision        : 0000
Serial          : 0000000000000000

root@a10Lime:~# uname -a
Linux a10Lime 3.4.90+ #3 PREEMPT Tue Jun 10 09:17:04 EEST
2014 armv7l GNU/Linux

root@a10Lime:~# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 7 (wheezy)" …

root@a10Lime:~# cat /proc/meminfo
MemTotal:        378392 kB
MemFree:          36480 kB
Buffers:           2640 kB
Cached:          184840 kB
SwapCached:           0 kB
Active:          202340 kB
Inactive:        103532 kB
Active(anon):    157320 kB
Inactive(anon):    2380 kB
Active(file):     45020 kB
Inactive(file):  101152 kB …
```

```
vfpv4 idiva idivt
CPU implementer : 0x41
CPU architecture: 7
CPU variant     : 0x0
CPU part        : 0xc07
CPU revision    : 4
Hardware        : sun7i
Revision        : 0000
Serial          : 0000000000000000

root@a20-Lime2-SOM:~# uname -a
Linux a20-Lime2-SOM 3.4.79+ #8 SMP PREEMPT Fri May 9
11:13:14 EEST 2014 armv7l GNU/Linux

root@a20-Lime2-SOM:~# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 7 (wheezy)" …

root@a20-Lime2-SOM:~# cat /proc/meminfo
MemTotal:        895876 kB
MemFree:         620036 kB
Buffers:           9464 kB
Cached:          143036 kB
SwapCached:           0 kB
SwapCached:           0 kB
Active:          170144 kB
Inactive:         85532 kB
Active(anon):    103368 kB
Inactive(anon):      92 kB
Active(file):     66776 kB
Inactive(file):   85440 kB …
```

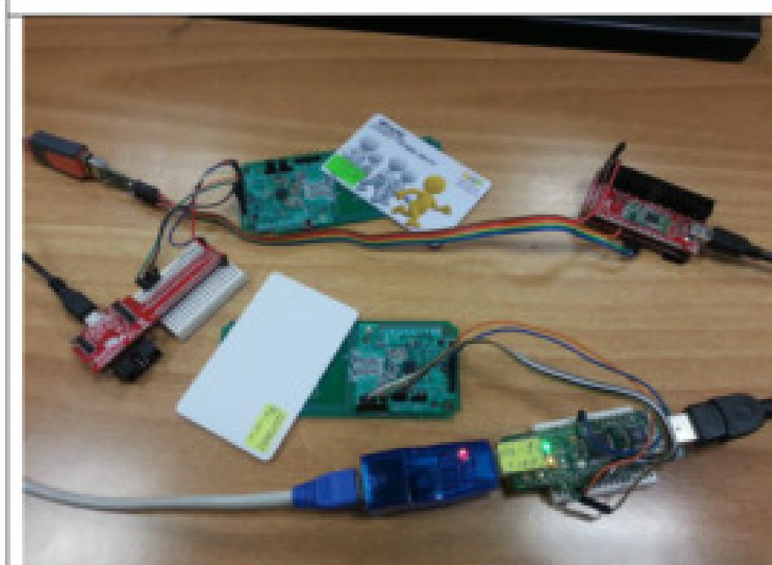
```
CPU architecture: 7
CPU variant     : 0x0
CPU part        : 0xc09
CPU revision    : 0
Hardware        : ARM-Versatile Express
Revision        : 0000
Serial          : 0000000000000000

root@debian-armhf:~# uname -a
Linux debian-armhf 3.2.0-4-vexpress #1 SMP
Debian 3.2.51-1 armv7l GNU/Linux

root@debian-armhf:~# cat /etc/os-release
PRETTY_NAME="Debian GNU/Linux 7
(wheezy)" …

root@debian-armhf:~# cat /proc/meminfo
MemTotal:        515404 kB
MemFree:         377388 kB
Buffers:           9564 kB
Cached:           55888 kB
SwapCached:           0 kB
Active:           79088 kB
Inactive:         46604 kB
Active(anon):     60280 kB
Inactive(anon):     112 kB
Active(file):     18808 kB
Inactive(file):   46492 kB …
```

Preliminary tests with PN532 board



PN532 board – SPI – atmega32u4 – USB (CDC) – PC Terminal (Serial)

Mifare Desfire EV1 card

```
Hello!
Found chip PN532
Firmware ver. 1.6
Waiting for an ISO14443A Card ...
Found an ISO14443A card
  UID Length: 7 bytes
  UID Value: 0x04 0x1C 0x65 0xD2 0xEA 0x22 0x80

Seems to be a Mifare Ultralight tag (7 byte UID)
Reading page 4
1C CE 00 AA  .?.?
```

Mifare Classic card

R/W on Mifare Classic card

```
Found an ISO14443A card
  UID Length: 4 bytes
  UID Value: 0xD6 0x8A 0x75 0xC1

Seems to be a Mifare Classic card (4 byte UID)
Trying to read 16 sectors:
sector 0: D6 8A 75 C1 E8 18 02 00 8F 23 25 1A 23 25 1A 01  ......#%..#%..
sector 1: 61 64 61 66 72 75 69 74 2E 63 6F 6D 00 00 00 00  adafruit.com....
sector 2: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
...
sector E: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
15 sectors were read successfully!
```

PN532 board – serial – A13-SOM – LAN – PC Terminal (SSH)

```
Linux a13-OLinuXino-Micro-SOM 3.4.90+ #11 PREEMPT Thu Jun 5 16:40:24 EEST 2014 armv7l
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent

root@a13-OLinuXino-Micro-SOM:~# nfc-list
nfc-list uses libnfc 1.7.0
NFC device: pn532_uart:/dev/ttyUSB0 opened
```

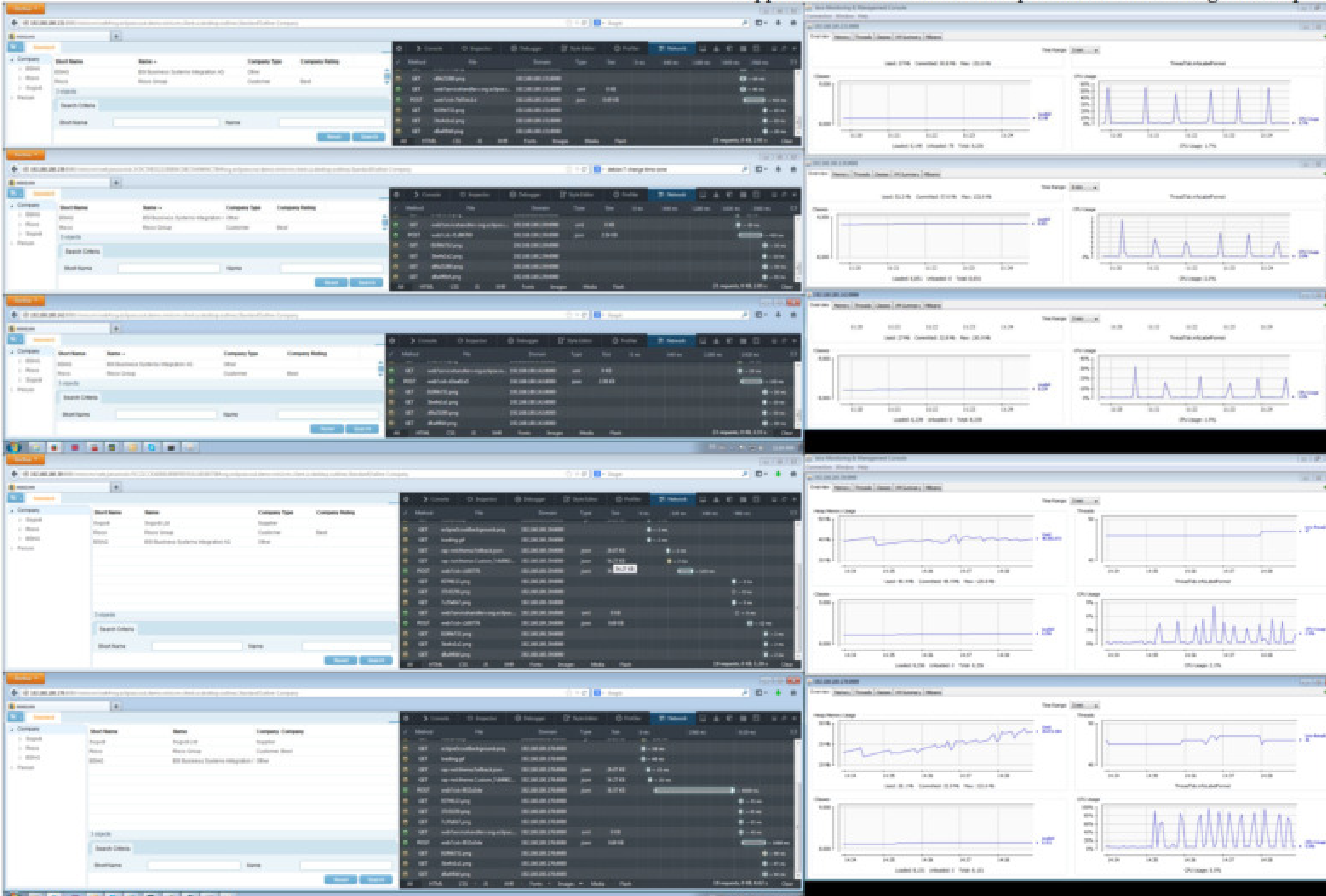
Mifare Desfire EV1 card

```
root@a13-OLinuXino-Micro-SOM:~# nfc-poll
nfc-poll uses libnfc 1.7.0
NFC reader: pn532_uart:/dev/ttyUSB0 opened
NFC device will poll during 30000 ms (20 pollings of 300 ms for 5 modulations)
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 03  44
       UID (NFCID1): 04  34  86  5a  47  21  80
      SAK (SEL_RES): 20
                ATS: 75  77  81  02  80
```

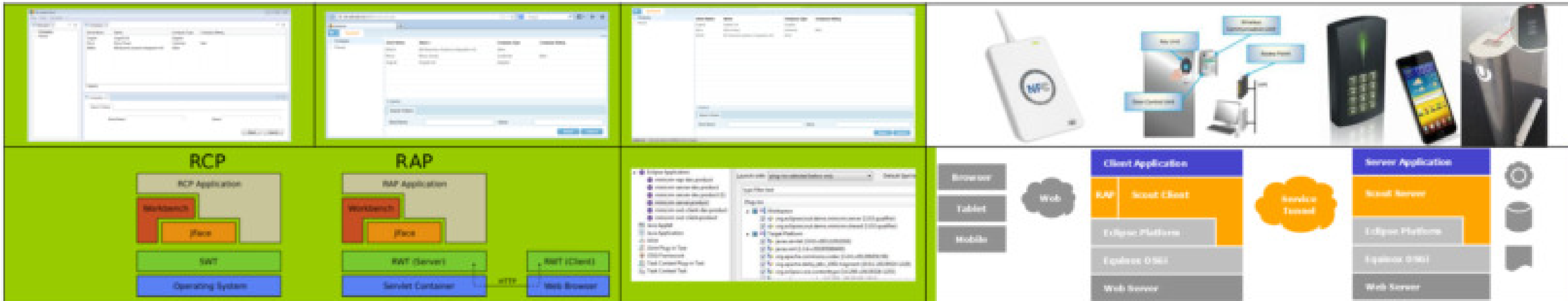
Mifare Classic card

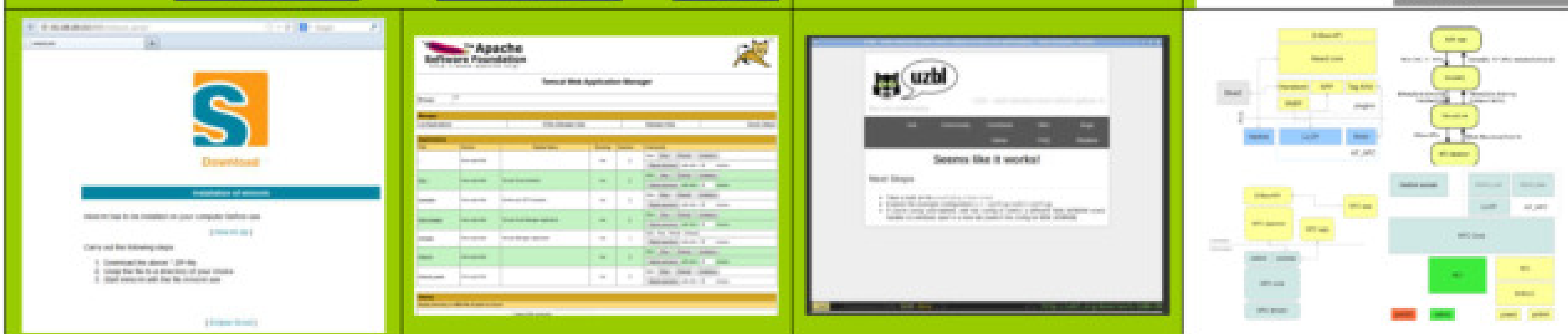
```
root@a13-OLinuXino-Micro-SOM:~# nfc-poll
nfc-poll uses libnfc 1.7.0
NFC reader: pn532_uart:/dev/ttyUSB0 opened
NFC device will poll during 30000 ms (20 pollings of 300 ms for 5 modulations)
ISO/IEC 14443A (106 kbps) target:
    ATQA (SENS_RES): 00  04
       UID (NFCID1): d6  8a  75  c1
      SAK (SEL_RES): 18
```

Application's Web Interface and performance measuring staff snapshot



Application service, Web clients interface and NFC framework (preliminary components overview)

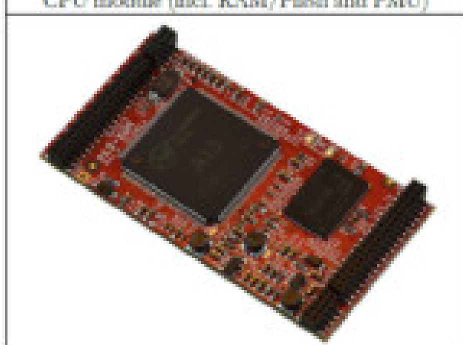




Open source NFC stack features matrix

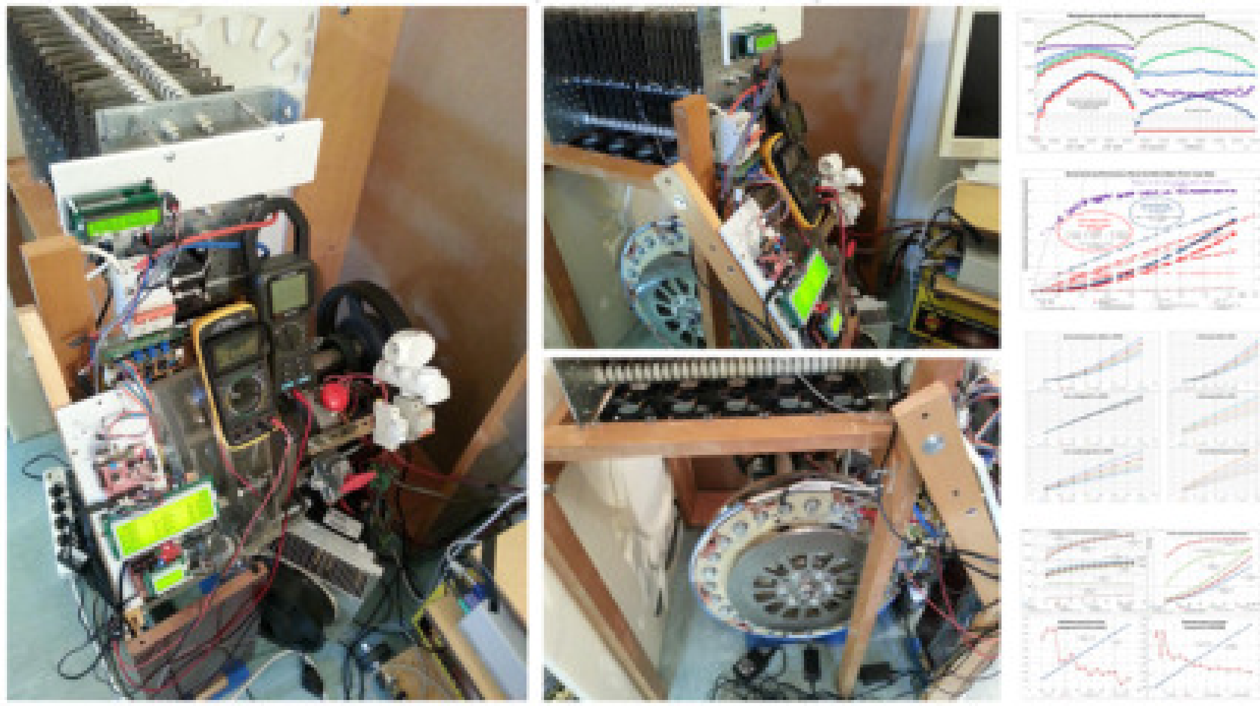
	Host Interfaces	Tag R/W	LLCP	Handover	Card Emulation
Linux-nfc	HCI, NCI, USB	Yes	NPP, SNEP, Handover, PHDC	Bluetooth, WiFi	Yes
Android	HCI, NCI	Yes	NPP, SNEP,Handover	Bluetooth	Yes
Inside Secure	HCI	Yes	SNEP, Handover	Bluetooth, WiFi	Yes
libnfc	USB, UART	Yes	No	No	Yes
nfcpy	USB	Yes	SNEP, Handover, PHDC	Bluetooth, WiFi	No

Application Service	Unified Web, Desktop, Mobile clients	lightweight webkit browser		
Eclipse scout framework for service oriented business applications		Linux X without window manager	OSGI based advertising, registration, management, access etc.	
Java, OSGI, Eclipse, Tomcat, Derby etc. staff			Eclipse Communication Framework	
Debian/Ubuntu/Yocto distribution/build				RTOS, libraries, middleware and RR FW
Uboot, Linux kernel, drivers and modules	Standard libraries and packages	Additional libraries and packages	NFC libraries or Linux NFC middleware	Built-in FW and communication protocol
CPU module (incl. RAM/Flash and PMU)	Standard interfaces and built-in peripherals	Extended/external peripherals and modules	NFC module and antenna (without MCU)	NFC module and antenna (with MCU)

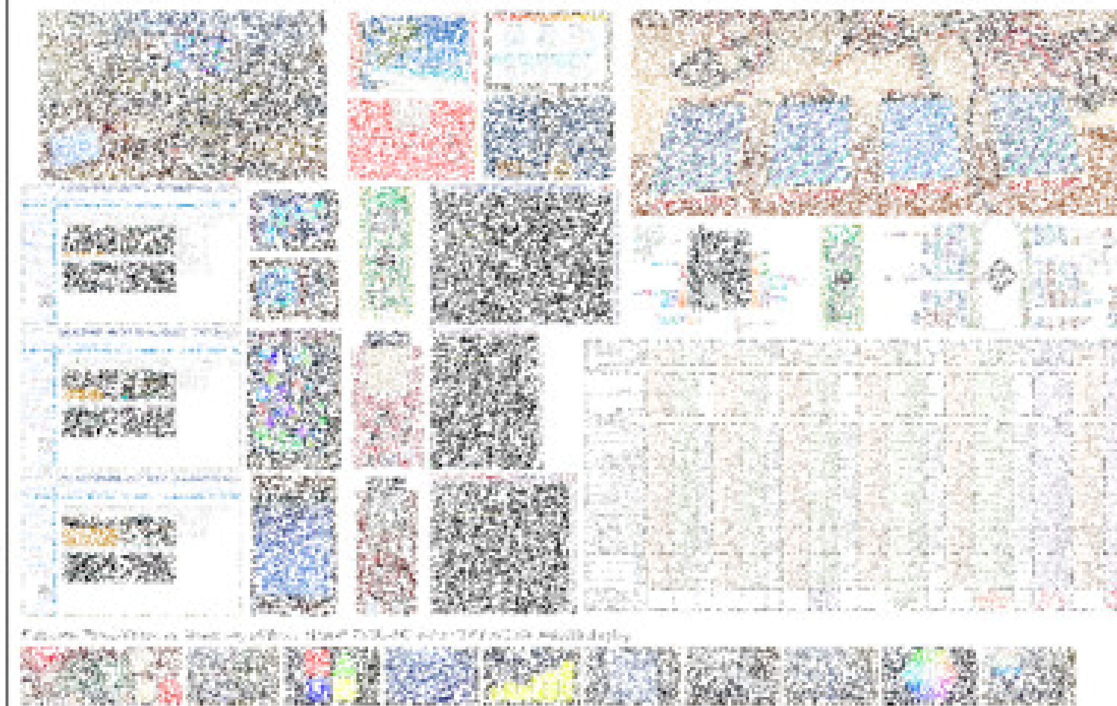




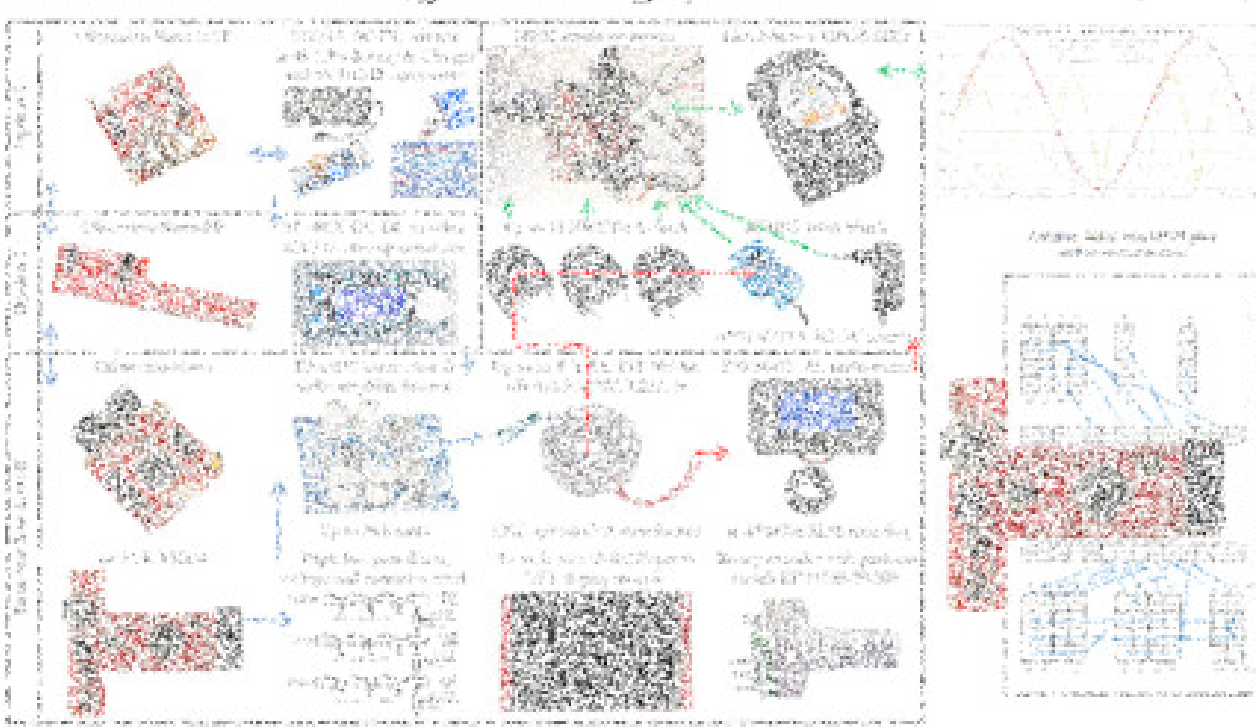
5kW Wind Generator, test bench and measured data (2013)



Unified Multicore Low Power IoT Platform (2023)



Multichannel Energy Metering system and test bench (2023)



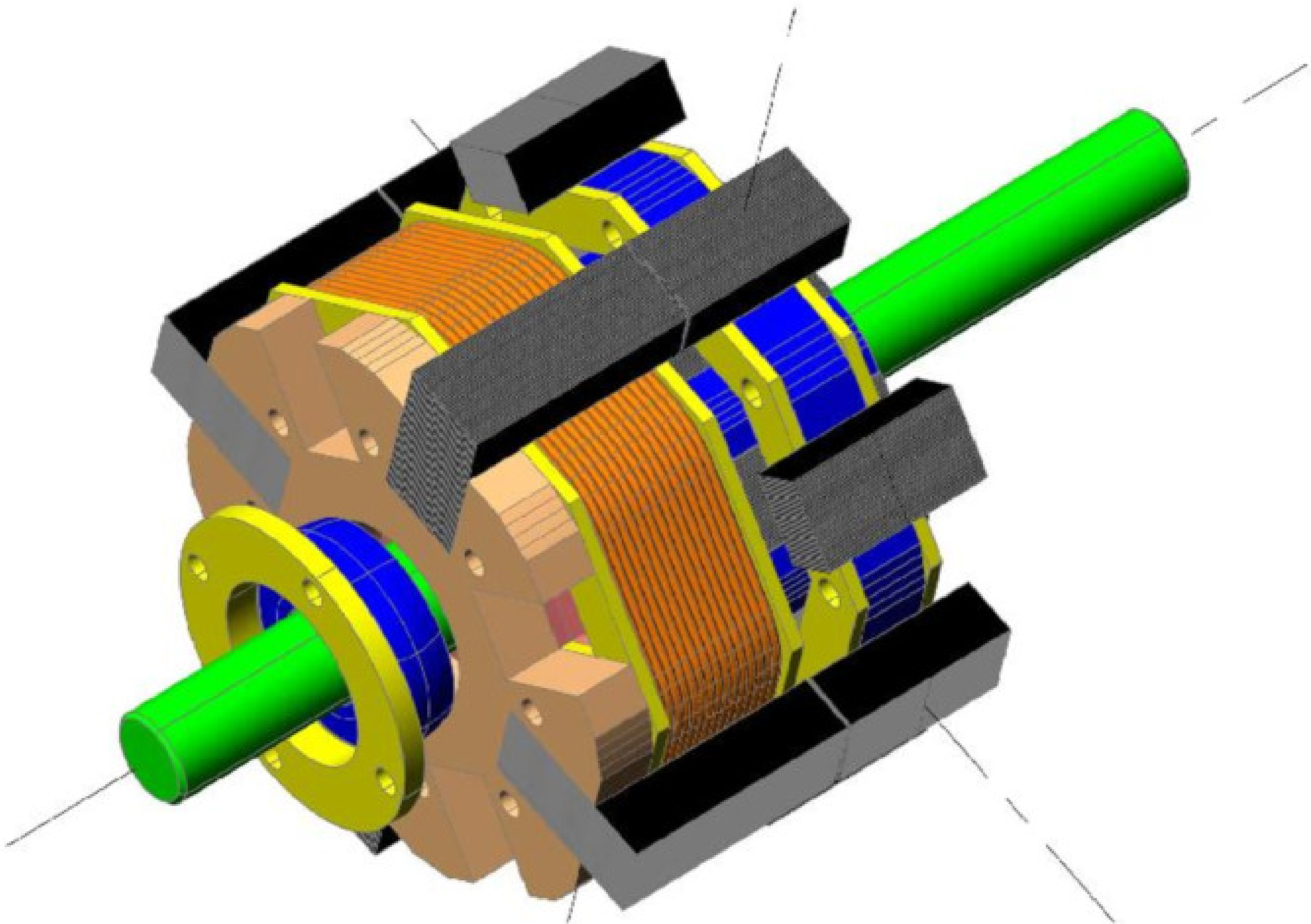
Adroid - the open S.T.E.A.M. robot platform (2024)

Eolyc’s Wind Generator

3 phase Axial Flux Ironless Generator
with NdFeB Permanent Magnets
R&D Process and Chronology

Startup with advanced ideas

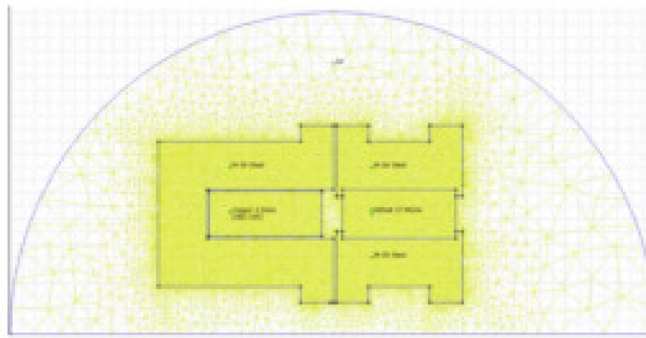
- Concentrating multiphase transverse flux topology with permanent magnet excitation



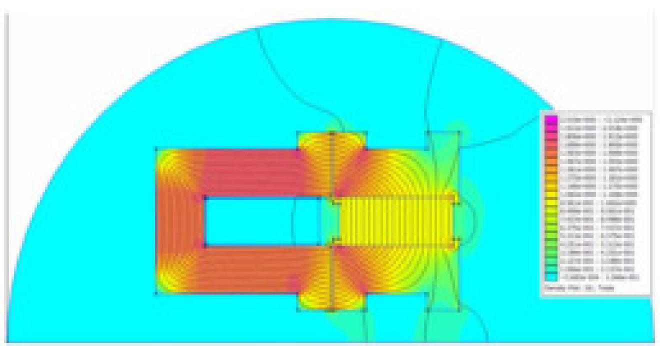
Concentrating transverse flux topology

- Simulation results for elementary generator

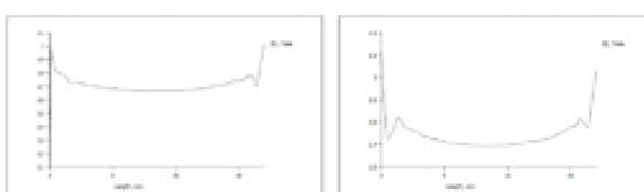
PM Axial Flux Concentrating Direct Drive Wind Generator



Planar model prepared in FEMM with geometry imported from AutoCAD .dxf file



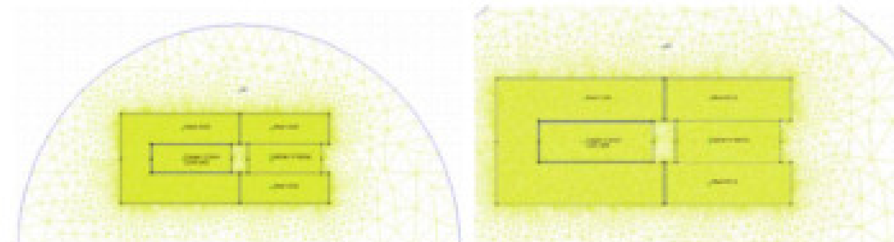
2D Magneto-static analysis result solved by Finite Element Method Magnetic package



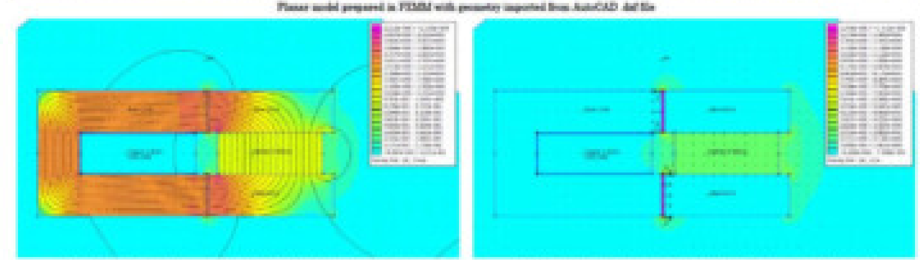
Flux Density along rotor's direction with maximal flux values

Flux Density along stator's direction with maximal flux values

PM Axial Flux Concentrating Direct Drive Wind Generator

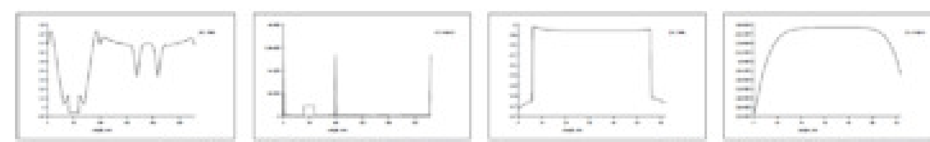


Planar model prepared in FEMM with geometry imported from AutoCAD .dxf file



2D Magneto-static analysis result solved by FEMM package (Flux Density)

2D Magneto-static analysis result solved by FEMM package (Field Intensity)

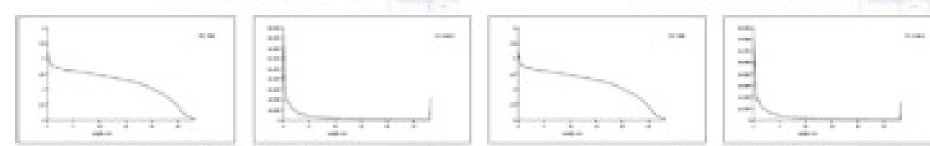


Density along the middle of cores and gaps

Density along the PM's axial direction

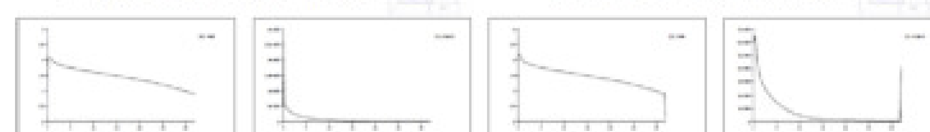
Density along the air gap's axial direction

Density along the air gap's radial direction



Density along bottom stator's direction with maximal flux values

Density along top stator's direction with maximal flux values

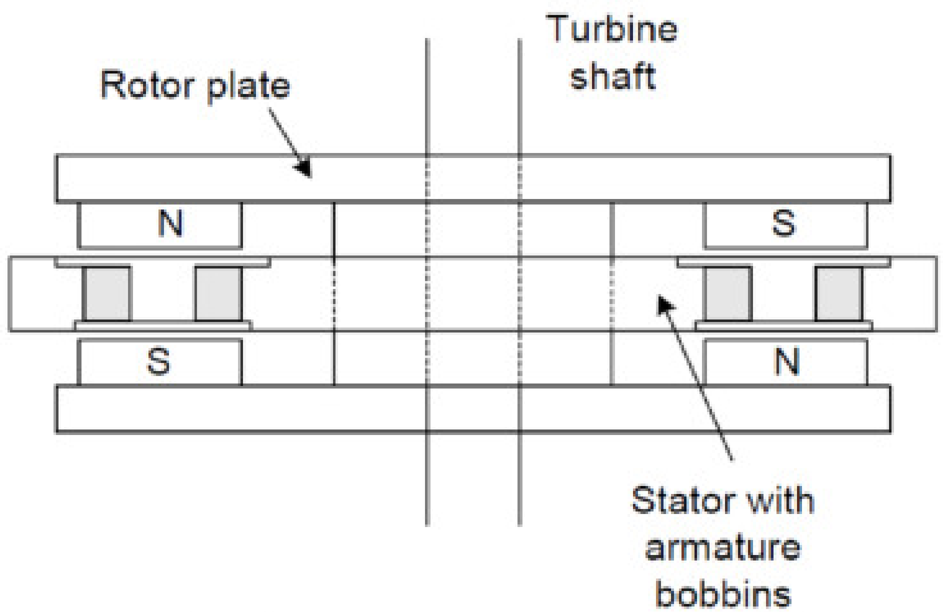


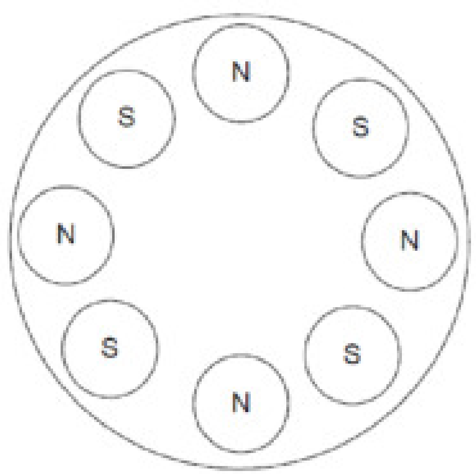
Density along bottom rotor's direction with maximal flux values

Density along top rotor's direction with maximal flux values

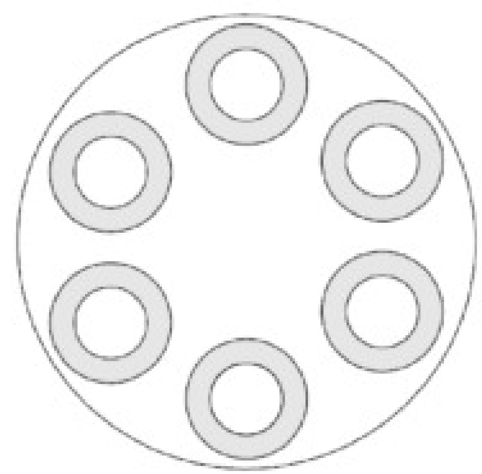
Following traditional ideas

- Multiphase Axial Flux Ironless Generator with permanent magnet excitation





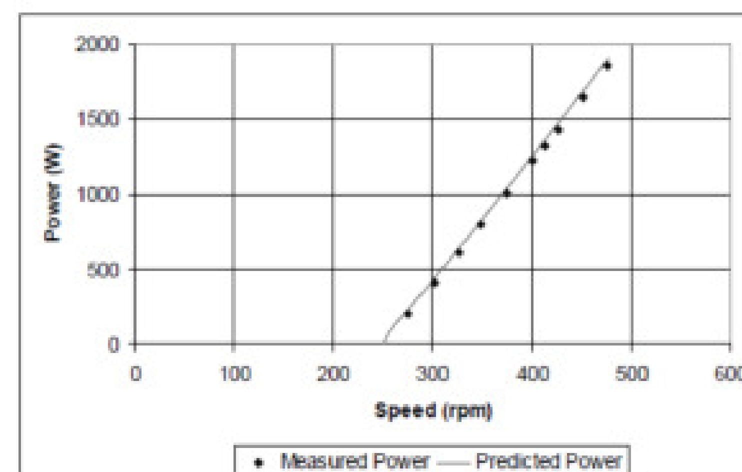
Rotor disc



Stator disc

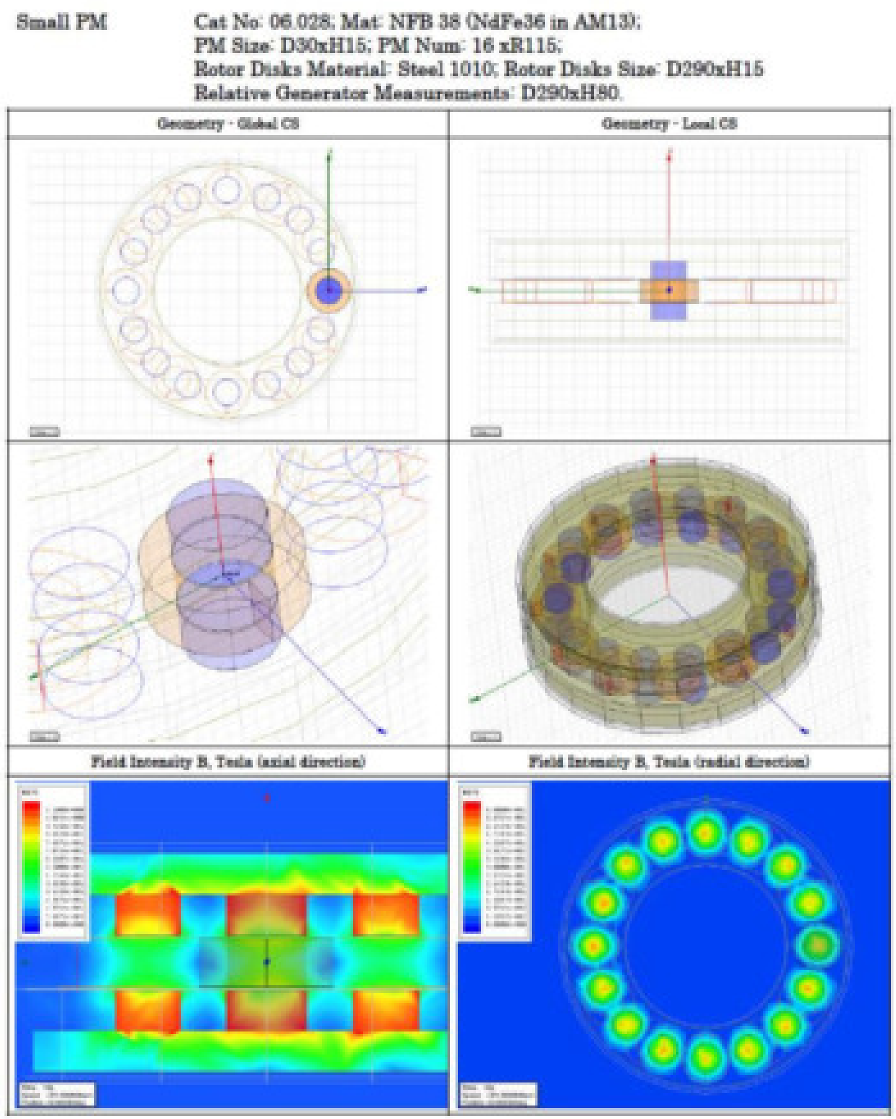
		1kW	2.5kW
Rated power	W	1000	2500
Rated speed	rpm	300	250
Rated frequency	Hz	40	33.3
Rated EMF (per coil)	V	33.6	205
Number of phases		3	3
Number of pole pairs		8	8
Number of armature coils		12	12
Generator diameter	mm	462	590
Generator length	mm	55	60

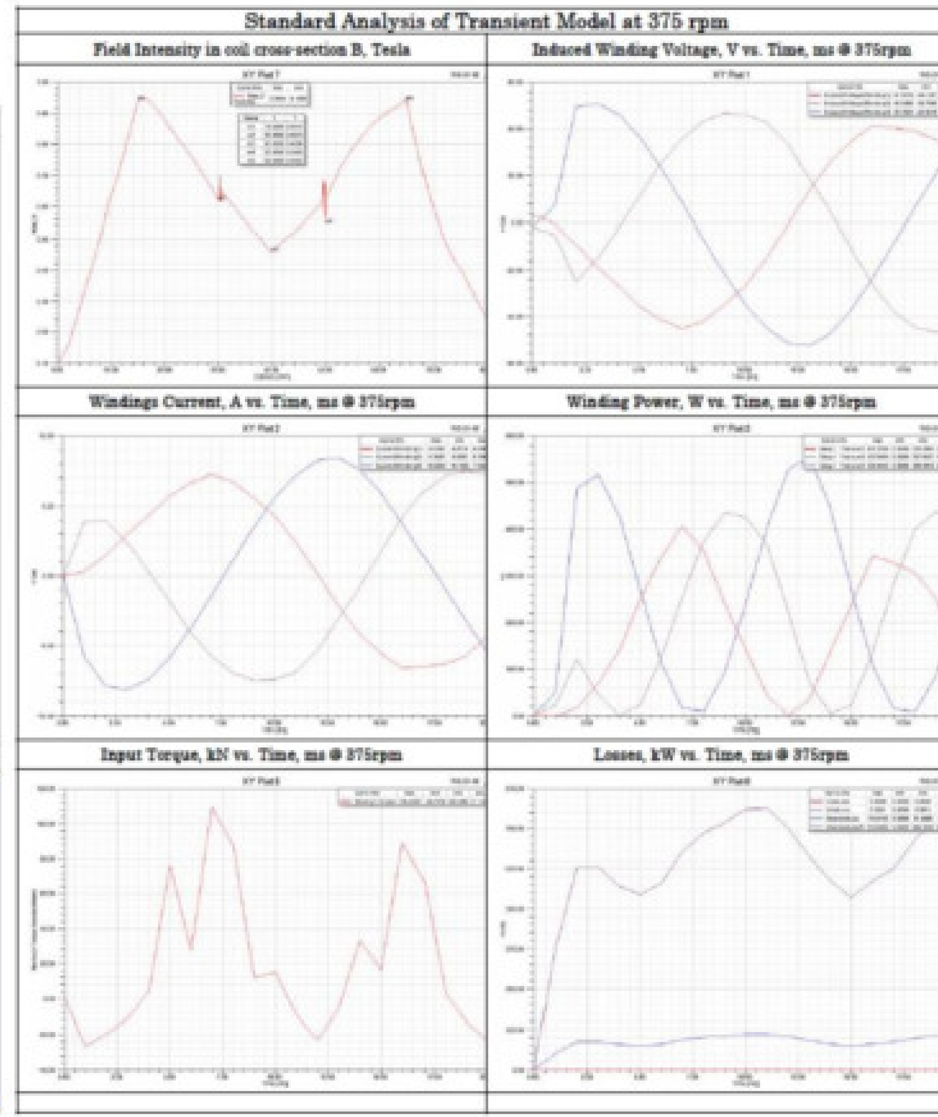
	1 kW		2.5 kW	
	Measured	Predicted	Measured	Predicted
Coil inductance (mH)	4.67	4.59	67	81
Coil resistance (ohms)	1.02	0.97	12.9	11.1
V/100rpm/coil	11.03	11.2	86.0	82.1



Axial Flux Ironless Generator

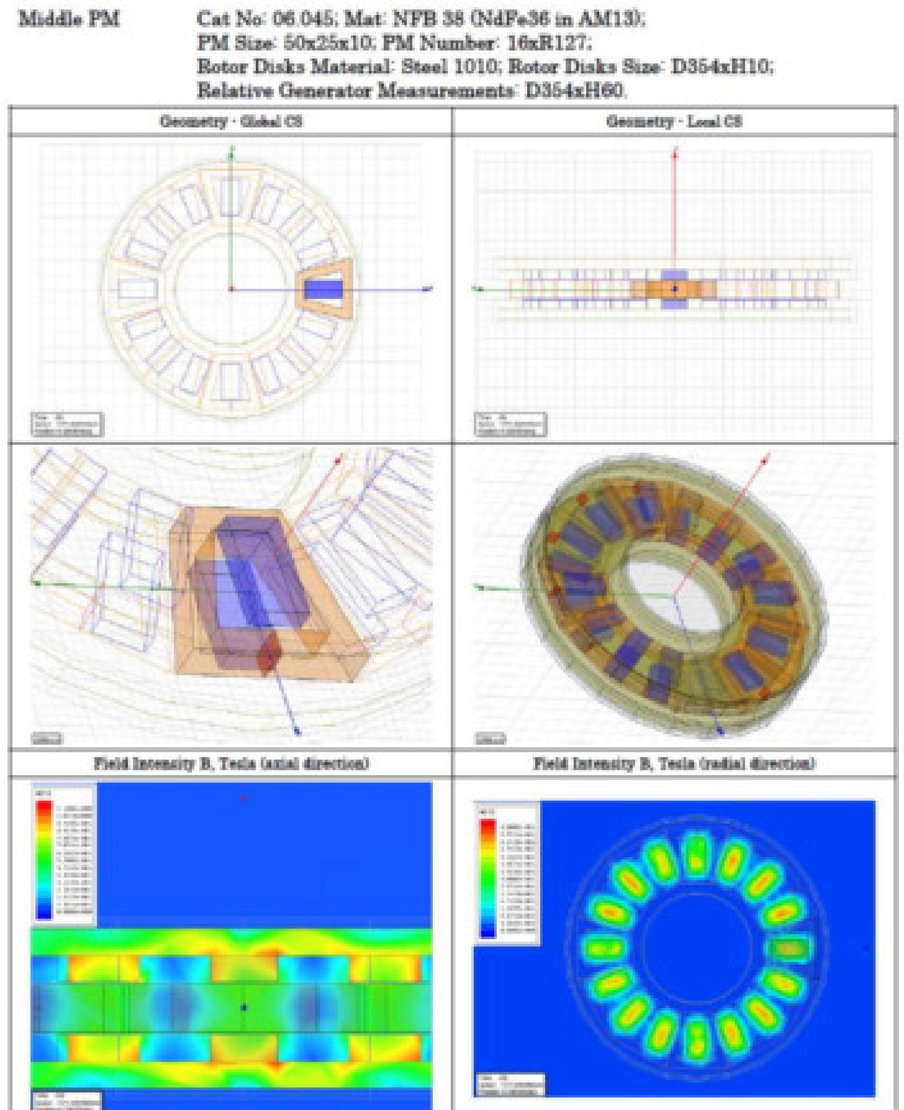
- 3D Simulation results – small cylinder magnets poles

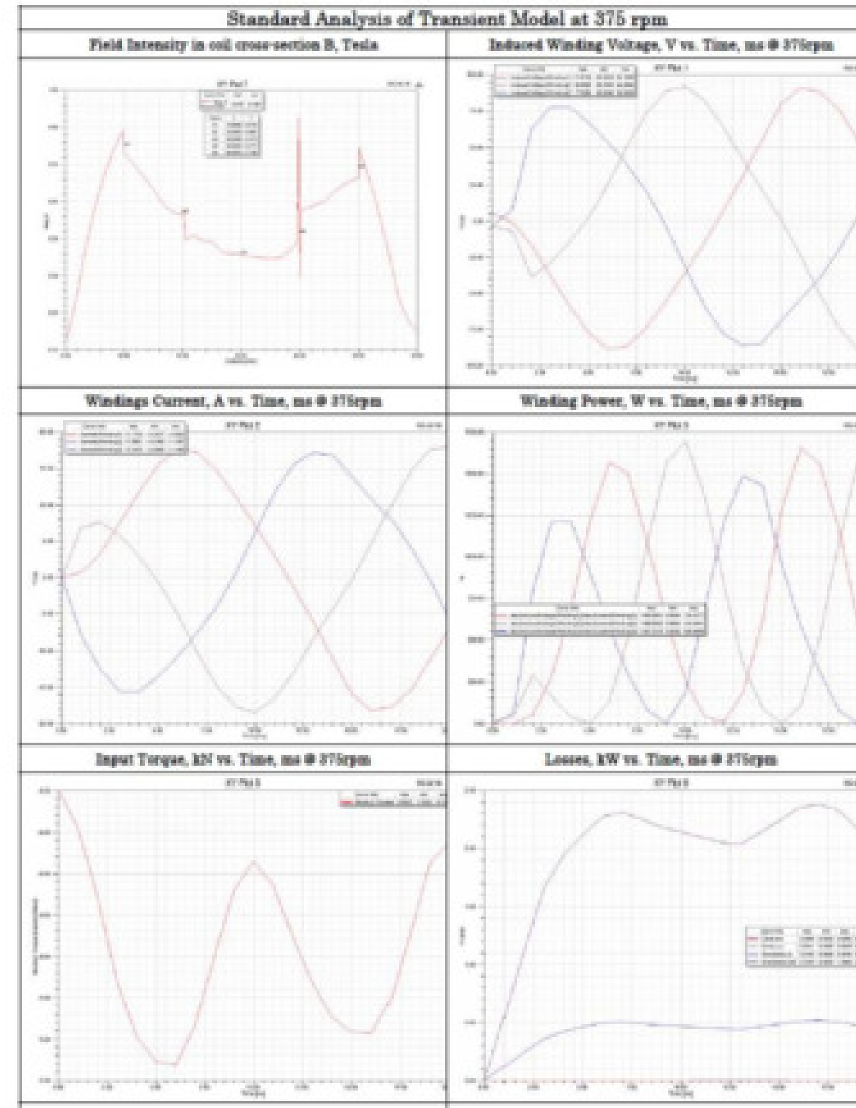




Axial Flux Ironless Generator

- 3D Simulation results – medium parallelepiped magnet poles



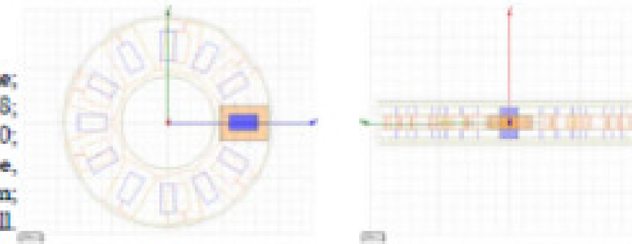


Axial Flux Ironless Generator

- 3D Simulation results – big parallelepiped magnet poles

Simulation case Model-06/Project-01-02-12-sq

Axial Flux Ironless Generator with 12 Poles and 9 Coils connected in 3 Phase Scheme;
NdFeB PM Cat. No 06.013: 24 x Size A2 x B1 x C0.5 in, Mat. NFB 38;
Rotor Disk: 2 x Size D374 x d170 x H6.35 mm, Mat. Steel 1010;
Coils: 9 pcs., Internal Size as PM, Rectangular Shape,
Cross-section 18x18mm, 72 turns, D2.0 mm;
Almost all dimensions are the same or close to once in article from Garrison F. Price at all.



M0-06_P-01-02-12-sq

PM Prise: 400.20

T [Nm]

	T [Nm] @ 2 Ohm Load	T [Nm] @ 4 Ohm Load	T [Nm] @ 6 Ohm Load
200	13.46	7.39	5.53
500	30.57	16.49	11.42
800	44.60	25.18	17.49

Pin [W]

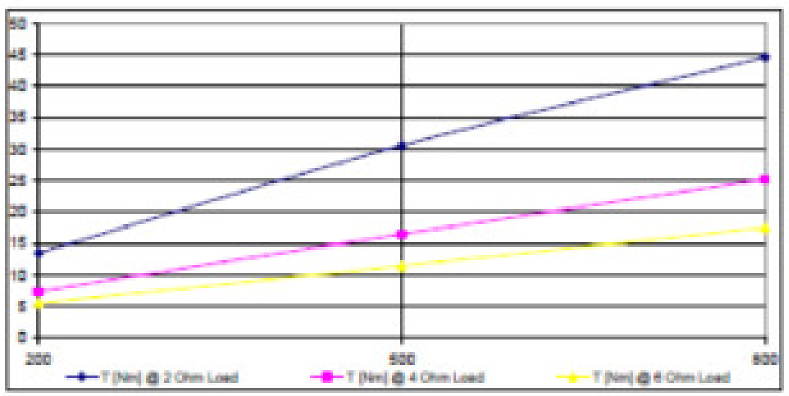
	Pin [W] @ 2 Ohm Load	Pin [W] @ 4 Ohm Load	Pin [W] @ 6 Ohm Load
200	281.9	154.8	115.8
500	1,600.7	863.3	597.9
800	3,736.2	2,109.8	1,465.0

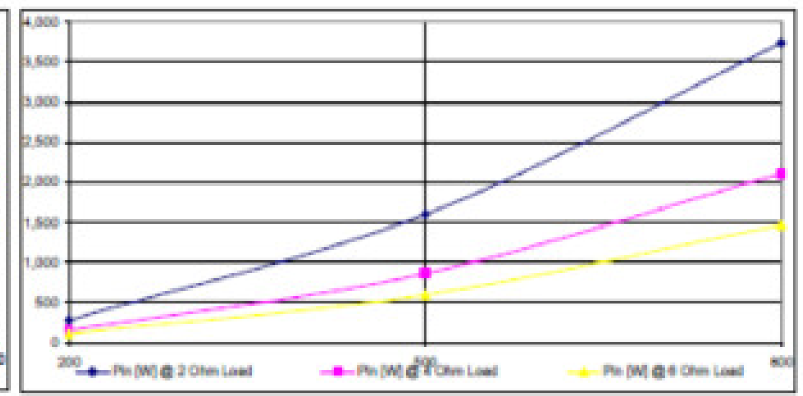
Pout [V]

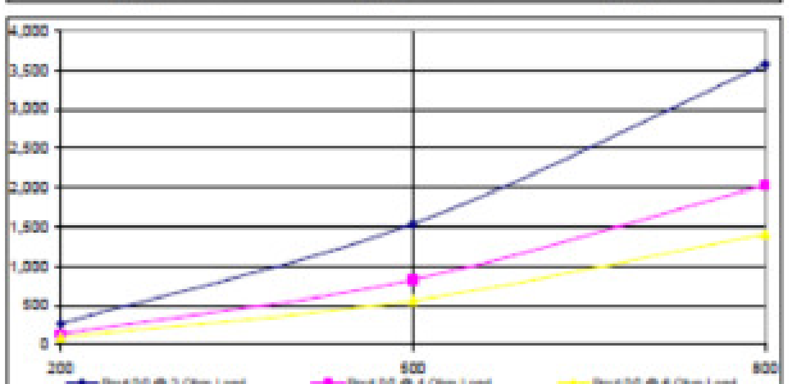
	Pout [V] @ 2 Ohm Load	Pout [V] @ 4 Ohm Load	Pout [V] @ 6 Ohm Load
200	267.99	137.27	92.20
500	1,537.92	826.69	562.99
800	3,564.64	2,029.17	1,404.63

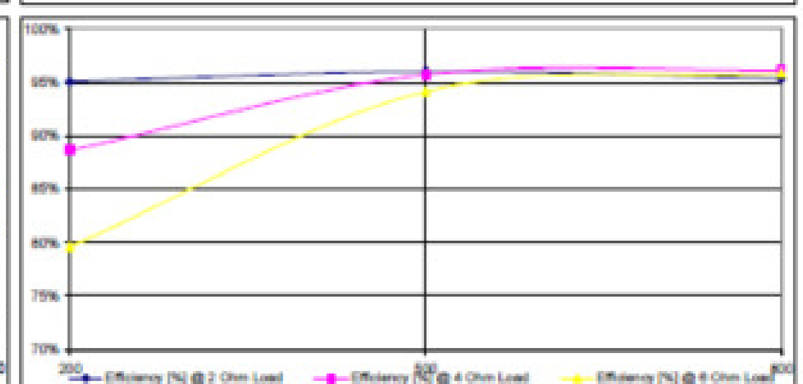
Efficiency [%]

	Efficiency [%] @ 2 Ohm L	Efficiency [%] @ 4 Ohm L	Efficiency [%] @ 6 Ohm Load
200	95.1%	88.7%	79.6%
500	96.1%	95.8%	94.2%
800	95.4%	96.2%	95.9%







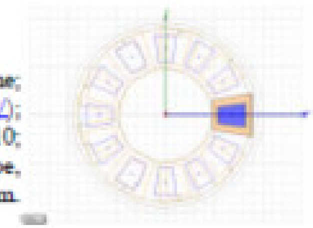


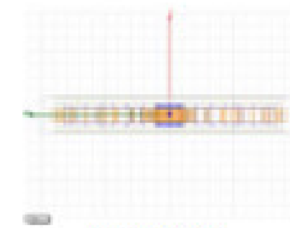
Axial Flux Ironless Generator

- 3D Simulation results – medium wedge magnet poles

Simulation case Model-07/Project-01-01-12

Axial Flux Ironless Generator with 12 Poles and 9 Coils connected in 3 Phase Scheme;
NdFeB PM: 24 x Wedge N42 55*44*30*8 (http://www.solbergavind.se/);
Rotor Disk: 2 x Size D357 x d191 x H8 mm, Mat. Steel 1010;
Coils: 9 pcs., Internal Size as PM, Trapezoidal Shape,
Cross-section 12x18mm, 72 turns, D2.0 mm.





PM Price: 268.80

M0-07_P-01-01-12

T [Nm]

	T [Nm] @ 4 Ohm Load	T [Nm] @ 7 Ohm Load	T [Nm] @ 10 Ohm Load
50	6.46	5.33	4.94
200	23.63	11.60	8.95
500	144.64	56.47	28.52

Pin [W]

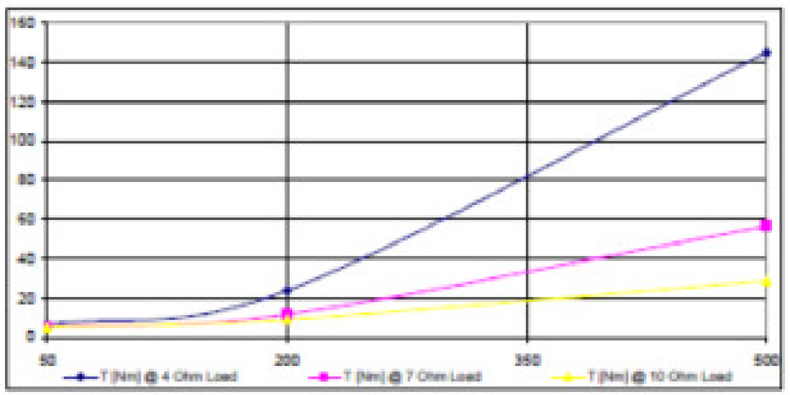
	Pin [W] @ 4 Ohm Load	Pin [W] @ 7 Ohm Load	Pin [W] @ 10 Ohm Load
50	33.8	27.9	25.9
200	495.0	242.9	187.4
500	7,573.3	2,966.7	1,493.4

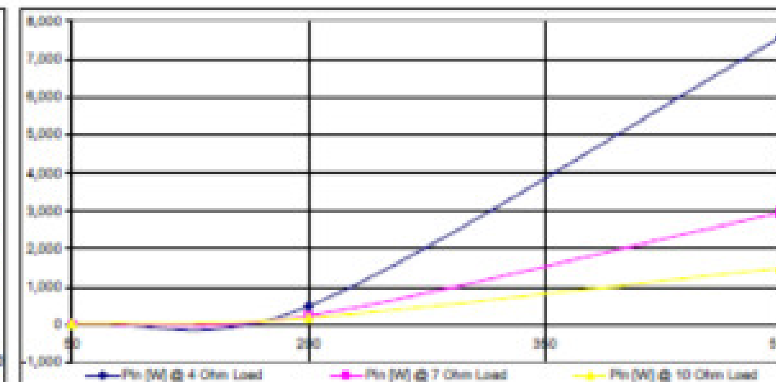
Pout [W]

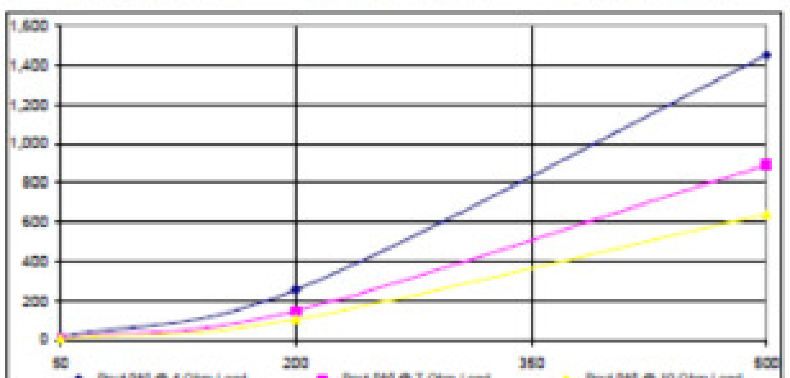
	Pout [W] @ 4 Ohm Load	Pout [W] @ 7 Ohm Load	Pout [W] @ 10 Ohm Load
50	16.66	9.56	6.71
200	256.03	149.82	105.80
500	1450.52	891.12	640.14

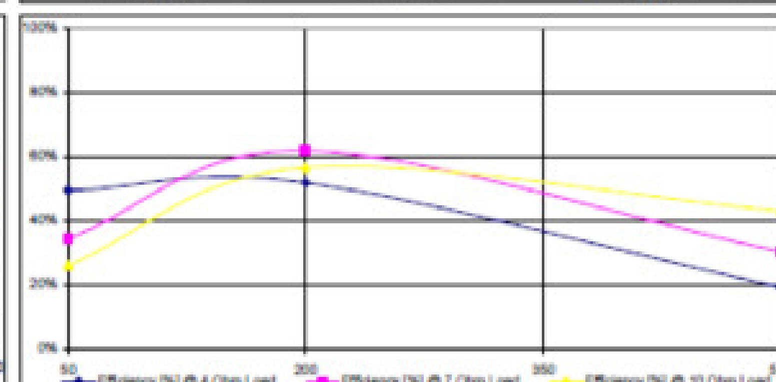
Efficiency [%]

	Efficiency [%] @ 4 Ohm L	Efficiency [%] @ 7 Ohm L	Efficiency [%] @ 10 Ohm Load
50	49.3%	34.3%	25.9%
200	51.7%	61.7%	56.6%
500	19.2%	30.1%	42.9%









Axial Flux Ironless Generator

- After simulation design summary – based on medium wedge magnet poles

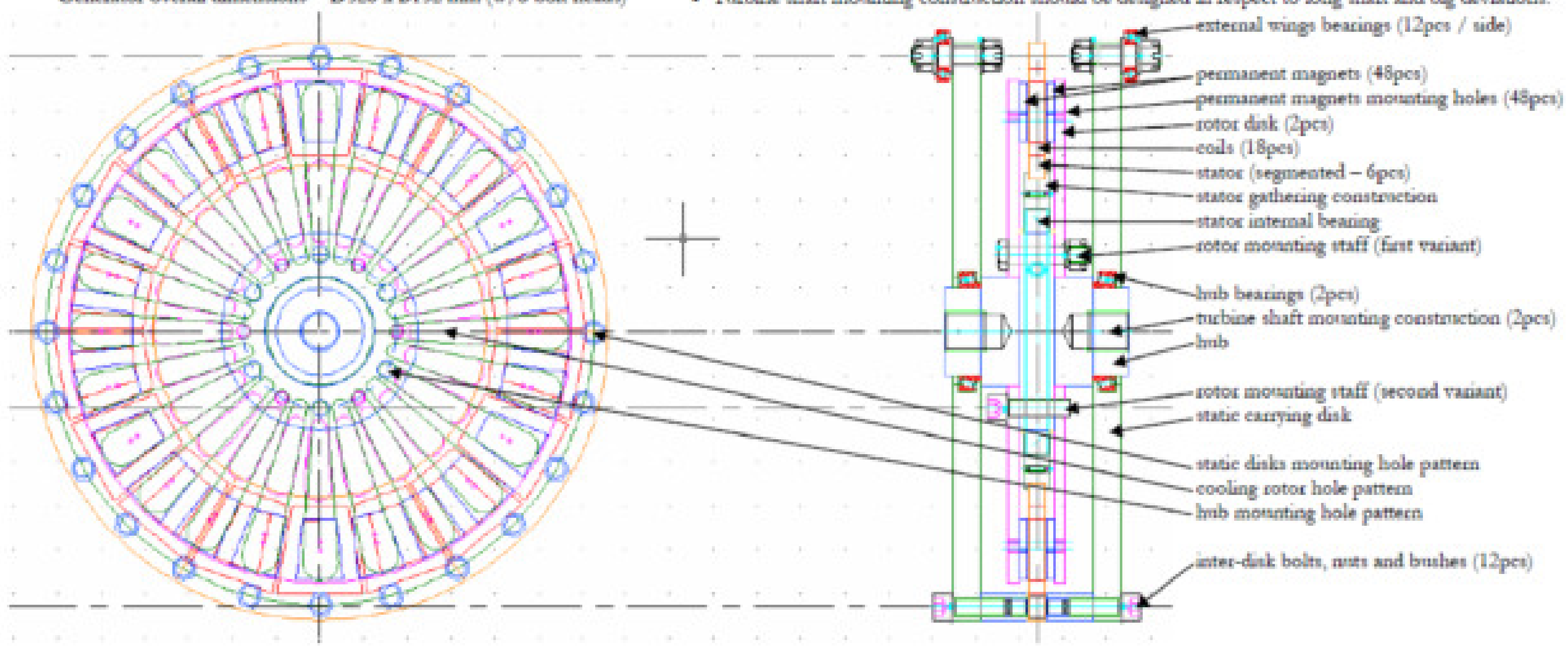
Axial Flux Ironless (AFIL) Generator parameters and construction

PM kind	Wedge 55x44x30x8 hole d3.4 mm
PM pieces / Poles / Pole pairs	48 / 24 / 12
Radial PM repetition ratio	0.70766
Radial Inter PM space	15.36
PM placement pattern	24 x R200 x d3.4 mm
Air gap total	16 mm
Steel Rotor Disk height	10 mm
Steel Rotor Disk External Diameter	D460 mm
Steel Rotor Disk Internal Diameter	D100 mm
Steel Rotor Disk Hole Patterns	24 x R200 x d3.4 mm (PM fixture) 6 x R140 x d16 mm (Hub fixture) 6 x R140 x d10.2 mm (Mount Tool)
Static Carrying Disk Diameter	528 mm
Static Carrying Disk Hole Patterns	12 x R252 d16 mm (inter-disk bolts) 12 x R252 d16 mm (external wings)
Hub Bearings	2 x Roller D110xd80xB20 mm
Stator Internal Bearing	Roller D180xd225xB22 (sx011836)
External wings bearings	Roller D47xd20xB12 (DIN 615)
Generator overall dimensions	D528 x B152 mm (w/o bolt heads)

Geometry	Poles 24 / Coils 18
Coils / Phases / Coils per Phase	18 / 3 / 6
Radial Coil repetition ratio	0.87718
Radial Inter PM space	8.31 @ 12 mm coil width
Coil placement pattern	3 x 6 x R200 mm
Air gap	2 x 1 mm
Coil height	14 mm
Coil cross-section	12 x 14 mm
Coil turns	42 x d2.0 mm
Plastic Stator Disk	D528 x d304 x H14 mm (in mounted state)
Plastic Stator Disk Segments	6
Coils/Phases per Segment	3

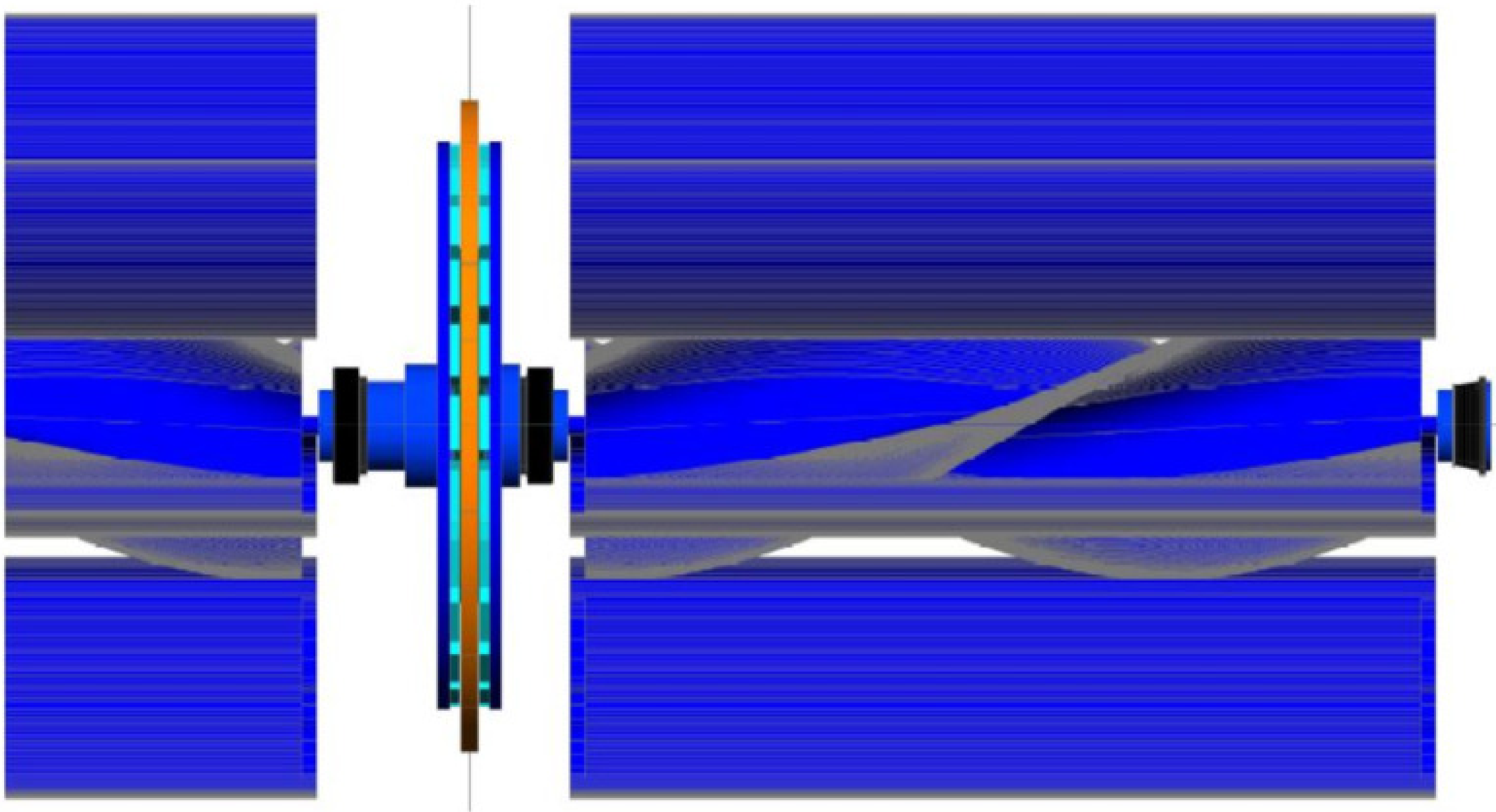
Notes:

- Construction is designed to fit both wedge 55x44x30x8 and 70x46x28x10 PMs;
- Both rotor and carrying disks should be made with maximal cooling efficiency;
- Stator gathering construction should be tuned to fit all incl. 24 and 16 poles 140/6 rotor disks;
- Construction has to be made in respect to turbine mounting and carrying;
- Construction should to be designed to be tolerant to external environment;
- Turbine shaft mounting construction should be designed in respect to long shaft and big deviations.



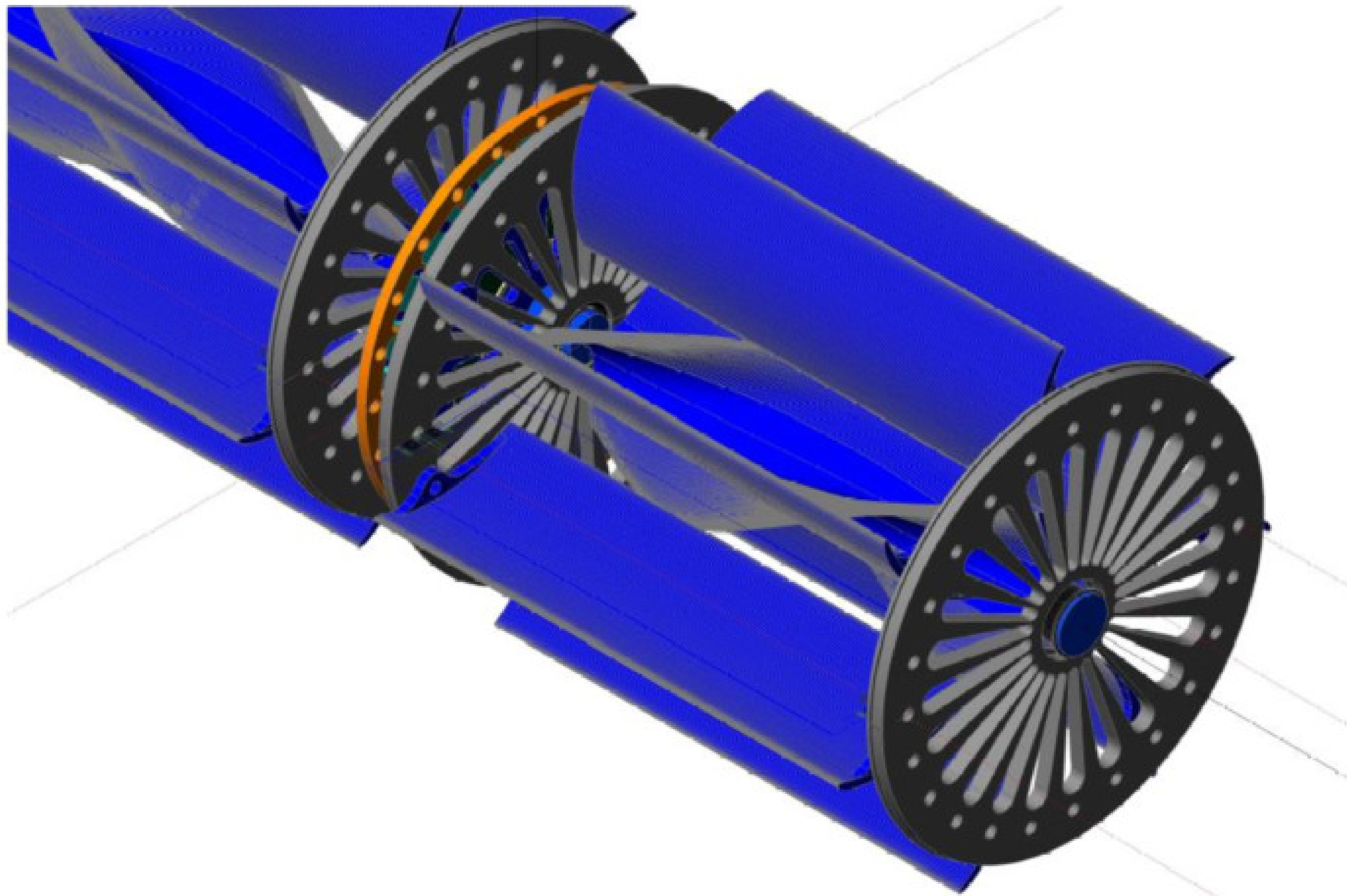
Generator Construction

- 3D model of wind turbine, generator rotor and stator – first revision



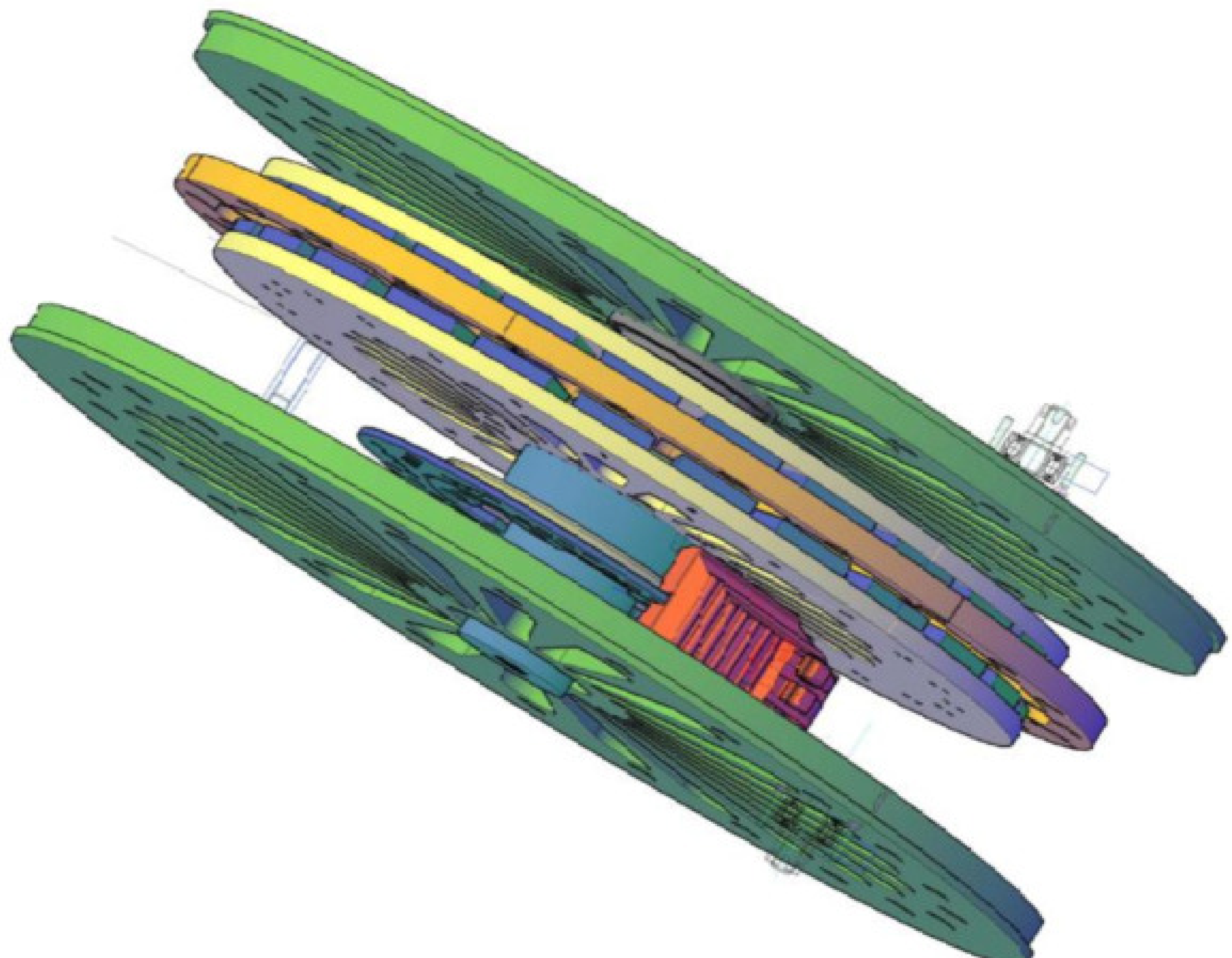
Generator Construction

- 3D model of wind turbine, generator rotor and stator – first revision



Generator Construction

- 3D model of wind turbine, generator rotor and stator – first revision

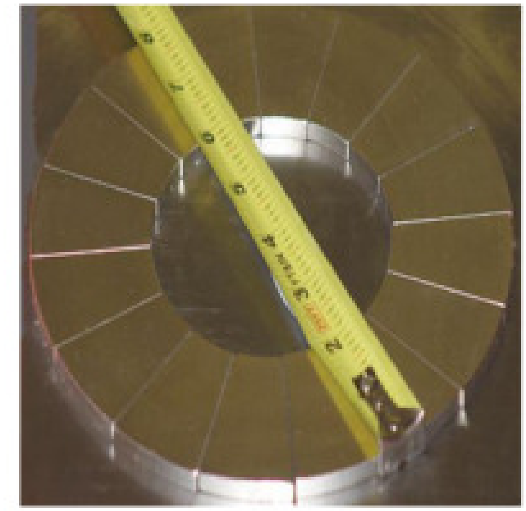


Generator Construction

- NdFeB Permanent Magnets – supplier selection and delivery

Dailymag Magnetic Technology (Ningbo) Limited
is a Chinese leading manufacturer and exporter
of permanent magnets etc.

Wind Generator NdFeB Magnets 22.5 degree
8 inch OD x 4 inch ID x 0.5 inch thick
Wedge Segment Shape, Grade N35~N52
Nickel-Copper-Nickel triple layer coated

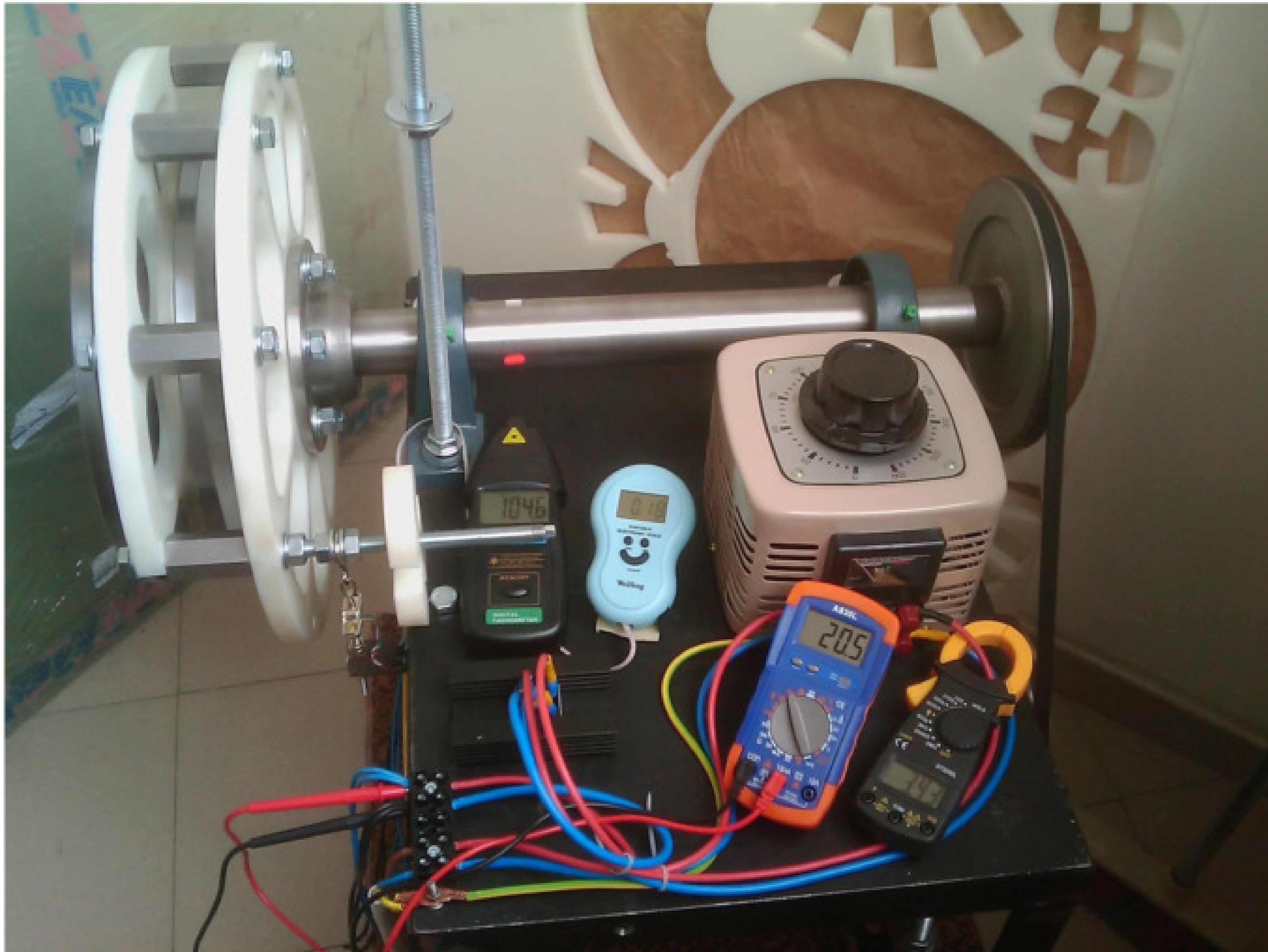


Generator Implementation

- The rotor disk

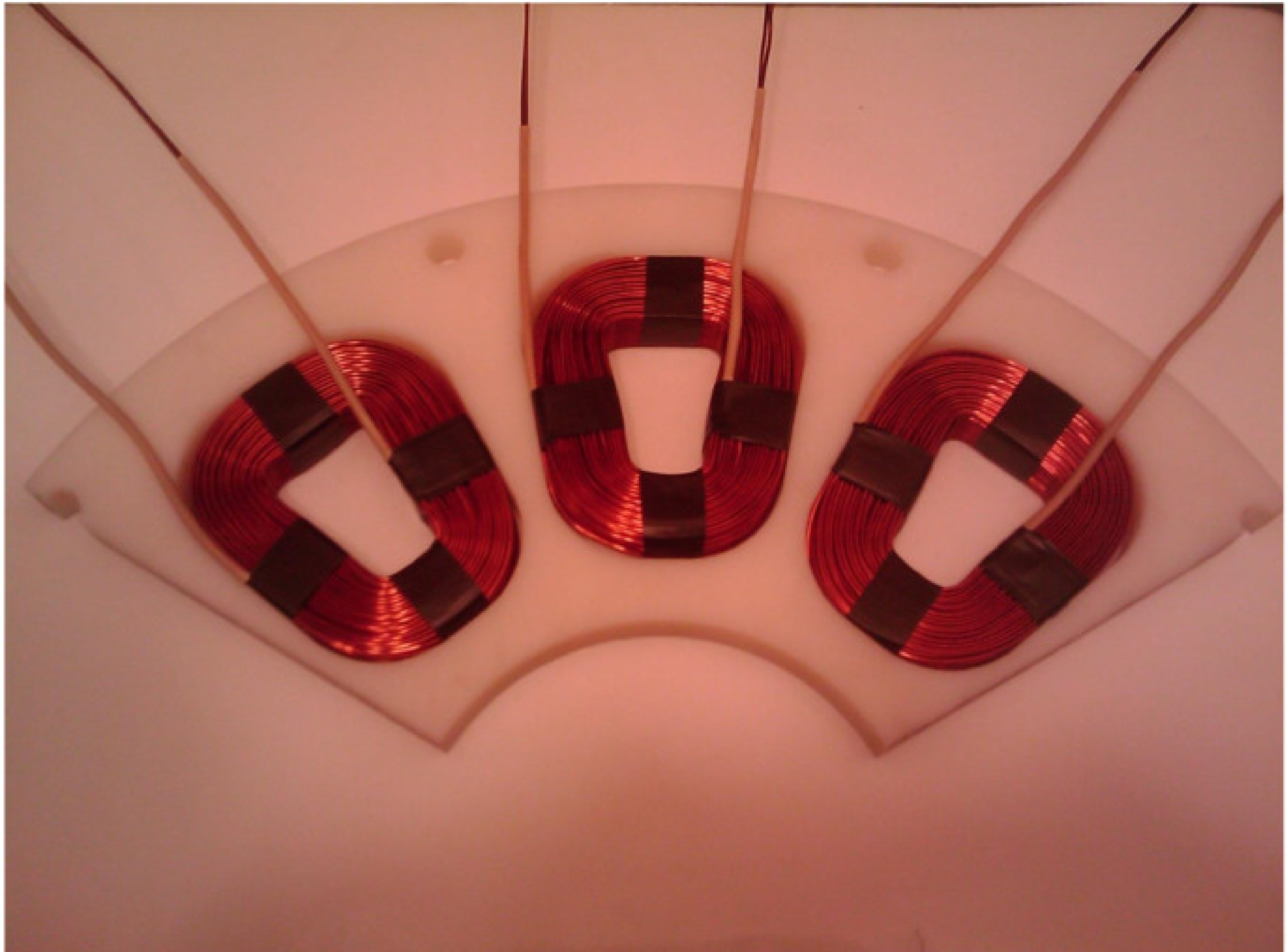
Generator Implementation

- The startup staff



Generator Implementation

- The stator segment

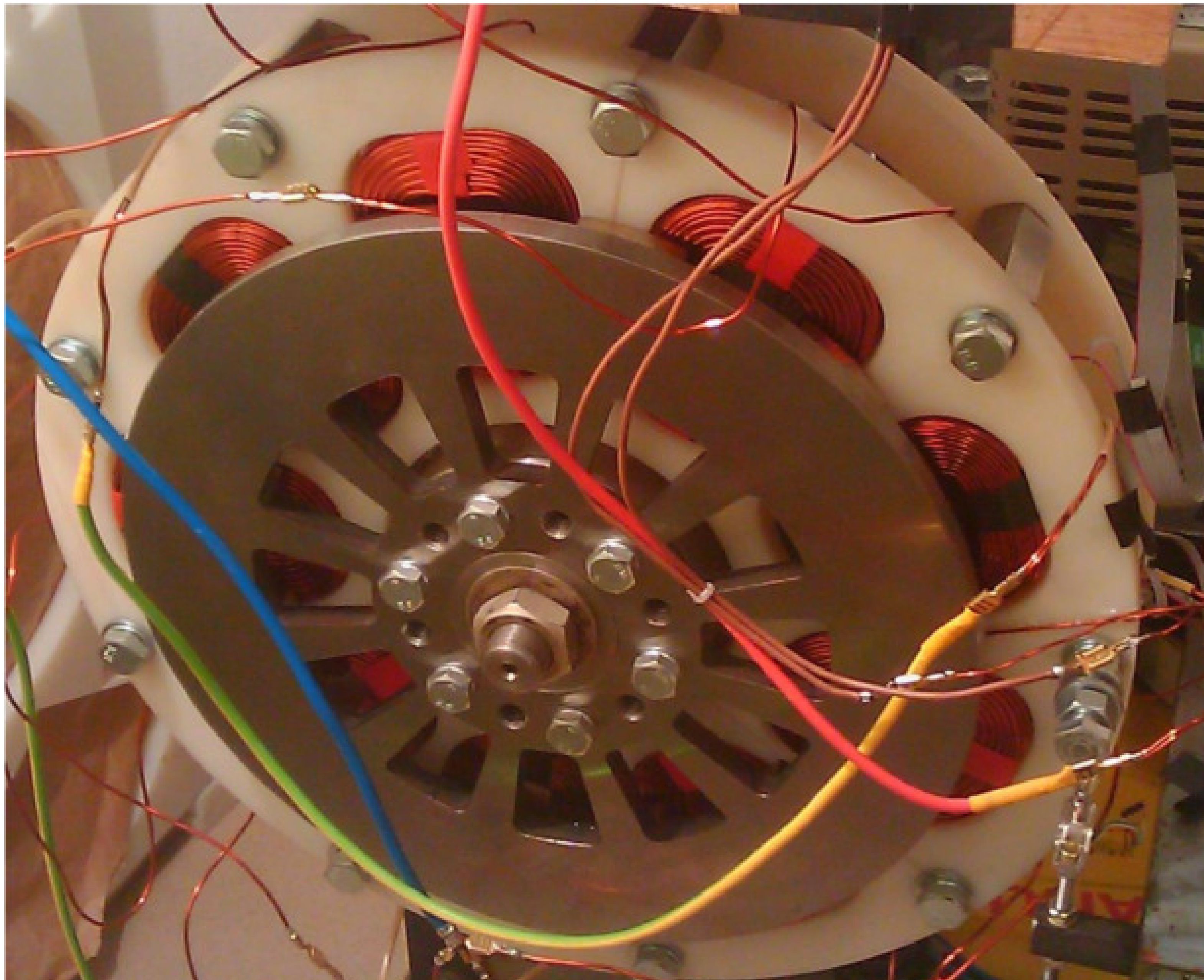


Generator Implementation

- The first assembling together

Generator Implementation

- Generator mounted on the test bed – (both first revision)



Test Bed Implementation

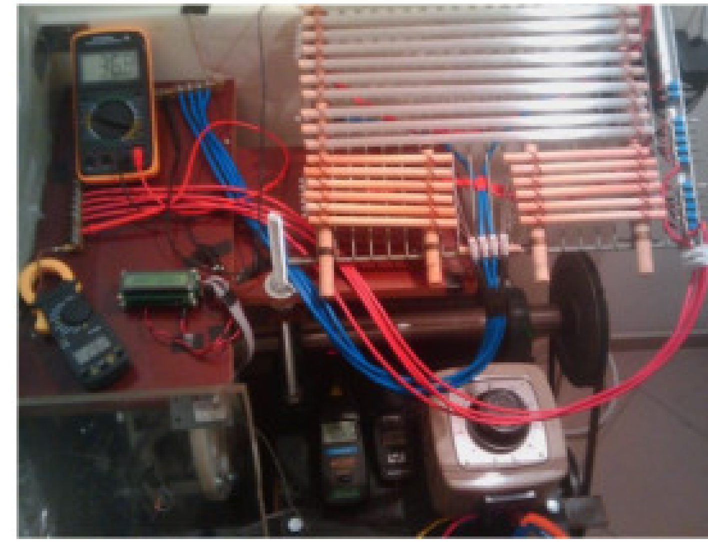
- Test bed – first revision

Generator Testing

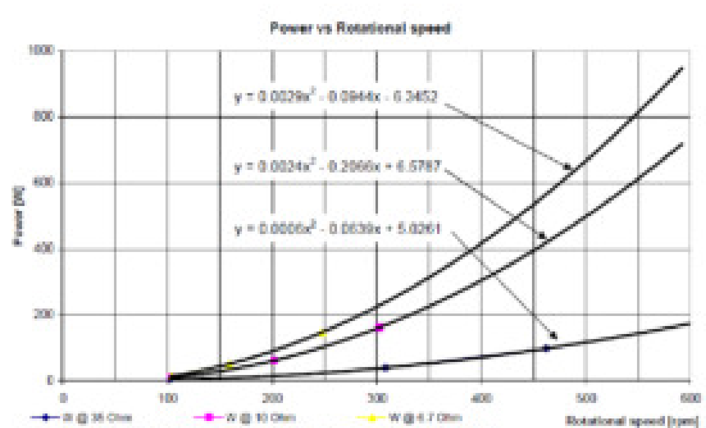
- Generator mounted on the test bed – energy production

First Light

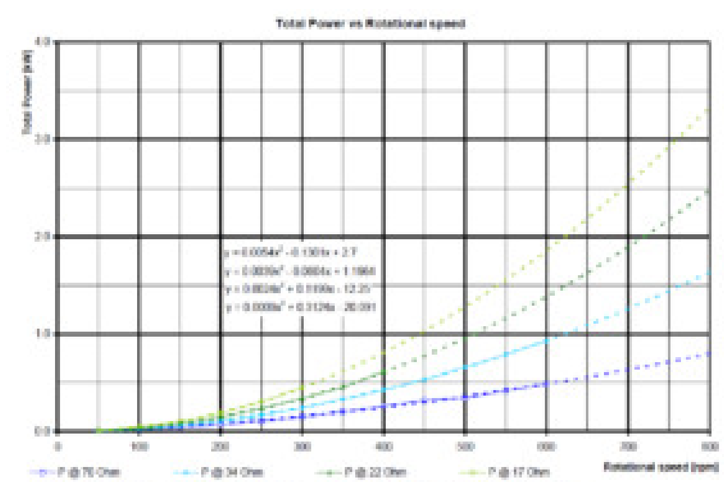
First Watts



First measured 100ths of Watts

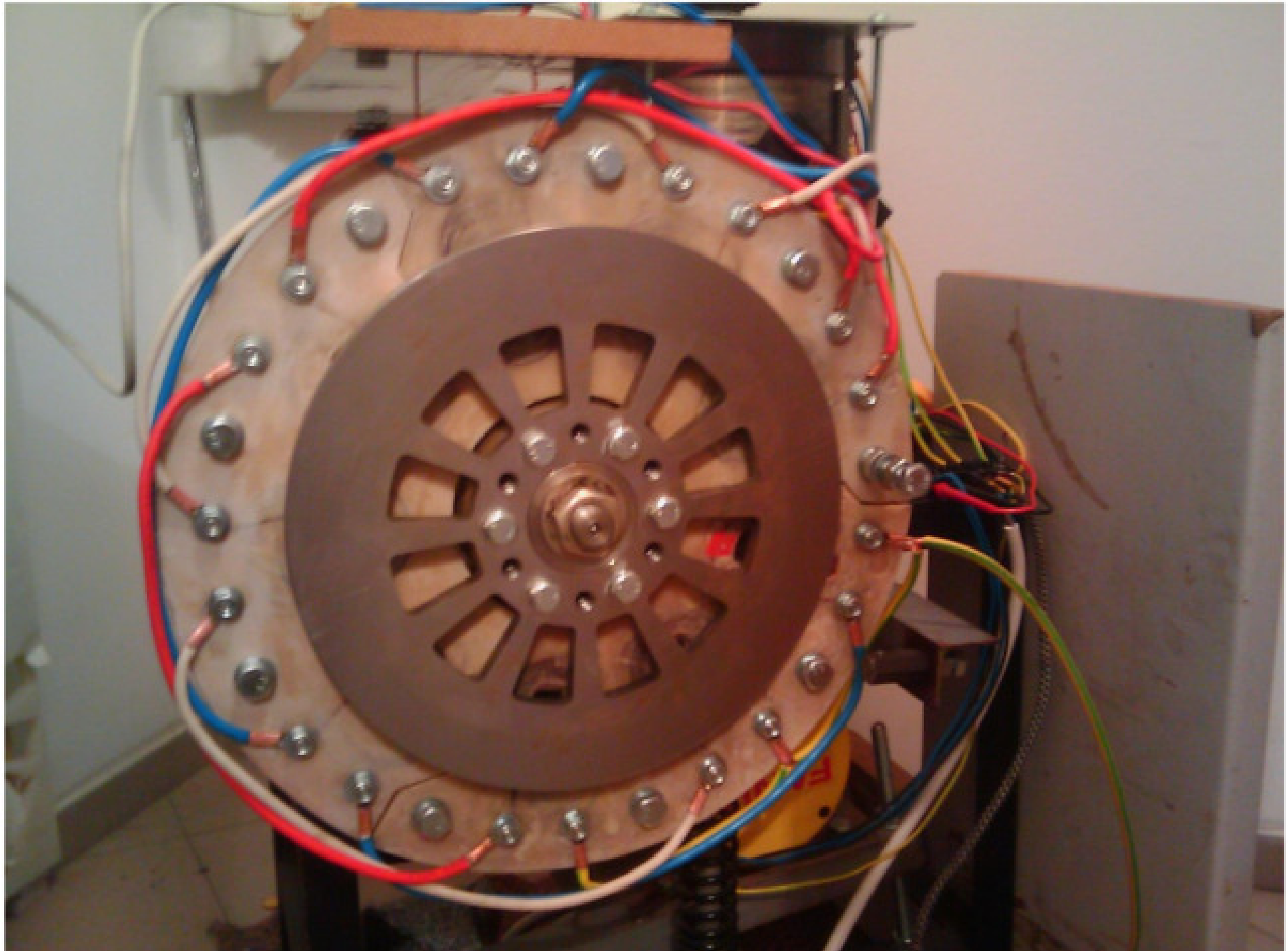


First measured 1000 Watts



Generator Improvements

- Wind Generator – first revision with modified stator (molding technology)



Generator and Turbine

- First assembling of the generator and the turbine

Generator and Turbine

- Rotational test of the generator and the turbine

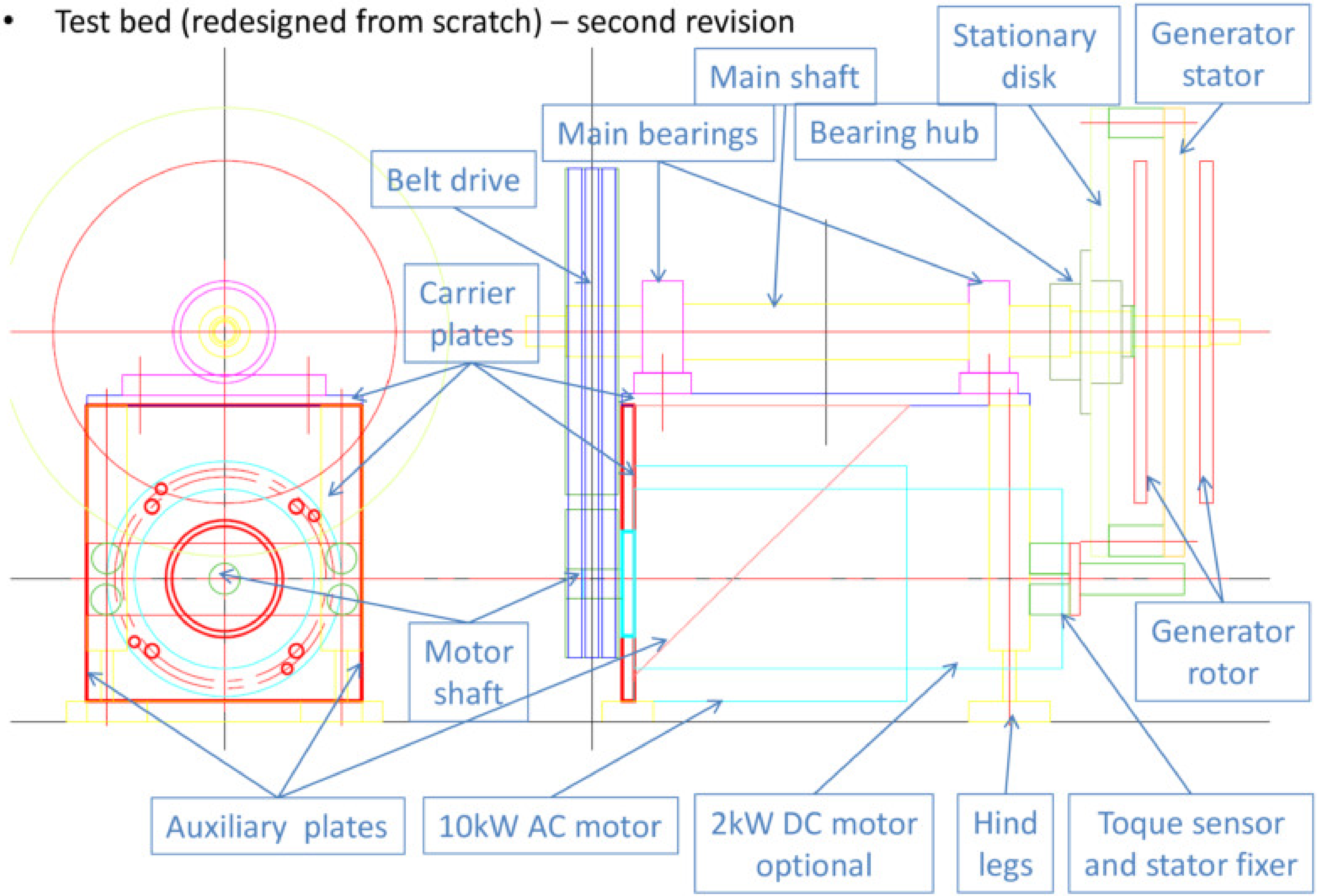
Generator, Turbine and Wings

- The generator, the turbine and the wings

The field test suite (Belmeken)

Test Bed Improvements

- Test bed (redesigned from scratch) – second revision



Mechanical System Improvement

- Test bed – belt drive is using V-Belts pulleys for taper bushes by Bea Ing. S.P.A.

PULEGGE A GOLE TRAPEZOIDALI PER BUSSOLA CONICA
V-BELTS PULLEYS FOR TAPER BUSHES

Descrizione e caratteristiche - Description and features

Le pulegge per cinghie trapezoidali sono costruite secondo le specifiche ISO 4183 / DIN 2211
Our V-belt pulleys are manufactured according to International Standard ISO 4183 / DIN 2211

Materiale - Material

Ghisa EN-GJL-200 (G20 - UNI 5007)
Cast iron EN-GJL-200 (G20 - UNI 5007)

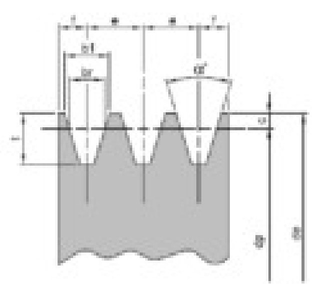
Trattamento e Bilanciatura
Protective treatment and Balancing

Tutte le pulegge standard sono protette con un trattamento superficiale di FOSFATAZIONE e BILANCIATE STATICAMENTE per essere idonee ad un funzionamento fino alla velocità periferica di 35 m/sec.
The surface of all our standard pulleys is protected by phosphated treatment. All the pulleys are Statically Balanced and can be used for peripheric speed up to 35 m/sec

Calcolo della velocità periferica (Vp)
Periferic speed table (Vp)

$$Vp = \frac{\pi \cdot dp \cdot n}{60 \cdot 1000} = \frac{dp \cdot n}{19100} = m/sec$$

dp = **diametro in mm** - diameter/mm
n = **giri al minuto** - revolutions per minute
Vp = **velocità in m/sec** - speed

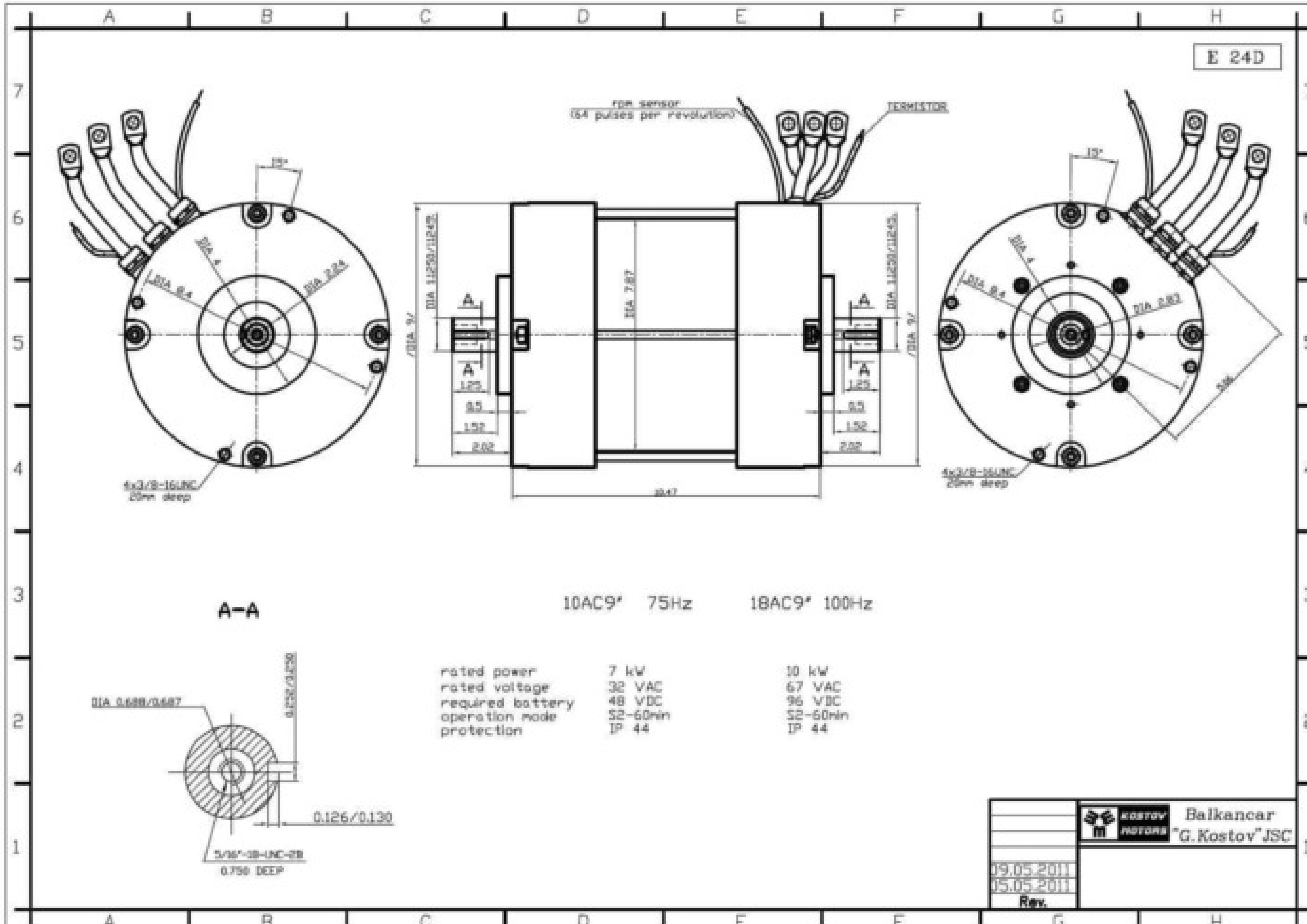


Dimensioni puleggia
Dimension of Pulleys

codice	dp mm	α gradi	b1 mm	br mm	e mm	f mm	c mm	t mm
SPZ (mm)	< 80 > 80	34° 38°	9,7	8,5	12	8	2	11
SPA (mm)	< 118 > 118	34° 38°	12,7	11	15	10	2,8	13,8
SPB (mm)	< 190 > 190	34° 38°	16,3	14	19	12,5	3,5	17,5
SPC (mm)	< 315 > 315	34° 38°	22,0	19	25,5	17	4,8	23,8

Electrical System Improvement

- Test bed with Kostov's AC Motor and Curtis' Controller



Electrical System Improvement

- Test bed with Kostov's AC Motor and Curtis' Controller

ON-ROAD AC INDUCTION MOTOR CONTROLLER

CURTIS

MODEL 1238R

Only Curtis AC can offer:

- **Curtis VCL—Vehicle Control Language** is an easy to use programming language that allows vehicle developers to write powerful logic functions and create a 'virtual system controller'. Curtis offers customers VCL development tools and training. Curtis also provides a VCL service where Curtis engineers will work with the OEM to create any custom VCL code required.
- **Indirect Field Orientation (IFO) vector control** algorithm generates the maximum possible torque and efficiency across the entire speed range. Advanced Curtis IFO vector control provides superb drive 'feel', improved speed regulation and increased gradeability.
- **Curtis Auto-Tune** function enables quick and easy characterization of the AC motor without having to remove it from the vehicle. Curtis AC controllers are fully compatible with any brand of AC motor.
- **Dual-Drive functionality** is standard, allowing correct control of applications featuring twin traction motors. This function ensures smooth and safe operation, minimal tire wear and correct load sharing between the traction motors at all times.
- **Configurable CANbus** connection allows communication with other CANbus enabled devices. Model 1238R is CANopen compatible and can be further customized and configured using VCL.
- **Integrated System Controller** - More than just a motor controller, it is also powerful system controller. It features a comprehensive allocation of multi-function I/O pins for use as analog inputs, digital inputs, contactor coil drivers and proportional valve drivers. In addition to this local I/O, this controller can use VCL to map and configure the remote I/O available on other CANbus devices, send messages to CAN displays and thus control and monitor the entire system.

DESCRIPTION

The Curtis Model 1238R provides energy efficient control of AC induction motors performing on-vehicle traction drive duties. It offers vehicle developers a highly cost-effective combination of power, performance and functionality.

APPLICATION

Designed for use as a traction controller for on-road electric and hybrid passenger vehicles using 72–96V system voltages, and other similar applications with low or medium duty cycles.

Patents Pending

FEATURES

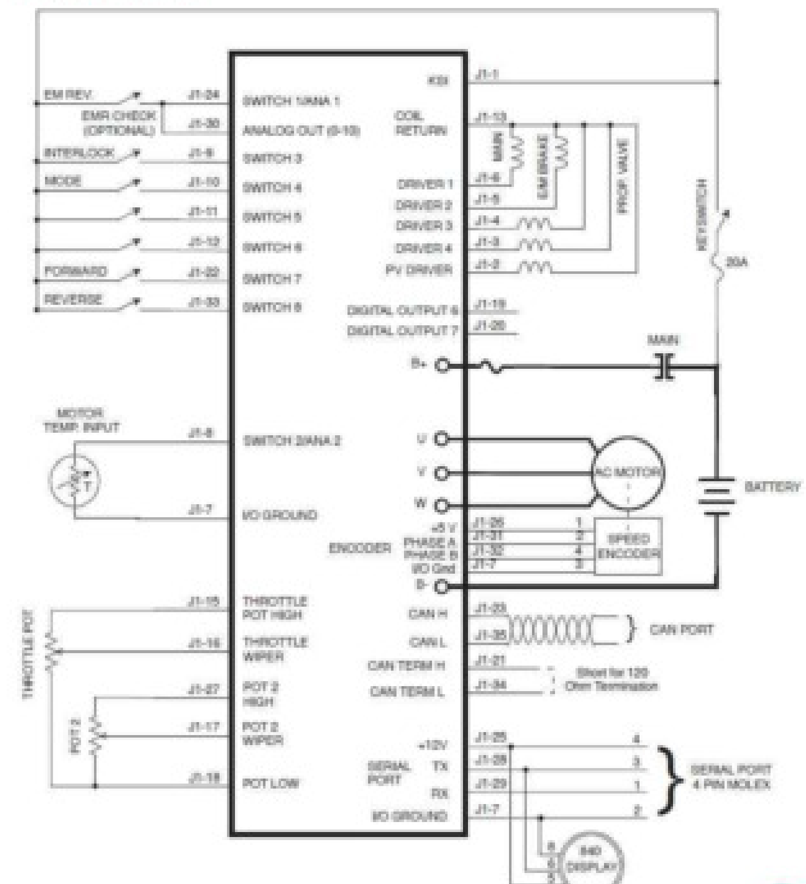
Advanced functionality, compact power

- High frequency, silent operation across the 0–300Hz stator frequency range.
- Two models of 1238R are available, offering outputs of 530A or 650A for a 72–96V nominal system voltage. This is a 72–96V nominal system voltage. This is a true 2 minute RMS rating, not a short duration 'boost' rating.

www.curtisinstruments.com

MODEL 1238R

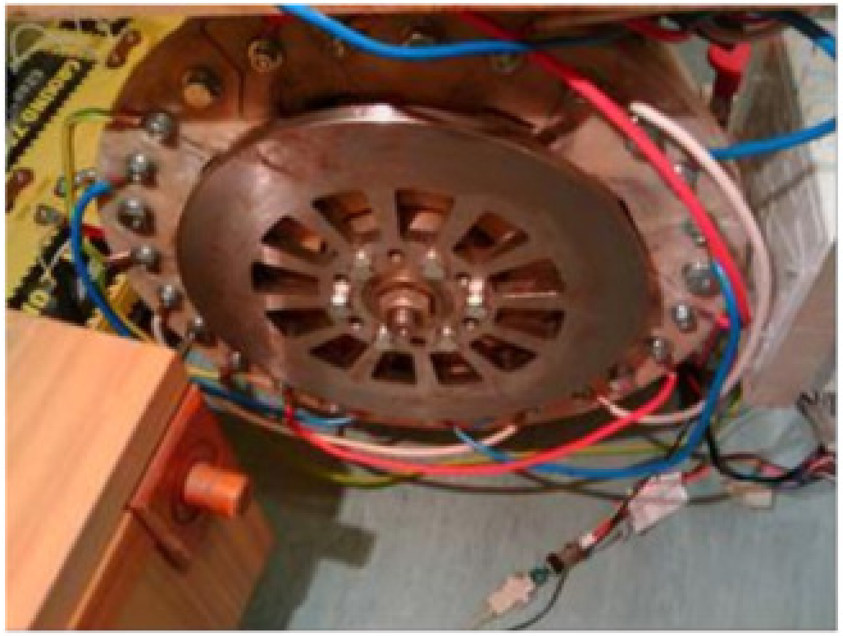
TYPICAL WIRING



WARRANTY Two year limited warranty from time of delivery.

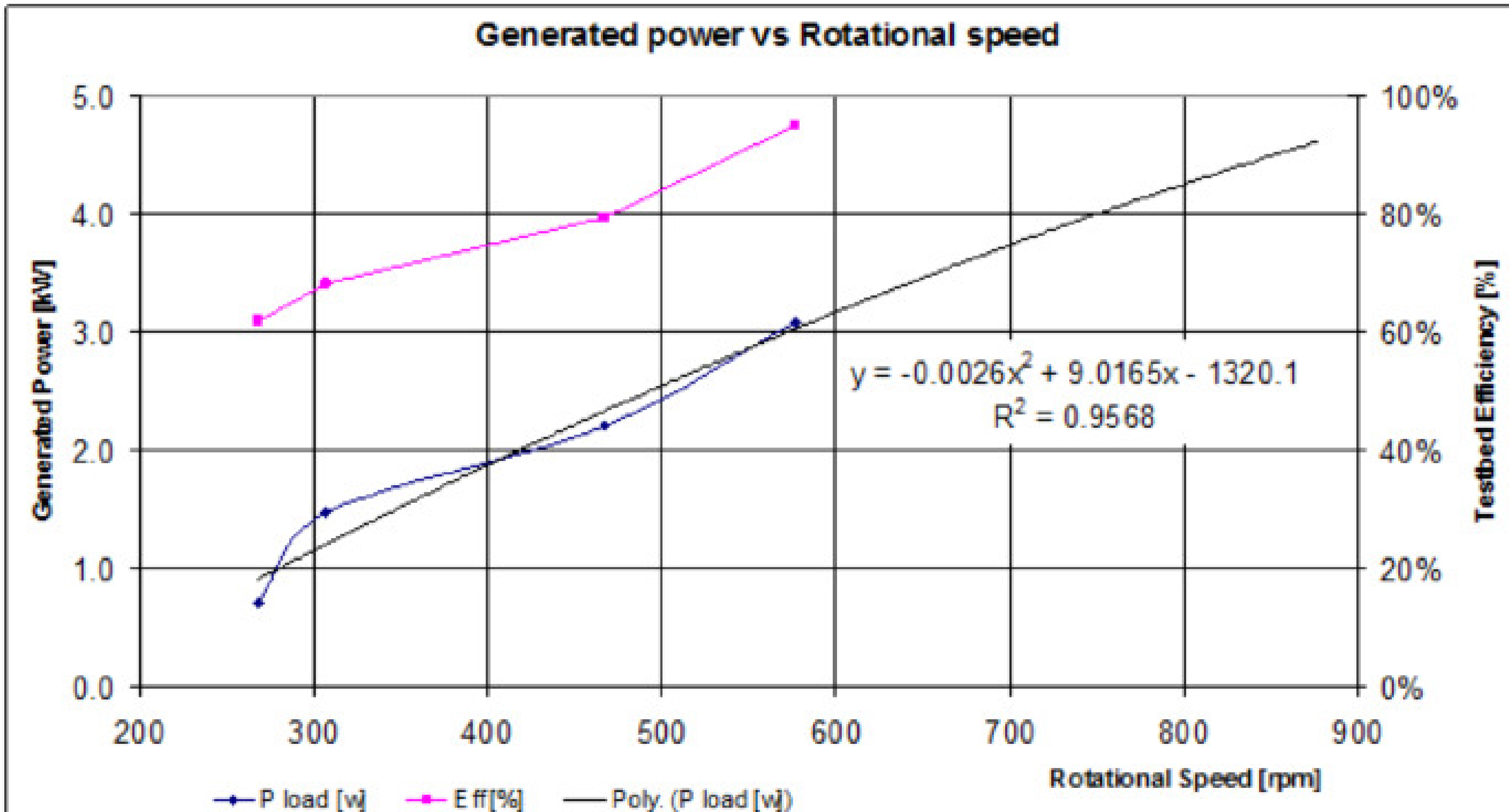
Test Bed Improvements

- Test bed (redesigned from scratch) – second revision



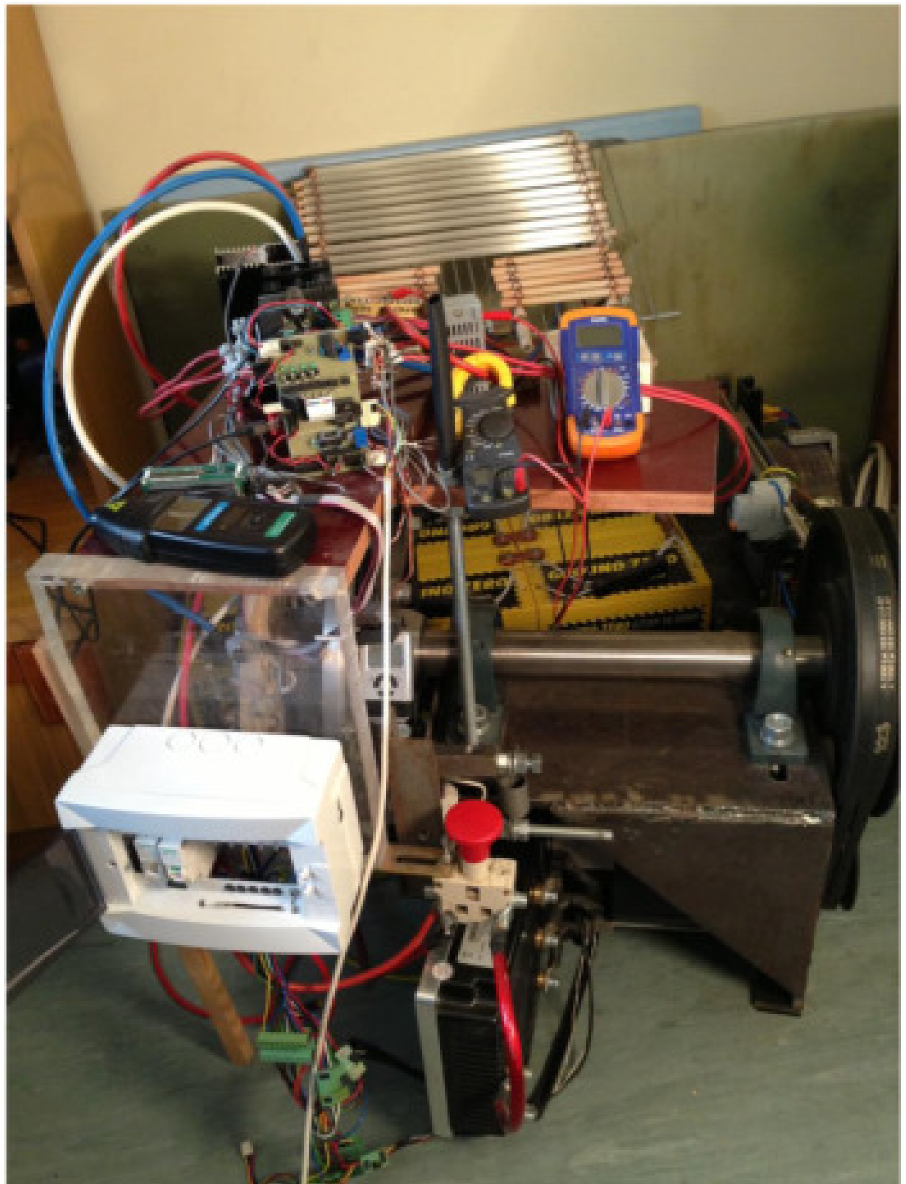
Generator Testing

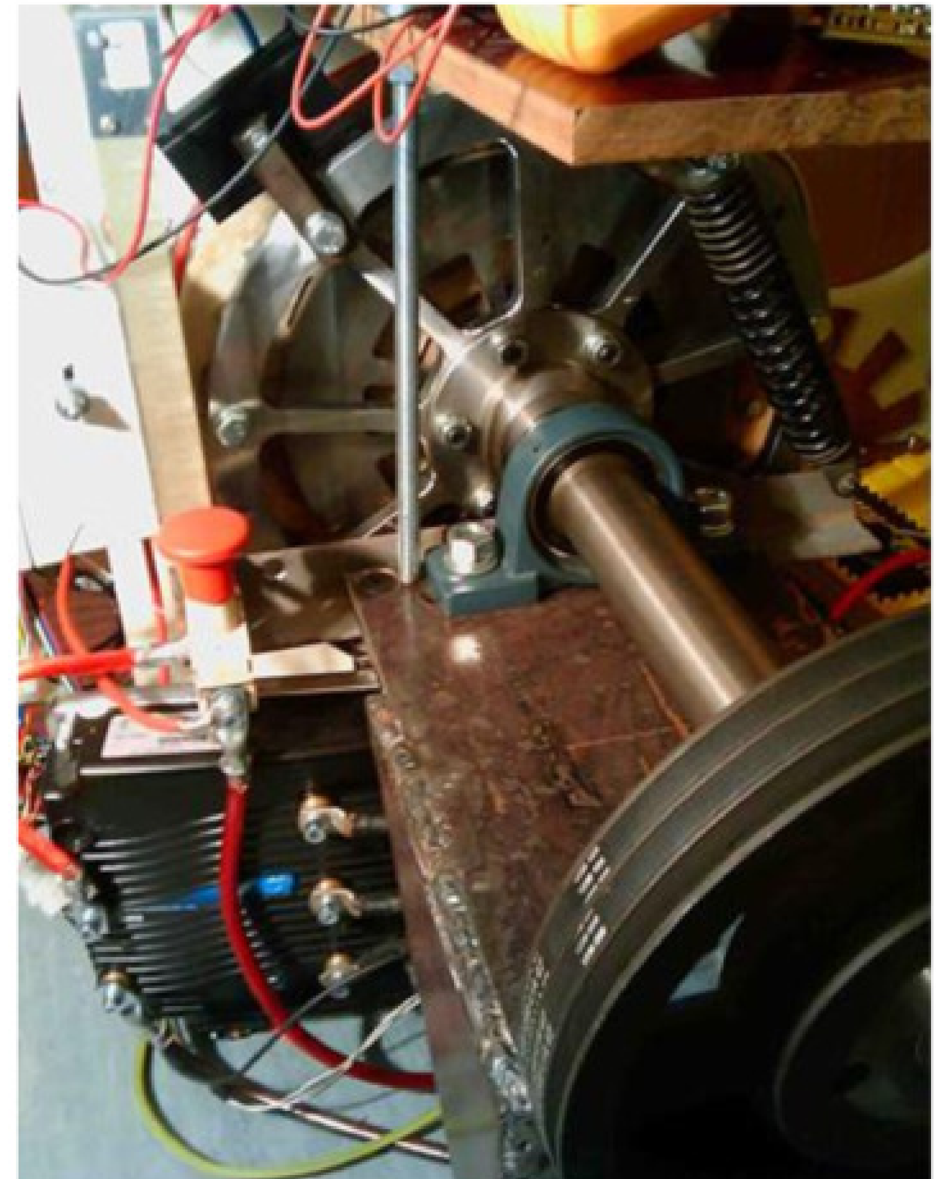
- Generator mounted on the second revision test bed – energy production



Test Bed Improvements

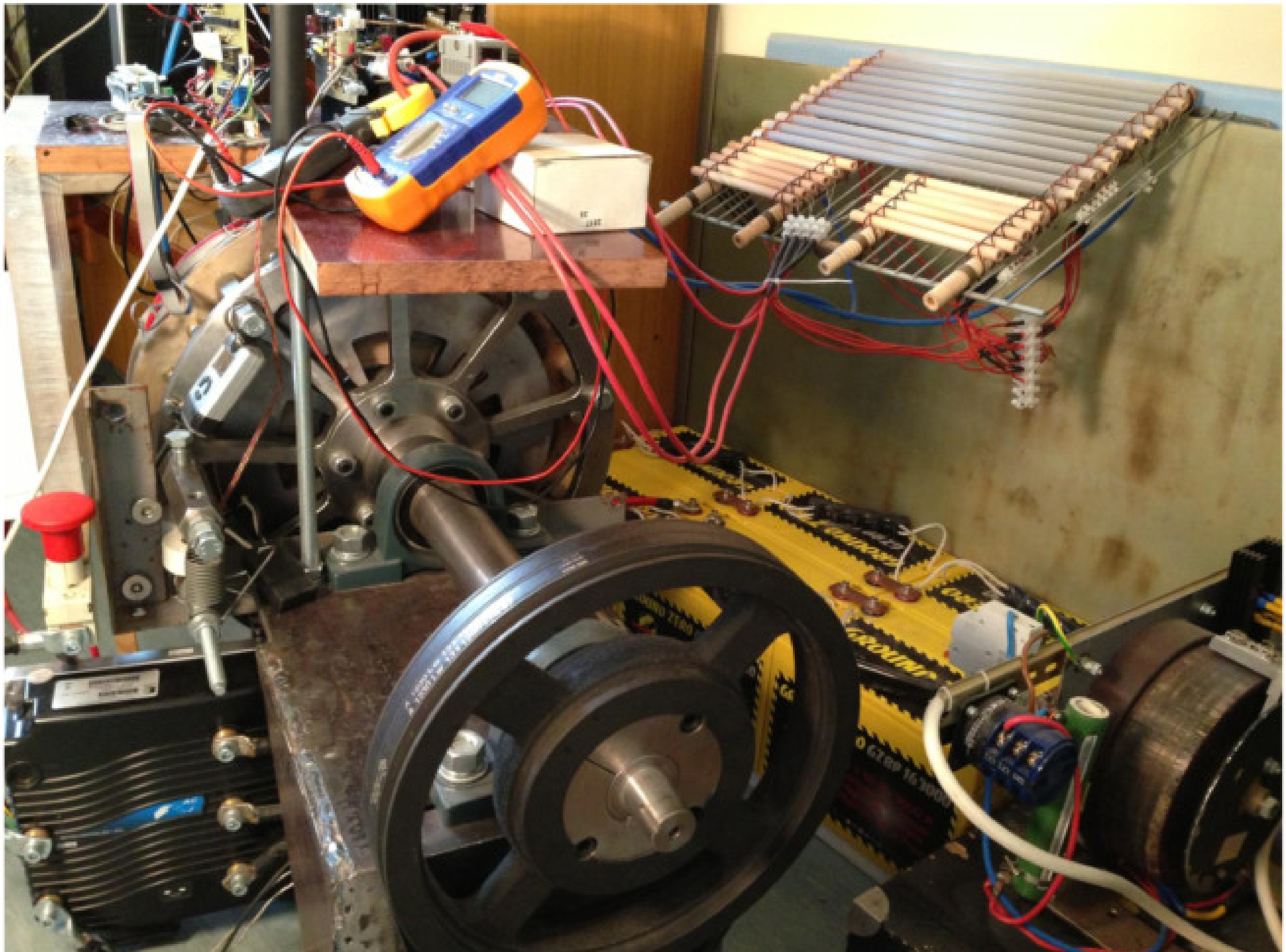
- Test bed – modified second revision





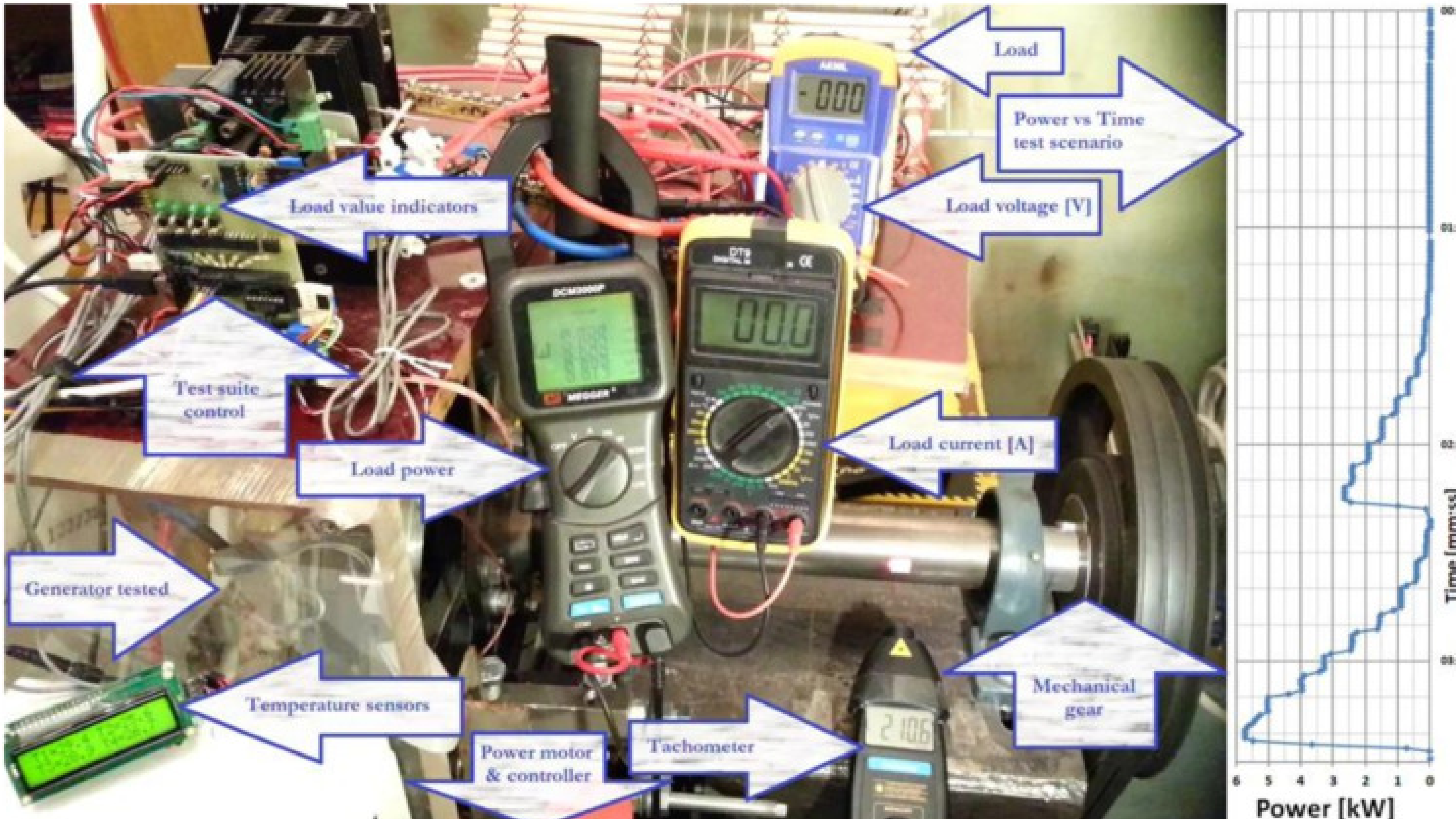
Test Bed Improvements

- Test bed – modified second revision



Test Bed Improvements

- Test bed – modified second revision (complete staff)

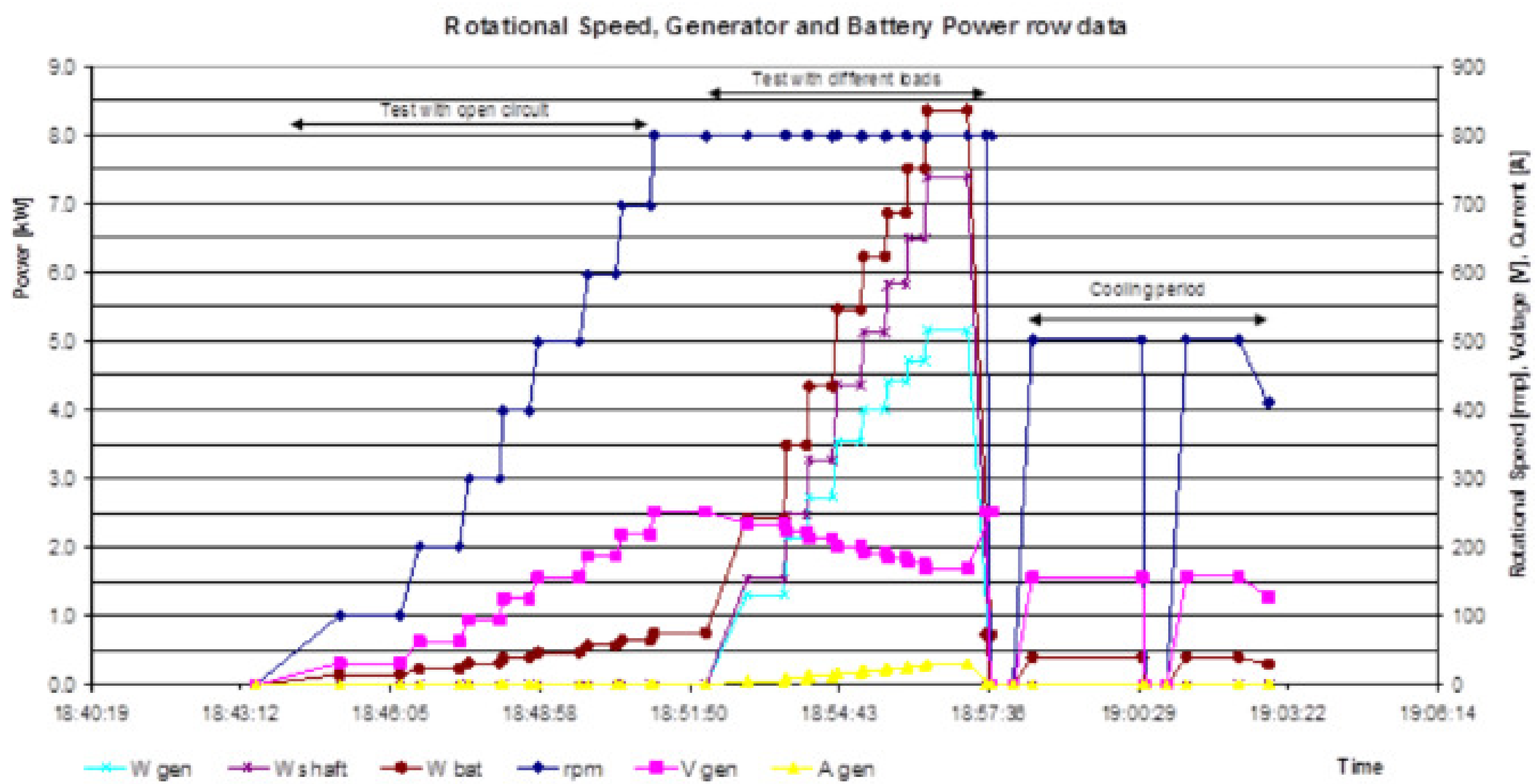


Generator Testing

- Generator mounted on the second revision test bed – energy production

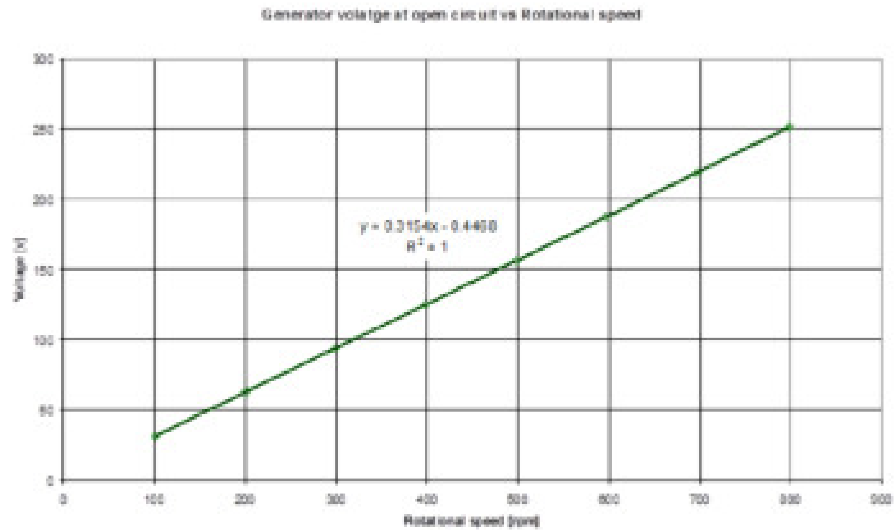
Tests after mounting torque and temperature sensors

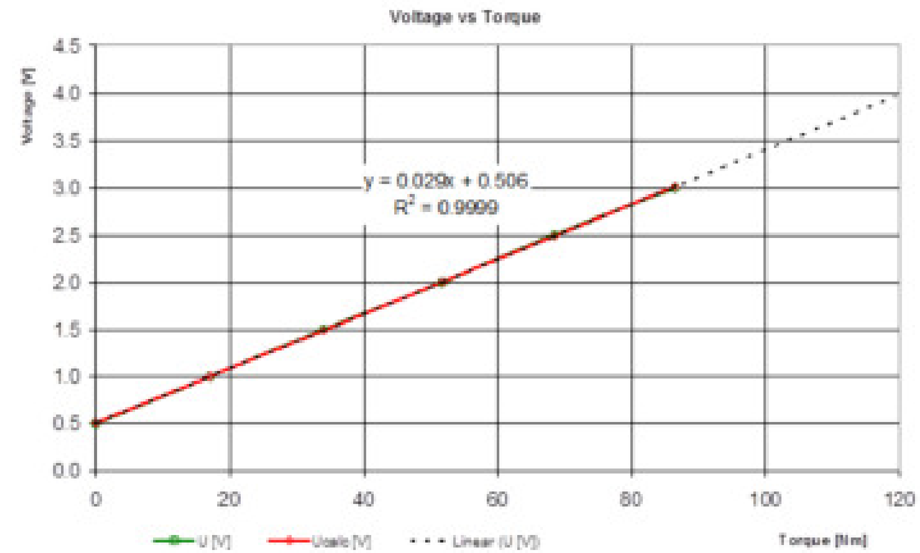
- Short time max power measured on 18 March – 5150 W (30.5A x 169V) @ 798 rpm and 5.5 Ohm load;
- Long time max power measured on 17 March – 4860 W (28.5A x 171.1V) @ 806 rpm and 6.2 Ohm load;
- Short time max power measured on 14 March – 4200 W (26.1A x 161.4V) @ 725 rpm and 6.2 Ohm load;
- Long term max power measured on 12 March – 3300 W (21.6A x 152.5V) @ 650 rpm and 7.1 Ohm load;
- More power output at rpm above 800 and load current above 30 A can be reached with better mechanical balancing and stator cooling.

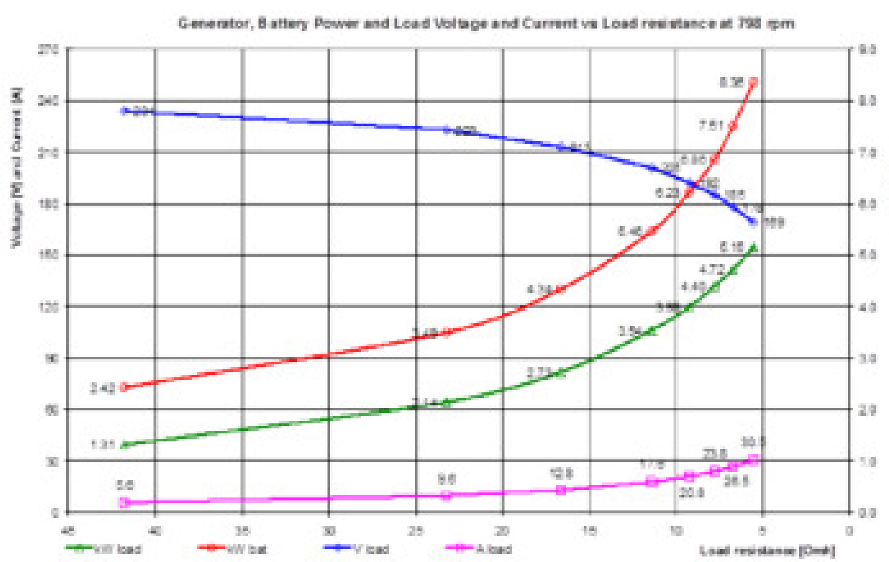


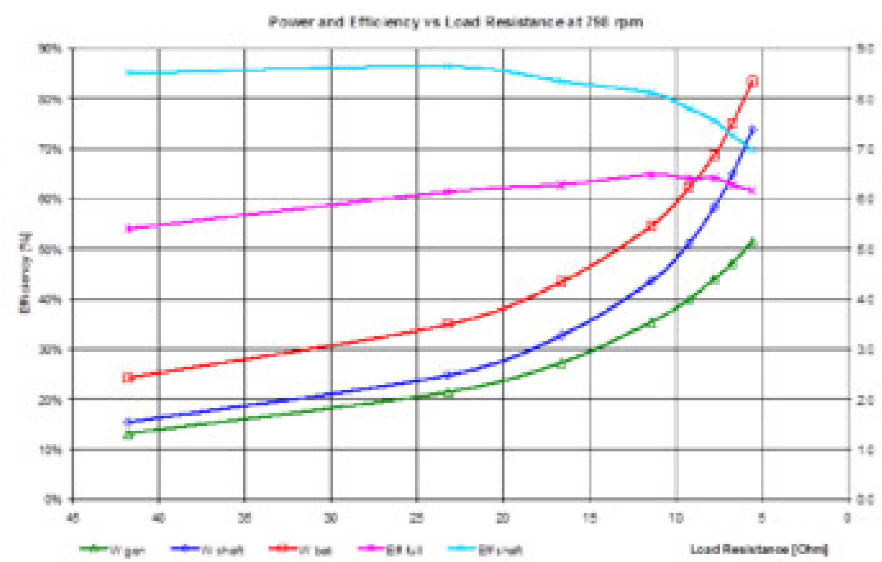
Generator Testing

- Generator mounted on the second revision test bed – measured power up to 5 kW



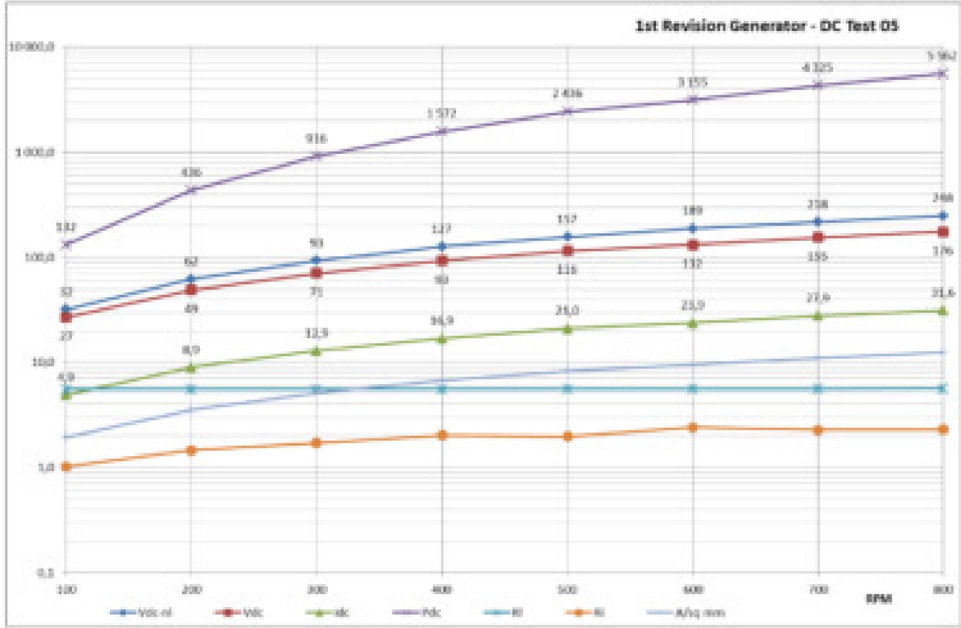


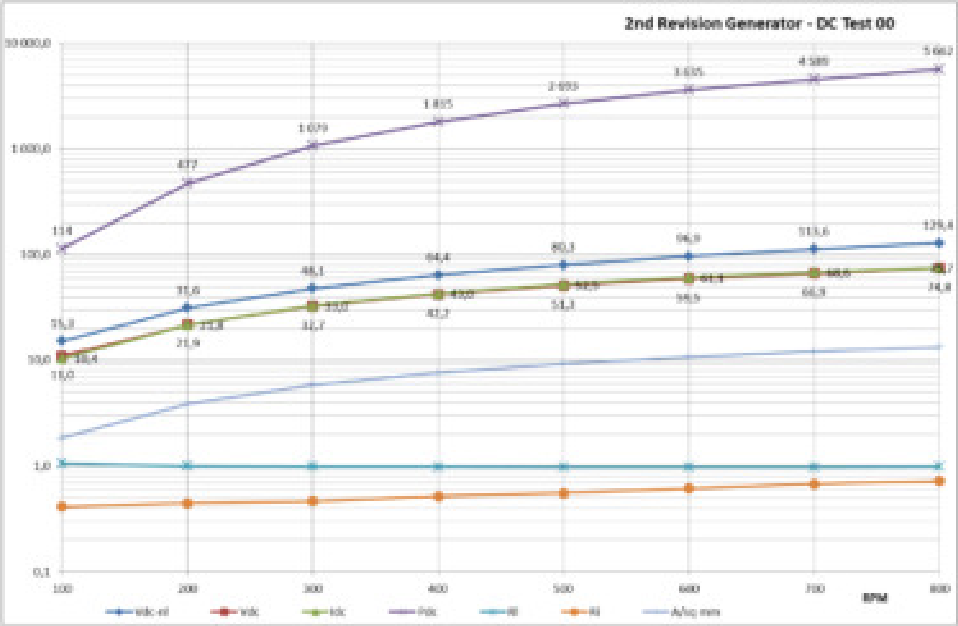


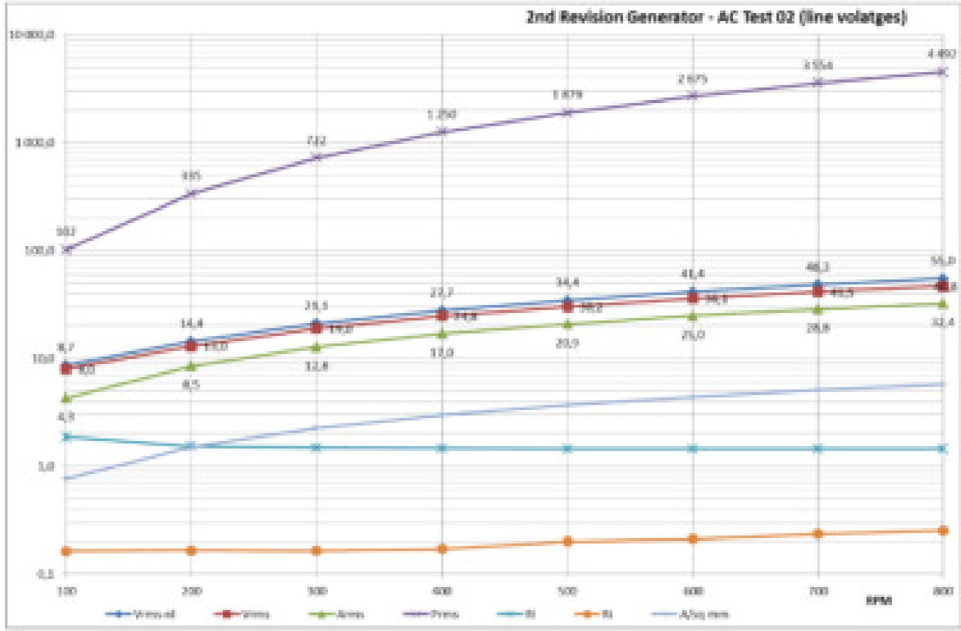


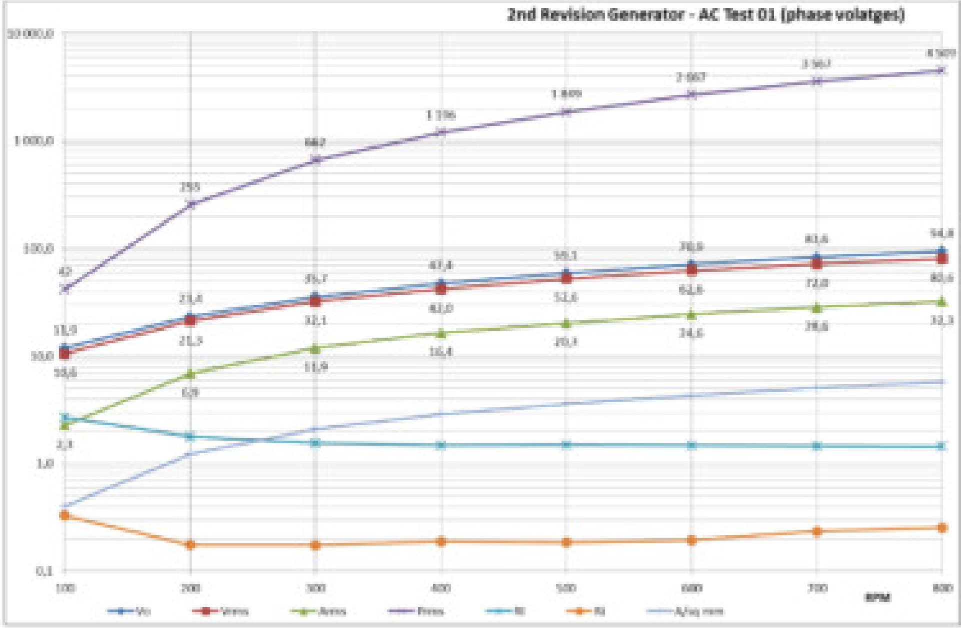
Generator Testing

- Generator mounted on the second revision test bed (AC/DC Load comparison)



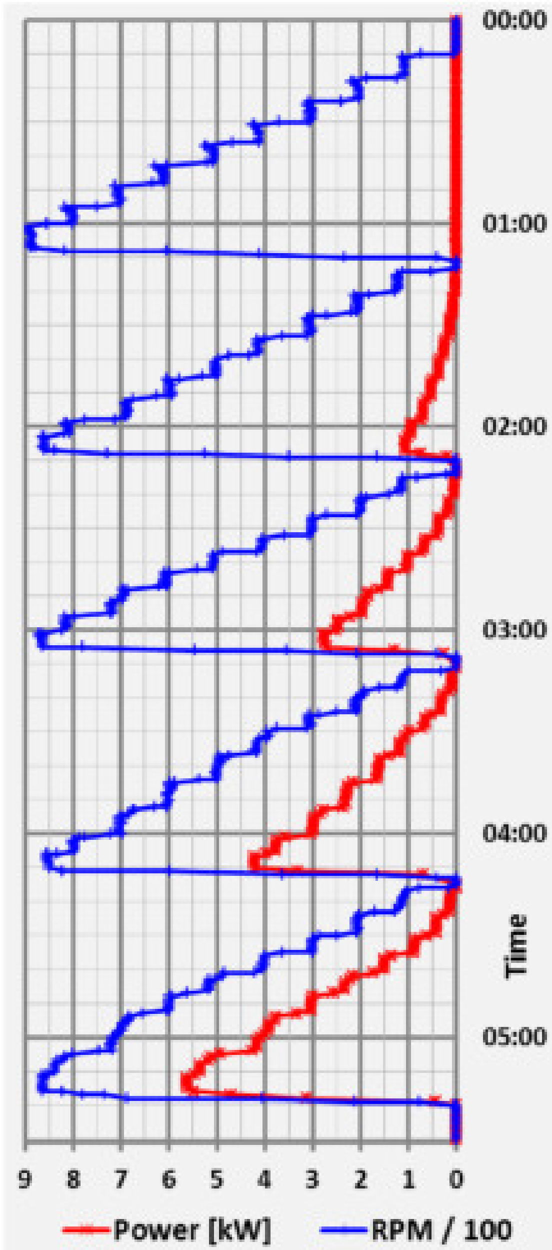


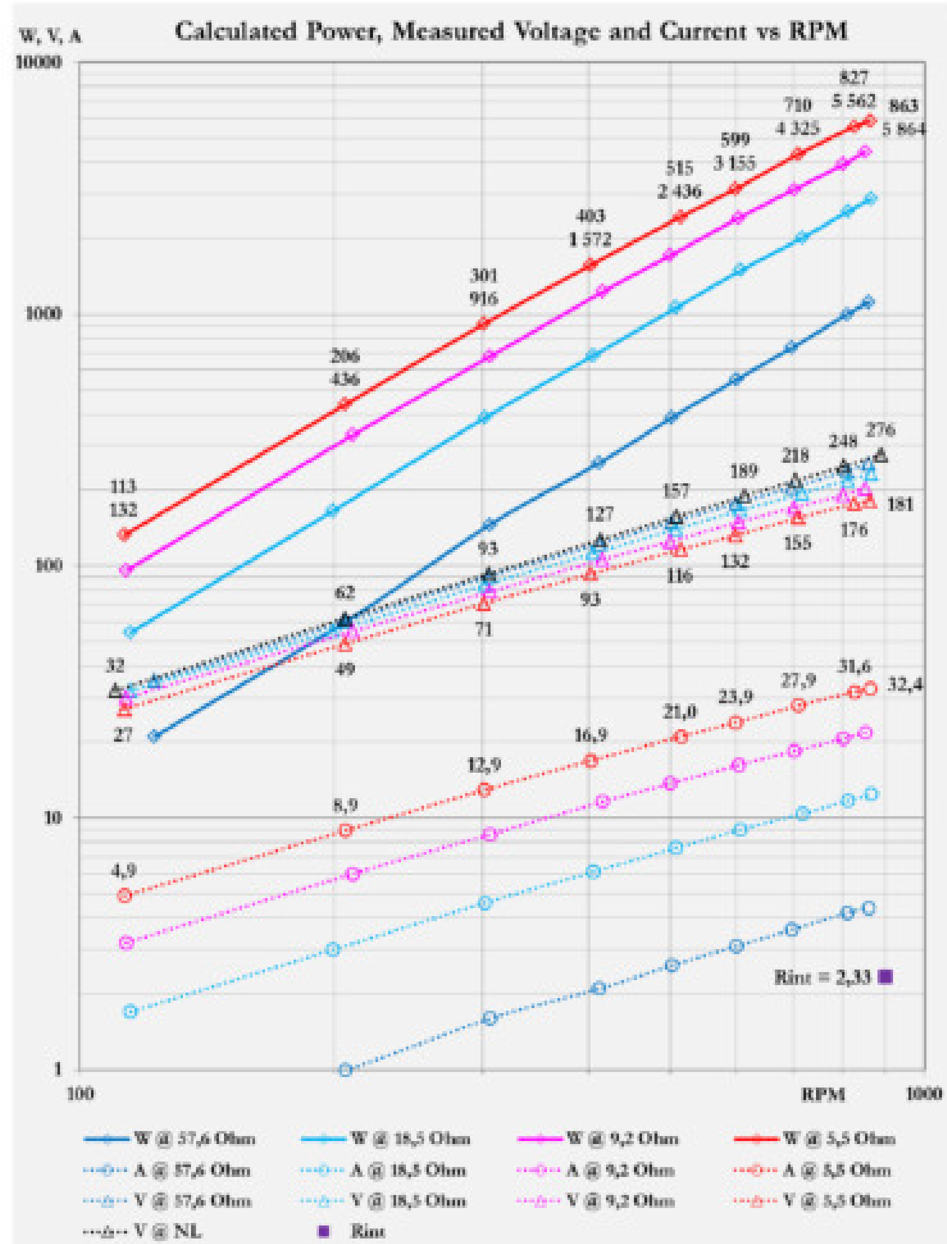




Generator Testing

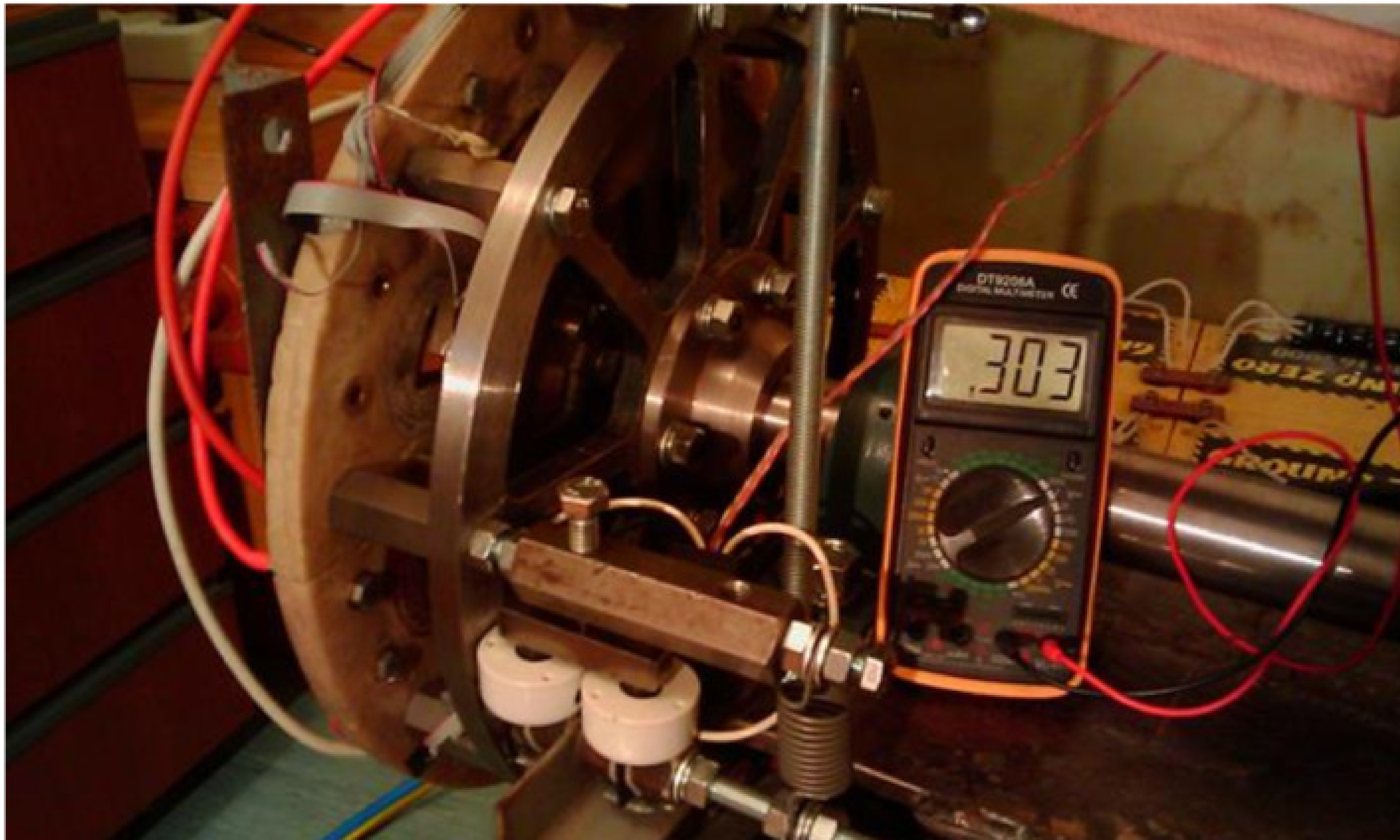
- Generator mounted on the second revision test bed (long test)





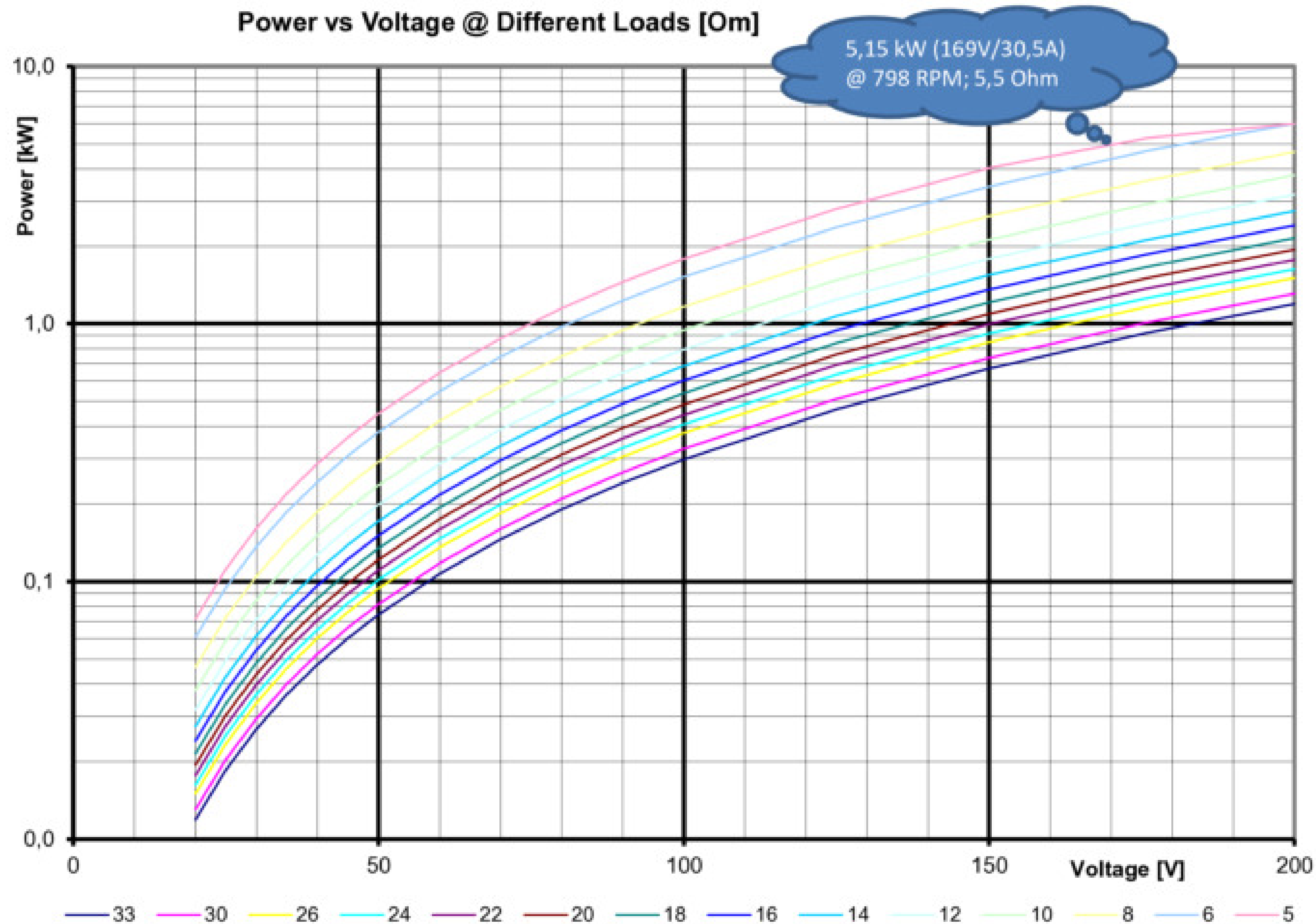
Test Bed Improvements

- Test bed – modified second revision with torque and temperature sensors



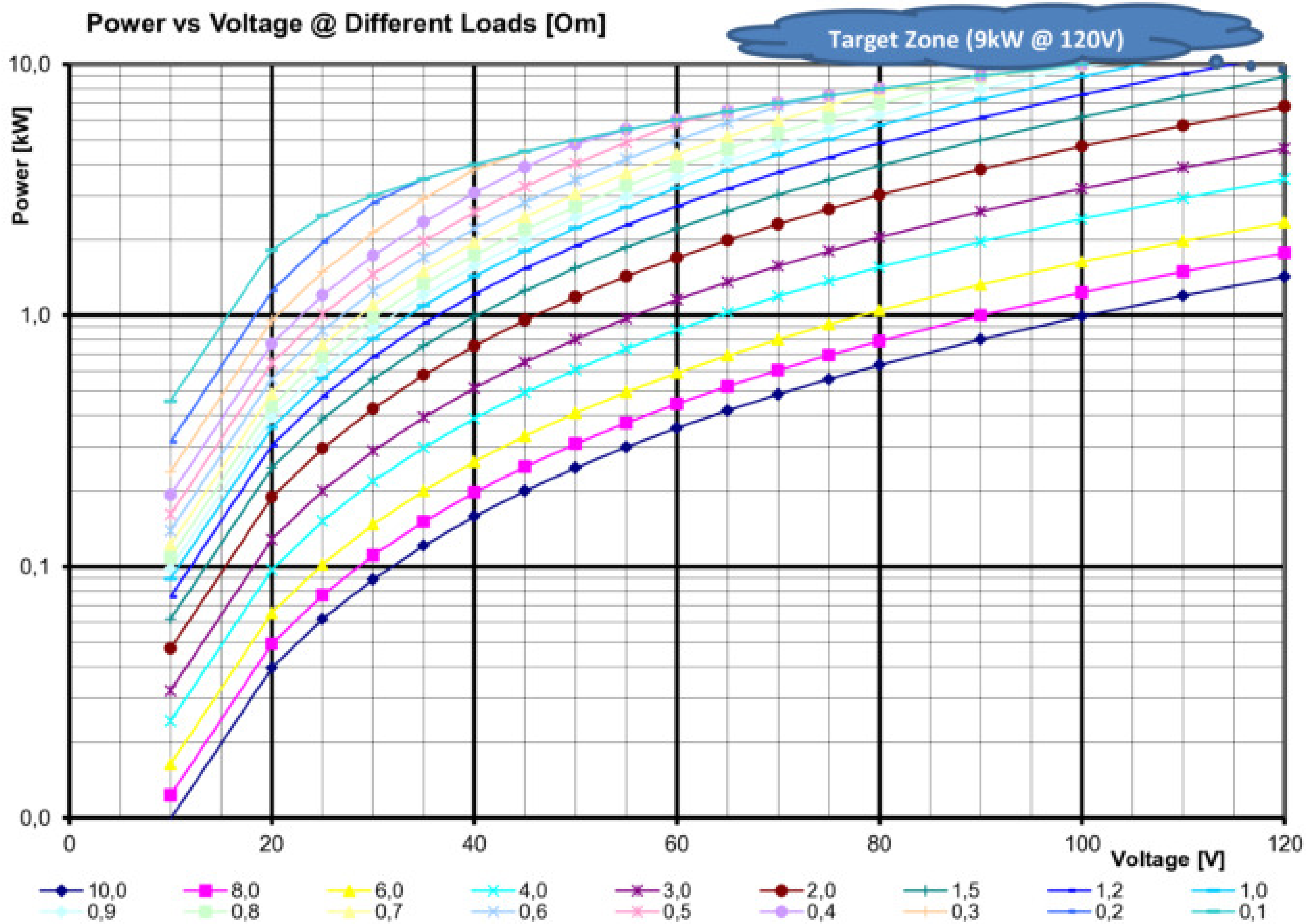
Next Generator Improvements

- ***Current Stator Base – wire size: D1.8 mm, 2.54 mm^2; coil: 120 windings***



Stator Redesign (Coil)

- ***Next Stator Base – wire size: 1.8 x 3.15 mm, 5.67 mm^2; coil: 60 windings***

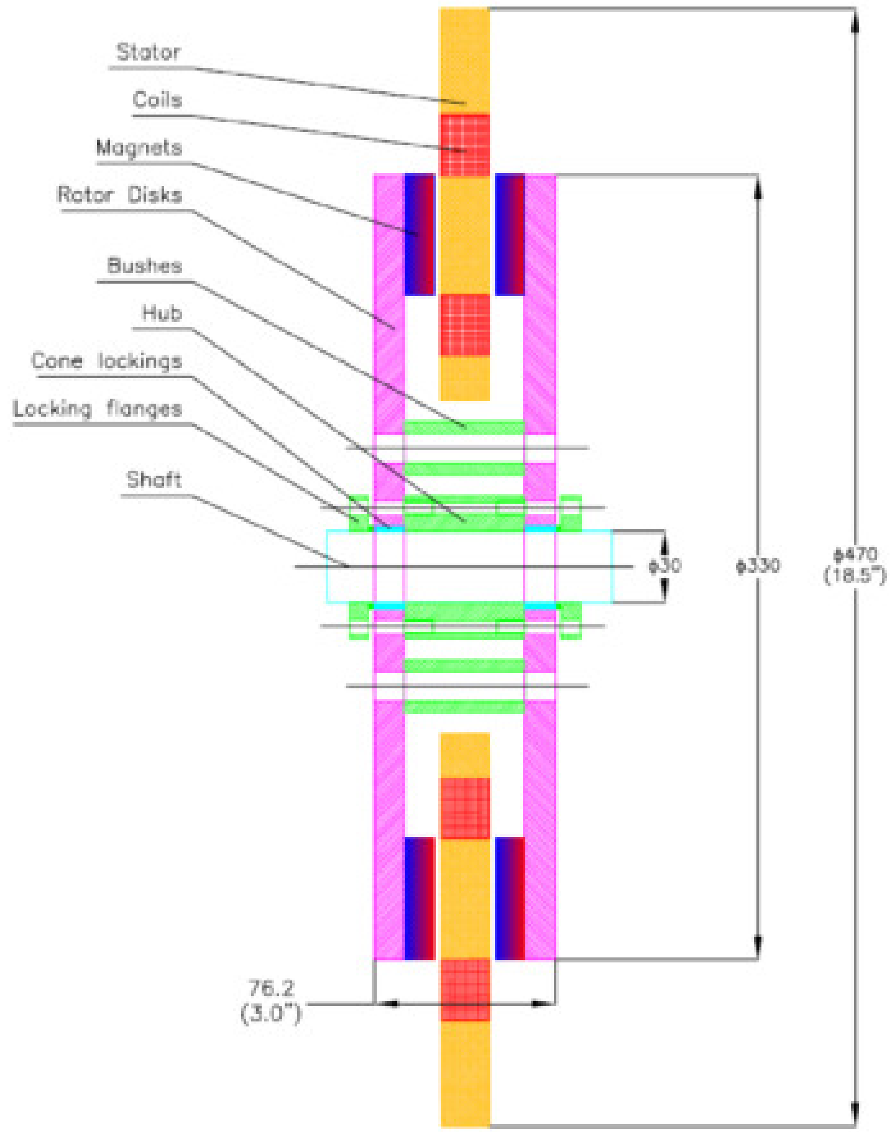


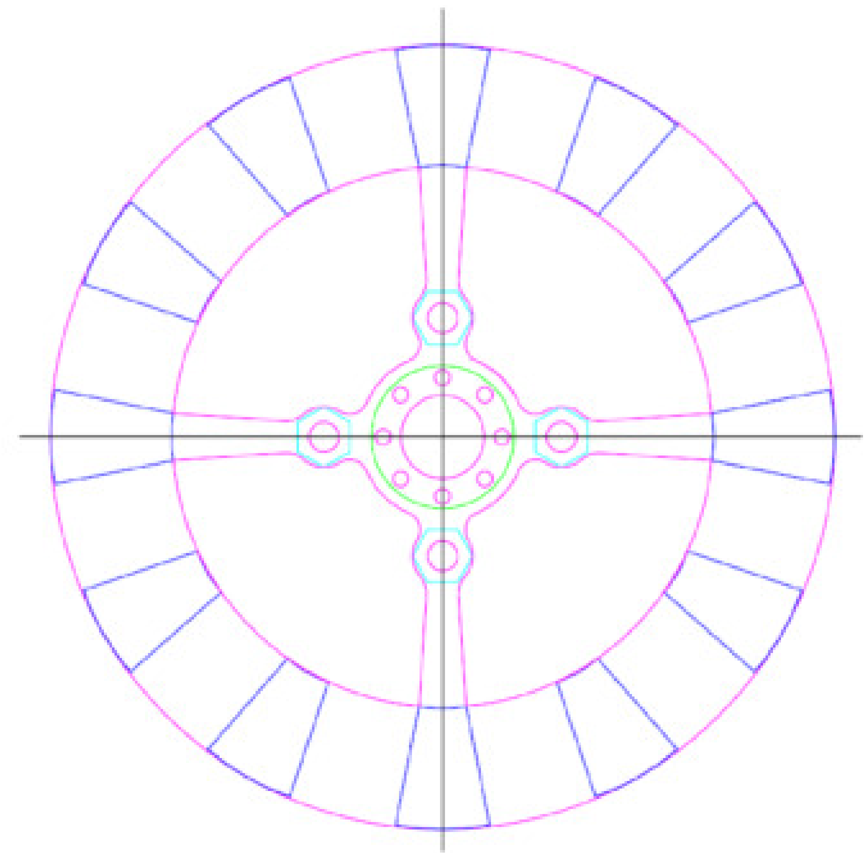
Stator Redesign (Coil)

- ***Production of the new coils***

Generator Redesign

- ***Rotor design with light hub and centered cone bushes and plotted stator***





Generator Redesign

- ***Modified to use centered cone bushes by Bea Ing. S.P.A. for mounting to the shaft***

CALETTATORI - DATI TECNICI
LOCKING ASSEMBLIES - THECNICAL DATA

Calcolo del minimo diametro esterno mozzo (DM)
Calculation of the minimum outside diameter of hub (DM)

$$DM \geq D \cdot K$$

D = diametro esterno calettatore (mm)
outside diameter of locking assemble (mm)
K = coefficiente (vedi tabella)
coefficient (see table)

Per il calcolo del valore K, non riportato in tabella applicare la seguente formula:
To calculate the "K" value not shown in the table, use the following formula:

$$K = \sqrt{\frac{\sigma\, 0{,}2 + (C \cdot PN)}{\sigma\, 0{,}2 - (C \cdot PN)}} \text{ (mm)}$$

σ 0,2 = carico di snervamento del materiale (N/mm²)
yield strength of the material (N/mm²)
C = fattore in funzione del tipo di applicazione
factor depending on the type of the application
PN = pressione superficiale del mozzo
surface pressure of the hub

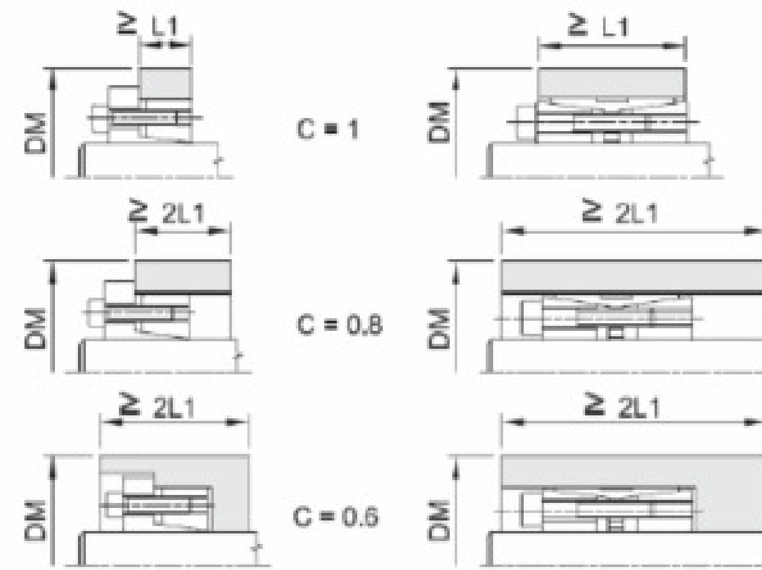


TABELLA DEL COEFFICIENTE "K" - COEFFICIENT "K" TABLE

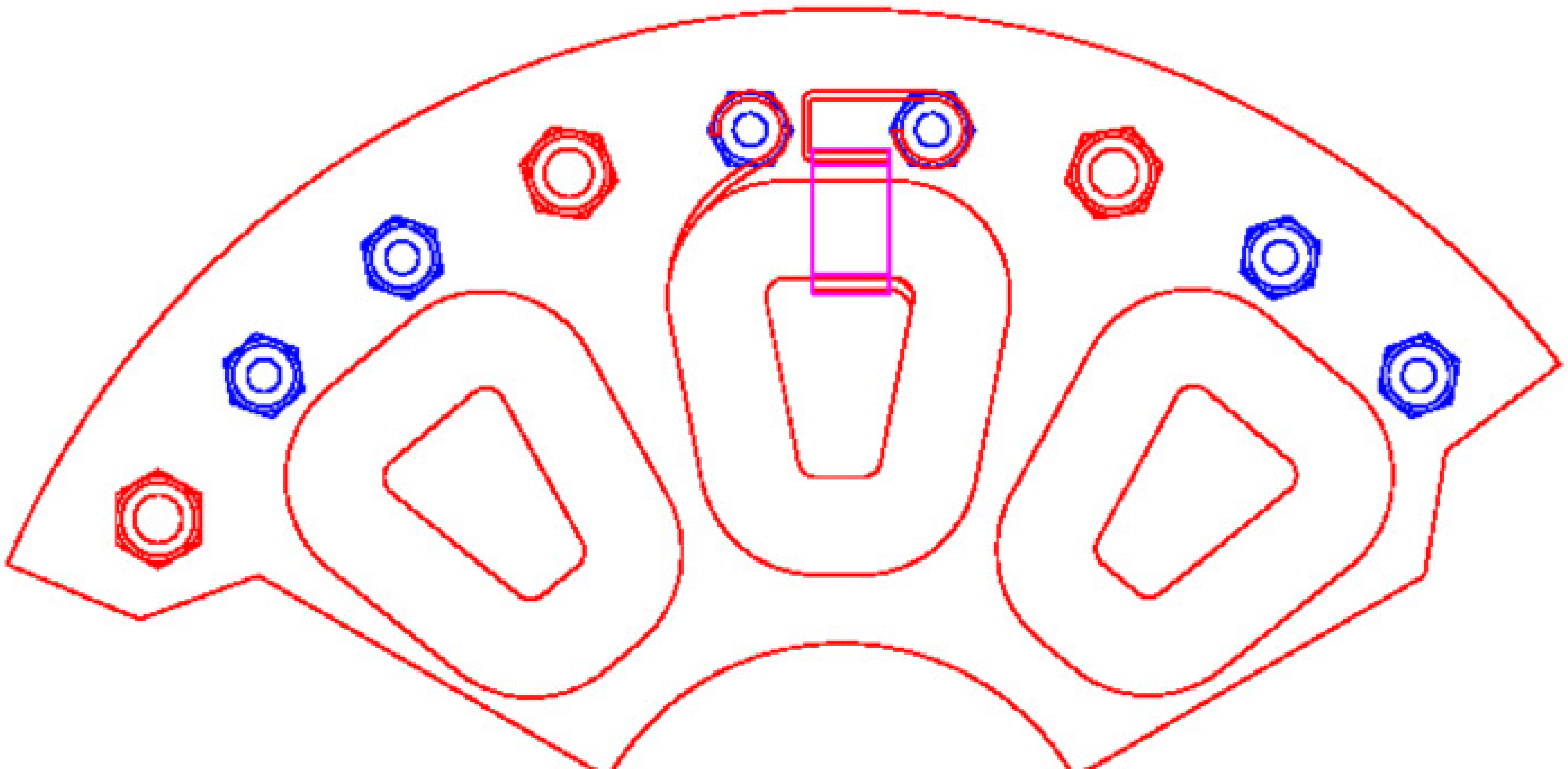
	GG-20			GG-30 GTS-35 Al.Si1MgMn			GGG-38 GS-400 St.42-3			GGG-50 GS-500 C-40			GGG-60 GS-600 C-45			GGG-70 GS-70 C-60								
	valori indicativi per il carico di snervamento σ 0,2 in N/mm²																							
	150			**200**			**250**			**300**			**350**			**400**			**450**			**600**		
pn N/mm²	0,6	0,8	1	0,6	0,8	1	0,6	0,8	1	0,6	0,8	1	0,6	0,8	1	0,6	0,8	1	0,6	0,8	1	0,6	0,8	1
80	1.39	1.58	1.81	1.28	1.39	1.53	1.21	1.30	1.39	1.18	1.24	1.31	1.15	1.20	1.26	1.13	1.18	1.22	1.11	1.15	1.20	1.08	1.11	1.14
85	1.42	4.63	1.90	1.30	1.42	1.57	1.23	1.32	1.42	1.19	1.26	1.34	1.16	1.22	1.28	1.14	1.19	1.24	1.12	1.16	1.21	1.09	1.12	1.15
90	1.46	1.69	2.00	1.32	1.46	1.62	1.25	1.34	1.46	1.20	1.28	1.36	1.17	1.23	1.30	1.15	1.20	1.26	1.13	1.18	1.22	1.09	1.13	1.16
95	1.49	1.75	2.11	1.34	1.49	1.68	1.26	1.37	1.49	1.21	1.30	1.39	1.18	1.25	1.32	1.15	1.21	1.27	1.14	1.19	1.24	1.10	1.14	1.17
100	1.53	1.81	2.24	1.36	1.53	1.73	1.28	1.39	1.53	1.22	1.31	1.41	1.19	1.26	1.34	1.16	1.22	1.29	1.14	1.20	1.25	1.11	1.14	1.18

Stator Redesign (Segment)

- ***Stator template for molding technology***

Stator Redesign (Segment)

- ***New stator geometry for plotting technology***



Stator Redesign (All segments)

- ***New stator after plotting and ready for coil mounting***

Stator Redesign (Complete staff)

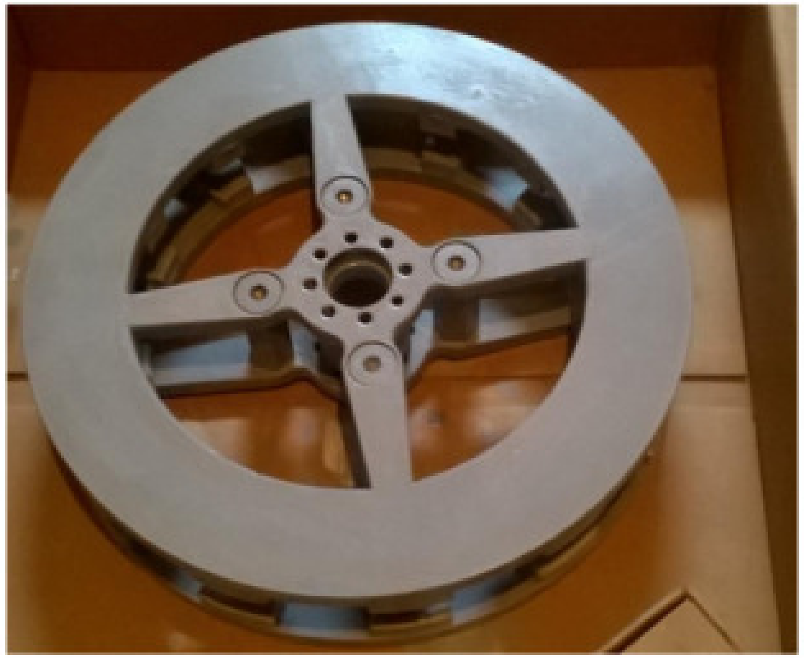
- ***New stator produced by plotting technology and Cu plate wires***

Rotor Redesign (Assembled)

- ***Assembled rotor with light hub, spacers and centered cone bushes***

Rotor Redesign (Painted)

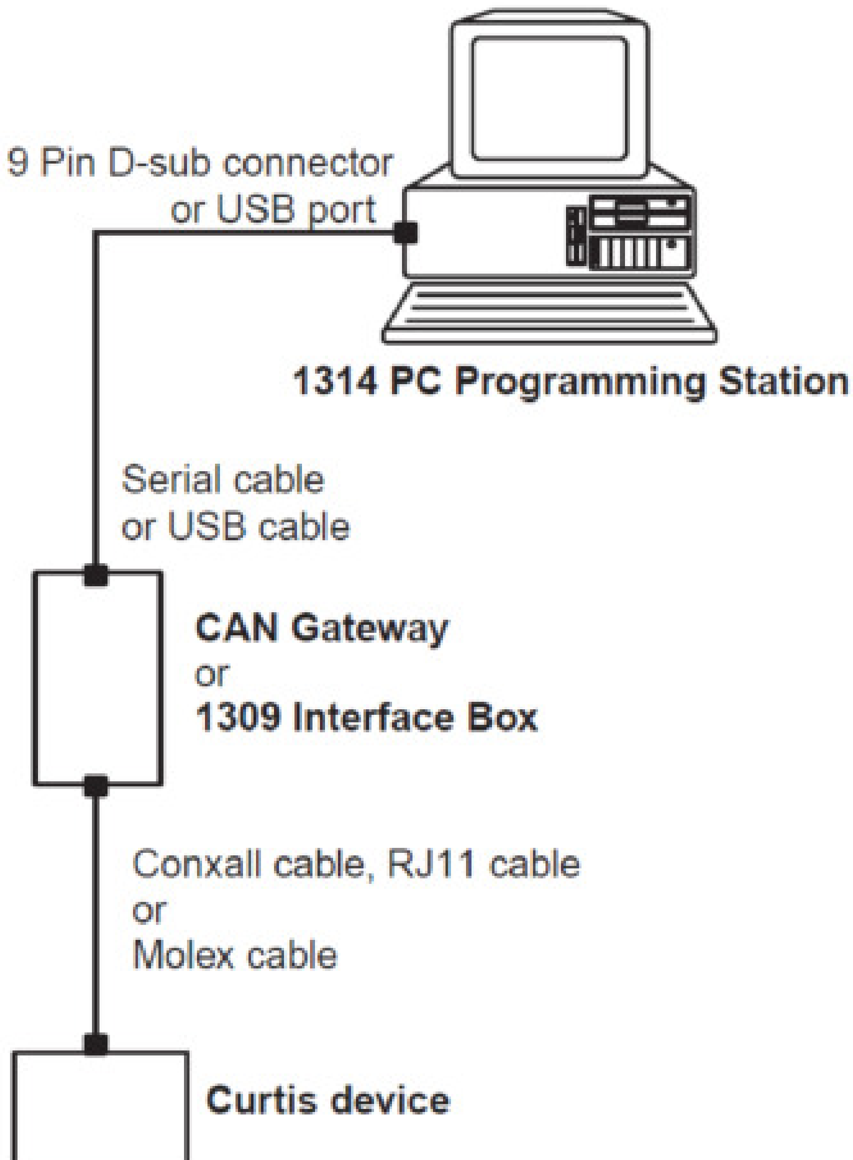
- ***Assembled and painted rotor, stator carrier disk both mounted on the field***

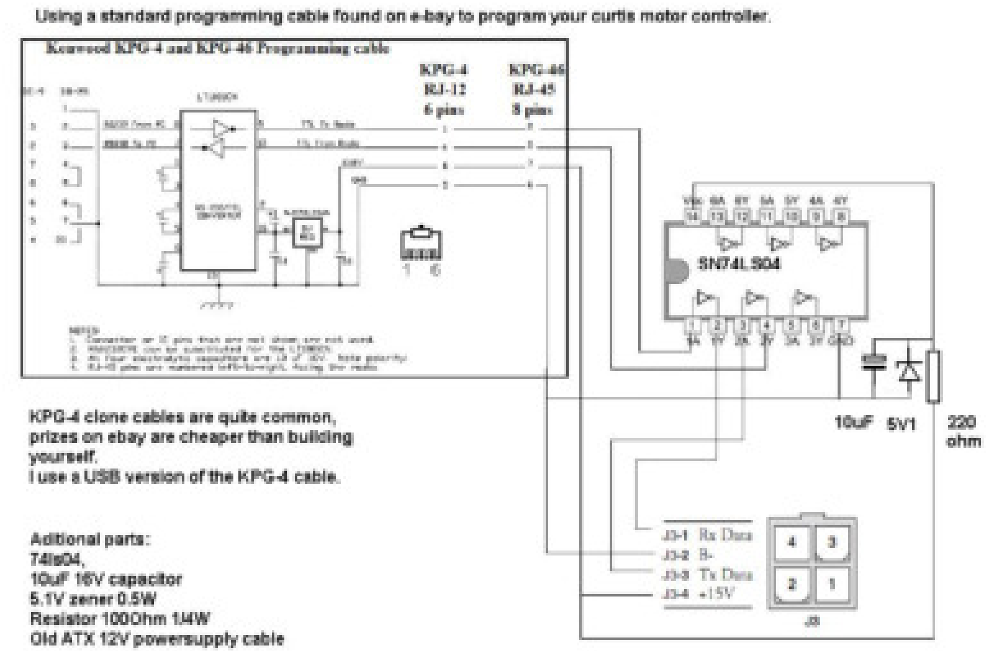



The field test suite (Shabla)

Electrical System Improvement

- Test bed – rev. 3 with PC Station interconnection and Cooler for Curtis' Controller



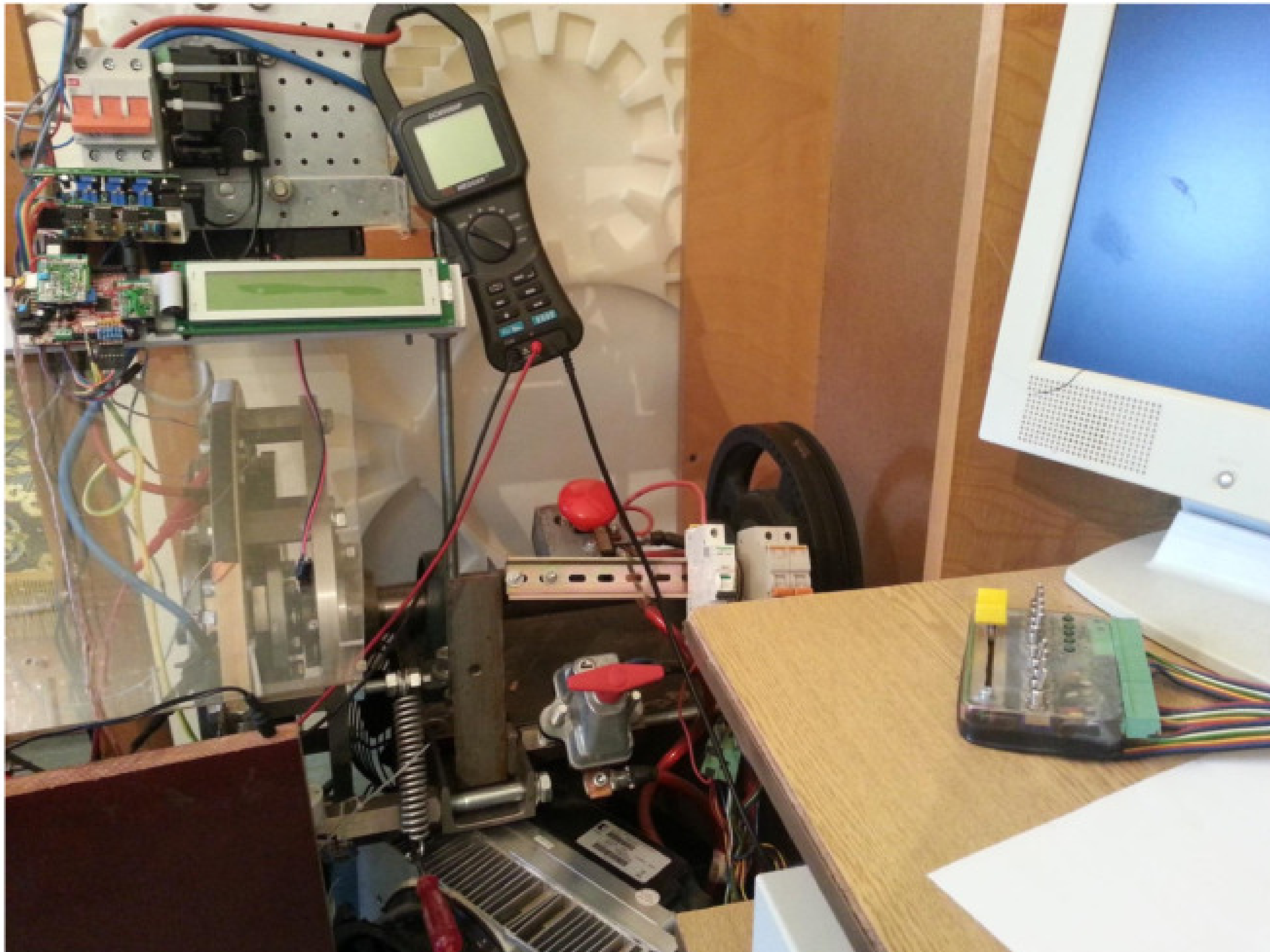


Electrical System Improvement

- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 1-st rev.

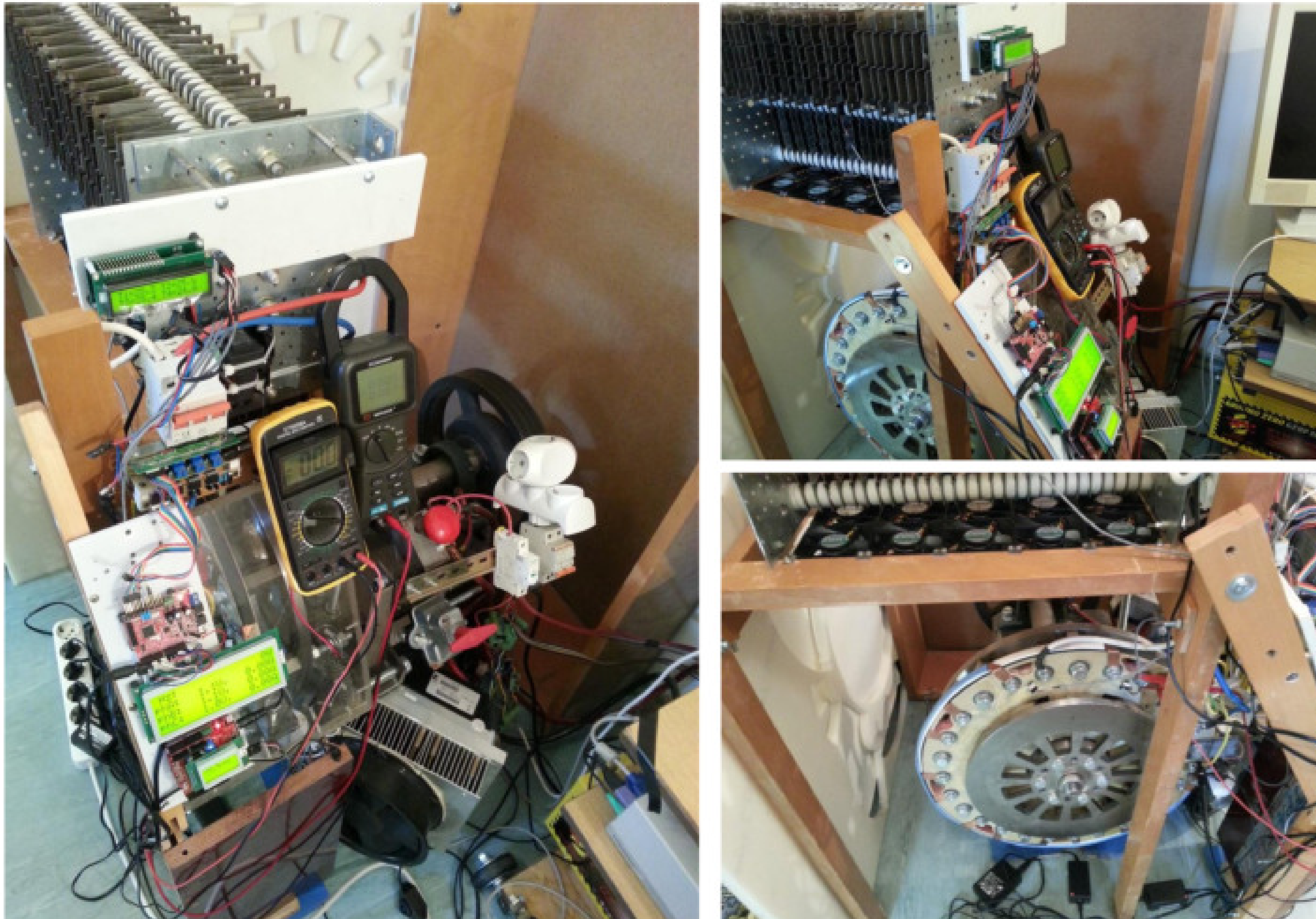
Electrical System Improvement

- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 1-st rev.



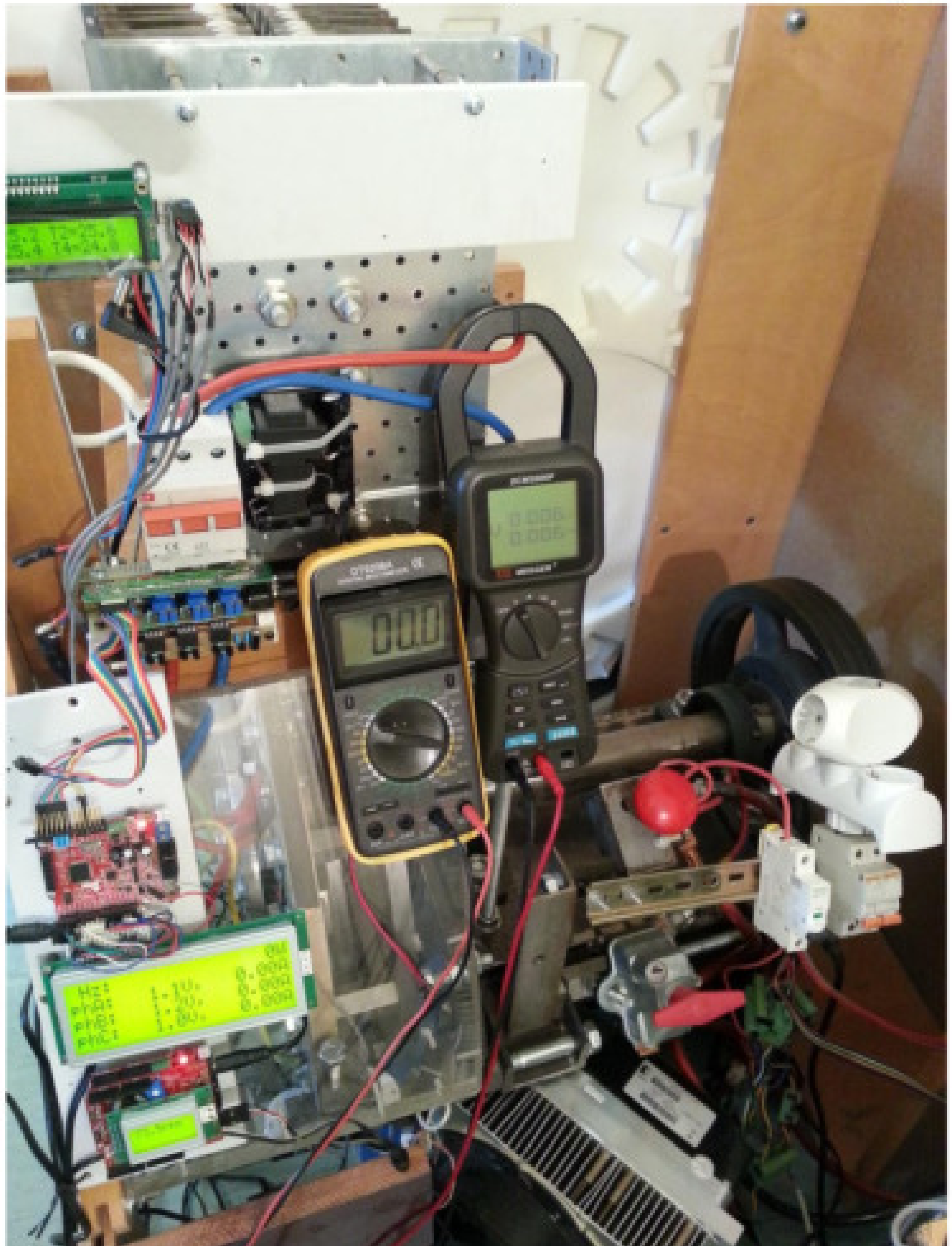
Electrical System Improvement

- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 2-nd rev.



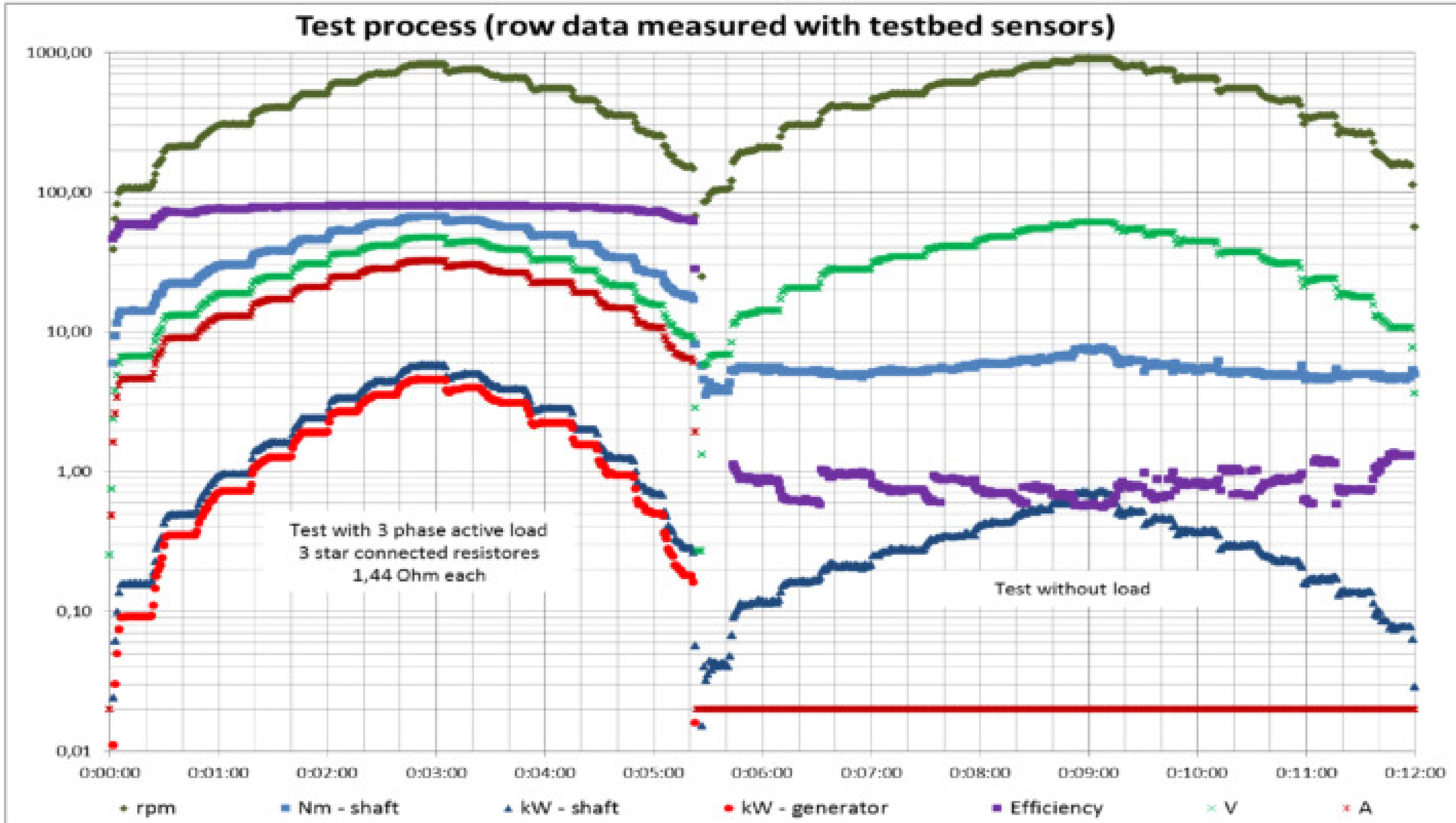
Electrical System Improvement

- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 2-nd rev.



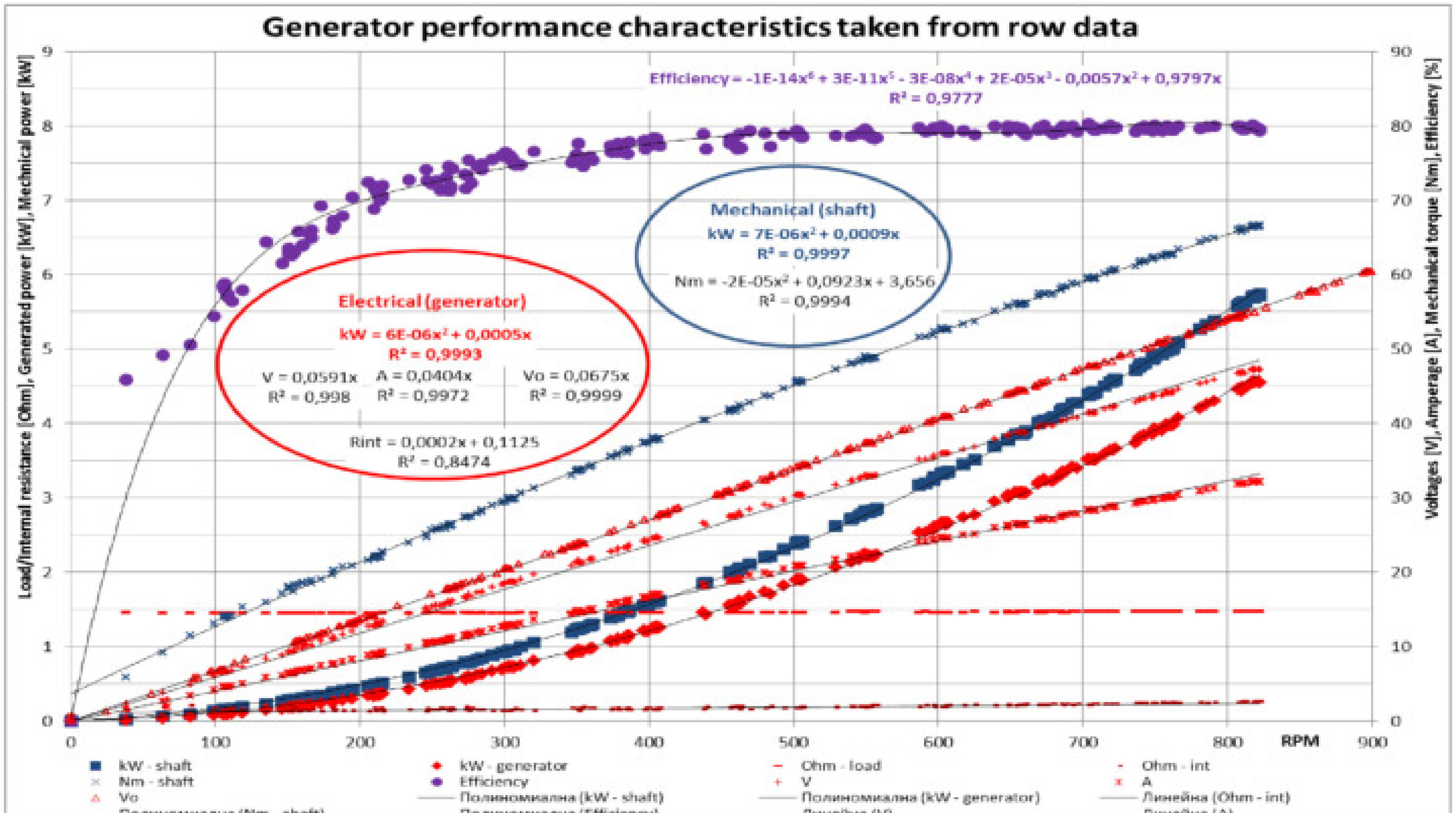
Measurement Results

- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 2-nd rev.



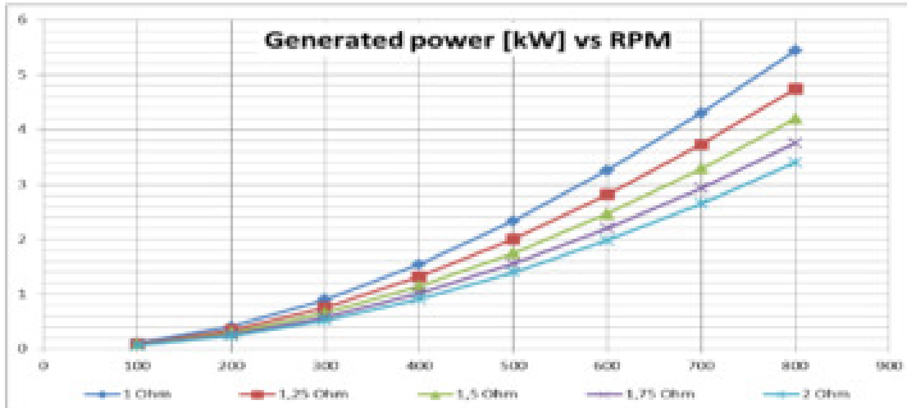
Measurement Results

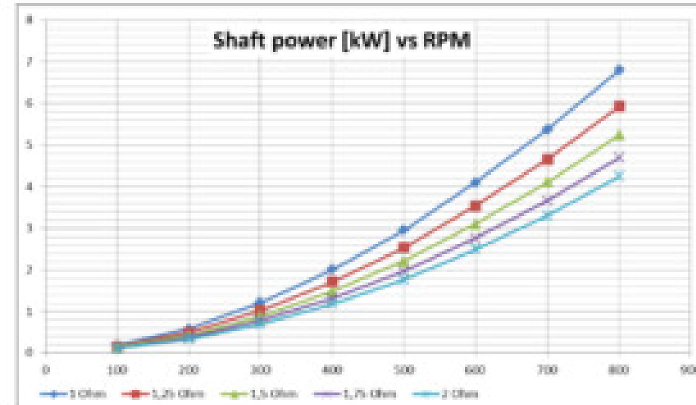
- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 2-nd rev.

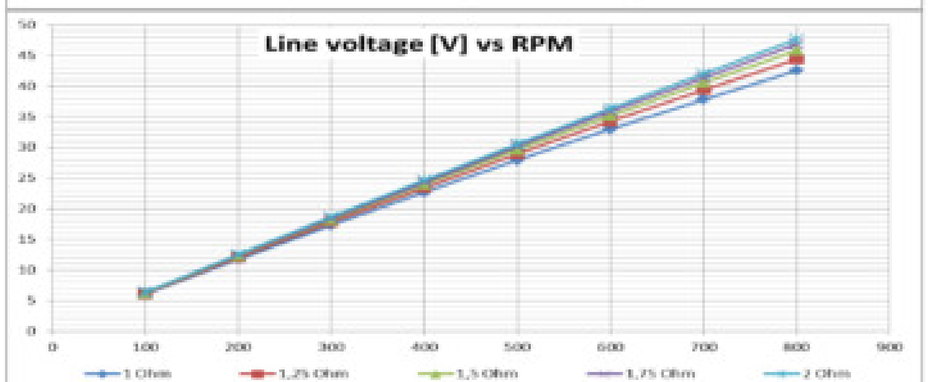


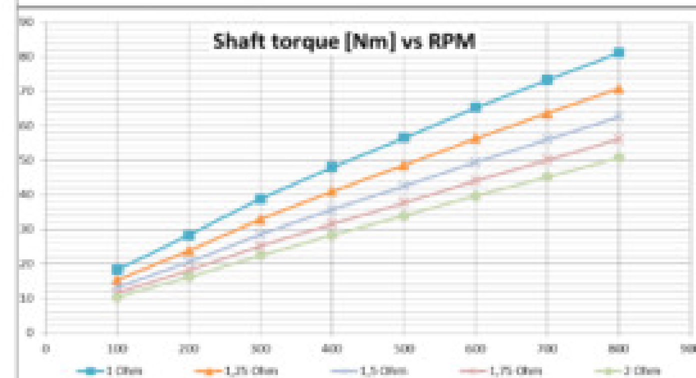
Measurement Results

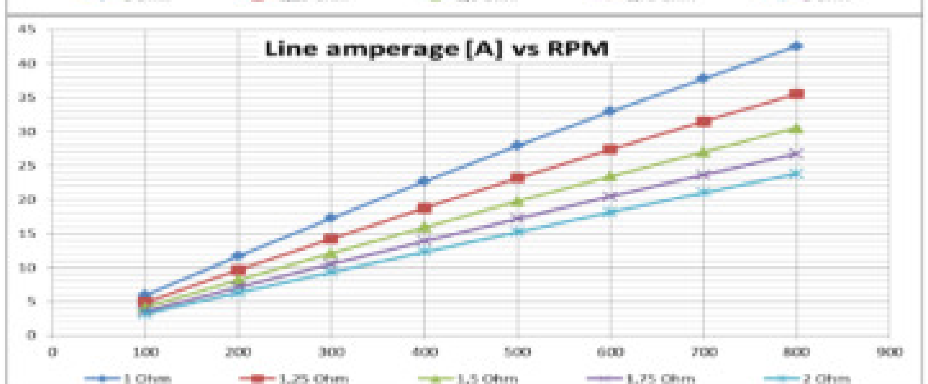
- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 2-nd rev.

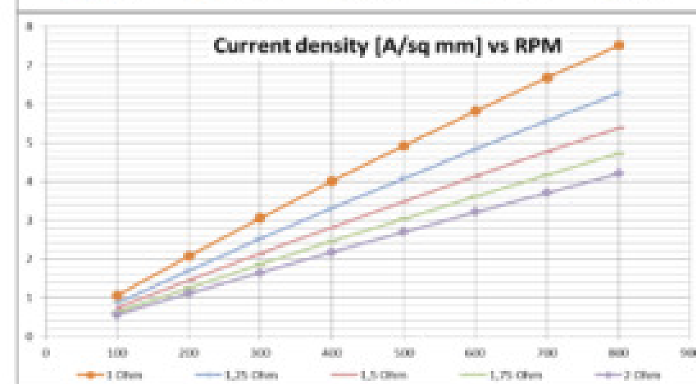






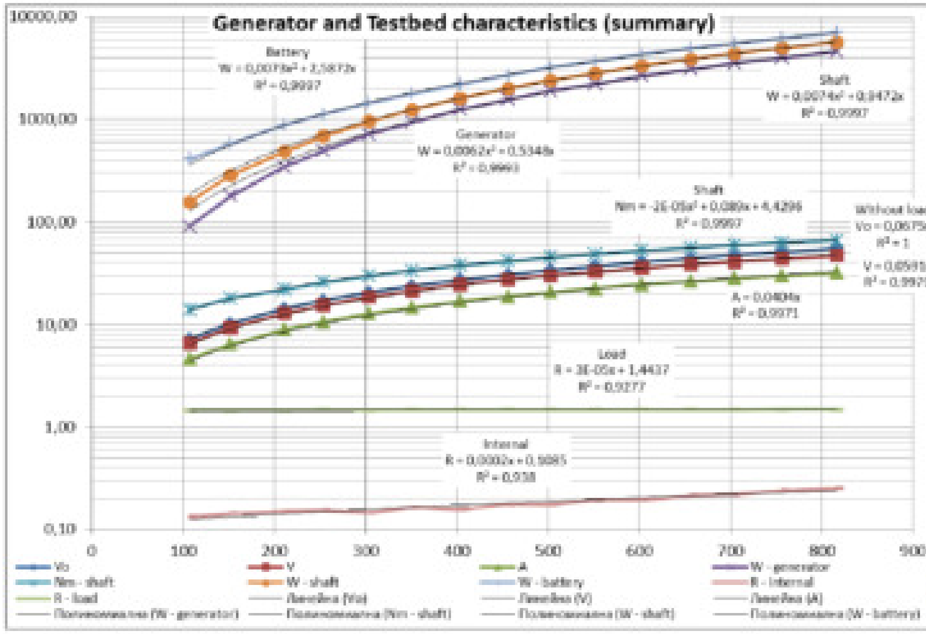


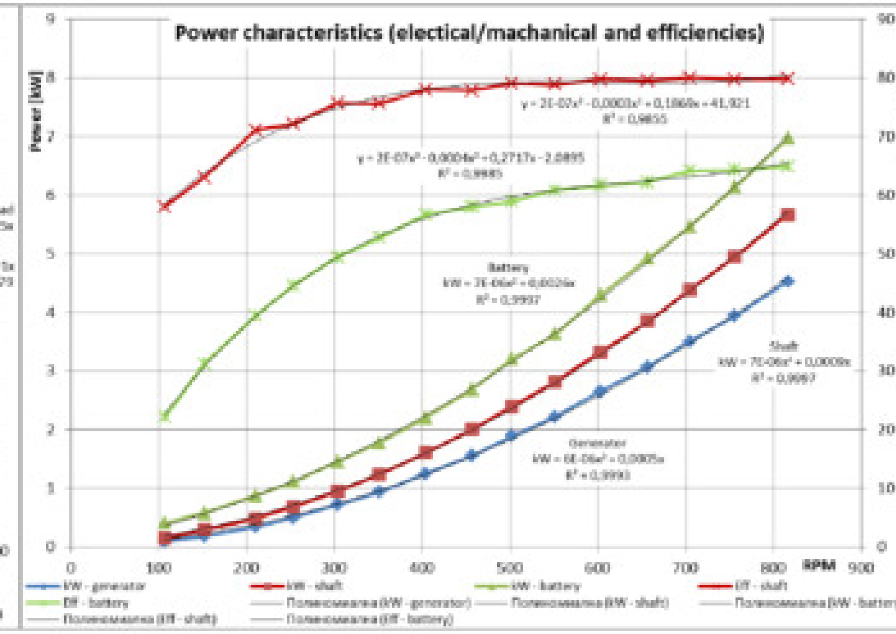


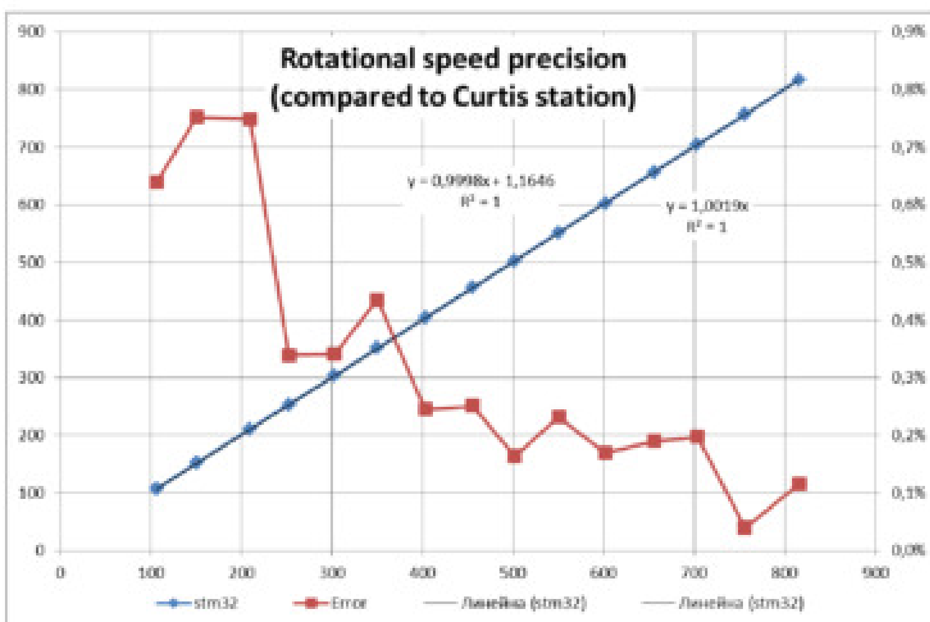


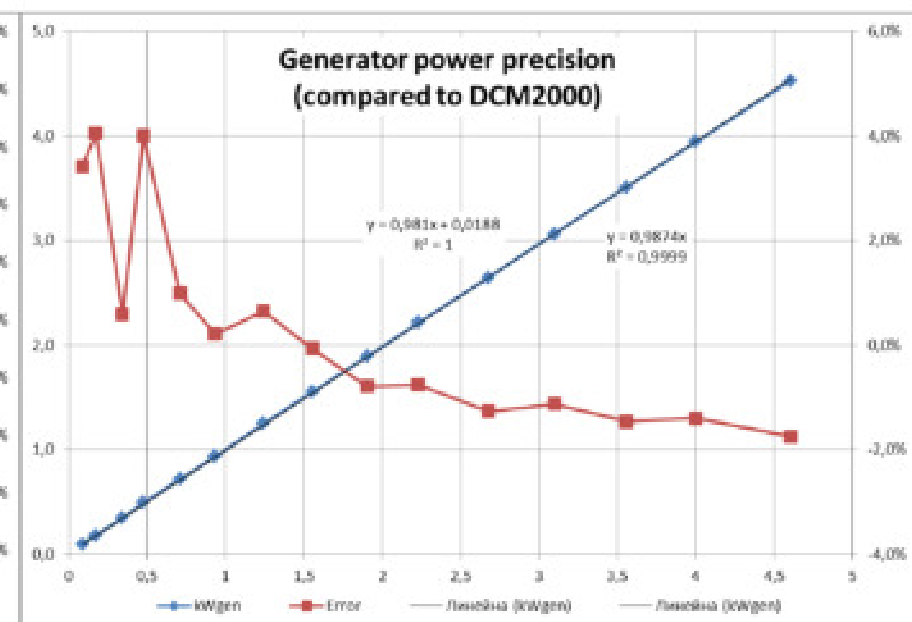
Measurement Results

- Generator 2-nd rev., Test bed 3-rd rev., 3-phase load and data acquisition 2-nd rev.







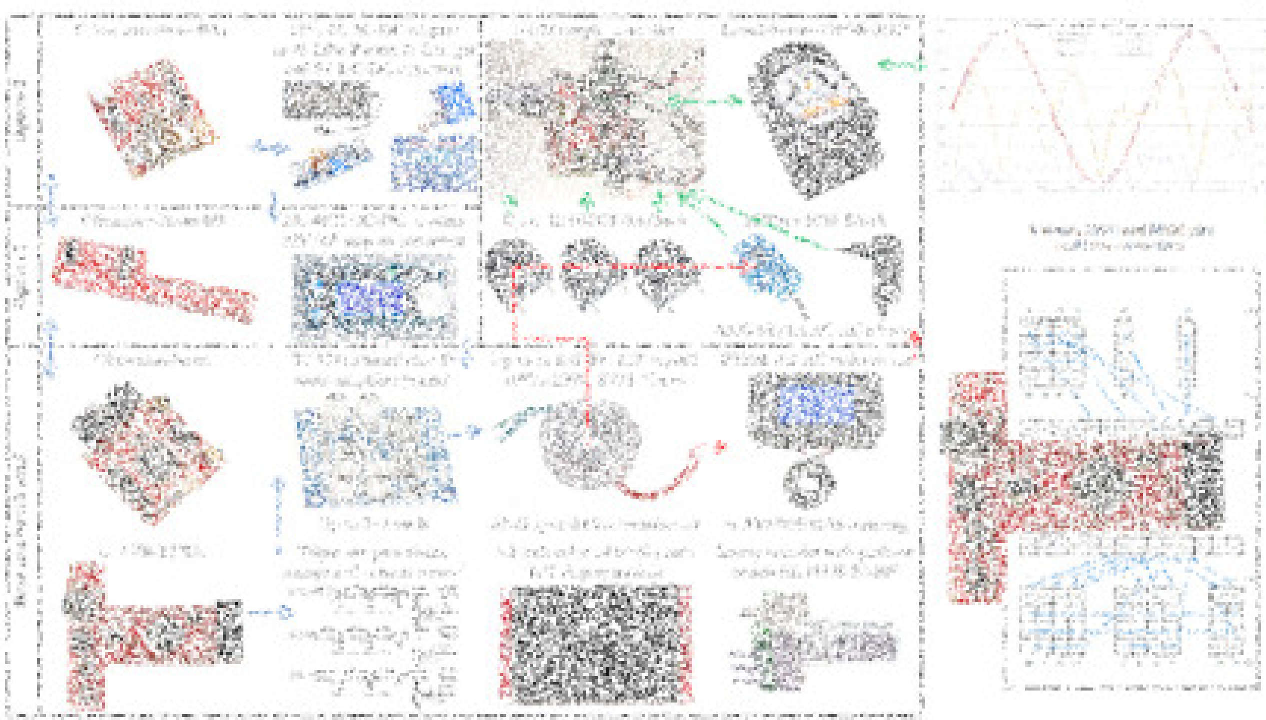


5kW Wind Generator, test bench and measured data (2013)

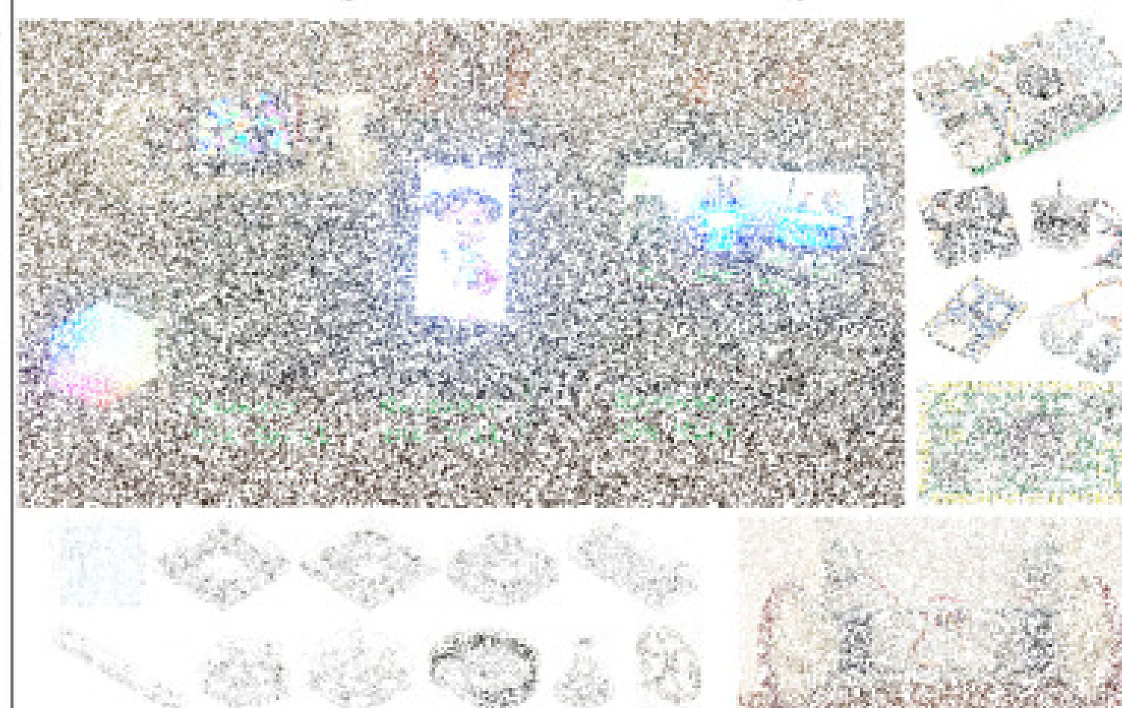
Unified Multicore Low Power IoT Platform (2023)

Full color PicoDVI test on Raspberry Pi Pico, RP2040-PICO-PC and 5" TFT HDMI 800x480 display

Multichannel Energy Metering system and test bench (2023)



Adroid - the open S.T.E.A.M. robot platform (2024)



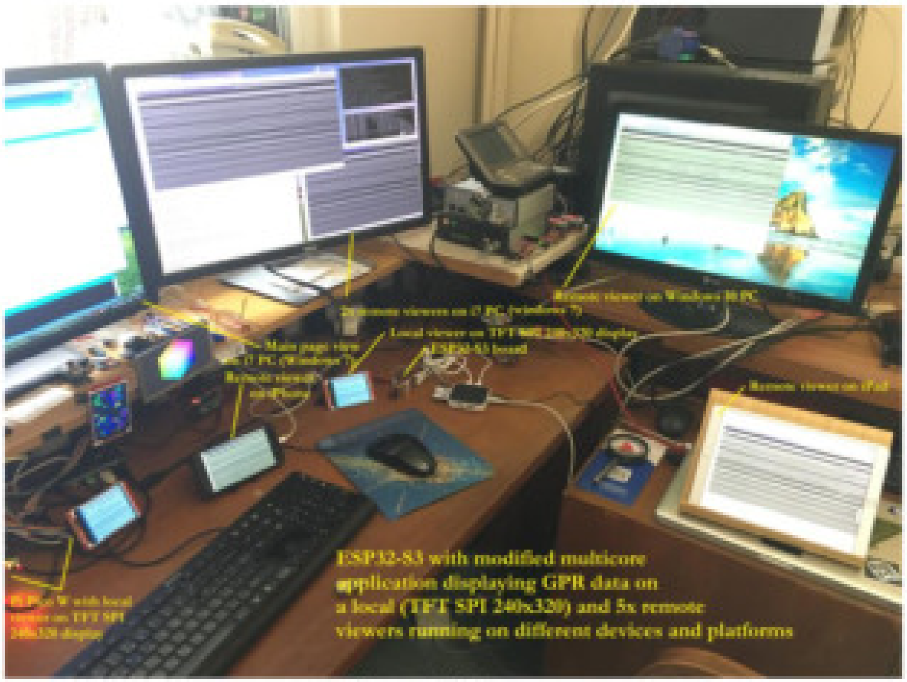
Modified multicore application as the proof-of-concept for local and remote visualization of GPR data via ESP32-S3 board with TFT SPI 240x320 display

The modified multicore application is modification of the unified multicore application for ESP32 and RP2040 with 3.2" TFT SPI 240x320 display to run on RP2040, ESP32-WROOM and ESP32-S3-WROOM based boards. The main bug (impossibility to display and edit SPIFFS files in the file browser in case of compiling application against Espressif ESP32 core 2.0 or later) is fixed by own code instead of using SPIFFSEditor library component (part of the core ESPAsyncWebServer library). SPIFFS structure is changed to separate private and public files. The development process was eased thanks to using USB OTG JTAG/Serial (USBSerial) and UART0 (Serial) interfaces for uploading the program and printing of debug messages respectively. Serial ports on UART1 (Serial1) and UART2 (Serial2) are used to connect GPR and GPS devices.

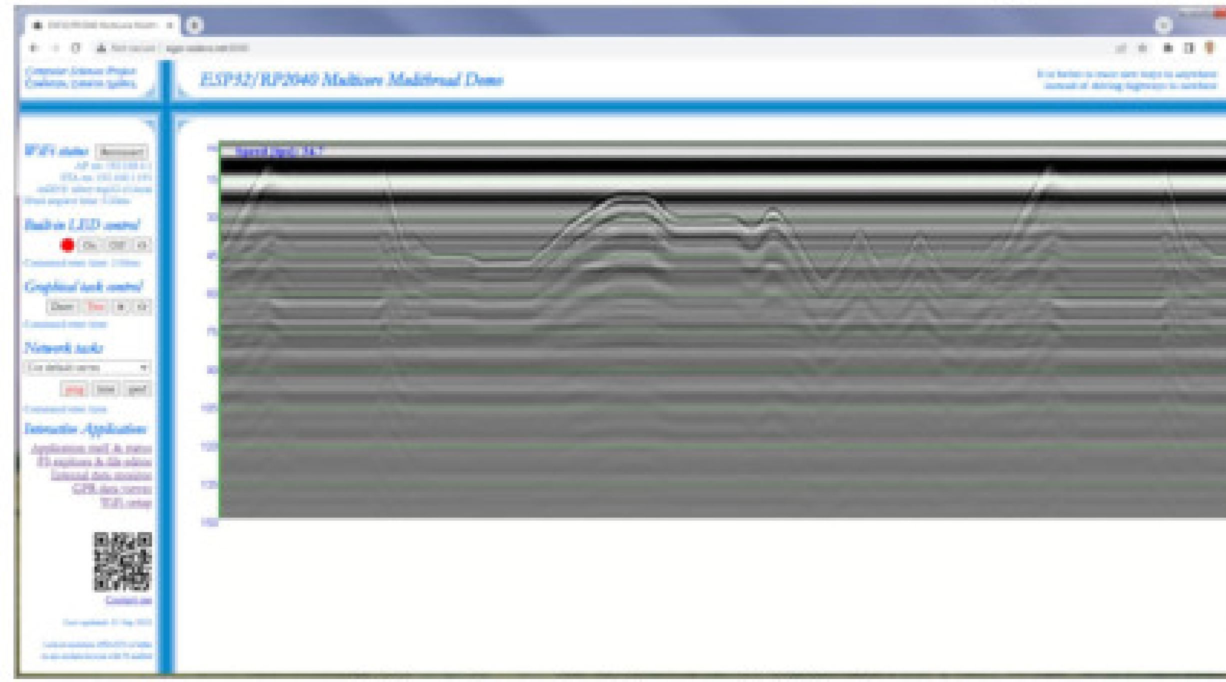
It is also added a code to read GPR data via serial port and display them on local TFT display. A code based on AsyncEventSource is also added to send GPR data to all registered remote clients. Demo version of remote GPR data viewer is developed as web application to prove the concept. As it is visible from the photo below 5x remote clients are served in addition to the local display. There is no disturbance in any of the served local and remote viewers at more than 30 tps (tracks per second). For the moment GPR data are sourced by written in JavaScript simulator reading them from a SEGY file. At the first tests application can run on all Pi Pico W, ESP32-WROOM and ESP32-s3-WROOM based boards but only on ESP32-s3-WROOM one it is working without problems and stable enough.

In the final application for GPR data visualization some of the components in current application will be removed or modified and others will be added. The control of the GPR device by the local and remote viewers is under discussion. In case of remote control of the GPR device the concurrence may cause problems so it has to be assessed its usefulness. There are following alternatives for the local control: touch screen, rotational encoder or buttons. It is cleaner remote viewers to control visualization only. Saving of GPR data to file locally and/or remotely has to be discusses as well.

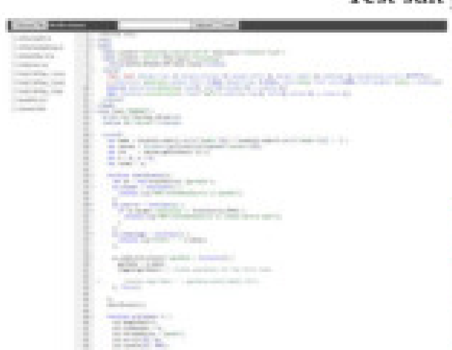
Modified multicore application in pictures



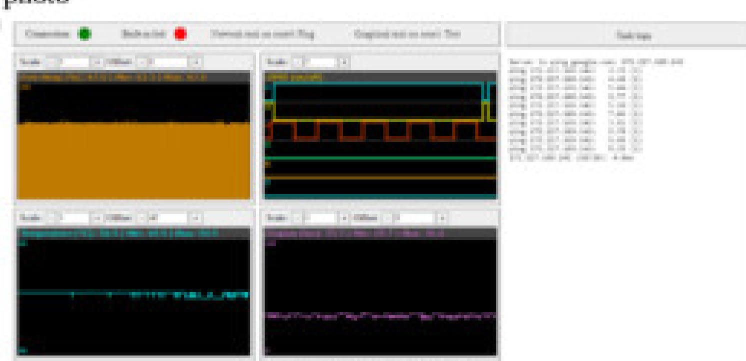
Test suit photo



Main web page view with remote GPR data viewer

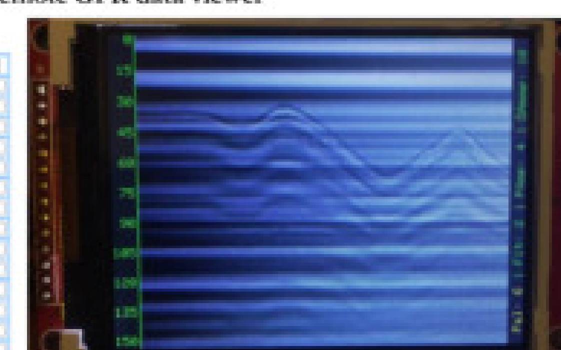


SPIFFS browser



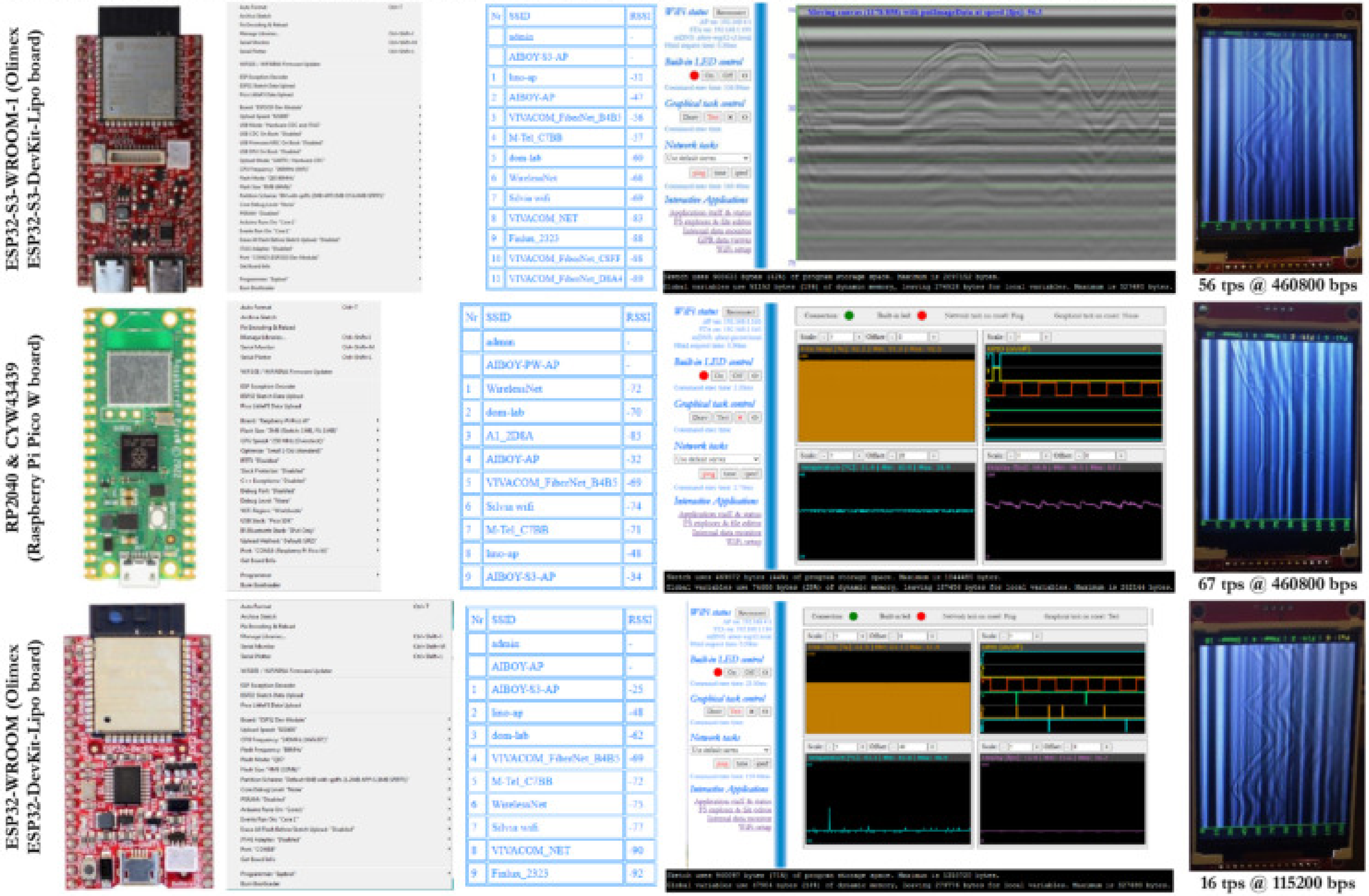
Internal data monitor

WiFi setup view



TFT SPI display with GPR data

Modified multicore application for ESP32-S3-WROOM-1, RP2040 & CYW43439 and ESP32-WROOM with 3.2" TFT SPI 240x320 display – summary in pictures



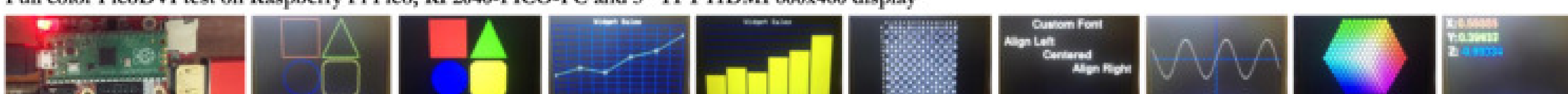
Unified multicore application for ESP32, RP2040 with 3.2" TFT SPI 240x320 display – summary in pictures and projects history

Projects history

- First cycle of tests included unified graphics test running on Arduino UNO (ATMega328), Arduino Leonardo (ATMega32u4), Arduino D1 R32 / ESP32, Raspberry Pi Pico W (RP2040) and Self-made AVR128db48 boards connected to 3.2" TFT SPI 240x320 display. Application is based on Adafruit performance tests for Adafruit_ILI9341 / Adafruit_GFX and adapted for TFT_ILI9341 and TFT_eSPI libraries. Meanwhile Olimex ESP32-S2 (WROOM and WROVER) boards were tested with multitasking "Hello world & RGB LED". Another test done is based on ESP32-CAM module as a base of own implementation of Wifi Camera Robot Car project. As a result following open source projects are posted on GitHub:
  - Unified-ILI9341-Graphic-Test
  - Unified-ILI9341-Graphic-Test-plus
- Next cycle of tests was performance assessment of networking capabilities of WiFi equipped ESP32 and Pi Pico W boards. Test applications are based on Arduino libraries ESPAsyncWebServer and AsyncWebServer_RP2040W. Special attention was paid to asynchronous web services, web sockets, WiFi management and unification possibility for both ESP32 and Pi Pico W platforms.
- Next step done was to adapt DrawWithDMA sketch created by Bodmer as example for TFT_eSPI library to work on ESP-WROOM-32 and RP2040 boards. Modified sketch is posted on GitHub as open source:
  - DrawWithDMA
- Next cycle of tests was directed to multicore task execution on dual core versions of ESP32 and RP2040 based boards. United version of AsyncFSBrowser demo with Unified graphic test (TFT-eSPI library case) and modified DrawWithDMA sketch was implemented as multi-file Arduino IDE project running on both Arduino D1 R32 ESP32 (ESP32-WROOM) and Raspberry Pi Pico W (RP2040) boards. It includes web server with web sockets service, TCP server for network performance assessment, internal SPI Flash FS file viewer and editor, monitor showing graphs of the free heap memory, the GPIO states, the internal temperature and the animation frame rate. It also includes accounts management of WiFi in AP and/or STA modes. Graphics part of the application is implemented as tasks alternatively running on second CPU core. Control is based on web sockets and includes built-in LED, running of network tasks like ping, time, iperf and switching of graphic tasks (Adafruit tests and DrawWithDMA animation). Results of network commands and Adafruit tests are printed on monitoring web page. Unified multicore application will be posted on GitHub as soon as become more stable.
- Next cycle of tests started is modification of Unified multicore application for working on ESP32-S3-R8N8 (Olimex ESP32-S3-DevKit-Lipo) to display locally and remotely GPR (Ground Penetrating Radar) data currently generated by simulator. First test done shows that displaying data locally on 3.2" TFT SPI 240x320 display is stable at speeds 60+ tracks per second (at 460800 bps over serial) while all network services work on the second CPU core.
- Next cycle of tests started is experimenting with Pi Pico PIO engine functionality. It was used Raspberry Pi Pico, Olimex RP2040-PICO-PC boards and 5" TFT HDMI 800x480 display and as a beginning adapted by Adafruit Arduino IDE version of PicoDVI library and example tests were running successfully.

All the time the performance table (next page) was updated with the benchmark results measured by Adafruit graphics and DrawWithDMA tests. The connection table (page 5) was also updated.

Full color PicoDVI test on Raspberry Pi Pico, RP2040-PICO-PC and 5" TFT HDMI 800x480 display



Benchmark of unified graphic and scroll tests built on Adafruit_ILI9341, TFT_ILI9341 and TFT_eSPI libraries

Arduino board / MCU	UNO / ATMega328			Leonardo / ATMega32u4			D1 R32 / ESP32			Pi Pico / RP2040			Pi Pico / RP2040 (Overclocked)			Unified App	AVR128db48	ESP32-S3
ILI9341 Library used (SPI clock)	Adafruit	TFT	Speed up	Adafruit	TFT	Speed up	Adafruit (3MHz)	TFT_eSPI	Speed up	Adafruit	TFT_eSPI (27MHz)	Speed up	Adafruit	TFT_eSPI (62.5MHz)	Speed up	TFT_eSPI with DMA	Adafruit	Adafruit (27MHz)
Memory usage [B]																		
Flash used:	23,736 of 32,256 (73.59%)	21,870 of 32,256 (67.80%)		25,874 of 28,672 (90.24%)	23,992 of 28,672 (83.68%)		237,600 of 1,310,720 (18.13%)	295,261 of 1,310,720 (22.52%)		327,772 of 2,093,056 (15.65%)	372,092 of 2,093,056 (17.78%)		327,868 of 1,568,768 (20%)	372,180 of 1,568,768 (23%)		505,232 of 1,044,480 (48%)	24,354 of 130,560 (18.65%)	295,261 of 1,310,720 (22.52%)
SRAM used:	950 of 2,048 (46.39%)	746 of 2,048 (36.43%)		915 of 2,560 (35.74%)	711 of 2,560 (27.77%)		37,264 of 327,680 (11.37%)	19,480 of 327,680 (5.94%)		71,324 of 262,144 (27.21%)	71,768 of 262,144 (27.38%)		71,324 of 262,144 (27%)	71,768 of 262,144 (27%)		74,912 of 262,144 (28%)	1,087 of 16,384 (6.63%)	19,480 of 327,680 (5.94%)
Benchmarks [us]								**~42°C**			**~32°C**			**~34°C**		**~36°C**		**~50°C**
Screen fill	1,496,456	870,220	1.720	1,503,900	874,600	1.720	2,120,995	274,575	9.097	604,056	281,577	2.145	497,451	107,972	4.607	107,567	1,603,604	274,575
Text	147,088	60,416	2.435	147,820	60,724	2.434	99,610	32,599	6.491	45,452	18,831	2.414	30,599	8,085	3.785	8,070	114,885	32,599
Lines	1,172,116	242,732	4.829	1,178,004	243,988	4.828	986,748	339,491	10.975	454,856	101,897	4.464	304,234	42,741	7.118	43,648	946,199	339,491
Horiz/Vert Lines	125,064	71,336	1.753	125,656	71,696	1.753	173,171	24,171	8.603	50,042	23,541	2.126	40,853	9,078	4.500	8,880	132,637	24,171
Rectangles (outline)	82,228	45,844	1.794	82,632	46,076	1.793	110,682	15,996	8.697	32,657	14,932	2.187	26,417	5,773	4.576	5,686	85,703	15,996
Rectangles (filled)	3,107,060	1,807,436	1.719	3,122,844	1,816,740	1.719	4,402,687	570,510	9.096	1,253,856	584,372	2.146	1,032,576	224,086	4.608	223,506	3,329,307	570,510
Circles (filled)	452,728	284,064	1.594	454,916	285,536	1.593	492,735	95,809	7.704	167,914	71,149	2.360	126,969	28,025	4.531	27,896	423,221	95,809
Circles (outline)	497,252	135,580	3.668	499,604	136,148	3.670	432,728	150,143	12.978	199,626	37,258	5.358	133,263	15,561	8.564	15,743	404,412	150,143
Triangles (outline)	261,056	59,496	4.388	262,392	59,808	4.387	225,959	74,819	10.265	101,400	23,636	4.290	68,473	10,319	6.636	10,463	213,681	74,819
Triangles (filled)	1,330,720	694,456	1.916	1,337,200	698,032	1.916	1,432,757	209,558	8.691	429,998	195,995	2.194	345,244	75,450	4.576	75,102	1,279,412	209,558
Rounded rects (outline)	228,892	100,004	2.289	230,024	100,532	2.288	230,767	62,675	11.013	92,280	23,635	3.904	65,233	9,576	6.812	9,602	200,582	62,675
Rounded rects (filled)	3,127,968	1,976,936	1.582	3,143,588	1,987,180	1.582	4,384,111	578,880	8.995	1,257,871	586,292	2.145	1,032,024	225,027	4.586	224,252	3,330,751	578,880
Fill screen by pixels	3,369,992	918,732	3.668	3,387,308	923,492	3.668	2,783,609	1,591,181	3.331	1,255,234	504,753	2.487	805,373	229,258	3.513	159,327	2,964,859	1,591,181
Fill screen by bitmaps	528,576	855,088	0.618	531,112	859,520	0.618	435,205	62,752	0.518	66,438	520,180	0.128	70,363	234,904	0.300	166,092	453,099	62,752
Scroll and fill screen	**532,988**	**855,696**	**0.623**	**535,808**	**860,132**	**0.623**	**439,860**	**67,668**	**0.520**	**69,357**	**521,011**	**0.133**	**71,933**	**235,385**	**0.306**	**166,606**	**457,946**	**67,668**
Min	82,228	45,844		82,632	46,076		99,610	15,996		32,657	14,932		26,417	5,773		5,686	85,703	15,996
Avg	**1,097,346**	**598,536**	**1.833**	**1,102,854**	**601,614**	**1.833**	**1,250,108**	**276,722**	**4.497**	**405,402**	**233,937**	**1.733**	**310,067**	**97,416**	**3.183**	**83,496**	**1,062,687**	**276,722**
Max	3,369,992	1,976,936		3,387,308	1,987,180		4,402,687	1,591,181		1,257,871	586,292		1,032,576	235,385		224,252	3,330,751	1,591,181
Sum	16,460,184	8,978,036		16,542,808	9,024,204		18,751,620	4,150,827		6,081,037	3,509,059		4,651,005	1,461,240		1,252,440	15,940,298	4,150,827
DrawWithDMA test (bounsing of 42 colored and numbered circles)								**36fps (Unified App)**			**17.8 fps at CPU 133MHz SPI 27MHz**			**46.5 fps at CPU 250MHz SPI 62.5MHz**	**2.6**	**46.5 fps at CPU 250MHz SPI 62.5MHz**		**60+ fps at GPR data visualisation**

Notes:

- Memory usage numbers are as reported in runtime and slightly different than one reported by the compiler;
- Preparing of the data for filling the screen by pixels or bitmaps are made to be as fast as possible;
- Numbers for "Scroll and fill screen" tests at TFT_ILI9341 and TFT_eSPI libraries should be revised;
- At combination ESP32 and Adafruit_ILI9341 library SPI frequency was lowered to 3MHz while in case of ESP32 S3 SPI frequency can be increased up to 27MHz but in unified application with WiFi networking TFT_eSPI library has some problems especially at using DMA;
- Numbers in "Speed up" column means the operation is that many times faster;
- Overclocking in case of Pi Pico includes increasing of SPI and CPU speeds up to 62.5MHz and 250MHz respectively and application of suggested solution by Bodemar in his Github issue 1460 (working reliably even with 30cm long wires);
- Cases with ESP32 (Unified App), overclocked RP2040 (Unified App) and ESP32-S3 (colored in light violet) were measured by Unified multicore application (in combination with AsyncFSWebBrowser).

Useful links for display of animation with DMA and speed assessment:

Raspberry Pi Pico with ILI9341 TFT and TFT_eSPI Arduino library using RAM & DMA
https://forum.arduino.cc/t/tft_espi-support-for-raspberry-pi-pico-added/702551
https://www.youtube.com/watch?v=njFXIzCTQ_Q
https://github.com/Bodmer/TFT_eSPI/issues/1460#issuecomment-1006661452

This application uses two sprites in RAM and DMA for filling display half buffer while updating the other half. The ILI9341 display operates reliably on Pi Pico up to 62.5MHz so frame rate up to ~43fps is possible with DMA. Overclocking CPU to 250MHz and applying Bodmer note makes it possible frame rates to go up to 46.5fps. The total consumption in overclocked mode of both Pi Pico and SPI TFT is 110mA. The application is unified to run on both RP2040 and ESP-WROOM-32 boards. In case of ESP32 frame rate was lower (~36fps).

Modified multicore application for ESP32-S3, RP2040 and ESP32 to display GPR data locally and remotely

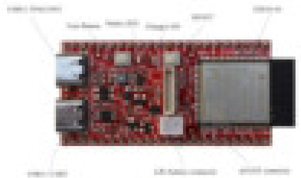
Modified multicore application is based on Unified multicore application which is combination of AsyncFSBrowser demo application, Unified graphic test (TFT-eSPI library) and DrawWithDMA sketches. It is targeted to ESP32-S3 based Olimex ESP32-S3-DevKit-Lipo board.

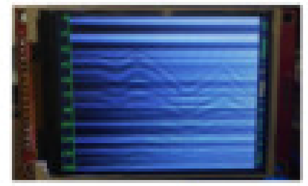
ESP32-S3 is a dual-core XTensa LX7 MCU, capable of running at 240 MHz. Apart from its 512 KB of internal SRAM, it also comes with integrated 2.4 GHz, 802.11 b/g/n Wi-Fi and Bluetooth 5 (LE) connectivity that provides long-range support. It has 45 programmable GPIOs and supports a rich set of peripherals. ESP32-S3 supports larger, high-speed octal SPI flash, and PSRAM with configurable data and instruction cache.

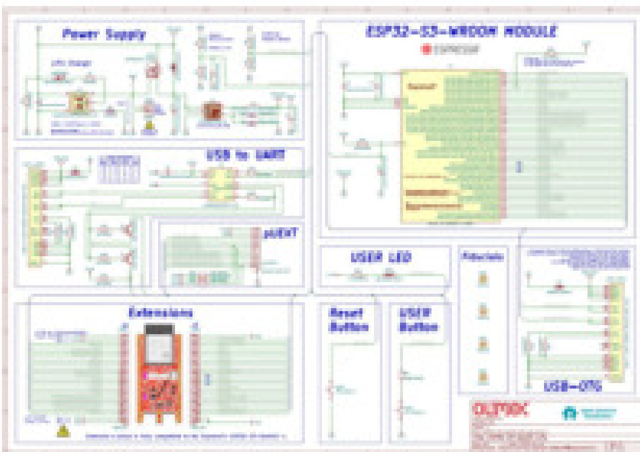
Olimex ESP32-S3-DevKit-Lipo board with ESP32-S3-WROOM-1-N8R8 has 8MB PSRAM and 8MB SPI Flash. It also has pUEXT and 2x USB C connectors (via CH340 USB-serial adapter and native OTG JTAG/Serial with on-chip PHY), LiPo battery charger and connector, user and reset buttons and user and charge LEDs. All GPIO pins are routed to 2x22 pins connectors compatible to the Espressif ESP32-S3-DevKitC-1.

Development is made on Windows 7 / 10 using Arduino IDE (ver. 1.8.9), the latest Espressif system version 2.0.14, AsyncWebServer, Adafruit_ILI9341, Adafruit_GFX etc. libraries (all latest versions). Node.JS (ver. 12.22.9) with serialport library (ver. 8.0.5) is used to develop and use "gpr-simulator" application.

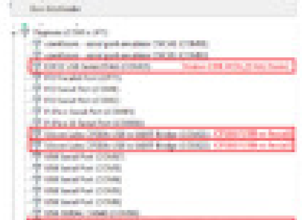
To power and connect the board to the computer may use powered 4+ USB Hub, 2x USB A to USB C cables and 2x CP2102 USB-to-Serial adapters. Install CH340 and CP210x drivers for Windows if required.







```
For user configuration of internal 8MB flash do following changes:
In ESP32 boards file: C:\Users\KC\AppData\Local\Arduino15\packages\esp32\hardware\esp32\2.0.14\boards.txt
add following lines to esp32s3.name=ESP32S3 Dev Module section:
    esp32s3.menu.PartitionScheme.users_8MB=8M with spiffs (2MB APP/2MB OTA/4MB SPIFFS)
    esp32s3.menu.PartitionScheme.users_8MB.build.partitions=users_8MB
    esp32s3.menu.PartitionScheme.users_8MB.upload.maximum_size=2097152
Add: C:\Users\KC\AppData\Local\Arduino15\packages\esp32\hardware\esp32\2.0.14\tools\partitions\users_8MB.csv
with following content:
    # Name,   Type, SubType, Offset,  Size, Flags
    nvs,      data, nvs,     0x9000,  0x5000,
    otadata,  data, ota,     0xe000,  0x2000,
    app0,     app,  ota_0,   0x10000, 0x200000,
    app1,     app,  ota_1,   0x210000,0x200000,
    spiffs,   data, spiffs,  0x410000,0x3e0000,
    coredump, data, coredump,0x7F0000,0x10000,
Restart Arduino IDE changes to take effect
```



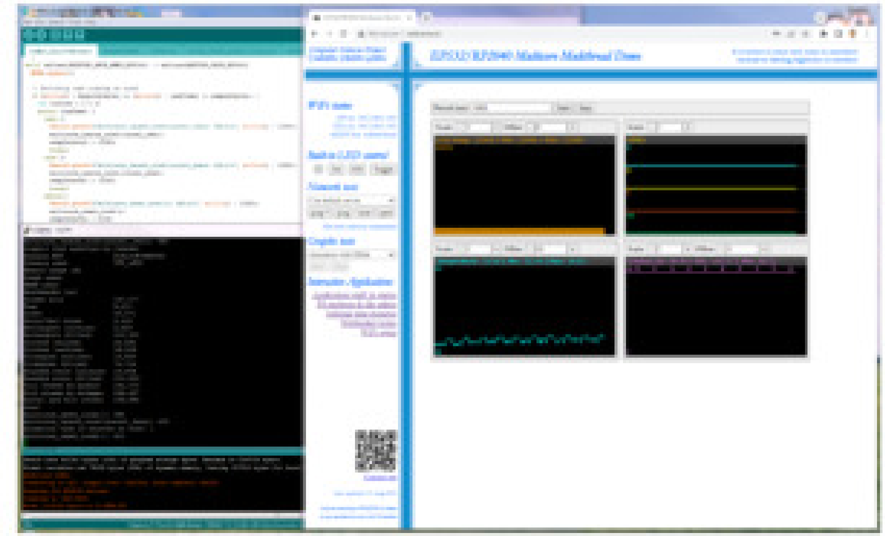
Save new copy of Unified multicore application under other name (like GPR_Display) and remove not relevant components like TCPServer.ino, DrawWithDMA.ino, Unified_ili9340_Graphic_Test_plus.ino and create new components GPR_Task.ino and GPS_Task.ino. Additional component SPIFFS based FSEditor.ino is added instead of SPIFFSEditor to overcome its problem if ESP32 core ver. 2.0 and later is used. Restructure SPIFFS file stuff to separate private and public files. Restart Arduino IDE and re-flash SPIFFS changes to take effect. Modify the stuff in GPR_Display.ino and WebSockets.ino according to changes made. Write required stuff in GPR_Task.ino to read data from Serial2 and display them on 3.2" TFT SPI 240x320 display. GPR_Task has to be set to run on CPU core0 (not running main application). Use board configuration in Arduino -> Tools as shown above. ESP32-S3 may be programmed via native USB OTG JTAG/Serial port (COM23). Serial0 port via CH230 USB-to-Serial adapter (COM20) is used for debug information printing and as a spare channel for application programming. ESP32-S3 Serial2 port is used to connect GPR and Serial1 port is reserved to connect GPS module later on.

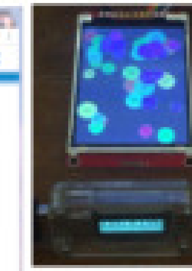
Web files are located in project subfolder named "data". They can be flashed to internal flash SPIFFS using Arduino tool "ESP32 Sketch Data Upload". Application "esp_tools_gui" may also be used to check ESP32-S3 information and configuration. For testing purposes JavaScript simulator is written to read GPR data from SEGY file and send them to ESP32-S3 Serial2 port (via CP2102 USB-to-Serial adapter on COM22). In future is planed JavaScript GPS simulator to be developed to read GPS data from file and send them to ESP32-S3 Serial1 port (via CP2102 USB-to-Serial adapter on COM21).

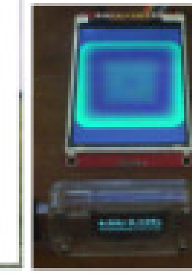
Notes:

- In future both USB OTG JTAG/Serial and CH230 USB to Serial0 ports may be used as USB Mass storage host (for storing of archive data files) and CDC device (for connecting to other computer) respectively but some issues have to be solved so currently they will be used for development purposes only;
- In case of more space needed SPIFFS partition scheme can be changed or ESP32-S3 module with more flash (16/32MB) can be used;
- Usage of SPI SD card slot located at 3.2" TFT SPI 240x320 display board for storing of archive files is also possible;
- Lack of ESP32-S3 reset after application flashing via USB OTG JTAG/Serial port bug is not observed;

Unified multicore application for ESP32 and RP2040 – more than combination of AsyncFSWebBrowser and graphic tests

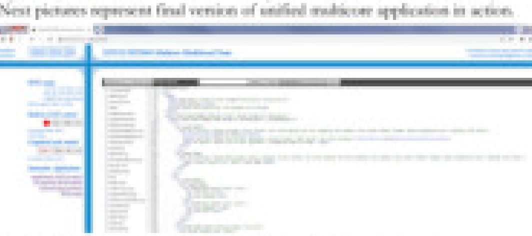




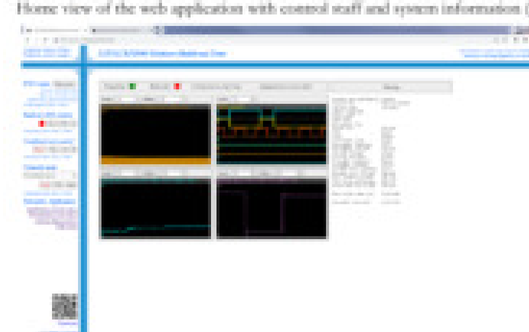


Unified multicore application is based on AsyncFSBrowser, Unified graphic test (TFT-eSPI library) and DrawWithDMA sketches. It is Arduino IDE multi-file project for ESP32-WROOM and RP2040 based boards. It is running on both Arduino D1 R32 ESP32 (ESP32-WROOM) and Raspberry Pi Pico W (RP2040) boards and implements web server and sockets, console with printout of the Adafruit TFT tests adapted for eTFT library and log of swapping graphical tasks to work on the other core, web application with internal monitor showing graphs of the free heap memory, the GPIO states, the internal temperature sensor and the animation frame rate. On the right are shown pictures of graphical tasks (animation and graphical tests) running alternatively on the other core.

The most attractive application feature is almost complete independency of the performance of tasks running on different CPU cores. Other impressive result is graphical performance of animation task (46 frames per second at 42 bouncing circles) and all Adafruit TFT tests adapted to work with the eTFT library (especially scrolling 320x240 graphics at speed of 0.8ms per 240 pixels line). Next pictures represent final version of unified multicore application in action.



Home view of the web application with control staff and system information (left) and control staff with internal FS file viewer and editor (right)



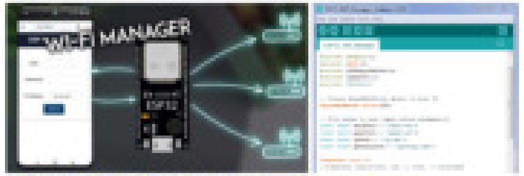
On the left most pictures are shown views with graphs of the internal system and application parameters (left) and WiFi setup page (right). It is evident that the results of the invoked commands and monitor (as well as other clients) are synchronized and ESP32 WiFi works in AP+STA mode (top left WiFi status). Graphical task log in the monitor view is showing the best results ever achieved (look at the table on page 7). This snapshot is taken after complete unification of the application for Pi Pico W (left) and ESP32 (right) platforms and adding network tasks control bar (only ping is implemented for the moment).

On the right most pictures are shown TFT displays in action connected to Pi Pico (top left) and ESP32 (top right) boards while running unified application. The current state of the application is not stable and has some bugs especially for ESP32 where bouncing circles are only moving at the bottom half of the screen, Adafruit tests do not work as expected and the application crashes frequently at exchanging of the graphic tasks. In case of Pi Pico applications is more stable but crashes form time to time at exchanging of the graphic tasks.

In the special modification of the unified multicore application running on ESP32-S3-R8N8 (right most bottom) is implemented preliminary test for displaying of data from GPR (Ground Penetration Radar) received via serial link and shown as 320x240 pixels scrolling graphics with up to 18 tracks per second speed.

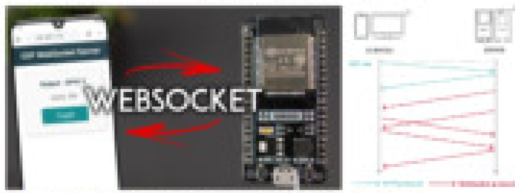
Network performance using AsyncWebServer and AsyncTCP libraries on Pi Pico W and ESP32 series of boards

Startup projects working on ESP32 S2 Olimex boards and based on ESPAsyncWebServer library for Arduino:

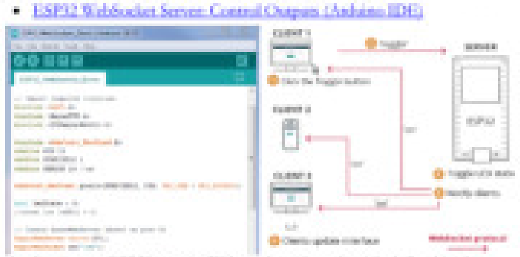


- ESP32: Create a Wi-Fi Manager (AsyncWebServer library)

Application uses SPIFS on ESP32 systems to hold web and configuration files which have to be written manually by "ESP32 Sketch Data Upload" tool of Arduino IDE. The application first runs in AP mode asking for connection credentials of the local router. After storing them in FS files and restart it runs in STA mode. Main web page allows controlling built-in LED.



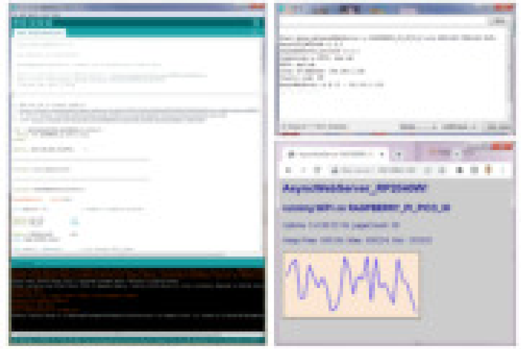
- ESP32 WebSocket Server: Control Outputs (Arduino IDE)



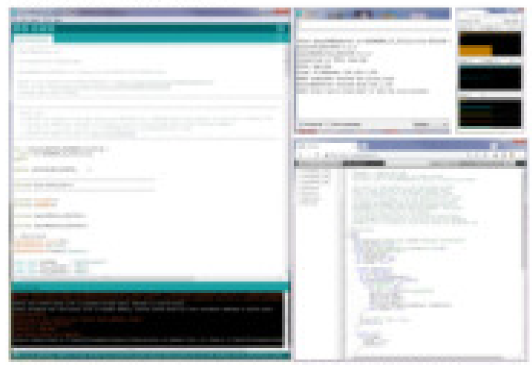
Application on ESP32 runs in STA mode with credentials defined in the sketch and open WebSocket server to control the LED. Its status can be changed by any client and will be updated at all the clients.

It was used Adafruit NeoPixel library to run above projects on Olimex ESP32 S2 series of boards with RGB instead of regular LED.

Startup projects on Raspberry Pi Pico W – AsyncWebServer for RP2040W library examples:



- Async_AdvancedWebServer



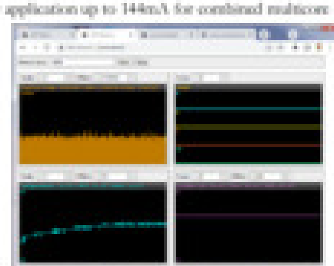
- AsyncFSWebServer (library ver. 1.5.0 did not run out of the box)

Application uses LittleFS library to access SPI flash FS. It also uses mDNS, basic authentication, AsyncWebSocket, AsyncEventSource and AsyncFSEditor_RP2040W library to show and edit files.

DrawWithDMA TFT_eSPI library test was compiled and run successfully on Raspberry Pi Pico W. Later on AsyncFSWebServer and DrawWithDMA combined multicore application was done by simply putting both files in a single project, renaming setup and loop functions in the second file to setup1 and loop1 and commenting the line Serial.begin(115200). Display drawing (42 circles) speed was the same (17.85fps) without appreciable change in the web access. Temperature measured by internal sensor is increased with approximately 2°C (up to 31°C). The heap is increased from 5kB up to 159kB. CPU overclocking to 250MHz did not speed up display drawing and web access but increase the temperature with approximately 3°C (up to 34°C). SPI speed can be changed in User_Setup.h of TFT_eSPI library. Changing it from 27MHz to 55MHz (2x) did not speed up display drawing but thanks to Bodmer comment and CPU clocking at 125MHz (SPI clock is 62.5MHz) display drawing can be speed up to 43-45fps @42 circles and 46.3fps @36 circles. Overclocking CPU to 250MHz (probably SPI clock is again 62.5MHz) increase display drawing speed up to 46.5fps @42 circles (2.6x) while working smoothly and reliably. Total consumption is increased form 110mA in case of overclocked DrawWithDMA single core application up to 144mA for combined multicore application.

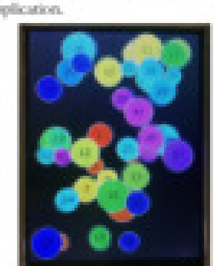
Remote file manager and editor

+



On-line monitor

+



SPI TFT display

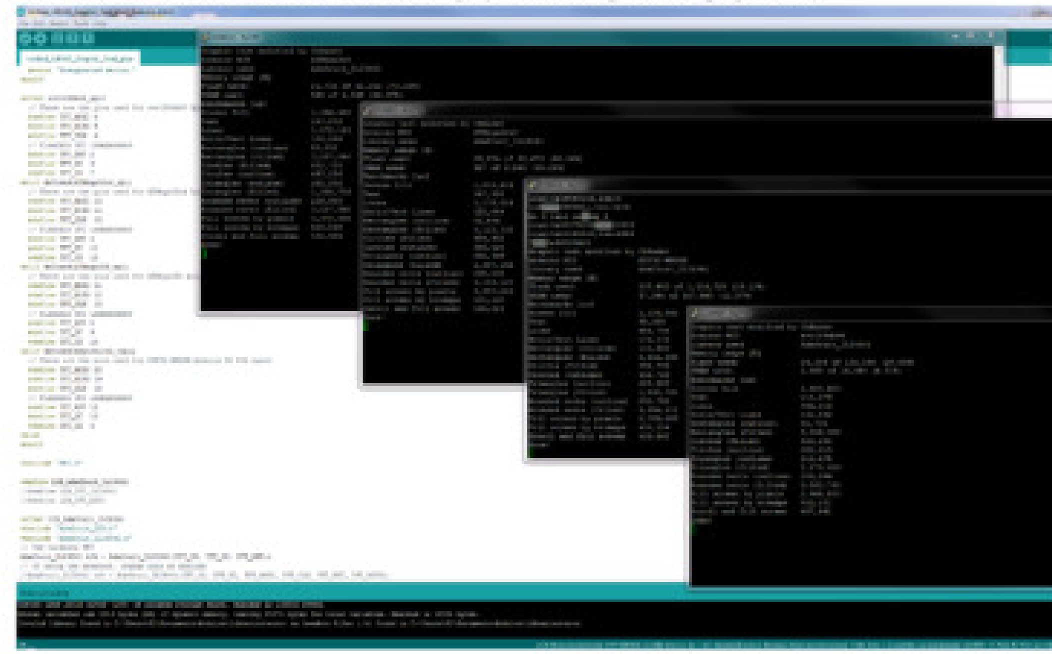
AsyncWebServer for RP2040W library built by Khoi Hoang is based on and modified from ESPAsyncWebServer library support of ESP32 and ESP8266 on Arduino cores. Next steps to be done for building of unified multicore application:

- Check of the code compatibility for both ESP32 and Pi Pico W boards;
- Dynamically running of different tasks on the second CPU core;
- Build unified web server application with WiFi working in AP and/or STA modes including its management, mDNS, LittleFS, WebSockets etc.

Connection setup for 3.2" 240x320 pixels TFT display with SPI interface

	3.2" TFT SPI LCD Display	Arduino UNO ATMega328	Olimexino32U4 ATMega32u4	Optiboot AVR128db48	Arduino R32 ESP-WROOM-32	Raspberry PI Pico RP2040	ESP32-S3-WROOM	Signal description (3.2" TFT SPI LCD Display)
1	VCC	VCC-3.3V	VCC-3.3V	VCC-3.3V	VCC-3.3V	VCC-3.3V	3.3V	3.3V power input (do not connect to 5V)
2	GND	GND	GND	GND	GND	GND	GND	GND
3	CS	D10	D13	0,SS, PA7	IO05	GP17	GPIO10	LCD chip select signal, low level enable
4	RESET	D8	D4	PA2 (0,SDA)	IO12	GP21	GPIO9	LCD reset signal, low level reset
5	DC/RS	D9	D11	PA3 (0,SCL)	IO13	GP20	GPIO14	LCD register / data selection signal, high level: register, low level: data
6	SDI(MOSI)	D11	D16	0,MOSI, PA4	IO23	GP16	GPIO11	SPI bus write data signal
7	SCK	D13	D15	0,SCK, PA6	IO18	GP18	GPIO12	SPI bus clock signal
8	LED	VCC-5V	VCC-5V	VCC-5V	VCC-5V	VCC-5V	5V	Backlight control, high level lighting, if not controlled, connect 5V for always bright
9	SDO(MISO)	D12	D14	0,MISO, PA5	IO19	GP19	GPIO13	SPI bus read data signal, if you do not need to the read function, you cannot connect it

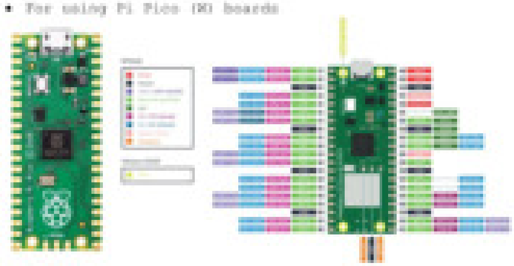
All 4 boards are connected to 3.2"SPI TFT display and running Unified graphic test



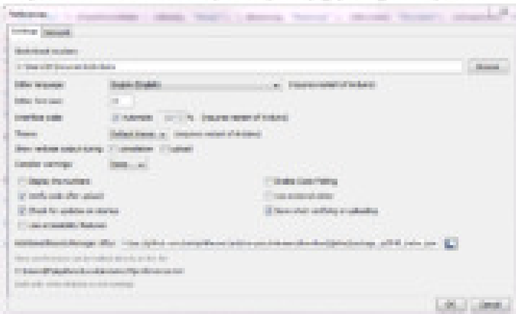
Arduino UNO (ATMega328) Olimexino-32U4 (ATMega32u4) Arduino D1 R32 (ESP32) Optiboot (AVR128db48)

Arduino Pi Pico (W) boards

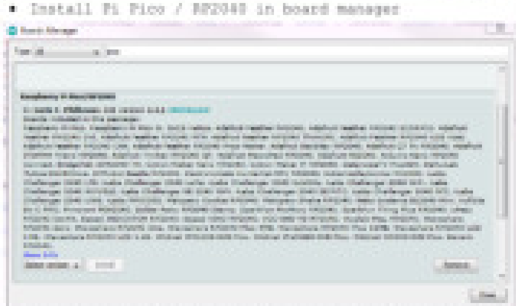
- For using Pi Pico (W) boards



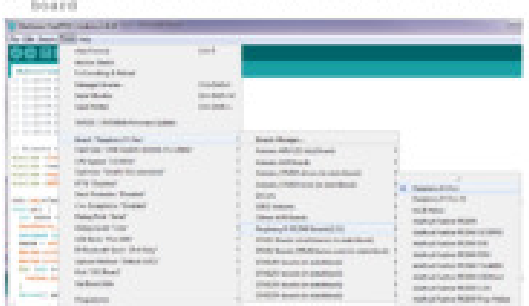
- In Preferences add URL: https://github.com/earlephilhower/arduino-pico/releases/download/global/package_rp2040_index.json



- Install Pi Pico / RP2040 in board manager



- Install "Raspberry Pi Pico" or "Raspberry Pi Pico W" board



- Connect the board to Windows PC while BOOTSEL button is pushed - "RPI-RP2" mass storage device should be appeared
- After uploading the sketch "Pico" or "Pico W" device will be appeared in "Device Manager"
- Update its device driver using Atmel USB to serial INF file changing [DeviceList.*] sections to:
  %PI_CDC_PICO%=DriverInstall, USB\VID_2E8A&PID_000A&MI_00 or
  %PI_CDC_PICO%=DriverInstall, USB\VID_2E8A&PID_F00A&MI_00
- Change [Strings] sections also to appropriate once

Multicore version of "Hello World and Blinking LED" common test for Pi Pico

- Open from File -> Examples -> (Examples for Raspberry Pi Pico) -> FreeRTOS -> Multicore FreeRTOS sketch and safe it in your Arduino sketch folder:

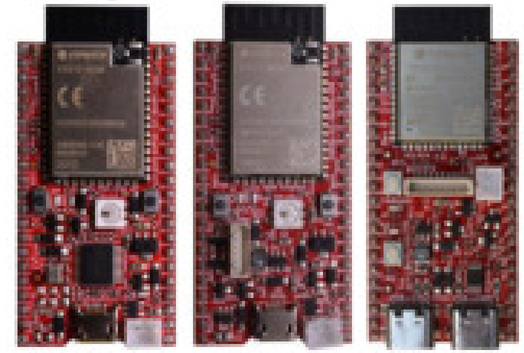
```
#include <FreeRTOS.h>
#include <task.h>
#include <map>
#include <EEPROM.h>
std::map<eTaskState, const char *> eTaskStateName {
{eReady, "Ready"}, { eRunning, "Running" }, {eBlocked,
"Blocked"}, {eSuspended, "Suspended"}, {eDeleted,
"Deleted"} };
void ps() {
  int tasks = uxTaskGetNumberOfTasks();
  TaskStatus_t *pxTaskStatusArray = new
TaskStatus_t[tasks];
  unsigned long runtime;
  tasks = uxTaskGetSystemState( pxTaskStatusArray, tasks,
&runtime );
  Serial.printf("# Tasks: %d\r\n", tasks);
  Serial.println("ID, NAME, STATE, PRIO, CYCLES");
  for (int i=0; i < tasks; i++) {
    Serial.printf("%d: %-16s %-10s %d %lu\r\n", i,
pxTaskStatusArray[i].pcTaskName,
eTaskStateName[pxTaskStatusArray[i].eCurrentState],
(int)pxTaskStatusArray[i].uxCurrentPriority,
pxTaskStatusArray[i].ulRunTimeCounter);
  }
  delete[] pxTaskStatusArray;
}
void blink(void *param) {
  (void) param;
  pinMode(LED_BUILTIN, OUTPUT);
  while (true) {
    digitalWrite(LED_BUILTIN, LOW);
    delay(750);
    digitalWrite(LED_BUILTIN, HIGH);
    delay(250);
  }
}
void setup() {
  Serial.begin(115200);
  xTaskCreate(blink, "BLINK", 128, nullptr, 1, nullptr);
  delay(5000);
}
volatile int val= 0;
void loop() {
  Serial.printf("C0: Blue leader standing by...\r\n");
  ps();
  Serial.printf("val: %d\r\n", val);
  delay(1000);
}
// Running on core1
void setup1() {
  delay(5000);
  Serial.printf("C1: Red leader standing by...\r\n");
}
void loop1() {
  static int x = 0;
  Serial.printf("C1: Stay on target...\r\n");
  val++;
  if (++x < 10) {
    EEPROM.begin(512);
    EEPROM.write(0,x);
    EEPROM.commit();
  }
  delay(1000);
}
```

- It demonstrates a simple use of the setup1()/loop1() functions for a multiprocessor run and following will be printed on the serial port while LED is blinking:

```
C1: Stay on target...
C0: Blue leader standing by...
# Tasks: 9
ID, NAME, STATE, PRIO, CYCLES
0: CORE0            Running     4 191473164
1: IDLE1            Running     0 3622023404
2: IDLE0            Ready       0 3371562651
3: BLINK            Blocked     1 5381238
4: CORE1            Blocked     4 103437988
5: USB              Blocked     6 3826967365
6: Tmr Svc          Blocked     2 21071
7: IdleCore1        Suspended   7 17213
8: IdleCore0        Suspended   7 88986822
val: 683
```

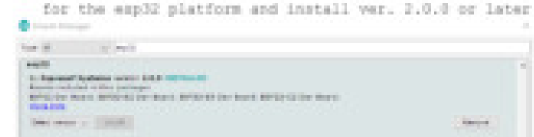
ESP32-S2/3 boards on Arduino IDE

- For using ESP32-S2 boards like:

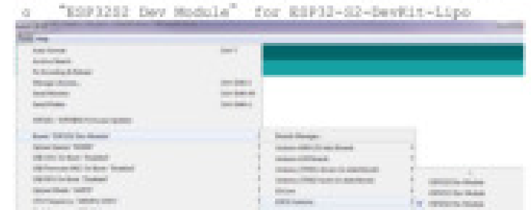


ESP32-S2-DevKit-Lipo | ESP32-S2-WROVER-DevKit-Lipo-USB | ESP32-S3-DevKit-Lipo

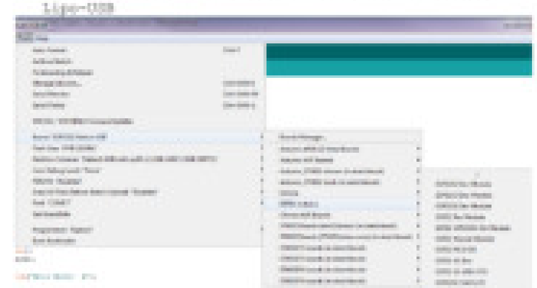
- Install the ESP32-S2 support for Arduino IDE
- In "File" → "Preferences" add URL: https://espressif.github.io/arduino-esp32/package_esp32_index.json
- In "Tools" → "Boards" → "Board Manager" search for the esp32 platform and install ver. 2.0.0 or later



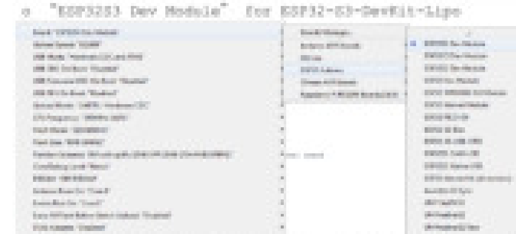
- Restart IDE and select board in "Tools" → "Board:
  - "ESP32S2 Dev Module" for ESP32-S2-DevKit-Lipo



  - "ESP32S2 Native USB" for ESP32-S2-WROVER-DevKit-Lipo-USB



  - "ESP32S3 Dev Module" for ESP32-S3-DevKit-Lipo



- Connect ESP32-S2/3-DevKit-Lipo and install driver for USB-Serial CH340 adapter if needed
- Connect ESP32-S2-WROVER-DevKit-Lipo-USB and put it in boot loader's mode (hold GPIO0 low while reset)
- Install driver with Zadig software if needed
  - Enable in "Options" → "List all devices"
  - Choose device "ESP32-S2 (Interface 2)"
  - And option "USB Serial (CDC)"
- Any time for programming ESP32-S2-WROVER-DevKit-Lipo-USB has to be put in boot loader's mode and reset manually after uploading the sketch

Multitasking "Hello World & RGB LED" test

```
/*
 * Requires Adafruit NeoPixel library
 */
#include <Adafruit_NeoPixel.h>
#define PIN 18
#define NUMPIXELS 1
#define PERIOD 10 //ms
Adafruit_NeoPixel pixels(NUMPIXELS, PIN,
                         NEO_GRB + NEO_KHZ800);
int colors[3];
void setup() {
  pixels.begin();
  for (int i = 0; i < 3; i++) colors[i] = 0;
#if 1
  Serial.begin(115200); // ESP32-S2-DevKit-Lipo
#else
  Serial.begin();// ESP32-S2-WROVER-DevKit-Lipo-USB
  // Wait for serial port to connect.
  // Needed for native USB port only.
  while (!Serial) ;
#endif
  Serial.println("Hello World!");
  vTaskDelay(1000 / portTICK_PERIOD_MS);
  xTaskCreate(loop2,"loop2", 2048, NULL,1,NULL);
}
int n = 0;
void loop2( void * parameter ) {
  while(1) {
    Serial.print("Hello World - "); Serial.println(n++);
    vTaskDelay(2000 / portTICK_PERIOD_MS);
  }
}
void loop () {
  for (int i = 0; i < 3; i++) {
    int j;
    for (j = 0; j < 256; j++) {
      colors[i] = j;
      pixels.setPixelColor(0, pixels.Color(colors[0],
                           colors[1], colors[2]));
      pixels.show(); delay (PERIOD);
    }
    for (j = 255; j >= 0; j--) {
      colors[i] = j;
      pixels.setPixelColor(0, pixels.Color(colors[0],
                           colors[1], colors[2]));
      pixels.show(); delay (PERIOD);
    }
  }
}
```

- Compiler messages for ESP32-S2-WROVER-DevKit-Lipo-USB

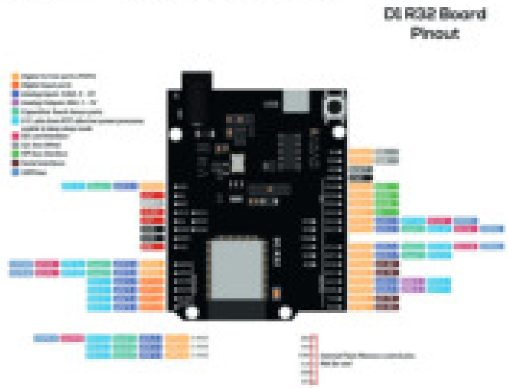
Sketch uses 291526 bytes (22%) of program storage space. Maximum is 1310720 bytes.
Global variables use 27596 bytes (8%) of dynamic memory, leaving 300084 bytes for local variables. Maximum is 327680 bytes.

- After running sketch on ESP32-S2-WROVER-DevKit-Lipo-USB composite device will be installed with TinyUSB DFU_RT, CDC and ESP32-S2 Firmware MSC devices.
- In terminal connected to USB-Serial CH340 following messages will be sent from ESP32-S2-DevKit-Lipo:

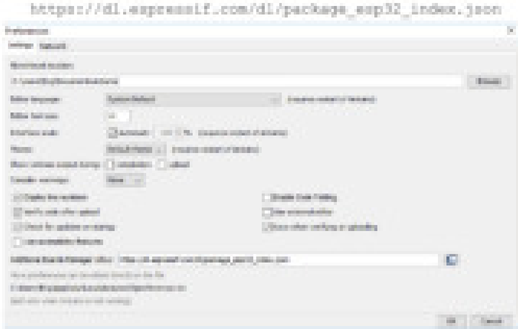
```
ESP-ROM:esp32s2-rc4-20191025                 System messages
Build:Oct 25 2019
rst:0x1 (POWERON),boot:0x8 (SPI_FAST_FLASH_BOOT)
SPIWP:0xee
mode:DIO, clock div:1
load:0x3ffe6100,len:0x524
load:0x4004c000,len:0xa70
load:0x40050000,len:0x2958
entry 0x4004c18c
Hello World!                      Sent from setup section
Hello World - 0    Sent from loop2 task and will count every 2 sec
Hello World - 1
```

Arduino D1 R32 ESP32 board

- For using ESP32 boards like D1 R32

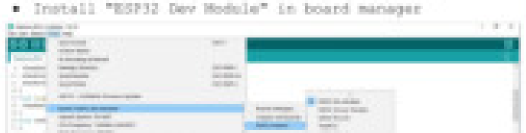


- In Preferences add URL:
  https://dl.espressif.com/dl/package_esp32_index.json

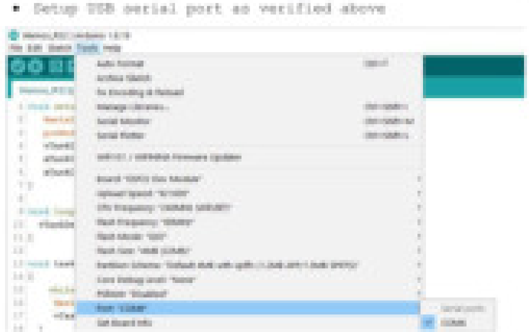


- Install esp32 in board manager

- Connect the board to Windows PC
- Install CH340 USB serial driver if needed and verify the port in "Device Manager": COM6 for example
- Install "ESP32 Dev Module" in board manager



- Setup USB serial port as verified above



Multitasking version of "Hello World & Blinking LED" test for ESP32

- Create new project "HelloWorld" and put the sketch:

```
void setup() {
  Serial.begin(112500);
  // By default the LED is connected to IO02
  pinMode(2, OUTPUT);
  // This will print default SPI pins
  Serial.println("Default SPI pins:");
  Serial.print("MOSI: "); Serial.println(MOSI);
  Serial.print("MISO: "); Serial.println(MISO);
  Serial.print("SCK: ");  Serial.println(SCK);
  Serial.print("SS: ");   Serial.println(SS);
  vTaskDelay(1000 / portTICK_PERIOD_MS);
  xTaskCreate(task1,"task1", 2048, NULL,1,NULL);
  xTaskCreate(task2,"task2", 2048, NULL,1,NULL);
}
void loop() {
  vTaskDelay(1000 / portTICK_PERIOD_MS);
}
void task1( void * parameter ) {
  while(1) {
    Serial.println("Hello World!");
    vTaskDelay(2000 / portTICK_PERIOD_MS);
  }
}
void task2( void * parameter) {
  while(1) {
    digitalWrite(2, HIGH);
    vTaskDelay(100 / portTICK_PERIOD_MS);
    digitalWrite(2, LOW);
    vTaskDelay(100 / portTICK_PERIOD_MS);
  }
}
```

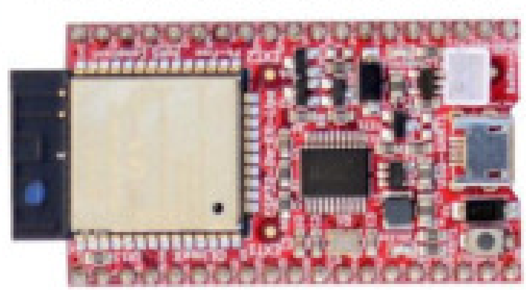
- After compilation will see:

```
Sketch uses 204926 bytes (15%) of program storage space.
Maximum is 1310720 bytes.
Global variables use 13416 bytes (4%) of dynamic memory,
leaving 314264 bytes for local variables. Maximum is
327680 bytes.
```

- After uploading sketch will see fast blinking LED and following messages in terminal to USB serial port:

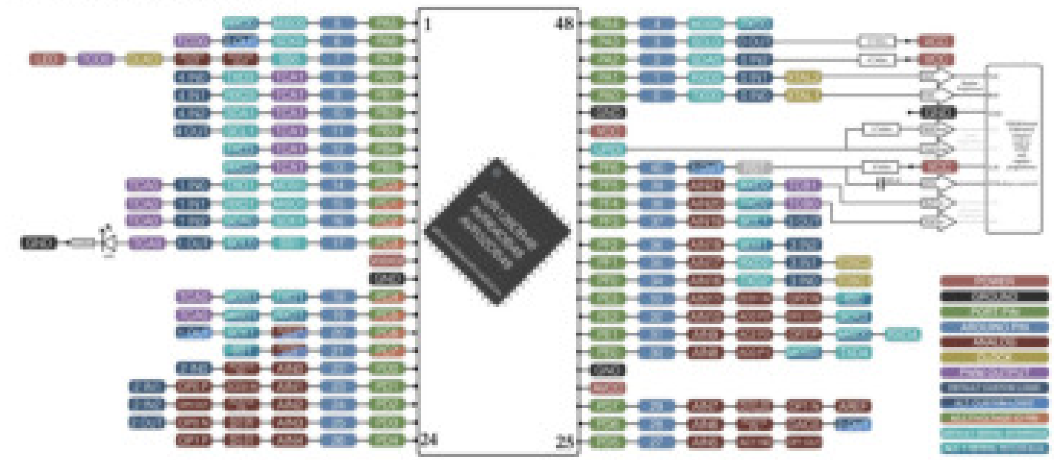
```
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x
00,wp_drv:0x00
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1216
ho 0 tail 12 room 4
load:0x40078000,len:10944
load:0x40080400,len:6388
entry 0x400806b4
Default SPI pins:                Default settings belongs to VSPI
MOSI: 23
MISO: 19
SCK: 18
SS: 5
Hello World!                     Will be repeated every 2 sec
Hello World!
```

Alternative ESP32 development boards from Olimex based on ESP32-WROOM-32 or ESP32-WROVER WiFi/BLE modules



Variants compatible with Arduino D1 R32 ESP32 board:
ESP32-DevKit-Lipo and ESP32-DevKit-Lipo-EA

AVR128db48 Arduino boards

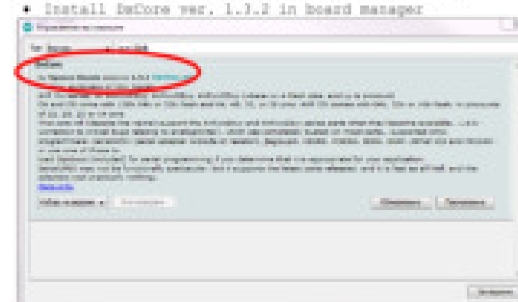


Self-made AVR128db48 Arduino like board based on QFP adapter board

Other boards notes:

- Arduino UNO – install windows driver for USB-Serial CH340 adapter,
- Olimexino Nano – install windows driver for Arduino Leonardo compatible boards,
- Set in Tools → Board → Arduino AVR Boards → Arduino UNO or Arduino Leonardo respectively,
- Set in Tools → Port → corresponding COM port,
- LED pin may be different for different boards – change it in "Blinking LED" test sketch.

- For using AVR128DB48 boards from Anton do:
- Add URL in Preferences:
  http://drazzy.com/package_drazzy.com_index.json
- Install DxCore ver. 1.3.2 in board manager



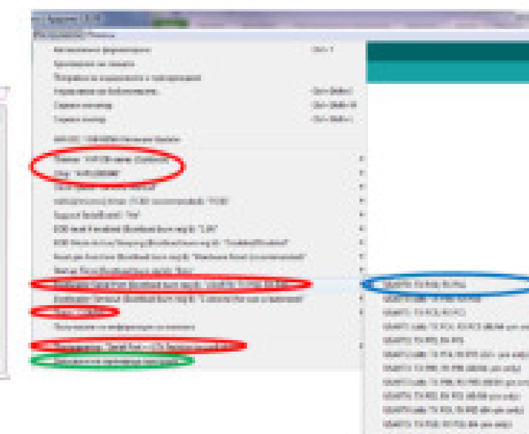
- Connect CP2102 USB to UART Bridge to Windows PC
- Install CP2102 USB driver if needed and verify the port: COM10 for example

UPDI programmer (to burn bootloader)

- Connect CP2102 USB to UART Bridge to the board
  - Rx – 4.7k res. – Tx – AVR128DB48 UPDI (pin 41),
  - DTR – 200nF – RST (p. 40), GND, VCC (3.3V)
- Programmer: "Serial Port + 4.7k Resistor (pyupdi style)"
- Usage: Tools → Burn Bootloader
- Usage: Sketch → Upload Using Programmer

Regular serial programmer

- Connect CP2102 USB to UART Bridge to the board
  - CP2102/TTL-232R Tx – AVR128DB48 Rx0 (p. 45)
  - CP2102/TTL-232R Rx – AVR128DB48 Tx0 (p. 44)
  - DTR – 200nF – RST (p. 40), GND, VCC (3.3V)
- Usage: Sketch → Upload



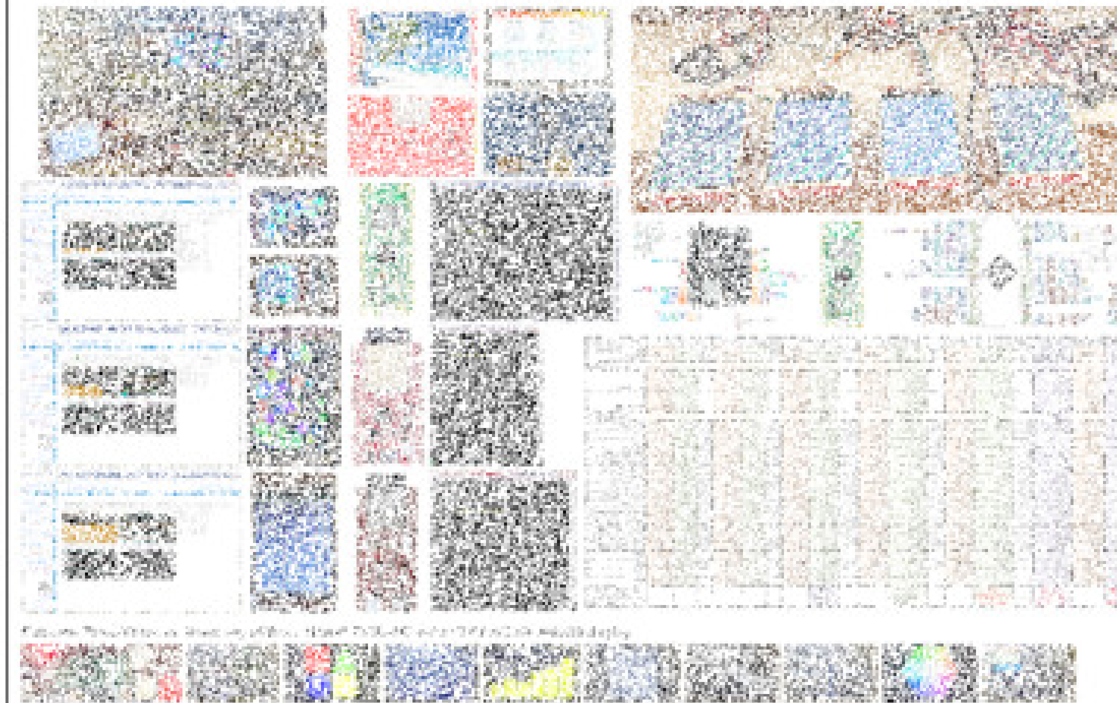
Bootloader serial port could be USART2 (alt), but using USART0 the PC COM port for both programming and serial communication with the sketch will be the same.

"Blinking LED" test for avr128db48

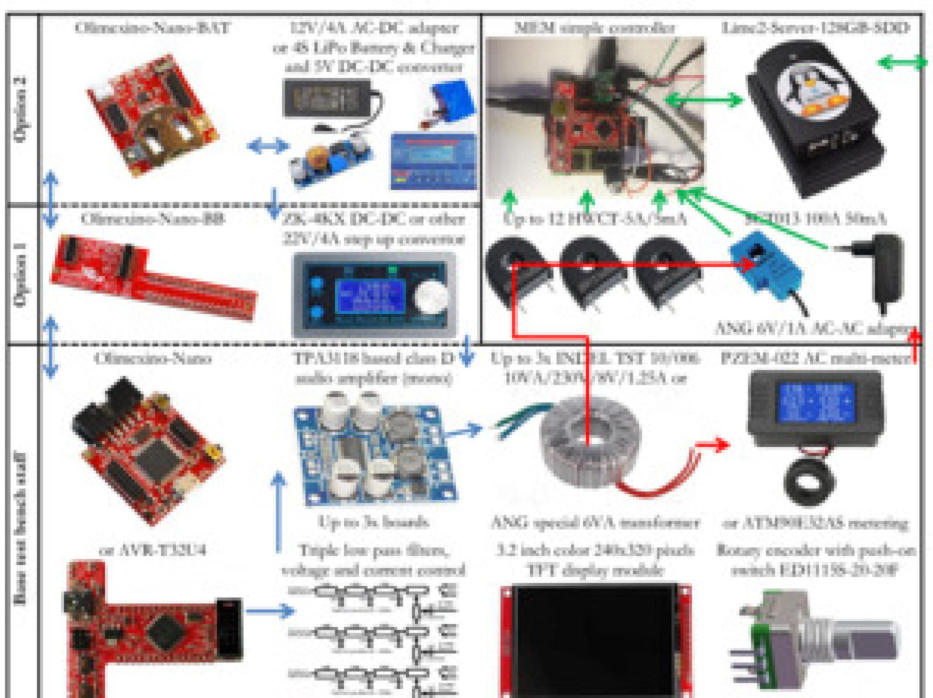
```
void setup() {
  // PIN_PC3 for avr128db48
  // may be different for other boards!
  pinMode(17, OUTPUT);
}
void loop() {
  digitalWrite(17, 1);
  delay(100);
  digitalWrite(17, 0);
  delay(100);
}
```

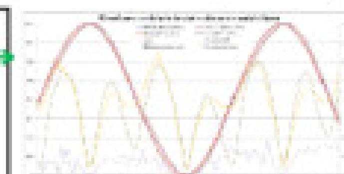
5kW Wind Generator, test bench and measured data (2013)

United Multicore Low Power IoT Platform (2023)

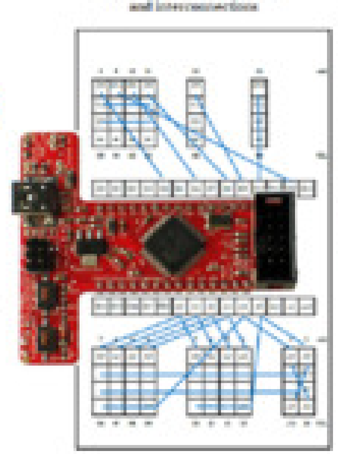


Multichannel Energy Metering system and test bench (2023)





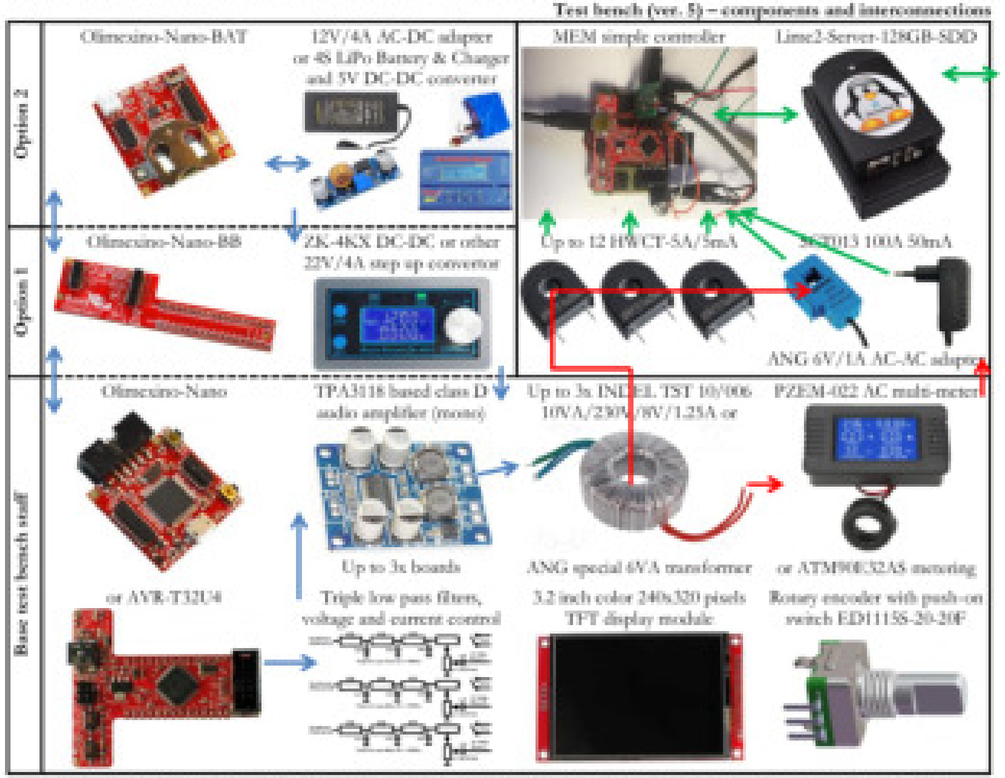
Arduino, MCU and MEM pins
and interconnections



Adroid - the open S.T.E.A.M. robot platform (2024)

After measuring of some transformers parameters and waiting for a special ANG transformer production the test bench software was extended in following directions. PWM outputs of timer 4 were increased to 3x for implementing of 3-phase voltage and current source. In case of using it as a single phase 2x PWM outputs were used for separate voltage and current generating with controllable phase difference in the range [-π, +π]. In 3-phase variant voltage and current for each phase will be generated by a single transformer without phase difference between them. The user interface was added and tested with rotary encoder with push-on switch ED1115S-20-20F (Code for Rotary Encoder with interrupts), 16x2 character LCD with I²C interface (LiquidCrystal I2C library for Arduino) and 3.2" 240x320 pixels TFT display with SPI interface. In case of TFT display Arduino Ucglib color graphics library was used for low memory footprint. For better level control of voltages and currents digital potentiometer of Microchip MCP4261 was added and Arduino library available from https://github.com/dreamcat4/Mcp4261 was used. After all above staff the sketch uses 92% flash and 34% RAM of Atmega32u4 memories resources and some of them will be freed after application optimization.

Test bench (ver. 5) – components and interconnections



It is planned none monolithic design in the latest test bench (ver. 5). MCU controller staff and power supply components will be mounted in a single plastic box produced by 3D printing. Transformers and probably audio amplifiers will be mounted in separate adapter like plastic boxes. All transformers will be identical with 6V (6VA w-shaped) or 8V (10VA toroidal) primary and 230V secondary coils and a special single turn current coil. Current coil load will be a copper wire with tunable length for producing currents up to 20A. In such a design the tester could be used as a single or three-phase voltage and current source with up to 3 transformer units. In case of single phase usage with at least 2 transformer units phase difference between voltage and current could be controlled.

The embedded software can be used in both single and three phase variants with 1, 2 or 3 transformer units. If test bench is used as single phase with 1 transformer unit or three-phase with 3 transformer units, phase difference between voltage and current could not be controlled. The only scenario to control phase difference between voltage and current is to use the test bench as a single phase source with at least 2 transformer units. In such a case phase A will be used as voltage source and phase B as current one.

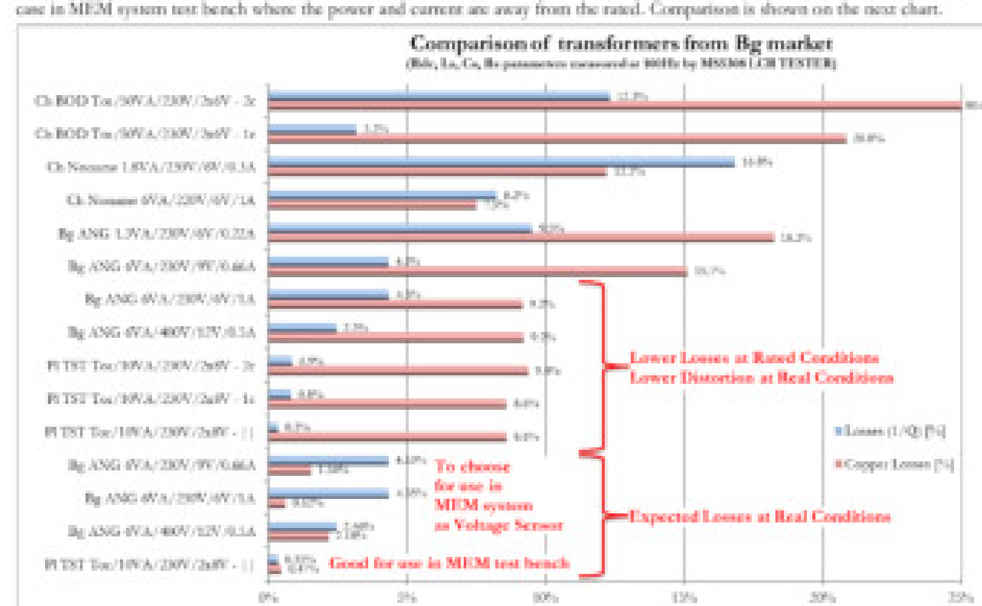
It is planned in future if have enough memory resources in Atmega32u4 AC multi-meter to be replaced by built-in metering part based on Microchip's ATM90E32AS IC. In such a case voltage transformers will be replaced by resistive dividers. Other possible feature could be added is serial connection to tested MEM system for sending data and command for calibration.

While trying to finish 3rd test bench version (putting it in a box) a number of new problems were found (mainly amplifier excitation because of noise coming from the potentiometer). The idea to replace it with a digital one was considered and extended with the idea to add secondary even third simulation channel. In such a way the voltage and current can be simulated separately and it will be possible to implement a phase difference between both. It will also be possible in case of 3 channels to simulate 3-phase mains. The software was successfully modified and 16x2 character LCD and rotary encoder were added for parameters setup. While adding the second channel for separate current simulation Mastech MS5308 LCR tester was found and some transformers offered on Bg market was measured and compared. The difference in measured and calculated Q is not investigated. The results are shown on the next table.

Parameters of transformers from Bg market (R_{DC}, L_s, C_s and R_s parameters measured at 100Hz by MS5308 LCR tester)

[illegible]

The big difference for BOD coils was not investigated. The red colored columns present the extrapolated computations to real use case in MEM system test bench where the power and current are away from the rated. Comparison is shown on the next chart.



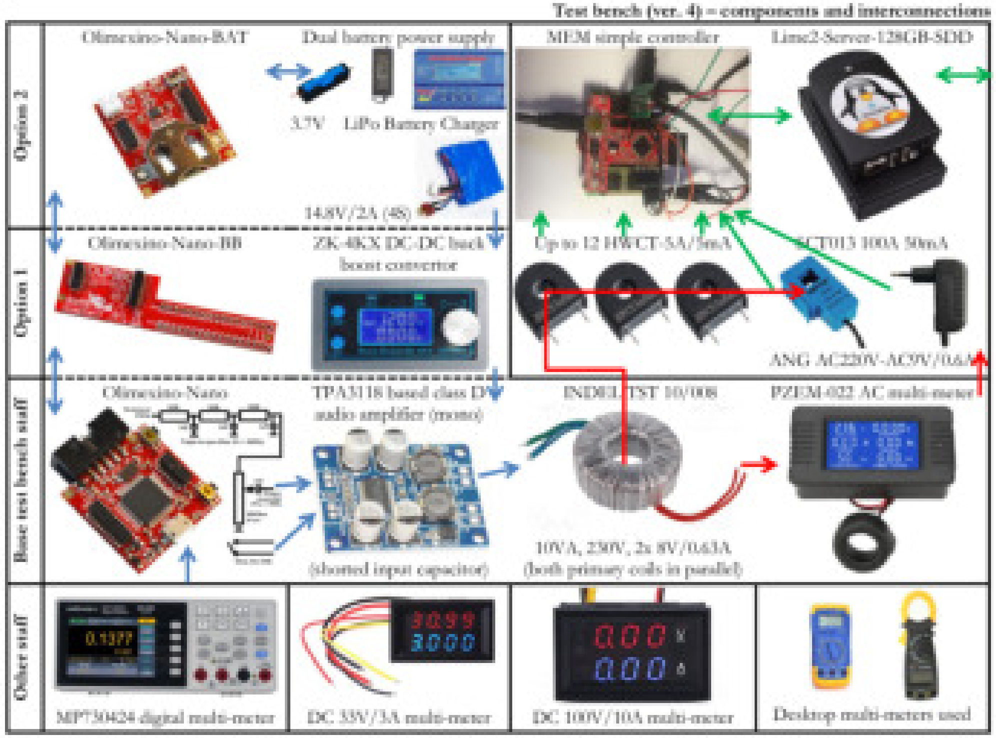
The conclusion is that the transformer TST 10/006 (toroidal, 10VA, 230Vac on a primary and 8Vac/0.63Aac on any of both secondary coils) produced by Polish company INDEL has the lowest losses but is relatively expensive. The transformers were tested in real conditions in which the secondary coil(s) are connected as a load to the 60W class D mono audio amplifier based on TPA3118 IC. In case of TST 10/006 both secondary coils was connected in parallel because their parameters are very close to each other.

The result of the tests is that the transformer TST 10/006 has the smallest distortion of the waveform for voltages higher than 240Vac on a primary coil and is the best candidate for usage in both voltage and current simulation channels of the test bench. Three of the transformers produced by Bulgarian company ANG are good candidates for usage as voltage sensor in MEM system. Till now ANG transformer 6VA/230V/9V/0.66A was used in tests of MEM system prototype and the observation is that the distortions of the waveform for voltages higher than 240Vac are relatively small. It could be replaced without any problems by ANG transformer 6VA/230V/9V/0.66A but adapters with 6VA/400V/12V/0.5A have to be ordered as a special product and the advantages must be assessed carefully especially if the price is bigger.

MEM system and test bench – precision assessment

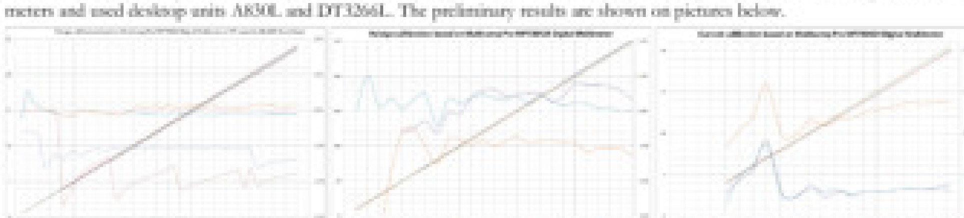
Before precision assessment some changes were made in test bench. DC-DC convertor was changed to ZK-4KX DC-DC buck boost one with precise tuning and measurement features. Peacefare's PZEM 022 AC digital multi-meter was added for precise measurement of mains simulation (voltage, frequency, current, power and power factor). The other change is in current coil. The wire was changed with magnet wire with 3.15mm diameter and part of its length was planned for current tuning. For nonfunctional part of the current coil is used 10mm² wire. All test bench staff is planned to be mounted on two aluminium or plastic panels.

Test bench (ver. 4) – components and interconnections



In addition to above changes it was planned to add AC measurement staff to the Olimexino-Nano firmware based on original emonLibCM library, LCD display, additional coil to the toroidal transformer (for voltage measurement) and precise LEM LTS 15-NP current transducer. In the final test bench version 4 this entire staff will substitute functionality of PZEM-022 AC multi-meter.

It was tested some calibration procedures based on class 1 Multicomp Pro MP730424 digital multi-meter. It is able to measure AC voltages precisely but the current measurement is up to 10Aac and requires high voltage. That is why the current measurement was done by using LTS 15-NP current transducer for 15A nominal AC/DC current, 0.2% accuracy and 0.1% linearity. In addition to the test bench output voltages and currents ZK-4KX convertor parameters were also measured together with some DC digital multi-meters and used desktop units A830L and DT3266L. The preliminary results are shown on pictures below.

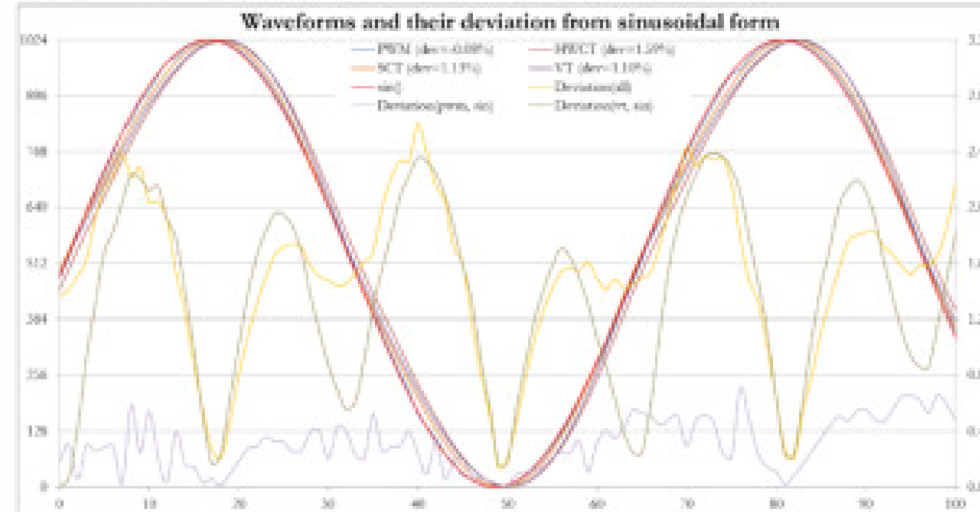


Interesting in the above measurement is that ZK-4KX convertor and DC 33V/3A panel digital multi-meter are precise in almost all range. ZK-4KX has good setup and tuning features as well. Unfortunately, PZEM-022 AC multi-meter is not precise enough so its substitution with Olimexino-Nano firmware based measurement is highly recommended.

The main conclusion from this phase is that for the final test bench and MEM system calibration other more precise and appropriate multi-meter like Megger's DCM2000P power clamp meter should be used. It is recommendable in addition to the mains true RMS parameters to measure total harmonics distortion, power, crest and distortion factors and other helpful once.

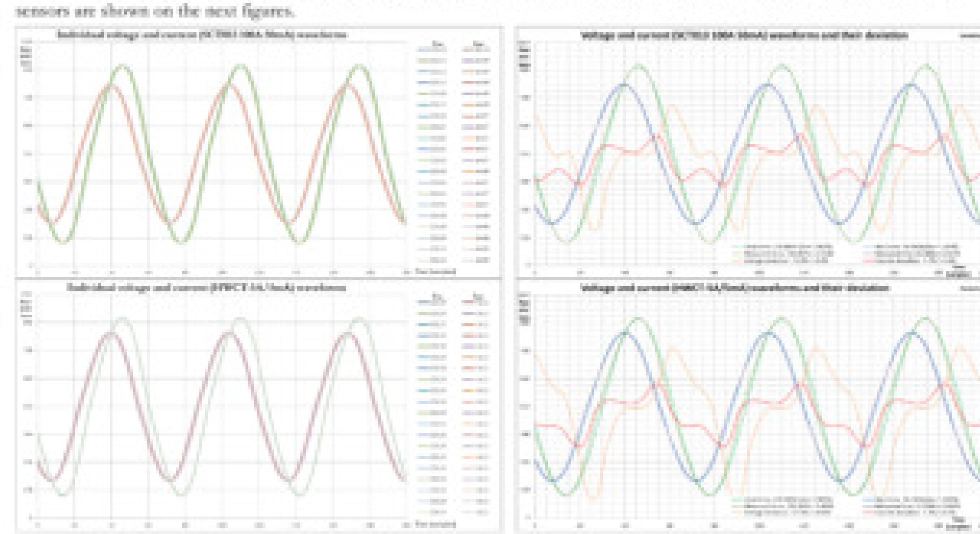
MEM system – voltage / current sensing and waveform distortion assessment

This assessment was accomplished with the help of MEM system and its test bench (ver. 3) with modified emonLibCM library (Version 2.2.2 15/9/2022) including the raw data capturing. Because of memory restriction only captured signals were measured (the voltage and a single current from the tested CT). The captured results from 20 measurement cycles were averaged and compared with the ideal sinusoidal waveform. The assessment was done off-line in Excel 2010 calculating RMS (Root Mean Square), CF (Crest Factor), PP (Peak-to-Peak) and other values. Deviation was calculated for both the waveforms (dev. = X_i / S_i – 1 in %) and the total values (dev = CF / $\sqrt[2]{2}$ – 1 in %). The waveforms as normalized raw data and their deviations from the sinusoidal form of the modulated PWM (the test bench generator), the voltage transformer (VT) and the current transformers (SCT013 100A 50mA and HWCT-5A/5mA) are shown on the next figure.



The total voltage and current deviations are in the interval from 1% to 1.6% while the generator's one (modulated PWM) is 0.08%. The relatively big dispersion of the signals is mainly caused by the discretization.

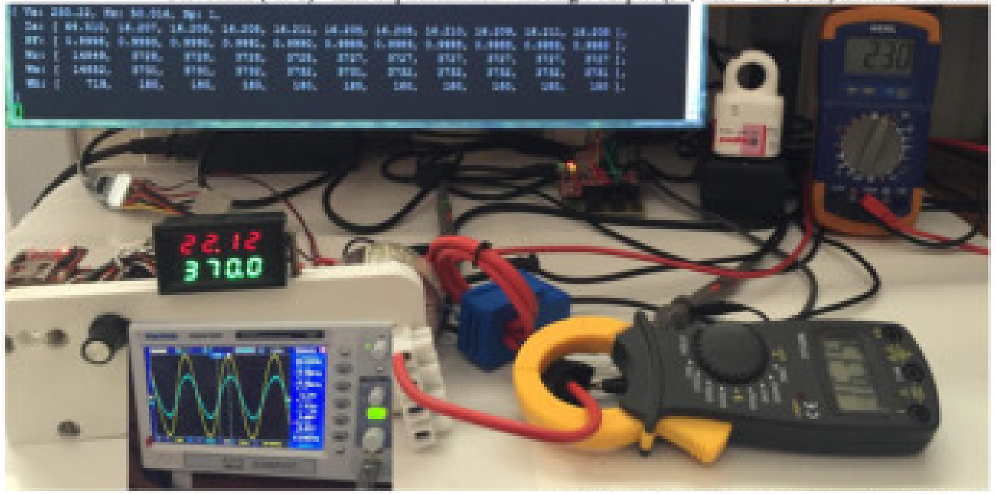
The individual and the averaged waveforms of the voltage and the currents for both SCT013 100A 50mA and HWCT-5A/5mA sensors are shown on the next figures.



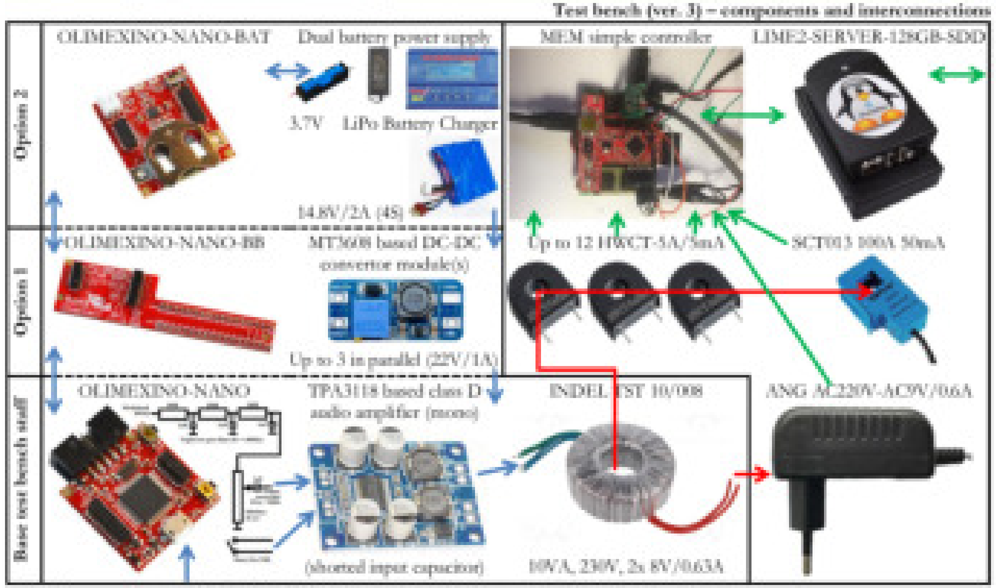
In addition to the waveform graphs calculated total RMS values and their deviations are shown. As it can be seen the latest version of the test bench grants extremely stable values for both the voltage and the currents (0.016% and less). Without matter that waveform deviations from sinusoidal form are relatively big (up to 17% at given individual points) the total deviations of the voltage and the currents are less than 1.15%.

MEM system – test scenario 3rd version final implementation

Test bench (ver. 3) – final implementation with single adapter (5V/1.5A + 12V/1.5A) instead of batteries



Test bench (ver. 3) – components and interconnections



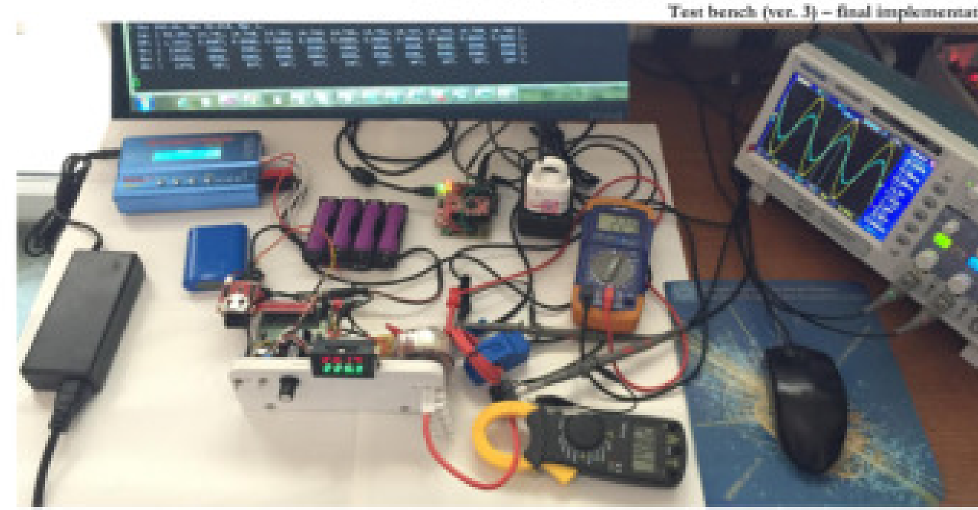
Notes about options of the final implementation:

- Dual battery supply is optional and can be replaced by external USB power supply for Olimexino-Nano and 12-15V/2A external power supply for DC-DC module(s);
- It also possible to replace the DC-DC module(s) with 20-24V/1A external power supply for the audio amplifier;
- The other option is to use dual voltage power supply (5V/1.5A & 12V/1.5A) instead of separate ones;
- The cheapest option is to use dual voltage power supply (5V/1.5A & 24V/1.5A) without DC-DC convertor.

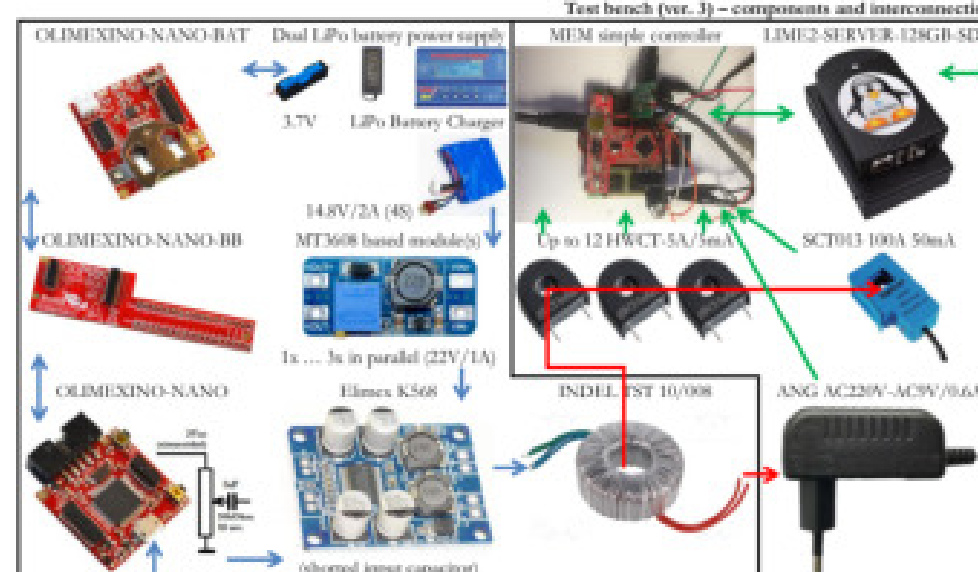
In case of single power supply special attention has to be paid to the common ground to prevent the influence of the high starting current of the audio amplifier on the MCU. The main 12V supply and the common GND should be connected to the DC-DC convertor. Voltage and current measurement device can also be connected between the power supply and the other boards. The output 22V of the DC-DC convertor and its GND should be connected to the audio amplifier. The common GND of the audio amplifier boards should be connected to the MCU board together with its output sinusoidal signal. Only the 5V of the power supply must be connected to the MCU board while the common GND will be taken via the audio amplifier board.

MEM system – test scenario 3rd version final implementation

Test bench (ver. 3) – final implementation



Test bench (ver. 3) – components and interconnections



Notes for final implementation:

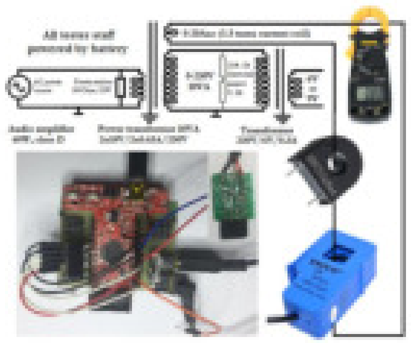
- Dual battery supply is used for separate powering of both Olimexino-Nano and audio amplifier;
- 4x 18650 LiPo batteries connected as 4S pack is provided with intermediate wires for balance charging with Imax B6 charger;
- BB-PWR-3608 is replaced with up to 3x connected in parallel MT3608 based modules with fine tuning of the output voltage;
- Using bigger power supply for the audio amplifier is required for avoiding signal distortion and reaching 250Vac values;
- Multi-turn potentiometer is used instead of linear one and additional load resistor at the amplifier output was removed;
- Input capacitor of the amplifier is shorted and external one with bigger capacity is added to avoid signal distortion.

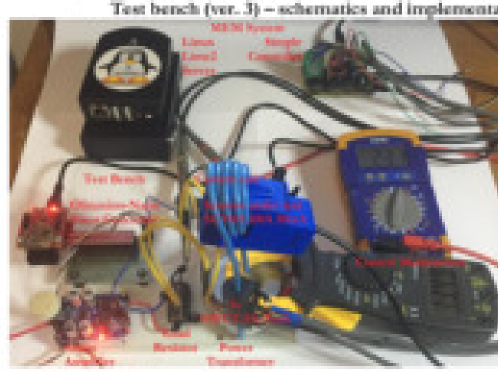
Unfortunately, the shortcut current depends on both wire temperature (require time to stabilize) and number of the strung current sensors (require adjustment by the multi-turn potentiometer or even by the wire length). Testing of both voltage and current sensors can be done separately using different voltage and current values of the transformer secondary coils. Calibration coefficients can be calculated individually by modified software for the controller of the MEM system and stored in nonvolatile memory.

Dual battery supply and DC-DC module(s) are optional and can be replaced by USB power for Olimexino-Nano and 20-24V/1A external power supply for the audio amplifier.

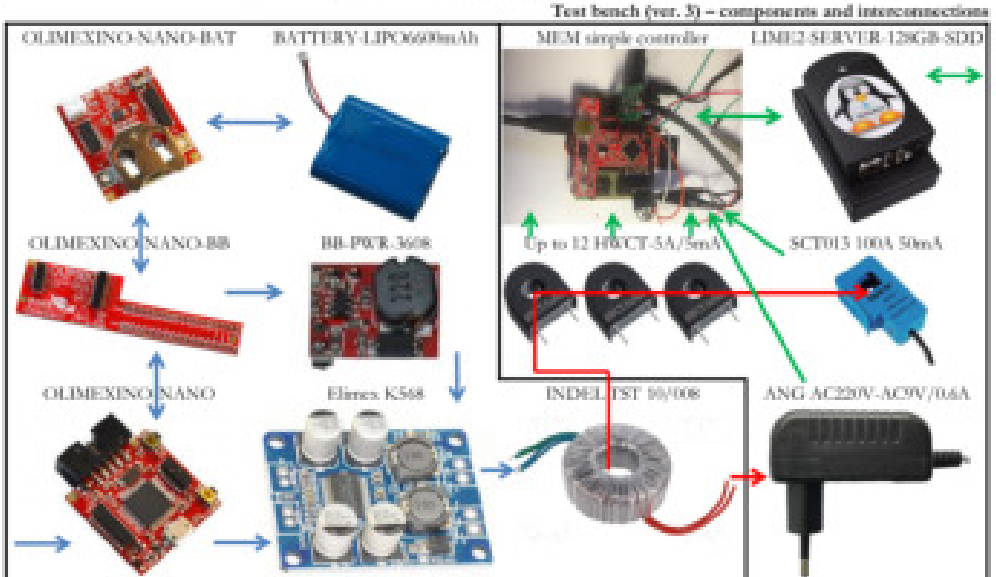
MEM system – test scenario 3rd version implementation

Test bench (ver. 3) – schematics and implementation





Test bench (ver. 3) – components and interconnections



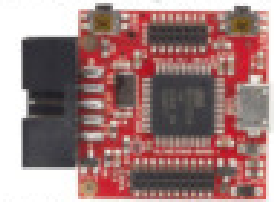
Notes:

- BATTERY-LIPO6600mAh is optional for granting complete isolation of the test bench;
- BB-PWR-3608 can be powered by 5V_USB or optionally by VBAT2 of OLIMEXINO-NANO-BB;
- OLIMEXINO-NANO-BB is stacked with a universal proto-board where low pass filter is mounted;
- Linear potentiometer 10k is mounted at the output of the low pass filter to regulate input voltage of Elimex K568
- Power resistor 390Ohm/25W is mounted in parallel to the primary coil of the transformer;
- Resistive divider with different ratios (43, 230 etc.) is mounted to the terminal block for secondary coil interconnections;
- Resistive divider GND has to be connected to OLIMEXINO-NANO and Elimex K568 GND if oscilloscope is used;
- High voltage of the transformer secondary coil has to be well isolated except the resistive divider GND if connected ;
- Current coil has to be made by a single 4mm² wire with isolation and shorted by appropriate terminal block;
- Wire length and turns of the current coil has to be tuned to reach $16A_{AC}$ at $230V_{AC}$ on the transformer secondary coil;
- Additional 4-5 turns has to be made from current coil wire outside the transformer for SCT013 100A 50mA sensor;
- Optionally oscilloscope and multimeters like DSO5102P, A830L and DT3266L can be used for monitoring electrical quantities;
- Control equipment with higher precision and individual calibration of the sensors is recommendable if better accuracy is required.

The current coil suffers from some drawbacks. When the current sensors are strung, the current on it decreases. For reaching higher current values, voltage on the amplifier input has to be increased. Unfortunately, the voltage on the secondary transformer coil will also grow up and reaching inadmissible values can damage the voltage sensor and the controller of the MEM system. To prevent any damages voltage sensor can be switched off. The limitation of the voltage and current values is the responsibility of the test bench operator. On the other hand shortcut current will fall down because of heating the wire therefore stabilization has be awaited

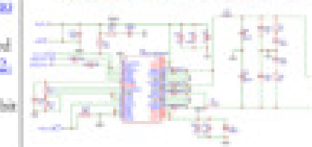
MEM system – test scenario 3rd version

Sinusoidal waveform generator 50/60Hz



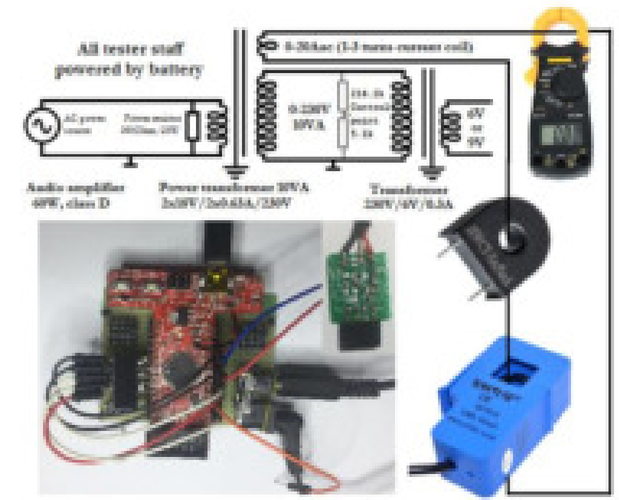
The hardware is based on Olimexino-Nano and includes a triple low-pass filter (Fc 796Hz). The software is based on modified version of FrequencyGenerator ver. 1.00.2. 5.21 written by Rick Groome. The waveform is created by 625 points with 9-bit resolution fast PWM on timer4 of ATmega32U4.

Audio amplifier 24V/60W, class D, mono

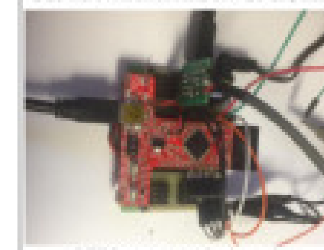


10VA 230V/2x8V/2x0.63A power transformer used as step-up

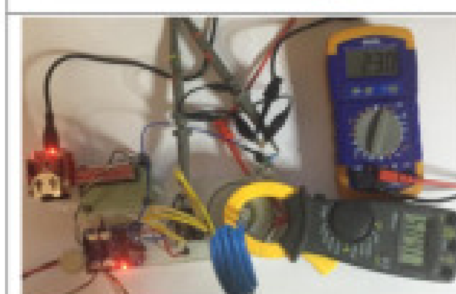
Voltage sensor under test: W-shaped transformer 1.8VA/230V/6V/0.3A or AC-AC adapter 230V/9V/6VA



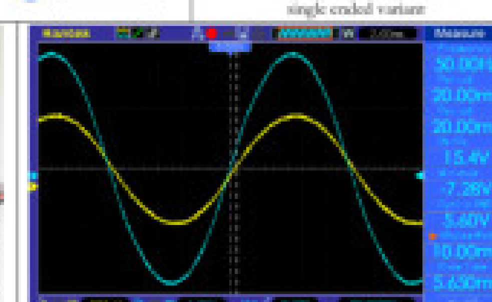
Current sensors under test: SCT013 100A 50mA and HWCT-5A/5mA



MEM system simple alternative single ended variant



Test scenario ver. 3 – amplifier loaded with power resistor (390Ohm, 25W) and the transformer primary coil



Oscillogram of shortcut current & HWCT & 47Ohm (channel 1, yellow) and output AC voltage divided by 43 (channel 2, blue)

Measured parameters with a triple low-pass filter (Cut-off frequency 796Hz) between the generator and the amplifier: generator frequency 50.00Hz; amplifier input up to $3V_{P-P}$; amplifier output $15.4V_{P-P}$ (on each arm @ $20V_{DC}$ and $1.1V_{P-P}$ input); transformer primary voltage $10.9V_{RMS}$ ($30.8V_{P-P}$); transformer secondary voltage $230V_{RMS}$ ($650V_{P-P}$); shortcut coil current $16.05A_{RMS}$ (3.16mOhm @ 1turn, 74cm 4mm² wire); main consumption 1.1W in idle mode ($0.06A_{DC}$ @ $20V_{DC}$); shortcut coil consumption 0.8 W ($16.05A_{RMS}$ @ 3.16mOhm); load resistor consumption 0.3W ($10.9V_{RMS}$ @ 390Ohm) and total power consumption 2.2W (0.11 A_{DC} @ $20V_{DC}$ or 0.09 A_{DC} @ $24V_{DC}$). Total power consumption is 3.0W (0.15 A_{DC} @ $20V_{DC}$) when MEM system voltage transformer is switched on.

MEM System – summary for both variants of the simple alternative in pictures and graphs

After performing series of tests and modifications of emonLibCM library (Version 2.2.2 15/9/2022) for the both variants (single ended and differential sensor connections) of MEM system (simple alternative) following results could be announced:

Pin assignment, main emonLibCM library modifications and test scenario

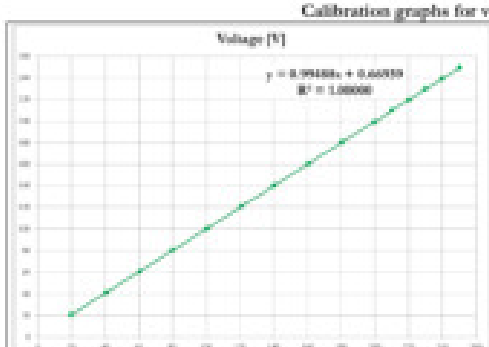
Printouts from both single ended and mixed and interrupt routine timings for mixed variants

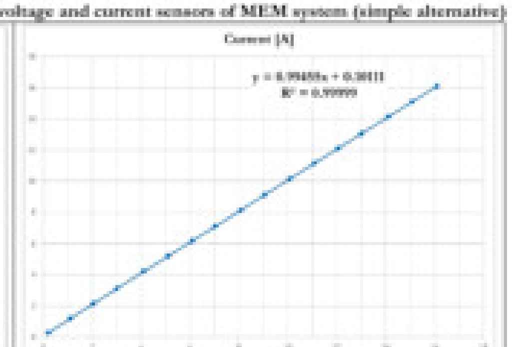
Regular printout (over USB serial) in JSON format for all measurement points (1 voltage and 11 currents) – single ended inputs variant

Regular printout (over USB serial) in JSON format for all measurement points (1 voltage and 13 currents) – mixed inputs variant

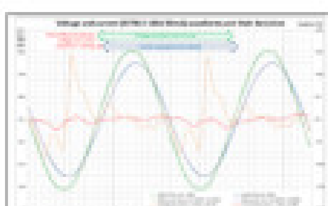
Oscillogram of interrupt timing for the modified emonLibCM (mixed variant)

Calibration graphs for voltage and current sensors of MEM system (simple alternative)

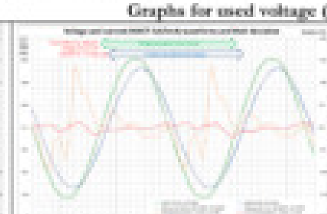




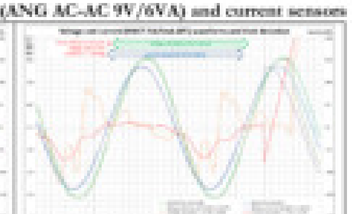
Graphs for used voltage (ANG AC-AC 9V/6VA) and current sensors



SCT013 100A 50mA current sensors over single ended connections



HWCT-5A/5mA current sensors over single ended connections

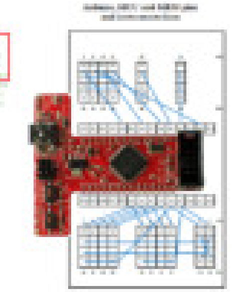


HWCT-5A/5mA current sensors over differential connections

The next step is the main and sensor boards to be (re-)designed for the final variant of the MEM system (simple alternative). The choice has to be made is if to use Olimex AVR-T32U4 board as-is or to modify it for a single main board solution. The modified board design is preferable because battery power and charging options can be added, which in combination with additional wireless module (like MOD-WIFI-ESP8266) will make its standalone usage possible.

MEM System (simple alternative) – problem with wrong calculations at mixed sequence

More tests were done with 14 sensors (1 VT and 13 CT) with mixed connection types (6 single ended and 8 differentially). Single ended inputs (1x VT, 1x SCT013 100A 50mA and 4x HWCT-5A/5mA sensors) and differential inputs (8x HWCT-5A/5mA sensors) share 5 real sensors (1x VT, 1x SCT013 100A 50mA and 3x HWCT-5A/5mA).



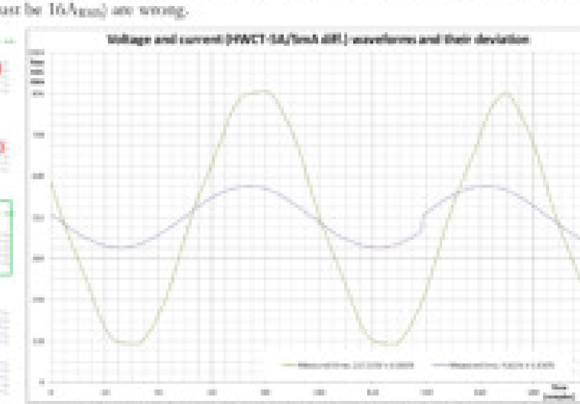
Interrupt routine related code added to the original emonLibCM library

Original emonPi with only 2 current inputs (Robert Wall)

Modified emonLibCM library (Version 2.2.2 15/9/2022)

As can be seen, the additional code takes 3us. It is also seen the part of the scan sequence with the last single ended input (conversion time 104us) and the first differential one (conversion time 112us). The times at current readings left for the main loop are 42us and 50us instead of 62us but there are no any side effects from that. The time distribution in the interrupt routine is quite similar to that from the original emonLibCM library. It can also be seen doubling and discarding of calculation for the first differential input. There is no observation for overlapping of ADC interrupts at all. These facts suggest that the additional code is not the cause for the wrong calculations as shown on the next left picture. There are also problems with debug scenario if capturing waveforms like in the next right picture where the shapes and current values (must be $16A_{RMS}$) are wrong.

Printouts vs ADCDuration and condition of the code for doubling the first differential input scan



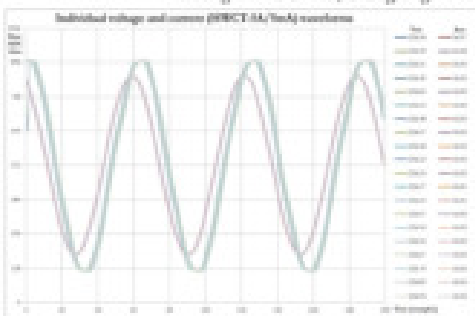
Averaged voltage and current waveforms for differential input (from 20 individual scan sets 96 point each taken sequentially)

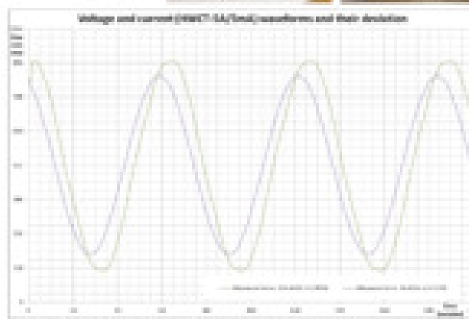
The problem with wrongly calculated figures is solved if ADCDuration value is set to 117us instead of 104us or 112us (derived empirically). It is also empirically derived (but not logically explained) that for the right functioning the code for doubling the first differential input scan have to use following condition: (channel_doubled && (doubled_channel == sample_index)). It can also be seen in the oscillogram (but not logically explained) that the first 112us conversion interval (coming from differentially connected CT) is from regular interrupt routine execution while the second one is from skipped interrupt processing.

MEM System (simple single ended and differential alternative) – history in pictures and graphs

After tests with sequence of only single ended inputs (1 voltage and 11 current sensors) two current sensors (HWCT-5A/5mA) connected to differential inputs is setup. One current sensor (SCT013 100A 50mA) is also connected to a single ended input. Other inputs are connected to the sensors as well. All (1 voltage and 13 current) sensors are scanned in a sequence as shown on a picture below. Preliminary printouts, test scenario and waveform graphs are shown on the pictures below.

Regular printout (over USB serial) in JSON format
for all measurement points (1 voltage and 13 currents)
wrong values in red, debug in green





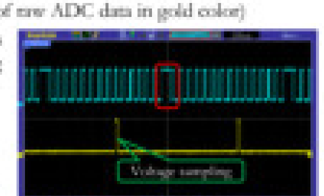
Results for voltage (ANG AC-AC 9V/6VA) and current (HWCT-5A/5mA connected to differential input) sensors (not complete)

Printout (in green) and changed emonLibCM parts: ADC MUX values setup in red, code to double the first channel in sequence after single ended to differential input change in blue and recalculating of raw ADC data in gold color)

Wrong calculation at switching from single ended to differential inputs is encountered. The reason is explained in ATmega32U4 datasheet (24.5 Changing Channel or Reference Selection).

Special care should be taken when changing differential channels. Once a differential channel has been selected, the stage may take as much as 125μs to stabilize to the new value. Thus conversions should not be started within the first 125μs after selecting a new differential channel. Alternatively, conversion results obtained within this period should be discarded.

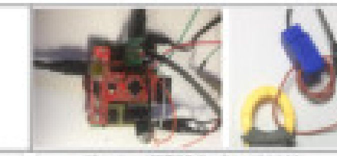
The same settling time should be observed for the first differential conversion after changing ADC reference (by changing the REFS1:0 bits in ADMUX).

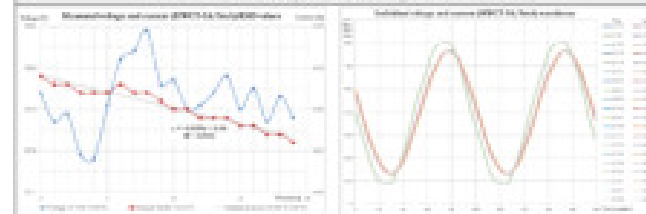


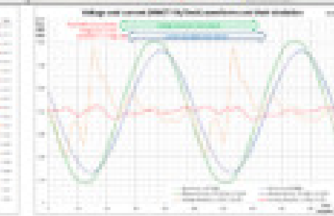
To avoid switching time problem the sampling of the first differential input is doubled and wrong values are discarded as shown on the oscillogram in red. Unfortunately, other values were wrongly calculated (mains frequency and PF – some zero, other greater than one) as shown on the picture above. Voltage and current values are accurate.

MEM System (simple single ended alternative) – history in pictures and graphs

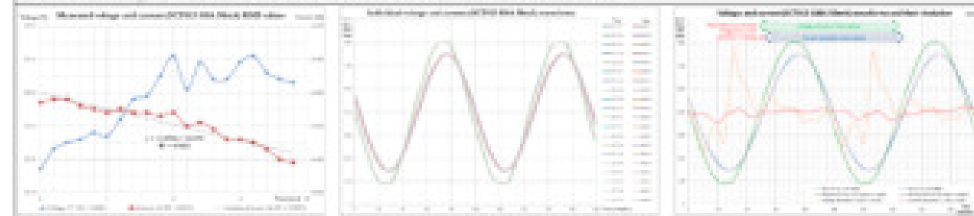
Regular printout (over USB serial) in JSON format
for all measurement points (1 voltage and 11 currents)



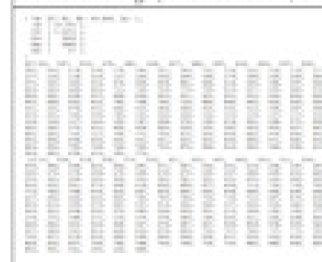




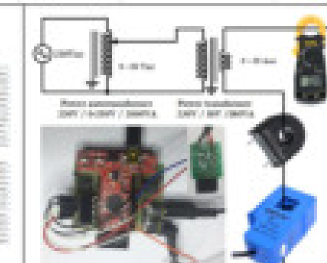
Results for voltage (ANG AC-AC 9V/6VA) and current (HWCT-5A/5mA) sensors



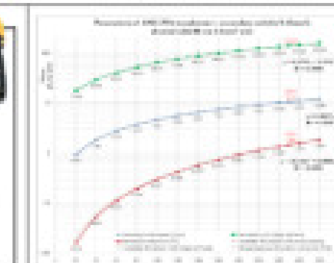
Results for voltage (ANG AC-AC 9V/6VA) and current (SCT013 100A 50mA) sensors



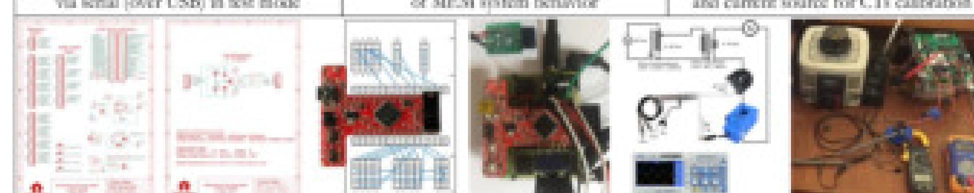
Printout of MEM system firmware via serial (over USB) in test mode



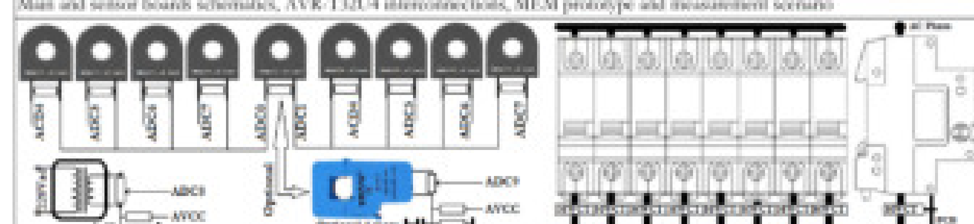
Test scenario for investigation of MEM system behavior



Test for using of VT as both voltage sensor and current source for CTs calibration



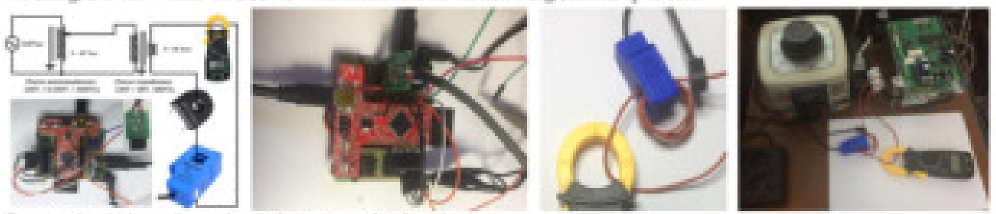
Main and sensor boards schematics, AVR-T32U4 interconnections, MEM prototype and measurement scenario



Initial idea: up to 9 current differential measurement points ($16A_{AC}$ each), total I_{RMS} ($64A_{AC}$) and V_{RMS} ($230V_{AC}$)

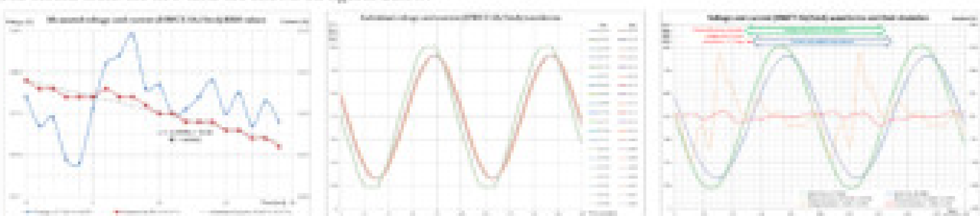
MEM System (waveform distortion and voltage / current sensing) – continued

The tests to investigate the influence of the waveform distortion over the measurement precision were continued with some changes and expansion. The voltage sensing input resistor divider was changed from 20k/1k to 12k/1k which increase the signal amplitude to 2.5Vp-p. HWCT-5A/5mA sensors was also connected via a sensor board to the system and investigated. Current simulation scenario was changed to use 4 winds for SCT013 100A 50mA sensors for simulating currents up to 80A.

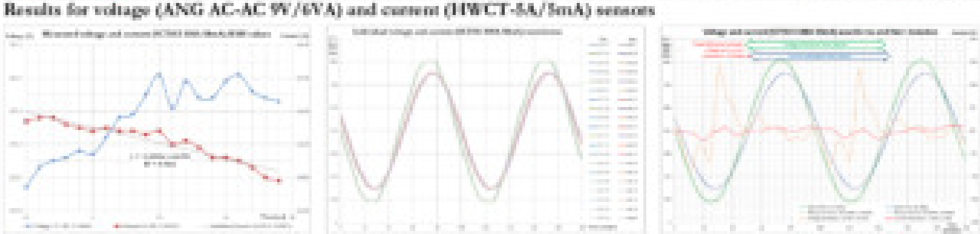


Current simulation schematics and test scenario after changes

The results from the new tests are shown on figures below.



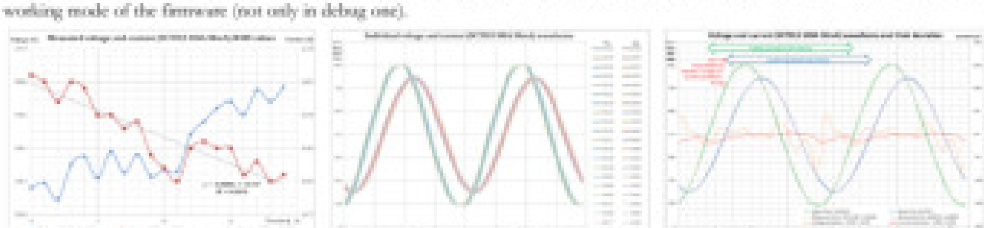
Results for voltage (ANG AC-AC 9V/6VA) and current (HWCT-5A/5mA) sensors



Results for voltage (ANG AC-AC 9V/6VA) and current (SCT013 100A 50mA) sensors

Unfortunately, it is unexplainable increase of deviation figures for both voltage (from -5.0%/+4.7% at the first tests to -8.5%/+14.5% and -9.6%/+13.9% at the next tests) and current (from -0.1%/+0.6% at the first tests to -2.3%/+0.8% and -2.0%/+1.3% at the next tests) waveforms. Fortunately, the correlation between voltage and current waveforms deviations remains approximately the same.

One more test was done with changes of the ADC frequency from CLK/128 to CLK/64 and CLK/32. While the firmware works fine at CLK/64 it stops working in normal way at CLK/32. At the test with CLK/64 (CLK/128 at debug mode) emonLibCM library was extended at the same time to scan all 12 ADCs and use 2 buffers with length of 64 (200 at debug mode) raw data each for capturing samples of voltage and one of the current inputs. These changes are optimal for the system to work normally and not to exceed the limits of the size of the used memory. The sketch was changed to send always via USB serial both captured waveform data and the measurement results of all channels (11 current and a voltage). The idea is to have opportunity to observe waveforms at normal working mode of the firmware (not only in debug one).



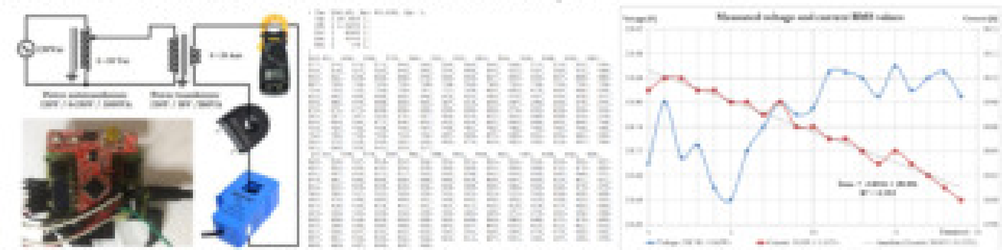
Results for voltage (ANG AC-AC 9V/6VA) and current (SCT013 100A 50mA) sensors at ADC frequency CLK/64

The result is decreasing of waveform data points from 96 to 32. As it can be seen distortion and saturation at voltage waveform is still visible but more tests should be done in case of bigger deviations from sinusoidal shape. On the other hand precise assessment should be done for influence of the bigger ADC frequency over the measurement precision in general. In case of negative influence waveforms capturing will be used at debug mode only.

MEM System (waveform distortion and voltage / current sensing)

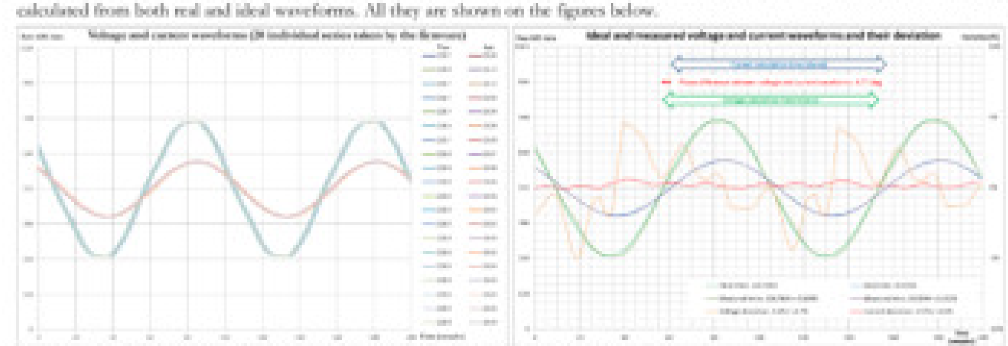
A new test to establish the influence of the waveform distortion was planned and executed. The test scenario scheme is shown in the figure below. ANG standard AC-AC adapter with 6VA transformer (input: 220-230Vac, output: 9Vac/0.66Aac) is used as voltage sensor. It gives 11.1Vac on the secondary coil at 230Vac on its primary coil. Olimex SNS-CURRENT-CT013-100A sensor is connected to CT0 input of the MEM system measuring total current.

Voltage and current values (200 samples each) are captured by a special addition to Robert Wall's emonLibCM library and compared in Excel 2010 with ideal sinusoidal waveform. The printout from the firmware via serial (over USB) output is shown in the figure below. Individual V_{RMS} and I_{RMS} values are calculated by the firmware and printed in the beginning of each row. Measured by the firmware RMS values in time are presented in the graph below and shows tendency I_{RMS} to fall in time probably depending on the temperature of the transformer with shortcut coil which will influence over calibration process.



The power schematics, the firmware output in test scenario and measured voltage and current values in time

Real voltage and current waveforms are averaged from 20 different takes 200 samples each captured one after the other and synchronously. Sinusoidal waveforms are calculated to best fit real once. Real and ideal RMS values and waveform deviations are calculated from both real and ideal waveforms. All they are shown on the figures below.



Individual, ideal and measured voltage and current waveforms and their deviations

As can be seen from the first graph the voltage curves have significant distortion while the current once are much closer to the ideal sinusoidal shape. The same can be seen even better in the second graph, where the averaged real waveforms are compared with the sinusoidal shapes. Deviations between real and ideal waveforms are also calculated and shown for voltage (-5.0% / +4.7%) and current (-0.1% / +0.6%) waveforms. It is clear that because of lower distortion and saturation lack the current waveform has more than one order of magnitude smaller deviation than the voltage one.

Calculated voltage and current RMS values are 228.781V ± 0.029% and 20.059A ± 0.152% respectively. Deviations of measured values mainly depend on mains voltage variation and transformer heating up. Time intervals for RMS calculations are also shown on the second graph. The phase difference between voltage and current waveforms is 4.77 degrees (4 samples by 208us) and mainly depends on the used AC-AC adapter and transformers for current simulation.

The conclusion from previous test and recommendations from ANG experts can be summarized that it is better to use ANG 6VA transformer as both voltage sensor and current source for calibration of CTs. The primary coil has to be for nominal voltage 230Vac. It has to have two secondary coils – one for 9V/10-20mA and other with 5-10 turns wind with 1.12mm wire (thicker is preferable). Long enough 1.5mm² wire with PVC insulation will be used to reach 16A shortcut current thanks to its resistance and for stringing up CTs. Saturation power can be as small as possible for avoiding transformer heating up.

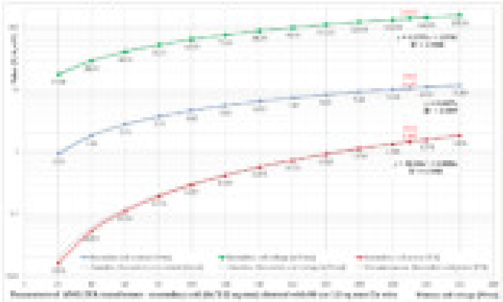
Following conclusions can be made:

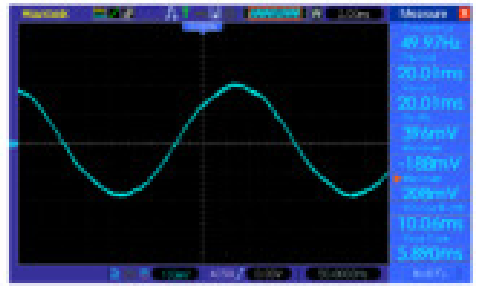
- Olimex SNS-CURRENT-CT013-100A sensor is precise enough for measuring of total currents up to 100A;
- ANG standard AC-AC 230V/9V/0.66A adapter with 6VA transformer is relatively good as voltage sensor;
- Resistive divider for the voltage input should be changed to increase signal amplitude for better precision;
- ANG 6VA transformer with additional secondary coil can be used as current source for CTs calibration;
- Calibration process should be short enough to avoid transformer temperature influence over the current.

Olimex SNS-CURRENT-HWCT-5A-5MA current sensor has to be tested as well. This test has to be repeated with a special ANG 6VA transformer prepared for MEM system usage as voltage sensor and CT calibration current source.

MEM System (voltage sensing and current transformers calibration)

Test with ANG special 2VA transformer (5600 turns x Φ0.07mm / 6 turns x Φ1.12mm) at 230V on a primary coil gives 19.8A on a shorted with 10cm. 1.5mm² wire and 10.43A if shorted with 88cm 1.5mm² wire(10.032mOhm theoretically). Measured voltage on a wire ends was 140mV$_{RMS}$ (396mV$_{P-P}$) which gives 1.46VA power at 230V$_{RMS}$ on a primary coil. Shortcut current waveform and parameters graph are shown on pictures below.





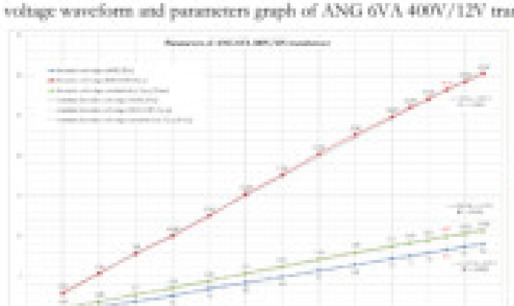
Shortcut current waveform

The waveform is relatively good and probably acceptable for a simple alternative of MEM System. Strange is the phase difference at AC50 synchronization (-37.8 degree) which has to be investigated. Calculated power of 1.46VA at 230V$_{RMS}$ on a primary coil and 10.43A if secondary coil is shorted with 88cm 1.5mm² gives potential for combination of both voltage sensing and current source for CTs calibration on a single transformer. The main idea is to have two secondary coils. The first will be used for voltage sensing giving 9V$_{RMS}$ for example. The second coil will be used as a current source for CTs calibration. It could be wound with 1mm² copper wire and working temperature up to 105°C (optionally with PVC isolation). The wire length of 1m for example will be enough to reach total resistance 10-20mOhms. Some of the wire will be used to wind 5-10 turns as a secondary coil. Other part of the wire will be left free for stringing up the CTs and to short the circuit at calibration. Final length of the wire will be tuned at production process to grant fixed shortcut current (16A$_{RMS}$ for example) at nominal voltage on a primary coil (230V$_{RMS}$). On site calibration will be done with measuring of the voltage from each CT and the voltage of the secondary coil which will be used at calculation of the calibration coefficients. The calibration time has to be minimized to avoid temperature increase which will influence wire resistance.

Maximal current in Amperes vs. insulation material and copper temperature of the wire

AWG	Wire diameter [mm]	Wire cross section [mm²]	Polyethylene Neoprene Polyurethane Polyvinylchloride (Semi-Rigid)	Polypropylene Polyethylene (High Density)	Polyvinylchloride PVC (Irradiated) Nylon	Kynar (135°C) Polyethylene (Cross-linked) Thermoplastic Elastomers	Kapton PTFE FEP PFA Silicone	Enameled copper wire	Resistance [mOhm/m]
		Temperature	60°C	90°C	105°C	125°C	200°C		20°C
14	1.63	2.082	27.0	30.0	33.0	40.0	45.0	5.9	8.58
15	1.45	1.651	23.0	26.0	28.5	35.0	38.5	4.7	10.81
16	1.29	1.309	19.0	22.0	24.0	26.0	32.0	3.7	13.64
17	1.15	1.039	17.0	19.5	21.0	25.0	28.0	2.9	17.19
18	1.02	0.824	15.0	17.0	18.0	20.0	24.0	2.3	21.68
19	0.91	0.653	12.5	14.5	15.5	17.0	20.5	1.8	27.34
20	0.81	0.519	10.0	12.0	13.0	14.0	17.0	1.5	34.42

For improvement of transformer characteristics as voltage sensor it is important to decrease distortion of the voltage waveform. A good idea is to use transformer with higher power (6VA for example) with primary coil for 400VRMS instead of 230V$_{RMS}$. Open coil voltage waveform and parameters graph of ANG 6VA 400V/12V transformer is shown on pictures below.

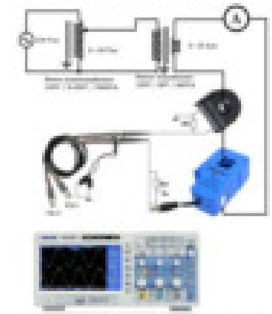


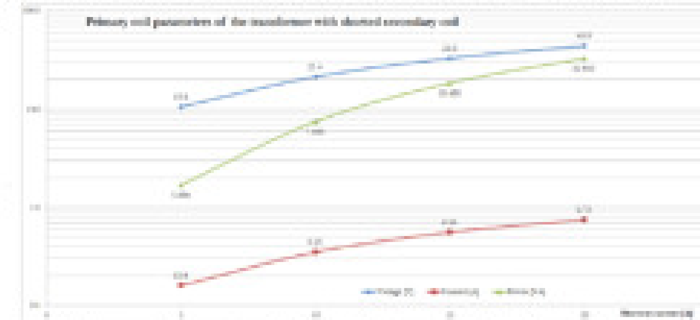
Open coil voltage waveform

The conclusion is that ANG special 2VA transformer has better secondary voltage waveform and lower saturation power. It makes this transformer preferable to 6VA one. The primary coil has to be for nominal voltage (230Vac). The special 2VA transformer has to have two secondary coils – one for 9V/10mA and other 5-10 turns wind with 1 or 1.5mm² wire long enough to reach 16A shortcut current thanks to its resistance at saturation power less than 1.5VA.

MEM System (precision improvement)

One of the ways to maximize precision of the MEM system is to make on-site calibration. The test scenario, where the CTs measure the shortcut current of secondary coil of a transformer, shows that it is possible to generate relatively high current at low power.





Test scenario and the result of measured primary coil parameters of 180VA transformer with shorted secondary coil

The idea for on-site MEM system calibration is to use voltage transformer (VT) with additional coil to generate big enough shortcut current. The shortcut wire can be used to string up the CTs, measure the current and recalculate calibration coefficients. One potential problem is the possibility the shortcut current to influence measurement coil voltage and has to be investigated. Fortunately, this influence can be measured and taken into account at calculation of the calibration coefficients.

Other influence over the system precision is the noise induced in CTs and connection cables. While nothing to do with CT itself the cable can be shielded and the location of the burden resistor and the filter has to be investigated.

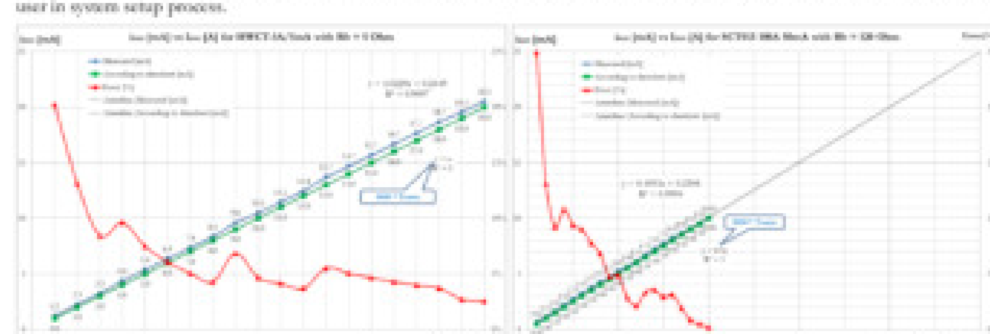

Sensor set variants

Sensor set CT:	HWCT-5A/5mA
Calibration coefficient:	21.28
Attached to MEM channel: #	
Current circuit name:	
lighting / kitchen / living room / parlour / bedroom 1 / bedroom 2 / bathroom 1 / bathroom 2 / boiler 1 / boiler 2 / oven 1 / oven 2 / refrigerator 1 / refrigerator 2 /	
other:	

Exemplary label

Without matter where they will be located all components (CT, the cable, the burden resistor and the filter components) can be mounted together and calibrated at production process. This is the main idea to develop separate sensor board carrying all relevant components. In addition recalibration can be made by the user and eventually periodically.

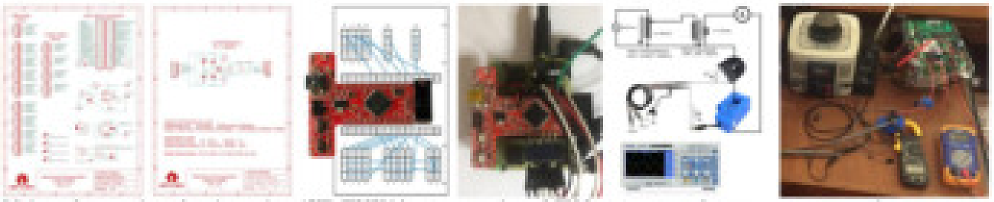
Individual calibration coefficient has to be printed on a label, fixed to the sensor set and used at MEM system on-site setup. The label can also be used to write down MEM channel and current circuit name attached to. All this information collected together will help the user in system setup process.


Calculation ratio and error for used CTs

Other big issue is the great error at measuring of small currents. This problem can be solved with using built-in operational amplifier to increase the signal from CTs. It is possible only if differential ADC inputs are in use. One drawback is the noise amplification but the influence over MEM system precision has to be investigated.

MEM System (simple alternative prototype and first tests with single ended ADC inputs)

The simple alternative of MEM system prototype as hardware is implemented with universal PCB (as a main board) stacked with Olimex' AVR-T32U4 board (Leonardo compatible). AVR-T32U4 board is preferred to Olimexino-Nano because all ADC inputs are free for use. Unfortunately, AVCC and AGND are not wired to the connector and it has no battery supply but these problems can be solved with adding its modified variant to the main board itself because it is OSHW licensed.



Main and sensor boards schematics, AVR-T32U4 interconnections, MEM prototype and measurement scenario

For testing is used scenario with 0-250V/2kVA autotransformer and 220V/18V/180VA transformer with secondary coil shorted. ANG AC-AC 9V/0.66A adapter is used as VT (connected to ADC13 and the oscilloscope channel 1) and a single SCT013-000 100A 50mA is used as CT (connected to ADC12 and the oscilloscope channel 2). Other 10 ADC inputs are connected to ADC12. Hantek's DSO5102P oscilloscope is used to observe waveforms. DT3266L multimeter with current clamp is used to measure short cut current.

Robert Wall's emonLibCM library and EmonTxV34CM_min.ino sketch are used for the first test. The main changes made in the sketch are to set pins 6-11 (ADC8-13) as inputs in setup function and to form appropriate JSON output to serial (over USB) channel. In emonLibCM.cpp except extending arrays from 4/5 to 11/12 members following essential changes (marked in red) were made:

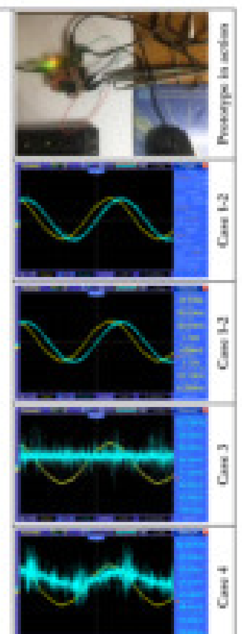
```
// <-- Sequence in which the analogue ports are scanned, first is Voltage, remainder are currents
static byte ADC_Muxbeq[max_no_of_channels+1] = {0x25,0x24,0x23,0x22,0x21,0x20, 0x07,0x06,0x05,0x04,0x01,0x00}; // added
static byte ADC_Sequence[max_no_of_channels+1] = { 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0}; // changed

// in function void EmonLibCM_interrupt() is added:
  if (ADC_Muxbeq[next] & (1 << MUX5))
    ADCSRB |= (1 << MUX5);
  else
    ADCSRB &= ~(1 << MUX5);
  ADMUX = ADCRef + (ADC_Muxbeq[next] & 0x1F);     // set up the next-but-one conversion

// instead of:
//  ADMUX = ADCRef + ADC_Sequence[next];     // set up the next-but-one conversion
```

The result sent via USB serial is:

```
EmonTx V3.4 EmonLibCM Continuous Monitoring Minimal Demo
Datalog period: 1.0 seconds
ADC input channels:
ch20: getLogicalChannel: 00, mux value: 0x25
ch01: getLogicalChannel: 00, mux value: 0x24
ch02: getLogicalChannel: 00, mux value: 0x23
ch03: getLogicalChannel: 00, mux value: 0x22
ch04: getLogicalChannel: 00, mux value: 0x21
ch05: getLogicalChannel: 00, mux value: 0x20
ch06: getLogicalChannel: 00, mux value: 0x07
ch07: getLogicalChannel: 00, mux value: 0x06
ch08: getLogicalChannel: 00, mux value: 0x05
ch09: getLogicalChannel: 00, mux value: 0x04
ch10: getLogicalChannel: 00, mux value: 0x01
ch11: getLogicalChannel: 00, mux value: 0x00
Case 1: Immediately after starting up at 5A
{ Vs: 227.76, Hz: 50.016, Sp: 1,
  Is: [illegible]
  PF: [illegible]
  Wr: [illegible]
  Wa: [illegible]
  Wh: [ 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0 ],
}
Case 2: Later on at 16A
{ Vs: 227.95, Hz: 50.016, Sp: 1,
  Is: [illegible]
  PF: [illegible]
  Wr: [illegible]
  Wa: [illegible]
  Wh: [illegible]
}
Case 3: Later on at 3A (second transformer switched off)
{ Vs: 223.49, Hz: 49.953, Sp: 1,
  Is: [illegible]
  PF: [illegible]
  Wr: [illegible]
  Wa: [illegible]
  Wh: [illegible]
}
Case 4: Later on at 0.15A (autotransformer moved to lower position)
{ Vs: 226.41, Hz: 49.953, Sp: 1,
  Is: [illegible]
  PF: [illegible]
  Wr: [illegible]
  Wa: [illegible]
  Wh: [illegible]
}
```



Problems found are marked in red. Wrong getLogicalChannel return result probably reflects to impossibility to setup some of the channels to scan (currently out of the scope of the development). Other problems are the big noise and wrong results in case 3. Used transformers in measurement scenario add big phase shift (~40°) compensated by calibration value (phaseCal_CT).

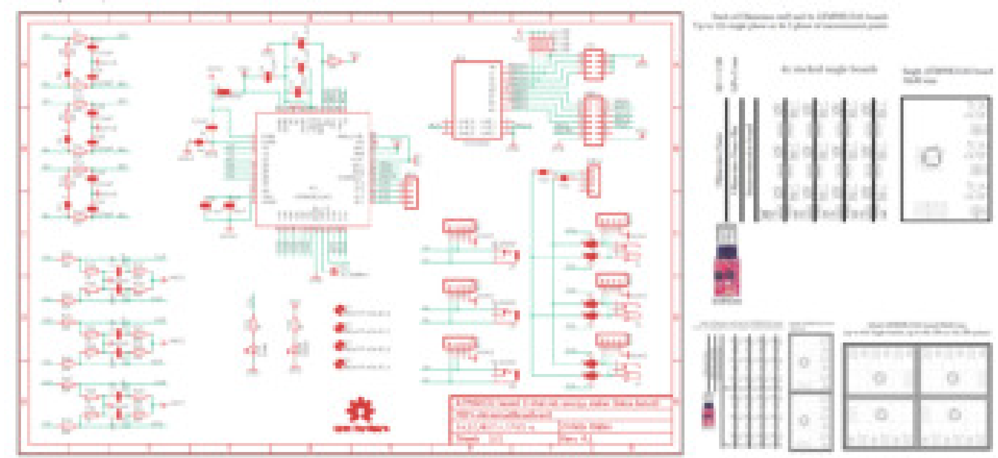
Multichannel Home Electricity Monitoring System (advanced alternative)

This alternative of the project is inspired by CircuitSetup's Expandable 6 Channel ESP32 Energy Meter based on Microchip's ATM90E32AS IC for precise metering of three-phase four-wire (3P4W, Y0) or three-phase three-wire (3P3W, Y or Δ) electricity consumption and ESP32 for communication. The solution consists of almost the same boards (base and add) with up to 2 voltage and 3 current inputs. Apart from the high price and strange 3 phase solution (to use 2 chips while single one supports that) it is very ineffective to have 2 almost the same boards with SPI connectivity. On the other hand it is inefficient to connect over audio jacks many CTs like SNS-CURRENT-HWCT-5A-5MA (offered by Olimex for 1€ instead of SNS-CURRENT-CT013-100A for 10€). For example for monitoring of my apartment with 9 local circuits (up to 16Aac each) and total input (up to 64Aac) the price is between $274.58 and $366.58 without VAT and delivery taxes. If add 115€ (without VAT and delivery taxes) for Olimex' OSHW Lime2-Server (minimal configuration with 120GB SSD) I have to pay approximately 800BGN without VAT and delivery taxes or something like 1000BGN (500€) total. The half of that price is for 2x 6 channel measurement boards and 9x YHDC SCT006. Provided I manage to save 10% (which is overrated) on my electricity bill this cost will be recouped in more than 6 years. This calculation does not include the cost of electrical appliances that must be replaced to actually reduce the bill. In conclusion, the purchase and installation of such a household electricity consumption monitoring system is more a matter of feeling in control of running costs mainly in view of the current crises.

For the above reasons, reducing the price and combining the monitoring system with other activities is imperative. A simple and lower in price alternative is discussed later on and it is based on Lime2-Server hardwired via USB with Olimexino-Nano, HWCT-5A/5mA CTs and OpenEnergyMonitor project (especially emonLibCM by Robert Wall and emonCMS). A simple interconnection board has to be developed and for the BOM of 150€ (including Lime2-Server, Olimexino-Nano, CTs, VT) the system can be used for additional home applications. The disadvantage is that such a system will not be precise enough (it is hard to reach 1% precision) to control the bill which is essential for a countries like Bulgaria.

As a result of the reasoning, a more precise solution based on the ATM90E32AS IC (4.6€ at small quantity) is proposed. To avoid the shortcomings of the CircuitSetup's system, a different design of the measurement board is planned. The main ATM90E32AS based board will consists of a single IC with 3x current and 3x voltage channels and SPI bus with enough GPIOs for selecting enough ICs individually. Using all 3x current and 3x voltage inputs many 3 phase consumers can be monitored easily. In addition to optional audio jacks a cheap pin header can be used to connect directly CTs. The IC voltage inputs can be selected to use external transformers (up to 3x on each board with optional power jacks) or single one connected to any of the voltage inputs. The powering of the boards will be over USB from Olimexino-Nano so the voltage sensors can be smaller like 1-2VA transformers and 1.3mm power jacks for reaching lower price and the board dimensions like 30x40mm. The optional jacks will also decrease dramatically the BOM and end price in case of using CTs like HWCT-5A/5mA (10 times cheaper than SCT-006/SCT-013). Double and quadruple board variants will be achieved by simple multiplication of a single board. All measurement board variants can be stacked with cheap 2.54mm pin header connectors. A simple interconnection board will also be developed for stacking with Olimexino-Nano. It will also carry LiPo battery when possible. On the other hand Olimexino-Nano can be stacked with Olimexino-Nano-Bat and ESP8266 WiFi module for wireless communication with Lime2-Server in case of usage of the monitoring unit as a remote node in multi-node installations.

Maximal number of single boards could be addressed via SPI is 16 so up to 48x 1 phase or up to 16x 3 phase measurement point are possible. The final BOM for comparable number of CTs will exceed the BOM (Olimex sale price of 150€ including Lime2-Server, Olimexino-Nano, CTs, VT) of the simple alternative with less than 50€ even if optional jacks (half of that price) are mounted. It can be supposed that for a typical installation with 12 measurement points with single voltage sensor and usage of HWCT-5A/5mA current sensors final price like 250€ (or 500BGN without VAT and delivery taxes) is feasible.



Single, double and quadruple variants of the measurement part of the system (ESP8266 and LiPo battery are optional)

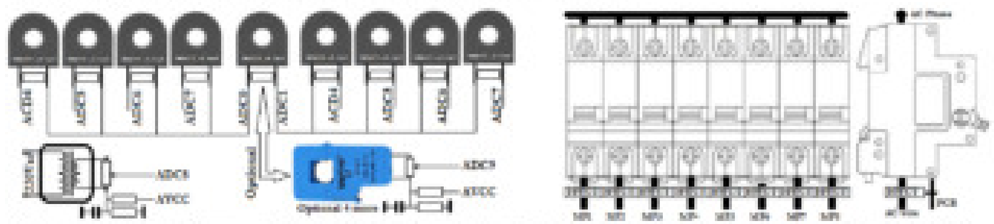
In addition to above a simple weather station and room climate monitoring units based on ESP8266 and appropriate temperature, humidity, light etc. sensors can be used. Currently only electric company's electrometer is hard to connect because in blocks of flats they are in locked cabinets in the basements. In houses they are located on the streets and optical counter sensor wirelessly connected to Lime2-Server by ESP8266 can be used but the problem is the vandal protection and distance.

And finally, the commercial usage of such a system is also relevant because of possibility to monitor all electric consumers (including 3 phase ones) for optimizing their electric bills. In such installations ESP8266 WiFi module could be changed to RS485/RS422 one (like MOD-RS485-ISO or MOD-RS485) with or without galvanic isolation usable for long distances of full or half duplex communication in noisy environments.

Notes: The prices above are approximate. The calculations do not include any production expenses. All they can be used for reference only.

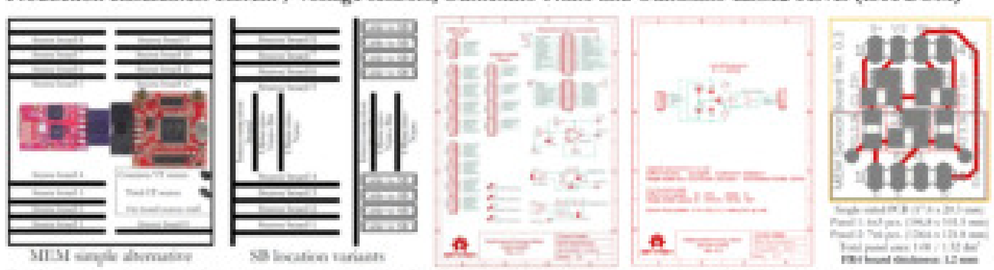
Multichannel Home Electricity Monitoring System (simple alternative)

The simple alternative as hardware will be based on low cost (a hundred euros) and low consumption (a few watts) Linux server like Olimex' OSHW LIME2-SERVER (with battery backup and SSD) and web application written on JavaScript and using Web Sockets (WS). Real time data will be processed by Arduino application working in Atmega32u4. Data will be archived on the SSD. The system could be connected via Gigabit Ethernet, Wi-Fi or 3G/4G Mobile and accessed locally or globally from Internet. The compact and contactless current sensors like HWCT-5A/5mA can be added to an existing electricity switchboard without modifications and additional safety requirements. In case of wireless communication in local network the system will be completely isolated and safe. Of course, general safety rules have to be applied when installing the sensors in the electricity switchboard. In case of wired communication the standard rules for computer equipment powering will be applied.



Real scenario: up to 9 current differential measurement points (16A each), total I_{RMS} ($64A_{AC}$) and V_{RMS} ($220V_{AC}$)

Production candidates: current / voltage sensors, Olimexino-Nano and Olinuxino-Lime2-Server (150€ BOM)



Different interconnection scenarios grant big flexibility of the simple alternative of MEM system

The idea to separate CT resistor and filter on an additional sensor board will simplify the main MEM board and will add flexibility. A single sensor board can be used for both differential and single ended connections to ADC. It can be used in both simple and advanced MEM alternatives. On the other hand the sensor board can be located at CT, main board sides even in the middle. The same idea can be applied for voltage sensors which will make possible to use different kinds of VTs.

Optionally, internal and external temperature (DS18B20 based) and other environmental sensors could be connected as well. The low consumption Lime2-Server with built-in UPS itself can be used to host also other home services like WEB, TOR, NextCloud, HomeAssistant etc.

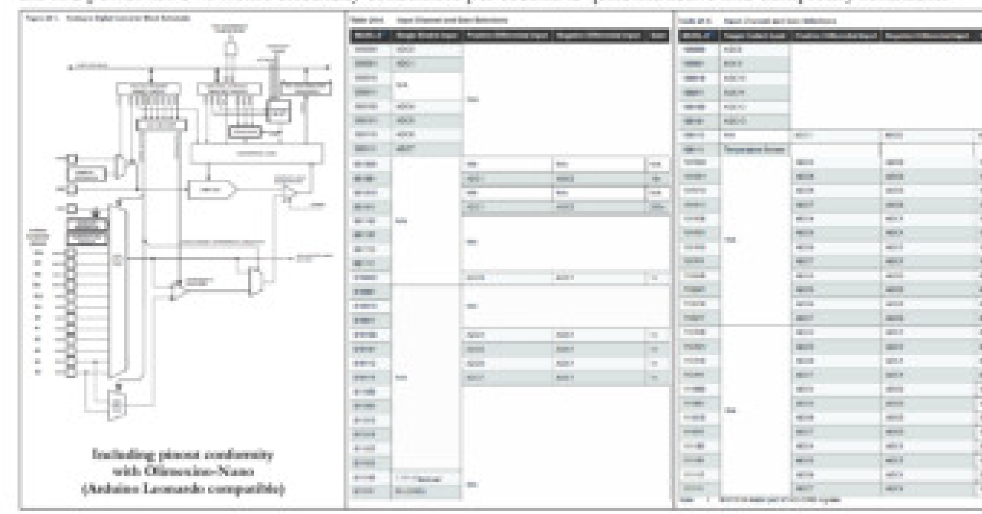
Built-in analytical and long term observation can be used for optimizing and decreasing the home electricity consumption, from the one hand. On the other hand, the metering accuracy can be certified by an authorized organization and served to control the electricity supply company.

The limitation in monitoring of two-way interconnected homes can be avoided in future by using advanced measurement algorithms and probably more resourceful real time processing unit (based on ARM Cortex-M4 MCUs like STM32F3xx/4xx). The other direction in Multichannel Home Electricity Monitoring System development could be the measurement and the observation of the self-produced electricity from photovoltaic, wind generator, geothermal etc. equipment targeting to reach sustainable green homes.

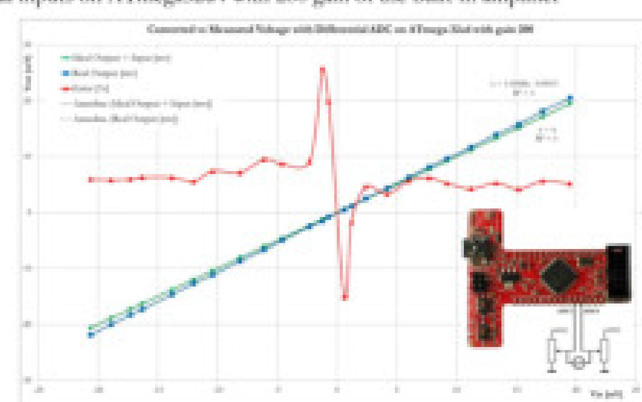
Consumption control based on time zones and priorities could be added as functionality in addition to metering. In case of unidirectional or two-way connectivity some home appliances can be limited to user defined time zones. In case of self-production of electricity all home appliances can be divided in groups with different priorities and corresponding powering rules. For implementing of such functionality current circuits should be switched on and off using relays controlled by real time processing unit. In case of self-production of electricity without two-way connectivity switching devices should also be used. In such advanced use cases a special electricity switchboard should be used as well.

HWCT-5A/5mA and SCT013-000 100A 50mA measurement scenario based on Arduino ATmega 32u4 (Olimex' OLIMEXINO-NANO board), David Pilling's Differential ADC library and Arduino DiffADCInterrupts sketch

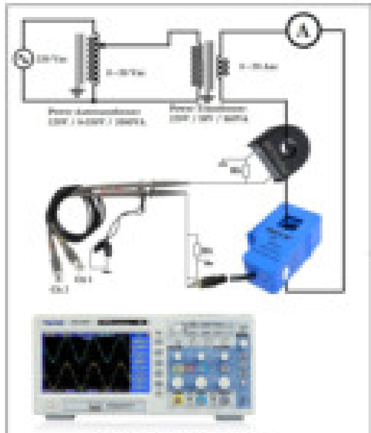
The idea is inspired by OpenEnergyMonitor project, David Pilling's Differential ADC library and appropriate current transformer (CT) sensors offered by Olimex. According to Figure 24-1 and Table 24-4 (Input Channel and Gain Selections) in the Atmel's AVR 8 Atmega32u4 datasheet it can be seen that MCU has 9 differential inputs combined with 3 gain selections for the built-in amplifier. A possible data capture scenario could be Free Running mode with auto trigger enabled and ADC interrupt at sampling end. In such a set up conversion takes 14 ADC clocks. In case of using 9.6 KHz sample rate (divide by 128 pre-scaler) it can be captured 178 samples for 20ms (1 AC cycle) which is enough for precise RMS value calculation of the selected channel. It probably will be possible to calculate sum of squared values in interrupt routine which will stop after each 178-th sample. In main loop it has to be set up the cannel selection before starting capture and will be calculated RMS for given value after that. Two more capture cycles have to be done for calculating of V_{AC} and total I_{AC} taken form electricity mains via transformer and SCT013 connected to the single ended ADC inputs (ADC8 – ADC13). As a result ten RMS values (V_{RMS} and $I1_{RMS}$ – $I9_{RMS}$) can be obtained for 200ms and up to 5 measurements per second will be possible. Of course, additional processing will down this rate but even a single measurement of the AC power for 8+1 home electricity consumers per second is quite attractive and completely reachable.



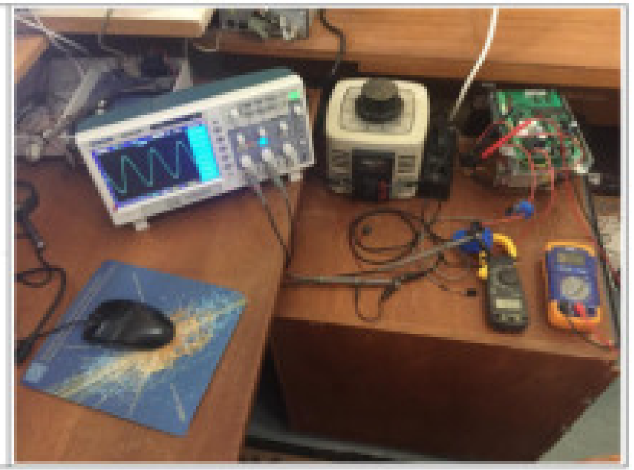
Testing differential inputs on ATmega32u4 with 200 gain of the built-in amplifier

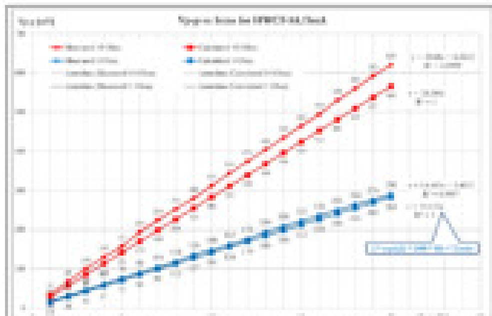


HWCT-5A/5mA and SCT013-000-100A-50mA sensors calibration charts

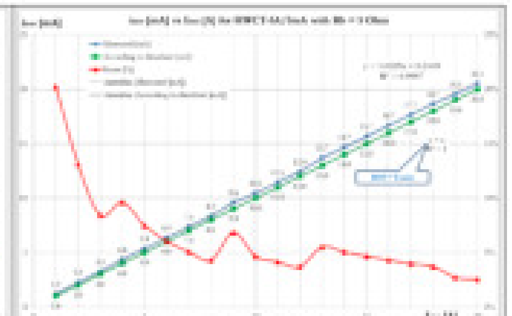


Modified test bed

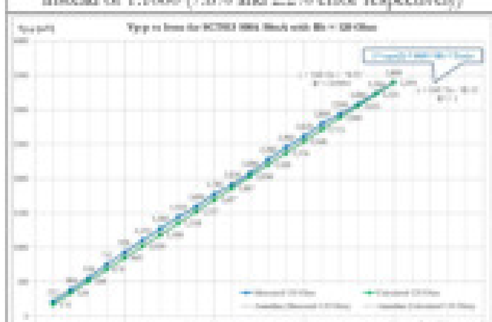




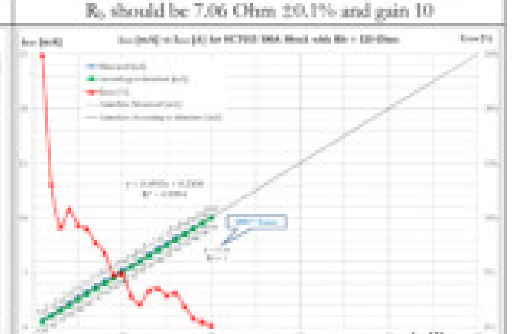
Back calculated T_{ratio} is 1:922 and 1:978 at 10 and 5 Ohm R_b instead of 1:1000 (7.8% and 2.2% error respectively)



To measure I_{RMS} up to 16A and V_{p-p} up to 3.3V R_b should be 7.06 Ohm ±0.1% and gain 10



Back calculated T_{ratio} is 1:2019 at 120 Ohm R_b instead of 1:2000 (9.4% error)

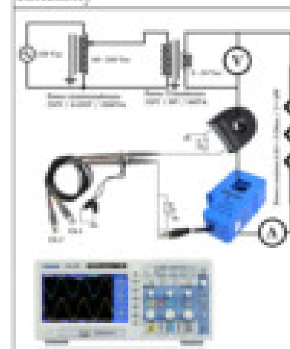


To measure I_{RMS} up to 64A and V_{p-p} up to 3.3V R_b should be 36.1 Ohm ±0.1% and gain 1

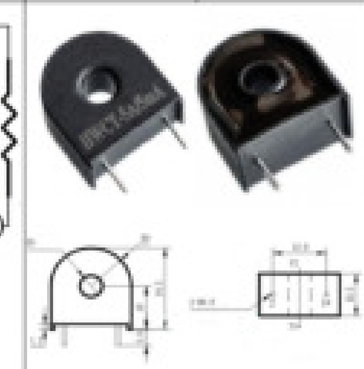
Notes: The measurement in the tests till now is not precise enough,
Power circuit current is not more than 20A for safety reasons,
More precise calibration should be done before final usage.

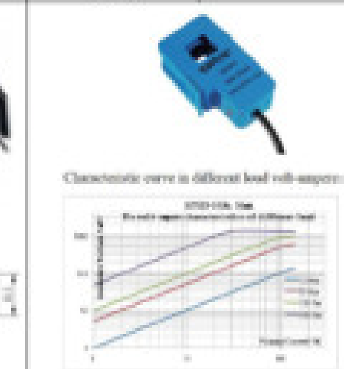
HWCT-5A/5mA and SCT013-000-100A-50mA comparison

Feature	HWCT-5A/5mA		SCT013-000 100A 50mA	
	Datasheet	Measured	Datasheet	Measured
Nominal current	5 A	1 A	100 A	1 A
Maximum current	20 A		120 A	
Turns ratio	1:1000	1:833	1:2000	1:1667
Current ratio	5A:5 mA	1A:1.2mA	100A:50mA	1A:0.6mA
DC resistance	155 Ohms	45 Ohms		103 Ohms
Load resistor	2 Ohms	120 Ohms	10 Ohms	120 Ohms
Accuracy	±2%		±1%	
Linearity			≤0.2%	



Test bed

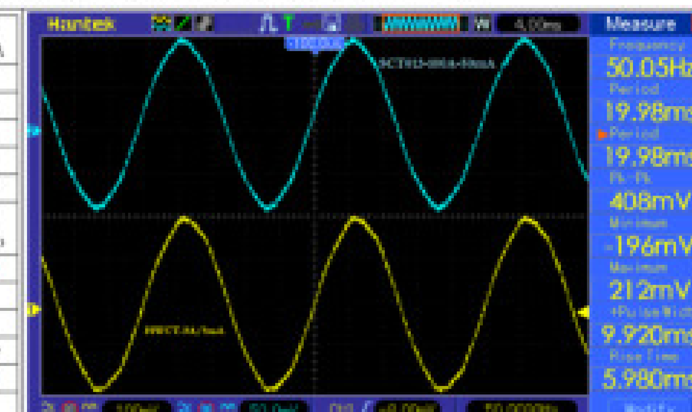




Characteristic curve in different load volt-ampere:

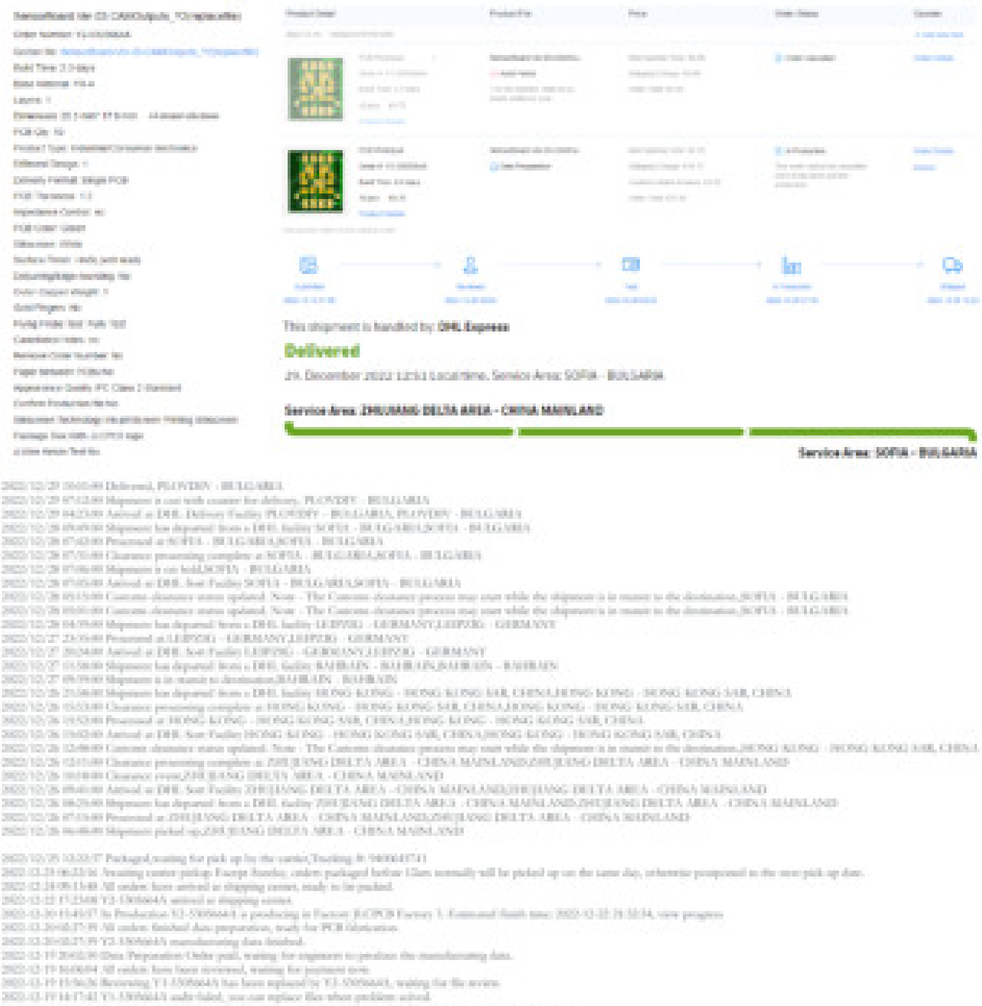
First test results (measured AC voltage at given AC current in the power chain)

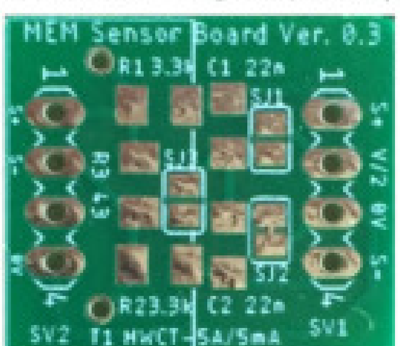
	HWCT-5A/5mA	SCT013-100A 50mA
Test conditions		
R_{power}	5 Ohms	5 Ohms
V	5 V_{RMS}	5 V_{RMS}
A	1 A_{RMS}	1 A_{RMS}
R	120 Ohms	120 Ohms
Calculations (datasheet based)		
$Vp\text{-}p = 2 * \sqrt{2} * I * R * \text{Turns ratio}$		
Vp-p	339.4 mV	169.7 mV
Test results		
Ch.	1	2
Scale	100 mV/div	50 mV/div
Vp-p	408 mV	204 mV
Diff.	20%	20%

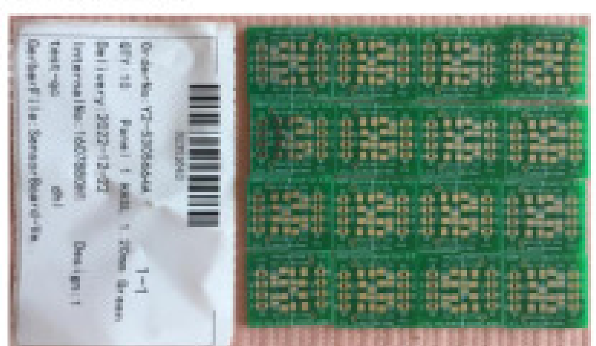


The big difference for the both sensors can eventually be explained mainly with not very precise measurement. On the other hand there is a quite big difference in declared in the data sheet DC resistance (155 Ohms) and measured (45 Ohms) for the HWCT-5A/5mA sensor.

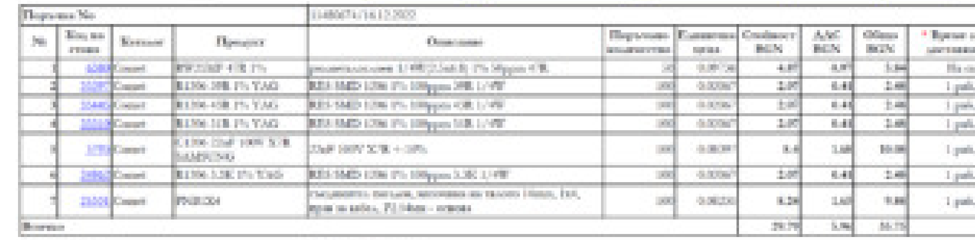
JLCPCB – Chineese PCB manufacturer (sensor board ver. 0.3 order)

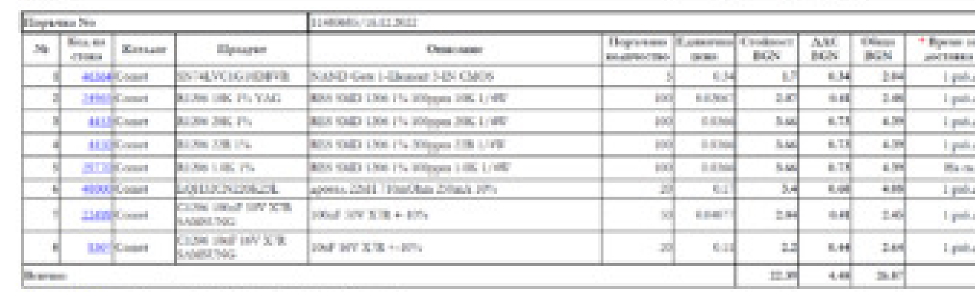






Component orders (Comet.bg, Farnell.com, Olimex.com, ANG-Bg.com etc.)





Olimex Ltd., 2 Pravda Str,Plovdiv 4000, Bulgaria

phone: +359-32-626259, fax: +359-32-621270