Brandenburg University of Technology Cottbus - Senftenberg

Prof. Dr.-Ing. Johannes Schiffer
Dr. Juan G. Rueda-Escobedo
Nicolai Lorenz-Meyer
Fachgebiet Regelungssysteme und Netzleittechnik
Module 12893 Laboratory Control and Network Control Technology, SoSe 2021

Example for a Project Plan

1 Introduction

The objective of this note is to provide an example on how to split a project into tasks and then how to elaborate a working plan. The example is based on the development of a micro Phasor Measurement Unit (μPMU) for phase and frequency measurements in electrical power systems.

The general approach for the project planning is as follows:

a) Divide the system into subsystems.

b) For each subsystem, recognize the different functions that it has to fulfill.

c) Transform the functions into tasks such as modeling, design, investigate, etc.

d) Group the tasks in categories and order them according to how they can be done.

e) Assign time to each task.

f) Distribute the tasks among the team members taking into consideration the categories and the ordering.

g) Report the task distribution and timing in a working plan.

2 Splitting the Project into Tasks

The main function of a μPMU is the monitoring of the following variables at the power outlet:

a) phase angle,

b) frequency,

c) voltage magnitude.

These variables have to be reported periodically to a server together with the time at which they were taken. Other specifications for the μPMU are:

a) it has to be small and discrete,

b) in case of a power failure, it should be able to operate for a short time and report the failure.

A general schematic of a μPMU is shown in Figure 1.

Now, to proceed, the functions and specifications for the system have to be transformed into subsystems which, together, should accomplish the system functions. Following this idea, we propose the following subsystems:

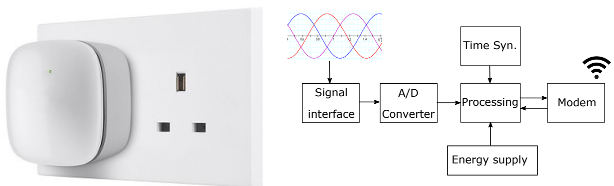

Figure 1: General schematic of a μPMU.

a) data acquisition,

b) data processing and computing,

c) power supply,

d) communication.

The subsystems and their interconnections are shown in Figure 2. For the aforementioned subsystems, we recognize the functions shown in Table 1. These functions are then converted into specific tasks, which are also presented in Table 1.

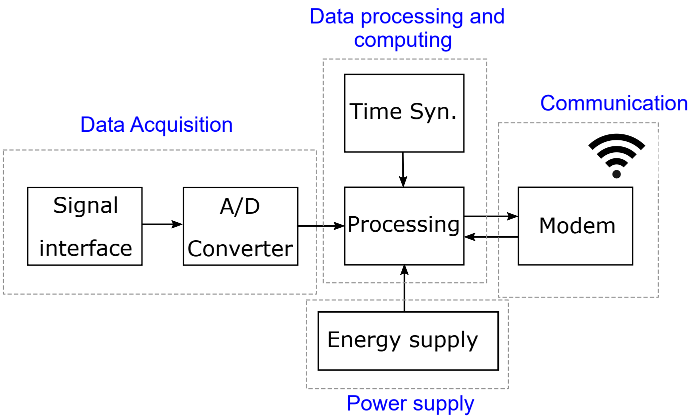

Figure 2: Subsystem division of the μPMU.

Subsystem	Functions	Tasks
Subsystem A: Data acquisition	• Adapt the voltage into a digital level. • Transform the physical variables into digital data. • Communicate the data to the Data processing subsystem.	TA1 Design an attenuation circuit to reduce the voltage from 230 V-peak to 1 V-peak. TA2 Implement an Analog-to-Digital converter with a low-pass filter. TA3 Establish a digital communication protocol between the Data acquisition subsystems and the Data processing subsystem.
Subsystem B: Data processing and computing	• Receive the signal data from the Data acquisition subsystem. • Obtain a time reference from the outside. • Process the data to obtain the phase angle, voltage magnitude and frequency. • Communicate the information to the outside.	TB1 Establish a digital communication protocol between the Data acquisition subsystems and the Data processing subsystem. TB2 Include a GPS to obtain the time and establish a communication protocol between the GPS and the Data processing subsystem. TB3 Design a discrete estimation algorithms to recover the phase angle, the voltage magnitude and the frequency from the data provided by the Data acquisition subsystem. TB4 Establish a digital communication protocol with the Communication subsystem. TB5 Select a micro processor to perform the different processes and program it.
Subsystem C: Power supply	• Provide energy for the different subsystems. • Store energy to use in case of disconnection.	TC1 Design an AC-DC converter to supply digital voltages of 3.3 V and 5 V. TC2 Design a battery charger, a DC-DC converter and a bypass to supply the energy in case of a power failure.
Subsystem D: Communication	• Transmit information from the Data processing subsystem to the outside and transmit information from the outside to the Data processing subsystem.	TD1 Specify a communication device (MODEM). TD2 Establish a digital communication protocol between the Data processing subsystem and the MODEM. TD3 Establish a digital communication protocol with an external server.

Table 1: Functions and tasks associated to the μPMU subsystems.

3 Task Organization

The next step is to organize the tasks in Table 1 into categories and to order them according to which can be done in parallel and which in series. A natural and straight-forward way to proceed is to group the tasks according to the subsystems. In Table 1, two short-hands are introduced:

a) A letter is assigned to each subsystem and to denote them, the notation Subsystem+*Letter* is introduced.

b) The different tasks are associated to a letter related to a subsystem and a number. The notation T+*Letter*+*Number* summarizes this.

The different letters will correspond to the categories. Now, it is needed to order the task according to which can be done in parallel and which in series. This is shown in Table 2. We briefly comment on the order of the tasks corresponding to Subsystem A and Subsystem B:

- Subsystem A
  - Tasks TA1 and TA2 are serialized since for designing the A/D, the behavior of the output of the attenuation circuit needs to be known. Task TA3 can be done in parallel with the A/D implementation once the A/D has been selected.
- Subsystem B
  - Task TB5 is fundamental for the subsystem since the rest of the tasks depend on the selected microprocessor. Note that this task appears twice, at the beginning and at the end. It appears in Slot 3 again because the programming task can only be done after the estimation algorithms and the communication protocols are specified.
  - Tasks TB1 and TB3 can be done in parallel once the microprocessor is selected.
  - Task TB2 can be started without knowing the microprocessor. Therefore, it can be done in parallel with TB5.
  - Finally, Task TB4 can only be started once a microprocessor has been selected.

Similar reasoning can be used to analyze the tasks corresponding to Subsystem C and Subsystem D. Note also that TA3=TB1 and that TB4=TD2. Hence, the people doing these tasks should be in coordination.

<table>
<tr><th>Subsystem</th><th>Slot 1</th><th>Slot 2</th><th>Slot 3</th></tr>
<tr><td rowspan="2">Subsystem A</td><td>TA1</td><td>TA2</td><td></td></tr>
<tr><td></td><td colspan="2">TA3</td></tr>
<tr><td rowspan="4">Subsystem B</td><td>TB5</td><td>TB1</td><td>TB5</td></tr>
<tr><td colspan="2">TB2</td><td></td></tr>
<tr><td></td><td>TB3</td><td></td></tr>
<tr><td></td><td colspan="2">TB4</td></tr>
<tr><td>Subsystem C</td><td>TC1</td><td>TC2</td><td></td></tr>
<tr><td rowspan="2">Subsystem D</td><td>TD1</td><td>TD2</td><td></td></tr>
<tr><td colspan="2">TD3</td><td></td></tr>
</table>

Table 2: Ordering for the different tasks.

Apart from ordering the tasks corresponding to each subsystem, a order in which the subsystems should be tackled is also needed. Notice that Subsystem A and Subsystem C are independent of each other, thus, they can be designed/implemented in parallel. However, Subsystem B depends on both, Subsystem A and Subsystem C, and Subsystem D depends on Subsystem B and Subsystem C. This relations establish the order in which each subsystem has to be addressed. This is reflected in Table 3.

Slot 1	Slot 2	Slot 3
Subsystem A Subsystem C	Subsystem B	Subsystem D

Table 3: Order in which each subsystem has to be addressed.

4 Working Plan

A working plan is a chart which reflects the workflow in a project. It has to clearly show the order in which the tasks have to be done, who has to do which task, and how much time is available for each activity. In the previous section, we establish an ordering between the subsystems and tasks. Now, we need to assign time to each of them and a team member to do them.

We proceed to assign the time framework for each task. The time for each task is an estimation and can be readjusted during the evolution of the project. However, note that every delay will perturb the workflow and will induce further delays in serialized tasks. For this example, the proposed time for each task is shown in Table 4.

Task	Duration	Task	Duration
TA1	7 days	TB5	20 days (2&18)
TA2	5 days	TC1	10 days
TA3=TB1	3 days	TC2	10 days
TB2	7 days	TD1	2 days
TB3	14 days	TD3	10 days
TB4=TD2	3 days		

Table 4: Proposed duration for each task.

To continue, we are going to assume that the team consists of three students, namely, Student A (green), Student B (blue) and Student C (yellow), all of them capable of accomplishing any of the tasks. Now, we need to collect the information contained in Table 2, Table 3 and Table 4 and present it together considering that at most three activities can be done simultaneously. Organizing the tasks and the time requires creativity, coordination between the team members and some iterations until a realizable working plan is obtained. An example of a working plan is shown in Figure 3.

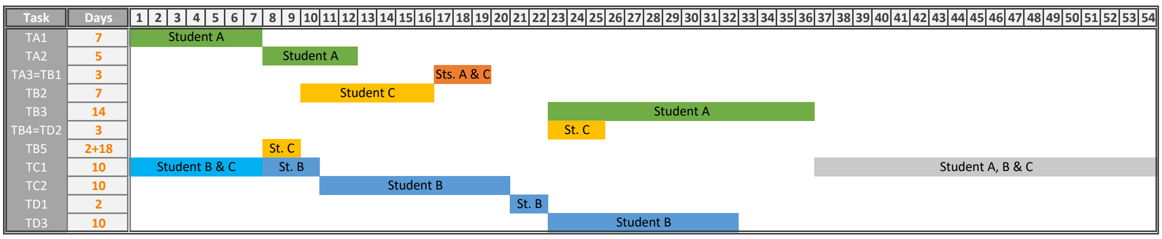

Figure 3: Example of a working plan.

The points to take into account while organizing the working plan are:

a) The number of simultaneous tasks cannot exceed the number of team members.

b) The order between tasks and the time assigned to each of them has to be respected.

c) Time-out has to be avoided as much as possible.

d) Avoid as much as possible the frequent change of activity of a team member. It is better if each team member is specialized in a subsystem or in a task category.

Note that the working plan in Figure 3 is not the unique alternative or the *best* one; it is just one possibility. The reader is encouraged to try to find a different configuration and to minimize the total number of days.

Some final recommendations:

- Apart from an order between tasks, prerequisites can be introduced. During the working plan elaboration, the tasks prerequisites should be taken into account for the order and distribution.
- Instead of days, the time assignation can be done in weeks. It is also convenient to explicitly consider the weekends.
- For a better organization, it is strongly recommended to give precise starting and ending dates to each task.
- Time for the project integration and testing should be allocated.