**Иркутский национальный исследовательский технический университет**

**ОТЧЕТ**

**по диагностике автомобильной дороги**

**{DataSourceName}**

**на участке км {RoadBegin\_km} — км {RoadEnd\_km}**

**Утверждаю**

**Проректор ИРНИТУ,**

**профессор Пашков А.Е.**

**ИРКУТСК 2017**

Оглавление

[Пояснительная записка 2](#_Toc487034856)

[Введение 2](#_Toc487034857)

[Схема дороги 3](#_Toc487034858)

[Состав работ по диагностическому обследованию 3](#_Toc487034859)

[Технология проведения полевых работ 3](#_Toc487034860)

[Копии сертификатов и свидетельств о поверке 5](#_Toc487034861)

[Технология камеральных работ 11](#_Toc487034862)

[Определение интенсивности движения и состава транспортного потока 11](#_Toc487034863)

[Ведомости дорожных объектов 12](#_Toc487034864)

[Ведомость начала и конца дороги 12](#_Toc487034865)

[Определение геометрических параметров элементов автомобильной дороги 12](#_Toc487034866)

[Радиусы кривых в плане 12](#_Toc487034867)

[Продольные уклоны 13](#_Toc487034868)

[Путь, пройденный машиной по GPS (каждые 100 м) 14](#_Toc487034869)

[Характеристики дорожной одежды и покрытия 15](#_Toc487034870)

[Ширина проезжей части 16](#_Toc487034871)

[Искусственные сооружения 17](#_Toc487034872)

[???Ведомость мостов 17](#_Toc487034873)

[Ведомость труб 17](#_Toc487034874)

[Ведомость ограждений 18](#_Toc487034875)

[Ведомость направляющих устройств 19](#_Toc487034876)

[Пересечения и примыкания 22](#_Toc487034877)

[Обследование обустройства дороги 22](#_Toc487034878)

[Ведомость освещения 22](#_Toc487034879)

[???Ведомость остановок 24](#_Toc487034880)

[Ведомость придорожного сервиса 24](#_Toc487034881)

[Ведомость километровых знаков 24](#_Toc487034882)

[Ведомость дорожных знаков 26](#_Toc487034883)

[Дефектная ведомость 28](#_Toc487034884)

# Пояснительная записка

## Введение

ИРНИТУ выполнены работы по диагностике с формированием банка дорожных данных автомобильной дороги общего пользования «{DataSourceName}».

***Начало дороги***

***Конец дороги***

## Схема дороги

## Состав работ по диагностическому обследованию

### Технология проведения полевых работ

Для подготовки к выполнению работ по диагностике и оценке ТЭС ИРНИТУ были предоставлены технический паспорт, ведомости наличия и технического состояния искусственных сооружений автомобильной дороги.

Программой полевых работ предусматривалось следующее:

* Обследование участка автомобильной дороги и ее элементов для составления отчета по диагностике и оценки транспортно-эксплуатационного состояния с линейным графиком (по ОДН 218.0.006-2002);
* Определение геометрических параметров участка автомобильной дороги с регистрацией характерных точек;
* Оценка ровности покрытия проезжей части;
* Оценка прочности дорожной одежды;
* Определение коэффициента сцепления колеса с покрытием;
* Изучение интенсивности и состава движения;
* Обследование инженерного оборудования и обустройства;
* Видеосъемка с формированием видеобанка;
* Фотографирование труб, мостовых пролетов автопавильонов и состояния проезжей части.

**Правила проведения заездов дорожной лаборатории**

- Заезды начинались и заканчивались только из положения, когда автомобиль не двигается. Движение начиналось только после 30 секундного интервала после произведения старта лаборатории на запись заезда. Запись заезда останавливается только после 30 секунд после остановки машины.

- Движение задним ходом во время проведения заезда запрещено. (Движение задним ходом будет воспринято датчиком пути как движение вперед).

- Соблюдается ограничение скорости движения лаборатории 40 км/ч. Чем выше скорость движения, тем более размазанными в окне видео получаются трещины на дороге и другие объекты. Повышенные вибрации при движении с большей скоростью также негативно влияют на точность собираемых лабораторией данных.

- Работа во время дождя не проводилась.

Материалы диагностики автомобильной дороги обработаны и систематизированы в виде таблиц и графиков, на основании которых произведена оценка транспортно-эксплуатационного состояния, составлены ведомости наличия и технического состояния элементов автомобильной дороги.

Определение прочности нежестких дорожных одежд проводилось по методике ОДН 218.1.052-2002, на основании данных установкой динамического нагружения «Дина-3М», характерные участки определялись на основании дефектной ведомости.

Определение модуля упругости дорожной одежды производилось на участках дорог с дефектами покрытия (наличие колейности глубиной более 20-35 мм, густая сетка трещин, просадки (пучины), проломы, поперечные волны (сдвиги), частые выбоины (расстояние 1 - 4 м)) и на участках с продольной ровностью более предельно-допустимых показателей.

**Описание используемого оборудования, приборов и программ**

Работы по обследованию автомобильной дороги проводились с помощью видеокомплекса дорожного ДВК-5, оснащенного следующими приборами:

- Видео-съемочный комплекс. Количество видеопотоков реального времени – 8; Наличие выносных камерных головок в термо-кожухе; Размер кадра 1600х1200. Степень сжатия – 3000-10000 Кбит/сек. Время непрерывной записи – до 10 часов.

- Бортовой компьютер. Корпус – специальный, переносной, ударопрочный, защищенный, процессор – двухъядерный 1.66 ГГц, оперативная память 2Gb, жесткий диск 2.5" 350Gb с блоком горячей замены накопителей 2х2.5", питание от бортовой сети – 12В, монитор 12.1" цветной TFT LCD с сенсорным экраном.

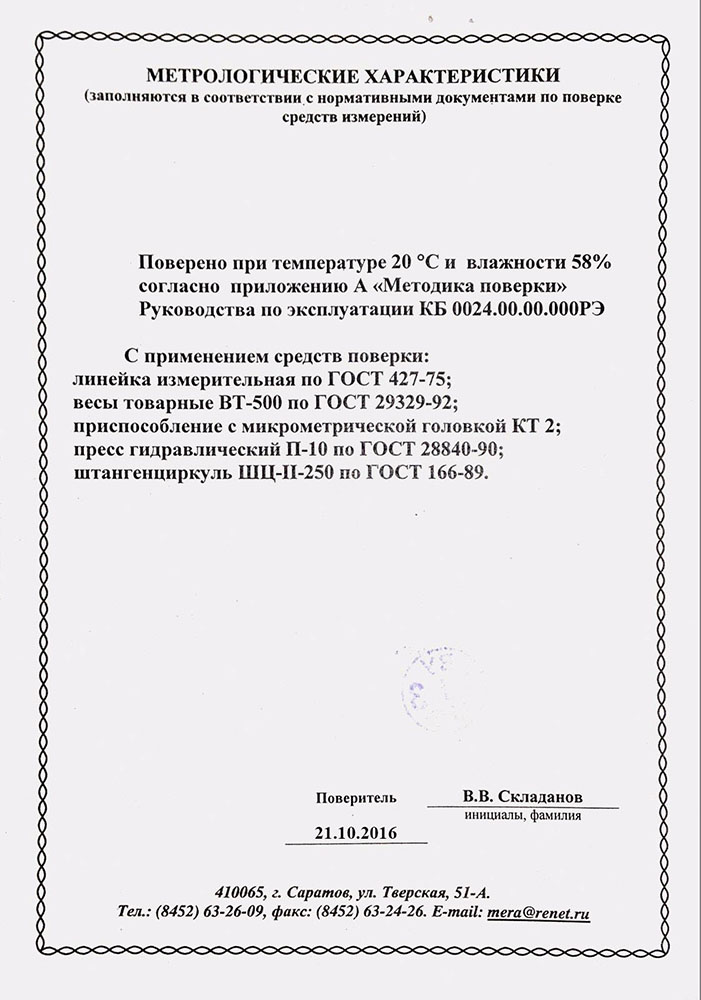
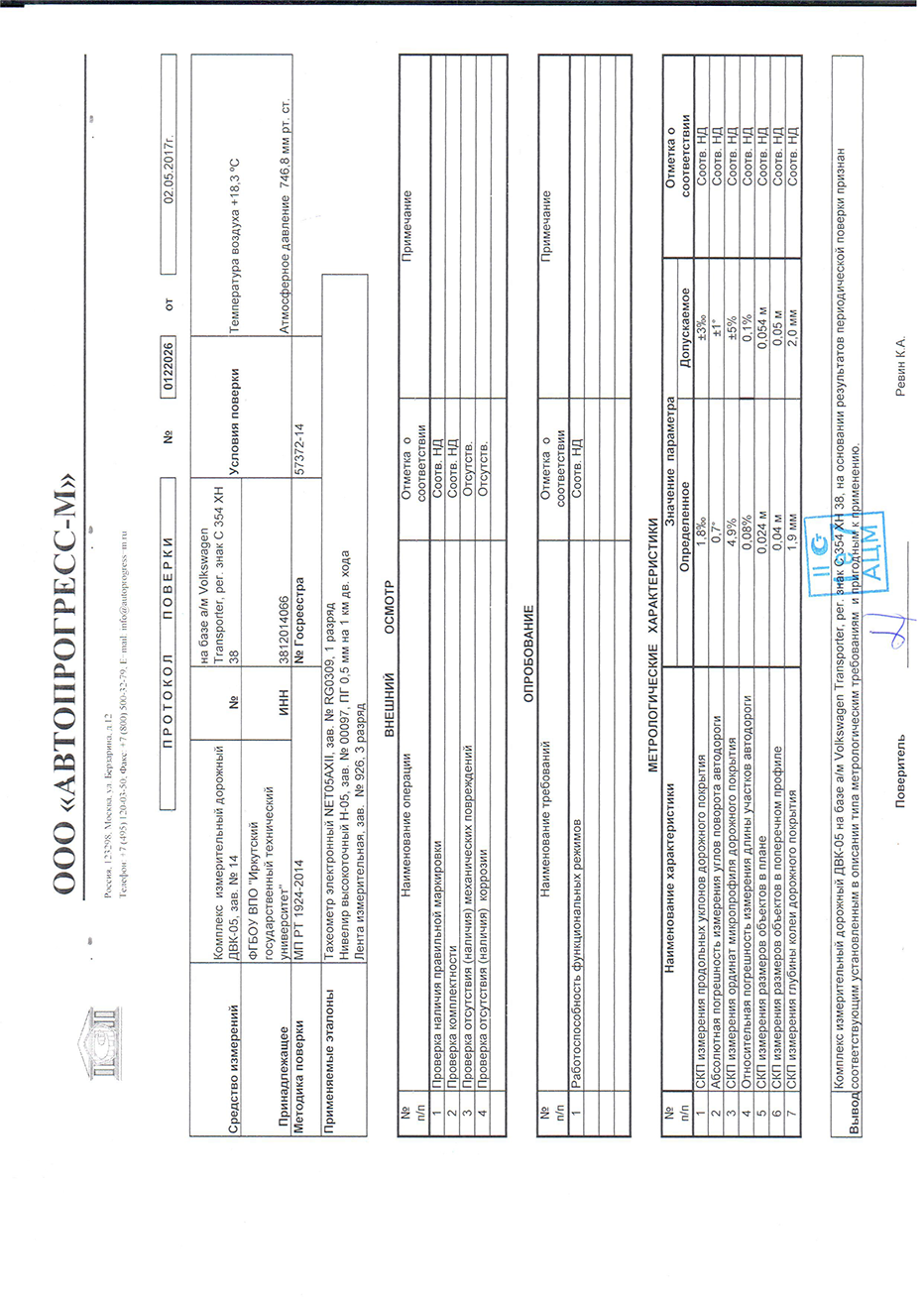
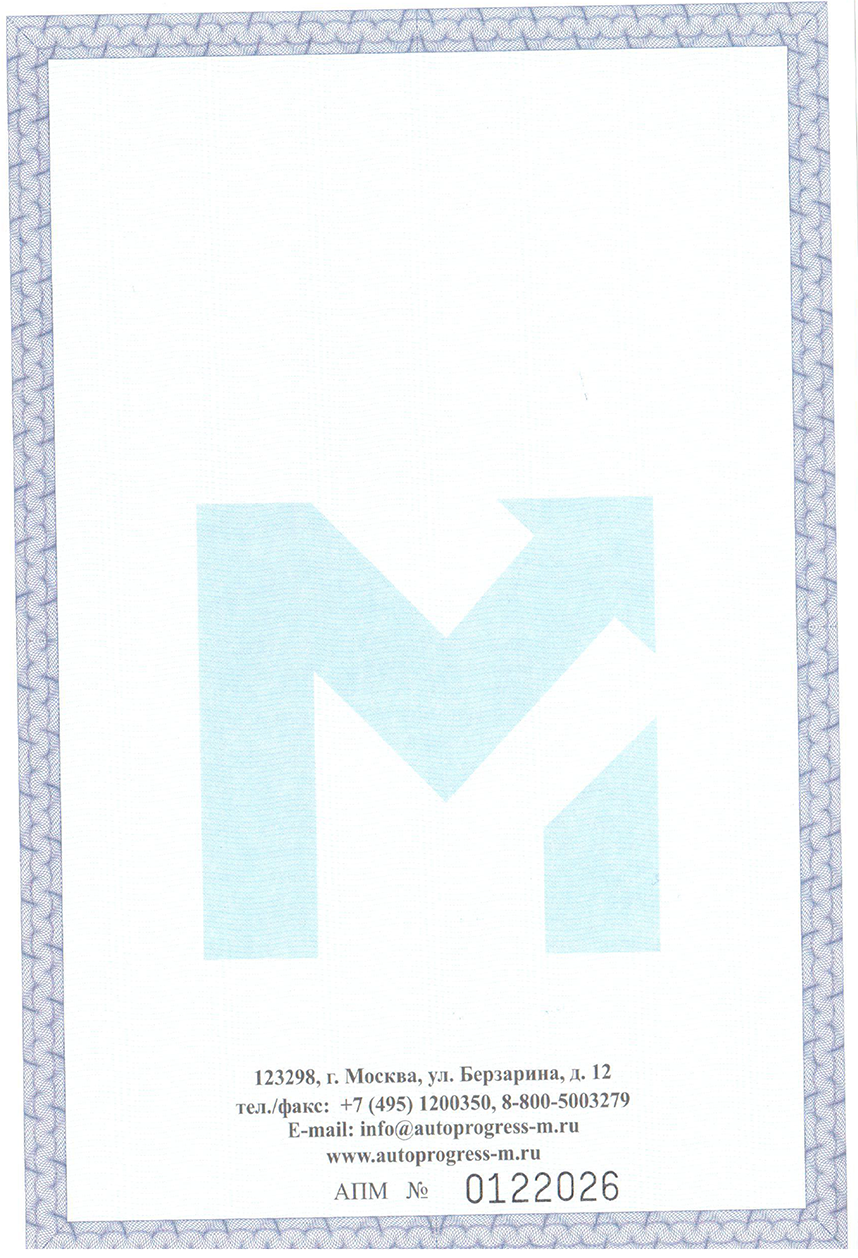
Оценка прочности дорожной одежды выполнялась установкой динамического нагружения «Дина-3М».Также в ходе обследования были использованы: нивелир TOPCON, рейка дорожная универсальная «КОНДОР», курвиметр, мерные ленты и рулетки.

При диагностике, была выполнена видеосъемка, позволяющая в камеральных условиях оценить состояние различных элементов автомобильной дороги. Обработка и хранение результатов измерений при диагностике дороги, расчеты и формирование отчетных документов, выполнено в программном продукте «Система мониторинга и поддержка управления сетью автомобильных дорог».

Оценка транспортно-эксплуатационного состояния автомобильной дороги выполнена в соответствии с методикой ОДН 218.0.006-2002.

Для расчета основных показателей транспортно-эксплуатационного состояния участка автодороги и вывода их в графическом виде использован программный комплекc «Система мониторинга и поддержки управления дорожной сетью».

### Копии сертификатов и свидетельств о поверке



### Технология камеральных работ

Основным средством камеральной обработки данных полевых работ является программный комплекс СВПД (разработчик НПО «Регион», г.Москва), а также программное обеспечение, разработанное на кафедре автомобильных дорог ИРНИТУ.

Программный комплекс СВПД обеспечивает применение фотограммометрических методов обработки видеорядов и обработку облака точек профилеметра (лазерный сканер) и создание ряда ведомостей.

Последовательность действий по внесению информации в базу данных после проведения заездов следующая:

С помощью программы DataManager считать информацию из файлов заездов прямых и обратных направлений, сделать расчет траектории и углов наклона камеры, и занести полученную информацию в базу данных;

В программе SVPD расставить необходимое количество реперов для сшивки заездов прямого и обратного направлений, а также по всей длине заездов “общелкать” осевую линию;

Программой SurveyMerger\_2D “сшить” заезды прямого и обратного направлений;

После “сшивки” заездов оператор программы SVPD может “общелкивать” все интересующие пользователя объекты в “сшитых” заездах.

Программное обеспечение, разработанное на кафедре автомобильных дорог ИРНИТУ, поддерживает создание и редактирование электронных ситуационных планов, формирование итоговых документов.

### Определение интенсивности движения и состава транспортного потока

Продолжительность учёта в среднем составляла 4 часа в сутки.

Выбор учётных пунктов основывался на возможности учёта транспортных потоков в сечении основной автомобильной дороги, примыканиях подъездов от неё.

Количественный учёт движения проводился с 9 часов утра до 12 часов дня и с 14 часов дня до 18 часов вечера.

Среднесуточная интенсивность движения определена в соответствии с ОДМ 218.4.005 –2010, утверждёнными от 12.01.2011 г. №13-р.

Изменения интенсивности движения по часам суток, дням недели и месяцам учтены коэффициентами неравномерности движения.

Среднесуточная интенсивность движения определена в соответствии с ОДМ 218.4.005 –2010, утверждёнными от 12.01.2011 г. №13-р.

Изменения интенсивности движения по часам суток, дням недели и месяцам учтены коэффициентами неравномерности движения.

Суточная среднегодовая интенсивность движения через часовую по результатам наблюдений определена по формуле:

*4Nч*

*Ncут*= -----------------------, авт/сутки

*Kt х KhxKrx 365*

где Kt,Kh, Kr – коэффициенты неравномерности движения, соответственно по часам суток (0,05; 0,06), дням недели (0,13;0,145), месяца года (0,091).

Развёрнутая характеристика среднесуточной интенсивности движения по характерным перегонам дороги в существующих условиях и на расчётный периоды приведена в сводной ведомости интенсивности движения.

# Ведомости дорожных объектов

## ???Ведомость начала и конца дороги

|  |  |  |  |
| --- | --- | --- | --- |
| Начало, км | Конец, км | Линейная протяженность, км | Идетнификационный номер |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(LP.StartPos), dbo.pp\_Km\_mFormat(LP.EndPos), convert(numeric(10,3), round((LP.EndPos - LP.StartPos)/1000.0, 3)),  '?'  FROM ListRoadParts LP  JOIN ListRoads R ON R.id\_ = LP.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = LP.NumDataSource  WHERE LP.NumRoad = @NumRoad  and LP.NumDataSource = @NumDataSource  and LP.NumPartType = 33 | | | |

## 

## Определение геометрических параметров элементов автомобильной дороги

### Радиусы кривых в плане

| Участок | Начало | Конец | Длина,м | Радиус | Начало | | Конец | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT case  when sign(LRP.value2) > 0 then rtrim(c.FullTitle)+' правого поворота'  when sign(LRP.value2) < 0 then rtrim(c.FullTitle)+' левого поворота'  else rtrim(C.FullTitle)  end [Участок],  dbo.pp\_Km\_mFormat(LRP.StartPos) [Начало], dbo.pp\_Km\_mFormat(LRP.EndPos) [Конец],  convert(int, (LRP.EndPos - LRP.StartPos)) [Длина, м],  ISNULL(convert(varchar(10), convert(int, NULLIF(abs(LRP.value2), 0))), '') [Радиус],  RSL1.Longitude,  RSL1.Latitude,  RSL2.Longitude,  RSL2.Latitude  FROM ListRoadParts LRP  JOIN dbo.RoadSLocation RSL1 ON RSL1.Displacement = round(LRP.StartPos, 0) and RSL1.NumDataSource = @NumDataSource  JOIN dbo.RoadSLocation RSL2 ON RSL2.Displacement = round(LRP.EndPos, 0) and RSL2.NumDataSource = @NumDataSource  LEFT JOIN Classifier c on c.id\_ = LRP.reference  LEFT JOIN RoadGeometry G on G.Displacement between LRP.StartPos and LRP.EndPos  and G.NumRoad = LRP.NumRoad  and G.NumDataSource = LRP.NumDataSource  WHERE LRP.NumPartType = 34  and LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  and LRP.reference = 2385045  GROUP BY LRP.StartPos, LRP.EndPos, LRP.Reference, C.FullTitle, LRP.value2, LRP.reference2, LRP.value4, RSL1.Longitude, RSL1.Latitude, RSL2.Longitude, RSL2.Latitude  ORDER BY LRP.StartPos | | | | | | | | |

### Продольные уклоны

| Начало | Конец | Продольный уклон | Начало | | Конец | |
| --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(LRP.StartPos) [Начало], dbo.pp\_Km\_mFormat(LRP.EndPos) [Конец],  LRP.Value1 [Продольный уклон],  RSL1.Longitude,  RSL1.Latitude,  RSL2.Longitude,  RSL2.Latitude  FROM ListRoadParts LRP  JOIN dbo.RoadSLocation RSL1 ON RSL1.Displacement = round(LRP.StartPos, 0) and RSL1.NumDataSource = @NumDataSource  JOIN dbo.RoadSLocation RSL2 ON RSL2.Displacement = round(LRP.EndPos, 0) and RSL2.NumDataSource = @NumDataSource  LEFT JOIN Classifier С ON С.id\_ = NULLIF(LRP.Reference, 0)  WHERE LRP.NumRoad = @NumRoad  and LRP.NumDataSource = @NumDataSource  and LRP.NumPartType = 32  ORDER BY LRP.StartPos | | | | | | |

### 

### Путь, пройденный машиной по GPS (каждые 100 м)

| № | Широта | Долгота | Местоположение |
| --- | --- | --- | --- |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT ROW\_NUMBER()OVER(ORDER BY RSL.Displacement), RSL.Latitude, RSL.Longitude, dbo.pp\_Km\_mFormat(RSL.Displacement)  FROM RoadSLocation RSL  WHERE RSL.NumRoad = @NumRoad and RSL.Displacement % 100 = 0  and RSL.NumDataSource = @NumDataSource | | | |

## Характеристики дорожной одежды и покрытия

| Начало участка, км | Конец участка, км | Начало участка, № столба + м | Конец участка, № столба + м | Конструкция дорожной одежды | Начало участка | | Конец участка | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*/4019/\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT  dbo.pp\_Km\_mFormat(SF.StartPos), dbo.pp\_Km\_mFormat(SF.EndPos),  ISNULL(dbo.ppSignalColumnDisplacement(@NumRoad, @NumDataSource, SF.StartPos), ''),  ISNULL(dbo.ppSignalColumnDisplacement(@NumRoad, @NumDataSource, SF.EndPos), ''),  RTRIM(CSF.FullTitle),  RSL1.Latitude, RSL1.Longitude, RSL2.Latitude, RSL2.Longitude  FROM ListRoadParts SF  JOIN ListRoads R ON R.id\_ = SF.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = SF.NumDataSource  JOIN Classifier CSF ON CSF.id\_ = SF.Reference  JOIN RoadSLocation RSL1 ON RSL1.NumDataSource = SF.NumDataSource and RSL1.Displacement = ROUND(SF.StartPos, 0)  JOIN RoadSLocation RSL2 ON RSL2.NumDataSource = SF.NumDataSource and RSL2.Displacement = ROUND(SF.EndPos, 0)  WHERE SF.NumRoad = @NumRoad  and SF.NumDataSource = @NumDataSource  and SF.NumPartType = 13 | | | | | | | | |

## Ширина проезжей части

| Начало участка, км | Конец участка, км | Ширина проезжей части | Начало участка | | Конец участка | |
| --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT  dbo.pp\_Km\_mFormat(W.StartPos), dbo.pp\_Km\_mFormat(W.EndPos),  W.Value2,  RSL1.Latitude, RSL1.Longitude, RSL2.Latitude, RSL2.Longitude  FROM ListRoadParts W  JOIN ListRoads R ON R.id\_ = W.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = W.NumDataSource  JOIN RoadSLocation RSL1 ON RSL1.NumDataSource = W.NumDataSource and RSL1.Displacement = ROUND(W.StartPos, 0)  JOIN RoadSLocation RSL2 ON RSL2.NumDataSource = W.NumDataSource and RSL2.Displacement = ROUND(W.EndPos, 0)  WHERE W.NumRoad = @NumRoad  and W.NumDataSource = @NumDataSource  and W.NumPartType = 14  ORDER BY W.StartPos | | | | | | |

## Искусственные сооружения

### ???Ведомость мостов

### Ведомость труб

| Местоположением, км | Местоположением, № столба + км | Расположение | Нахождение на балансе управления автодорог | Широта | Долгота |
| --- | --- | --- | --- | --- | --- |
|
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(T.Displacement),  ISNULL(dbo.ppSignalColumnDisplacement(@NumRoad, @NumDataSource, T.Displacement), ''),  rtrim(CPL.FullTitle),  'V',  RSL.Latitude, RSL.Longitude  FROM ListTubes T  JOIN ListRoads R ON R.id\_ = T.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = T.NumDataSource  JOIN RoadSLocation RSL ON RSL.NumDataSource = T.NumDataSource and RSL.Displacement = ROUND(T.Displacement, 0)  LEFT JOIN Classifier CPL ON CPL.id\_ = T.NumPlace  LEFT JOIN ListMaterials m ON m.id\_ = t.NumMaterial  LEFT JOIN Classifier c ON c.id\_ = t.NumCondition  LEFT JOIN Classifier c2 ON c2.id\_ = t.NumTypeTubeCuttingShape  WHERE t.NumDataSource = @NumDataSource  and t.NumRoad = @NumRoad  ORDER BY t.Displacement | | | | | |

### Ведомость ограждений

| Начало | Конец | Слева | Справа | Тип | Материал | Начало участка | | Конец участка | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(B.StartPos) [Начало], dbo.pp\_Km\_mFormat(B.EndPos) [Конец],  convert(int, (case when B.NumPlace = 257 then B.EndPos - B.StartPos else 0 end)) [Слева, м],  convert(int, (case when B.NumPlace = 258 then B.EndPos - B.StartPos else 0 end)) [Справа, м],  rtrim(CBc.FullTitle) [Тип],  rtrim(CMt.FullTitle) [Материал],  RSL1.Longitude,  RSL1.Latitude,  RSL2.Longitude,  RSL2.Latitude  FROM ListBarriers B  JOIN dbo.RoadSLocation RSL1 ON RSL1.Displacement = round(B.StartPos, 0) and RSL1.NumDataSource = @NumDataSource  JOIN dbo.RoadSLocation RSL2 ON RSL2.Displacement = round(B.EndPos, 0) and RSL2.NumDataSource = @NumDataSource  JOIN dbo.Classifier CPl ON Cpl.id\_ = B.NumPlace  LEFT JOIN dbo.Classifier CBc ON CBc.id\_ = B.NumBarrierConstruction  LEFT JOIN dbo.ListMaterials CMt ON CMt.id\_ = B.NumMaterial  LEFT JOIN dbo.Classifier CC ON CC.id\_ = NULLIF(B.NumCondition, 0)  WHERE B.NumRoad = @NumRoad  and B.NumDataSource = @NumDataSource  and B.NumTypeGuidingStructure = 0  ORDER BY B.StartPos | | | | | | | | | |

### Ведомость направляющих устройств

| Начало | Конец | Длина | Слева, шт | Справа, шт | Начало участка | | Конец участка | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT \*  INTO #Barriers  FROM ListBarriers  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  and NumTypeGuidingStructure = 181  ORDER BY StartPos  SELECT new.id\_,  max(case  when LRA.id\_ is not NULL then 'Примыкание'  when LT.id\_ is not NULL then 'Труба'  else 'Насыпь'  end) zone  INTO #Zones  FROM #Barriers new  LEFT JOIN ListRoadAttachments LRA ON new.EndPos - new.StartPos < 80  and LRA.NumDataSource = @NumDataSource  and (LRA.StartPos - 5 between new.StartPos and new.EndPos or LRA.EndPos + 5 between new.StartPos and new.EndPos)  and LRA.NumPlace = new.NumPlace  LEFT JOIN ListTubes LT ON new.EndPos - new.StartPos < 65  and LT.NumDataSource = @NumDataSource  and LT.NumPlace = 262  and LT.Displacement between new.StartPos and new.EndPos  GROUP BY new.id\_  SELECT dbo.pp\_Km\_mFormat(round(new.StartPos, 0)) [Начало], dbo.pp\_Km\_mFormat(round(new.EndPos, 0)) [Конец],  round(new.EndPos, 0) - round(new.StartPos, 0) [Длина],  (case when new.NumPlace = 257 then new.GuidingStructuresCount else 0 end) [Слева, шт],  (case when new.NumPlace = 258 then new.GuidingStructuresCount else 0 end) [Справа, шт],  Z.Zone [Зона расположения],  RSL1.Longitude,  RSL1.Latitude,  RSL2.Longitude,  RSL2.Latitude  FROM #Barriers new  JOIN dbo.RoadSLocation RSL1 ON RSL1.Displacement = round(new.StartPos, 0) and RSL1.NumDataSource = @NumDataSource  JOIN dbo.RoadSLocation RSL2 ON RSL2.Displacement = round(new.EndPos, 0) and RSL2.NumDataSource = @NumDataSource  JOIN #Zones Z ON Z.id\_ = new.id\_  JOIN Classifier C31 ON C31.id\_ = new.NumPlace  LEFT JOIN Classifier CC ON CC.id\_= NULLIF(new.NumCondition, 0)  ORDER BY round(new.StartPos, 0)    DROP TABLE #Zones  DROP TABLE #Barriers | | | | | | | | |

## Пересечения и примыкания

| Местоположение, км | Расположение | Название | Угол примыкания, градусы | Покрытие | Широта | Долгота |
| --- | --- | --- | --- | --- | --- | --- |
| Тип |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(LRA.StartPos + (LRA.EndPos - LRA.StartPos)/2), CP.Fulltitle, LRA.AttachmentName, convert(int, LRA.CrossingCorner),  (case when LRA.NumTech = 193 then 'асфальтобетон' else 'переходное' end),  RSL.Latitude, RSL.Longitude  FROM ListRoadAttachments LRA  JOIN ListRoads R ON R.id\_ = LRA.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = LRA.NumDataSource  JOIN Classifier CP ON CP.id\_ = LRA.NumPlace  JOIN RoadSLocation RSL ON RSL.NumDataSource = LRA.NumDataSource and RSL.Displacement = ROUND(LRA.StartPos + (LRA.EndPos - LRA.StartPos)/2.0, 0)  WHERE LRA.NumRoad = @NumRoad  and LRA.NumDataSource = @NumDataSource  ORDER BY LRA.StartPos | | | | | | |

## Обследование обустройства дороги

### Ведомость освещения

| Начало участка, км | Конец участка, км | Длина участка, км | Расположение | Начало участка | | Конец участка | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Широта | Долгота | Широта | Долгота |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  DECLARE @MinDistance int  SELECT @MinDistance = 55  DECLARE @Displacement int, @NumPlace int, @LastDisplacement int, @LastNumPlace int  CREATE TABLE #LampsParts (StartPos int, EndPos int, NumPlace int, NumLamps int)  DECLARE LAMPS\_CUR CURSOR FOR  SELECT L.Displacement, L.NumPlace  FROM ListRoadLamps L  WHERE L.NumRoad = @NumRoad  and L.NumDataSource = @NumDataSource  ORDER BY L.Displacement  OPEN LAMPS\_CUR  FETCH NEXT FROM LAMPS\_CUR INTO @Displacement, @NumPlace  INSERT INTO #LampsParts (StartPos, EndPos, NumPlace, NumLamps)  VALUES (@Displacement, @Displacement, @NumPlace, 1)    WHILE @@FETCH\_STATUS = 0  BEGIN  SELECT TOP 1 @LastDisplacement = EndPos, @LastNumPlace = NumPlace FROM #LampsParts ORDER BY EndPos desc    IF @LastNumPlace = @NumPlace and @Displacement - @LastDisplacement <= @MinDistance  UPDATE #LampsParts SET EndPos = @Displacement, NumLamps = NumLamps + 1 WHERE EndPos = @LastDisplacement  ELSE  INSERT INTO #LampsParts (StartPos, EndPos, NumPlace, NumLamps)  VALUES (@Displacement, @Displacement, @NumPlace, 1)  FETCH NEXT FROM LAMPS\_CUR INTO @Displacement, @NumPlace  END  CLOSE LAMPS\_CUR  DEALLOCATE LAMPS\_CUR  SELECT rtrim(R.ShortTitle), rtrim(R.FullTitle) + REPLACE(LDS.FullTitle, 'Агломерация 2017', ''),  dbo.pp\_Km\_mFormat(L.StartPos), dbo.pp\_Km\_mFormat(L.EndPos), (L.EndPos - L.StartPos) / 1000.0,  rtrim(CPL.FullTitle),  RSL1.Latitude, RSL1.Longitude, RSL2.Latitude, RSL2.Longitude  FROM #LampsParts L  JOIN ListRoads R ON R.id\_ = @NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = @NumDataSource  LEFT JOIN dbo.Classifier CPL ON CPL.id\_ = L.NumPlace  JOIN RoadSLocation RSL1 ON RSL1.NumDataSource = @NumDataSource and RSL1.Displacement = ROUND(L.StartPos, 0)  JOIN RoadSLocation RSL2 ON RSL2.NumDataSource = @NumDataSource and RSL2.Displacement = ROUND(L.EndPos, 0)  ORDER BY L.StartPos  DROP TABLE #LampsParts | | | | | | | |

### ???Ведомость остановок

### Ведомость придорожного сервиса

| Местоположение, км | Местоположение, № столба + км | Расположение | Нахождение на балансе управления автодорог | Широта | Долгота |
| --- | --- | --- | --- | --- | --- |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT dbo.pp\_Km\_mFormat(S.Displacement), dbo.ppSignalColumnDisplacement(S.NumRoad, S.NumDataSource, S.Displacement),  rtrim(CPL.FullTitle), rtrim(S.ServiceName),  RSL.Latitude, RSL.Longitude  FROM ListRoadSideService S  JOIN ListRoads R ON R.id\_ = S.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = S.NumDataSource  JOIN RoadSLocation RSL ON RSL.NumDataSource = S.NumDataSource and RSL.Displacement = ROUND(S.Displacement, 0)  LEFT JOIN dbo.Classifier CPL ON CPL.id\_ = S.NumPlace  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  ORDER BY S.Displacement | | | | | |

### Ведомость километровых знаков

| Местоположение начального столба, км | № столба | | Расстояние между столбами, м | Широта | Долгота |
| --- | --- | --- | --- | --- | --- |
| Начальный | Конечный |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT \*  INTO #KMs  FROM ListRoadSigns S  WHERE S.NumRoad = @NumRoad  and S.NumDataSource = @NumDataSource  and S.NumRoadSign = 43752  CREATE TABLE #OrderedKMS (id int identity(1,1), id\_ int)  INSERT INTO #OrderedKMS (id\_)  SELECT id\_  FROM #KMs  ORDER BY Displacement    SELECT dbo.pp\_Km\_mFormat(KM1.Displacement), KM1.SignLabel, ISNULL(KM2.SignLabel, ''), ISNULL(convert(int, round(KM2.Displacement - KM1.Displacement, 0)), 0),  RSL.Latitude, RSL.Longitude  FROM #OrderedKMS T1  JOIN #KMs KM1 ON KM1.id\_ = T1.Id\_  JOIN ListRoads R ON R.id\_ = KM1.NumRoad  JOIN ListDataSources LDS ON LDS.id\_ = KM1.NumDataSource  JOIN RoadSLocation RSL ON RSL.NumDataSource = KM1.NumDataSource and RSL.Displacement = ROUND(KM1.Displacement, 0)  LEFT JOIN #OrderedKMS T2 ON T2.id = T1.id + 1  LEFT JOIN #KMs KM2 ON KM2.id\_ = T2.Id\_  ORDER BY KM1.Displacement  DROP TABLE #KMs  DROP TABLE #OrderedKMS | | | | | |

### Ведомость дорожных знаков

| Тип знака | Номер по ГОСТ | [Наименование] | Адрес | Прямое | Обратное | Широта | Долгота |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT '1' Code, 'Предупреждающие' Type INTO #SignTypes UNION  SELECT '2' Code, 'Знаки приоритета' Type UNION  SELECT '3' Code, 'Запрещающие' Type UNION  SELECT '4' Code, 'Предписывающие' Type UNION  SELECT '5' Code, 'Знаки особых предписаний' Type UNION  SELECT '6' Code, 'Информационные' Type UNION  SELECT '7' Code, 'Знаки сервиса' Type UNION  SELECT '8' Code, 'Знаки доп. информации' Type  SELECT \*  INTO #Signs  FROM ListRoadSigns  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  ORDER BY Displacement  SELECT  ST.Type [Тип знака], rtrim(C48.ShortTitle) [Номер по ГОСТ], rtrim(C48.FullTitle) + (case new.SignLabel when '' then '' else (' ('+ ltrim(rtrim(convert(varchar(100),new.SignLabel))) +')') end) [Наименование],  dbo.pp\_Km\_mFormat(round(new.Displacement, 0)) [Адрес],  (case when new.NumMoveDirection = 246 then rtrim(Cpl.FullTitle) else '' end) [Прямое],  (case when new.NumMoveDirection = 247 then rtrim(Cpl.FullTitle) else '' end) [Обратное],  RSL.Latitude,  RSL.Longitude  FROM #Signs new  JOIN dbo.RoadSLocation RSL ON RSL.Displacement = round(new.Displacement, 0) and RSL.NumDataSource = @NumDataSource  JOIN Classifier C48 ON C48.id\_ = new.NumRoadSign  LEFT JOIN #SignTypes ST ON ST.Code = LEFT(C48.ShortTitle, 1)  LEFT JOIN Classifier Cpl ON Cpl.id\_ = new.NumPlace  LEFT JOIN Classifier CC ON CC.id\_ = NULLIF(new.NumCondition, 0)  ORDER BY LEFT(C48.ShortTitle, 1), round(new.Displacement, 0)  DROP TABLE #Signs  DROP TABLE #SignTypes | | | | | | | |

## Дефектная ведомость

| Начало | Конец | Выбоины покрытия | Колейность | Съезды | Разрушение покрытия | Неровности |
| --- | --- | --- | --- | --- | --- | --- |
| **[CONTENT][REPLACE\_TABLE\_ON\_EMPTY:отсутствуют]**  DECLARE @NumRoad int, @NumDataSource int  SELECT @NumDataSource=/\*$NumDataSource\*//\*$\*/  SELECT @NumRoad = NumRoad  FROM ListDataSources  WHERE id\_ = @NumDataSource  SELECT R.Defect\_id, rtrim(LD.FullTitle) DefectTitle, R.StartPos, R.EndPos,  P.Param\_id, P.ParamVal  INTO #Defects  FROM av\_Defect\_Registry R  JOIN av\_DefRegParams P ON P.DefReg\_id = R.id\_  JOIN av\_ListDefects LD ON LD.id\_ = R.Defect\_id  WHERE NumRoad = @NumRoad  and NumDataSource = @NumDataSource  and P.Param\_id in (30, 31)  DECLARE @step int, @StartPos int, @EndPos int, @SP int, @EP int  SET @step = 100  SELECT @StartPos = R.StartPos, @EndPos = R.EndPos  FROM dbo.ListRoadParts R  WHERE R.NumRoad = @NumRoad  and R.NumDataSource = @NumDataSource  and R.NumPartType = 33  CREATE TABLE #RoadParts (StartPos int, EndPos int)  SELECT \* INTO #RPDefects FROM #Defects WHERE 1 = 0  SET @SP = @StartPos  WHILE @SP < @EndPos  BEGIN  SET @EP = case when @SP + @step > @EndPos then @EndPos else @SP + @step end  INSERT INTO #RoadParts (StartPos, EndPos)  VALUES (@SP, @EP)  INSERT INTO #RPDefects (DefectTitle, Defect\_id, StartPos, EndPos, Param\_id, ParamVal)  SELECT D.DefectTitle, D.Defect\_id, @SP, @EP, D.Param\_id,  D.ParamVal \* (case when D.StartPos >= @SP and D.EndPos <= @EP then 1 else 1.0\*(dbo.pp\_GetMinValue(@EP, D.EndPos) - dbo.pp\_GetMaxValue(@SP, D.StartPos))/(D.EndPos - D.StartPos) end)  FROM #Defects D  WHERE (D.StartPos between @SP and @EP or @SP > D.StartPos and @EP <= D.EndPos)  SET @SP = @EP  END  SELECT RP.StartPos, RP.EndPos,  sum(case when D.Defect\_id = 2 then D.ParamVal else 0 end) as [Выбоины покрытия],  sum(case when D.Defect\_id = 16 then D.ParamVal else 0 end) as [Колейность],  sum(case when D.Defect\_id = 53 then D.ParamVal else 0 end) as [Съезды],  sum(case when D.Defect\_id = 103 then D.ParamVal else 0 end) as [Разрушение покрытия],  sum(case when D.Defect\_id = 104 then D.ParamVal else 0 end) as [Неровности]  FROM #RoadParts RP  LEFT JOIN #RPDefects D ON D.StartPos = RP.StartPos  GROUP BY RP.StartPos, RP.EndPos  DROP TABLE #Defects  DROP TABLE #RoadParts  DROP TABLE #RPDefects | | | | | | |