

Author: Víðir Bjarkason

Optimal Bicycle Gear Selection Using Multi-Sensor Data Fusion

The development of a prototype optimal gear selection system and integration with a commercial electronic transmission system for a fully automated gear shifting



University College of Southeast Norway
Campus Porsgrunn/Faculty of Technology
Department of Electrical Engineering, IT and Cybernetics
Kjølnes ring 56
<http://www.usn.no>

© 2016 Víðir Bjarkason. All rights reserved

This thesis is worth 30 ECTS

"Life is like riding a bicycle. To keep your balance, you must keep moving"
- Albert Einstein

Abstract

The latest technology in electronic transmission systems (ETS) and wireless sensors for bicycles bring a high potential to automate gear shifting on bicycles. An optimal gear selection system (GSS) for road bicycles with an ETS was developed. The system interacted with a commercial Shimano Di2 6870 ETS, resulting in a fully automatic transmission system (ATS) that selected an optimal gear for the user. The interconnection between the ETS and GSS required a customization on the ETS hardware. Mechanical button switches on ETS were replaced with a transistor based circuit to enable electronic control of gear shifts. Optimal gearing and system behavior was determined from knowledge acquisition from both practical and theoretical aspects. Knowledge-based system was developed for optimal gear selection, using fuzzy logic control. Sensors were used to measure speed, pedaling rate (cadence), pedaling power and riding position. By fusion of the sensor data and the acquired knowledge, the GSS selected an optimal gear that maintained the cadence within a predefined optimal range for a given situation. NI LabVIEW was used for development of the GSS and ANT+ toolkit for LabVIEW was used for data acquisition from ANT+ bike sensors.

The GSS prototype was installed on a demo bike equipped with an ETS and tested on a stationary indoor trainer, according to a test plan document. The system performance was tested in terms of maintaining cadence at a preset goal cadence. Two separate tests were done for goal cadences $CAD_{goal} = 80RPM$ and $CAD_{goal} = 90RPM$. The results showed that the system performed well in maintaining a consistent cadence, but the average cadence the system maintained was higher than the set goal cadence. The effect of each individual input variable on the system was tested and results confirmed that the GSS performed according to the predefined requirements.

Keywords: ANT+, Cadence, Data acquisition, Electronic transmission system, Fuzzy logic, Gear selection system, LabVIEW, Pedaling power.

Contents

List of Figures	xv
List of Tables	xvii
Abbreviations	xviii
1 Introduction	1
2 System overview	3
2.1 Requirements for prototype gear selection system	5
2.1.1 Software requirements for prototype	5
2.1.2 Hardware requirements for prototype	5
2.2 Prototype system structure	5
3 Literature survey	7
3.1 Electronic transmission systems	7
3.2 Automatic transmission systems	8
3.3 Previous studies on optimal cadence	9
3.4 ANT protocol	10
3.4.1 ANT basics	10
3.4.2 ANT+	11
3.4.3 ANT private key	11
4 Hardware	13
4.1 Commercial electronic transmission system	13
4.2 DAQ device	14
4.3 ANT+ sensors	15
4.3.1 Speed sensor	15
4.3.2 Cadence sensor	15
4.3.3 Power meter	15
4.4 ANT USB stick	16
4.5 FlexiForce sensor	16
4.6 Cycling computer	18
4.7 Smart trainer	18
5 Connecting to electronic transmission	19
5.1 Modification of button switches	19
5.2 Connection circuit design	20

6	Software for prototype development	23
6.1	LabVIEW	23
6.2	ANT+ toolkit by iNU Solutions	23
6.3	Indirectly used software	24
7	Data acquisition	27
7.1	Approach of choice: ANT+ Toolkit for LabVIEW	27
7.2	Alternative I: Using Dynastream's ANT+ source code	28
7.3	Alternative II: Modify existing ANT+ compatible software	28
8	Optimal gear selection	31
8.1	Knowledge acquisition	31
8.1.1	Optimal cadence	31
8.1.2	Mechanical restrictions	32
8.1.3	Riding position	33
8.2	Optimization approach	33
8.3	A knowledge-based system	34
8.3.1	Gear shifting sequence	34
8.3.2	Cadence and speed relationship	35
8.3.3	Classification of pedaling power	41
8.3.4	Shifting based on riding position	42
9	Programming of gear selection system	43
9.1	Program overview	43
9.2	Graphical user interface	45
9.3	Process 1: Data acquisition	46
9.3.1	FlexiForce data acquisition	46
9.3.2	ANT+ data acquisition	46
9.4	Process 2: Data processing	46
9.5	Process 3: Fuzzy logic gear selection	47
9.6	Process 4: Electronic transmission control	47
10	Prototype test plan	49
10.0.1	Test 1	50
10.0.2	Test 2	50
10.0.3	Test 3	50
11	Results and discussion	51
11.1	ANT device profile implementation test	51
11.2	Detection of riding position	52
11.3	Testing of prototype according to test document	53
11.3.1	Test 1 - Riding in the saddle	53
11.3.2	Test 2 - Riding out of the saddle	57
11.3.3	Test 3 - Coasting	58
11.3.4	Checklist for prototype test plan	60
12	Future work	61
12.1	LabVIEW software evolution	61
12.2	Hardware platforms	62
12.2.1	Gear selection system on smartphone	62
12.2.2	Gear selection system on cycling computer	64
12.2.3	Gear selection system integrated in electronic transmission	64
12.3	The future of automatic transmission systems	65

13 Conclusions	67
A Appendix	69
A.1 Master's thesis task description	71
A.2 Project Abstract	73
A.3 Gear ratios	75
A.4 Speed and cadence tables	77
A.5 Fuzzy sets	79
A.6 Fuzzy system crisp output	81
A.7 Additional panels on graphical user interface	83
A.8 LabVIEW syntax	85
A.9 ANT+ device profile data pages	91
A.9.1 SPD data page 5	91
A.9.2 CAD data page 0	91
A.9.3 PWR data page 16	92
A.9.4 PWR data page 19	93
A.10 Test plan document	95
References	103

List of Figures

2.1	Overview of how traditional MTS and how it is operated by the user. Control levers are based on cable tension.	3
2.2	ETS is operated by the user in the same manner as traditional transmission. Control levers are based on electronic switches.	3
2.3	ATS consisting of a gear selection module, sensors and an ETS. The design of the GSS module and the communication to the ETS are the objectives of this report.	4
2.4	The GSS reads speed sensor, cadence sensor, power sensor and force sensor and uses the data to determine the optimal gear for the given situation and takes actions based on the multi-sensor data. In this context, processing of the multi-sensor data is defined data fusion.	4
2.5	Overview of the ATS, using the GSS prototype for gear selection. PC runs the GSS software on LabVIEW, that acquires wireless sensor data from USB connected ANT stick. Force sensor data acquisition is handled by a DAQ device. The software determines optimal gear based on the sensor data and signals the DAQ device to trigger gearshifts on the ETS through a custom made connection circuit.	6
2.6	Complete overview of devices and sensors that the ATS consists of. The GSS prototype (PC, ANT USB Stick, DAQ and connection circuit) can be carried by the rider. The blue line shows the DAQ connection to the seat located force sensor. The red line shows the connection between the connection circuit and the ETS. Figure based on (<i>Shimano Ultegra 6870 Series Dealer's Manual</i> , 2014).	6
3.1	Breakdown of ANT/ANT+ according to the OSI model.	10
3.2	Basic structure of an ANT message. Message content includes channel number and data payload of 8 bytes. Extensions of the basic message format also exist (<i>ANT Message Protocol and Usage - Rev. 5.1</i> , 2014, p. 34).	10
3.3	An ANT+ ecosystem, showing examples of different use cases (<i>ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0.</i> , 2014, p. 8).The ULP quality of the ANT protocol makes it well suited for personal area networks (PAN).	11
4.1	Normal installation of the Shimano Di2 ETS. The Wireless unit SM-EWW01 (not shown) was connected between Junction (B) and the Rear Derailleur. The SW-R671 shifting switches (not shown) are an optional equipment and replaced the Dual Control Levers (<i>Shimano Ultegra 6870 Series Dealer's Manual</i> , 2014).	13
4.2	Shimano Di2 6870 front derailleur. The unit is motor driven and has 2 gear settings (<i>Ultegra Di2</i> , 2015).	14
4.3	Shimano Di2 6870 rear derailleur. The unit is motor driven and has 11 gear settings (<i>Ultegra Di2</i> , 2015).	14

4.4	The NI USB-6008 has $4.7k\Omega$ onboard pull-up resistor for the digital channels (<i>NI USB-6008/6009 User Guide</i> , 2015, p. 20). The external pull-up resistor configuration was not used.	14
4.5	How private ANT is used for dual sided sensors as the Garmin Vector 2 (<i>ANT+ Device Profile - Bicycle Power. Revision 4.2.</i> , 2015, p. 30).	15
4.6	Garmin ANT+ speed sensor (<i>Bike Speed Sensor</i> , 2016).	16
4.7	Garmin ANT+ cadence sensor (<i>Bike Cadence Sensor</i> , 2016).	16
4.8	Garmin Vector 2 pedal based power meter and ANT+ pedal pods (<i>Vector 2</i> , 2016).	16
4.9	Garmin ANT USB stick for sending/receiving ANT+ signals (<i>USB ANT Stick</i> , 2016)	16
4.10	FlexiForce A201 100lb model force sensor is flexible and robust and suits well for surface mounting on bicycle saddle (<i>FlexiForce Standard Model A201 - Datasheet Rev. A</i> , n.d.).	16
4.11	FlexiForce force sensor mounted on bicycle saddle using clear tape. The sensor presence is not noticeable while seated on the saddle. Here the pressure sensitive part of the sensor is located at the center of the saddle.	17
4.12	Connecting FlexiForce sensor to the NI USB-6008 unit. Drive voltage supplied by NI USB-6008 was set to 5V. Figure created using Fritzing (<i>Fritzing</i> , 2016)	17
4.13	Screenshoot from the Garmin Edge 520 showing ANT+ sensor parameter data fields and current gear on the Di2 transmission. The Garmin unit has access to the Di2 ANT private key to receive the current gear status.	18
5.1	Overview of how the GSS prototype interacts with the ETS.	19
5.2	Shimano SW-R671 left and right switches (<i>R671 Remote Triathlon Shifter</i> , 2015).	19
5.3	Disassembled button switch (SW-R671) (Sarti, 2013). Each colored ring shows the contacts for each button switch.	20
5.4	Disassembled button switch (SW-R671). Voltage across green and red contacts is +3.3V. Green wires were connected to common ground on DAQ device.	20
5.5	Overview of transistor based connection circuit that is the link between DAQ device and SW-R671 switches. One transistor is required for each button. Another transistor was connected in the same manner to the other two unused connections of the SW-R671 circuit.	21
5.6	Voltage divider showing DAQ digital output with internal resistor $R_{internal}$ which is $4.7k\Omega$. The base resistor $R_B = 10k\Omega$. The base voltage V_B is 0.7V. Output voltage V_O is the voltage measured at the DO on the DAQ device.	21
5.7	Simplified overview of the DAQ and connections to the transistors. Each transistor acts as either an open or a closed switch, depending on whether it is receiving a high or a low signal from the DAQ. As the internal resistance of the original button switch was unknown, resistors R_C were chosen to be 220Ω . This was for protecting the SW-R671 circuits, in case if short circuiting of the contacts could cause damage on the SW-R671 circuitry over time.	22
5.8	The circuit that controls shifting using NI USB-6008 (<i>National Instruments- Image Gallery</i> , 2009) digital outputs and four 2N3704 transistors. The 2N3704 pin configuration from left to right is Emitter-Collector-Base (ECB). $R_B = 10k\Omega$ and $R_E = 220\Omega$. Diagram created using Fritzing (<i>Fritzing</i> , 2016)	22

6.1	Diagram showing overview of interconnection and the purpose of different software used for developing the optimal gear selection system. LabVIEW with ANT+ toolkit is required to run the optimal gear selection system. SimulANT+ was used for testing reception of ANT+ signals when developing the Optimal Gear Selection Program and Zwift was used to control the wheel resistance when testing the Optimal Gear Selection System indoors on a stationary smart trainer. Both computers run on Windows 10 operating system.	25
8.1	Cross chaining illustrated. The chain runs across the drivetrain centerline (displayed in red). Operating the drivetrain in this condition causes additional drivetrain stress, accelerated component wear and decreased efficiency (<i>Why Avoid Cross Chaining Gears On Your Bike</i> , 2012).	32
8.2	The gear shifting sequence consists of 14 of 22 possible sprocket combinations of the 2 chainrings at the front and the 11 sockets on the rear cassette. Shifting between 7th and 8th gear requires shifting the front derailleur.	34
8.3	Fuzzy set for gear selection based on speed for $CAD_{goal} = 80RPM$. Fuzzy set with equivalent information for $CAD_{goal} = 90RPM$ can be found in Figure A.1 in Appendix A.5.	38
8.4	Fuzzy output membership functions. Singletons were used to represent the 14 gears available. Fuzzy set output membership functions for $CAD_{goal} = 90RPM$ are identical.	38
8.5	Testing of fuzzy set for $CAD_{goal} = 80RPM$ in Fuzzy System Designer in LabVIEW. For $SPD = 19km/h$, three rules are invoked simultaneously. Gear is chosen according to the rule with the highest weight each time, in this case the 4th gear. At $SPD = 19km/h$, 3rd,4th and 5th gear have the weights, 0.19, 0.96 and 0.31 respectively.	39
8.6	Crisp gear outputs for each speed for $CAD_{goal} = 80RPM$. Average cadence for 2nd to 13th gear was approximately $80RPM$. A fuzzy set with equivalent information for $CAD_{goal} = 90RPM$ can be found in Figure A.4 in Appendix A.6.	40
8.7	Speeds on boundary between gears are shown in red. When GSS is operated at steady pace at some of the boundary regions, the system will response to the fluctuations by shifting gears, unless a time delay for gearshifts is implemented. The time delay allows temporary float to a lower gear without a performing the corresponding gearshift.	40
8.8	Crisp outputs from power zone classification. The three power zones do not overlap each other.	41
9.1	The GSS consists of the the optimal gear selection process which is divided into subprocesses. The data fusion process consists of the data processing and Fuzzy Logic Gear Selection processes.	43
9.2	Placement of the four subprocesses within the GSS program loop structure. The arrows indicate data flow between the loops.	43
9.3	Optimal gear selection program flowchart. The Optimal gear selection process is divided into 4 subprocesses: Data acquisition, data processing, fuzzy logic gear selection and electronic transmission control.	44
9.4	The main panel on the GUI. The dashboard displays sensor readings and transmission control status. User can choose to manually control the control inputs to test the system.	45

9.5	Detailed tasks within the electronic transmission control subprocess. Information flow between different loops is indicated with arrows. The electronic transmission control is handled by loops A and B. Loop B is only executed when an optimal gear value (red line) from loop C is changed. Variables are shared between loops using local variables in LabVIEW.	47
9.6	More detailed explanation of tasks inside loop B. Required actions for each derailleur are determined according to the gear shifting sequence. Then pulses are generated for each derailleur to perform the corresponding action.	48
10.1	Overview of how all components of the GSS prototype are internally connected, and how they connect to the ETS and sensors.	49
11.1	Filtered speed and cadence measurements, showing the linear relationship. Pedaling was stopped at $t \approx 12s$. Filtering was done using moving average filter in LabVIEW.	51
11.2	Voltage reading over around half minute interval where rider started standing and changed the riding position four times. While standing, no pressure was applied to the sensor and the voltage reading measured was $0.09V$. Samples were collected at $4Hz$ and drive voltage was $5V$	52
11.3	Plot of same voltage measurement as in Figure 11.2, but here with moving average filter for smoothing of the signal both for reduce the effect of oscillations in measurements due to movements of the rider and noise. Samples were collected at $4Hz$ and sensor drive voltage was $5V$	52
11.4	Cadence as speed is increased. The average cadence on the interval $10s - 50s$ was $95RPM$. The system is set to have no additional time delay for upshifts, but time delay for downshifts was set to $1s$. This additional time delay for downshifting causes the cadence to drop when speed is decreased.	53
11.5	Current gear (G_C) of the ETS, and the next appropriate gear to shift in (G_N) is calculated by the GSS. When $G_N > G_C$, the system shifts gear without a delay. The graph shows when accelerating, G_C follows G_N with $0.2s$ delay, but when downshifting the system the total delay was $1.2s$. That can be seen on the graph when decelerating and G_C lags G_N	54
11.6	Graph showing G_C and corresponding combinations of S_F and S_R . Shifting between 7th and 8th gear requires 1 shift on the front derailleur and two shifts on the rear derailleur. This causes the system to temporarily go in a sprocket combination equivalent to 9th gear in the shifting sequence. This appears as a spike on the graph at $t \approx 30s$. The same occurs when shifting from 8th down to 7th gear, then the system is temporarily into a sprocket combination equivalent to 6th gear in the shifting sequence. This appears as a negative spike on the graph at $t \approx 60s$ where G_C drops to 6 before reaching 7.	54
11.7	Cadence as speed is increased at power zone 3. The average cadence on the interval $10s - 40s$ was $99RPM$. Time delays for downshift are noticeable as cadence drops when decelerating.	55
11.8	Current gear G_C of the ETS and the next appropriate gear to shift in G_N is calculated by the GSS.	56
11.9	Graph showing current gear G_C and what combination of front sprocket S_F and rear sprocket S_R for all 14 gears. As discussed earlier, shifting between 7th and 8th gear causes the spike at $t \approx 25s$ when upshifting and a negative spike at $t \approx 55s$ when downshifting.	56

11.10	The affect of riding position on gear shifts. Riding in standing position did only affect shifting between 7th and 8th gear in the shifting sequence. The disabled front derailleur shifting can be seen at $t \approx 21 s$ as the GSS did not shift up to 8th gear until rider sat on the saddle. At $t \approx 71 s$ the rider sits down which enables the system to shift down from 8th gear.	57
11.11	The current gear G_C and combinations of front sprocket S_F and rear sprocket S_R for all 14 gears. Riding in a standing position only affected gear shifts when shifting between 8th and 9th gear.	58
11.12	Graph showing how speed increased when pedaling and speed decreased after pedaling was stopped. The wheel kept spinning freely after pedaling was stopped because the resistance unit of the stationary trainer was disconnected at $t \approx 41 s$. The delays in cadence compared to speed measurements are visible on the graph at $t \approx 5 s$, $t \approx 45 s$ and $t \approx 70 s$	58
11.13	The system does not perform gearshifts when $CAD = 0$ and no value for next gear G_N is calculated and G_C remains unchanged until measured $CAD \neq 0$. . .	59
11.14	Indicating zero cadence from power measurement was not used as zero tracking of power was unreliable, as seen at $t \approx 45 s$. The power measurement was only used for determining which power zone the GSS operated in, to set CAD_{goal} accordingly.	59
12.1	Overview of how an embedded device could replace the hardware that the current prototype system uses for sensor data acquisition, data processing and connection to the ETS. A compact embedded device could easily be mounted on a bicycle.	62
12.2	Overview of how the system could be made into a more consumer friendly solution that is more software orientated. The only additional hardware module required to turn an ETS into an ATS is a custom made wireless interface module, shown in orange. Only the smartphone would be carried by the user.	63
12.3	Overview of a smartphone controlled ATS, a solution that is completely software based. The GSS software runs on the smartphone. Here it is assumed that the ETS enables wireless connection the smartphone to perform gearshifts based on commands from the GSS.	63
12.4	Overview of a cycling computer controlled ATS, where the GSS is a completely software based solution. It is assumed that the cycling computer will have wireless access to the ETS to perform gearshifts.	64
12.5	Overview of the ultimate integration solution of GSS that runs within ETS internal computer, making the GSS completely invisible to the user. The cycling computer is not part of the ATS, but would by used for monitoring and make configurations to the ATS.	65
A.1	Fuzzy set for gear selection based on speed for $CAD_{goal} = 90 RPM$	79
A.2	Fuzzy rules for the classification of power into 3 power zones.	80
A.3	Fuzzy rules for the classification of power into 3 power zones. Singletons represent the 3 power zones.	80
A.4	Crisp gear outputs for each speed for $CAD_{goal} = 90 RPM$. Average cadence for 2nd to 13th gear was approximately $90 RPM$	81
A.5	The transmission control panel on the GUI. Gear shift speed and time delays can be adjusted. Manual control of ETS for initial calibration to 1st gear. In the left corner there are indicators for indicating when shift signals are sent to the DAQ. The system tracks current gear status by counting pulses sent to the DAQ device.	83
A.6	The sequence panel on the GUI. Input of gear shifting sequence. Default values are set according to sequences in knowledge base.	83

A.7	The sensor values panel on the GUI displaying the raw sensor values from speed, cadence, power and FlexiForce sensors.	84
A.8	The ANT+ panel by Darren Mather, iNU Solutions (<i>ANT+ Device Drivers</i> , 2014). Allows connection speed, cadence and power sensor via the ANT USB-m stick.	84
A.9	Driving the FlexiForce sensor using DAQ Assistant and 5V drive voltage. Filter was set to signal smoothing and half-width of moving average was set to 5. Sensor voltage readings above 0.098V indicate the rider is sitting on the saddle. For the "False" case of the inner case structure, the sensor drive voltage is set to zero. Location: Loop C.	85
A.10	Calculate SPD according to the ANT+ SPD Profile (<i>ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0.</i> , 2014). Note that the "m1" indicates value from previous loop run. Location: Loop C. .	85
A.11	Implementation of speed calculation. The "CALC SPD" sub-VI gets event time and revolution count values from last loop run and the previous loop run values from feedback nodes. A new speed value is only calculated when the speed revolution counter has increased from previous loop run (True case). When the counter has not increased between runs (False Case), speed value is not updated. When the stop indicator boolean is on, then SPD is set to zero. Location: Loop C	86
A.12	Calculate CAD according to the ANT+ SPD Profile (<i>ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0.</i> , 2014). Note that the "m1" indicates value from previous loop run. Location: Loop C. .	86
A.13	Implementation of cadence calculation. The "CALC CAD" sub-VI gets event time and revolution count values from last loop run and the previous loop run values from feedback nodes. A new speed value is only calculated when the speed revolution counter has increased from previous loop run (True case). When the counter has not increased between runs (False Case), speed value is not updated. As there is no stop indicator boolean, the 20 latest revolution count values are collected in an array. The sub-VI "CAD APPROX ZERO?" compares the indices of the matrix and if they are all the same, the CAD is set to zero. That corresponds to setting cadence to zero for $CAD < 15 RPM$. Location: Loop C. .	87
A.14	Fuzzy logic gear selection process in loop C. The "Fuzzy PWR Zone" sub-VI classifies power into Zone 1,2 or 3. The "True" case for zero cadence is empty. The GSS keeps track of the current gear (G_C) and concurrently calculates an optimal gear according to sensor measurements. The new value for an optimal gear is shown as Next Gear (G_N). Gearshifts are made when $G_N \neq G_C$. When $G_N = G_C$, the system is in the optimal gear and no gearshift is performed. . . .	87
A.15	An isolated syntax section from loop B showing how case structure is used to choose a gear shifting sequence based on riding position and front gear status. The syntax shows the sequence to choose when the rider start in a standing position. The value "NextGear" comes from loop C and corresponds to the optimal gear.	88
A.16	Transmission control (loop A). The program keeps track of the current gear by counting shifting pulses received from loop B. The sub-VI "DO" receives the pulses and for each of the four shifting actions (front upshift, front downshift, rear upshift, rear downshift) it allocates the signals to the corresponding digital output channel on the DAQ device. The sub-VI "SEQ?" receives the front sprocket (S_F) and rear sprocket (S_R) combinations and calculates the current gear (G_C) according to the gear shifting sequence.	89

A.17 Electronic transmission control (loop B). This syntax is placed within an event structure, which is triggered when "NextGear" value from loop C changes. In the first frame, current gear of the system is compared to the updated value of an optimal gear, which comes from loop C. Number of required upshifts/downshifts for each derailleur is calculated. In the second frame, pulses are generated in loops that are set to run as many times as the number of required gearshifts. Location: Loop B. 90

A.18 The general SPD data page sent by ANT+ SPD sensor. The main data page is sent at ~4Hz. Screenshot from Dynastream’s ANT+ Bike Speed device profile document (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014, p. 22*) 91

A.19 Data page 0 sent by ANT+ CAD sensor. The main data page is sent at ~4Hz. Screenshot from Dynastream’s ANT+ Bike Cadence device profile document (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014, p. 31*). 91

A.20 Data page 16 sent by ANT+ PWR sensor. The datapage contains computed instantaneous power. Screenshot from Dynastream’s ANT+ Bicycle Power device profile document (*ANT+ Device Profile - Bicycle Power. Revision 4.2., 2015, p. 31*). 92

A.21 Data page 19 is an optional page sent by some ANT+ power sensors, including the Garmin Vector 2. The data can be useful for implementing a soft sensor to determine cyclist’s riding position. Screenshot from Dynastream’s ANT+ Bicycle Power device profile document (*ANT+ Device Profile - Bicycle Power. Revision 4.2., 2015, p. 42*). 93

List of Tables

- 8.1 The gear shift sequence consists of 14 different sprocket combinations. The sequence was designed to prevent cross chaining and minimize number of front derailleur shifts through the sequence. The ID for each gear indicates the sprocket combination. 35
- 8.2 Speed (*km/h*) at different cadence values for each of the 14 gears in the sequence. Colors are used to group speed into sections with increments of 10*km/h* for better readability. 36
- 8.3 Gear selection for $CAD_{goal} = 80RPM$. Gear is changed when cadence reaches either low limit of 70*RPM* or high limit 90*RPM*. The cadence at same speed after a gearshift confirms that each gearshift maintains the cadence at approximately 80*RPM*. Table A.2 in Appendix A.4 shows equivalent data for $CAD_{goal} = 90RPM$ 36
- 8.4 Speed at low and high cadence limits. For $CAD_{goal} = 80RPM$ a cadence float of $\pm 10RPM$ is allowed. Lower cadence limit for each gear was defined at 70*RPM* and the upper limit at 90*RPM*. Note the extended speed ranges for 1st gear and 14th gear. Fuzzy sets for gear selection were created according to this data. Table A.3 in Appendix A.4 shows equivalent data for $CAD_{goal} = 90RPM$ 37
- 8.5 A gear shifting sequence used when starting riding out of the saddle while in front sprocket number 1. Gears 7-14 in the sequence consist of the same sprocket combinations. 42
- 8.6 A gear shifting sequence used when starting riding out of the saddle while in front sprocket number 2. Gears 1-8 in the sequence consist of the same sprocket combinations. 42

- 11.1 A filled checklist for testing of optimal gear selection system prototype. 60

- A.1 Gear ratios for 53/39T front sprockets and 11-28T rear cassette. Sprocket combinations are sorted by gear ratio in an ascending order. Gear sprocket combinations eliminated from the gear shifting sequence to avoid cross-chaining are marked in red. Combinations eliminated to avoid unnecessary front derailleur shifts are marked in blue color. The final sequence consists of 14 gear combinations. 75
- A.2 For $CAD_{goal} = 90RPM$. Gear selection based on cadence. Gear is changed when cadence reach either low limit of 80*RPM* or high limit of 100*RPM*. Fuzzy sets for gear selection were created according to this data. 77
- A.3 For $CAD_{goal} = 90RPM$ a cadence float of $\pm 10RPM$ is allowed. Lower cadence limit for each gear was defined at 80*RPM* and the upper limit at 100*RPM*. . . . 77

Preface

This thesis was written in spring 2016, on fourth semester of the Systems and Control Engineering masters program in University of Southeast Norway on Porsgrunn campus. The FMH606 master's thesis is a mandatory 30 credit course. Supervisor of the project was Professor Saba Mylvaganam and co-supervisor was Associate Professor Håkon Viumdal. The author has background in mechatronics engineering B.Sc. from Reykjavik University. The signed task description for the project can be found in Appendix A.1 and a signed project abstract document can be found in Appendix A.2 .

Thanks to Professor Kanagasabapath Mylvaganam for the cooperation throughout the project. Special thanks to Darren Mather owner at iNU Solutions for supporting the project and making this possible by providing software license and assistance with programming. Thanks to UCSN for providing the necessary equipment for the project. Thanks to my friend Unnar Drafnarson for lab work assistance, fun times on the bike and inspiring discussions about the project over dozens of cups of coffee. I am grateful to my family for the support they have provided me throughout my studies.

Porsgrunn, 3. June 2016

Víðir Bjarkason

Abbreviations

Abbreviation	Definition
AI	Artificial Intelligence
ATS	Automatic Transmission System
BJT	Bipolar Junction Transistor
CAD	Cadence
CBJ	Collector-Base Junction
DAQ	Data Acquisition
DO	Digital Out
DSTP	DataSocket Transfer Protocol
EBJ	Emitter-Base Junction
ETS	Electronic Transmission System
FPGA	Field-Programmable Gate Array
GSS	Gear Selection System
GUI	Graphical User Interface
I/O	Input Output
MTS	Mechanical Transmission System
NI	National Instruments
NPN	Negative Positive Negative
OSI model	Open Systems Interconnection model
PAN	Personal Area Network
PC	Personal Computer
PWR	Power
RPM	Rounds Per Minute
RT	Real Time
SISO	Singular Input, Singular Output
SPD	Speed
ULP	Ultra Low Power
USB	Universal Serial Bus
VI	Virtual Instrument

Chapter 1

Introduction

Optimal gear selection for competitive cyclists can increase performance by improving pedaling efficiency. Changing gears at a given speed influences the cyclist's pedaling rate (cadence). An optimal cadence in cycling sports is a topic that has been researched and results have shown that pedaling at an optimal cadence is an important factor for achieving best pedaling efficiency (Scarf, Jobson, Passfield & Reed, 2016). Riders can select an optimal gear by themselves, by monitoring the cadence on a cycling computer and select a gear to maintain the cadence within an optimal range. Regulating cadence manually in this manner costs concentration from cyclists and they will usually spend more time in sub-optimal gears than if the gear shifting process would be automatic.

Modern technology in electronic transmission systems (ETS) for bicycles has brought the potential of automating gear shifts. ETS have become increasingly popular since the introduction of the commercial Shimano Di2 system in 2009 (*Company History*, 2016). ETS are a superior platform for an automatic transmission system (ATS), as an electronic signal can be used to actuate gear shifts. The current ETS for road bicycles on the market do not have an automatic feature, but require the user to push buttons to shift gears. Instead of the user input for shifting gears, a gear selection system (GSS) can be designed to tell the ETS when to perform the gearshifts. The Shimano Di2 system can be modified to be controlled by a GSS with minor changes on the hardware. When considering ATS versus manually operated systems for bicycles, other qualities than convenience may not be obvious for the recreational cyclist. ATS can be beneficial in competitive cycling disciplines like prolonged road cycling, where performance and pedaling efficiency are important.

A GSS can be used to select an optimal gear and concurrently send commands to the ETS to perform the corresponding gearshifts. The product of integrating a GSS with an ETS, is an ATS. To determine an optimal gear for a given situation, the GSS can acquire data from wireless sensors mounted on the bicycle. Useful measurands to use for optimal gear selection are speed, pedaling power, pedaling rate (cadence) and riding position. Optimal gear selection systems that select an optimal gear based on multi-sensor data have already been implemented. Currently there is a product on the market that does this. The product is still in a prototype phase and consists of a head unit that is located on a bicycle's handlebar and acquires wireless sensor data from bike sensors and communicates to a Shimano Di2 ETS to perform gearshifts (Maker, 2014).

This report describes how the latest technology in wireless sport sensors and ETS for bicycles was used to build an Automated Transmission System (ATS). The development and design of the GSS was split into three main tasks: communications with ETS, data acquisition, and optimal gear selection. The first task was to make the appropriate hardware modifications to the ETS to enable the interconnection to the GSS. The next task was data acquisition from the wireless

ANT+ bike sensors that use the ANT protocol. Source code to build ANT data acquisition applications in C++ is available online, but as the GSS software was programmed in LabVIEW, a new approach was taken (Dynastream Innovations Inc., 2016). ANT+ toolkit by iNU Solutions was used for ANT data acquisition in LabVIEW (Mather, 2015). This delivered the advantage that the entire GSS software could be programmed in LabVIEW, which makes this solution novel. The main task was to process and implement the acquired data for choosing an optimal gear. The GSS was designed as a knowledge based system and a systematic approach was taken in the knowledge acquisition. In the context of this report, the term data fusion refers to the process of fusing the multi-sensor data and the knowledge in the knowledge base to select an optimal gear.

All the necessary technology for developing a state of the art ATS is currently available. The GSS prototype developed in this project uses the latest sensor technology and the most recent innovations in software for ANT sensor data acquisition. The GSS software is entirely based on LabVIEW, which makes an exceptional platform for future developments of the system.

Chapter 2

System overview

Bicycle transmission system can be categorized into Mechanical Transmission Systems (MTS) and Electronic Transmission Systems (ETS). MTS are the traditional systems that rely on cable tension for making gear changes. MTS usually consists of two derailleurs manually operated by the user as shown in Figure 2.1. Normally there are two separated controllers, one for each derailleur. Change in cable tension is used for adjusting the derailleur positions that guides the chain between different sized sprockets (Brown, 2007). When referring to bicycle derailleur gears, the more general term "transmission" will be used for convenience. Clear distinction will be made between MTS and ETS.



Figure 2.1: Overview of how traditional MTS and how it is operated by the user. Control levers are based on cable tension.

ETS operate in similar manner as mechanical system except instead of using cable tension for controlling derailleur movements, electronic motors are used. The user interface is similar, two buttons for each derailleur for downshifts and upshifts. There is noticeable difference MTS and ETS as the electronic levers only require a click of a button while the mechanical levers require more force to be applied to pull or slack the cable connected to the derailleur to be controlled. Figure 2.2 shows how ETS is operated by the user.



Figure 2.2: ETS is operated by the user in the same manner as traditional transmission. Control levers are based on electronic switches.

Creating an ATS has become easier with the introduction of ETS. Instead of user input, an ATS has to decide when to shift gears and execute the gearshifts based on sensor inputs. By integrating a gear selection module to an ETS, the outcome will be an ATS, as shown in Figure 2.3.



Figure 2.3: ATS consisting of a gear selection module, sensors and an ETS. The design of the GSS module and the communication to the ETS are the objectives of this report.

When designing an ATS, the GSS is the most important aspect as it replaces the user input. The rider usually has a good sense for what gear is best suited for the given situation based on environmental factors, body conditions and personal preferences etc. The GSS and the control inputs are shown in Figure 2.4.

The objective of building an ATS is twofold:

1. To design and make a GSS prototype that determines the optimal gear for given situation, based on data from different bike sensors
2. To integrate the GSS prototype with a commercial ETS to achieve an ATS

Optimal Gear Selection Using Multi-Sensor Data Fusion



Figure 2.4: The GSS reads speed sensor, cadence sensor, power sensor and force sensor and uses the data to determine the optimal gear for the given situation and takes actions based on the multi-sensor data. In this context, processing of the multi-sensor data is defined data fusion.

2.1 Requirements for prototype gear selection system

The goal was making a GSS prototype that selects optimal gear and interacts with a commercial ETS to perform gearshifts automatically. The outcome of integrating the GSS and ETS was an ATS that selected optimal gear, using multi-sensor data fusion. The connection between the GSS and ETS required modifications on some of the ETS hardware.

The prototype GSS must fulfill the following requirements:

2.1.1 Software requirements for prototype

- Acquires data from ANT+ sensors
- Processes raw sensor data
- System is capable of observing the cyclist's riding position
- Uses speed, cadence, power and riding position as inputs
- Takes into account practical restrictions what gear sprocket combinations are selected
- Determines optimal gear for the given situation using AI assisted techniques
- Interacts with ETS to automatically shift into the optimal gear

2.1.2 Hardware requirements for prototype

- Runs on a battery powered device that is mobile enough to be carried by the rider.
- Interacts with Shimano Di2 6870 ETS via NI USB-6008 DAQ device
- Uses ANT+ sensors for measuring speed, power and cadence

2.2 Prototype system structure

The system overview in Figure 2.5 shows the main hardware components of the GSS prototype. The ATS consists of the GSS prototype, ETS and sensors.

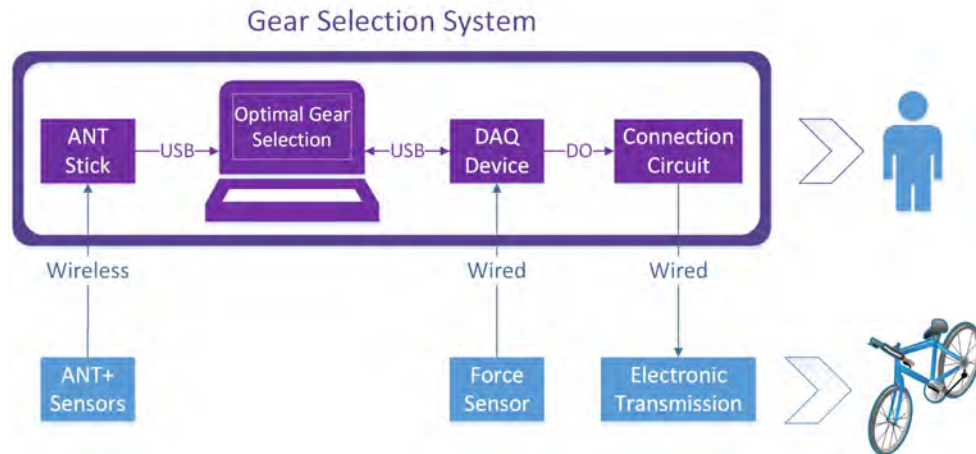


Figure 2.5: Overview of the ATS, using the GSS prototype for gear selection. PC runs the GSS software on LabVIEW, that acquires wireless sensor data from USB connected ANT stick. Force sensor data acquisition is handled by a DAQ device. The software determines optimal gear based on the sensor data and signals the DAQ device to trigger gearshifts on the ETS through a custom made connection circuit.

The sensors and ETS were installed on a demo bike for testing. A detailed overview of installation of the prototype system components on the demo bicycle is shown in Figure 2.6. The GSS consists of a PC, DAQ, connection circuit and ANT USB-m stick, and can be carried by the rider in a backpack. That makes the GSS prototype portable enough to be field tested.

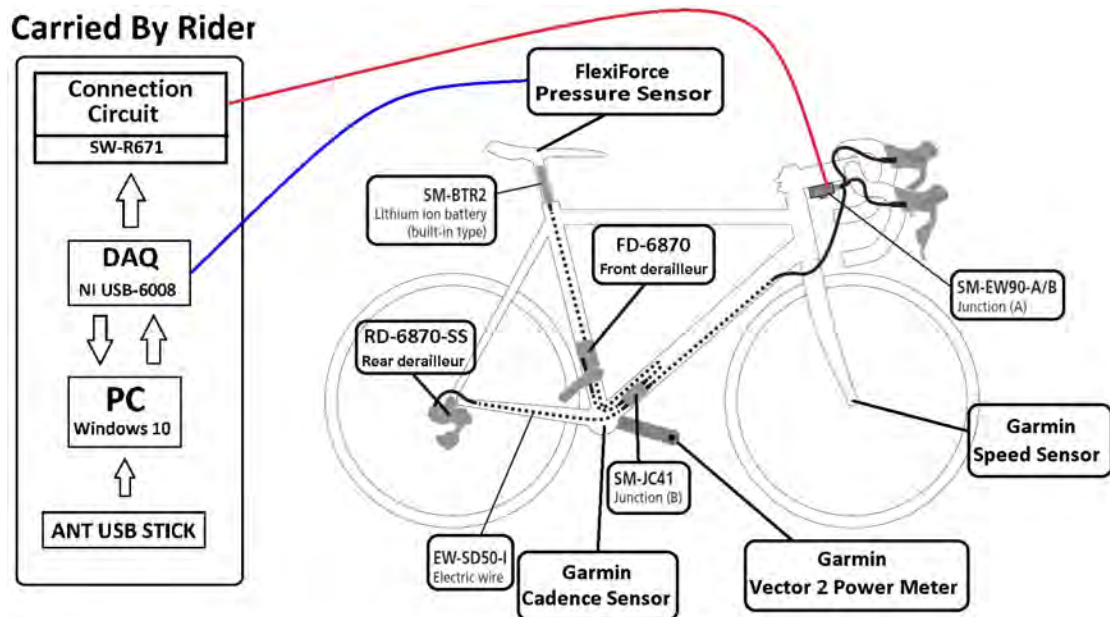


Figure 2.6: Complete overview of devices and sensors that the ATS consists of. The GSS prototype (PC, ANT USB Stick, DAQ and connection circuit) can be carried by the rider. The blue line shows the DAQ connection to the seat located force sensor. The red line shows the connection between the connection circuit and the ETS. Figure based on (*Shimano Ultegra 6870 Series Dealer's Manual*, 2014).

Chapter 3

Literature survey

3.1 Electronic transmission systems

ETS were first introduced in the year 1990 (Sweatman, n.d.) but did not become commercially successful until the introduction of Shimano Di2 in 2009 (*Company History*, 2016).

The main advantage of ETS is that they offer the possibility of being electronically controlled by a GSS. ETS have some other advantages over Mechanical Transmission Systems (MTS). Some of the advantages of ETS over MTS are listed below:

- Increased freedom in control lever positioning.
- Easier operation of control levers.
- More consistent shifting performance.
- Easier tuning and adjustment (Huang, 2009)

Some of the disadvantages of ETS compared to MTS are listed below

- Rely on battery power.
- More complex design.
- More expensive (Huang, 2009)

ETS have become popular in competitive cycling, where shifting performance is important and cost is not. ETS are a superb platform for developing an ATS.

3.2 Automatic transmission systems

Currently there are no ATS available for competitive cycling, but there is a GSS available called BioShift. The system is an additional hardware module that is fitted on a bicycle and connects to a Shimano Di2 ETS. The system interacts with the Shimano ETS and automatically selects an optimal gear for the rider based on multi-sensor data. The BioShift system has the ability to adopt to the rider's riding style and uses learning algorithms to take fatigue into account (*Bio Shift - Just Pedal*, n.d.).

The BioShift system brought the idea of creating a GSS from scratch and make it interact with a commercial ETS. The BioShift system prototype is marketed as a standalone product that collects data from wireless bike sensors and interacts with Shimano Di2 ETS. Instead of being an additional hardware module, future versions of GSS in general have the potential to be software that runs transparently on ETS as they evolve (Maker, 2014).

Shimano has been the leading manufacturer of commercial ETS for bicycles since the release of Di2 (Digital Integrated Intelligence) systems in 2009 (*Company History*, 2016). Shimano's ETS use a software called Shimano E-tube Project for adjustments of the transmission central computers. Shimano has not made an automatic gear shifting feature option available for road bike ETS, but in December 2015 the Shimano E-tube Project adopted a fully automatic gearshift update for 8-speed ETS designed for city bikes. The automated gearshifting is designed for increased convenience and comfort (*E-TUBE PROJECT*, 2015).

In late 2015, SRAM Corporation introduced their first ETS. The system is called eTAP and is the first ETS that performs gearshifts wirelessly. The shifter buttons on located on the handlebars communicate wirelessly to the motor driven derailleurs for shifting gears. This technology brings the potential of using a GSS to control gearshifts on the ETS without any hardware modifications (Rossiter, 2015).

With present ETS technology and growing number of wireless sensors, advanced ATS not only have not only become a possibility: such systems could have a high potential to be preferred as an alternative over manually operated gear shifting systems in a wide range of cycling applications, as they evolve in the future.

3.3 Previous studies on optimal cadence

Various studies in sports biomechanics have been carried out on optimal pedaling rate for cyclists. The optimal cadence differs between individuals and usually it is not the same as the rider's self selected cadence. Few aspects of previous researches will be introduced to state the importance of paying attention to cadence in cycling training. This knowledge is important when it comes to develop a GSS. How to estimate cyclist's optimal cadence values will not be discussed as such study is not within the scope of this report. Road cycling may seem to be a simple sport in terms of monotonous movements of the legs, but there are some aspects that need to be considered to achieve best possible pedaling efficiency (Beer, 2008).

One aspect of optimal cadence that has been researched, is efficiency through various intensity levels. For same power output, more force must be applied to the pedals at low cadence compared to high cadence. Pedaling with a cadence above optimal range comes at higher metabolic cost and increased heart rate compared to lower cadences, because excessive energy is spent on mechanical work moving the lower limb up and down. Low cadences at high intensity levels cause cyclist to fatigue sooner. The reason is the cyclist must apply greater amount of force on the pedals, compared when spinning at higher cadence with same power output. Equation 3.1 shows how pedaling power can be calculated from the torque τ and the rotational frequency ω . Another reason that supports the idea of keeping the cadence relatively high is increased muscle blood flow by skeletal muscle pump (Gotshall, Bauer & Fahrner, 1996). Another study that supports the idea of relatively high cadence sweetspot shows that neuromuscular efficiency altered depending on cadence. Cadences of (60, 70, 80, 90, 100) *RPM* were tested and the results showed best neuromuscular efficiency for cadence of 80 *RPM* and 90 *RPM* (Takaishi, Yasuda, Ono & Moritani, 1996).

$$P = (\tau [N \cdot m] \cdot \omega [rad/s]) [W] \quad (3.1)$$

Yet another study on determining athlete's personal optimal cadence confirmed that such cadence actually exist. The study defined the optimal cadence as the cadence that produced the highest power output for a given heart rate. The study showed about 6% decrease in cyclist power output when cadence was 20 *RPM* below or above the optimal value (Scarf et al., 2016). Optimal cadence for each individual varies with different intensity values. Research showed that optimal cadence increases with increased power output. Comparison the power output and optimal cadence showed a linear variation between the two variables (Coast & Welch, 1985).

Researches confirm that cadence is an important parameter in cycling regarding pedaling efficiency. Cadence was the primary variable to be controlled when the optimal gear selection system was designed. The information from these researches were used in the optimal gear selection system design.

3.4 ANT protocol

The ANT protocol was used for wireless sensor communications in the project. ANT™ is the property of Dynastream Innovations Inc. Information and figures from Dynastream’s online documents were used in this report with a written consent from Dynastream Innovations Inc. To access Dynastream’s online documents, the author signed up for an ANT+ Adopter account on ANT™ webpage: This is ANT (*The Wireless Sensor Network Solution*, 2016).

3.4.1 ANT basics

The ANT™ protocol is designed for ultra-low power (ULP) wireless sensor networks. The protocol offers flexibility in terms of scalability and network topologies. The ANT protocol covers the physical, network and transport OSI layers (*The OSI Model’s Seven Layers Defined and Functions Explained*, 2014), the full protocol stack is shown in Figure 3.1. The application and presentation layers are defined by the user. ANT supports 125 different radio frequencies from 2400MHz to 2524Hz, with 1MHz increments. Channel 2457MHz is reserved for ANT+. The default message rate for broadcast data from master to slave is 4Hz, but the message rate can be set from 0.5Hz to over 200Hz. Applications usually do not require such high message rates and the rate will also be limited by the system calculation capabilities (*ANT Message Protocol and Usage - Rev. 5.1*, 2014).

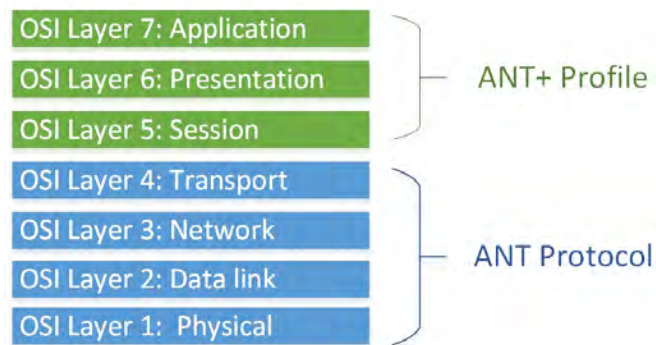


Figure 3.1: Breakdown of ANT/ANT+ according to the OSI model.

Compared to other protocols that can easily be scaled in terms of number of devices and network topologies, ANT is claimed to have the least overhead and therefore the lightest protocol in its class. Sensors powered by coin battery that run on ANT are claimed to have battery life up to three years (*ANT+ Brand*, 2016). Advantages of ANT over alternative protocols like Bluetooth Low Energy are more varieties of topology configurations and implementation simplicity (Frenzel, 2012). Figure 3.2 shows the basic structure of an ANT message.

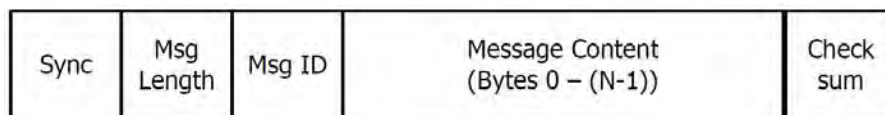


Figure 3.2: Basic structure of an ANT message. Message content includes channel number and data payload of 8 bytes. Extensions of the basic message format also exist (*ANT Message Protocol and Usage - Rev. 5.1*, 2014, p. 34).

Chapter 4

Hardware

To test the GSS prototype, an ETS system and a demo bike was required. This chapter describes the hardware that was used for building an ATS from the prototype GSS and a commercial ETS.

4.1 Commercial electronic transmission system

Shimano Di2 6870 ETS main components are listed below and the installation of the main components is shown in Figure 4.1.

- Shimano Ultegra Di2 6870 front derailleur (FD-6870), shown in Figure 4.2.
- Shimano Ultegra Di2 6870 rear derailleur (RD-6870-SS), shown in Figure 4.3.
- Shimano Di2 shifting switch for aero bar (SW-R671).
- Shimano Di2 Junction (A) (SM-EW90-A).
- Shimano Di2 Junction (B) SM-JC-41).
- Shimano Di2 Wireless Unit (SM-EWW01).
- Shimano Di2 Battery (SM-BTR2-1).

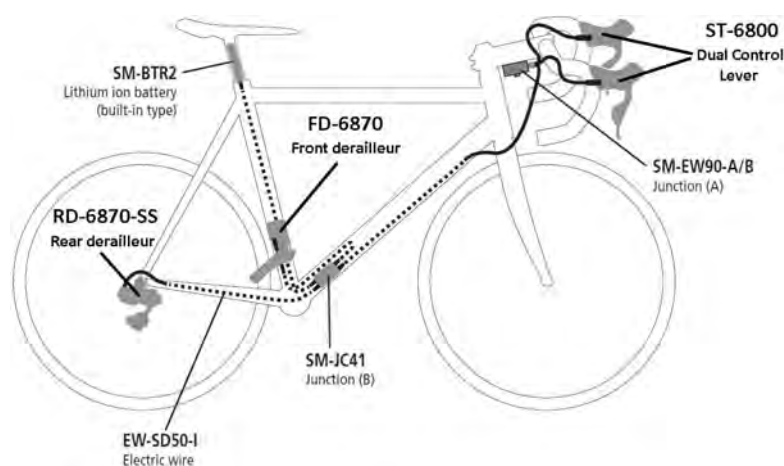


Figure 4.1: Normal installation of the Shimano Di2 ETS. The Wireless unit SM-EWW01 (not shown) was connected between Junction (B) and the Rear Derailleur. The SW-R671 shifting switches (not shown) are an optional equipment and replaced the Dual Control Levers (*Shimano Ultegra 6870 Series Dealer's Manual*, 2014).

The ETS uses two motor driven derailleurs to perform gearshifts. The Front derailleur has 2 gear settings and the rear derailleur has 11 gear settings. Both units are connect to central computer for control via bus that also supplies the derailleur units with power.



Figure 4.2: Shimano Di2 6870 front derailleur. The unit is motor driven and has 2 gear settings (*Ultegra Di2*, 2015).



Figure 4.3: Shimano Di2 6870 rear derailleur. The unit is motor driven and has 11 gear settings (*Ultegra Di2*, 2015).

4.2 DAQ device

To enable GSS to interact to ETS, NI USB-6008 DAQ device was used. The device offers full compatibility with NI LabVIEW. The device has both digital and analog I/O. Signals to actuate gear shifts were sent to ETS using 4 digital out channels. Reading of FlexiForce sensor was done using single analog in channel which was used for voltage reading. The DAQ device was also used for supply 5V voltage to drive a force sensor. The NI USB-6008 5V digital out channels have built in $4.7\text{k}\Omega$ pull-up resistor (*NI USB-6008/6009 User Guide*, 2015) as shown in Figure 4.4.

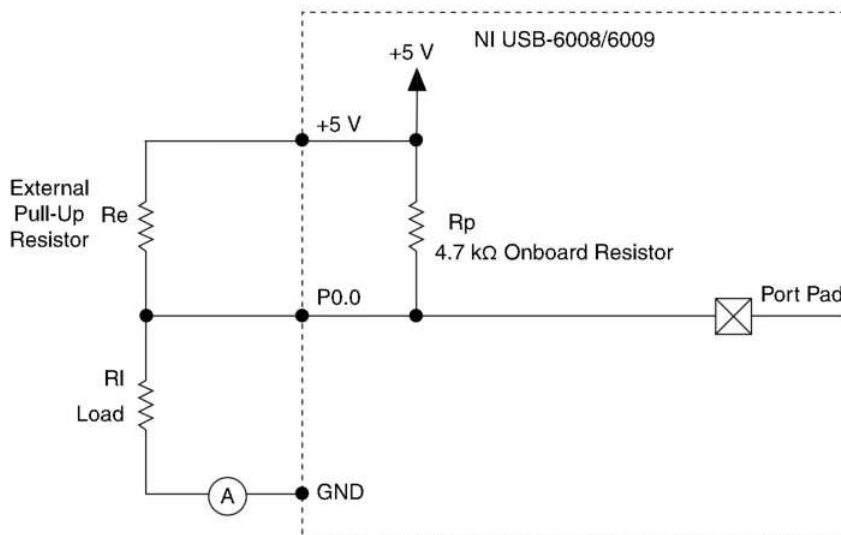


Figure 4.4: The NI USB-6008 has $4.7\text{k}\Omega$ onboard pull-up resistor for the digital channels (*NI USB-6008/6009 User Guide*, 2015, p. 20). The external pull-up resistor configuration was not used.

4.3 ANT+ sensors

4.3.1 Speed sensor

The Garmin ANT+ Speed sensor shown in Figure 4.6 is designed to be mounted on the hub of either the front or rear wheel. Unlike conventional speed sensor that give pulse each time the sensor passes a stationary magnet fitted on the bicycle frame, this sensor uses an accelerometer to count rotations. The data broadcast from the ANT+ speed sensor includes revolution count and the time of the measurement (event time). These two variables and wheel circumference are used to calculate the instantaneous speed in kilometers per hour (km/h).

4.3.2 Cadence sensor

The pedaling rate (cadence) is measured using the same principles as speed. The cadence sensor shown in Figure 4.7 measures the rotational frequency of the crank. The sensor broadcasts revolution count and event time, that is used to calculate the rotational speed in Rounds Per Minute (RPM). The cadence can also be calculated by using data from the speed sensor. Then the rotational speed of the rear wheel and drive train gear ratio is used to calculate the instantaneous cadence. The freewheel mechanism in the rear wheel hub allows the driven axle to rotate faster than the drive axle, enabling the rider to stop pedaling at any time (Murphy, 2014). The condition of freewheeling, often referred to as "coasting" has to be taken into account when designing a GSS.

4.3.3 Power meter

The cyclist's pedaling output power in *Watts*(W) was measured using a power meter. Garmin Vector 2, shown in Figure 4.8 was used for this purpose in the project, which is a pedal based power meter. The Vector 2 broadcasts data on instantaneous power and cadence, which makes a standalone cadence sensor unnecessary when Vector 2 power meter is used (*Vector 2*, 2016). The Garmin Vector 2 is a dual sided sensor, the two units communicate internally using private ANT and then the power data is broadcasted using ANT+ as shown in Figure 4.5.

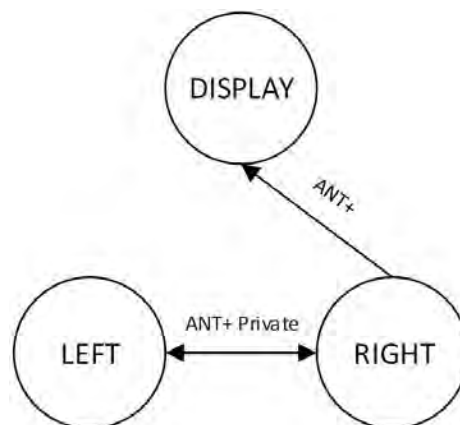


Figure 4.5: How private ANT is used for dual sided sensors as the Garmin Vector 2 (*ANT+ Device Profile - Bicycle Power. Revision 4.2., 2015, p. 30*).

4.4 ANT USB stick

The ANT USB-m stick, shown in Figure 4.9 was used to enable reception of wireless ANT+ sensor signals on a PC. The device can connect up to 8 ANT+ devices simultaneously (*USB ANT Stick*, 2016). It was used both for acquiring sensor signals and to broadcast simulated sensor data.



Figure 4.6: Garmin ANT+ speed sensor (*Bike Speed Sensor*, 2016).



Figure 4.7: Garmin ANT+ cadence sensor (*Bike Cadence Sensor*, 2016).



Figure 4.8: Garmin Vector 2 pedal based power meter and ANT+ pedal pods (*Vector 2*, 2016).



Figure 4.9: Garmin ANT USB stick for sending/receiving ANT+ signals (*USB ANT Stick*, 2016)

4.5 FlexiForce sensor

The FlexiForce® A201 shown in Figure 4.10 is an ultra-thin piezo-resistive force sensor that is designed to be flexible and easy to use (*FlexiForce Standard Model A201 - Datasheet Rev. A*, n.d.). The force sensor was mounted on the seat of a bicycle and used to give indication of whether the cyclist is riding in or out of the saddle.



Figure 4.10: FlexiForce A201 100lb model force sensor is flexible and robust and suits well for surface mounting on bicycle saddle (*FlexiForce Standard Model A201 - Datasheet Rev. A*, n.d.).

The FlexiForce Sensor was mounted on the test bike saddle surface using clear tape as shown in Figure 4.11. Ideal location of the sensor depends on each individual rider seating position, but the goal with the positioning was getting force acting on the sensor all times while seating. Movements of the rider whilst pedaling will make the force measured by the sensor oscillate, but those variations are handled by signal filtering.



Figure 4.11: FlexiForce force sensor mounted on bicycle saddle using clear tape. The sensor presence is not noticeable while seated on the saddle. Here the pressure sensitive part of the sensor is located at the center of the saddle.

Unlike the ANT+ sensors used in the project, the FlexiForce sensor requires a wired connection to a DAQ device for drive voltage and voltage reading. Circuit in Figure 4.12 was made to supply the FlexiForce sensor with drive voltage and then read the output voltage from the sensor (ELLEN, 2012). The sensor is rated for maximum $445N$ (equivalent to approximately $45kg$). The force sensitive area on the sensor is relatively small compared to the saddle surface area and when mounted on a padded bicycle saddle, a small fraction of the rider's body weight rests on the sensor. To make the sensor as sensitive as possible, the drive voltage was kept at $5V$. The sensor becomes more sensitive as the drive voltage is increased. Lower drive voltages could be used to achieve similar sensitivity using the $111N$ version of the FlexiForce A201 sensor (*FlexiForce Standard Model A201 - Datasheet Rev. A, n.d.*).

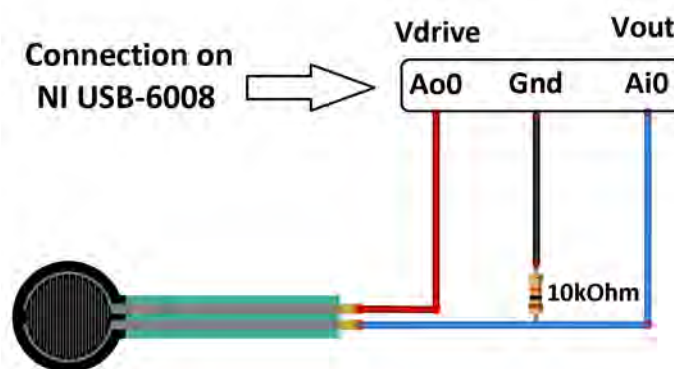


Figure 4.12: Connecting FlexiForce sensor to the NI USB-6008 unit. Drive voltage supplied by NI USB-6008 was set to $5V$. Figure created using Fritzing (*Fritzing, 2016*)

4.6 Cycling computer

Garmin Edge 520 cycling computer was used to monitor sensors and current gear status of the Shimano Di2 ETS when testing the GSS. As the GSS software had to calculate speed and cadence values from raw sensor data in accordance with ANT protocol specifications, the Garmin Edge cycling computer was used as a reference when testing the data acquisition part of the GSS program (*Edge 520*, 2016)..

The Shimano Di2 ETS uses private ANT for broadcasting the current gear. The value is available to the Garmin Device, but the GSS can not receive it unless having the private key, which requires a license from Shimano. A screenshot from the Garmin unit is shown in Figure 4.13. Keeping track of current gear status of the ETS without having a signal indicating the current gear from the ETS was solved in the GSS programming.

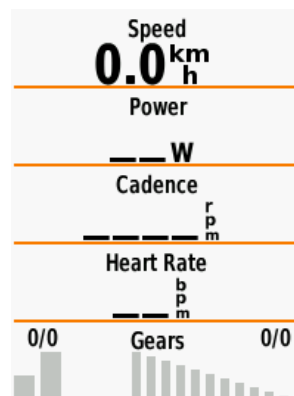


Figure 4.13: Screenshot from the Garmin Edge 520 showing ANT+ sensor parameter data fields and current gear on the Di2 transmission. The Garmin unit has access to the Di2 ANT private key to receive the current gear status.

4.7 Smart trainer

Tacx Vortex Smart stationary trainer was used to enable indoor testing and operation of the prototype GSS. The trainer has a resistance unit that is adjusted to contact the rear tire for road resistance simulation. The Vortex Smart resistance can be controlled from a computer or a smartphone to simulate riding in different slopes. When not connected to an external device, the Vortex has progressive resistance that increases as speed is increased and gives a decent approach to the feeling of riding on flat roads. The Vortex supports ANT+ communications and can provide the user with cadence, power and speed data through a cycling computer. The Vortex can be controlled by Tacx mobile application, Zwift or from the Garmin Edge 520 (*Vortex-T2180*, 2016).

When the resistance unit is in contact with the rear wheel, it always gives a certain minimum amount of resistance. When pedaling is stopped, the resistance unit stops the wheel within few seconds. When coasting was desired for experimental purposes, the resistance unit was disconnected to allow the rear wheel to spin freely.

Chapter 5

Connecting to electronic transmission

To perform gearshifts, one-way communications from the GSS to the ETS were used. The custom made connection circuit was designed to imitate a button click of each of the total 4 buttons switches of the ETS, when a signal from the DAQ was received. A detailed setup of the connection circuit is shown in Figure 5.1. The button switches connect to the ETS central computer via bus. This method uses the original circuitry in the SW-R671 switches to signal to the ETS computer, except the triggering of the switches was done electronically rather than mechanically.

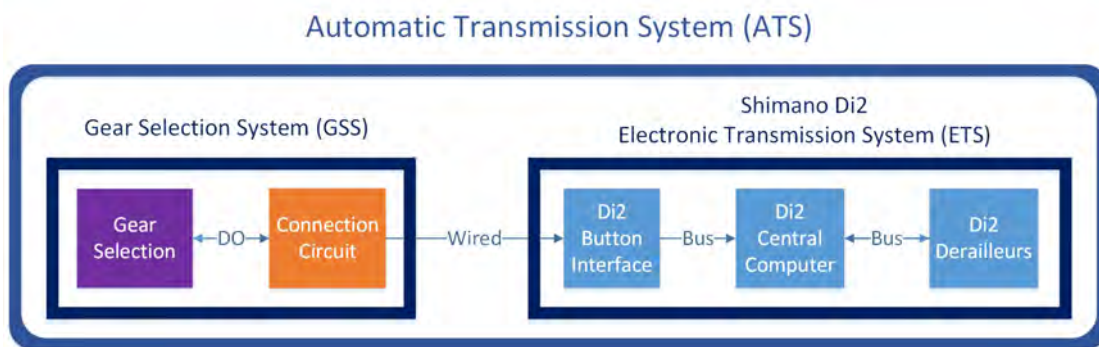


Figure 5.1: Overview of how the GSS prototype interacts with the ETS.

5.1 Modification of button switches

The connection circuit that enabled connection of the GSS to the ETS was designed to operate with Shimano SW-R671 shifting switches shown in Figure 5.2. Two switches were used, one for each derailleur. The switches were disassembled and modified so they could be triggered electronically (Sarti, 2013). This made the SW-R671 switches the only components of the Shimano ETS that needed to be modified to allow a connection to the GSS. The switches are detachable from the ETS junction box and this solution was a convenient way to integrate the gear selection module with the Shimano ETS with minimal hardware modifications.



Figure 5.2: Shimano SW-R671 left and right switches (*R671 Remote Triathlon Shifter*, 2015).

By disassembling the SW-R671 switches and removing the mechanical button switches, the internal circuit becomes exposed as shown in Figure 5.3. Beneath each mechanical button switch are two contact areas that are short-circuited when a button is pressed. To operate the switches electronically, wires were soldered to each of the contacts as shown in Figure 5.4 and wired to the connection circuit.



Figure 5.3: Disassembled button switch (SW-R671) (Sarti, 2013). Each colored ring shows the contacts for each button switch.



Figure 5.4: Disassembled button switch (SW-R671). Voltage across green and red contacts is +3.3V. Green wires were connected to common ground on DAQ device.

5.2 Connection circuit design

BJT transistors of NPN type (2N3704) (*Fairchild Semiconductor - 2N3704*, 2012) were used as switches to control whether current flows between the two contact terminals for each button shown on Figure 5.3. The transistors were used to replace mechanical action to actuate shifting one transistor for each button.

A BJT transistor has three modes of operation: cutoff, active and saturation. Transistor can be used as an amplifier or a switch. When transistor is used as a switch, it is operated in two states: ON (saturation) or OFF (cut-off). Following are model describes the three states of a transistor for DC current.

- Cutoff mode (open switch): the EBJ and CBJ are reverse biased and no current flows in the base and the emitter: $I_B = 0$ and $I_C = 0$.
- Active mode: the EBJ is forward biased and CBJ is reverse biased.
- Saturation mode (closed switch): the CEJ of the transistor acts like a short circuit and saturation current I_{CE} flows over the junction. When the transistor is in ON state, both EBJ and CBJ are forward biased (Sedra & Smith, 2011, p. 247)

A simple connection circuit was designed using common emitter configuration of the NPN transistors, shown in Figure 5.5. The SW-R671 button switch contacts were connected across collector and emitter terminals on the transistor. When the transistor is in cut-off mode, the collector and emitter terminals act as an open circuit, as a button has not been pressed. When the transistor is driven to active mode, the current starts flowing between collector and emitter terminals. With increased base current the transistor can be driven into saturation where the collector and emitter terminals act as a short-circuit.

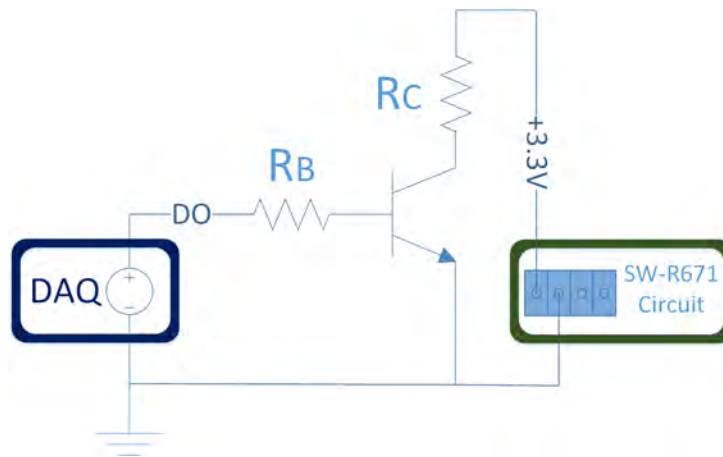


Figure 5.5: Overview of transistor based connection circuit that is the link between DAQ device and SW-R671 switches. One transistor is required for each button. Another transistor was connected in the same manner to the other two unused connections of the SW-R671 circuit.

The resistor R_B was selected so a reasonable amount of current was drawn from the DAQ device digital output channels. By choosing R_B small enough, the transistor will be driven into the state of saturation but operation in active mode is sufficient to trigger the SW-R671 circuit to execute gearshifts. The circuit shown in Figure 5.6 shows how a DO channel from NI DAQ-6008 device supplies current to the transistor base.

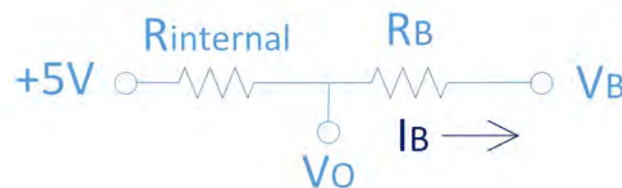


Figure 5.6: Voltage divider showing DAQ digital output with internal resistor $R_{internal}$ which is $4.7k\Omega$. The base resistor $R_B = 10k\Omega$. The base voltage V_B is $0.7V$. Output voltage V_O is the voltage measured at the DO on the DAQ device.

With $R_B = 10k\Omega$, the base current I_B was calculated using voltage divider, where the internal pull-up resistor $R_{internal} = 4.7k\Omega$ of the DAQ device was taken into account as shown in Equation 5.1. This resulted in base current $I_B \approx 0.3mA$, which was sufficient to make the ETS receive the same response from the SW-R671 circuit as when a button is pressed.

$$I_B = \frac{5V - V_B}{R_{internal} + R_B} = \frac{5V - 0.7V}{4.7k\Omega + 10k\Omega} \approx 0.3mA \quad (5.1)$$

A circuit showing connections of all transistors to the DAQ device is shown in Figure 5.7. In a default state corresponding to a button has not been pressed, no current flows from DAQ output to into the transistor base. Then the transistor acts as an open circuit. When the DAQ sends a signal to the transistor, current flows from DAQ DO into the transistor base, the circuit will act as a conductor with resistance R_C . The complete connection circuitry and connections to the DAQ device and the SW-R671 circuits is shown in Figure 5.8.

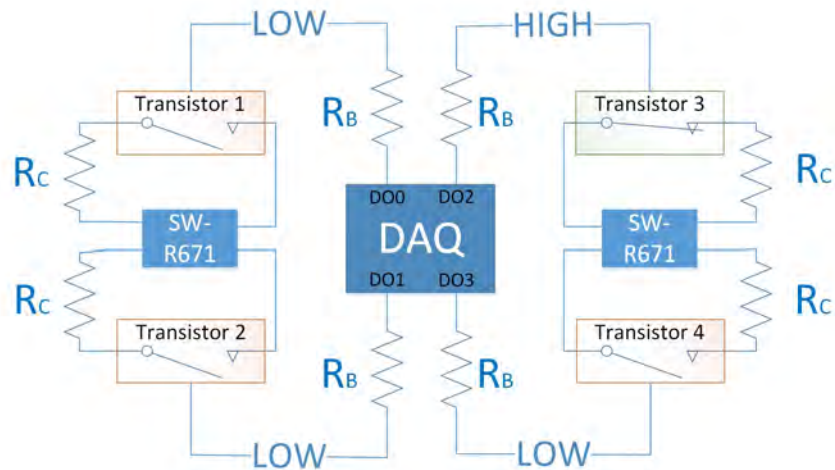


Figure 5.7: Simplified overview of the DAQ and connections to the transistors. Each transistor acts as either an open or a closed switch, depending on whether it is receiving a high or a low signal from the DAQ. As the internal resistance of the original button switch was unknown, resistors R_C were chosen to be 220Ω . This was for protecting the SW-R671 circuits, in case if short circuiting of the contacts could cause damage on the SW-R671 circuitry over time.

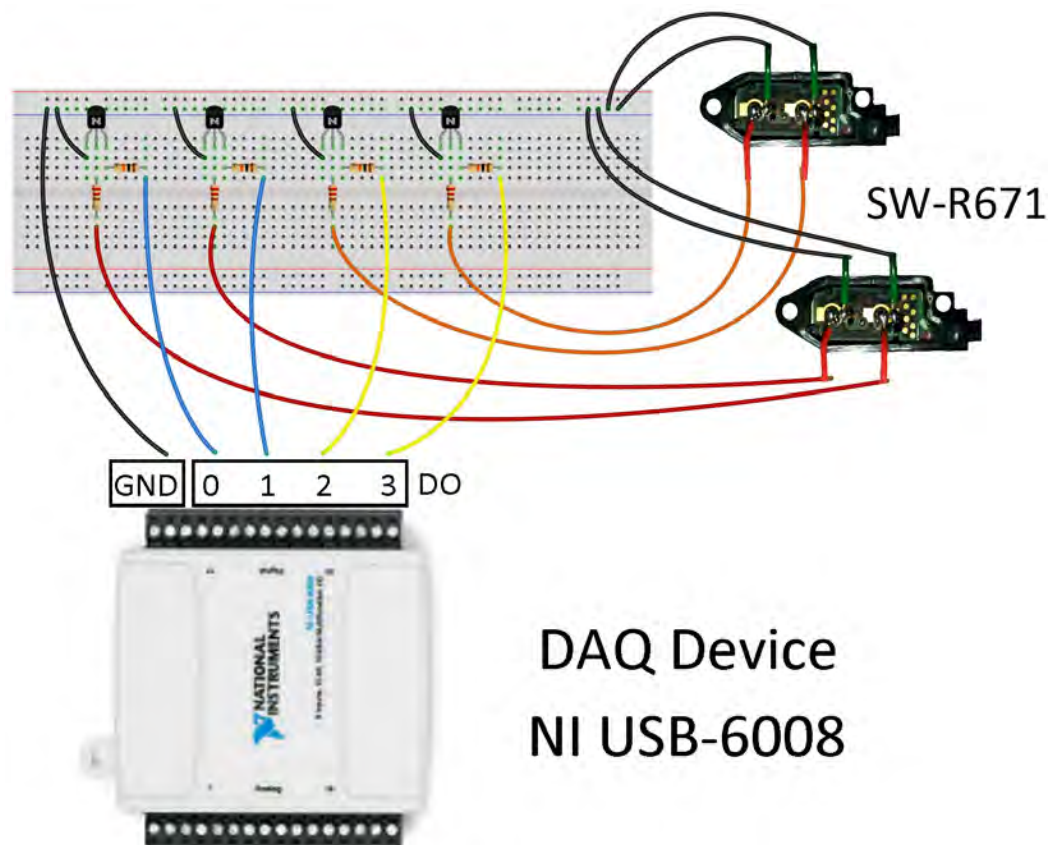


Figure 5.8: The circuit that controls shifting using NI USB-6008 (*National Instruments- Image Gallery*, 2009) digital outputs and four 2N3704 transistors. The 2N3704 pin configuration from left to right is Emitter-Collector-Base (ECB). $R_B = 10k\Omega$ and $R_E = 220\Omega$. Diagram created using Fritzing (*Fritzing*, 2016)

Chapter 6

Software for prototype development

This chapter lists the software used for development of the prototype GSS.

6.1 LabVIEW

NI LabVIEW is a graphical programming language that is widely used in the industry. As a student at HSN, the author had access to the NI LabVIEW 2015 development platform which was used in this project (*LabVIEW System Design Software*, 2016).

6.2 ANT+ toolkit by iNU Solutions

The ANT+ Toolkit developed by iNU Solutions provides device drivers to perform data acquisition from ANT+ sensors and devices in LabVIEW. Currently this is the only commercial solution for using LabVIEW with the ANT+ protocol. The toolkit requires use of specific driver for the ANT USB-m stick to work with LabVIEW, creation and installation of the driver requires the use of a NI VISA driver wizard (Mather, 2015). What makes the implementation of the ANT+ toolkit exciting is the usage of a low cost sensors in an industrial grade application like LabVIEW. Current implementations of the protocol has mainly been in sports, fitness and wellness applications, but there are dozens of opportunities in industrial applications as well (Dynastream Innovations Inc., 2016).

The introduction of the iNU Solutions toolkit brings the ANT protocol a step further into the world of industrial automation (*ANT+ Device Drivers*, 2014). The ANT+ toolkit enables the user to receive the raw data from the sensors in the form of counters and event timers for periodic event sensors as speed and cadence sensors. The user of the toolkit still has to finalize the implementation by processing of the raw sensor data, such as calculating speed and cadence values according to each device's profile specifications.

6.3 Indirectly used software

Software that was used for supporting the development of the project, for testing of equipment and making equipment configurations.

SimulANT+

SimulANT+ is a software tool for ANT+ development that was used to both simulate ANT+ transmissions and read ANT+ sensors for testing purposes (*SimulANT+*, 2016). SimulANT+ communicates with the ANT USB-m stick using a software driver which is different from the driver required when using the ANT USB-m stick in LabVIEW. The two drivers can not be used for one ANT USB-m stick at the same time.

Device drivers used in Windows for the two different applications:

- SimulANT+: ANT USB-m v.1.2.40.201 (The default driver installed by Windows)
- LabVIEW: Custom driver created using NI-VISA Driver Wizard

Shimano E-Tube Project

The E-tube software was used to make configurations on the Shimano Di2 ETS via USB connection to a PC. The Di2 shifter buttons actions are programmable and can be configured with Shimano E-tube software. This means that after the button has been connected to the control circuit, the action for each button can be programmed afterwards to achieve desired functionality of the system (*E-TUBE PROJECT*, 2015).

Zwift

Zwift is a software that provides a virtual reality for indoor training on a stationary trainer. Zwift offers wireless automatic control of a smart trainer, where the simulated road resistance is controlled automatically, based on the virtual environment and ANT+ bike sensor data. The wireless control of the trainer is based on the ANT protocol (*Zwift*, 2016).

Software interconnection

The interconnection of various software used for developing the optimal gear selection system is shown in Figure 6.1. A PC runs the GSS software on LabVIEW with ANT+ toolkit installed. When PC1 is running the optimal gear selection program, PC2 can be used to broadcast simulated sensor signals using SimulANT+ software to test if ANT+ data acquisition is working correctly. While operating the ATS with GSS running on PC 1, Zwift cycling simulator software can be run on PC 2 for automated resistance control of the smart trainer based on the virtual environment. This allows testing the GSS under dynamic conditions. Shimano E-tube project software was only used to make initial adjustments to the ETS after installation on the demo bike.

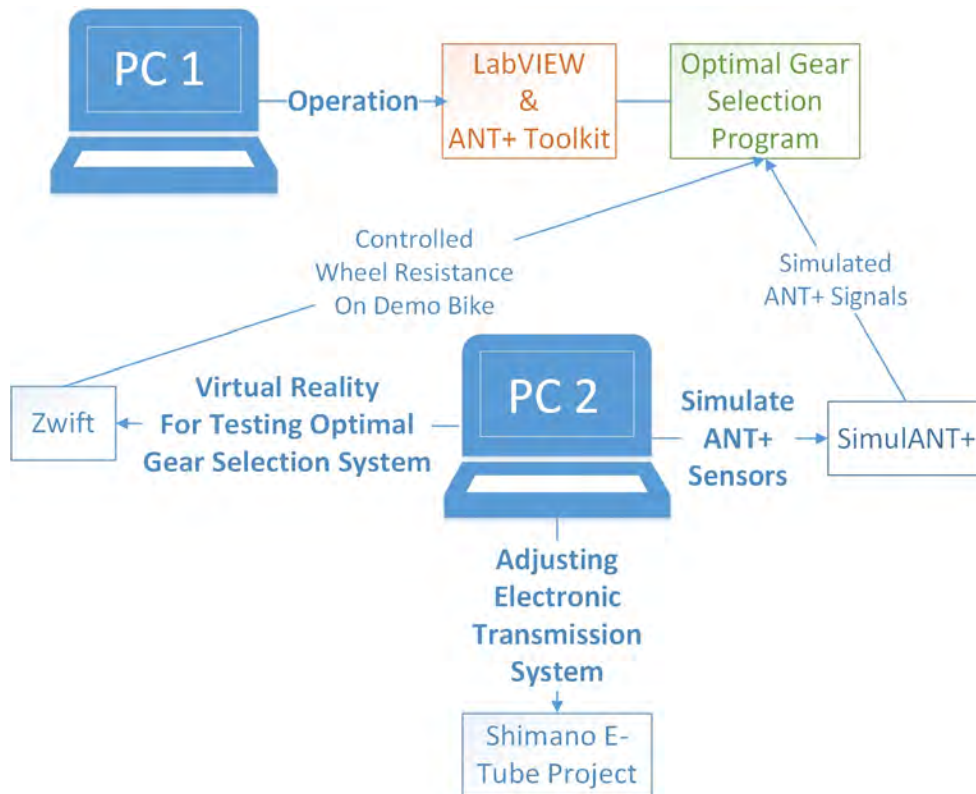


Figure 6.1: Diagram showing overview of interconnection and the purpose of different software used for developing the optimal gear selection system. LabVIEW with ANT+ toolkit is required to run the optimal gear selection system. SimulANT+ was used for testing reception of ANT+ signals when developing the Optimal Gear Selection Program and Zwift was used to control the wheel resistance when testing the Optimal Gear Selection System indoors on a stationary smart trainer. Both computers run on Windows 10 operating system.

Chapter 7

Data acquisition

Data acquisition from ANT+ sport sensors was done using the iNU Solutions ANT+ toolkit. ANT+ sensors were used to measure speed, cadence and power from the corresponding sensors. Riding position was measured by using a force sensor as described in Section 4.5.

Before deciding to use the iNU Solutions ANT+ toolkit for data acquisition, several other alternative methods were considered. Research was done on different methods to acquire the sensor data and to make it available in LabVIEW. The available options for data acquisition of ANT+ sensors will be discussed and some of the advantages and disadvantages of each approach.

7.1 Approach of choice: ANT+ Toolkit for LabVIEW

The ANT+ toolkit is a relatively new solution for ANT+ data acquisition and is still in development. This approach was chosen because of its strength of directly integrating the sensor data acquisition into LabVIEW. This eliminates the need of running separate application handling data acquisition of ANT+ sensors. The main concern when choosing this method was uncertainty for success, as the toolkit is relatively new. On the other hand, the author of this report was fortunate enough to be in direct cooperation with the author of the toolkit, and he provided support and necessary upgrades for the toolkit to enable simultaneous data acquisition of all the ANT+ sensors used in the project.

The ANT+ toolkit enables the user to read raw data from ANT+ sensors. Then protocol specifications must be implemented to calculate the measurands from the raw data. This is valid for speed and cadence sensors, while the power meter broadcasts a calculated power value.

Advantages

- Offers full integration in LabVIEW
- Potential to adopt the system for mobile platform from NI

Disadvantages

- Not freeware
- Limited experience

7.2 Alternative I: Using Dynastream's ANT+ source code

First option considered was using a software development kit available on the ANT webpage (*The Wireless Sensor Network Solution*, 2016) to build a program that pairs with the ANT+ sensors and reads the sensor data. The next step would be making the live sensor data available in LabVIEW, where the raw sensor data would be interpreted according to the ANT+ device profiles documents.

Taking this approach to acquire ANT+ sensor data would have been a more safe choice than using the ANT+ toolkit for LabVIEW, but would have required communications between the data acquisition program and LabVIEW. The DataSocket Transfer Protocol (DSTP) could be used in LabVIEW for data sharing between programs (*DataSocket Transfer Protocol (dstp) Overview*, 2016).

Advantages

- Freeware, open source code
- Proven and tested code
- Online support

Disadvantages

- Time consuming, low level programming
- Requires additions to make live sensor data accessible in LabVIEW application
- Does not support NI mobile platforms

7.3 Alternative II: Modify existing ANT+ compatible software

This method was the one the author started working with before discovering the ANT+ toolkit by iNU Solutions for LabVIEW. The concept of modifying a software that already has support for reading ANT+ bike sensors was explored, and at first glance this seemed to be the most convenient way to access the sensor data. The original idea was to use the existing software to acquire and process the sensor data and to modify it to enable communication with LabVIEW.

An open-source training analysis software called Golden Cheetah (*GoldenCheetah*, 2016) has the capabilities of reading ANT+ sport sensors and display the live data on a dashboard. The C++ source code is available on Github (*Golden Cheetah Open Source Project*, 2016). The original idea was exporting data from Golden Cheetah to LabVIEW, by programming an additional function to the Golden Cheetah code that allows data socket communications with LabVIEW using DSTP (*DataSocket Transfer Protocol (dstp) Overview*, 2016).

The journey of exploring this method of acquiring sensor data will not be described here in detail, but the main steps in the process were the following:

1. Install Linux Ubuntu 14.04 LTS (*Download Ubuntu Desktop*, 2016) virtual machine.
2. Build source code using Qt Creator (*Qt - Home*, 2016).
3. Install required drivers for ANT USB-m stick to work.
4. Explore Qt toolkits for network communications to send sensor data to LabVIEW.

Advantages:

- ANT+ sensor profiles already implemented

Disadvantages:

- Complex code implementation
- Golden Cheetah software has large overhead
- Qt Creator is not freeware
- Developer's source code is made to be built on Linux
- The gear selection prototype system would rely on both Windows and Linux

As seen above, this alternative for ANT+ sensor data acquisition has many disadvantages and was aborted when the ANT+ toolkit was offered as an option. From the author's experience with exploring different methods for data acquisition, the second choice after using the ANT+ toolkit would be using Dynastream's ANT+ source code for building a stand-alone data acquisition application.

Chapter 8

Optimal gear selection

In this chapter, the process of developing a knowledge-based system for selecting an optimal gear based on multi-sensor data will be described. Data fusion is used to calculate the optimal gear based on the multi-sensor data and the acquired knowledge..

8.1 Knowledge acquisition

To develop a knowledge-based control system that replaces human operator input for optimal gear selection, both practical human operator expertise and theory was applied. First step was knowledge acquisition: gathering necessary information regarding how the gear selection system should behave depending on various input variables. Some of the knowledge representation was done in the form of rules implemented with fuzzy logic (Negnevitsky, 2011, pp. 25–54).

Three main aspects were considered when designing the GSS behavior: Optimal cadence, mechanical restrictions of the ETS and the cyclist's riding position. Each of the aspects was collected in the gear selection knowledge base. The process of selecting an optimal gear used data fusion, where the sensor data and data from the knowledge base was fused to determine how the ETS should be controlled.

8.1.1 Optimal cadence

In competitive cycling, maintaining cadence within an optimal range is important for best pedaling efficiency. The gear selection system will choose optimal gear to keep the cadence within a predefined optimal range. The corresponding gear will therefore be considered as the optimal gear for the given situation. Based on previous researches on optimal cadence, a reasonable range of $(80 - 90) RPM$ was chosen as the optimal range to maintain (Takaishi et al., 1996). Optimal cadence varies between individuals and the range selected is not meant to be "the correct" value, but a reasonable starting point for developing the GSS. As discussed in Section 3.3, optimal cadence increases with increased levels of pedaling power. To apply this property in an useful manner, pedaling power was classified into 3 zones and optimal cadence was assigned to each of the zones.

The following information about optimal cadence was used for the knowledge base:

- Optimal cadence range is assumed to be on the range $(80 - 90) RPM$
- Optimal cadence increases with increased intensity levels

It should be noted that in general, it is not normal to keep the cadence at the optimal range at all times. Taking that approach in designing the system is not ideal for racing where conditions can be very dynamic and keeping the cadence at an even rate is usually not preferred. Keeping the cadence within certain range is preferred in situations where a relatively even pace is preferred, making the approach suitable for prolonged cycling events (Stebbins, Moore & Casazza, 2014).

8.1.2 Mechanical restrictions

Due to the mechanical properties of the modern front and rear derailleur systems, there are certain restrictions on how they should be operated in best practice. An important factor for drivetrain efficiency is that the chain alignment between front chain ring sprockets and cassette sprockets is as close to being in line with the sprockets as possible. Choosing a gear that combines the smaller front sprocket and a small rear sprocket, or the larger front sprocket with a large rear sprocket, causes cross chaining. Cross chaining is the condition where the chain crosses the centerline of the drivetrain, as shown in Figure 8.1. Shifting into gears that cause cross chaining puts unnecessary strain on the chain, causing accelerated mechanical wear on the chain and the sprockets (*Why Avoid Cross Chaining Gears On Your Bike*, 2012).

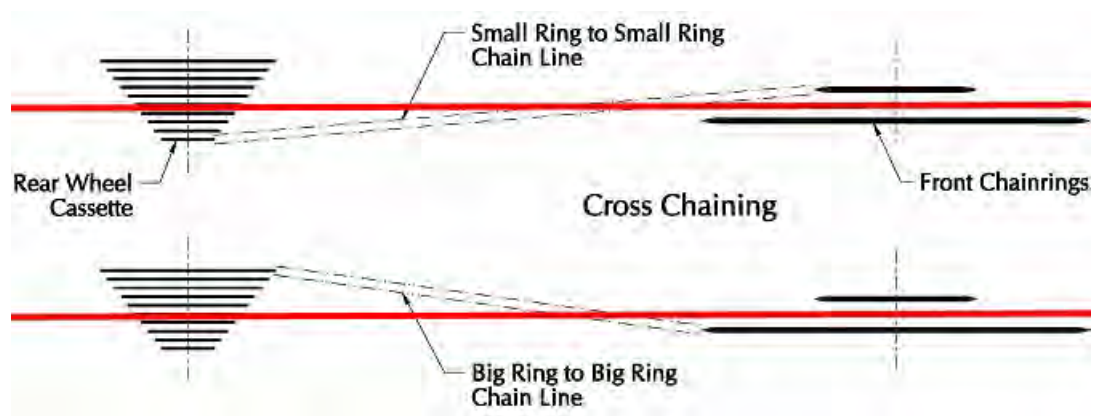


Figure 8.1: Cross chaining illustrated. The chain runs across the drivetrain centerline (displayed in red). Operating the drivetrain in this condition causes additional drivetrain stress, accelerated component wear and decreased efficiency (*Why Avoid Cross Chaining Gears On Your Bike*, 2012).

The Shimano Ultegra 6870 ETS used on the project has 2 sprockets on the front and 11 sprockets on the rear (2x11). That gives 22 possible gear combinations, but to avoid cross-chaining not all of the 22 possible combinations are used. To perform shifts and allow the chain to travel across sprockets, the drive train sprockets have specially designed profiles called shift ramps, that allow a smooth and easy shifting. For the chain to glide between sprockets, rotation of the drive train is required (Brown, 2008).

The following information about mechanical restrictions was utilized in the knowledge base:

- Selection of sprocket combinations should avoid cross-chaining
- When crank is not rotating ($cadence \approx 0RPM$), gearshifts should not be executed

8.1.3 Riding position

Riding positions were classified into two positions: riding in the saddle and riding out of the saddle. Under most circumstances cyclist sits in the saddle while pedaling, but there are two scenarios where riding out of the saddle is common: when climbing and when sprinting. When riding out of the saddle gearshifts on the front sprockets are usually not desired as they are not as smooth as shifting the rear cassette. Shifting the front derailleur under load can also cause the chain to drop off. Before standing up for climbing, an experienced rider has already chosen an appropriate gear and does required gear changes by shifting the rear cassette. In the case of sprinting the same applies: experienced cyclist has usually selected appropriate gear before standing up and sprinting, and any required gearshifts while riding out of the saddle will be done on the rear cassette, not the front rings (Maker, 2014).

Different gear shifting sequences were made for two different riding positions. The following information about gear shifting operation depending on riding position was used in the knowledge base:

- When riding out of the saddle, gear shifting on front derailleur should be disabled

8.2 Optimization approach

When designing the behavior of the knowledge based system for the GSS, the main focus was put on goal cadence (CAD_{goal}). This is a design approach that suits for certain applications that require steady performance and maximum efficiency. Prolonged road cycling events are a good example where pacing and efficiency are important and this optimization approach can be useful. The drawback of a design where cadence is kept within certain optimal range at all times, is that the cyclist has limited freedom to either decrease or increase the cadence above the optimal range without the system reacting by changing gears to maintain optimal cadence. This property of the system becomes a weakness when the system is used for commuting, where lower cadences are usually preferred. That is a problem can be solved by adjusting CAD_{goal} according to pedaling power measurement. When the measured pedaling power is lower than normally achieved when in training or competition, the system can adjust to maintain a lower cadence than the suggested optimal cadence for training.

The following 4 steps were taken in the design of the GSS, one step for each system input variable.

1. SPD: Select gear based on speed measurement to maintain cadence within predefined optimal range
2. CAD: Use cadence measurement to disable all gearshifts when $CAD = 0RPM$
3. PWR: Use power measurement to adjust the predefined optimal cadence range
4. Riding position: Disable front derailleur shifts when riding out of the saddle

Other design approaches than maintaining specific goal cadence are possible, for example goal power, and even goal heart rate (Maker, 2014).

8.3 A knowledge-based system

The necessary knowledge to build a knowledge based system was acquired in Section 8.1. In this section, the knowledge is utilized to design rules and methods for the knowledge based system.

8.3.1 Gear shifting sequence

A gear shifting sequence shown in Table 8.1 was designed for the GSS. After eliminating gear combinations that cause cross chaining, 14 gears remain. The sequence shown in Figure 8.2 consists of 14 different front and rear sprocket combinations. The sequence design fulfills gear shifting operation according to mechanical restrictions discussed in Section 8.1.2.

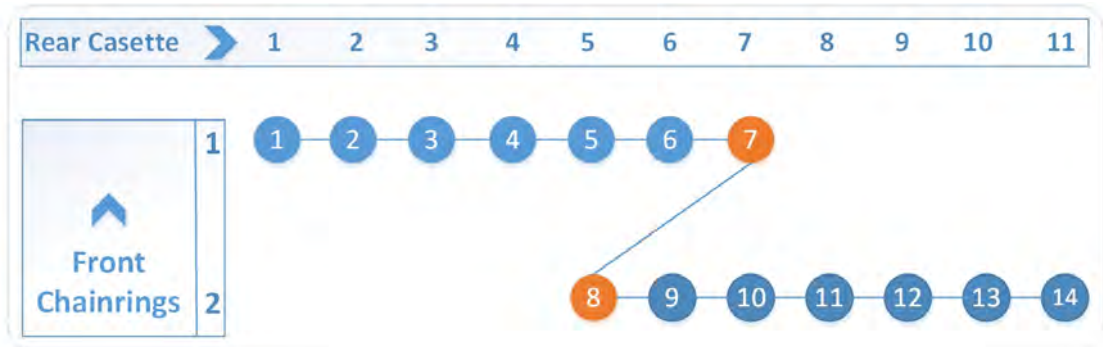


Figure 8.2: The gear shifting sequence consists of 14 of 22 possible sprocket combinations of the 2 chainrings at the front and the 11 sockets on the rear cassette. Shifting between 7th and 8th gear requires shifting the front derailleur.

In the gear shifting sequence 1st gear has the lowest available gear ratio and the 14th gear has the highest available ratio. To identify each of the 14 gears in a way that describes the sprocket combination, each gear was given a unique Identity (ID) number consisting of the front sprocket number S_F (1-2) and rear sprocket number S_R (1-11). Equation 8.1 shows how ID is calculated for 1st gear, which consists of combining $S_F = 1$ and rear $S_R = 1$. Table 8.1 shows the gear shifting sequence and the sprocket combination and ratio of each gear.

$$ID = 100 \cdot S_F + S_R = 1 \cdot 100 + 1 = 101 \quad (8.1)$$

The following steps were taking in designing the gear shifting sequence:

1. Calculate final gear ratio R for all 22 sprocket combinations,
2. Sort the list by final gear ratio in ascending order
3. Eliminate the sprocket combinations leading to cross chaining
4. Eliminate sprocket combinations that lead to more than necessary front derailleur shifts

The full list of possible sprocket combinations and which combinations were eliminated from the gear shifting sequence can be found in Table A.1 in Appendix A.3.

Table 8.1: The gear shift sequence consists of 14 different sprocket combinations. The sequence was designed to prevent cross chaining and minimize number of front derailleur shifts through the sequence. The ID for each gear indicates the sprocket combination.

No. In sequence	Front no.	Rear no.	Ratio	ID
1	1	1	1.39	101
2	1	2	1.56	102
3	1	3	1.70	103
4	1	4	1.86	104
5	1	5	2.05	105
6	1	6	2.29	106
7	1	7	2.60	107
8	2	5	2.79	205
9	2	6	3.12	206
10	2	7	3.53	207
11	2	8	3.79	208
12	2	9	4.08	209
13	2	10	4.42	210
14	2	11	4.82	211

8.3.2 Cadence and speed relationship

Speed measurement was used as the primary variable to select an optimal gear. To determine the correct gear to maintain cadence within specific range, the system must have information on the gear ratios available and the wheel Circumference (C). Wheel circumference used for the speed calculations was $C \approx 2111 \text{ mm}$ (*BikeCalc.com - How to calculate Bicycle Wheel Size*, 2014). Gear ratio R for each gear was used to calculate which gear to select according to speed, to maintain the cadence within a desired range. Equation 8.2 shows how cadence was calculated from speed. The equation was also used for deriving speed from cadence.

$$CAD = \frac{60 \cdot SPD [km/h]}{C [m] \cdot R \cdot 3.6} [RPM] \quad (8.2)$$

Gear ratio R in Equation 8.3 corresponds to a number of revolutions of the driven axle for each revolution of the drive axle. In other words, it corresponds to the number of wheel rotations for each pedaling revolution. T_F is the number of teeth on the front chainring sprocket and T_R is the number of teeth on the rear wheel cassette sprocket. The front chainring had 2 sprockets with (39,53) teeth, and the rear cassette had 11 sprockets with (28,25,23,21,19,17,15,14,13,12,11) teeth.

$$R = \frac{T_F}{T_R} \quad (8.3)$$

When making a rule base for which gear to choose to maintain cadence within a specified range, the following four steps were taken:

1. Speed was calculated for a set of cadence values for all 14 gears, see Table 8.2.
2. Cadence range high and low limits were defined, see Table 8.3
3. At upper speed limit of each gear, cadence was calculated for next gear above. At lower speed limit of each gear, cadence was calculated for next gear gear below, see Table 8.3
4. The calculated speeds for high and low cadence limits defined the speed range for each gear where cadence is within the limits, see Table 8.4.

Table 8.2: Speed (km/h) at different cadence values for each of the 14 gears in the sequence. Colors are used to group speed into sections with increments of $10km/h$ for better readability.

CAD [RPM] Gear no. in Sequence	50	60	70	80	90	100
1	8.8	10.6	12.3	14.1	15.9	17.6
2	9.9	11.9	13.8	15.8	17.8	19.8
3	10.8	12.9	15.1	17.2	19.4	21.5
4	11.8	14.1	16.5	18.9	21.2	23.6
5	13.0	15.6	18.2	20.8	23.4	26.0
6	14.5	17.4	20.3	23.2	26.1	29.0
7	16.5	19.8	23.1	26.4	29.6	32.9
8	17.7	21.2	24.7	28.3	31.8	35.3
9	19.8	23.7	27.7	31.6	35.6	39.5
10	22.4	26.8	31.3	35.8	40.2	44.7
11	24.0	28.8	33.6	38.4	43.2	48.0
12	25.8	31.0	36.2	41.3	46.5	51.7
13	28.0	33.6	39.2	44.8	50.4	56.0
14	30.5	36.6	42.7	48.8	55.0	61.1

The GSS was designed to be capable of maintaining CAD_{goal} of approximately $60RPM$, $80RPM$ and $90RPM$ depending on pedaling power. Data sets were made for each value of CAD_{goal} , where cadence variations of $\pm 10RPM$ are allowed. Table 8.3 shows how the lower and upper limits for $CAD_{goal} = 80RPM$ are set to be $70RPM$ and $90RPM$, respectively. When cadence reaches an upper limit or a lower limit, a gearshift should be performed to maintain the cadence within a range of $70RPM - 90RPM$. When cadence is below the lower limit, a downshift to next gear should be performed to keep the cadence within the range. When cadence is above the high limit, an upshift to next gear should be executed to keep the cadence within the range. Table 8.3 shows the cadence after gearshifts, at speeds where cadence is at low/high limits to confirm that changing gear would be successful in maintaining the cadence within the desired range.

Table 8.3: Gear selection for $CAD_{goal} = 80RPM$. Gear is changed when cadence reaches either low limit of $70RPM$ or high limit $90RPM$. The cadence at same speed after a gearshift confirms that each gearshift maintains the cadence at approximately $80RPM$. Table A.2 in Appendix A.4 shows equivalent data for $CAD_{goal} = 90RPM$.

CAD (LOW) 70 RPM	Speed [km/h]	Cadence at same speed after downshift	CAD (HIGH) 90 RPM	Speed [km/h]	Cadence at same speed after upshift
2 to 1	13.8	79	1 to 2	15.9	80
3 to 2	15.1	76	2 to 3	17.8	83
4 to 3	16.5	77	3 to 4	19.4	82
5 to 4	18.2	77	4 to 5	21.2	82
6 to 5	20.3	78	5 to 6	23.4	81
7 to 6	23.1	79	6 to 7	26.1	79
8 to 7	24.7	75	7 to 8	29.6	84
9 to 8	27.7	78	8 to 9	31.8	80
10 to 9	31.3	79	9 to 10	35.6	80
11 to 10	33.6	75	10 to 11	40.2	84
12 to 11	36.2	75	11 to 12	43.2	84
13 to 12	39.2	76	12 to 13	46.5	79
14 to 13	42.7	76	13 to 14	50.4	76

Speed range defined for each gear in Table 8.4 is based on Table 8.3. Extensions were made to speed ranges for 1st gear and 14th gear.

Table 8.4: Speed at low and high cadence limits. For $CAD_{goal} = 80RPM$ a cadence float of $\pm 10RPM$ is allowed. Lower cadence limit for each gear was defined at $70RPM$ and the upper limit at $90RPM$. Note the extended speed ranges for 1st gear and 14th gear. Fuzzy sets for gear selection were created according to this data. Table A.3 in Appendix A.4 shows equivalent data for $CAD_{goal} = 90RPM$.

CAD [RPM] Gear no. in Sequence	70 [RPM] (LOW) Speed [km/h]	90 [RPM] (HIGH) Speed [km/h]
1	0.0	15.9
2	13.8	17.8
3	15.1	19.4
4	16.5	21.2
5	18.2	23.4
6	20.3	26.1
7	23.1	29.6
8	24.7	31.8
9	27.7	35.6
10	31.3	40.2
11	33.6	43.2
12	36.2	46.5
13	39.2	50.4
14	42.7	-

The calculated speed range for each gear was used to create a Singular Input-Singular Output (SISO) fuzzy set that used speed as input and had appropriate gear as the output. The speed ranges for each gear overlap each other. For example, when $SPD = 19km/h$, the 3rd, 4th and 5th gear could be selected as a suitable gear according to Table 8.4. The GSS must be capable of choosing the most suitable gear at all speeds. The degree of membership of each membership function was used to determine which gear is the most suitable for the given speed.

A fuzzy set was created according to the data in Table 8.4 using the speed ranges to define 14 different membership functions. Each of the membership functions in Figure 8.3 represents a range of speed that corresponds to one of the 14 output membership functions in Figure 8.4 that represent the available gears. The Fuzzy System Designer tool in LabVIEW was used to design and implement the fuzzy sets. Membership functions of symmetric triangular shape were defined for representing each speed range and the corresponding gear was represented with a singleton.

Fuzzy logic was a powerful tool to choose the most proper gear when more than one gears met the requirement of delivering $CAD = CAD_{goal} \pm 10$. For gears with overlapping speed ranges, fuzzy logic determined which gear to choose based on the rule with highest degree of membership. Figure 8.5 shows testing of the fuzzy set in the LabVIEW Fuzzy System Designer.

Input membership functions

Speed

Membership function	Shape	Points
One	Triangle	0 ; 8 ; 15.9
Two	Triangle	13.8 ; 15.8 ; 17.8
Three	Triangle	15.1 ; 17.3 ; 19.4
Four	Triangle	16.5 ; 18.9 ; 21.2
Five	Triangle	18.2 ; 20.8 ; 23.4
Six	Triangle	20.3 ; 23.2 ; 26.1
Seven	Triangle	23.1 ; 26.4 ; 29.6
Eight	Triangle	24.7 ; 28.3 ; 31.8
Nine	Triangle	27.7 ; 31.7 ; 35.6
Ten	Triangle	31.3 ; 35.8 ; 40.2
Eleven	Triangle	33.6 ; 38.4 ; 43.2
Twelve	Triangle </td <td>36.2 ; 41.4 ; 46.5</td>	36.2 ; 41.4 ; 46.5
Thirteen	Triangle	39.2 ; 44.8 ; 50.4
Fourteen	Triangle	42.7 ; 100 ; 100

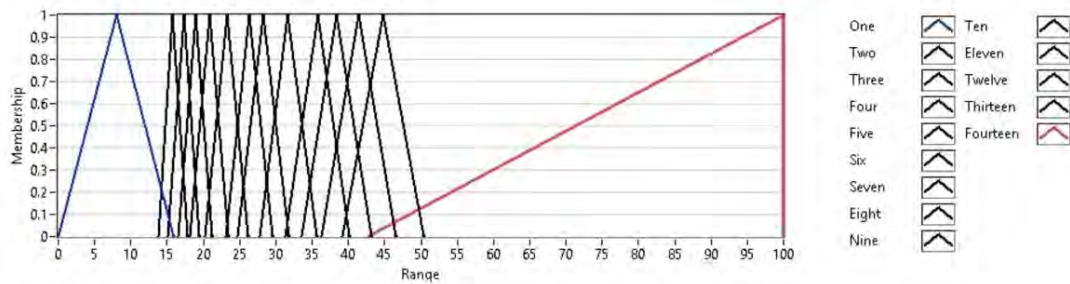


Figure 8.3: Fuzzy set for gear selection based on speed for $CAD_{goal} = 80RPM$. Fuzzy set with equivalent information for $CAD_{goal} = 90RPM$ can be found in Figure A.1 in Appendix A.5.

Output membership functions

Gear Number

Membership function	Shape	Points
1	Singleton	1
2	Singleton	2
3	Singleton	3
4	Singleton	4
5	Singleton	5
6	Singleton	6
7	Singleton	7
8	Singleton	8
9	Singleton	9
10	Singleton	10
11	Singleton	11
12	Singleton	12
13	Singleton	13
14	Singleton	14

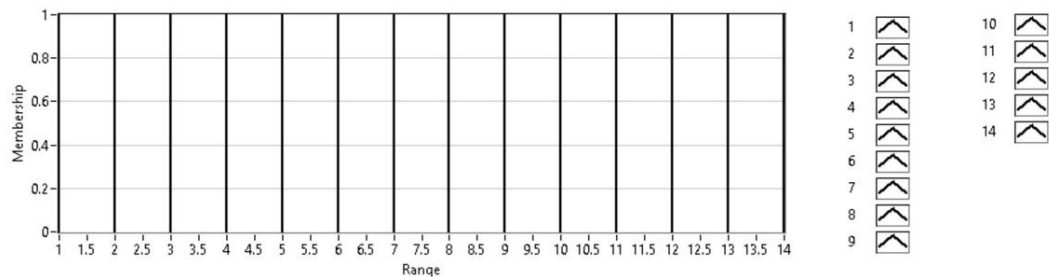


Figure 8.4: Fuzzy output membership functions. Singletons were used to represent the 14 gears available. Fuzzy set output membership functions for $CAD_{goal} = 90RPM$ are identical.

The fuzzy system rule base was defined as follows:

- IF 'Speed' IS 'One' THEN 'Gear Number' IS '1'
- IF 'Speed' IS 'Two' THEN 'Gear Number' IS '2'
- IF 'Speed' IS 'Three' THEN 'Gear Number' IS '3'
- IF 'Speed' IS 'Four' THEN 'Gear Number' IS '4'
- IF 'Speed' IS 'Five' THEN 'Gear Number' IS '5'
- IF 'Speed' IS 'Six' THEN 'Gear Number' IS '6'
- IF 'Speed' IS 'Seven' THEN 'Gear Number' IS '7'
- IF 'Speed' IS 'Eight' THEN 'Gear Number' IS '8'
- IF 'Speed' IS 'Nine' THEN 'Gear Number' IS '9'
- IF 'Speed' IS 'Ten' THEN 'Gear Number' IS '10'
- IF 'Speed' IS 'Eleven' THEN 'Gear Number' IS '11'
- IF 'Speed' IS 'Twelve' THEN 'Gear Number' IS '12'
- IF 'Speed' IS 'Thirteen' THEN 'Gear Number' IS '13'
- IF 'Speed' IS 'Fourteen' THEN 'Gear Number' IS '14'

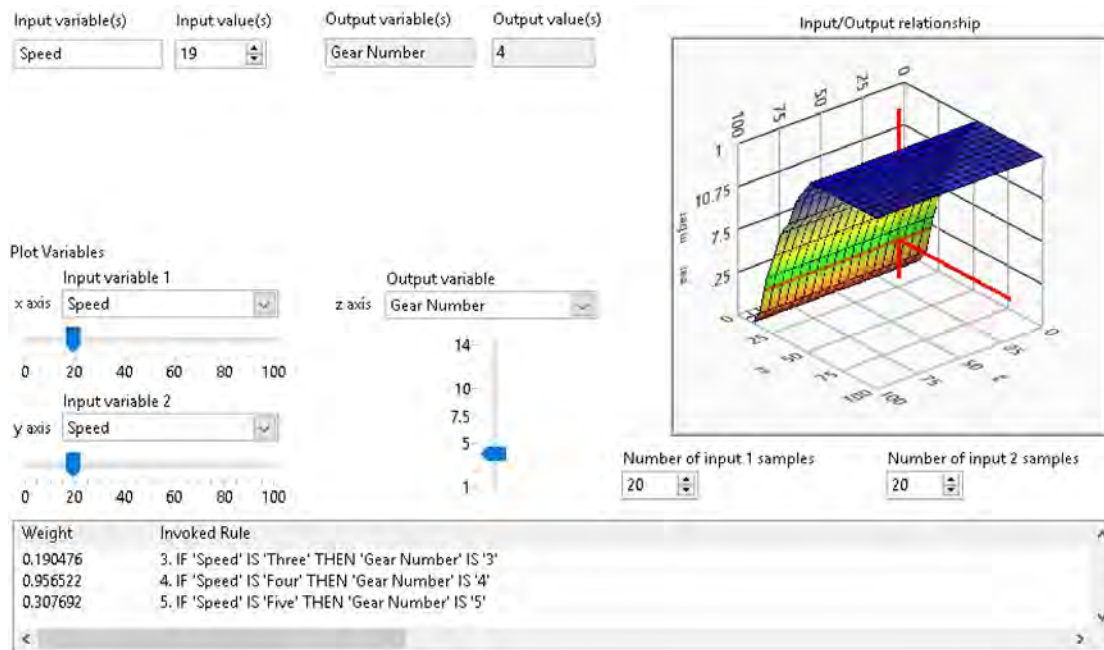


Figure 8.5: Testing of fuzzy set for $CAD_{goal} = 80RPM$ in Fuzzy System Designer in LabVIEW. For $SPD = 19km/h$, three rules are invoked simultaneously. Gear is chosen according to the rule with the highest weight each time, in this case the 4th gear. At $SPD = 19km/h$, 3rd, 4th and 5th gear have the weights, 0.19, 0.96 and 0.31 respectively.

The fuzzy set calculates a crisp output for which gear to chose for a given speed. A simulated input was fed into the fuzzy set and the output logged. The corresponding cadence was calculated and logged at the same time. Figure 8.6 shows the result where the fuzzy set selected a gear for various speeds to maintain cadence at approximately $CAD_{goal} = 80RPM$. The result showed that the average cadence for gears 2-13 was approximately $80RPM$, confirming that the fuzzy system does maintain an average cadence of $80RPM$. The 1st gear was excluded when calculating the average cadence because the lower limit for speed range for 1st gear had been extended down to $0km/h$. The 14th gear was excluded when calculating average cadence as the upper speed limit was undefined.

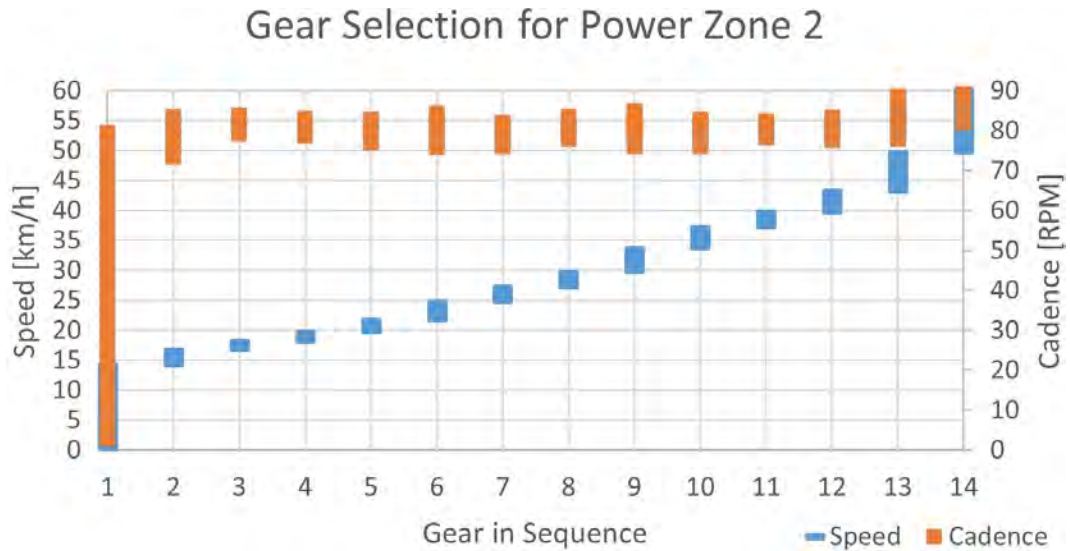


Figure 8.6: Crisp gear outputs for each speed for $CAD_{goal} = 80RPM$. Average cadence for 2nd to 13th gear was approximately $80RPM$. A fuzzy set with equivalent information for $CAD_{goal} = 90RPM$ can be found in Figure A.4 in Appendix A.6.

When the GSS selects gear according to Figure 8.6, a problem rises when cycling at speeds on the boundary regions between two gears, shown in Figure 8.7. For speeds on the boundary regions, small fluctuations in the speed will cause the GSS to shift between the two gears defining the boundaries, as the speed fluctuates. To reduce the effect of speed fluctuations at boundary regions, the GSS was designed to have a time delay on gearshifts. If the required gearshift was an upshift, the gearshift is performed without a delay. If the required gearshift was a downshift, the system was set to wait 1 s until performing the gearshift. By implementing a time delay for downshift, the system can allow speed variations without the system changing gear, until speed requiring a downshift has been established for 1 s.

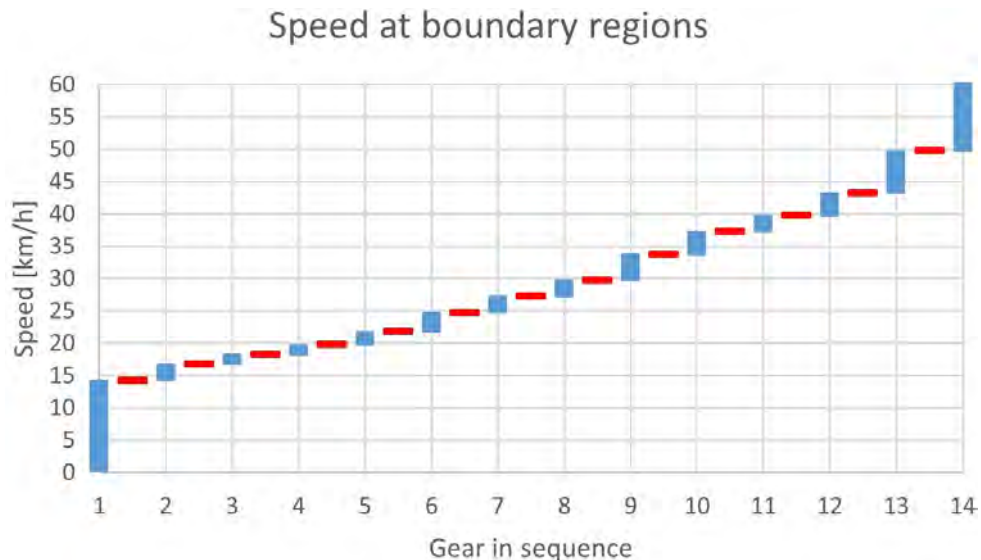


Figure 8.7: Speeds on boundary between gears are shown in red. When GSS is operated at steady pace at some of the boundary regions, the system will response to the fluctuations by shifting gears, unless a time delay for gearshifts is implemented. The time delay allows temporary float to a lower gear without a performing the corresponding gearshift.

8.3.3 Classification of pedaling power

The GSS was designed to maintain different values for CAD_{goal} depending on power output. As optimal cadence was known to increase with increased power output (Coast & Welch, 1985), the GSS was designed to seek to maintain average cadence $CAD_{goal} = 80RPM$ at medium power levels (power zone 2) and increase the CAD_{goal} to 90 RPM for high power outputs (power zone 3). For power output levels that are lower than normally achieved when training (power zone 1), the GSS seeks to maintain $CAD_{goal} = 60RPM$. This was to make the GSS more usable for everyday riding and commuting at low intensity levels.

CAD_{goal} was defined for 3 different power zones:

1. Optimal cadence for power zone 1: 60RPM
2. Optimal cadence for power zone 2: 80RPM
3. Optimal cadence for power zone 3: 90RPM

A Fuzzy set was made for classifying pedaling power into the three power zones where power magnitude was described by linguistic variables shown below:

- Low Power: 0W – 150W
- Medium Power: 100W – 400W
- High Power: 300W – 1500W

The fuzzy system rule base was defined as follows:

1. IF 'Power [W]' IS 'Low Power' THEN 'Power Zone' IS 'Zone 1'
2. IF 'Power [W]' IS 'Medium Power' THEN 'Power Zone' IS 'Zone 2'
3. IF 'Power [W]' IS 'High Power' THEN 'Power Zone' IS 'Zone 3'

The fuzzy system classifies the power into 3 zones, the crisp outputs are shown in Figure 8.8. The fuzzy set membership functions are shown in Figures A.2 and A.3 in Appendix A.5.

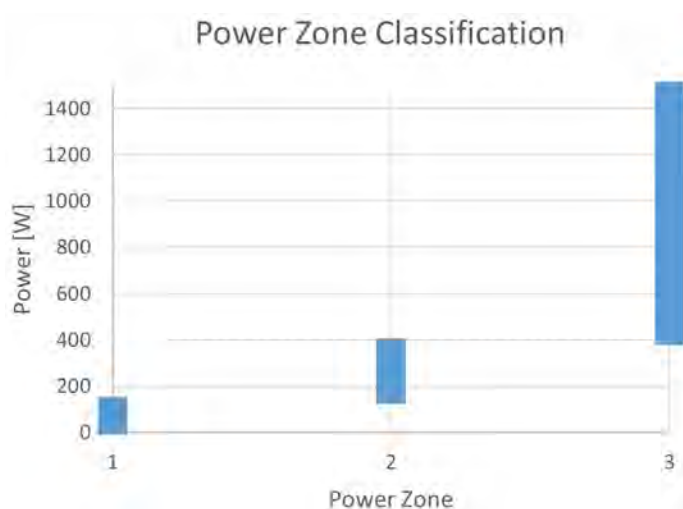


Figure 8.8: Crisp outputs from power zone classification. The three power zones do not overlap each other.

8.3.4 Shifting based on riding position

Front derailleur shifts were disabled while riding out of the saddle. This was implemented by making two modified versions of the default gear shifting sequence described in Section 8.3.1. The GSS uses one of these two sequences while the rider is riding out of the saddle. The sequence used is determined by the gear the rider is in when he/she starts riding out of the saddle. The sequence in Table 8.5 was used when starting riding out of the saddle while in front sprocket number 2 (8th gear or higher in the default sequence). The sequence in Table 8.6 was used when starting riding out of the saddle while in front sprocket number 1 (7th gear or lower in the default sequence). When changing from riding out of the saddle and back to riding in the saddle, the GSS will switch back to the default gear shifting sequence in Table 8.1 in Section 8.3.1, to enable front derailleur gearshifts.

Table 8.5: A gear shifting sequence used when starting riding out of the saddle while in front sprocket number 1. Gears 7-14 in the sequence consist of the same sprocket combinations.

No. In sequence	Front Sprocket no.	Rear Sprocket No.	Ratio	ID
1	1	1	1.39	101
2	1	2	1.56	102
3	1	3	1.70	103
4	1	4	1.86	104
5	1	5	2.05	105
6	1	6	2.29	106
7	1	7	2.60	107
8	1	7	2.60	107
9	1	7	2.60	107
10	1	7	2.60	107
11	1	7	2.60	107
12	1	7	2.60	107
13	1	7	2.60	107
14	1	7	2.60	107

Table 8.6: A gear shifting sequence used when starting riding out of the saddle while in front sprocket number 2. Gears 1-8 in the sequence consist of the same sprocket combinations.

No. In sequence	Front Sprocket No.	Rear Sprocket No.	Ratio	ID
1	2	5	2.79	205
2	2	5	2.79	205
3	2	5	2.79	205
4	2	5	2.79	205
5	2	5	2.79	205
6	2	5	2.79	205
7	2	5	2.79	205
8	2	5	2.79	205
9	2	6	3.12	206
10	2	7	3.53	207
11	2	8	3.79	208
12	2	9	4.08	209
13	2	10	4.42	210
14	2	11	4.82	211

Chapter 9

Programming of gear selection system

9.1 Program overview

All programming of the GSS was done in LabVIEW. The main components of the program include data acquisition, data processing, Fuzzy logic gear selection and Electronic Transmission Control, as shown in Figure 9.1. Functions of the program will be introduced in this chapter. Syntax sections are listed in Appendix A.8.

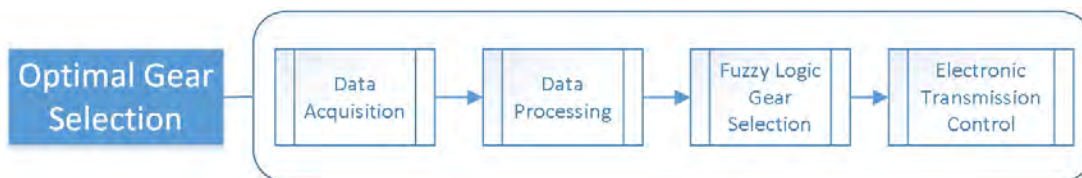


Figure 9.1: The GSS consists of the the optimal gear selection process which is divided into subprocesses. The data fusion process consists of the data processing and Fuzzy Logic Gear Selection processes.

The GSS program handles the four subprocesses in parallel loops. The placing of each subprocess in loop structure is shown in Figure 9.2.

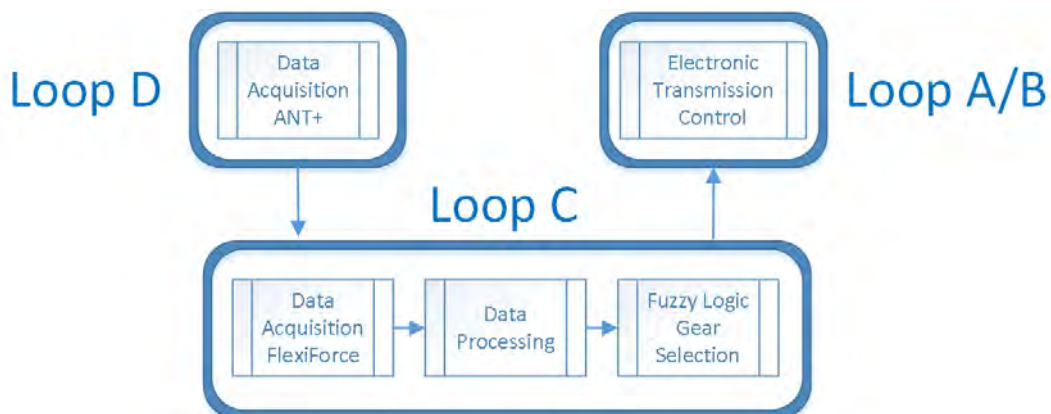


Figure 9.2: Placement of the four subprocesses within the GSS program loop structure. The arrows indicate data flow between the loops.

A detailed flowchart of the GSS program is shown in Figure 9.3. The process starts with acquiring data from sensors in the Data Acquisition subprocess. In Data Processing subprocess, desired measurands are calculated from the raw sensor data and filtering is applied where appropriate. The Fuzzy Logic Gear Selection subprocess includes the knowledge-based logic to select an optimal gear based on the sensor data. The information on an optimal gear is transferred to the Electronic Transmission Control subprocess which handles the necessary actions to shift the ETS into the optimal gear.

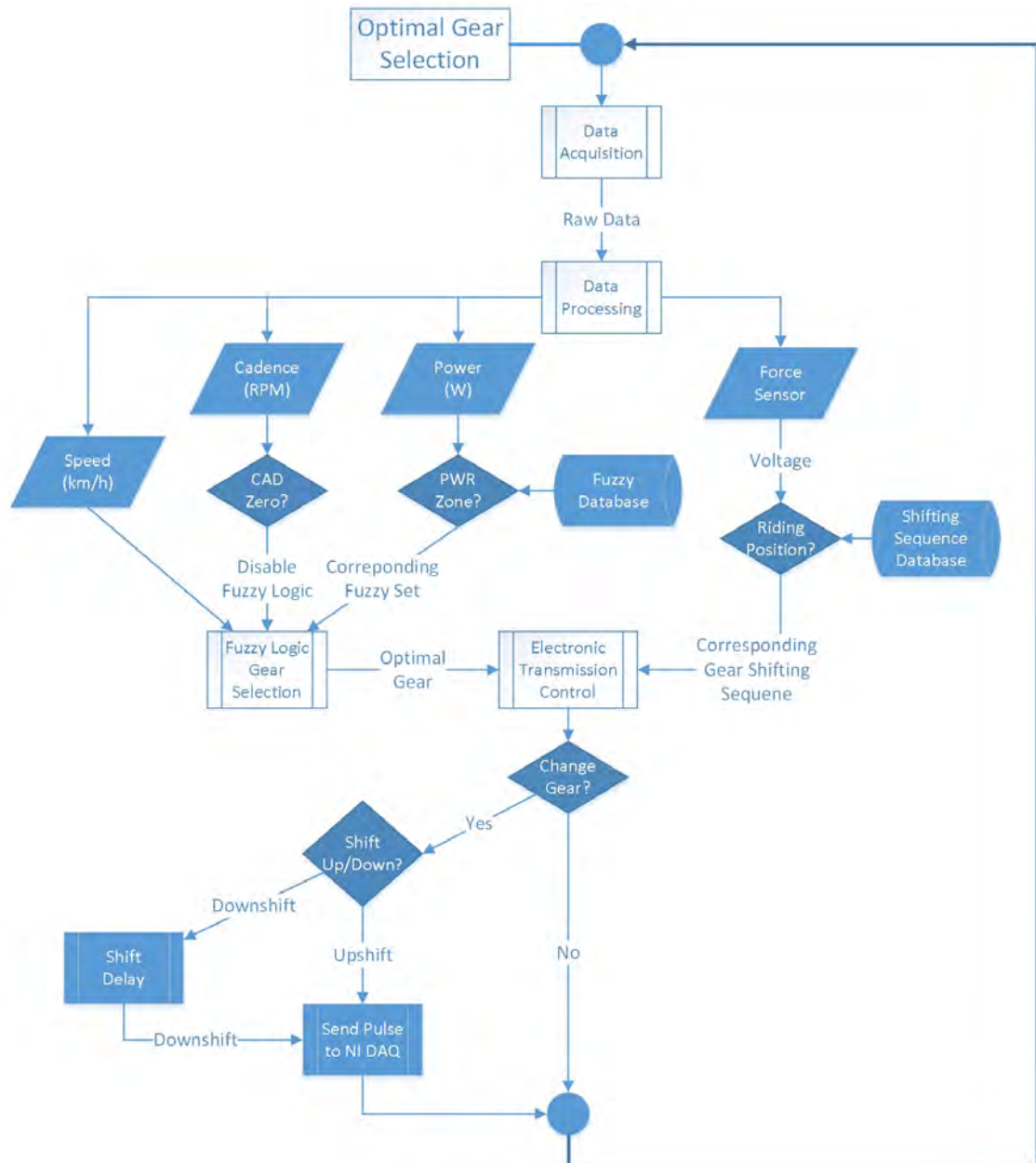


Figure 9.3: Optimal gear selection program flowchart. The Optimal gear selection process is divided into 4 subprocesses: Data acquisition, data processing, fuzzy logic gear selection and electronic transmission control.

9.2 Graphical user interface

When the GSS software is operated, user inputs are required after the system is started for connecting to the ANT+ sensors and to select which sensors the system will use as control inputs. For testing the GSS software, the GUI can be used for simulating sensor inputs, monitoring system status, log variables, making adjustments to system parameters and to manually control the ETS if desired. Figure 9.4 shows the main dashboard which displays the sensor readings and transmission control status. The user can choose to use simulated sensor inputs and test the system by manually adjusting control inputs.

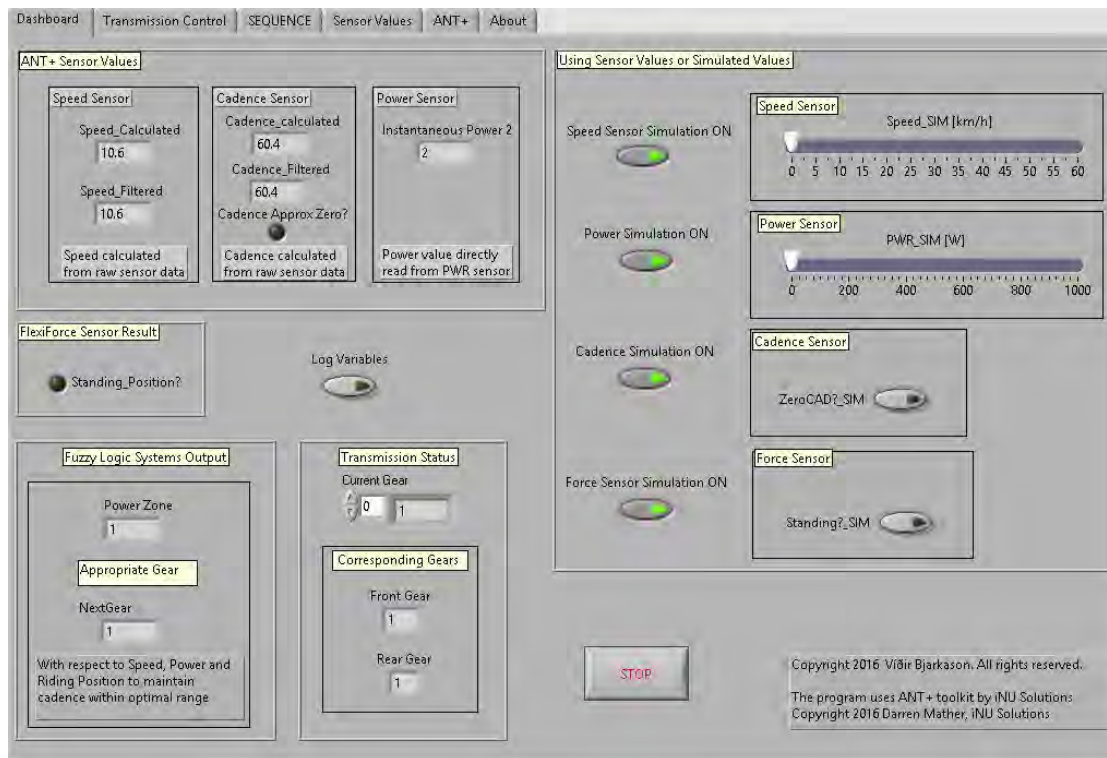


Figure 9.4: The main panel on the GUI. The dashboard displays sensor readings and transmission control status. User can choose to manually control the control inputs to test the system.

When the GSS software is started, the system assumes the ETS is in 1st gear. The transmission control panel shown in Figure A.5 in Appendix A.7 allows the user to execute gearshifts on the ETS manually for initial calibration of the system. Adjustments can be made on time between gearshifts when multiple shifts are executed. The time delay that occurs from the moment a gearshift is required until it is executed can be adjusted individually for upshifts and downshifts. The sequence panel shown in Figure A.6 in Appendix A.7 displays information from the knowledge base on the gear shifting sequences discussed in Section 8.3.1 and Section 8.3.4.

The values received from sensors are displayed on the sensor values panel shown in Figure A.7 in Appendix A.7. Speed and cadence sensors receive *cumulative revolution count* and *event time [s]* and the values are displayed for each sensor. These values are used to calculate the corresponding measurands which are displayed on the dashboard panel. Data acquisition from ANT+ sensor was achieved using ANT+ toolkit from iNU Solutions and an example syntax for acquiring data from the sensors simultaneously. The user connects each sensor by following instructions on the ANT+ panel shown in Figure A.8 in Appendix A.7.

9.3 Process 1: Data acquisition

9.3.1 FlexiForce data acquisition

Data was acquired from the seat mounted FlexiForce sensor using the NI USB 6008 device. Signal Filtering was done using LabVIEW's built-in moving average filter. The syntax for the data acquisition process can be seen in Figure A.9 in Appendix A.8.

9.3.2 ANT+ data acquisition

A syntax from iNU Solutions using the ANT+ toolkit was implemented for the ANT+ data acquisition. A decode VI included in the ANT+ toolkit receives an incoming data packet and outputs data for the available data pages. The raw data received from the sensors was fed to the next sub process: the data processing. The code for ANT+ data acquisition is copyrighted by iNU Solutions and will not be presented in the report.

9.4 Process 2: Data processing

Speed profile implementation

Speed data was read from ANT speed device profile Data Page 5, shown in Figure A.18 in Appendix A.9.1. ANT+ SPD device profile document defines how instantaneous speed is calculated from the raw sensor data. Equation 9.1 shows how speed is calculated using the *cumulative revolution count* and *event time [s]*. The multiplication constant 1024 comes from the time unit which is 1/1024 seconds (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014, p. 24*). The program calculated the speed according to the equation, the code can be found in Figure A.10 in Appendix A.8. The syntax for implementing the speed calculations can be seen in Figure A.11 in Appendix A.8.

$$Speed = \frac{Circumference \cdot (RevCount_N - RevCount_{N-1}) \cdot 1024}{(MeasTime_N - MeasTime_{N-1})} [m/s] \quad (9.1)$$

Cadence profile implementation

Cadence data was read from ANT speed device profile Data Page 0, shown in Figure A.19 in Appendix A.9.2. ANT+ CAD device profile document defines how instantaneous cadence is calculated from the raw sensor data. Equation 9.2 shows how cadence is calculated (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014, p. 36*). The program calculated the speed according to the equation, the code can be found in Figure A.12 in Appendix A.8. The syntax for implementing the speed calculations can be seen in Figure A.13 in Appendix A.8.

$$Cadence = \frac{60 \cdot (RevCount_N - RevCount_{N-1}) \cdot 1024}{(MeasTime_N - MeasTime_{N-1})} [RPM] \quad (9.2)$$

The implementation of the cadence calculation differs from the speed calculations as the cadence sensor did not broadcast a stop value indicator. When the sensor is not rotating, the event counter does not get updated between loop runs over time. To determine when the cadence was zero, a method was made to check how many loop runs the event counter value stays the same. That was done by putting the event counter in an array with the latest 20 values. In each loop run a comparison was made to check whether all the indices of the array are the same or not. If all 20 indices were the same, it was an indicator that the event counter had not updated in 20 loop runs in a row. That corresponds to setting the cadence to zero when $CAD < 15 RPM$, as $20 \cdot 200ms = 4s$ and 1 revolution per 4 seconds corresponds to $0.25 Hz = 15 RPM$.

Power profile implementation

Power data was read from ANT speed device profile Data Page 16, shown in Figure A.20 in Appendix A.9.3. The Garmin Vector 2 Power meter calculates and broadcasts the power value, therefore no additional programming was required to achieve the power value. Equation 9.3 shows how average power is calculated according to ANT+ PWR profile document (*ANT+ Device Profile - Bicycle Power. Revision 4.2., 2015, p. 41*).

$$Power_{AVE} = Torque_{AVE} \cdot AngularVelocity_{AVE} [Watts] \quad (9.3)$$

9.5 Process 3: Fuzzy logic gear selection

An optimal gear is selected using fuzzy logic. The fuzzy set uses speed as input variable. The power zone is used to determine which fuzzy set is used, as different fuzzy sets are used to achieve different goal cadence. When the cadence is zero, the process of calculating an optimal gear is disabled. Fuzzy logic is also used to classify the power into 3 zones. In each loop run, a value for an optimal gear is calculated. After an optimal gear has been determined, the signal goes through a VI that adds a time delay, if the required gearshift is a downshift. The default 1 s upshift delay allowed speed fluctuations when riding at speeds at boundary regions without undesired gearshifts being performed, as discussed in Section 8.3.2. The syntax is shown in Figure A.14 in Appendix A.8.

9.6 Process 4: Electronic transmission control

The electronic transmission control process handles the execution of gearshifts on the ETS and keeps track of the current gear. The tasks accomplished by the process are shown in Figure 9.5. A complete syntax for loop A can be found in Figure A.16 in Appendix A.8 and the syntax for loop B can be found in Figure A.17 in Appendix A.8.

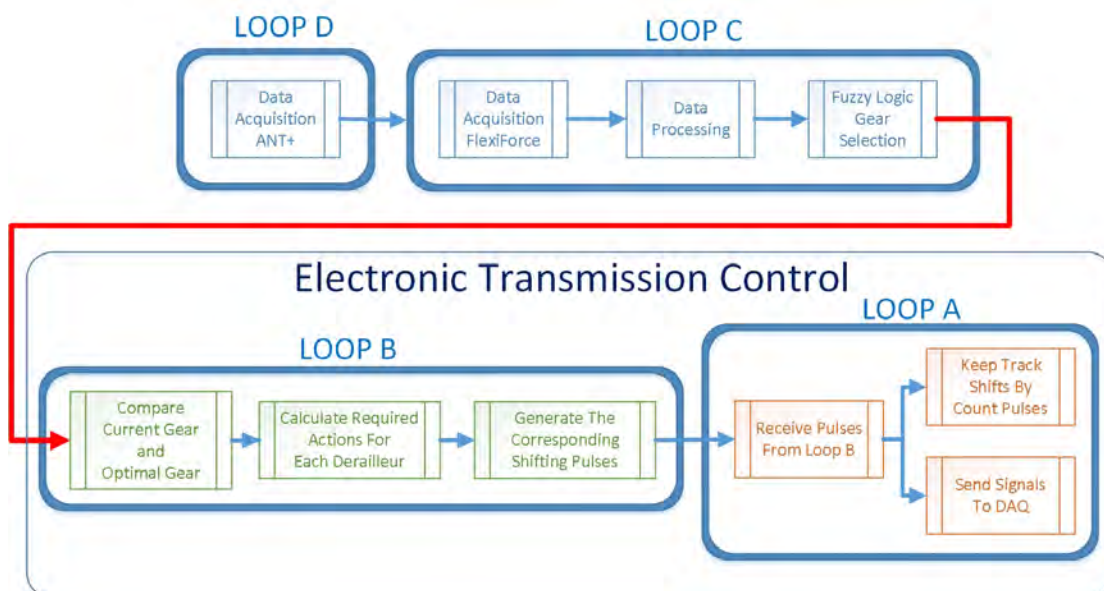


Figure 9.5: Detailed tasks within the electronic transmission control subprocess. Information flow between different loops is indicated with arrows. The electronic transmission control is handled by loops A and B. Loop B is only executed when an optimal gear value (red line) from loop C is changed. Variables are shared between loops using local variables in LabVIEW.

When the value for an optimal gear is changed, a task in loop B will compare the new optimal gear to the current gear the system is in. Required actions for each of the derailleurs are calculated according to the gear shifting sequence. Different gear sequences are used when riding in standing position, which disable gearshifts that require front derailer shifts. The syntax for choosing an appropriate gear shifting sequence can be seen in Figure A.15 in Appendix A.8. An overview of the process within loop B that generates pulses to perform shifts on each derailleure after the appropriate action(s) have been determined, is shown in Figure 9.6.

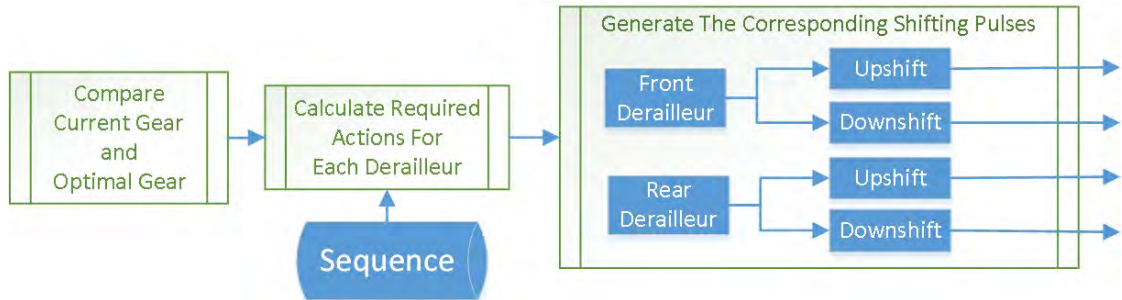


Figure 9.6: More detailed explanation of tasks inside loop B. Required actions for each derailleure are determined according to the gear shifting sequence. Then pulses are generated for each derailleure to perform the corresponding action.

Chapter 10

Prototype test plan

This chapter describes how the GSS prototype was tested. A demo bike with an ETS and bike sensors installed was used for testing. The GSS prototype was connected to the ETS and the system tested with the demo bike mounted on a stationary indoor trainer. Figure 10.1 shows the complete overview of the components of the GSS and how they were interconnected with the ETS and the sensors.

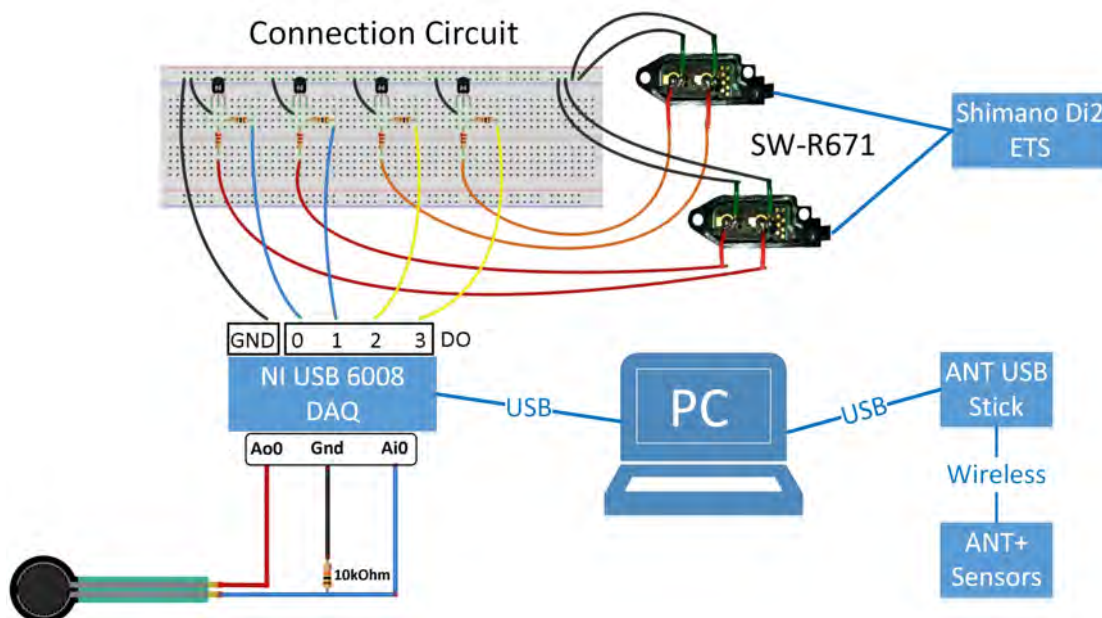


Figure 10.1: Overview of how all components of the GSS prototype are internally connected, and how they connect to the ETS and sensors.

The tests described in this test plan are used to confirm that the prototype system fulfills requirements regarding operational performance. The prototype GSS uses speed, cadence, power and riding position as input variables. Measurements should be taken from the prototype GSS while operating on a stationary indoor trainer. To test the effects on the input variables on the GSS, 3 separate tests are performed to isolate the effect of each input variable. The stationary smart trainer is used to provide progressive resistance that increases with speed.

A test plan document was created as a guide for testing the GSS prototype. The test plan document gives guidelines in set-up of the prototype system and how to perform the tests. A checklist with criteria for passing/failing the tests is included in the test document. A printable version of the test plan document can be found in Appendix A.10.

10.0.1 Test 1

Riding in the saddle: To test the GSS performance in maintaining optimal cadence for different power zones. Power zone is kept constant by simulating the power sensor output. The rider pedals and builds up speed until the GSS has shifted through the majority of the gears. Then the rider should slow down to rest. This test is divided into two separate tests where the GSS seeks to maintain different goal cadences.

Test 1a

Optimal cadence according to power zone 2. Optimal gear selection is set according to power zone 2. The GSS goal cadence is $80RPM$.

Expected results: When speed is increased, the GSS should perform gearshifts to maintain the preset $80RPM$ goal cadence. The average cadence is expected to be $(80 \pm 5)RPM$.

Test 1b

Optimal cadence according to power zone 3. Optimal gear selection is set according to power zone 3. The GSS goal cadence is $90RPM$.

Expected results: When speed is increased, the GSS should perform gearshifts to maintain the preset $90RPM$ goal cadence. The average cadence is expected to be $(90 \pm 5)RPM$.

10.0.2 Test 2

Riding out of the saddle: To test the effect of the riding position on the GSS, the rider should ride in a standing position when a gearshift between 7th and 8th gear is required. Riding positions should be altered. The rider starts pedaling out of the saddle, builds up speed until the system has shifted to the highest gear allowed in a standing position. Then the rider sits down so the system can shift the front derailleur. Speed should be increased until the system has shifted into the highest gear. After that the rider should slow down to rest.

Expected results: When riding in a standing position, the GSS should not perform gearshifts that require front derailleur movements. According to the gear shifting sequence, shifting from 7th to 8th gear, or vice versa will be disabled until the rider moves to a sitting position.

10.0.3 Test 3

Coasting: To test the effect of stopping the pedaling movement on the GSS, the rider starts pedaling and builds up speed until the GSS has shifted higher than the 10th gear. To allow coasting, disconnect the resistance unit on the stationary trainer from rear wheel and stop pedaling at the same time. The rider lets the rear wheel coast for approximately $30s$ and starts pedal again before slowing down to rest.

Expected results: When pedaling is stopped the cadence measurement should indicate $CAD = 0$. Then the GSS should not perform any gearshifts. The GSS starts shifting gears when pedaling is started again. Note: an assistant person is required to disconnect the resistance unit from the rear wheel, while the rider is on the bike.

Chapter 11

Results and discussion

11.1 ANT device profile implementation test

This Section covers the fulfillment of the requirements for processing the raw sensor data acquired from the ANT+ sensors. The speed and cadence measurands were calculated from the raw sensor data. To confirm the implementation of calculating the measurands was correct, the sensor data was logged.

Measurements were taken for the SPD and CAD sensors simultaneously to confirm they were working correctly and the implementations of the ANT+ device profiles were correct. According to Equation 8.2, speed and cadence are linearly related and studying trends in simultaneous readings of the two sensors reveal any erroneous behavior in the sensor readings. The demo bike was mounted on the stationary trainer with the resistance unit disconnected to allow the back wheel to spin freely when the crank was rotated. For the log session, the bicycle was kept in 7th gear. The logging was started from a rolling start, the wheel accelerated to a approximately 65 km/h then let freewheel until coming to a halt. Figure 11.1 shows the filtered speed sensor and cadence sensor readings. The speed is a linear function of the cadence and this relationship can be seen on the graph as the shape of the curves is similar, until the crank rotation is stopped at time $t \approx 12\text{ s}$ and the back wheel starts coasting. The results from the two separate sensor measurements show clearly the linear relationship between speed and cadence, and confirm that data acquisition for both sensor is working properly.

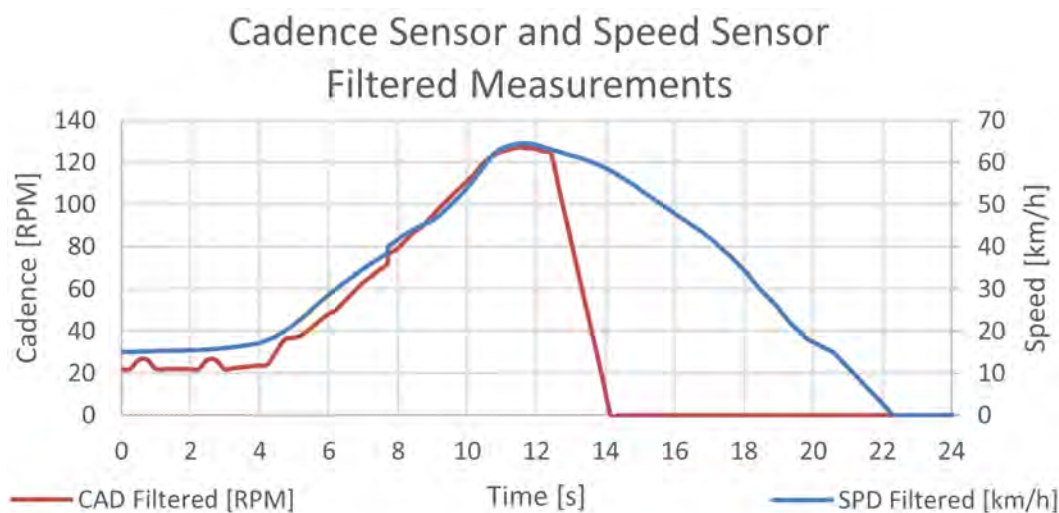


Figure 11.1: Filtered speed and cadence measurements, showing the linear relationship. Pedaling was stopped at $t \approx 12\text{ s}$. Filtering was done using moving average filter in LabVIEW.

11.2 Detection of riding position

The FlexiForce sensor was tested on the demo bike on a stationary trainer with a 75 kg male rider. The rider started in a standing position, then sat on the saddle and pedaled. The rider started in a standing position and then alternated between standing and sitting positions four times. Figure 11.2 shows unfiltered sensor readings and Figure 11.3 shows filtered sensor readings from the sensor log. These results proved that the FlexiForce sensor could be used to determine whether the rider is riding in or out of the saddle. When not sitting on the saddle, filtered voltage measurement was always below 0.1 V, which is the criteria used to determine the riding position.

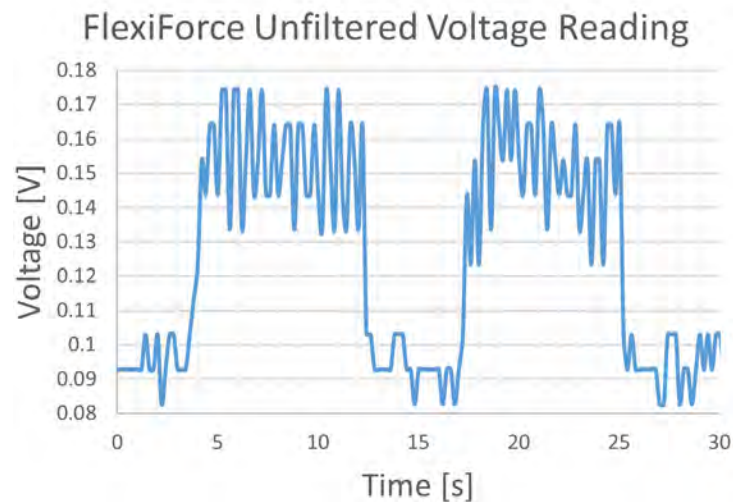


Figure 11.2: Voltage reading over around half minute interval where rider started standing and changed the riding position four times. While standing, no pressure was applied to the sensor and the voltage reading measured was 0.09 V. Samples were collected at 4 Hz and drive voltage was 5 V.

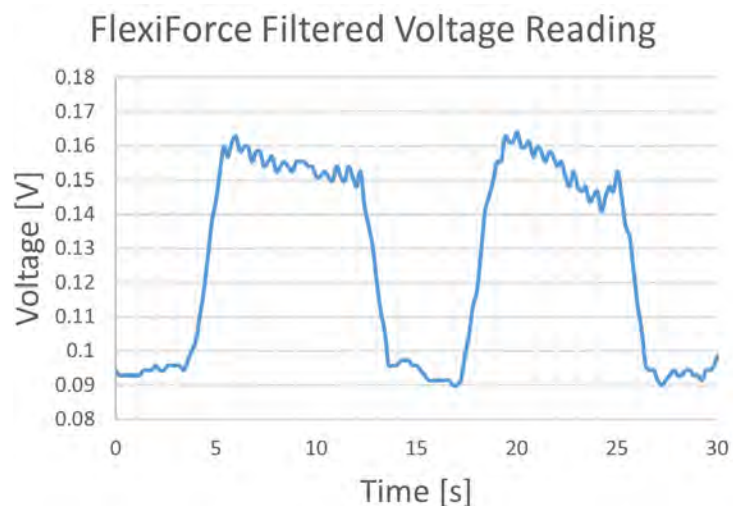


Figure 11.3: Plot of same voltage measurement as in Figure 11.2, but here with moving average filter for smoothing of the signal both for reduce the effect of oscillations in measurements due to movements of the rider and noise. Samples were collected at 4 Hz and sensor drive voltage was 5 V.

11.3 Testing of prototype according to test document

This section covers testing of the system according to the test plan in Chapter 10. Measurements were taken from the prototype GSS and the results analyzed.

11.3.1 Test 1 - Riding in the saddle

This test was done to test if the GSS system was successful maintaining the set goal cadence (CAD_{goal}) which was $80RPM$ for power zone 2 and $90RPM$ for power zone 3. The test also confirmed that the system performed gearshifts according to the gear shifting sequence defined in Section 8.3.1. The GSS uses 3 different values for optimal cadence, depending on power output. Power was classified into zone 1, zone 2 and zone 3. The CAD_{goal} for the three zones was $60RPM$, $80RPM$ and $90RPM$ respectively. Results for system operation in power zones 2 and 3 will be presented.

Test 1a - Power zone 2

Cadence variations as speed is increased when the system is operating in power zone 2 are shown in Figure 11.4. The CAD_{goal} for power zone 2 was $80RPM$, but the test showed that the GSS kept an average cadence of $95RPM$ on the time interval $10s - 50s$, which is $15RPM$ or approximately 19% higher than the goal cadence that was determined as optimal. The cadence was consistent, but the average cadence was considerably higher than the $80RPM$ goal cadence. The GSS updated G_C accordingly to G_N after a $0.2s$ time delay, which can be explained by the $200ms$ loop timing in the GSS program. The latency from the ETS receiving a signal until performing a gearshift can be neglected. Average cadence closer to the $80RPM$ goal cadence could possibly have been achieved by reducing the latency in the GSS program. According to the check list in the document in Appendix A.10, the test failed.

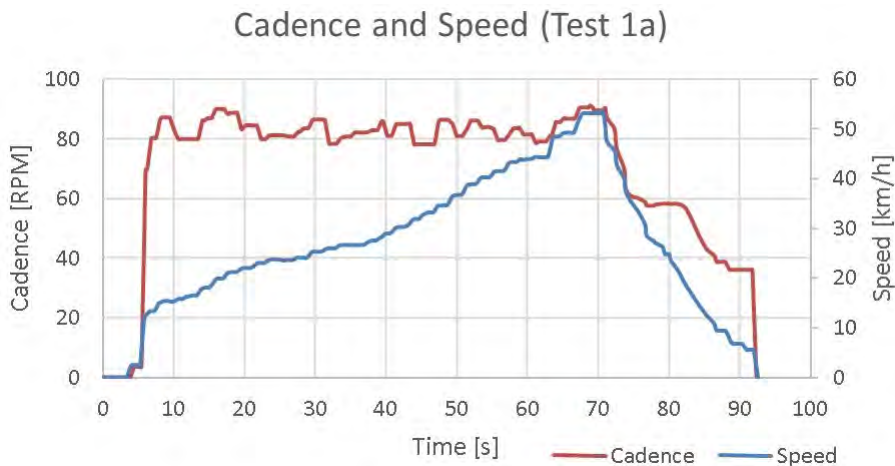


Figure 11.4: Cadence as speed is increased. The average cadence on the interval $10s - 50s$ was $95RPM$. The system is set to have no additional time delay for upshifts, but time delay for downshifts was set to $1s$. This additional time delay for downshifting causes the cadence to drop when speed is decreased.

When the GSS calculates that an upshift is required ($G_C < G_N$), the system sends signal to the ETS to perform the gearshift without an additional delay (other delay than the internal delay in the GSS program). When the GSS calculates that a downshift is required ($G_C > G_N$), each downshift is performed with a $1s$ additional delay, as seen on Figure 11.5. The $1s$ additional time delay setting for downshifts causes the cadence to drop below CAD_{goal} when decelerating.

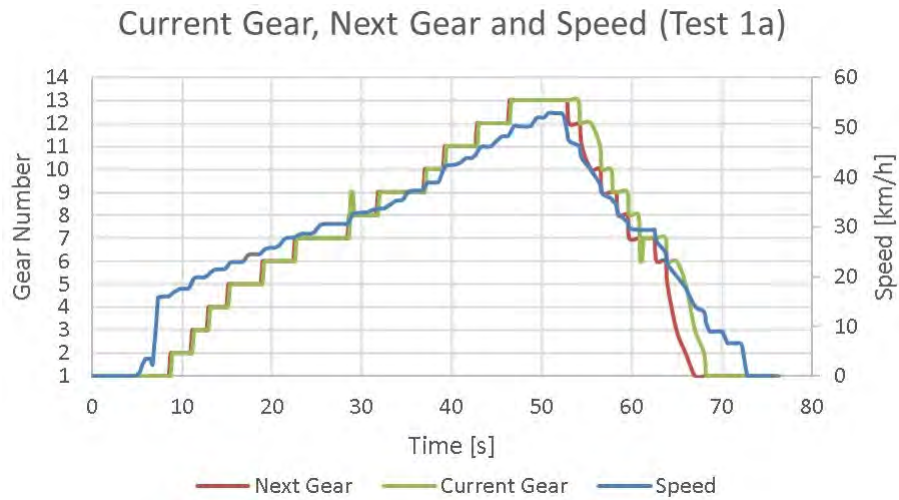


Figure 11.5: Current gear (G_C) of the ETS, and the next appropriate gear to shift in (G_N) is calculated by the GSS. When $G_N > G_C$, the system shifts gear without a delay. The graph shows when accelerating, G_C follows G_N with 0.2s delay, but when downshifting the system the total delay was 1.2s. That can be seen on the graph when decelerating and G_C lags G_N .

To confirm that the gear shifting sequence was working correctly the current gear (G_C), front sprocket (S_F) and rear sprocket (S_R) combinations were analyzed as seen in Figure 11.6.



Figure 11.6: Graph showing G_C and corresponding combinations of S_F and S_R . Shifting between 7th and 8th gear requires 1 shift on the front derailleur and two shifts on the rear derailleur. This causes the system to temporarily go in a sprocket combination equivalent to 9th gear in the shifting sequence. This appears as a spike on the graph at $t \approx 30s$. The same occurs when shifting from 8th down to 7th gear, then the system is temporarily into a sprocket combination equivalent to 6th gear in the shifting sequence. This appears as a negative spike on the graph at $t \approx 60s$ where G_C drops to 6 before reaching 7.

According to the gear shifting sequence discussed in Section 8.3.1, the relationship between G_C and S_R can be expressed as in Equation 11.1

$$G_C = \begin{cases} S_R & | S_R = [1, 7] \\ S_R + 3 & | S_R = [8, 14] \end{cases} \quad (11.1)$$

The current gear G_C and the corresponding sprocket combinations of S_F and S_R are shown in Figure 11.6. The front derailleur only shifted when the GSS shifted between 7th and 8th gear, as expected. When upshifting from 7th to 8th gear, the system increases S_F from 1 to 2 and decreases S_R from 7 to 4. This causes the system to temporarily be in $S_F = 2$ and $S_R = 6$ which corresponds to $G_C = 9$. Similar situation occurs when downshifting from 8th gear to 7th gear, then the system is temporarily in the combination $S_F = 1$ and $S_R = 6$ which corresponds to $G_C = 6$.

Test 1b - Power zone 3

Cadence variations as speed is increased when the system is operating in power zone 3 are shown in Figure 11.7. The CAD_{goal} for power zone 3 was $90RPM$. The results showed that the system kept average cadence of $99RPM$ on the time interval $10s - 40s$, which is $9RPM$ or 10% higher than CAD_{goal} that was determined as optimal. The average cadence was higher than the CAD_{goal} for the same reason as in the previous test discussed in Section 11.3.1. According to the check list in the document in Appendix A.10, the test is passed.

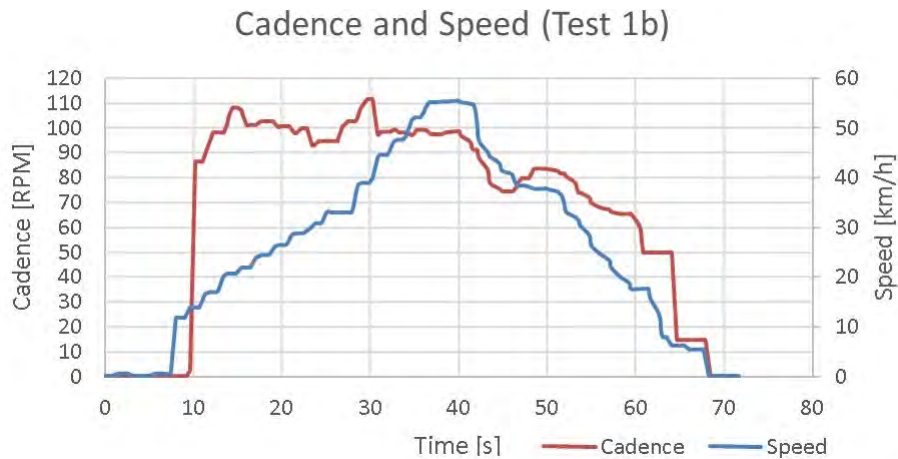


Figure 11.7: Cadence as speed is increased at power zone 3. The average cadence on the interval $10s - 40s$ was $99RPM$. Time delays for downshift are noticeable as cadence drops when decelerating.

Both tests 1a and 1b show that the GSS kept a considerably higher average cadence than the preset goal cadence. The accuracy could be improved with proper tuning of the GSS, either by reducing latency in the system or by tuning the fuzzy sets.

When slowing down, the GSS calculates a new optimal gear but executes the gearshift with 1 s delay as shown in Figure 11.8. To confirm that the system was shifting gears correctly according to the shifting sequence, the current gear (G_C), front sprocket (S_F) and rear sprocket (S_R) combinations shown in Figure 11.9 were compared to Equation 11.1 and the result shows that the GSS was shifting gears correctly, according to the sequence.

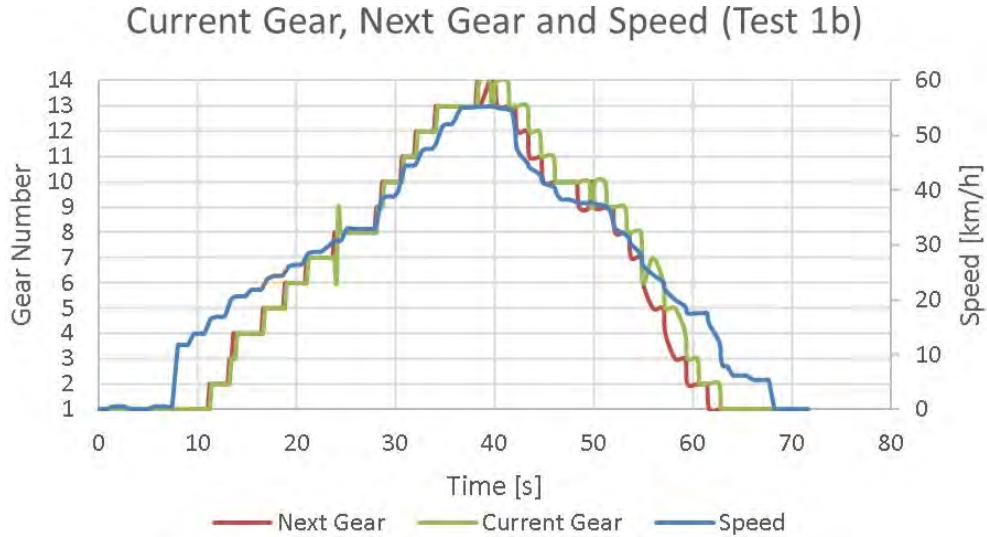


Figure 11.8: Current gear G_C of the ETS and the next appropriate gear to shift in G_N is calculated by the GSS.

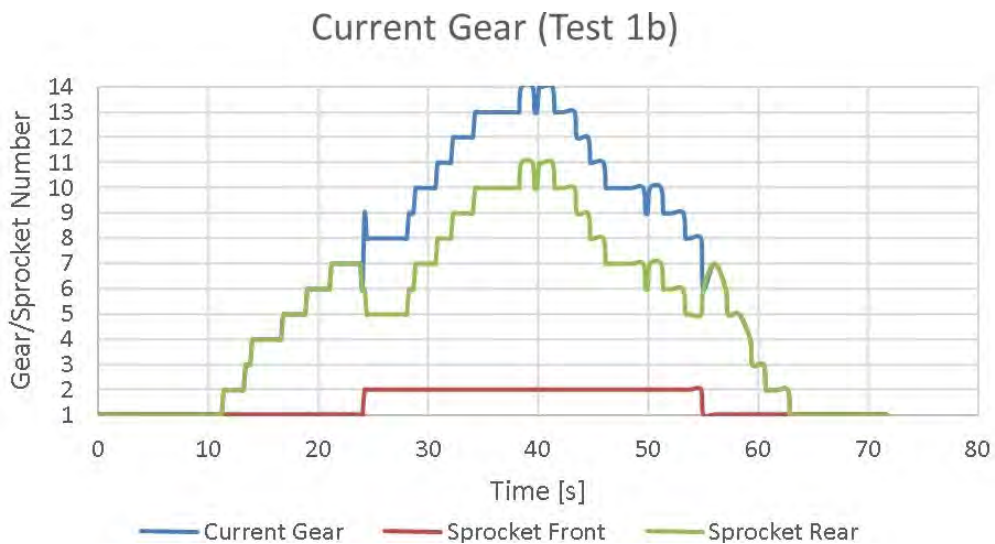


Figure 11.9: Graph showing current gear G_C and what combination of front sprocket S_F and rear sprocket S_R for all 14 gears. As discussed earlier, shifting between 7th and 8th gear causes the spike at $t \approx 25$ s when upshifting and a negative spike at $t \approx 55$ s when downshifting.

11.3.2 Test 2 - Riding out of the saddle

This test showed how the system behaved when making gearshifts when riding out of the saddle. Then the GSS should not allow gearshift on the front derailleur. According to the gear shifting sequence, riding out of the saddle will only affect the gearshifts if shifting between 7th and 8th gear is required. The GSS will enable shifting on the front derailleur when the rider is back in the saddle. Figure 11.10 shows how the riding position affects gear shifting. At $t \approx 21\text{ s}$ the next gear is updated to $G_N = 9$, but the current gear was $G_C = 8$ until the rider sat down at $t \approx 23\text{ s}$. When slowing down while riding out of the saddle the system did not shift down to a lower gear than 8th gear. The system recommended next gear to be $G_N = 7$ at $t \approx 61\text{ s}$, but the system did not take any action until the rider sat down at $t \approx 71\text{ s}$. At that time, next gear had been updated to $G_N = 2$, so when the rider sat down the system shifted multiple times to achieve $G_C = G_N = 2$.

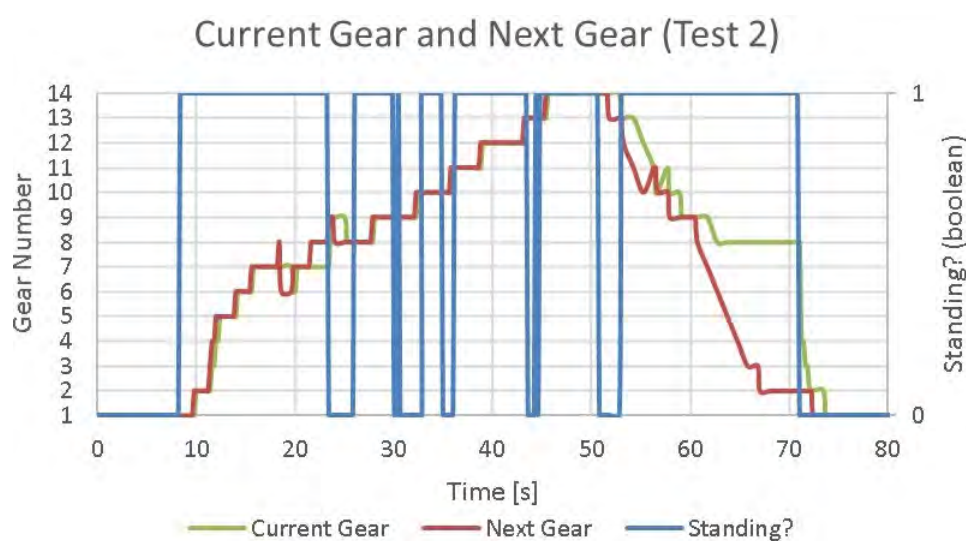


Figure 11.10: The affect of riding position on gear shifts. Riding in standing position did only affect shifting between 7th and 8th gear in the shifting sequence. The disabled front derailleur shifting can be seen at $t \approx 21\text{ s}$ as the GSS did not shift up to 8th gear until rider sat on the saddle. At $t \approx 71\text{ s}$ the rider sits down which enables the system to shift down from 8th gear.

The standing riding position should only affect the gear shifting action when front derailleur shifts are required. Figure 11.11 shows how riding position affected the gear shifting sequence. Front sprockets shifts were not performed unless the rider was in a sitting position. This test confirmed both that the implementation of the FlexiForce sensor was working and that the GSS was working properly when handling the two different riding positions. According to the check list in the document in Appendix A.10, the test is passed.

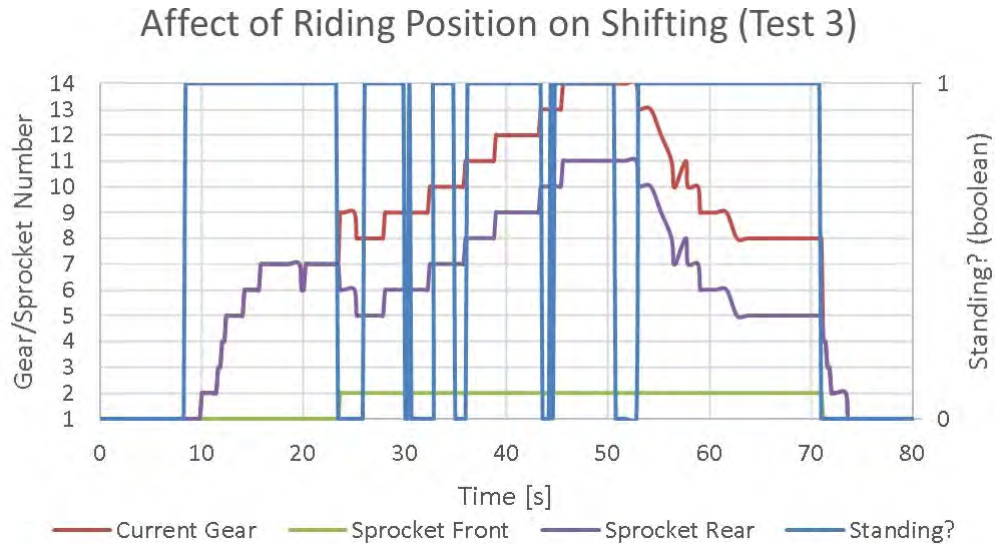


Figure 11.11: The current gear G_C and combinations of front sprocket S_F and rear sprocket S_R for all 14 gears. Riding in a standing position only affected gear shifts when shifting between 8th and 9th gear.

11.3.3 Test 3 - Coasting

The GSS used cadence measurements only for indicating when the rider was not pedaling to disable gearshifts, as gearshifts are not desired unless the crank is rotating. A simple test was performed to confirm that cadence measurement was working for informing the GSS when the cadence was 0RPM . Figure 11.12 shows the results from the test. The results showed an average delay of approximately 3 s in the cadence measurement when compared to the speed measurement. The speed started to drop when pedaling was stopped. The delay in cadence zero tracking caused the GSS to change gears even the rider was not pedaling. According to the check list in the document in Appendix A.10, the test is passed as the GSS does disable gearshifts when the cadence was measured as zero.

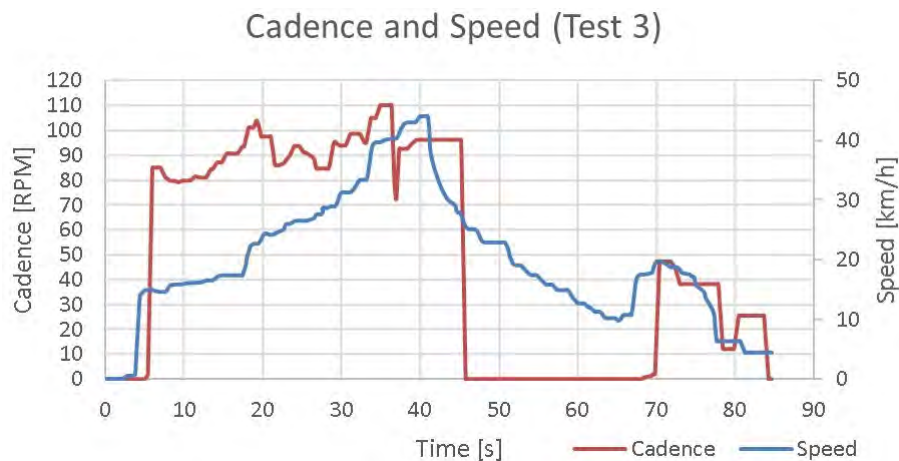


Figure 11.12: Graph showing how speed increased when pedaling and speed decreased after pedaling was stopped. The wheel kept spinning freely after pedaling was stopped because the resistance unit of the stationary trainer was disconnected at $t \approx 41\text{ s}$. The delays in cadence compared to speed measurements are visible on the graph at $t \approx 5\text{ s}$, $t \approx 45\text{ s}$ and $t \approx 70\text{ s}$.

This latency in the cadence zero tracking could cause problem for the system when pedaling is stopped frequently. The system did not execute gearshifts until 3 s after the rider started pedaling again after coasting. This is shown in Figure 11.13.

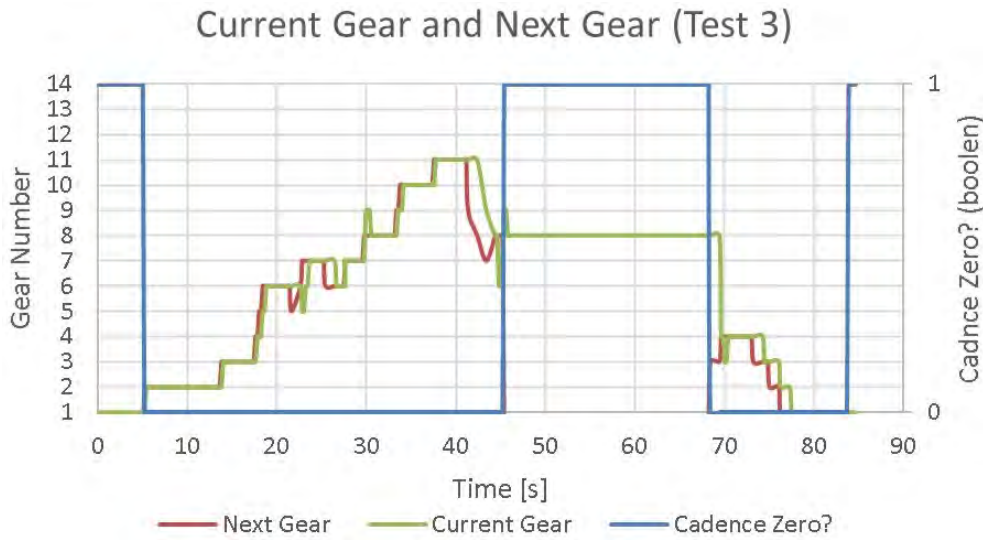


Figure 11.13: The system does not perform gearshifts when $CAD = 0$ and no value for next gear G_N is calculated and G_C remains unchanged until measured $CAD \neq 0$.

Another possible way to indicate when cadence is zero is by power measurement. When not pedaling, the power meter output should be zero according to Equation 9.3. The result from an experiment in Figure 11.14 shows that the power measurement responded to the discontinuation of pedaling 2 s earlier than the cadence sensor measurement did. Due to the unreliable zero tracking of the power measurement, the cadence measurement from cadence sensor was preferred to indicate when not pedaling, despite having a slower zero tracking.

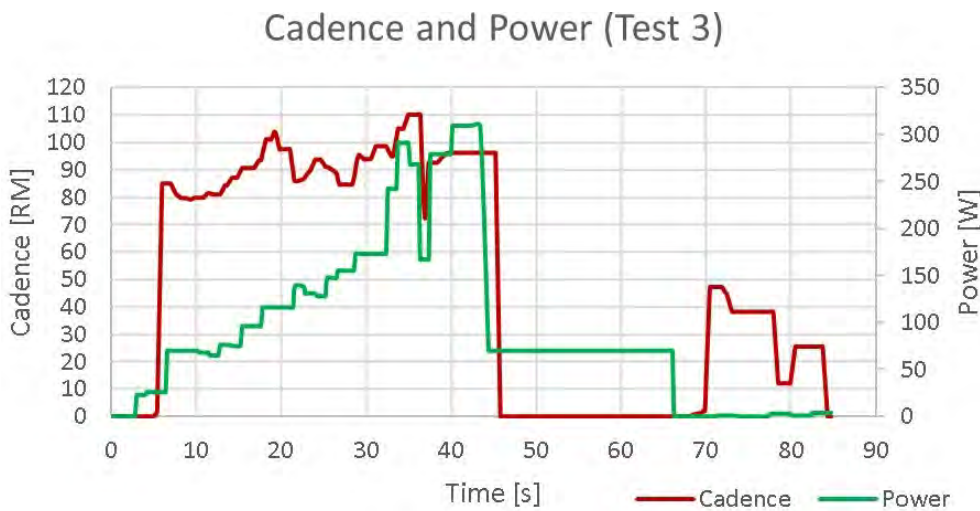


Figure 11.14: Indicating zero cadence from power measurement was not used as zero tracking of power was unreliable, as seen at $t \approx 45$ s. The power measurement was only used for determining which power zone the GSS operated in, to set CAD_{goal} accordingly.

11.3.4 Checklist for prototype test plan

The filled checklist from the prototype test plan document is shown in Table 11.1. The checklist was filled by the author when testing the GSS prototype according to the test plan document, which can be found in Appendix A.10

Table 11.1: A filled checklist for testing of optimal gear selection system prototype.

Test #	Description	Criteria	Passed	Not Passed	Comments
Test 1a	Verify that 80 <i>RPM</i> average cadence is maintained	Power Zone 2 Acceleration (± 10) <i>RPM</i>		X	Average cadence 95 <i>RPM</i>
-					
Test 1b	Verify that 90 <i>RPM</i> average cadence is maintained	Power Zone 3 Acceleration (± 10) <i>RPM</i>	✓		Average cadence 99 <i>RPM</i>
-					
Test 2	Verify that front derailleur shifts are disabled while riding out of the saddle.	Shift between 7th and 8th gear	✓		Successful
-					
Test 3	Verify that all gearshifts are disabled while not pedaling		✓		A 3 s time delay of zero tracking for cadence measurement can cause the GSS to shift gears after rider stops pedalling
-					

Chapter 12

Future work

In this chapter, possible future extensions to the current prototype optimal gear selection system are discussed. Future developments of the GSS prototype could include hardware/software platform transformations and improvements to the optimal gear selection methods. Possible platform solutions will be discussed, as well as the future development of ETS. Assumptions will be made regarding what features could possibly appear in future versions of ETS, which is important as the future of standalone GSS systems highly depends on how ETS will evolve.

12.1 LabVIEW software evolution

This project has laid a foundation for a more advanced gear selection system that could combine more AI technologies to make a hybrid system. Adding neural networks to make a neuro-fuzzy system could make the system intelligent and enable it to adopt system parameters automatically by learning (Negnevitsky, 2011, pp. 268–269).

Important consideration regarding evolution of the software is which platform should be used. Current software runs on LabVIEW on Windows, which is a suitable platform for prototyping. Currently the ANT+ toolkit does not support real time (RT) targets like the NI myRIO. According to the author of the ANT+ toolkit, iNU Solutions have planned to add a support for NI RT targets in near future.¹

An exciting aspect of software improvements is the possibility of measuring same number of measurands by using fewer sensors. That can be achieved by implementation of a soft sensor (Kadlec, Gabrys & Strandt, 2009). The prototype system uses a force sensor for detecting riding position, but a possible solution is a soft sensor that uses data from the power sensor to determine whether the cyclist is riding in a standing or sitting position. Certain types of power sensors, including the Garmin Vector 2, broadcast the additional Data Page 19 that includes data that could be useful for making such soft sensor. This additional data page contains the parameters *torque effectiveness* and *pedal smoothness* for each pedal. The additional Data Page 19 is shown in Figure A.21. Soft sensor solution for detecting ride position has already been implemented in the embedded software on Garmin edge cycling computers (*Edge 520*, 2016). Torque effectiveness and pedal smoothness will not be discussed in this report, information on these parameters can be found in Dynastream’s ANT+ protocol documents (*ANT+ Device Profile - Bicycle Power. Revision 4.2.*, 2015). Future development of the system software could also possible include more approaches to optimal gear selection. Instead of maintaining goal cadence, the system could be designed to select gear to help the rider maintaining a specific goal power output (Maker, 2014).

¹Mather, D., personal communication, May 29, 2016

12.2 Hardware platforms

The current optimal gear selection system prototype could be extended so it could run on compact, embedded device that could be mounted on the bike, as shown on Figure 12.1. A suitable device for this application could be the NI myRio, which has decent processing capabilities, analog and digital ports and USB port for connecting USB ANT Stick for wireless sensor communications. The system could even take advantage of the myRIO FPGA (Field-programmable gate array) (NI myRIO, 2016). The NI myRio is capable of performing data acquisition from the analog FlexiForce sensor and send digital signals to ETS connection circuit to perform gear-shifts. The current obstacle for running the GSS software on the NI myRIO is that the ANT+ toolkit currently does not support RT targets.

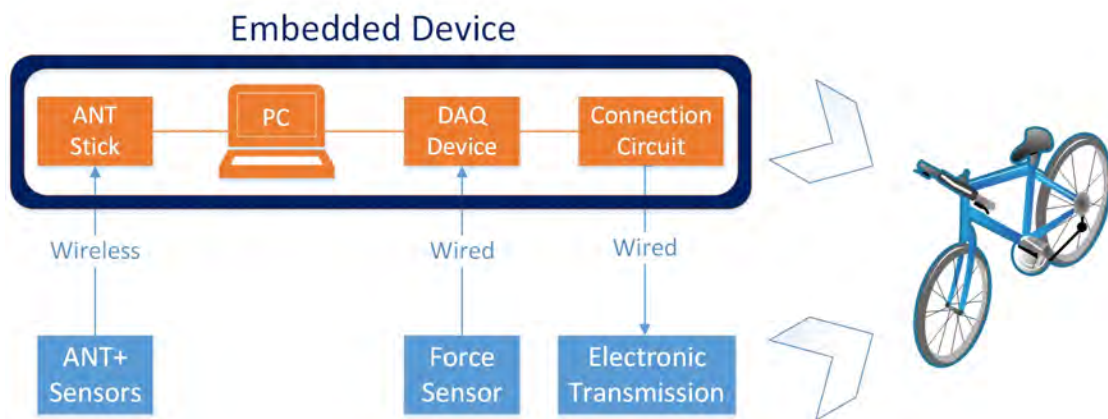


Figure 12.1: Overview of how an embedded device could replace the hardware that the current prototype system uses for sensor data acquisition, data processing and connection to the ETS. A compact embedded device could easily be mounted on a bicycle.

12.2.1 Gear selection system on smartphone

The next big leap bringing the prototype system closer to a product prototype would be moving it from a hardware based solution towards being a software orientated solution that runs on a hardware platform the user has already adopted. The transformation to hardware platforms other than platforms by NI, would require the GSS being programmed in other programming language than LabVIEW.

An example how an ATS could be structured if a smartphone or a cycling computer would be used as a platform for running the GSS software is shown in Figure 12.2. At this point, all the sensors would have to be wireless. For the current prototype GSS, this means that the seat mounted force sensor either needs to be updated to a wireless force sensor or an alternative method for detecting riding position has to be used.

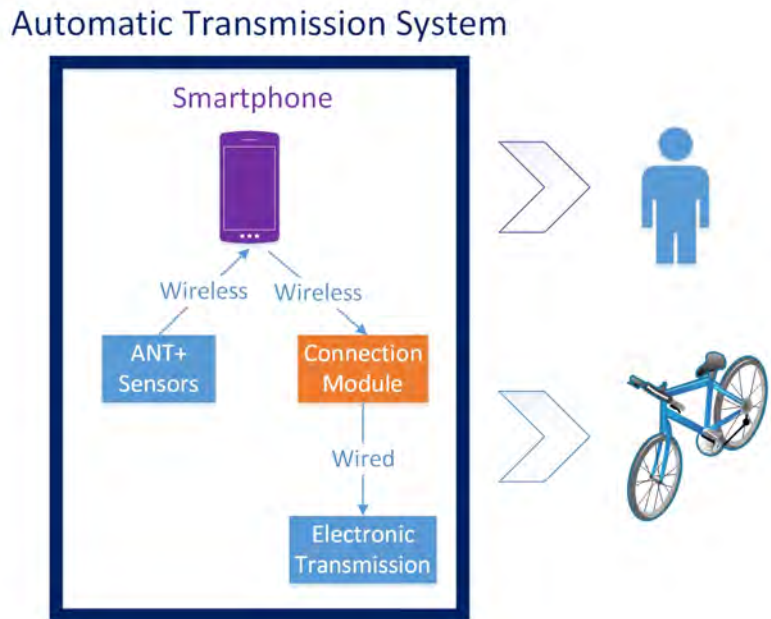


Figure 12.2: Overview of how the system could be made into a more consumer friendly solution that is more software orientated. The only additional hardware module required to turn an ETS into an ATS is a custom made wireless interface module, shown in orange. Only the smartphone would be carried by the user.

The only additional system module left that would remain hardware based is the ETS connection module. This could be a small embedded device that physically connects to the ETS and receives wireless signal from a smartphone, possibly by using encrypted private ANT communications. Figure 12.3 shows how the smartphone controlled system structure if the ETS would allow wireless connection for shifting gears, eliminating the need of a dedicated hardware module for communications.

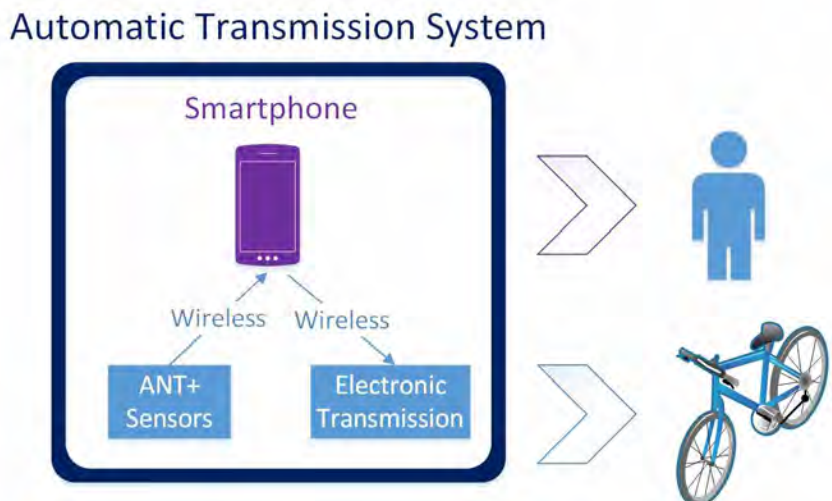


Figure 12.3: Overview of a smartphone controlled ATS, a solution that is completely software based. The GSS software runs on the smartphone. Here it is assumed that the ETS enables wireless connection the smartphone to perform gearshifts based on commands from the GSS.

In this way, users that already own an ETS and ANT+ bike sensors could upgrade their ETS to function as an ATS only by purchasing the GSS app on their smartphone. Then the GSS would be a completely software based solution, given that a cycling computer or a smartphone is required for running the software. Smartphones have become ubiquitous in recent years and riders often carry one while riding their bike. In general, smartphones have high computation power capabilities and support wireless sensor communications. That is why a smartphone could be an ideal platform for running the optimal gear selection software. Most smartphones operate on either Android or IOS, so making a software for these two platforms would be enough to reach the majority of smartphone users (*IDC: Smartphone OS Market Share, 2015*).

12.2.2 Gear selection system on cycling computer

Cycling computers have become an essential accessory for competitive cyclists and with increased processing power of modern cycling computers, the potential to run a gear selection software is there. The problem is that there are many vendors of cycling computers that use their own embedded software. That makes it difficult to create a software that is compatible with multiple brands of cycling computers. Designing the GSS an entirely software based solution running on a cycling computer is possible, if future versions of ETS systems will offer access for wireless control as shown in Figure 12.4. Using a cycling computer instead of a smartphone has an advantage: the cycling computer is located in front of the rider at all times, where the rider can make adjustments to the GSS if necessary.

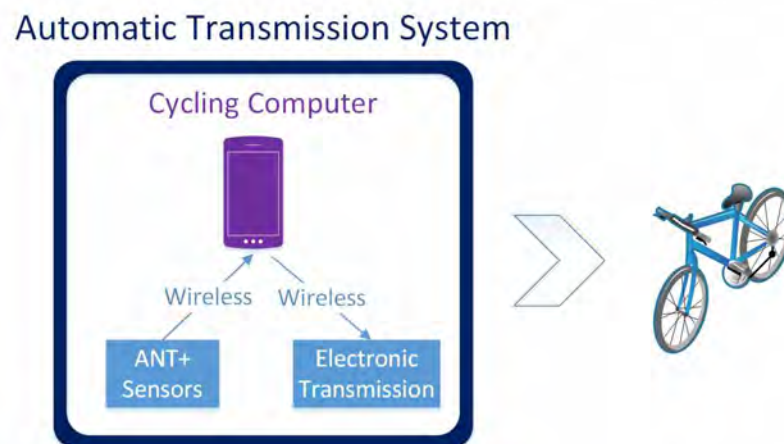


Figure 12.4: Overview of a cycling computer controlled ATS, where the GSS is a completely software based solution. It is assumed that the cycling computer will have wireless access to the ETS to perform gearshifts.

12.2.3 Gear selection system integrated in electronic transmission

The approaches to future system design discussed in this section, rely on an open access to the ETS control. Currently this access is restricted by the vendors of the ETS, giving them the ultimate advantage for developing a fully integrated ATS: a system where a GSS runs transparently within an ETS (BikeRadar, 2014). Such system structure is shown in Figure 12.5. The cycling computer would then only be necessary for monitoring purposes and as input panel for making adjustments to the ATS if needed. This design of an ATS does not support the use of third party GSS as the software would be embedded within the ETS.

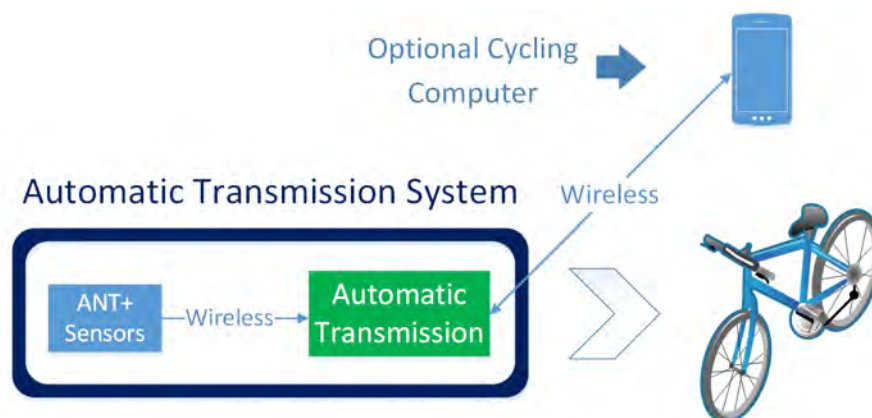


Figure 12.5: Overview of the ultimate integration solution of GSS that runs within ETS internal computer, making the GSS completely invisible to the user. The cycling computer is not part of the ATS, but would be used for monitoring and make configurations to the ATS.

12.3 The future of automatic transmission systems

The technology to make an optimal bicycle gear selection system that is entirely a software based, is already out there. In late 2015, SRAM Corporation introduced an ETS called SRAM RED eTAP. The eTAP is the first commercial ETS that uses wireless communications between transmission system modules. The shifting levers operated by the user communicate wirelessly to the transmission computer. Encrypted communications with a custom protocol are used to prevent the system being hacked (Rossiter, 2015). This restriction makes wireless control of the ETS by third party GSS dependent on approval from the manufacturer of the ETS.

It is logical to conclude that the future of GSS made by third parties is in the hands of the vendors of the ETS. An example how vendors are restricting access to data that helps developing automatic gear selection systems is the private ANT communication used by the Shimano Wireless unit. The unit broadcasts current gear status of the Di2 ETS and can be displayed on the Garmin Cycling computer. Without access to current gear status data, the GSS has to keep track of the current gear by counting every shift command sent to the ETS. That means the GSS requires an initial calibration of gear status every time before using the system. Keeping track of current gear by counting shift commands sent to the ETS is also not as reliable as having the correct status signal from the ETS available at all times.

Chapter 13

Conclusions

This report described the design of a gear selection system (GSS) and integration with Shimano Ultegra Di2 6870 electronic transmission system (ETS). A prototype GSS was built and connected to the ETS through a custom made electronic circuit. The outcome of the GSS and ETS integration was a fully automatic transmission system (ATS) that selected a gear based on sensor data from four different sensors. The GSS acquired data from three ANT+ sensors for measuring speed, pedaling rate (cadence) and pedaling power. The GSS also acquired data from an analog connected force sensor that was mounted on the bicycle saddle for indicating whether the cyclist was riding in or out of the saddle.

The gear selection software was programmed in LabVIEW and the prototype GSS was operated on a PC. The GSS acquired data from the wireless ANT+ sensors using ANT+ toolkit (*ANT+ Device Drivers*, 2014). To enable communications to the ETS, a custom made circuitry was built. The circuitry of the hand operated shifting levers of the ETS were modified and transistors used to replace the mechanically operated button switches. This modification allowed the GSS to electronically perform gearshifts on the ETS using a DAQ device.

As researches have shown that cadence has a large effect on pedaling efficiency, the optimal gear selection was performed with respect to optimal cadence (Gotshall et al., 1996). The GSS was designed to shift gears to maintain an average cadence that corresponds to the optimal cadence. The GSS was developed as a knowledge based system in a systematic manner, and the prototype GSS fulfilled the initial requirements defined in Section 2.1. A test plan and a test plan document was created, and can be found in Appendix A.10. The system was installed on a demo bike with ETS and tested on a stationary indoor trainer according to the test plan.

Results showed that the prototype GSS was capable of performing gearshifts to maintain a consistent cadence. The GSS kept an average cadence that was higher than the preset goal cadence (CAD_{goal}). The GSS was designed to keep different CAD_{goal} for different pedaling output power, which was classified into zones. The GSS was designed to maintain an average cadence of $80RPM$ for zone 2 and an average cadence of $90RPM$ for zone 3. Results from system test showed that average cadence for zone 2 was $95RPM$, which is approximately 19% above the corresponding optimal cadence. For power zone 3 the average cadence was $99RPM$, which is 10% above the corresponding optimal cadence. These results indicate that the GSS is capable of maintaining a consistent cadence, but would require proper tuning to maintain a more accurate goal cadence.

The evolution of wireless cycling sensors and ETS in terms of capabilities and price has raised the potential for GSS. Future developments for the prototype GSS can include more advanced gear shifting algorithms, more compact hardware platforms and implementation of soft sensors.

Appendix A

Appendix

1. Appendix A.1: Master's thesis task description
2. Appendix A.2: Project Abstract
3. Appendix A.3: Gear ratios
4. Appendix A.4: Speed and cadence tables
5. Appendix A.5: Fuzzy sets
6. Appendix A.6: Fuzzy system output
7. Appendix A.7: Additional panels on graphical user interface
8. Appendix A.8: LabVIEW syntax
9. Appendix A.9: ANT+ device profile data pages
10. Appendix A.10: Test plan document



Campus Porsgrunn/Faculty of Technology
Department of Electrical Engineering, IT and Cybernetics

FMH606 Master's Thesis

Title: „Optimal Bicycle Gear Selection using Multisensor Data Fusion”

UCSN supervisor: Saba Mylvaganam

Task background:

Cycling computers and dedicated sensors for athlete diagnostics include speedometers, power-meters, cadence sensors and pulse sensors. These devices are widespread in present day cycling sports and many commercial vendors are catering to the market. Electronic gear selecting systems that utilize data from these sensors can be implemented and in combination with electronic driven shifters, a fully automatic gear shifting systems can be achieved.

Electronic gear shifters for bicycles have become increasingly popular in recent years. The most known brand is Shimano Di2. These systems are manually operated from the handlebars, in the same manner as in the case of conventional shifters but instead of wire tension based derailleur movements, motors are used.

The current additional gear selection systems commercially available use speed, power, cadence (pedaling rate) and heart rate as input variables to estimate optimal gearing selection at each time. This new approach has given promising results, but using these three input variables makes the system limited, when compared to the athlete as the operator, who is more versatile in optimizing gearing himself/herself as he/she is aware of his/her physical conditions and the environment.

With growing number of commercially available sensors made for communicating with on-board bike computers, new possibilities are opening for both recreational and competitive cyclists. Data fusion can give new/additional useful information. For example, soft sensor has been implemented for certain power-meters where the output data is used to determine whether athlete is standing or sitting on the bike, which is important for how gear shifts are executed.

Input Variables from sensor devices

- Speed (Speed Sensor / Tacx Stationary trainer)
- Power (Garmin Vector power-meters / Tacx Stationary trainer)
- Crank Cadence (Garmin Cadence sensor / Tacx Stationary trainer)
- Athlete's heart rate (Garmin chest strap heart rate sensor)
- Current gear (Shimano Di2 D-Fly Transmitter)

Variables achieved by soft-sensor

- Athlete's Riding position - Standing/Sitting (Garmin Vector 2 Power-meters)

Output variables:

Final gear ratio is achieved by combination of front and rear gears. To achieve the optimal gearing, the system takes action by upshifting/downshifting.

- Front derailleur: Shift up, shift down
- Rear derailleur: Shift up, shift down

Task description:

1. Literature review on the equipment used in the project.
2. Literature review on protocols used within the system (ANT/ANT+).
3. Define the functionalities and appropriate behavior of the system
4. Extract live sensor data from sensors to PC and make the data accessible
5. Develop program that includes shifting-control algorithms that use multiple variables as input (possibly using Fuzzy logic or other AI-techniques)
6. Use DAQ device for actuate Shimano electronic shifters to perform gear-shifts
7. Build a functional prototype system.
8. Make a test plan for the system.
9. Test the system according to the test plan.
10. Submit a project report following the guidelines of TUC with the necessary codes well documented for testing and assessment.

Outcome:

An automated gear selection system that uses multiple sensor variables to AI-assisted optimization in the selection of right gearing matched to the pedaling power, speed and condition of the cyclist. The prototype system software will run on PC and the bicycle will be operated on a stationary trainer connected to virtual reality cycling simulator.

Student category:

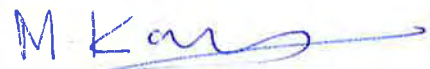
SCE student with prior work experience in mechanical design, PLC programming. Also have worked with various types of load cell sensors.

Practical arrangements:

Some hardware and software for the project are available in the sensor lab at UCSN. Student will provide the bicycle that the system will operate on. UCSN, at the departmental level will provide necessary support and try to establish necessary contact to external collaborators. UCSN will facilitate the student to acquire some of the necessary modules for the present study.

Signatures:

Student (date and signature):  01.02.2016

Supervisor (date and signature):  01.02.2016



MASTER'S THESIS, COURSE CODE FMH606

Student: Víðir Bjarkason

Thesis title: Optimal Bicycle Gear Selection Using Multi-Sensor Data Fusion

Signature:

Number of pages: 130

Keywords: ANT+, Cadence, Data acquisition, Electronic transmission system, Fuzzy logic, Gear selection system, LabVIEW, Pedaling power.

Supervisor: Kanagasabapath Mylvaganam Sign.:

2nd supervisor: Håkon Viumdal Sign.:

Censor: Sign.:

Availability: Open

Archive approval (supervisor signature): Sign.: **Date :**

Abstract:

The latest technology in electronic transmission systems (ETS) and wireless sensors for bicycles bring a high potential to automate gear shifting on bicycles. An optimal gear selection system (GSS) for road bicycles with an ETS was developed. The system interacted with a commercial Shimano Di2 6870 ETS, resulting in a fully automatic transmission system (ATS) that selected an optimal gear for the user. The interconnection between the ETS and GSS required a customization on the ETS hardware. Mechanical button switches on ETS were replaced with a transistor based circuit to enable electronic control of gear shifts. Optimal gearing and system behavior was determined from knowledge acquisition from both practical and theoretical aspects. Knowledge-based system was developed for optimal gear selection, using fuzzy logic control. Sensors were used to measure speed, pedaling rate (cadence), pedaling power and riding position. By fusion of the sensor data and the acquired knowledge, the GSS selected an optimal gear that maintained the cadence within a predefined optimal range for a given situation. NI LabVIEW was used for development of the GSS and ANT+ toolkit for LabVIEW was used for data acquisition from ANT+ bike sensors.

The GSS prototype was installed on a demo bike equipped with an ETS and tested on a stationary indoor trainer, according to a test plan document. The system performance was tested in terms of maintaining cadence at a preset goal cadence. Two separate tests were done for goal cadences $CAD_{goal} = 80RPM$ and $CAD_{goal} = 90RPM$. The results showed that the system performed well in maintaining a consistent cadence, but the average cadence the system maintained was higher than the set goal cadence. The affect of each individual input variable on the system was tested and results confirmed that the GSS performed according to the predefined requirements.

University College of Southeast Norway accepts no responsibility for results and conclusions presented in this report.

A.3 Gear ratios

Table A.1: Gear ratios for 53/39T front sprockets and 11-28T rear cassette. Sprocket combinations are sorted by gear ratio in an ascending order. Gear sprocket combinations eliminated from the gear shifting sequence to avoid cross-chaining are marked in red. Combinations eliminated to avoid unnecessary front derailleur shifts are marked in blue color. The final sequence consists of 14 gear combinations.

Ascending Ratio Order	Front Sprocket no.	Rear Sprocket no.	Ratio
1	1	1	1.39
2	1	2	1.56
3	1	3	1.70
4	1	4	1.86
5	2	1	1.89
6	1	5	2.05
7	2	2	2.12
8	1	6	2.29
9	2	3	2.30
10	2	4	2.52
11	1	7	2.60
12	1	8	2.79
13	2	5	2.79
14	1	9	3.00
15	2	6	3.12
16	1	10	3.25
17	2	7	3.53
18	1	11	3.55
19	2	8	3.79
20	2	9	4.08
21	2	10	4.42
22	2	11	4.82

A.4 Speed and cadence tables

Table A.2: For $CAD_{goal} = 90RPM$. Gear selection based on cadence. Gear is changed when cadence reach either low limit of $80RPM$ or high limit of $100RPM$. Fuzzy sets for gear selection were created according to this data.

CAD (LOW) 80 RPM	Speed [km/h]	Cadence at same speed after downshift	CAD (HIGH) 100 RPM	Speed [km/h]	Cadence at same speed after upshift
2 to 1	15.8	90	1 to 2	17.6	89
3 to 2	17.2	87	2 to 3	19.8	92
4 to 3	18.9	88	3 to 4	21.5	91
5 to 4	20.8	88	4 to 5	23.6	91
6 to 5	23.2	89	5 to 6	26.0	90
7 to 6	26.4	91	6 to 7	29.0	88
8 to 7	28.3	86	7 to 8	32.9	93
9 to 8	31.6	89	8 to 9	35.3	89
10 to 9	35.8	91	9 to 10	39.5	88
11 to 10	38.4	86	10 to 11	44.7	93
12 to 11	41.3	86	11 to 12	48.0	93
13 to 12	44.8	87	12 to 13	51.7	92
14 to 13	48.8	87	13 to 14	56.0	92

Table A.3: For $CAD_{goal} = 90RPM$ a cadence float of $\pm 10RPM$ is allowed. Lower cadence limit for each gear was defined at $80RPM$ and the upper limit at $100RPM$.

CAD [RPM] Gear no. in Sequence	80 [RPM] (LOW) Speed [km/h]	100 [RPM] (HIGH) Speed [km/h]
1	0.0	17.6
2	15.8	19.8
3	17.2	21.5
4	18.9	23.6
5	20.8	26.0
6	23.2	29.0
7	26.4	32.9
8	28.3	35.3
9	31.6	39.5
10	35.8	44.7
11	38.4	48.0
12	41.3	51.7
13	44.8	56.0
14	48.8	-

A.5 Fuzzy sets

Input membership functions

Speed

Membership function	Shape	Points
One	Triangle	0 ; 8.8 ; 17.6
Two	Triangle	15.8 ; 17.8 ; 19.8
Three	Triangle	17.2 ; 19.35 ; 21.5
Four	Triangle	18.9 ; 21.25 ; 23.6
Five	Triangle	20.8 ; 23.4 ; 26
Six	Triangle	23.2 ; 26.1 ; 29
Seven	Triangle	26.4 ; 29.65 ; 32.9
Eight	Triangle	28.3 ; 31.8 ; 35.3
Nine	Triangle	31.6 ; 35.55 ; 39.5
Ten	Triangle	35.8 ; 40.25 ; 44.7
Eleven	Triangle	38.4 ; 43.2 ; 48
Twelve	Triangle	41.3 ; 46.5 ; 51.7
Thirteen	Triangle	44.8 ; 50.4 ; 56
Fourteen	Triangle	48.8 ; 100 ; 100

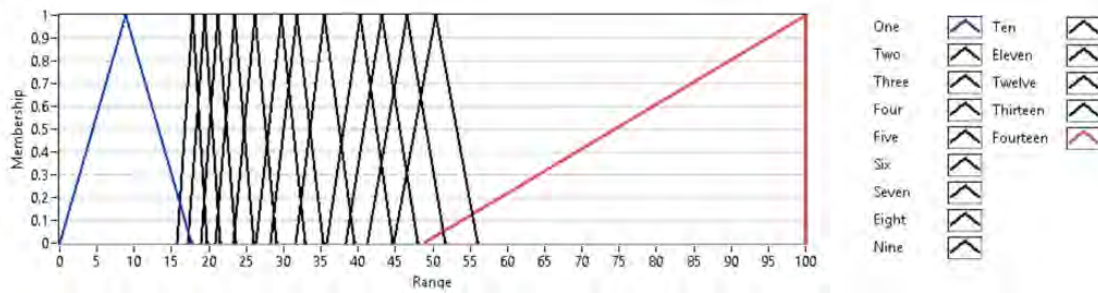


Figure A.1: Fuzzy set for gear selection based on speed for $CAD_{goal} = 90RPM$.

Input membership functions

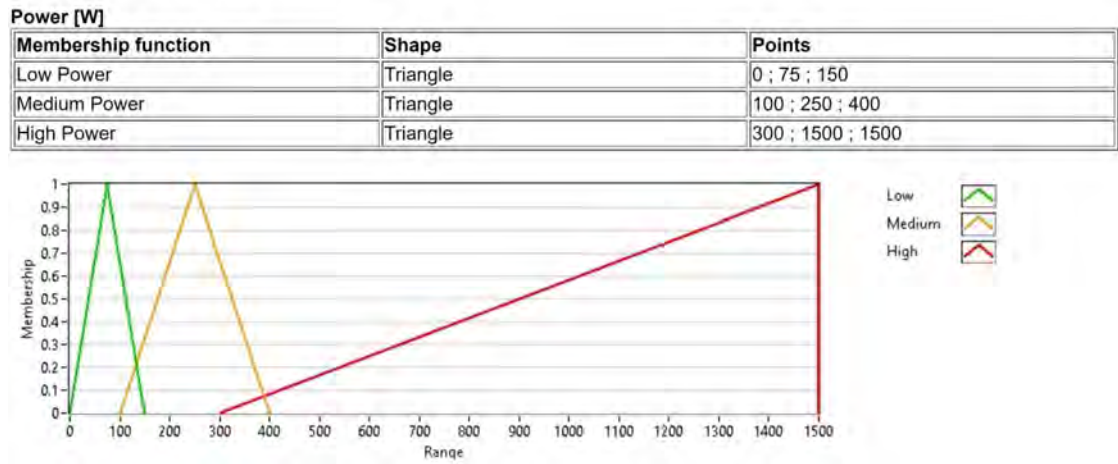


Figure A.2: Fuzzy rules for the classification of power into 3 power zones.

Output membership functions

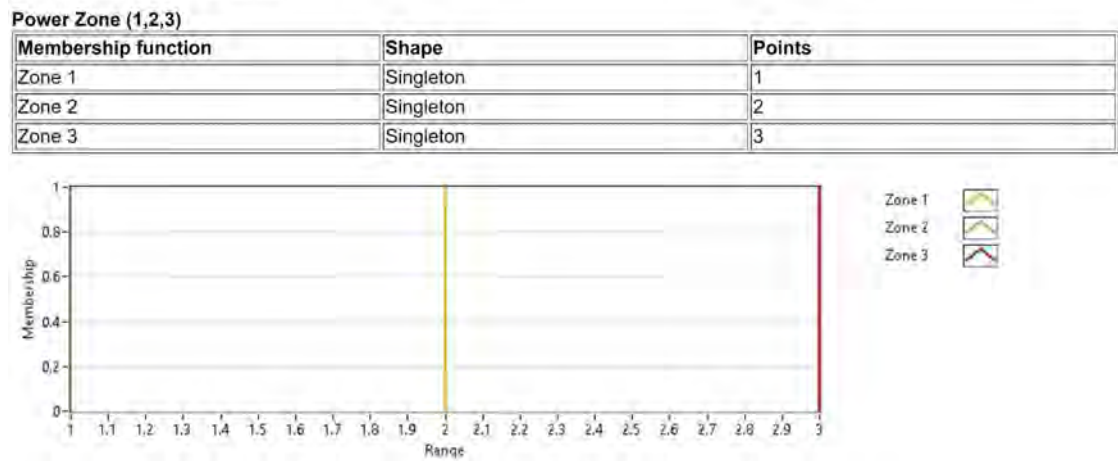


Figure A.3: Fuzzy rules for the classification of power into 3 power zones. Singletons represent the 3 power zones.

A.6 Fuzzy system crisp output

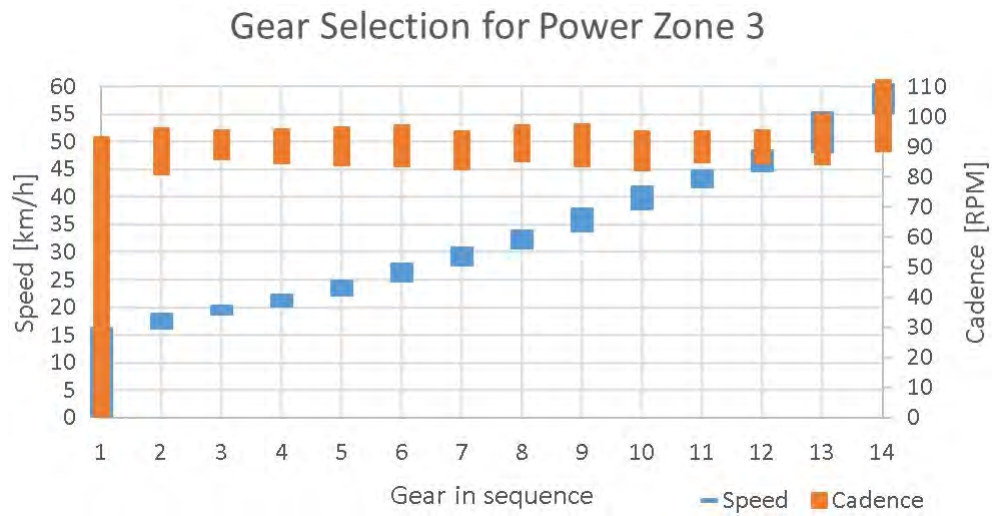


Figure A.4: Crisp gear outputs for each speed for $CAD_{goal} = 90RPM$. Average cadence for 2nd to 13th gear was approximately $90RPM$.

A.7 Additional panels on graphical user interface

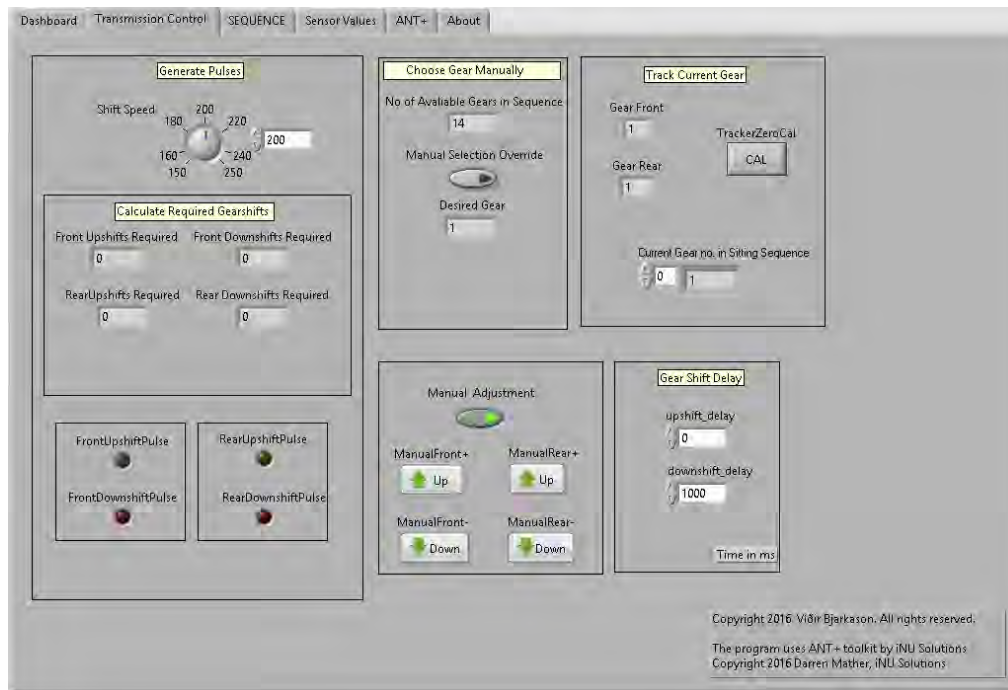


Figure A.5: The transmission control panel on the GUI. Gear shift speed and time delays can be adjusted. Manual control of ETS for initial calibration to 1st gear. In the left corner there are indicators for indicating when shift signals are sent to the DAQ. The system tracks current gear status by counting pulses sent to the DAQ device.

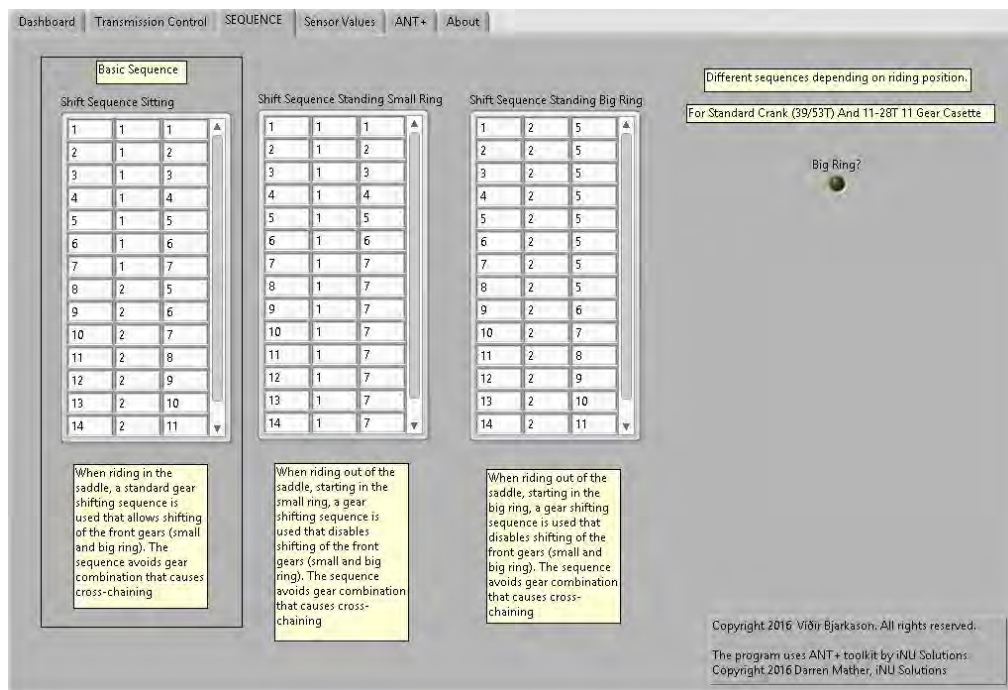


Figure A.6: The sequence panel on the GUI. Input of gear shifting sequence. Default values are set according to sequences in knowledge base.

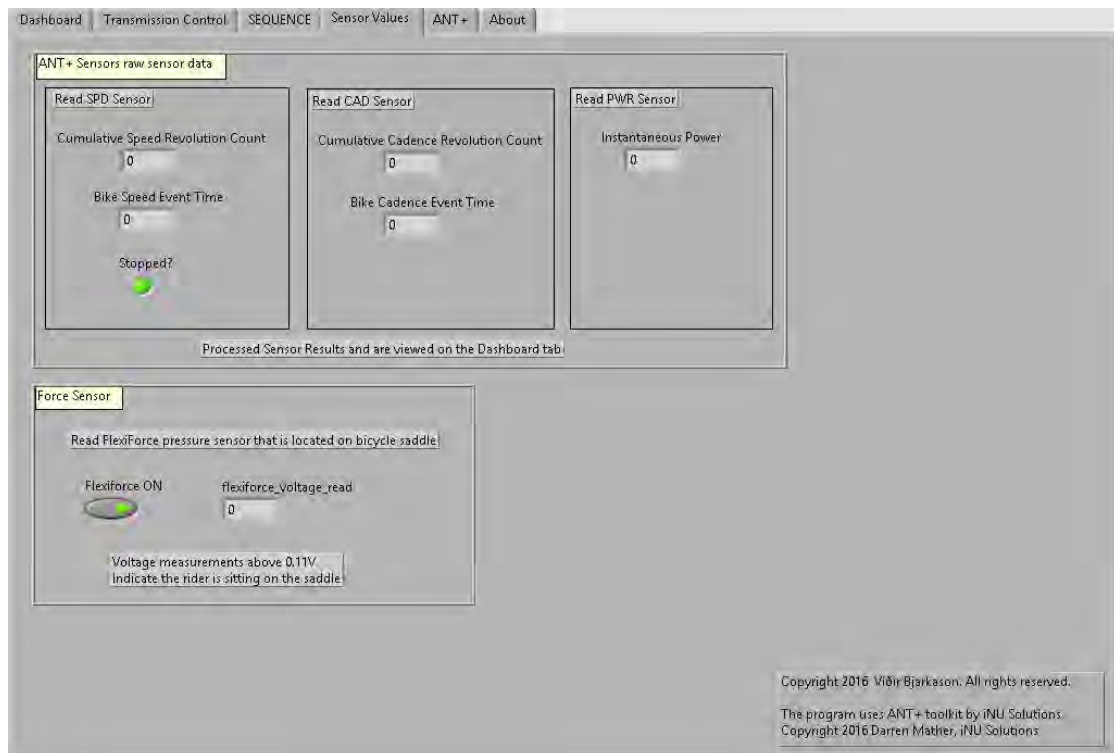


Figure A.7: The sensor values panel on the GUI displaying the raw sensor values from speed, cadence, power and FlexiForce sensors.

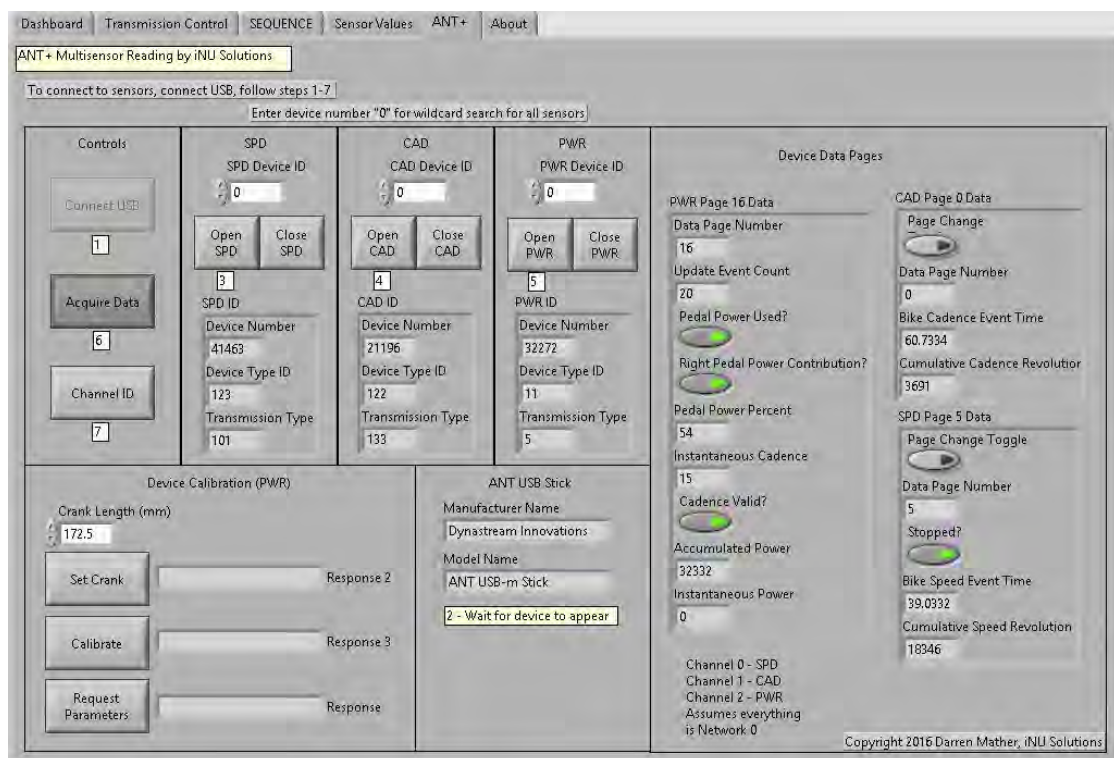


Figure A.8: The ANT+ panel by Darren Mather, iNU Solutions (*ANT+ Device Drivers*, 2014). Allows connection speed, cadence and power sensor via the ANT USB-m stick.

A.8 LabVIEW syntax

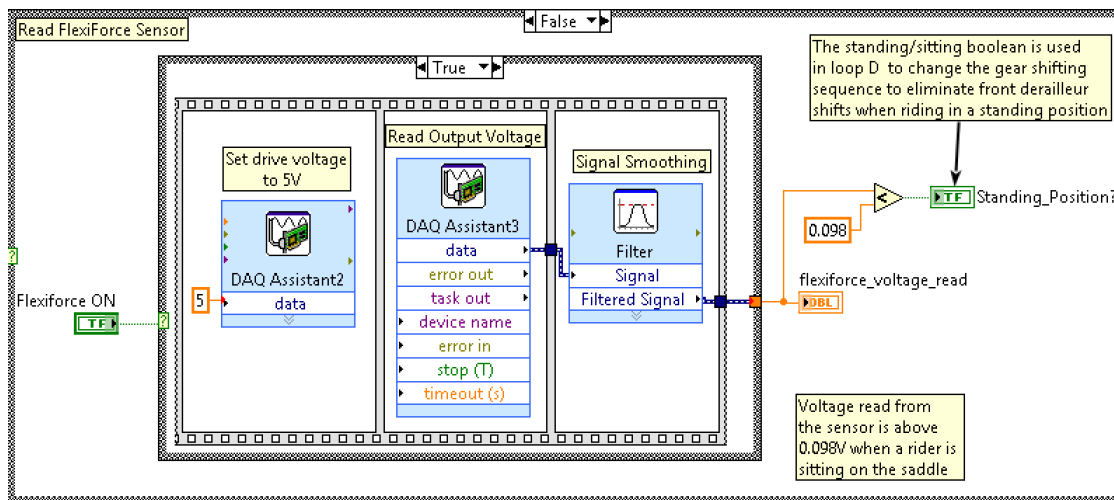


Figure A.9: Driving the FlexiForce sensor using DAQ Assistant and 5 V drive voltage. Filter was set to signal smoothing and half-width of moving average was set to 5. Sensor voltage readings above 0.098 V indicate the rider is sitting on the saddle. For the "False" case of the inner case structure, the sensor drive voltage is set to zero. Location: Loop C.

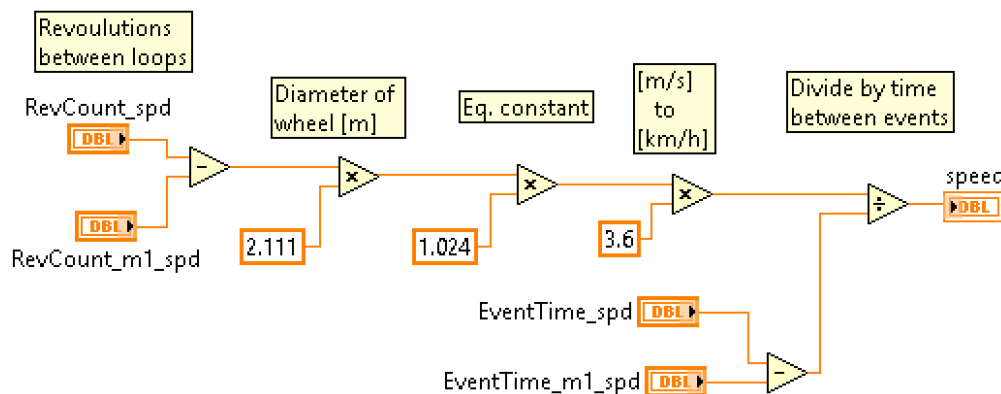


Figure A.10: Calculate SPD according to the ANT+ SPD Profile (ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014). Note that the "m1" indicates value from previous loop run. Location: Loop C.

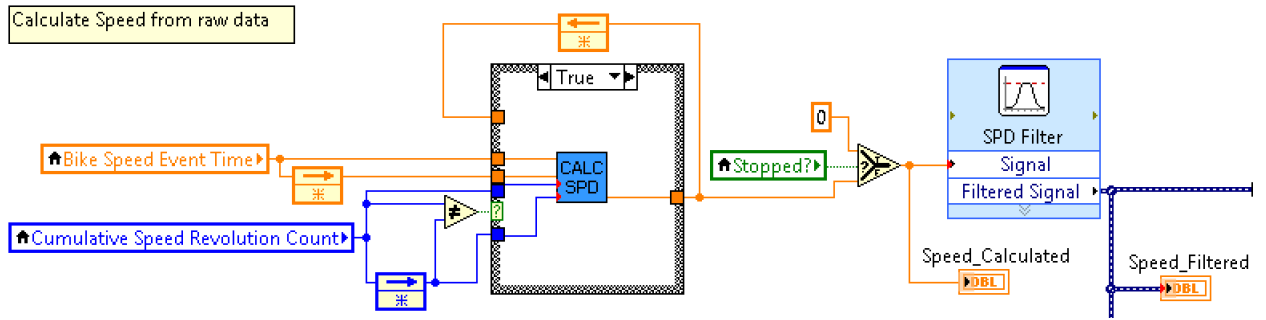


Figure A.11: Implementation of speed calculation. The "CALC SPD" sub-VI gets event time and revolution count values from last loop run and the previous loop run values from feedback nodes. A new speed value is only calculated when the speed revolution counter has increased from previous loop run (True case). When the counter has not increased between runs (False Case), speed value is not updated. When the stop indicator boolean is on, then SPD is set to zero. Location: Loop C

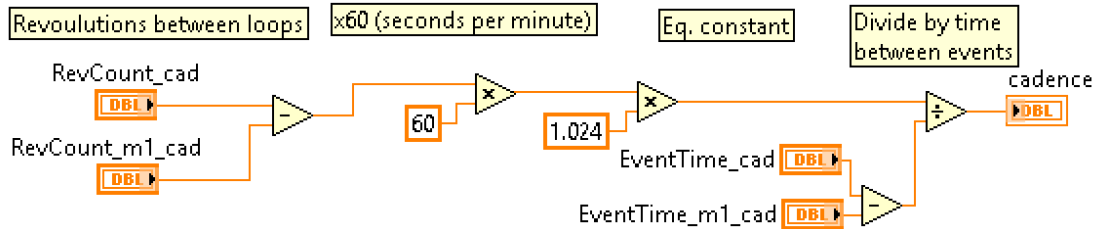


Figure A.12: Calculate CAD according to the ANT+ SPD Profile (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014*). Note that the "m1" indicates value from previous loop run. Location: Loop C.

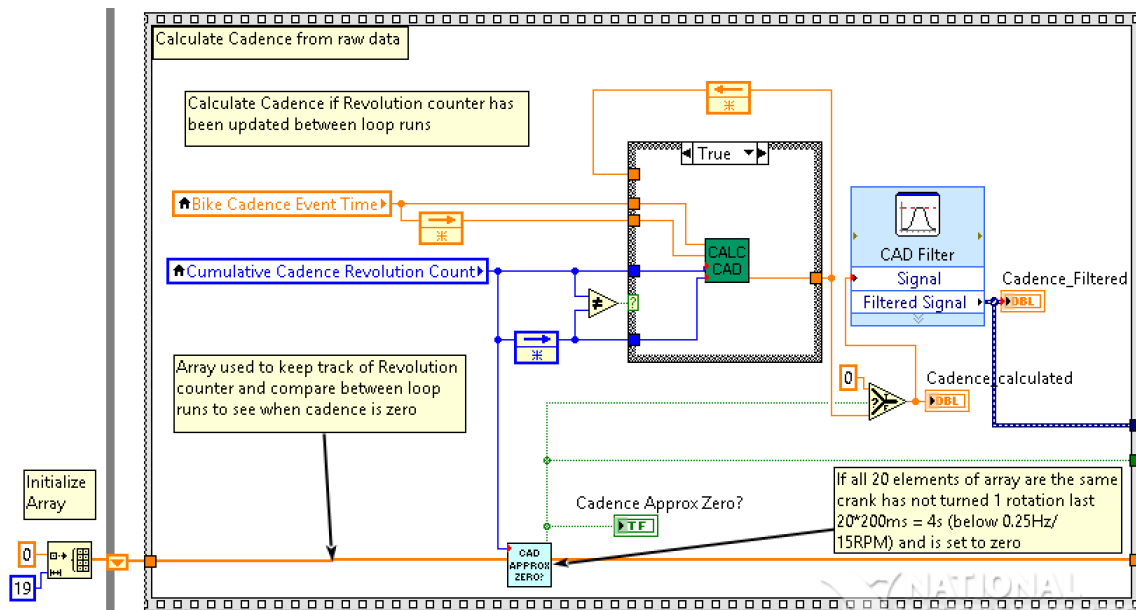


Figure A.13: Implementation of cadence calculation. The "CALC CAD" sub-VI gets event time and revolution count values from last loop run and the previous loop run values from feedback nodes. A new speed value is only calculated when the speed revolution counter has increased from previous loop run (True case). When the counter has not increased between runs (False Case), speed value is not updated. As there is no stop indicator boolean, the 20 latest revolution count values are collected in an array. The sub-VI "CAD APPROX ZERO?" compares the indices of the matrix and if they are all the same, the CAD is set to zero. That corresponds to setting cadence to zero for $CAD < 15 RPM$. Location: Loop C.

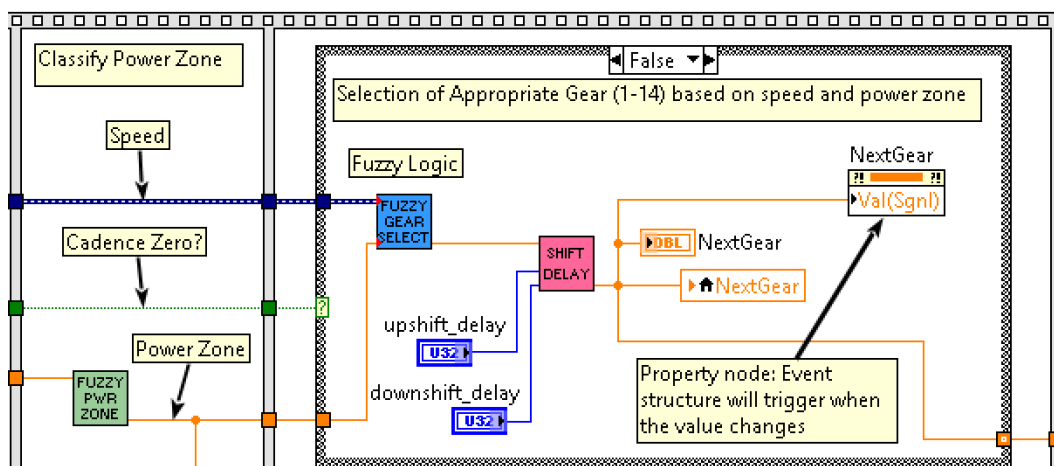


Figure A.14: Fuzzy logic gear selection process in loop C. The "Fuzzy PWR Zone" sub-VI classifies power into Zone 1,2 or 3. The "True" case for zero cadence is empty. The GSS keeps track of the current gear (G_C) and concurrently calculates an optimal gear according to sensor measurements. The new value for an optimal gear is shown as Next Gear (G_N). Gearshifts are made when $G_N \neq G_C$. When $G_N = G_C$, the system is in the optimal gear and no gearshift is performed.

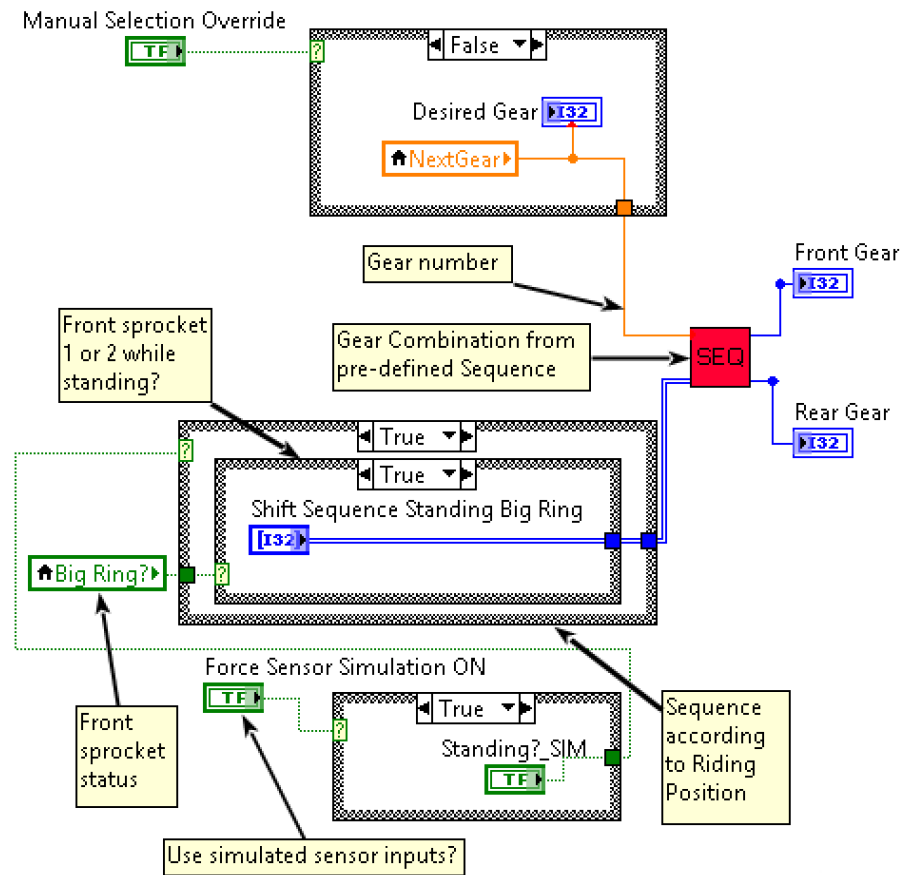


Figure A.15: An isolated syntax section from loop B showing how case structure is used to choose a gear shifting sequence based on riding position and front gear status. The syntax shows the sequence to choose when the rider start in a standing position. The value "NextGear" comes from loop C and corresponds to the optimal gear.

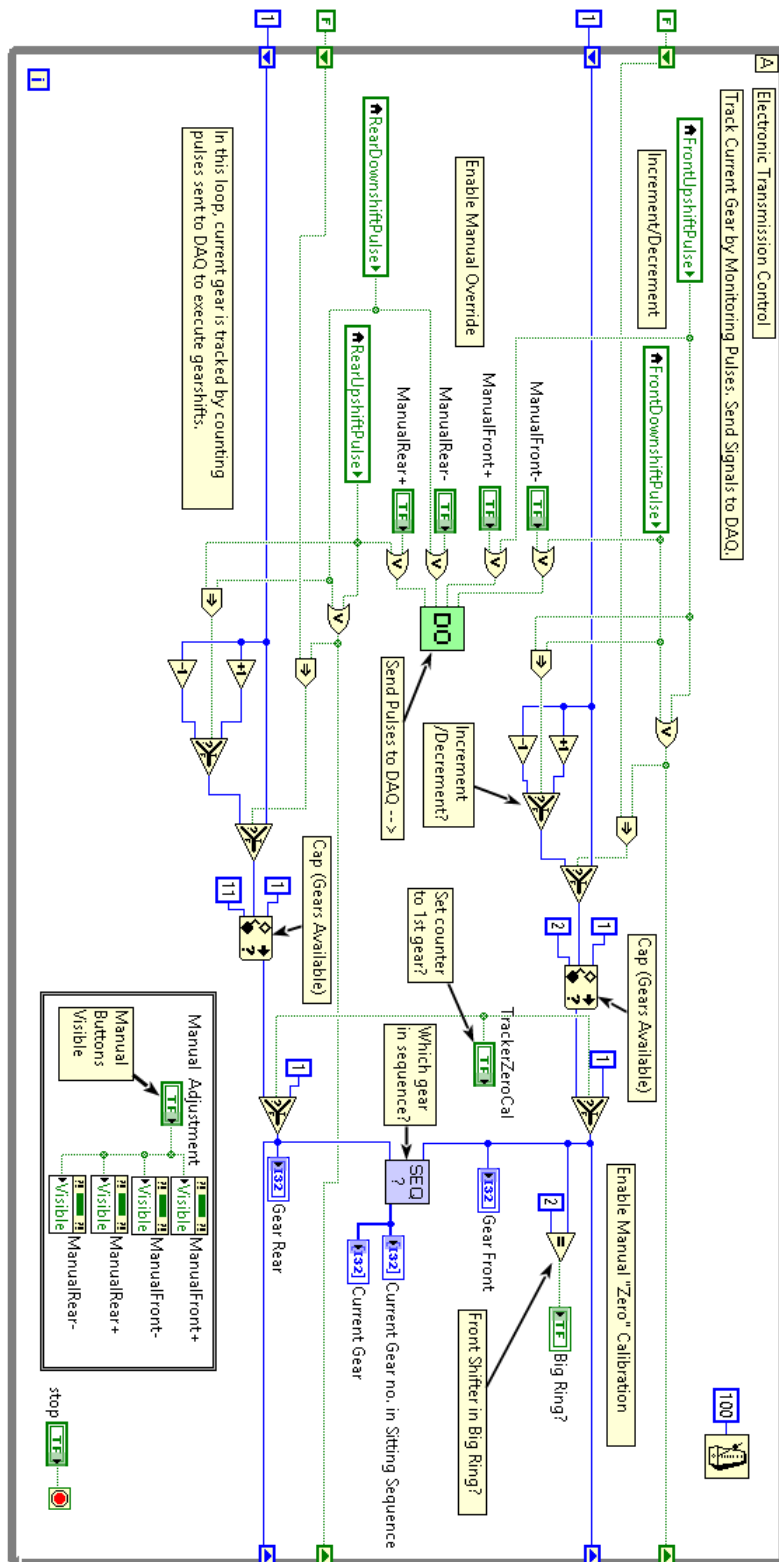


Figure A.16: Transmission control (loop A). The program keeps track of the current gear by counting shifting pulses received from loop B. The sub-VI "DO" receives the pulses and for each of the four shifting actions (front upshift, front downshift, rear upshift, rear downshift) it allocates the signals to the corresponding digital output channel on the DAQ device. The sub-VI "SEQ?" receives the front sprocket (S_F) and rear sprocket (S_R) combinations and calculates the current gear (G_C) according to the gear shifting sequence.

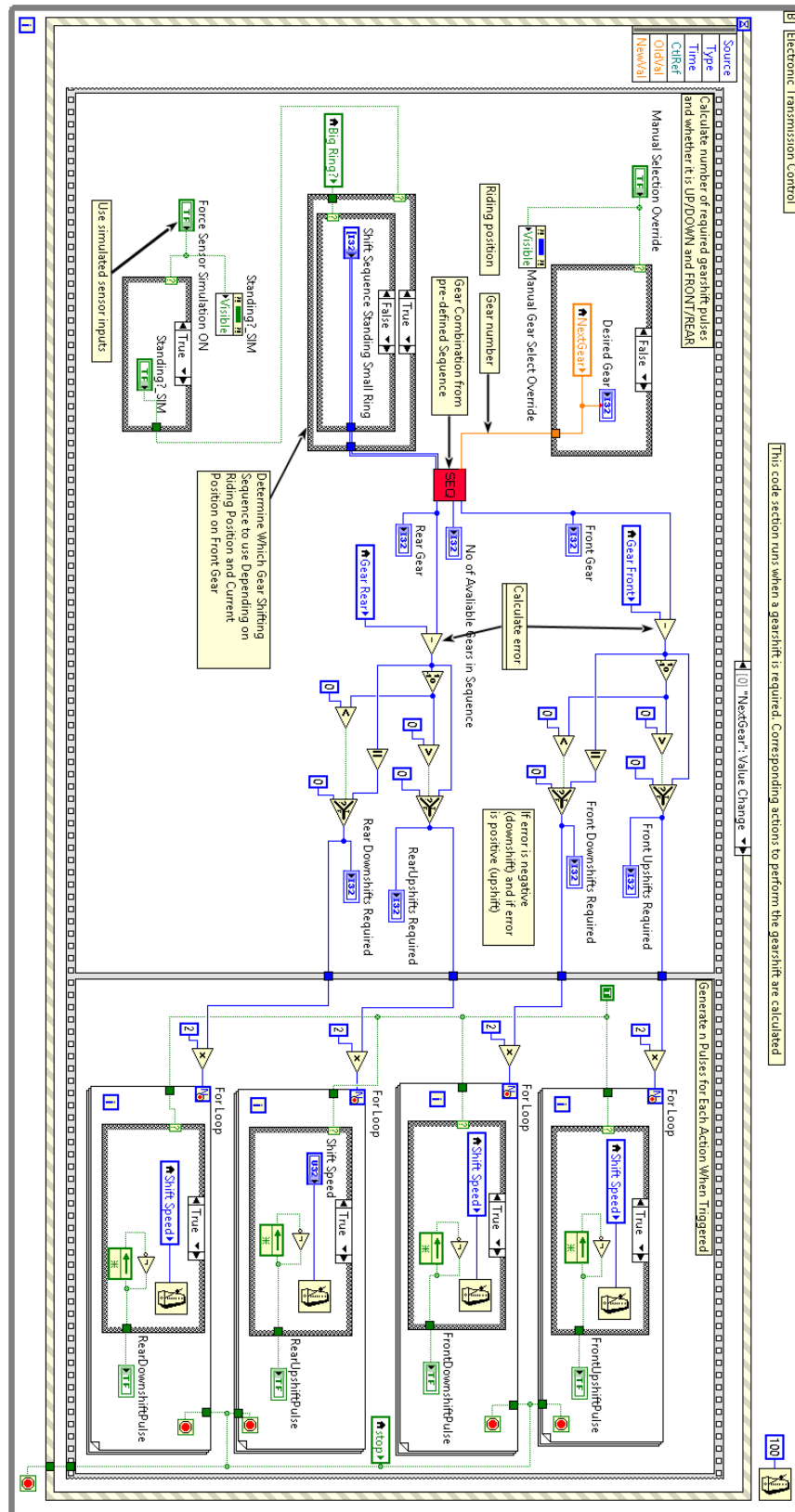


Figure A.17: Electronic transmission control (loop B). This syntax is placed within an event structure, which is triggered when "NextGear" value from loop C changes. In the first frame, current gear of the system is compared to the updated value of an optimal gear, which comes from loop C. Number of required upshifts/downshifts for each derailleur is calculated. In the second frame, pulses are generated in loops that are set to run as many times as the number of required gearshifts. Location: Loop B.

A.9 ANT+ device profile data pages

A.9.1 SPD data page 5

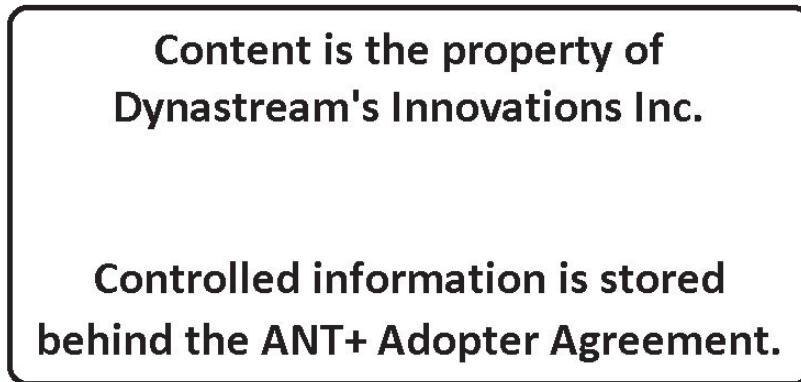


Figure A.18: The general SPD data page sent by ANT+ SPD sensor. The main data page is sent at ~4Hz. Screenshot from Dynastream's ANT+ Bike Speed device profile document (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014, p. 22*)

A.9.2 CAD data page 0

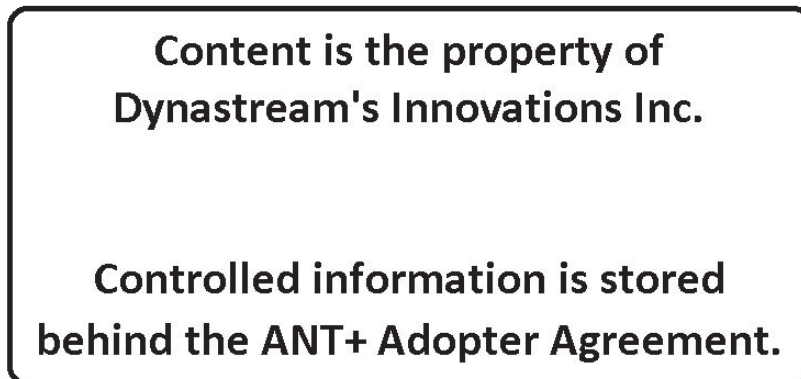
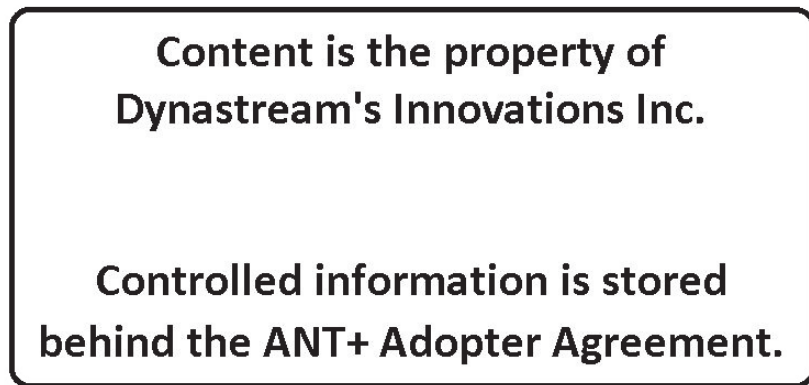


Figure A.19: Data page 0 sent by ANT+ CAD sensor. The main data page is sent at ~4Hz. Screenshot from Dynastream's ANT+ Bike Cadence device profile document (*ANT+ Device Profiles Bike Speed, Bike Cadence, Combined Bike Speed & Cadence. Revision 2.0., 2014, p. 31*).

A.9.3 PWR data page 16

**Content is the property of
Dynastream's Innovations Inc.**

**Controlled information is stored
behind the ANT+ Adopter Agreement.**

Figure A.20: Data page 16 sent by ANT+ PWR sensor. The datapage contains computed instantaneous power. Screenshot from Dynastream's ANT+ Bicycle Power device profile document (*ANT+ Device Profile - Bicycle Power. Revision 4.2., 2015, p. 31*).

A.9.4 PWR data page 19

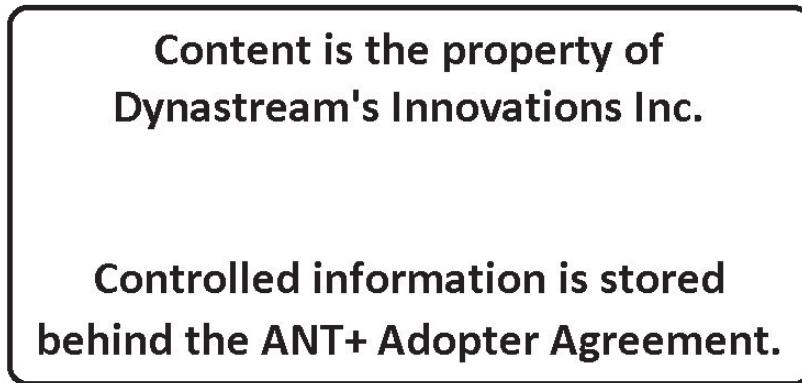


Figure A.21: Data page 19 is an optional page sent by some ANT+ power sensors, including the Garmin Vector 2. The data can be useful for implementing a soft sensor to determine cyclist's riding position. Screenshot from Dynastream's ANT+ Bicycle Power device profile document (*ANT+ Device Profile - Bicycle Power. Revision 4.2., 2015, p. 42*).



Test Plan Document

Author: Víðir Bjarkason

Test Plan for Prototype Optimal Gear Selection System

Contents

1	Introduction	2
2	System description	2
3	Prototype system setup	3
4	Prepare for testing	4
5	Testing of system	5
5.1	Test 1	5
5.1.1	Test 1a	5
5.1.2	Test 1b	5
5.2	Test 2	5
5.3	Test 3	5
5.4	Test plan checklist	6
6	Approval	6

1 Introduction

This test plan document was written for testing a prototype Gear Selection System (GSS) that was developed and built in the Master’s thesis "Optimal bicycle gear selection using multi-sensor data fusion." The thesis was written in spring 2016, on fourth semester of the Systems and Control Engineering masters program in University of Southeast Norway on Porsgrunn campus. Following procedures described in this document for testing the GSS should be carried out to make sure the system is functioning as intended.

2 System description

The goal with the project was to make an fully Automatic Transmission System (ATS) by integrate the GSS prototype with a commercial Electronic Transmission System (ETS) as shown in Figure 1. The ATS consists of the GSS, ETS and bike sensors. The GSS prototype system is designed to be carried by the user. Each module of the ATS is shown in Figure 2.

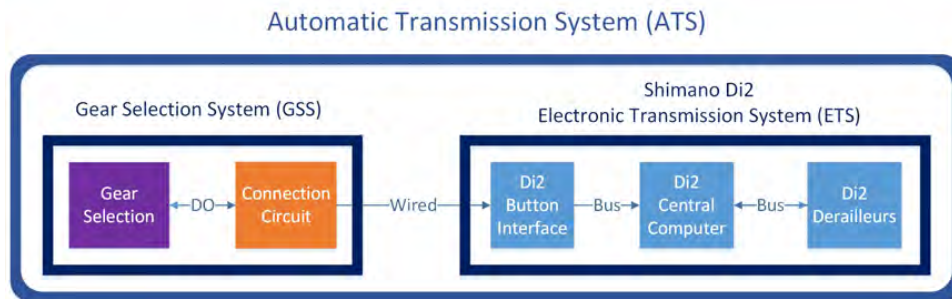


Figure 1: Overview of how the GSS prototype interacts with the ETS.

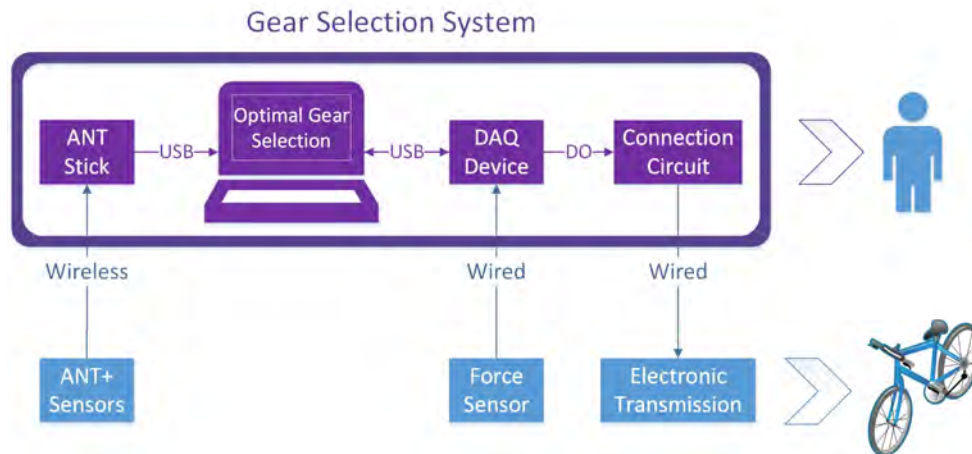


Figure 2: Overview of the complete ATS prototype. PC runs GSS software on LabVIEW that acquires wireless sensor data from USB connected ANT stick. Force sensor data acquisition is handled by the NI DAQ-6008 device. The software determines optimal gear based on the sensor data and signals the DAQ device to trigger gearshifts on the ETS through a custom made connection circuit.

3 Prototype system setup

The ETS, GSS and bike sensors should be connected as shown in Figure 3. When testing the system, the GSS hardware (PC, DAQ, ANT stick and the connection circuit) should be securely placed on a desk. The ETS should be installed on a demo bike that is operated on a stationary indoor trainer. The seat-mounted FlexiForce sensor should be installed on the bicycle seat using tape as shown in Figure 4.

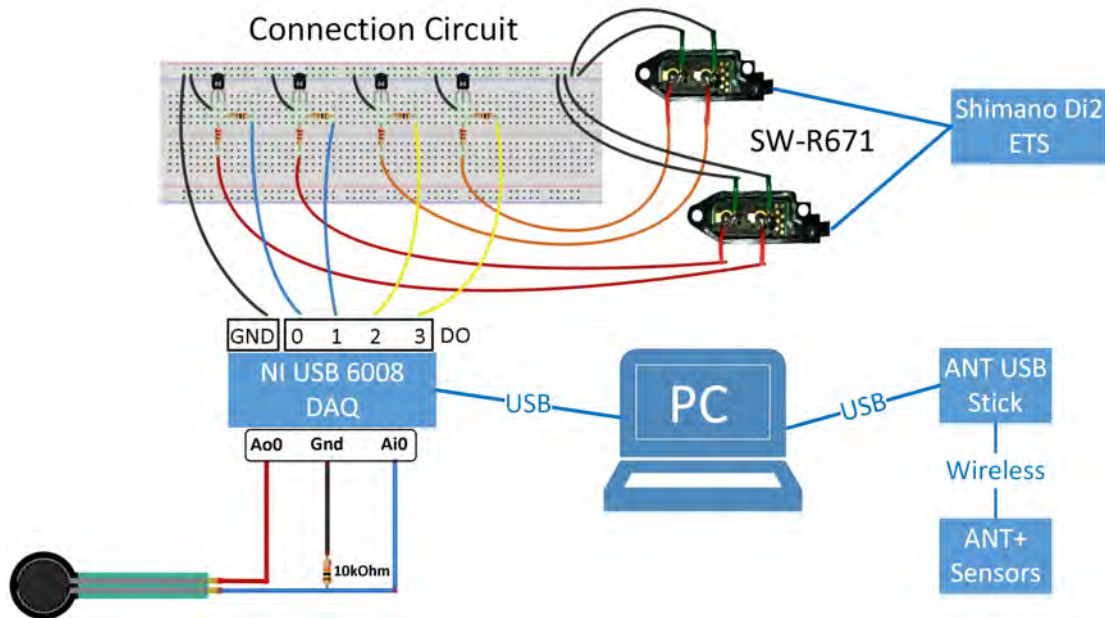


Figure 3: Overview of how all components of the GSS prototype are internally connected, and how they connect to the ETS and sensors.



Figure 4: FlexiForce pressure sensor mounted on bicycle saddle using clear tape. Ideal location of the pressure sensitivity part of the sensor depends on the saddle type and rider. An area where relatively high pressure is applied on the saddle while sitting should be chosen.

4 Prepare for testing

The two operations listed below require manual adjustments of the ETS. These adjustments are normally done by using the hand operated shifting buttons. As the shifting buttons have been modified to operate electronically, manual adjustments have to be done from the Transmission Control panel on the GUI of the GSS, shown in Figure 5.

1. Connect all the hardware as described in Section 3. Index gears according to Shimano Di2 user Manual.
2. Make sure the electronic transmission system is in 1st gear before start the operation of the GSS.

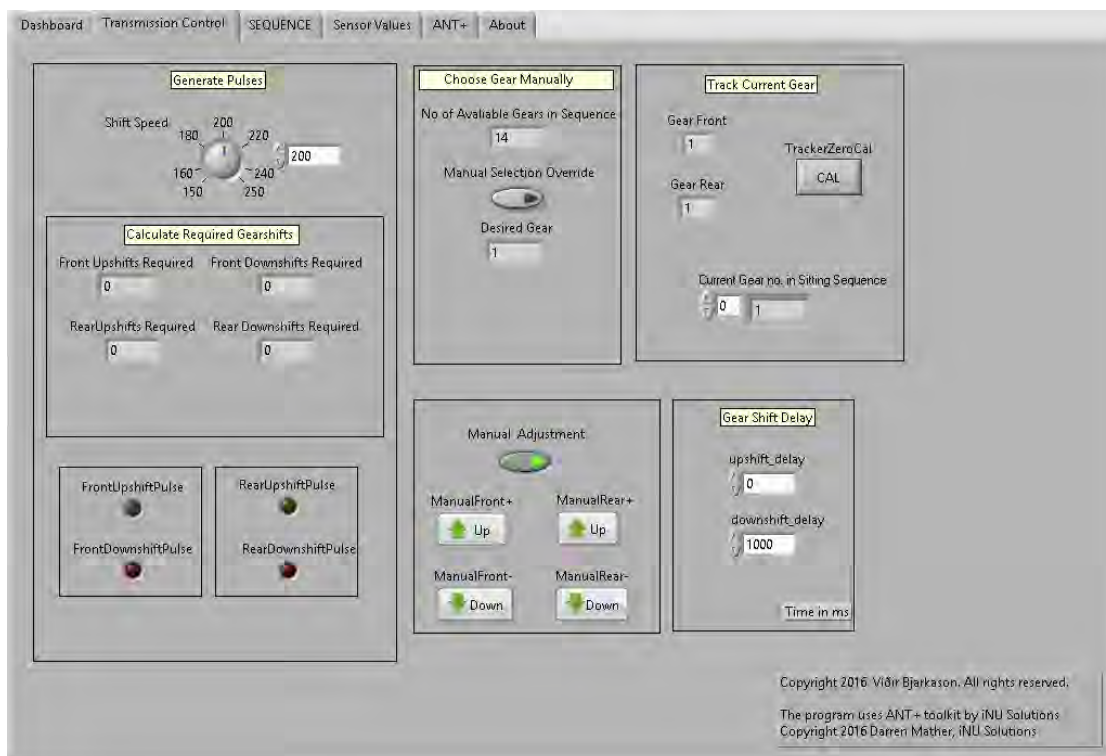


Figure 5: The transmission control panel on the GUI. Gear shift speed and time delays can be adjusted. Manual control of ETS buttons is done by selecting Manual Adjustment and then clicking the arrow buttons.

5 Testing of system

This section describes testing procedures for the prototype GSS. The GSS prototype was connected to the ETS and the system tested with a demo bike mounted on a stationary indoor trainer.

5.1 Test 1

Test 1 - Riding in the saddle: To test the GSS performance in maintaining optimal cadence for different power zones. Power zone is kept constant by simulating the power sensor output. The rider pedals and builds up speed until the GSS has shifted through the majority of the gears. Then the rider should slow down to rest. This test is divided into two separate tests where the GSS seeks to maintain different goal cadences.

5.1.1 Test 1a

Optimal cadence according to power zone 2. Optimal gear selection is set according to power zone 2. The GSS goal cadence is $80RPM$.

Expected results: When speed is increased, the GSS should perform gearshifts to maintain the pre-set $80RPM$ goal cadence. The average cadence is expected to be $(80 \pm 5)RPM$.

5.1.2 Test 1b

Optimal cadence according to power zone 3. Optimal gear selection is set according to power zone 3. The GSS goal cadence is $90RPM$.

Expected results: When speed is increased, the GSS should perform gearshifts to maintain the pre-set $90RPM$ goal cadence. The average cadence is expected to be $(90 \pm 5)RPM$.

5.2 Test 2

Riding out of the saddle: To test the effect of the riding position on the GSS, the rider should ride in a standing position when a gearshift between 7th and 8th gear is required. Riding positions should be altered. The rider starts pedaling out of the saddle, builds up speed until the system has shifted to the highest gear allowed in a standing position. Then the rider sits down so the system can shift the front derailleur. Speed should be increased until the system has shifted into the highest gear. After that the rider should slow down to rest.

Expected results: When riding in a standing position, the GSS should not perform gearshifts that require front derailleur movements. According to the gear shifting sequence, shifting from 7th to 8th gear, or vice versa will be disabled until the rider moves to a sitting position.

5.3 Test 3

Coasting: To test the effect of stopping the pedaling movement on the GSS, the rider starts pedaling and builds up speed until the GSS has shifted higher than the 10th gear. To allow coasting, disconnect the resistance unit on the stationary trainer from rear wheel and stop pedaling at the same time. The rider lets the rear wheel coast for approximately 30 seconds and starts pedal again before slowing down to rest. **Expected results:** When pedaling is stopped the cadence measurement should indicate $CAD = 0$. Then the GSS should not perform any gearshifts. The GSS starts shifting gears when pedaling is started again. Note: an assistant person is required to disconnect the resistance unit from the rear wheel, while the rider is on the bike.

5.4 Test plan checklist

Check list in Table 1 should be filled out by the technician that performs the tests on the GSS prototype. After testing and filling the checklist, the technician(s) should sign the test document in Section 6.

Table 1: A checklist for testing of optimal gear selection system prototype. For test 1a and 1b, the average cadence can alter by (± 10) RPM to pass the test.

Test #	Description	Criteria	Passed	Not Passed	Comments
Test 1a	Verify that 80 RPM average cadence is maintained				
-					
Test 1b	Verify that 90 RPM average cadence is maintained				
-					
Test 2	Verify that front derailleur shifts are disabled while riding out of the saddle.				
-					
Test 3	Verify that all gearshifts are disabled while not pedaling				
-					

6 Approval

The signed individuals confirm they have tested the prototype system according to the procedures defined in this test plan document.

Name:

Name:

Role:

Role:

Date:

Date:

References

- Ant+ brand.* (2016). Dynastream Innovations Inc. Retrieved from <https://www.thisisant.com/business/go-ant/ant-brand>
- Ant+ device drivers.* (2014, June). Retrieved from http://inusolutions.com/en/Projects/Entries/2014/6/27_ANT%2B_Device_Drivers.html
- ANT+ Device Profile - Bicycle Power. Revision 4.2.* (2015, October). Dynastream Innovations Inc. Retrieved 2016-07-03, from <https://www.thisisant.com/resources/bicycle-power/>
- Ant+ device profiles bike speed, bike cadence, combined bike speed & cadence. revision 2.0.* (2014, May). Dynastream Innovations Inc. Retrieved 2016-03-23, from <https://www.thisisant.com/resources/bicycle-speed-and-cadence/>
- Ant message protocol and usage - rev. 5.1.* (2014, April). Dynastream Innovations Inc. Retrieved from <https://www.thisisant.com/resources/ant-message-protocol-and-usage/>
- Beer, J. (2008, May). *Technique: Cadence matters.* Retrieved from <http://www.bikeradar.com/gear/article/technique-cadence-matters-16394/>
- Bike cadence sensor.* (2016). Garmin Ltd. Retrieved 2016-03-20, from <https://buy.garmin.com/en-GB/GB/shop-by-accessories/fitness-sensors/bike-cadence-sensor/prod517580.html>
- Bikecalc.com - how to calculate bicycle wheel size.* (2014). Bikecalc.com. Retrieved from http://www.bikecalc.com/wheel_size_math
- BikeRadar. (2014, October). *The bike that shifts itself - bioshift first ride.* Retrieved from <https://www.youtube.com/watch?v=pJYivtHcl8o>
- Bike speed sensor.* (2016). Garmin Ltd. Retrieved 2016-03-20, from <https://buy.garmin.com/en-GB/GB/shop-by-accessories/fitness-sensors/bike-speed-sensor/prod517136.html>
- Bio shift - just pedal.* (n.d.). Retrieved from <http://www.baronbiosys.com/>
- Brown, S. (2007). *Derailer Adjustment.* Retrieved from <http://sheldonbrown.com/derailer-adjustment.html>
- Brown, S. (2008). *Pins & Ramps.* Retrieved from http://www.sheldonbrown.com/gloss_p.html#pins
- Coast, J. R. & Welch, H. G. (1985, February). Linear increase in optimal pedal rate with increased power output in cycle ergometry. *European Journal of Applied Physiology and Occupational Physiology*, 53(4), 339–342. Retrieved from <http://link.springer.com/>

article/10.1007/BF00422850 doi: 10.1007/BF00422850

Company history. (2016). Retrieved from <http://www.shimano.com/content/Corporate/english/index/portal-site/history1.html> (Shimano Inc. 2016)

DataSocket Transfer Protocol (dstp) Overview. (2016, January). Retrieved 2016-05-28, from <http://www.ni.com/white-paper/3223/en/>

Download ubuntu desktop. (2016). Retrieved from <http://www.ubuntu.com/download/desktop/>

Dynastream Innovations Inc. (2016). *Ant/ant+ defined*. Retrieved from <https://www.thisisant.com/developer/ant-plus/ant-antplus-defined/>

Edge 520. (2016). Retrieved 2016-05-28, from <https://buy.garmin.com/en-GB/GB/sports-recreation/cycling/edge-520/prod166370.html>

ELLEN. (2012, May). *Flexi force sensor - arduino*. Arduino.learn();. Retrieved 2016-04-09, from <http://arduino.sundh.com/2012/05/flexi-force-sensor/>

E-tube project. (2015, December). Shimano Inc. Retrieved from <http://e-tubeproject.shimano.com/index.html>

Fairchild semiconductor - 2n3704. (2012, July). Datasheetcatalog. Retrieved from <http://pdf.datasheetcatalog.com/datasheet/fairchild/2N3704.pdf>

Flexiforce standard model a201 - datasheet rev. a. (n.d.). Tekscan. Retrieved from <https://www.tekscan.com/sites/default/files/resources/FLX-A201-A.pdf>

Frenzel, L. (2012, November). *What's the difference between bluetooth low energy and ant?* Retrieved from <http://electronicdesign.com/mobile/what-s-difference-between-bluetooth-low-energy-and-ant>

Fritzing. (2016). Retrieved from <http://fritzing.org/>

GoldenCheetah. (2016). Retrieved from <http://www.goldencheetah.org/>

Golden cheetah open source project. (2016). Retrieved 2016-03-26, from <https://github.com/GoldenCheetah>

Gotshall, R., Bauer, T. & Fahrner, S. (1996, February). Cycling Cadence Alters Exercise Hemodynamics. *International Journal of Sports Medicine*, 17–21. Retrieved from https://www.researchgate.net/profile/Robert_Gotshall/publication/14426392_Cycling_Cadence_Alters_Exercise_Hemodynamics/links/546a35470cf20dedafd3841e.pdf doi: 10.1055/s-2007-972802

Huang, J. (2009, August). *Shimano Dura-Ace Di2 electronic transmission review*. Retrieved from <http://www.bikeradar.com/gear/category/components/gear-shifters/product/dura-ace-di2-transmission-34981/>

IDC: Smartphone OS Market Share. (2015, August). Retrieved from <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>

- Kadlec, P., Gabrys, B. & Strandt, S. (2009, April). Data-driven Soft Sensors in the process industry. *Computers & Chemical Engineering*, 33(4), 795–814. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0098135409000076> doi: 10.1016/j.compchemeng.2008.12.012
- Labview system design software*. (2016). National Instruments. Retrieved from <http://www.ni.com/labview/>
- Maker, R. (2014, October). *Hands-on with the bioshift automated bike shifting system*. Retrieved from <http://www.dcrainmaker.com/2014/10/bioshifts-automated-shifting.html>
- Mather, D. (2015, March). *Antusb-m driver installation. ni-visa and windows 7 rev. 1.0*. iNU Solutions.
- Murphy, B. (2014, July). *Science Behind the Magic | Freehub Design*. Retrieved from <http://blog.artscyclery.com/science-behind-the-magic/science-behind-the-magic-freehub-design/>
- National Instruments- Image Gallery*. (2009). Retrieved from http://sine.ni.com/gallery/app/ui/page?nodeId=201986&mTitle=USB-6008&mGallery=set_usb-6008
- Negnevitsky, M. (2011). *Artificial intelligence - a guide to intelligent systems* (third edition ed.). Harlow, England: Pearson Education Limited.
- Ni myrio*. (2016). Retrieved from <http://www.ni.com/myrio/>
- Ni usb-6008/6009 user guide*. (2015, July). National Instruments. Retrieved from <http://www.ni.com/pdf/manuals/371303n.pdf>
- The OSI Model's Seven Layers Defined and Functions Explained*. (2014, June). Retrieved from <https://support.microsoft.com/en-us/kb/103884>
- Qt - home*. (2016). Retrieved from <http://www.qt.io/>
- R671 remote triathlon shifter*. (2015). Shimano. Retrieved 2016-03-20, from <http://cycle.shimano-eu.com/content/seh-bike/en/home/components1/road/ultegra-di2/sw-r671.html>
- Rossiter, W. (2015, August). *Sram red etap ushers in wireless shifting era*. Retrieved from <http://www.bikeradar.com/road/news/article/sram-red-etap-ushers-in-wireless-shifting-era-45095/>
- Sarti, D. (2013, May). *Hacking the Heck Out of My Di2 9070*. Retrieved from <http://dinosarti.com/blog/2013/5/23/hacking-the-heck-out-of-my-di2-9070>
- Scarf, P., Jobson, S., Passfield, L. & Reed, R. (2016). Determining optimal cadence for an individual road cyclist from field data. *Taylor & Francis Online*. Retrieved from <http://www.tandfonline.com/doi/pdf/10.1080/17461391.2016.1146336> doi: 10.1080/17461391.2016.1146336
- Sedra, A. S. & Smith, K. C. (2011). *Microelectronics circuits - international edition* (sixth ed.). New York: Oxford University Press.

- Shimano Ultegra 6870 Series Dealer's Manual*. (2014, December). Shimano Inc. Retrieved 2016-06-03, from <http://si.shimano.com/php/download.php?file=pdf/dm/DM-UL0001-02-ENG.pdf>
- Simulant+*. (2016). Dynastream Innovations Inc. Retrieved 2016-03-25, from <https://www.thisisant.com/resources/simulant-1>
- Stebbins, C. L., Moore, J. L. & Casazza, G. A. (2014, January). Effects Of Cadence on Aerobic Capacity Following a Prolonged, Varied Intensity Cycling Trial. *Journal of Sports Science & Medicine*, 13(1), 114–119. Retrieved from <http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3918546/>
- Sweatman, M. (n.d.). *Browning derailleurs*. Retrieved 2016-05-08, from http://www.disraeliegears.co.uk/Site/Browning_derailleurs_page_2.html
- Takaishi, T., Yasuda, Y., Ono, T. & Moritani, T. (1996, December). Optimal pedaling rate estimated from neuromuscular fatigue f or cyclists. *Medicine & Science in Sports & Exercise*, 28(12), 1492–1497. Retrieved from http://journals.lww.com/acsm-msse/Fulltext/1996/12000/Optimal_pedaling_rate_estimated_from_neuromuscular.8.aspx
- Ultegra di2*. (2015). Shimano. Retrieved 2016-03-20, from <http://cycle.shimano-eu.com/content/seh-bike/en/home/components1/road/ultegra-di2.html>
- Usb ant stick*. (2016). Garmin Ltd. Retrieved 2016-03-25, from <https://buy.garmin.com/en-US/US/shop-by-accessories/fitness-sensors/usb-ant-stick-/prod10997.html>
- Vector 2*. (2016). Garmin Ltd. Retrieved 2016-03-20, from <https://buy.garmin.com/en-GB/GB/sports-recreation/cycling/vector-2/prod510941.html>
- Vortex-T2180*. (2016). Retrieved 2016-03-17, from <https://www.tacx.com/en/products/trainers/vortex-smart>
- Why Avoid Cross Chaining Gears On Your Bike*. (2012). Retrieved from <http://www.bicyclechainrings.com/crosschaining.html>
- The wireless sensor network solution*. (2016). Dynastream Innovations Inc. Retrieved from <https://www.thisisant.com/>
- Zwift*. (2016). Retrieved from <http://zwift.com>