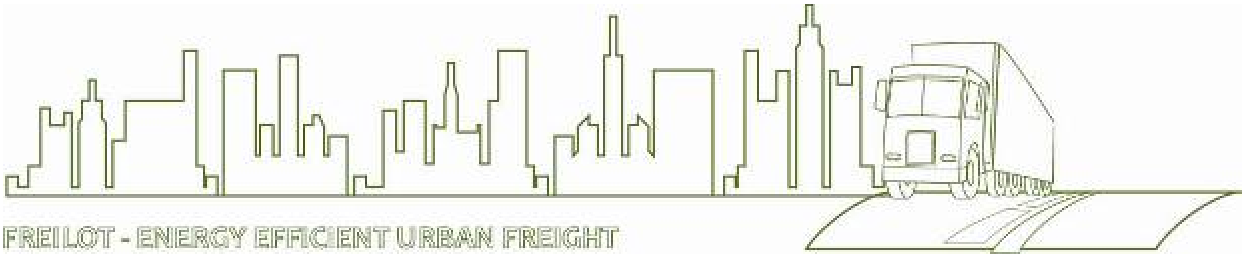
FREILOT

Urban Freight Energy Efficiency Pilot

D.FL.2.5 Krakow prototype



Version number	Version 2.0
Main author	Mariusz Karp, Peek Traffic
Dissemination level	PU
Lead contractor	ERTICO – ITS Europe
Due date	31.11.2010
Delivery date	31.01.2011
Delivery current version	01.07.2011

Information and Communications Technologies Policy Support Programme (the "ICT PSP")
Information Society and Media Directorate-General
Grant agreement no.: 238930
Pilot type B

Revision and history sheet

Version history			
Version	Date	Main author	Summary of changes
0.1	30.11.2010	Mariusz Karp	First draft
1.0	17.01.2011	Mariusz Karp	Updated the document based on the peer review comments and added Executive summary
1.1	27.06.2011	Dominique Gaymu	Updates in In-vehicle systems
2.0	30.06.2011	Maria Tevell	Update of the Volvo systems

	Name	Date
Prepared	Mariusz Karp	30/06/2011 (version 2.0)
Reviewed	Eric Koenders	30/06/2011 (version 2.0)
Authorised	Zeljko Jeftic	01/07/2011 (version 2.0)

Circulation	
Recipient	Date of submission
European Commission	01/07/2011 (version 2.0)
Project partners	01/07/2011 (version 2.0)

Authors (full list)

Mariusz Karp, Peek Traffic

Dominique Gaymu, Volvo

Maria Tevell, Volvo

Project co-ordinator

Zeljko Jeftic
ERTICO – ITS Europe
Avenue Louise 326
1050 Brussels, Belgium

Tel.: +32 2 300 07 31
Fax: +32 2 300 07 01
Email: z.jeftic@mail.ertico.com

Legal Disclaimer

The information in this document is provided “as is”, and no guarantee or warranty is given that the information is fit for any particular purpose. The above referenced consortium members shall have no liability for damages of any kind including without limitation direct, special, indirect, or consequential damages that may result from the use of these materials subject to any liability which is mandatory due to applicable law.

Table of Contents

Table of Figures

Abbreviations and Definitions

Abbreviation	Definition
ASN.1	Abstract Syntax Notation 1
BER	Basic Encoding Rules
CAM	Cooperative Awareness Message
CVIS	Cooperative Vehicle Infrastructure Systems
EC-1	EuroController-1, a Peek traffic controller
EC-2	EuroController-2, a Peek traffic controller
FIFO	First In, First Out method
ETSI	European Telecommunications Standards Institute
LDM	Local Dynamic Map
OBU	On Board Unit
OSGi	Open Services Gateway initiative
RSU	Road Side Unit
PAV	Priority Application Vehicle side
PAR	Priority Application Roadside
RFI	Radio Frequency Interference
SPITS	Dutch ITS project developing deployable cooperative systems
TCP/IP	Transmission Control Protocol/Internet Protocol

Executive Summary

FREILOT pilot site Krakow uses two from the four defined services to improve transport efficiency.

The main transit route (national road no 4 and no 75) from eastern border towards Krakow has been fitted with Energy Efficient Intersection Control systems from PEEK Traffic. Here c.a. 10 heavy goods vehicles from local logistic and bus operators get priority when they leave or enter their home distribution centre. Cooperative technology at 8 intersections (Figure 1) provides the trucks with a speed advice and a visual indication of the red or green time left. This allows the driver to anticipate the changes of the traffic light, reducing the number of stops. A dedicated onboard computer handles the radio communication and the driver display.

Five of the Temperi trucks will be equipped with the Volvo Eco Driving Support system. This system monitors the behaviour of the driver and gives feedback and advice how to improve fuel efficiency.

In Krakow the speed and acceleration limiters as well as delivery space management system of FREILOT will not be implemented because the fleet operators are not interested in such options and delivery space is not a scarce resource in Krakow.

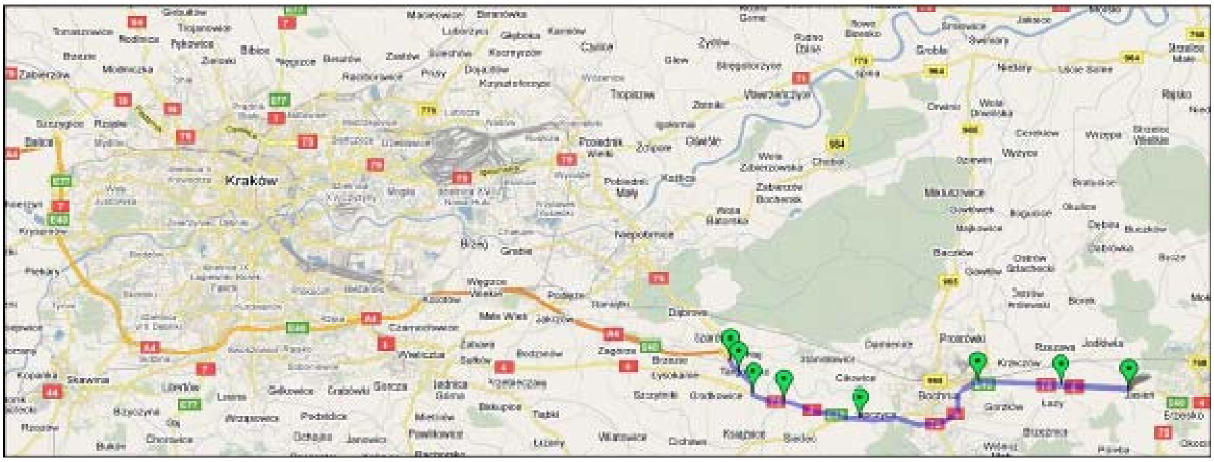


Figure 1 - FREILOT equipped intersections in Krakow

1. Introduction

This document describes the actual implementation of the FREILOT scheme in Krakow, Poland.

For all FREILOT services the technology that has been used is described, followed by the installations at the roadside and inside the trucks. In the current version of the deliverable this is done for the Energy Efficient Intersection Control system.

In Krakow the Adaptive Speed Limiter, the Acceleration Limiter and Delivery Space Booking are not implemented, consequently such a system is not described here.

2. Energy Efficient Energy Efficient Intersection Control

2.1. System components

The Energy Efficient Intersection Control system is built around two systems: a roadside system (RSU) which connects to the traffic control systems, and an on-board unit (OBU) which communicates with the truck driver. Each system contains two computing units: the router and the host. This is shown in Figure 2.

The router handles all radio communication: the GPS receiver and the 802.11p WLAN radio. The latter is used to communicate between the RSU and the OBU.

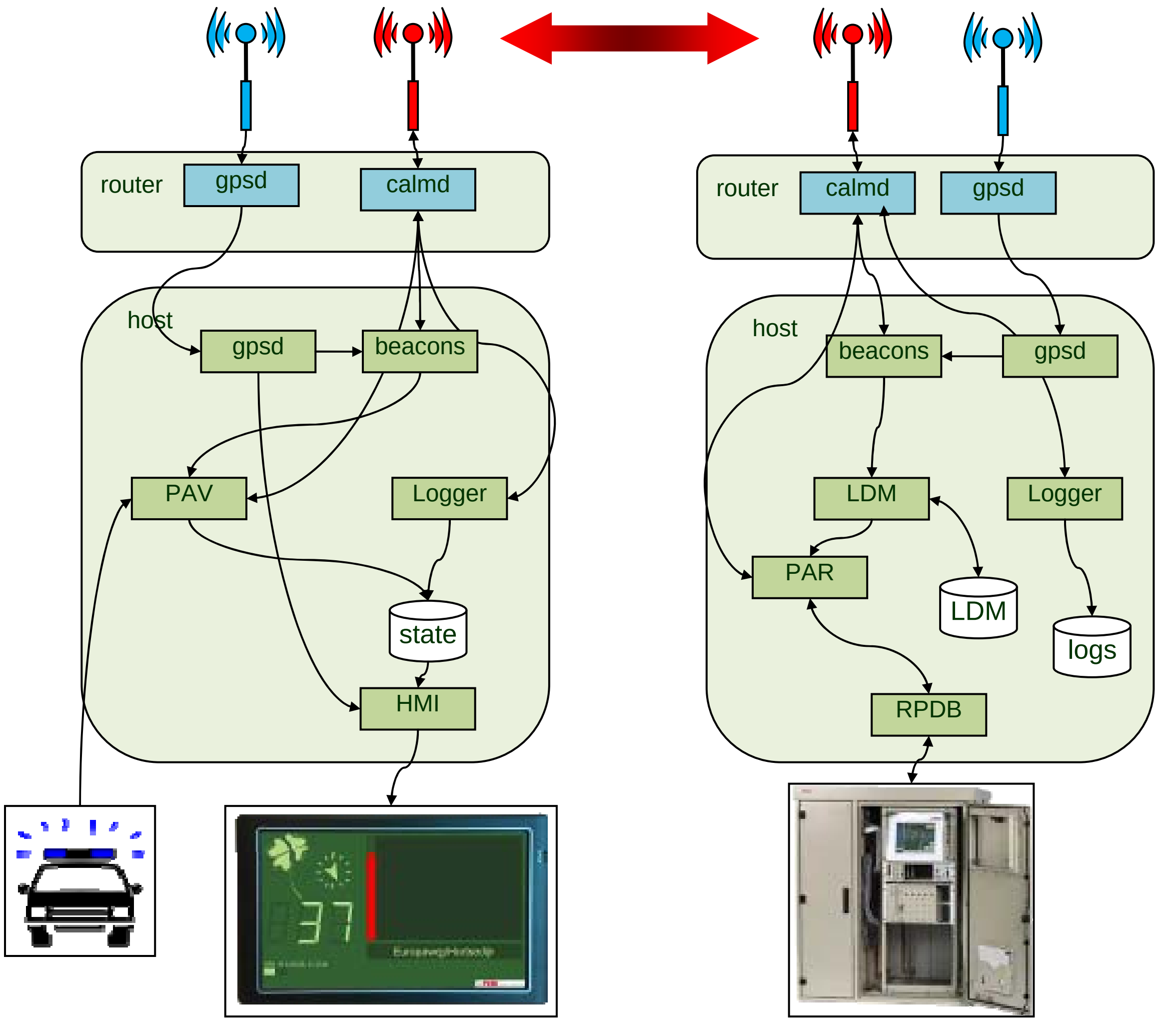


Figure 2 - The Energy Efficient Intersection Control components

The host computer contains the software components that implement the priority application and position logging. The design of the application can be found in D.FL.2.1 Implementation plan.

The following components are shown in the drawing:

Gpsd	GPS receiver protocol decoding
Calmd	ISO CALM M5 implementation
Beacons	Beacon (CAM message) transmission and reception
PAV	Priority Application Vehicle side

PAR	Priority Application Roadside
Logger	Vehicle position log handler
State	Current state of the OBU
HMI	Human Machine Interface
LDM	Local Dynamic Map
Logs	Log storage in files
RPDB	Remote Parameter DataBase protocol handler

2.1.1. Software

The FREILOT software on the OBU and RSU is built from and on Open Source components. The following Open Source software is used (not an exhaustive list):

The Linux kernel (2.6.32) http://www.kernel.org

The ath5k WLAN driver http://wireless.kernel.org/en/users/Drivers/ath5k

The openWRT Linux distribution http://openwrt.org/

The Voyage Linux distribution http://linux.voyage.hk/

The Knopflerfish OSGi implementation http://www.knopflerfish.org/

Sun/Oracle Java http://www.oracle.com/technetwork/java/javase/downloads/index.html

The gpsd deamon http://gpsd.berlios.de/

CVIS CAM decoding/encoding library https://cvis.be/trac/cvis-comm

The system software is currently using the CVIS CAM definition and CALM FAST. As the software development within Peek follows the standardisation efforts in Europe the systems will be updated to the latest available software release. As an example, the systems will be upgraded to conform to the results of the SPITS project, which uses the ETSI CAM definition.

The applicable source code of the software created for FREILOT can be obtained from Peek Traffic (email:eric.koenders@peektraffic.nl).

The FREILOT RSU is connected to existing traffic light controllers from Peek (EC-2 and EC-1). In Krakow the intersections works as stand alone controlled. To create the FREILOT priority scheme the following changes have been made to the traffic controller configuration:

2.1.2. Hardware

For FREILOT only off-the-shelf hardware is used to build the OBU and the RSU. The following major parts are used:

Router CPU:	Mikrotik routerboard 411
WLAN radio:	Mikrotik R52H
Host PC:	LEC-3000
Panel PC:	PDX-089T

2.1.3. Roadside unit

The RSU router is designed to be mounted on a mast/gantry. It contains the WLAN and GPS antenna's, the WLAN radio and a processor. It is connected to the host computer and powered via Ethernet.

This design has been chosen to minimise the length of the antenna cable (reducing signal loss) while at the same time allowing a line of sight between the antenna and the oncoming vehicles. The router enclosure is shown in Figure 3.

Figure 3 - The RSU router

The host system in the RSU is an off-the-shelf embedded PC which is suited for installation in an outdoor cabinet, see Figure 4. The host has two Ethernet interfaces, one is connected to the router, and one is used to connect to the traffic controller.

Figure 4 - The RSU host computer

During in-house tests it turned out that older traffic controllers of the EC-1 type were not able to handle the constant interrogation by the FREILOT host. This problem has been solved by upgrading the processing unit of EC-1 traffic controllers with an EC-2 compatible processor.

2.1.4. Onboard unit

Like the RSU the OBU is composed of two computing units: the router with the WLAN radio and antenna, and the host containing the application software and the graphical display. To ease installation the OBU is mounted in the vehicles with a suction cup.

Figure 5 - The onboard unit

The OBU antenna is a small directional patch antenna. It sits at the back of the OBU and points through the windscreen towards an upcoming intersection. Using this antenna greatly simplifies the installation. As a consequence the vehicle has almost no radio range at its back side. For FREILOT this poses no problem because the vehicle is always driving towards an intersection when it should get priority.

In the original design the WLAN radio was placed inside the panel (host) computer. During operational tests it turned out that the internal power supply of the host computer cannot deliver enough power for the radio. This leads to spurious resets of the computer, especially when the OBU is powered with 24V. The miniPCI slot inside the host computer should conform to the standard, but apparently it does not comply to the power requirements of the standard.

This problem has been solved by moving the WLAN radio from the host into a separate router unit, which is mounted at the back side of the host. Both units are connected via Ethernet, similar to the RSU.

After installation some drivers have reported that the OBU interferes with some GPS receivers, for example of a navigation system. Measurements at Peek have shown that this is caused by RFI from the host computer. The only solution is to replace the host computer by another type, this is currently under investigation. In the mean time the trucks will use the current units. The truck drivers are informed to switch off the unit if they drive outside Krakow and the units disturbs GPS reception.

To solve the GPS reception problems a replacement for the OBU display has been developed which is based on Android. When enough new units are produced the OBU's in Krakow will be replaced with the new ones. A prototype of the Android based OBU is shown in Figure 6.

Figure 6 - The new Android based OBU

The onboard unit provides the truck driver with operational data like heading, actual speed and GPS reception quality. When the truck reaches an equipped intersection the name of the intersection is shown. If priority is provided, depending on the situation, a speed advice is shown, and a bar counting down the current state of the traffic light. Using the countdown bar the driver can anticipate and adapt his speed to cross the intersection without stopping.

The driver display is show in Figure 7.

Figure 7 - FREILOT OBU in action

2.2. Roadside installations

2.2.1. #1 Szarów

At #1 the router is placed in a lighting pole of signal group K4R at the North side of the exit from Highway A4.

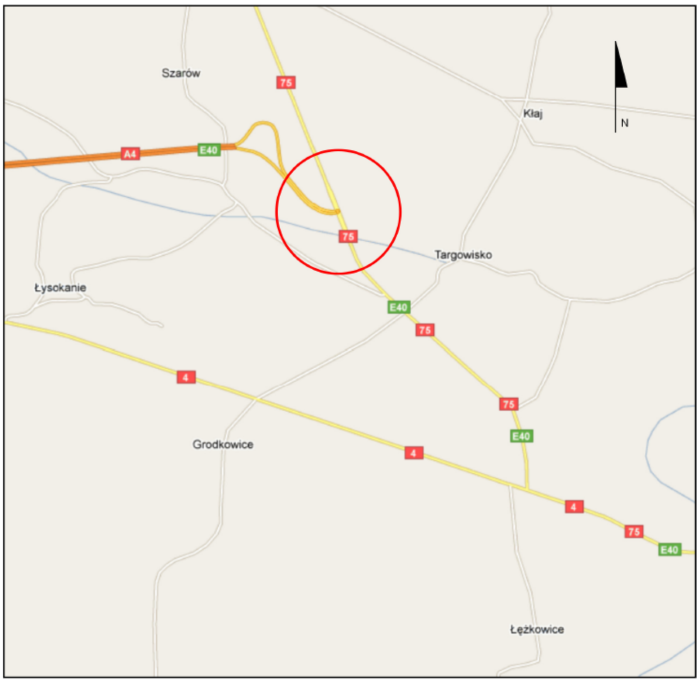


Figure 8 - Intersection location #1

Figure 9 - Router at #1

2.2.2. #2 Targowisko1

At #2 the router is placed in a lighting pole of signal group K2L above of road no 75.

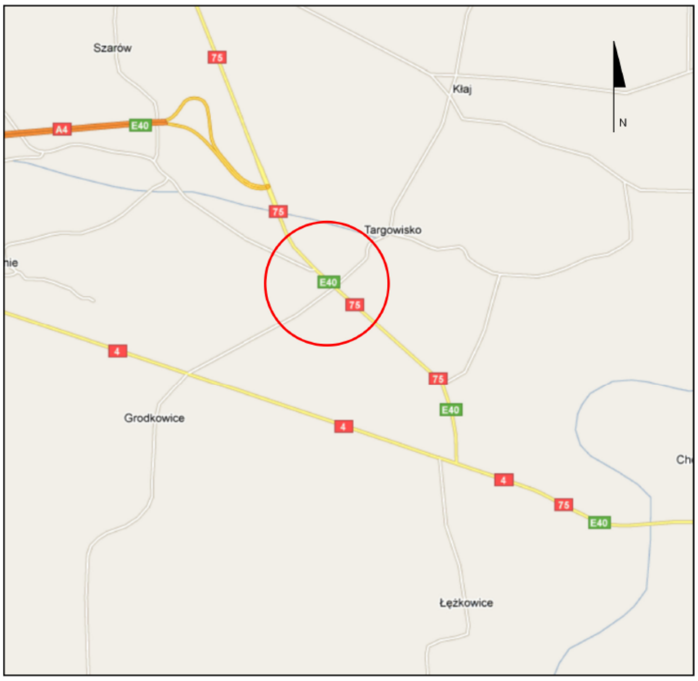


Figure 10 - Intersection location #2

Figure 11 - Router at #2

2.2.3. #3 Targowisko2

At #3 the router is placed in a lighting gantry of signal group K2L and K2P above of north entry of road no 75 .

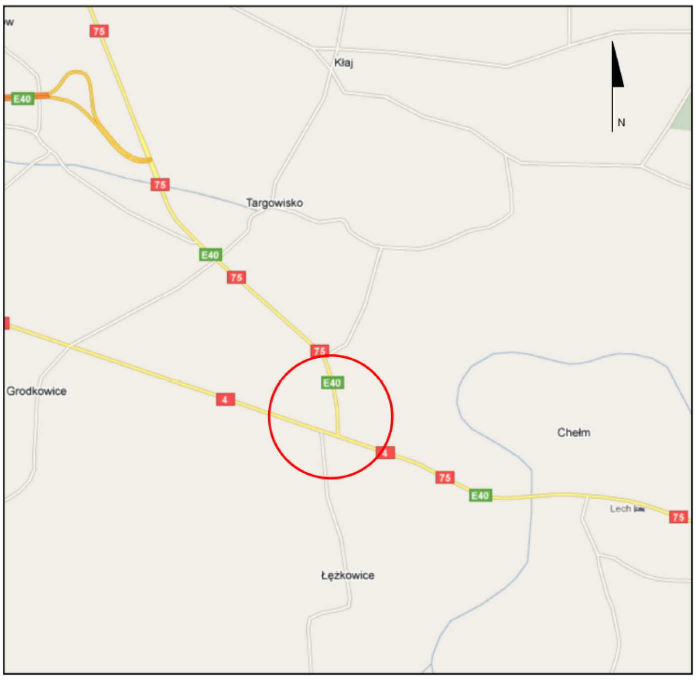


Figure 12 - Intersection location #3

Figure 13 - Router at #3

2.2.4. #4 Chełm

At #4 the router is placed in a lighting pole of signal group K2L and K2 above of east entry of road no 4.

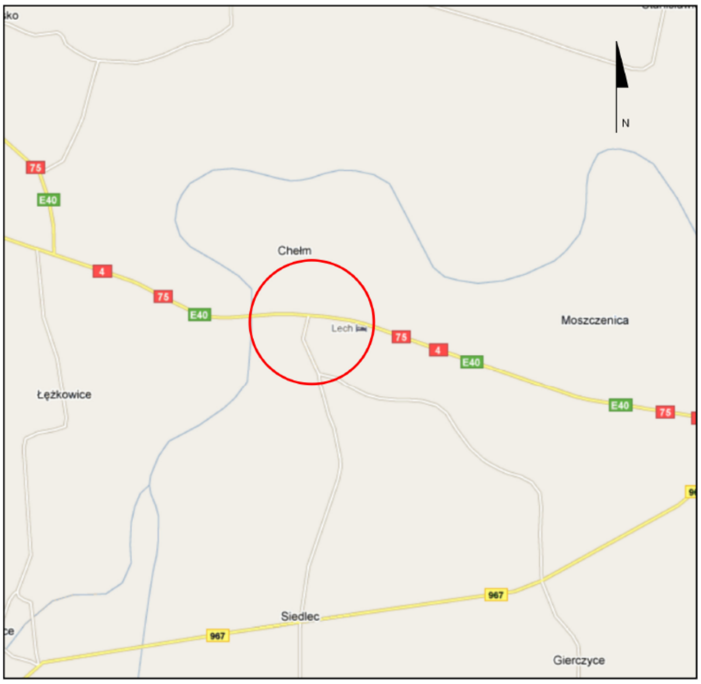


Figure 14 - Intersection location #4

Figure 15 - Router at #4

2.2.5. #5 Łapczyca

At #5 the router is placed in a lighting pole of signal group K2L and K2 above of east entry of road no 4.

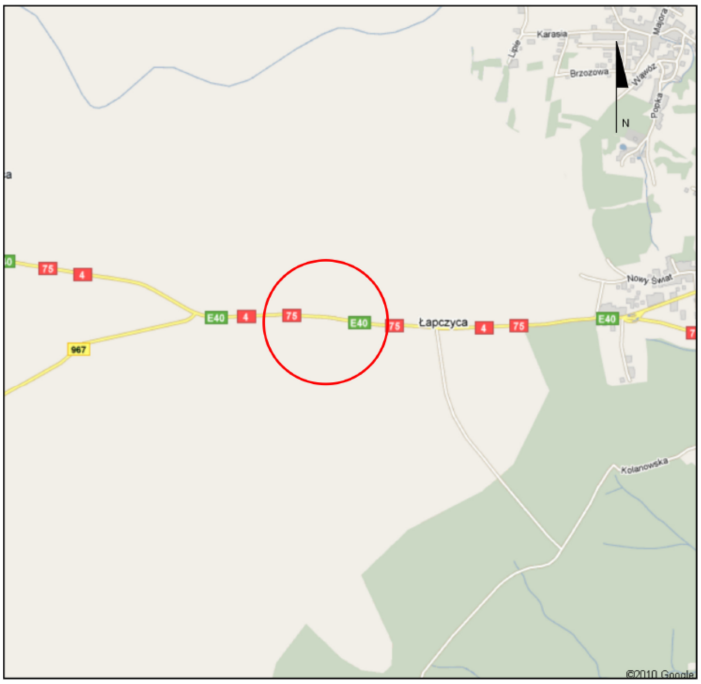


Figure 16 - Intersection location #5

Figure 17 - Router at #5

2.2.6. #6 Bochnia

At #6 the router is placed in a lighting pole of signal group K1 above of north entry (side road).

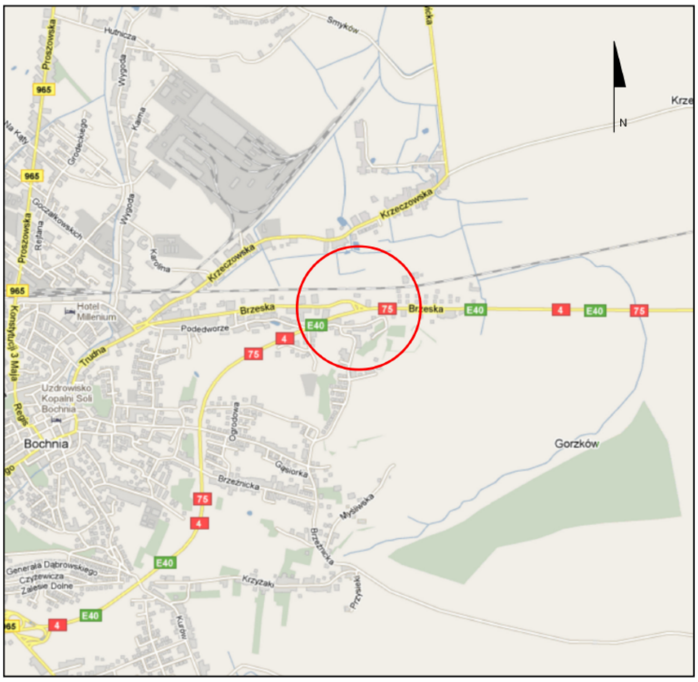


Figure 18 - Intersection location #6

Figure 19 - Router at #6

2.2.7. #7 Rzezawa

At #7 the router is placed in a lighting pole of signal group K1 above of north entry (side road).

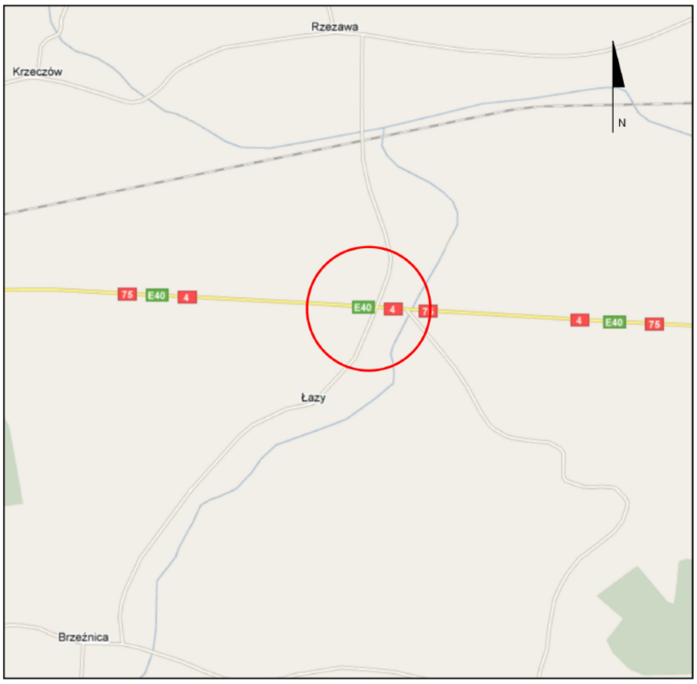


Figure 20 - Intersection location #7

Figure 21 - Router at #7

2.2.8. #8 Jasień

At #8 the router is placed in a lighting pole of signal group K1 above of north entry (side road).

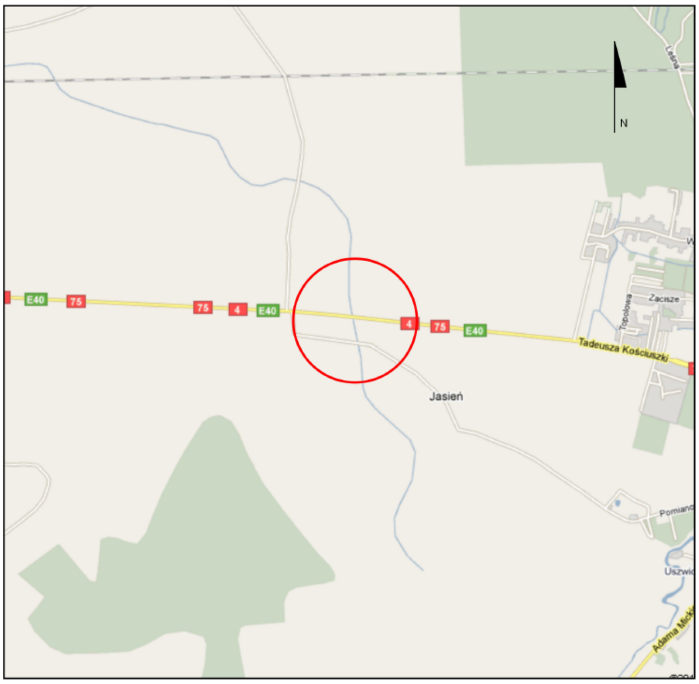


Figure 22 - Intersection location #8

Figure 23 - Router at #7

2.3. Vehicle installations

2.3.1. Trucks

In the trucks the FREILOT onboard unit is mounted to the windscreen with a suction cup. The power is connected to the truck's 24V supply.

Figure 24 - FREILOT OBU in a van den Broek truck (picture from Helmond)

In Krakow 5 buses from bus operator Szwagropol and 5 trucks from other operator are equipped with the onboard unit of the FREILOT Energy Efficient Intersection Control system.

Figure 25 - OBU mounted in a truck

3. Eco Driving Support

As these in-vehicle functionalities have similar implementation across the different pilot sites the explanation of System Components and Protocols and file formats will be presented in only one deliverable, D.FL.2.2 Lyon prototype. What is presented in this report is additional information on vehicle installations on the trucks in Krakow.

3.1. Vehicle installations in Krakow

We have selected 5 Trucks from the Temperi company. These trucks have a long haulage use.

K01 :	K02	K03	K06	K07
Eco Driving System	Eco Driving System	Eco Driving System	Eco Driving System	Eco Driving System
KR 763CY	KR 655CY	KR 864CR	KR 785CR	KR 195EA
B480115	B480027	B472542	B472551	B488545
Volvo FH	Volvo FH	Volvo FH	Volvo FH	Volvo FH
(No picture available)	(No picture available)	(No picture available)	(No picture available)	(No picture available)

The installations on the vehicles have been performed for two trucks at the beginning of May 2011 and for the three others at the beginning of June. Currently the vehicles are recording the baseline data.

3.2. System verification

The EDS system as such has been verified in Lyon. Since the system does not interfere with the truck software the process does not have to be repeated for all sites. However, the system has to be calibrated for each truck model once the system is installed on the truck in order to adjust parameters to the specifications of this type of trucks. The planning of the calibration is not easy as Temperi often sends the truck on missions that last for two weeks. At the moment the trucks are in baseline period and the lack of calibration is not an issue as long as it is done before the activation of the system.

3.3. Planning for the entire installations schedule for Krakow site

Figure 26 shows implementation planning for Krakow site. It shows that all the trucks have been recording baseline data since June 2011.

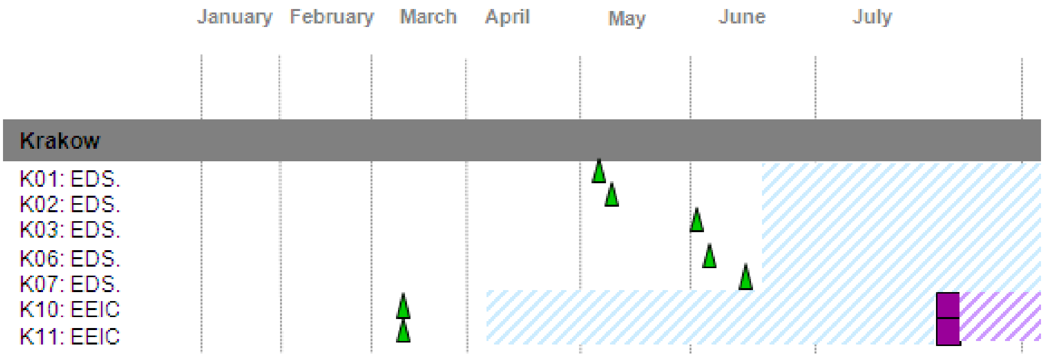


Figure 26 - The Krakow installation plan

4. Protocols & file formats-Energy Efficient Intersection Control

4.1. Router interface protocol

The Interface between the applications on the host and the router is defined in ASN.1 and encoded with the BER encoding. The messages are sent and received via a TCP/IP connection on port 1234.

```
CalmFast DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- API Message
RequestMessage ::= SEQUENCE {
  serial    INTEGER(0..65535),
  payload          CHOICE {
    open   [0]       OpenRequest,
    close  [1]       EmptyRequest,
    send   [2]       SendRequest,
    publish          [3]      PublishRequest,
    unpublish        [4]      EmptyRequest,
    monitor          [5]      EmptyRequest,
    unmonitor        [6]      EmptyRequest
  }
}

ReplyMessage ::= SEQUENCE {
  serial    INTEGER(0..65535),
  payload          CHOICE {
    openack          [0]      OpenReply,
    closeack         [1]      CloseReply,
    msg    [2]       MsgReply,
    service          [3]      MsgReply,
    fault  [4]       FaultReply
  }
}

OpenRequest ::= SEQUENCE {
   sid             INTEGER(0..4294967295)
}

EmptyRequest ::= SEQUENCE {
}

SendRequest ::= SEQUENCE {
   url             UrlString,
   data    ByteString
}

PublishRequest ::= SEQUENCE {
  data             ByteString,
  interval INTEGER(0..1000)
}

OpenReply ::= SEQUENCE {
  url              UrlString
}

CloseReply ::= SEQUENCE {
  data             INTEGER(0..255)
}

MsgReply ::= SEQUENCE {
  url              UrlString,
  data             ByteString
}
```

```
FaultReply ::= SEQUENCE {
  msg                  OCTET STRING
}

UrlString ::= OCTET STRING

ByteString ::= OCTET STRING

END
```

4.2. CAM protocol

Each cooperative system sends a beacon message twice per second. The beacon contains a so called Cooperative Awareness Message, in short CAM. Based on the received CAM messages a system can create a view of all systems in the neighbourhood. On the RSU this view is stored in the LDM.

The CAM's used in FREILOT are compatible with the CVIS and Safespot projects. The definition of the CAM message in mean time has been standardised by ETSI as ETSI TS 102 637-2, the FREILOT CAM messages do not yet comply to this new standard. The CAM messages are defined in ASN.1 and encoded using the BER encoding.

```
Interoperability DEFINITIONS IMPLICIT TAGS ::=
BEGIN

-- CAM Message
CooperativeAwarenessMessage ::= SEQUENCE {
  header               CamHeader,
  payload                        CamPayload                 -- Fixed + Optional Payload
}

-- CAM Header
CamHeader ::= SEQUENCE {
  sequenceNumber                 INTEGER(0..65535),
  nodeID               NodeID,
  nodeType                       NodeType,
  timestamp                      Timestamp,       -- This is the time of the message transmission
  position                       Position2D,
  positionVariance               PositionVariance,
  nodeSpeed                      SpeedModule,
  nodeHeading                    Heading,
  priority             Priority
}

-- CAM Payload
CamPayload ::= CHOICE {
  vehicleCamPayload              [0] VehicleCamPayload,
  rsuCamPayload                  [1] RsuCamPayload
}

VehicleCamPayload ::= SEQUENCE {
  vehicleElevation               Elevation,
  vehicleType                              VehicleType,
  vehicleSize                              VehicleSize,
  longitudinalAcceleration  LongitudinalAcceleration,
  yawRate                                  YawRate,
  exteriorLights                 ExteriorLights,
  accelerationControl            AccelerationControl,
  covarianceMatrix                         IncompleteCovarianceMatrix,
  numOfItemsInTaggedListINTEGER(0..32),
  taggedList                               CamTaggedList   OPTIONAL
}
```

```
RsuCamPayload ::= SEQUENCE {
  nearestRsu                  NearestRsu,
  temperature                 Temperature,
  visibilityRange             VisibilityRange,
  weather                     Weather,
  positionCovariance          PositionCovariance,
  numOfItemsInTaggedListINTEGER(0..32),
  taggedList                  CamTaggedList   OPTIONAL
}

-- CAM data definition
AccelerationControl ::= BIT STRING {
  brakePedal                  (0),      -- B'10000'
  gasPedalAct                 (1),      -- B'01000'
  cruiseControl               (2),      -- B'00100'
  adaptiveCruiseControl       (3),      -- B'00010'
  limiter                     (4)       -- B'00001'
}

CamTaggedList ::= OCTET STRING
-- The tagged list is not encoded using ASN.1, but according to SAFESPOT rules.
-- For a complete definition, please refer to SAFESPOT SP7 document
-- "Data format and messages"

Elevation ::= INTEGER(-5000..32767)
-- LSB=0.2m, with a 1Km negative offset
-- It represents the elevation respect to the sea level
-- Actual range (-1000..6534) m. The value +32767 means value not available

ExteriorLights ::= BIT STRING {
--  allLightsOff              ( ),        B'00000000'
  lowBeamHeadlightsOn         (0),      -- B'10000000'
  highBeamHeadlightsOn        (1),      -- B'01000000'
  leftTurnSignalOn            (2),      -- B'00100000'
  rightTurnSignalOn           (3),      -- B'00010000'
--  hazardSignalOn            ( ),        B'00110000'
  automaticLightControlOn     (4),      -- B'00001000'
  daytimeRunningLightsOn      (5),      -- B'00000100'
  fogLightOn                  (6),      -- B'00000010'
  parkingLightsOn             (7)       -- B'00000001'
}

Heading ::= INTEGER(0..65535)
-- LSB of 0.0054931640625 degrees
-- The current heading of the vehicle, expressed in units of 0.005493247 degrees
-- from North, such that 65535 such degrees represent (360degrees - 1 LSB)
-- North shall be defined as the axis defined by the WSG-84 coordinate system and
-- its reference ellipsoid. Increasing when turning counter-clockwise.

IncompleteCovarianceMatrix ::= SEQUENCE {
  ab        INTEGER(0..255),      -- Latitude-Longitude covariance
  ac        INTEGER(0..255),      -- Latitude-Heading covariance
  ad        INTEGER(0..255),      -- Latitude-Speed covariance
  af        INTEGER(0..255),      -- Latitude-YawRate covariance
  bc        INTEGER(0..255),      -- Longitude-Heading covariance
  bd        INTEGER(0..255),      -- Longitude-Speed covariance
  bf        INTEGER(0..255),      -- Longitude-YawRate covariance
  cc        INTEGER(0..255),      -- Heading variance
  cd        INTEGER(0..255),      -- Heading-Speed covariance
  cf        INTEGER(0..255),      -- Heading-YawRate covariance
  dd        INTEGER(0..255),      -- Speed variance
  df        INTEGER(0..255),      -- Speed-YawRate covariance
  ff        INTEGER(0..255)       -- YawRate variance
}
-- Incomplete covariance matrix are reported as a sequence of 13 bytes
-- described in Annex 1
```

```
LatitudePosition ::= INTEGER(-720000000..720000000)
-- LSB = 1/8 micro degree
-- Actual range: (-90..+90) degrees
-- Position of the geometrical centre of the object

LongitudePosition ::= INTEGER(-1440000000..1440000000)
-- LSB =1/8 micro degree
-- Actual range: (-180..+180) degrees
-- Position of the geometrical centre of the object

LongitudinalAcceleration ::= INTEGER(-2000..2000)
-- LSB units of 0.01 m/s^2
-- Actual range: +/- 20 m/s^2 (about +/- 2g)

NearestRsu ::= SEQUENCE OF Position2D
-- If unavailable, latitude and longitude are set to 0

NodeID ::= INTEGER
-- Indicates a unique identifier on VANET
-- Max possible value: 2^64-1 = 18446744073709551615

NodeType ::= ENUMERATED {
  reserved        (0),
  vehicle (1),
  rsu             (2)
}

Position2D ::= SEQUENCE {
  long2D LongitudePosition,
  lat2D   LatitudePosition
}

PositionCovariance ::= INTEGER(0..255)
-- Latitude-Longitude covariance as described in Annex 1

PositionVariance ::= SEQUENCE {
  aa              INTEGER(0..255),        -- Latitude variance
  bb              INTEGER(0..255)         -- Longitude variance
}
-- Position variances are reported as described in Annex 1

Priority ::= ENUMERATED {
  reserved                (0),
  cam                     (1),
  emergency               (2),
  high                    (3),
  medium                  (4),
  low                     (5)
}
-- VANET Message Priority

SpeedModule ::= INTEGER(-32768..32767)
-- Units of 0.01 m/s
-- Actual range (-327.68..327.67) m/s
-- Speed = 327.65 means no speed available
-- Negative values imply the vehicle in moving in reverse

Temperature ::= INTEGER(0..191)
-- in deg C with a +40 offset
-- Temperature=0 means -40°C, Temperature=140 means 100°C

Timestamp ::= SEQUENCE {
  seconds                 INTEGER(0..4294967295),         -- seconds since 1970/01/01
  milliseconds    INTEGER(0..999)                         -- milliseconds
}
-- The Unix UTC representation reduced to milliseconds resolution
```

```
VehicleAttribute ::= BIT STRING {
  safespot          (0),        -- B'1000'
  safeprobe         (1),        -- B'0100'
  cvis              (2),        -- B'0010'
  crashed           (3)         -- B'0001'
}

VehicleDescription ::= ENUMERATED {
  unknown                           (0),
  car                               (1),
  lightGoodsVehicle                 (2),
  heavyGoodsVehicle                 (3),
  publicTransportVehicle    (4),
  pedalCycle                        (5),
  emergencyVehicle                  (6),
  worksVehicle              (7),
  exceptionalSizeVehicle    (8),
  vehicleWithTrailer        (9),
  highSidedVehicle                  (10),
  minibus                           (11),
  taxi                              (12),
  tram                              (13),
  trolleyBus                        (14),
  train                     (15),
  postBus                           (16),
  schoolBus                         (17),
  militaryVvehicle          (18),
  motorcycle                        (19),
  sledge                    (20),
  assistanceVehicle                 (21),
  vehicle                   (31)
}

VehicleHeight ::= INTEGER(0..255)
-- LSB units of 0.05 m
-- The height of the vehicle excluding any antenna(s), expressed in units of 5cm
-- In case of vehicles with adjustable ride heights, camper shells, and other
-- devices which may cause the overall height to vary, the largest possible height
-- will be transmitted

VehicleLength ::= INTEGER(0..16383)
-- LSB units of 0.01 m
-- The length of the vehicle expressed in centimeters, unsigned

VehicleSize ::= SEQUENCE {
  width             VehicleWidth,
  length    VehicleLength,
  height    VehicleHeight
}

VehicleType ::= SEQUENCE {
  vehicleDescription                VehicleDescription,
  vehicleAttribute          VehicleAttribute
}

VehicleWidth ::= INTEGER(0..1023)
-- LSB units of 0.01 m
-- The width of the vehicle expressed in centimeters, unsigned

VisibilityRange ::= INTEGER(0..255)
-- LSB units of 5 m
-- Actual range: 0-1000 m
-- A value > 200 means data not available

Weather ::= ENUMERATED {
  weatherbug                                (0),
```


FREILOT - ENERGY EFFICIENT URBAN FREIGHT

```
  other                          (1),
  unknown                                    (2),
  noPrecipitation                (3),
  unidentifiedSlight             (4),
  unidentifiedModerate           (5),
  unidentifiedHeavy                          (6),
  snowSlight                                 (7),
  snowModerate                   (8),
  snowHeavy                                  (9),
  rainSlight                                 (10),
  rainModerate                   (11),
  rainHeavy                                  (12),
  frozenPrecipitationSlight  (13),
  frozenPrecipitationModerate    (14),
  frozenPrecipitationHeavy       (15)
}

YawRate ::= INTEGER(-32768..32767)
-- LSB units of 0.01 degrees/s
-- Actual range (-327.68..327.67)
-- YawRate = 327.67 means no yaw rate available
-- The YawRate of the vehicle, a signed value (positive when counterclockwise).
-- The YawRate Element reports the vehicle's rotation in degrees per second

END
```

4.3. Priority protocol

The OBU can request priority at the RSU of an intersection. The priority request and replies form the priority protocol. Priority messages are exchanged via CALM FAST datagrams and service announcements. The service identifier used is 42.

The Priority protocol is defined in ASN.1 and encoded using the BER encoding rules.

```
Priority DEFINITIONS ::= BEGIN

Intersection ::= SEQUENCE {
        nodeid          INTEGER,
        name            OCTET STRING,
        random          INTEGER(0..4294967295),
        arms            SEQUENCE OF IntersectionArm
}

IntersectionArm ::= SEQUENCE {
        heading         INTEGER (0..359),
        lanes           SEQUENCE OF Lane
}

Lane ::= SEQUENCE {
        stoplineid      INTEGER(0..65535),
        signalstate     SignalState,
        lanes           SEQUENCE OF LaneDirection
}

LaneDirection ::= ENUMERATED {
  ahead                 (1),
  righ                  (2),
  left                  (4),
  aheadright            (3),
  aheadleft             (5),
  aheadrightleft    (7),
  reverse           (0)
}
```

```
SignalState ::= ENUMERATED {
  uncontrolled      (0),
  green             (1),
  amber             (2),
  red               (3),
  redamber          (4),
  flashingamber     (5),
  dark              (6)
}

PrioRequest ::= SEQUENCE {
        nodeid        INTEGER,
        priority      INTEGER(0..9),
        stoplineid    INTEGER(0..65535),
        key           OCTET STRING
}

PrioResponse ::= SEQUENCE {
        stoplineid    INTEGER(0..65535),
        distance      INTEGER(-32768..32767),
        speedlimit    INTEGER(0..255),
        timetillgreen INTEGER(0..255),
        timetillred   INTEGER(0..255),
        speedadvice   INTEGER(0..255),
        status        PrioStatus,
        reason        PrioReason
}

PrioStatus ::= ENUMERATED {
        rejected        (0),
        accepted        (1),
        extendgreen     (2),
        priogreen       (3),
        emergencygreen  (4),
        finished        (5),
        error           (6)
}

PrioReason ::= ENUMERATED {
        none            (0),
        noservice       (1),
        speedlimit      (2),
        notauthorized   (3)
}

END
```

4.4. Log protocol

Vehicle logs are transmitted to the RSU via CALM FAST datagrams.

```
Logging DEFINITIONS ::= BEGIN

LogMessage ::= SEQUENCE {
        serial          INTEGER(0..65535),
        name            OCTET STRING,
        log             SEQUENCE OF LogLine
}

LogLine ::= SEQUENCE {
        timestamp               Timestamp,
        long2D          INTEGER(-1440000000..1440000000),
        lat2D           INTEGER(-720000000..720000000),
        heading         INTEGER (0..359),
        speed                   INTEGER (0..200),
        emergency               INTEGER (0..1),
        intersectionid  INTEGER,
        prioresponse            PrioResponse
}

LogAck ::= SEQUENCE {
        serial          INTEGER(0..65535)
}

-- The Unix UTC representation reduced to milliseconds resolution
Timestamp ::= SEQUENCE {
seconds         INTEGER(0..4294967295), -- seconds since 1970/01/01
milliseconds     INTEGER(0..999)      -- milliseconds
}

PrioResponse ::= SEQUENCE {
        stoplineid      INTEGER(0..65535),
        distance INTEGER(-32768, 32767),
        speedlimit      INTEGER(0, 255),
        timetillgreen   INTEGER(0, 255),
        timetillred     INTEGER(0, 255),
        speedadvice     INTEGER(0, 255),
        status          PrioStatus,
        reason          PrioReason
}

PrioStatus ::= ENUMERATED {
        rejected (0),
        accepted        (1),
        extendgreen     (2),
        priogreen       (3),
        emergencygreen (4),
        finished (5)
}

PrioReason ::= ENUMERATED {
        noservice       (0),
        speedlimit      (1),
        notauthorized   (2)
}

END
```

4.5. Log file format

The OBU samples vehicle data once per second. This data is kept in a FIFO store. When the vehicle is close to a RSU the log data is sent to the RSU to be archived.

Vehicle logs are stored on the RSU in a basic text format files. In a file each line contains one data sample. Each data sample contains a number of fields separated by semicolons. The following fields are used:

- A timestamp with millesecond resolution in `ssss.mmm` format,
- The URL of the vehicle address,
- The longitude in decimal format,
- The latitude in decimal format,
- The heading in degrees from North, counter clockwise,
- The actual speed in Km/h,
- The ID of the intersection where priority is requested (a number),
- Emergency vehicle (1) or not (0),
- The distance to the stop line in meters,
- The current speed advice received in Km/h,
- The current speed limit received, in Km/h
- The ID of the stop line (a number)
- The time until green in seconds (0 means not known),
- The time until red in seconds (0 means not known),
- The status (a number, see the Priority protocol in 4.3) ,
- The reason (a number, see the Priority protocol in 4.3).