

PYTHAGORAS

**Использование  
полевого кодирования.**

## Введение

При импорте файла полевых измерений в Pythagoras, программа считывает выбранный файл, преобразует результаты измерений в координаты на чертеже и, если в поле использовались коды, то создает линии и условные знаки. Однако, Pythagoras может выполнить эти действия только, если программа знает структуру данных этого файл, смысл содержащихся в нем значений и, если использовались коды, то каким образом Вы хотите преобразовать эти коды в линии и условные знаки.

Преобразование кодов в линии, условные знаки и текст представляет собой очень сложный процесс. Pythagoras накладывает определенные ограничения на порядок использования кодов. Однако, даже в рамках этих ограничений, Вы можете задавать свои собственные коды. Более того, Вы можете также добавить ряд параметров, указывающих на то, как будет преобразовываться код: условный знак, линию или текст. Кроме того, эти параметры задают цвет создаваемых объектов, слой и многое другое.

Pythagoras может считывать файлы, образующийся при работе с большинством моделей тахеометров и накопителей данных, однако, программе необходимо указать ряд параметров для преобразования кодов в условные знаки, линии и текст. Для этого создается *файл описания кодов*.

*Файл описания кодов (CDF)* представляет собой текстовый файл, в котором содержится перечень действительных кодов, интерпретируя которые, Pythagoras, создает либо условный знак, либо линию, либо текст.

Файл описания кодов необходим, если Вы намерены использовать расширенную кодировку. Перед импортом данных необходимо загрузить файл. Если файл содержит ошибки, программа сообщит Вам об этом.

# Принципы расширенной кодировки

## Коды и опции

При использовании кодировки на местности с последующим автоматическим созданием чертежа в Pythagoras, Вам необходимо ввести для каждого полученного результата, по крайней мере, его код. Этот код показывает, какая именно точка была измерена (например, угол здания). За некоторыми кодами могут или должны следовать одна или более опций и параметров. Эти опции дают больше информации о типе точки. Если для здания выбран код BD, то буква S означает начало линии. Таким образом код BDS обозначает начало здания. Для некоторых опций требуются дополнительные параметры. Если при измерении полотна дороги Вы хотите создать параллельные линии, то в качестве параметров необходимо использовать опцию «parallel line».

## О КОДАХ

Все измерения, содержащиеся в файле накопителя данных, должны иметь код, определенный в файле описаний кодов. (При этом, конечно же, файл описаний кодов должен быть загружен).

Существуют следующие типы кодов:

- Коды для простых точечных условных знаков.
- Коды для простых линий
- Коды для полилиний.
- Коды для точек, ориентированных относительно линии, к которой они привязаны.
- Текстовые коды
- Коды для прямоугольных объектов.

*Длина кода* (минимум - 2, максимум - 8 символов), указывается в заголовке файла описаний кодов. Коды для полилиний и прямоугольников **обязательно** должны иметь длину, указанную в заголовке. Для всех других типов длина кода может быть короче. Например, если длина кода составляет 2 символа, то «B» – это действительный код для точечного условного знака, а не для полилинии.

### 1. Простые точки

Коды такого рода позволяют создать просто точку или точечный условный знак. Свойства точки и слой, к которому она принадлежит, определяются в файле описаний кодов.

**Также можно добавить до 3 текстовых фрагментов.** Эти текстовые фрагменты могут иметь фиксированное или различное содержание. Текстовые фрагменты с использованием кода данного типа создаются по тем же правилам, что и при использовании текстовых кодов.

Свойства и содержание текста (-ов) определяются в файле описаний кодов.

Так как тексты являются отдельными объектами, то Вы можете потом перемещать их по чертежу.

Если непосредственно на точку измерение не выполнялось, отражатель был слева, справа или перед точкой наблюдения, то Pythagoras может выполнить уравнивание полученных результатов. Для этого необходимо, чтобы этот промер и положение отражателя относительно точки, на которую выполнялось измерение, были заданы в виде опции к коду.

### 2. Простые линии

Pythagoras создает точку для каждого измерения. Точка, на которую было выполнено измерение, создается, как невидимый объект в слое, который определяется кодом простой линии. Если после кода идет номер действительной точки, то эти две точки соединяются линией. Свойства и слой этой линии определяются в файле описаний кодов.

Вы можете также задать, чтобы точка, на которую выполняется измерение, соединялась бы линией с предыдущей точкой без указания номера точки:

- а) используйте символ точки (.) : точка, на которую выполняется измерение, соединяется с предыдущей точкой, которая имеет тот же самый код.
- б) используйте две точки (..) : **точка, на которую выполняется измерение, соединяется с предыдущей точкой по типу «простая линия».**

Никаких ограничений по количеству одновременно открытых простых линий не существует. Более того, при открытых простых линиях Вы также можете выполнять измерения на другие точки. В отличие от полилиний (см. ниже) Вам не нужно начинать или заканчивать линию данного типа.

*Примечание : В инструментах от Wild (Leica) символ «.» показывается цифрой 1, а символ «..» - цифрой 2.*

### 3. Текст

При использовании текстового кода (**TEXT**) на чертеже создается горизонтально ориентированный текстовый фрагмент. Никакая точка при этом не создается. Положение текстового фрагмента на чертеже соответствует тому, как это было зафиксировано в накопителе данных. Если текст отсутствует, то данная кодировка не действует.

Свойства созданного текста и его содержание задаются в файле описаний кодов. По выбору в область содержания текста можно включить поля для переменных значений, например:

- Высота точки;
- Закодовый текст, хранящийся в файле накопителя данных;
- Текст из **STRING Table**. **STRING Table** – это справочная таблица, заданная в заголовке файла описаний кодов. В этом случае в файле накопителя данных за кодом должен следовать текст, используемый в качестве указателя, какой текст следует подгружать из справочной таблицы в поле для переменных значений.

Вы можете использовать этот код, чтобы добавить в чертеж любого рода комментарии.

### 4. Полилинии : линии и дуги, задаваемые серией точек.

При съемке улиц часто бывает необходимо измерить одновременно несколько линий (например, границу дорожного полотна, домов, электрических кабелей, ограждений, и т.д.). Pythagoras позволяет иметь открытыми одновременно до 10 полилиний. В отличие от простых линий, полилинии должны быть однозначно открытыми (начальная точка) и закрытыми (конечная точка). Помимо этого полилинии предоставляют следующие дополнительные возможности:

- Параллельные линии (до 3) на определенном удалении от полилинии;
- Перпендикулярные линии (либо влево, либо вправо) в точках, лежащих на полилинии;
- Дуги могут быть частью полилинии;
- Полилинии с невидимыми точками. (Точки участков полилинии, перпендикулярных отрезку предыдущей полилинии или не ее (**ego**) продолжении);
- Смещение различных кодов, принадлежащих одной и той же полилинии;
- Точечный условный знак имеет ту же ориентировку, что и полилиния, к которой он принадлежит (например, начало улицы).

#### Стиль точки – стиль линии

Pythagoras создает точки с использованием стиля, заданного в файле описаний кодов. Если стиль не задан, то точка будет невидимой.

Вторая и последующие точки соединенные с предыдущей образуют серию точек. Результатом такого объединения может быть линия или дуга, в зависимости от опций, которые указаны после кода.

#### Группы

Коды полилиний и *точек, привязанных к линии*, можно разделить на группы. (Более подробно о подразделении на группы см. Главу «Файл описания кодов»). В поле Вы можете одновременно выполнять измерения на точки, относящиеся к различным линиям, до тех пор пока коды этих линий принадлежат к различным группам.

Например, Вы можете выполнять измерения в следующей последовательности:

<u>Точка</u>	<u>Код</u>	<u>Описание</u>
1	WAS	Начало стены
2	BDS	Начало здания 1
3	BD	2-я точка здания 1
4	BD	3-я точка здания 1
5	DH	Дверь здания 1
6	BDE	Конечная точка здания 1
7	WAE	Конец стены.

Примечание : В этом примере код BD – это код полилинии для зданий, DH – код для «Точки, привязанной к линии», а WA – код полилинии для стены. Коды BD и DH принадлежат одной и той же группе (в данном случае - группе «Здания»), код WA принадлежит другой группе (в данном случае – группе «Стены»).

Pythagoras создает линии для здания, соединяя точки 1, 2 и 5, и создает линию от точки 3 до точки 6, показывающую стену. Для точки 4 Pythagoras создает соответствующий условный знак, ориентированный вдоль линии, соединяя точки 2 и 5.

Если различные коды типичной полилинии принадлежат к одной и той же группе, то их можно комбинировать при выполнении измерений последовательности точек, но считается, что они принадлежат к одной и той же полилинии. Код будет использоваться следующим образом : из 2 точек, по которым должна быть сформирована линия или касательная дуга, вторая точка в файле задает свойства линии (дуги) и слой, к которому она принадлежит. Из 3 точек, по которым должна быть сформирована дуга, третья точка в файле задает свойства дуги и слой, к которому она принадлежит. Сами точки создаются в слое, который задается кодом.

#### Начальная точка – Конечная точка полилинии

Первая точка полилинии должна быть задана, как начальная, а последняя точка – как конечная точка полилинии. Если BD – это код для здания, а S и E – символы начальной и конечной точки соответственно, то код BDS указывает на начальную точку здания, а код BDE – на конечную точку.

И хотя кажется более логичным начать измерения с начальной (S) точки, а закончить – конечной (E) точкой, Pythagoras позволяет выполнять измерения и в обратной последовательности. Это сделано по причинам, которые уходят корнями в прошлое. Давным-давно две конечные точки полилинии назывались Левая и Правая, если смотреть на здание со стороны фасада. Поэтому первая точка полилинии начиналась с кода, указывающего, с какой стороны обмеривается здание. Если Вы привыкли проводить обмер здания именно таким образом, то лучше использовать другие символы, чтобы задать начальную и конечную точки.

#### Видимые и невидимые точки полилинии

Видимые точки – это точки, координаты которых были измерены. Невидимыми мы называет такие точки, результаты измерения на которые были получены не с помощью тахеометра, а с использованием, например, мерной рулетки. Применение eccentric опции с указанием смещения за кодом точки полилинии позволяет показать, что точка расположена слева, справа или на продолжении предыдущего сегмента полилинии. Невидимые точки записываются в накопитель данных, но Pythagoras игнорирует их координаты.

Линейный сегмент, образованный соединением невидимой точки с предыдущей точкой полилинии, должен быть перпендикулярным или лежать на продолжении предыдущего линейного сегмента.

Направление и длина задаются в виде опции к коду.

Возможные направления :

- a. Перпендикуляр на предыдущей линии.
  - влево
  - вправо
- b. Линия на продолжении предыдущей линии.

Комментарий по ориентации Влево - Вправо

Точка находится слева от полилинии, если она расположена с левой стороны последнего сегмента полилинии.

Очевидно, что первые 2 точки полилинии не могут быть невидимыми.

Пример:

<u>Точка</u>	<u>Код</u>	<u>Описание</u>
1	BDS	Начало здания 1
2	BD	2-я точка здания 1
3	BD L200	3-я точка здания 1, на перпендикуляре и слева от линии 1-2
4	BD R250	4-я точка здания 1, на перпендикуляре и справа от линии 2-3
5	BDE	Последняя точка здания 1

Линия 2-3 имеет длину 200 см и представляет собой перпендикуляр влево от линии 1-2. Если линия была бы перпендикулярна в другом направлении, то следовало бы ввести значение BD R200.

Вышеприведенное определение справедливо, если в заголовке файла описаний кодов содержится LINEMODE(START\_END). По историческим причинам и в целях обеспечения совместимости с более ранними версиями Pythagoras, по умолчанию присутствует LINEMODE(LEFT\_RIGHT). Если все еще используется режим LEFT\_RIGHT, то для обозначения перпендикуляра обычно используется 1 символ. Знак перед ним показывает, в каком направлении (положительном или отрицательном) идет линия.

Главное отличие между двумя режимами заключается в следующем:

START\_END : направление относительно последней линии.

LEFT\_RIGHT : направление перпендикуляра относительно первой линии.

Примечание : Если используются файлы описаний кодов и файлы накопителей данных, созданные в версии Pythagoras 4.5x и более ранних версиях, то результаты измерений, импортированные при работе с более поздними версиями Pythagoras будут идентичными.

Рекомендуется использовать режим LINEMODE(START\_END), так как он более удобен при работе в поле.

### Дуги

Дуги задаются добавлением опции **arc** после кода полилинии. Кривая задается минимум по 3 последовательным точкам. Первые 3 точки формируют дугу. Последующие точки образуют касательную к предыдущей дуге.

Во избежание осцилляций при измерении дуг с переменным радиусом рекомендуется измерять небольшую линию в точках, где радиус дуги меняется.

### Линии, перпендикулярные полилинии

Чтобы получить перпендикуляр в точке на полилинии, необходимо добавить *perpendicular line option* за кодом точки, где должен начинаться этот перпендикуляр. Есть 2 кода, чтобы задать перпендикуляр; один код – для перпендикуляра влево, а другой – для перпендикуляра вправо. Тип автоматически создаваемой линии определяется соответствующим кодом.

Длина перпендикуляра задается в файле описаний кодов.

Этот тип перпендикуляра особенно удобен, если нужно измерить начальную и конечную точку здания, не выполняя при этом обмер всего здания.

#### Автоматически создаваемые параллельные линии

При создании полилинии Вы можете одновременно создать максимум 4 параллельные линии. Эти **параллельные линии создаются за полную** полилинию.

Вместе с кодом начальной точки полилинии Вы можете дать до 3 значений, показывающих расстояние до параллельных линий. Положительное значение обычно означает, что параллельная полилиния будет слева от измеренной полилинии, а отрицательное значение, - что параллельная линия будет справа.

Примечание: Вы можете изменить смысл вышеупомянутого значения LEFT = POSITIVE на LEFT = NEGATIVE, поставив для этого в заголовке файла описаний кодов параметр SIGNCONVENTION(RIGHTPOS). По умолчанию задан параметр SIGNCONVENTION(LEFTPOS).

Полилиния может содержать дуги. В этом случае создаются параллельные дуги. Эти параллельные дуги/линии автоматически сочленяются в точках пересечений.

**Параллельные линии не нужно оставлять параллельными по всему полигону.** Расстояния между ними могут варьироваться от точки к точке. Каждый раз при изменении расстояний необходимо добавлять все новые расстояния. Если новые расстояния не добавлены, то используются предыдущие значения расстояний.

Пример :

<u>Код1</u>	<u>Код2</u>	
BS	20+45 -120	линии на удалении соответственно 20см, 45см и 120см от измеренной точки.
BS	20+45 -100	линии на удалении соответственно 20см, 45см и 100см от измеренной точки.

**Последняя линия не будет параллельной,** но будет на удалении 120см (1м 20см) от первой точки и 100см (1м) от второй точки.

Примечание : единица измерения расстояний для опции - см.

## **5. Точки, привязанные к полилинии**

Типы кода могут иметь место только при измерении полилинии одной группы. Точка будет привязана к линии / дуге, образованной предыдущей и последующей точкой полилинии. Опять же, в чертеж точка добавляется, но если точечный условный знак является вращаемым, то он будет вращаться параллельно вместе с линией / дугой, к которой он привязан.

#### Стиль точки

Pythagoras создает точки с использованием стиля, указанного в файле описаний кодов. Если стиль не задан, то точка выглядит в виде небольшого крестика.

### Местоположение условного знака относительно полилинии (слева или справа)

Если стиль точки – это асимметричный условный знак, то может возникнуть необходимость добавить за кодом опцию направления. Когда местоположение не указано, то условный знак ставится слева от линии. Если добавить опцию RIGHT, то условный знак ставится справа от полилинии.

### Точка, спроецированная на полилинию

При необходимости на чертеже можно отобразить не саму точку, на которую были выполнены измерения, а ее проекцию на соответствующую линию. Если Вы хотите использовать эту функцию, то в файле описаний кодов, в свойствах кода необходимо указать параметр PROJECTION.

### Текст

При создании точки Pythagoras может создать на чертеже до 3 текстовых фрагментов. Содержание и свойства этих текстовых фрагментов зависит от свойств соответствующего кода в файле описаний кодов.

В файле описаний кодов задается ориентация текста: горизонтальная или наклонная. Положение текста на чертеже задается следующим образом:

1. Определение точки в файле описаний кодов может содержать параметр : POSITION(x, y). Этот параметр задает положение в мм (на странице) начального положения текста относительно точки.
2. Положение текста в документе **вычисляется относительно системы координат с началом в точке, где осью X является направление соответствующей линии** (или касательной к соответствующей линии).

## **6. Прямоугольники**

При измерении прямоугольников они должны измеряться, как две последовательные точки. Эти точки образуют одну сторону прямоугольника. Первая точка должна указываться, как начальная, вторая точка – как конечная.

Ширина прямоугольника задается, как параметр вместе с первой точкой прямоугольника. Значение этого параметра может быть положительным или отрицательным. Если значение положительно, то **прямоугольник простирается влево, если смотреть от начальной точки на конечную**. back of the rectangle will be left of the front of the rectangle (if seen from starting point to the end point).

Примечание 1: Вы можете изменить значение LEFT = POSITIVE на LEFT = NEGATIVE, поставив в заголовке файла описаний кодов параметр SIGNCONVENTION(RIGHTPOS). По умолчанию задан параметр SIGNCONVENTION(LEFTPOS)

Примечание 2: Допускается выполнение измерений сначала на конечную точку, а потом – на начальную точку.

Четыре точки прямоугольника добавляются в чертеж в качестве невидимых точек. Точки, на которые не были выполнены измерения, не имеют номера. Свойства линий и слой, к которому они принадлежат, определяются кодом прямоугольника и задаются в файле описаний кодов. В файле описаний кодов задаются также тип создаваемого прямоугольника. В зависимости от типа, прямоугольник имеет 1, 2 или ни одной диагонали.

Другая опция добавляет текст. Текст всегда будет параллельным **сторонам прямоугольника**. Начало текста привязано к середине неизмеренной стороны прямоугольника. Свойства, содержание, положение и слой, к которому привязан текст задаются в файле описаний кодов.

## ОБ ОПЦИЯХ

В зависимости от типа кода (простая точка, полилиния, и т.д.), можно задать одну или несколько опций.

**Ниже Вы познакомитесь с группами опций, после которых дается более подробное описание самих опций.**

Опции делятся на две группы :

Первая группа - Global options (ключевое слово GLOBAL\_OPTION) в файле описаний кодов.

Первая группа опций не имеет параметров (никаких дополнительных значений). Опция задается одним символом. За кодом может идти нулевое значение, одна или несколько опций.

- *START option* : один символ, показывающий начальную точку полилинии или прямоугольника.
- *END option* : один символ, показывающий конечную точку полилинии или прямоугольника.
- *ARC option* : один символ, показывающий, что точка полилинии лежит на дуговом сегменте.
- *PERPENDICULAR\_LEFT option* : один символ, чтобы добавить перпендикуляр влево от полилинии. Эта же опция может быть использована для точек, относящихся к полилинии, чтобы показать, что созданный символ должен быть слева от полилинии.
- *PERPENDICULAR\_RIGHT option* : один символ, чтобы добавить перпендикуляр вправо от полилинии.
- *CLOSE option*:: Эта опция может использоваться только для последней точки полилинии. (Опцию END задавать не нужно). Конечная точка соединяется с начальной точкой.

Эти опции добавляются в код без разделителя. (за исключением Leica). Некоторые из этих опций можно объединить, и последовательность может быть случайной.

Вторая группа : Extended global options (ключевое слово EXT\_GLOBAL\_OPTION) в файле описаний кодов.

Вторая группа опций требует, чтобы были заданы одно или несколько дополнительных значений (параметров). Параметром может быть строка, номер точки или расстояние. Единицу измерения расстояний можно выбрать в диалоговом окне опций кодировки. См. «Configure Data Collector».

В зависимости от типа кода, можно задать 1 или несколько опций. Более того, в файле описаний кодов вместе с определением кода может содержаться свойство OPTION, указывающее, что соответствующий код может иметь только одну опцию. Если определение кода допускает наличие несколько опций, то необходимо задать дополнительный параметр (всегда 1 символ), чтобы указать тип опции, следующей за кодом. Файл описаний кодов задает эти аббревиатуры. См. параметры EXT\_GLOBAL\_OPTION.

Если в коде допускается только один тип опции, то аббревиатуру **следует** опустить, а в поле кода можно добавить только параметр, отделяемый соответствующим разделителем, который принят для Вашей модели накопителя данных. Если код может иметь несколько опций, то для их идентификации перед параметром должна идти аббревиатура. **Возможные опции данного типа:**

- *TEXT option* : параметром является строка. Эта строка представляет собой либо текст, отображаемый на чертеже, либо аббревиатуру, чтобы подгрузить текст из STRING таблицы.
- *POINTID option* : параметр является номер точки.
- *ECCENTRIC (INVISIBLE POINT) option* : опции IN EXTENSION, PERPENDICULAR\_LEFT и PERPENDICULAR\_RIGHT могут быть использованы для

изменения измеренных координат, если фактически точка является эксцентричной (не была измерена) по отношению к измеренным координатам. Для точек на полилинии опция `eccentric` показывает положение невидимых точек на полилинии.  
(См. *Invisible points*).

- `WIDTH option` : параметром должно быть расстояние. Используется только для того, чтобы задать глубину прямоугольника.
- `PARALLEL_LINES option` : можно добавить до 3 значений расстояния, разделяя их знаком.

## 1. Использование опций первой группы

### Опция START

Опция начала должна добавляться за первой точкой полилинии или прямоугольника. Она показывает, что последуют 1 или несколько точек полилинии, и что все эти точки должны соединяться вплоть до конечной точки данной полилинии. Если задана начальная точка полилинии, то могут появиться и другие связанные с нею точки, и Pythagoras автоматически ориентирует условный знак в направлении полилинии.

Одновременно можно открыть несколько (до 10) полилиний, если они принадлежат другой группе. Поскольку относящиеся к полилинии точки (водоприемник, дверь дома), также принадлежат группе, Pythagoras распознает ориентировку символа, даже если открыты несколько полилиний.

### Опция END

Опция *End* должна идти за последней точкой полилинии или прямоугольника.

Примечание 1 : Если используется режим `LINEMODE(LEFT_RIGHT)`, то опции `START` и `END` называются опциями `LEFT` и `RIGHT` соответственно. Если же используется режим `LEFT_RIGHT`, то линии могут начинаться либо с `LEFT`, либо с `RIGHT`. Более подробно об этом см. Главу «Файл описаний кодов».

Примечание 2 : Прямоугольник всегда представляет собой последовательность 2 точек. Эти 2 точки обязательно должны быть последовательными в файле.

### Опция ARC

Pythagoras не имеет специальных кодов, указывающих начальную и конечную точки дуги. Вместо этого, каждая точка, являющаяся частью дуги, а именно: начальная точка, конечная точка и все промежуточные точки дуги должны содержать опцию `ARC`. Pythagoras строит дугу по первым трем точкам. Все последующие точки соединяются с предыдущей точкой посредством дуги, которая является касательной к предыдущей дуге. Опцию `Arc` можно использовать только для точек полилинии.

Пример : Приведенная ниже последовательность кодов (с учетом, что `A` – код дуги)

`BSS, BSA, BSA, BSA, BSE` создает : линию от Точки 1 до Точки 2, дугу через точки 2, 3 и 4, и линию от Точки 4 до Точки 5.

### Опция PERPENDICULAR\_LEFT

При съемке зданий, а также в других ситуациях, часто бывает необходимо отделить два здания с помощью перпендикуляра на измеряемой полилинии. Опции `PERPENDICULAR_LEFT` и

PERPENDICULAR\_RIGHT позволяют получить перпендикуляр на полилинии в точке, для которой задана эта опция. Перпендикулярная линия идет либо влево, либо вправо от полилинии.

Эту же опцию можно использовать для точек, относящихся к полилинии, чтобы указать, что созданный условный знак должен быть слева от полилинии. **Если эта опция отсутствует, то предполагается, что точка (условный знак) находится слева от полилинии.**

### Опция PERPENDICULAR\_RIGHT

См. опцию PERPENDICULAR\_LEFT.

### Опция CLOSE:

Эту опцию можно использовать только для последней точки полилинии. (опцию END задавать не нужно). Конечная точка соединяется с начальной точкой.

### Пример:

Ниже приводится действительная последовательность кодов с опциями.

BDS : начало полилинии  
 BDA : эта точка и 2 последующие точки образуют дугу.  
 BDA  
 BDA  
 BDL : точка (также создается перпендикуляр влево)  
 DH : слева от полилинии создается условный знак «Дверь».  
 (Если поставить DHR, то условный знак будет справа от полилинии)  
 BD : точка  
 BDC : Конечная точка, которая соединяется с первой точкой полилинии.

где :

BD : код для здания (из типа полилинии).  
 DH : код для двери здания  
 S : START,  
 E : END  
 C : CLOSE  
 A : ARC

## **2. Использование опций второй группы**

### Опция TEXT

Опция TEXT:

#### *1. %T в свойстве CONTENT файла описаний кодов*

Параметр текста ставится на чертеже, если свойство CONTENT кода имеет в описании текста строку %T. Значение %T заменяется на параметр текста.

Пример:

```
CDF :
VL = POINT (
    ....
    TEXT (
        CONTENT("Value = %T")
    )
)
```

Если за кодом идет параметр текста «ABC», то на чертеже появляется текст : «Value = ABC».

## 2. %S в свойстве CONTENT файла описаний кодов

Параметр текста используется в STRING table, в качестве указателя. Значение %S заменяется значением из string table.

Примечание:

CDF :

header

```
TEST = CDF (LENGTH (2)
           STRINGS (T1 = "Text 1"
                   PB = "This is a text"
                   )
           )
VL = POINT (
           ....
           TEXT (
               CONTENT ("Value = %S")
           )
           )
```

Если за кодом идет параметр текста «PB», то на чертеже появляется текст : «Value = This is a text».

Опция TEXT может использоваться для кодов типа Text, Point, Точка, привязанная к полилинии, или Прямоугольник. Для кодов типа Текст опция TEXT является обязательной.

## Опция ECCENTRIC используется для отдельных точек : код типа POINT

Данная опция используется для изменения координат, полученных для точки, расположенной в стороне от точки, на которую невозможно выполнить измерения, но координаты которой хотелось бы получить. Большинство современных тахеометров и накопителей данных имеют функцию для определения координат точек, на которые невозможно выполнить прямые измерения. В этом случае мы рекомендуем использовать функцию, заложенную в приборе, и не использовать опцию ECCENTRIC, имеющуюся в Pythagoras.

Для определения координат фактически недоступной точки предусмотрены три кода:

1. ECCENTRIC (INEXTENSION) – в случае, если точка, на которую выполняются измерения, находится перед фактически недоступной точкой ;
2. ECCENTRIC (PERPENDICULAR\_LEFT) – в случае, если точка, на которую выполняются измерения расположена слева от фактически недоступной точки;
3. ECCENTRIC (PERPENDICULAR\_RIGHT) - в случае, если точка, на которую выполняются измерения расположена справа от фактически недоступной точки.

Координаты измеренной точки относительно точки, координаты которой хотелось бы определить, всегда рассматриваются со стороны станции наблюдения.

Примечание 1 : Pythagoras требуется местоположение станции, чтобы определить фактические координаты. При использовании этой опции Вы обязаны фиксировать результат измерения на каждую станцию в соответствии с условными обозначениями, используемыми Pythagoras.

Примечание 2 : Опция Eccentric имеет другой смысл, если используется для точек полилинии. (См. далее).

#### Опция ECCENTRIC, используемая для точек полилинии : код типа POLYLINE

Если опция Eccentric используется вместе с кодом точки полилинии, то эта опция показывает, что фактически недоступная точка находится слева, справа или на продолжении предыдущего отрезка полилинии. **Это противоречит тому, что отдельные недоступные точки, которые находятся слева, справа или на продолжении линии наблюдения от станции до измеряемой точки.**

Эта опция была описана ранее в данном руководстве в разделе *Visible and invisible points in a polyline* следующим образом :

"Видимые точки – это точки, координаты которых были определены в результате измерений. Невидимыми точками называются такие точки, координаты которых не были определены непосредственно при измерении тахеометром, но были получены другим путем, например, с помощью мерной рулетки. Если за кодом точки полилинии используется опция Eccentric вместе с расстоянием, то это указывает на то, что точка находится слева, справа или на продолжении предыдущего отрезка полилинии. Невидимые точки регистрируются в накопителе данных, но Pythagoras игнорирует эти координаты."

#### Опция POINTID

Эту опцию могут иметь только коды типа *Simple Line*. Если за кодом идет номер точки, то линия создается от действительной точки до точки с указанным номером.

Символ точки и двоеточия указывает соответственно на соединение с предыдущей точкой, имеющей тот же самый код, и на соединение с предыдущей точкой типа **Simple Line**.

#### Опция WIDTH

Эта опция обязательна для прямоугольников. Данный параметр задает ширину прямоугольника.

#### Опция PARALLEL LINES

Эта опция может использоваться только для точек полилинии. Более подробно об этом см. *Automatic generated parallel lines* в описании полилиний.

## III. Файл описаний кодов

### Функция

Файл описаний кодов представляет собой чисто текстовый файл, где перечисляются коды, которые могут быть использованы в процессе импорта файлов с накопителя данных, если в Pythagoras включена опция «extended code conversion». За каждым из кодов в файле описаний кодов может идти подробное описание того, как создается объект, например, линия, дуга, точка, текст должны быть визуализированы на чертеже. При прочтении данной главы мы рекомендуем в качестве иллюстративного материала использовать пример файла описаний кодов, который находится на инсталляционной дискете.

### СТРУКТУРА И СИНТАКСИС ФАЙЛА ОПИСАНИЙ КОДОВ

Файл описаний кодов состоит из 3 разделов :

- (1) заголовок,
- (2) аббревиатуры (сокращения) для опций и
- (3) описание кодов.

Заголовок и аббревиатура для опций являются необязательными.

Все разделы имеют одинаковую структуру и состоят из одного или более блоков, которые выглядят следующим образом :

```
NAME = KEYWORDS ([ATTRIBUTE]) .
```

And ATTRIBUTE = KEYWORDS ([ATTRIBUTE]) ]

NAME – это строка, содержащая только буквенно-цифровые символы «a».. «z», «A».. «Z», «0».. «9». Она используется, чтобы указать для опции имя, код или аббревиатуру.

KEYWORDS используются для идентификации блока. Они пишутся заглавными буквами внутри данного документа. Keywords используются в ATTRIBUTE, отменяя свойства по умолчанию для объекта, созданного по коду.

Блок можно распространить на несколько линий в текстовом файле. Пустые линии пропускаются. Комментарий можно добавить в самом блоке или между блоками, поставив перед ним точку с запятой « ; ». По завершении комментария нажмите клавишу возврата каретки (ENTER). Например,

```
BL = POINT (                ; this is comment
           LAYER ("my layer") ; just add some comment
         )
; this is a comment line
```

#### Примечание для пользователей Windows :

Если Вы в файле описаний кодов используете специальные символы для названия слоев или строк текста (например, ÿ, é, ...), помните, что Pythagoras работает с набором символов, имеющимся в Windows. Набор символов Windows отличается от аналогичного набора символов для DOS. Чтобы избежать проблем при создании файла описаний кодов, используйте текстовый редактор Windows, например, Notepad.

## ЗАГОЛОВОК

Ключевое слово для заголовка – файл описаний кодов. В заголовке содержится имя файла описаний кодов.

Заголовок файла описаний кодов может содержать свойства:

- LENGTH
- STRINGS
- LINEMODE
- SIGNCONVENTION

### LENGTH

Длина кодов ограничена числом, задаваемым свойством LENGTH. Если задать LENGTH(3), то это означает, что код ограничен 3 символами (*полями*). Далее, все коды типа Полилиния, Прямоугольник и Точки, связанные с полилинией, должны иметь такую длину. Коды для Текста, Простых точек и Простых линий могут быть короче. Во избежание недоразумений мы рекомендуем присвоить всем кодам одну и ту же длину.

По умолчанию длина равна 2 символам.

### STRINGS

За STRINGS следуют до 100 строк, в которых задается аббревиатура (до 100 символов). Этот перечень строк, который называется также Справочной таблицей, используется для вставки в строку параметра тип текста (типовой текст), заданного в файле накопителя данных. Если нужно выполнить съемку парка, то на чертеже необходимо указать названия всех деревьев; Вы можете для всех деревьев задать один код (например, TR), и задать в STRINGS список названий в том порядке, в каком они будут отображаться на чертеже вместе с аббревиатурой. Если в накопителе данных ввести код "TR T1", то на чертеже отобразится название «Quercus Rubra».

#### Синтаксис STRINGS

```
STRINGS (
    <Abbreviation> = <Name>
    <Abbreviation> = <Name>
    ....
)
```

### LINEMODE

За свойством linemode должен идти либо параметр LEFT\_RIGHT, либо START\_END. Это свойство задает, какой режим используется при измерении полилиний.

```
LINEMODE (START_END)
```

Если используется режим START\_END, то первая точка полилинии должна иметь опцию START, а последняя точка - опцию END. (В более ранних, нежели 4.6 версиях Pythagoras опции START и END назывались соответственно LEFT и RIGHT.) При измерении полилиний с использованием режима START END, последний отрезок полилинии всегда является опорной линией, если используются опции PERPENDICULAR\_LEFT, PERPENDICULAR\_RIGHT, ECCENTRIC(PERPENDICULAR\_LEFT) и ECCENTRIC(PERPENDICULAR\_RIGHT).

```
LINEMODE (LEFT_RIGHT)
```

Если используется режим LEFT\_RIGHT, то первая точка полилинии может иметь опцию LEFT или RIGHT, а последняя точка должна иметь противоположную опцию. (Начиная с версии 4.6, опции LEFT и RIGHT называются опциями START и END.) При измерении полилиний с использованием режима LEFT\_RIGHT первый отрезок полилинии всегда

является опорным, если используются опции PERPENDICULAR\_LEFT, PERPENDICULAR\_RIGHT, ECCENTRIC(PERPENDICULAR\_LEFT) и ECCENTRIC(PERPENDICULAR\_RIGHT).

По историческим причинам по умолчанию задан режим LINEMODE (LEFT\_RIGHT). Pythagoras версии до 4.5 не поддерживал этот параметр и импортирование всегда происходило с использованием режима LEFT\_RIGHT. Поскольку режим START\_END более понятен и удобен для работы в поле, мы рекомендуем использовать LINEMODE (START\_END).

### SIGNCONVENTION

За свойством signconvention должен идти либо параметр LEFTPOS, либо - RIGHTPOS. Это свойство задает :

- будет ли положительным расстояние для параллельных линий либо слева, либо справа от полилинии,
- будет ли положительной ширина для прямоугольника либо слева, либо справа от измеренной линии прямоугольника.

Если используется режим LINEMODE (LEFT\_RIGHT), то SIGNCONVENTION задает :

- будет ли положительным расстояние для параллельных линий либо слева, либо справа от полилинии,
- будет ли положительной ширина для прямоугольника либо слева, либо справа от измеренной линии прямоугольника,
- будет ли перпендикулярная линия слева или справа от полилинии,
- будет ли положительным значение для невидимого перпендикуляра слева или справа от первой линии полилинии.

Пример заголовка:

```
MYCDF = CDF(LENGTH(3)           ; Length of code = 3
          STRINGS(T1 = "Big tree" ; Text for parameter T1
              T2 = "Small tree"  ; Text for parameter T2
          )
          LINEMODE(START_END)    ; Use of START_END convention
          SIGNCONVENTION(LEFTPOS) ; LEFT of polyline is positive.
        )
```

Рекомендуется поставить в заголовке файла описаний кодов LINEMODE (START\_END) .

### **АББРЕВИАТУРЫ ДЛЯ ОПЦИЙ**

Как уже объяснялось выше, Вы можете, а в некоторых случаях и обязаны, добавить вслед за кодом опции, чтобы привести более подробную информацию об измеряемой точке. Существуют 2 группы опций. Первая группа используется, к примеру, для того, чтобы указать, что данная точка является начальной точкой полилинии, конечной точкой полилинии, точкой дуги, и т.д. Вторая группа опций используется, например, для добавления текста к точке, чтобы указать ширину прямоугольника, расстояние между параллельными линиями и т.д.

В файле описаний кодов эти 2 группы опций называются : Global options (ключевое слово GLOBAL\_OPTION) и Extended global options (ключевое слово EXT\_GLOBAL\_OPTION).

Теоретически, за кодом могут следовать различные опции Группы 2. Например, в случае прямоугольника требуется, чтобы за первой точкой шла опция WIDTH, но также можно добавить и текст, который должен выводиться на чертеж вместе с этим прямоугольником. Если Вы хотите, чтобы за кодом следовали 2 опции, то перед параметрами опций требуется поставить аббревиатуру, чтобы указать значение опции.

Как будет более подробно показано ниже, при обсуждении свойства OPTION, которое может принадлежать коду, аббревиатура для опций расширенной кодировки (Extended global options) требуется только в исключительных случаях. Практически используется аббревиатура для опций ECCENTRIC.

Все опции имеют аббревиатуру по умолчанию. Вы можете отменить заданные по умолчанию параметры. Аббревиатура может быть длиной только в один символ. В группе ***global options*** все аббревиатуры должны быть одинаковыми. Это же условие справедливо для группы ***extended global options***. Будьте внимательны, если Вы заменили только часть сокращений. Другие сокращения по-прежнему могут иметь параметры, которые были заданы по умолчанию, так что сокращения во всей группе могут быть неодинаковые.

Перечень всех значений по умолчанию :

```
L = GLOBAL_OPTION (START) ; Аббревиатура для начальной точки линии (слева)
R = GLOBAL_OPTION (RIGHT) ; Аббревиатура для конечной точки линии (справа)
B = GLOBAL_OPTION (ARC) ; Аббревиатура для точки дуги
D = GLOBAL_OPTION (PERPENDICULAR_LEFT) ; Аббревиатура для перпендикуляра влево
E = GLOBAL_OPTION (PERPENDICULAR_RIGHT) ; Аббревиатура для перпендикуляра вправо
C = GLOBAL_OPTION (CLOSE) ; Аббревиатура для замкнутой полилинии

* = EXT_GLOBAL_OPTION (PARALLEL_LINES) ; Нет аббревиатуры для параллельных линий
T = EXT_GLOBAL_OPTION (TEXT) ; Аббревиатура для Text
W = EXT_GLOBAL_OPTION (WIDTH) ; Аббревиатура для Width
V = EXT_GLOBAL_OPTION (ECCENTRIC (INEXTENSION)) ; Аббревиатура для In Extension
L = EXT_GLOBAL_OPTION (ECCENTRIC (PERPENDICULAR_LEFT)) ; Аббревиатура для Perpendicular
left
R = EXT_GLOBAL_OPTION (ECCENTRIC (PERPENDICULAR_RIGHT)) ; Аббревиатура для Perpendicular right
```

Мы рекомендуем заменить все эти аббревиатуры.

Примечание :

Символ звездочка "\*" можно использовать в группе ***extended global options***, чтобы указать, что эта опция используется без аббревиатуры. Если первый символ кода 2 (часть, которая содержит опции группы 2) не начинается с действительной аббревиатуры кода 2, то считается, что эта опция имеет аббревиатуру \*.

## КОДЫ

Все коды, которые Вы хотите использовать, должны быть перечислены в файле описаний кодов. Для каждого из них должны быть указаны, как минимум, тип кода : простая точка, простая линия, прямоугольник, текст, полилиния или точка, относящаяся к полилинии. Для осмысления этих типов см. разделы выше в данном Руководстве. По каждому из этих типов в Pythagoras предусмотрены свойства по умолчанию, которые можно отменить, добавив соответствующее ключевое слово и значение.

Список ключевых слов для каждого типа кодировки:

```
Тип «простая точка» POINT
Тип «простая линия» LINE
Тип «прямоугольник» RECTANGLE
Тип «текст» TEXT
Тип «полилиния» POLYLINE
Точка, относящаяся к линии POINT_WITHIN_POLYLINE
```

Поэтому самое простое определение кода выглядит следующим образом :

```
512 = POINT ()
```

При загрузке файла описаний кодов с таким определением, то Вы можете импортировать файлы, в которых присутствует код 512. Однако, все коды по умолчанию будут иметь свойство **POINT**. Параметры по умолчанию описываются ниже.

Другое простое определение кода 120 :

```
120 = POLYLINE()
```

Это позволяет использовать код 120 для полилиний. Поэтому первая точка полилинии должна иметь опцию START, а последняя точка полилинии – опцию END. Полилиния создается на чертеже, начинаясь от первой точки, соединяя все промежуточные точки кода 120, пока не достигнет конечной точки полилинии. Поскольку в файле описаний кодов для кода 120 не было задано никаких свойств, то линия будет сплошной, черной и создана в активном слое.

### Range (Последовательность, ряд, серия ...?)

Обычно большинство файлов описаний кодов представляют собой последовательность определений простых кодов вместе с их свойствами. Однако, возможно задать последовательность кодов. На примере, который приводится ниже, поясняется, что же понимается под последовательностью:

```
B1..B4 = POINT(
    STYLE(0, 3)
    LAYER(Trees)
    COLOUR(GREEN)
)
```

Это определение указывает на то, что коды B1, B2, B3 и B4 (последовательность B1..B4) являются действительными и имеют одинаковые свойства. Вы можете потом переписать в файл описаний кодов любой или все коды, заданные в какой-либо последовательности. Однако, переписанный код будет иметь все предыдущие свойства, если Вы специально не зададите другое.

Так, если определение кода:

```
B3 = POINT (
    STYLE(0, 2)
)
```

соответствует определению последовательности кодов B1..B4, которая дана выше, то точка с кодом B3 по-прежнему будет в слое «Trees» и зеленого цвета, но уже с другим стилем.

### Ошибка кодировки

Если код не задан в файле описаний кодов, то Pythagoras создает в текущем слое условный знак красного цвета с надписью **ОШИБКА** сверху.

В этом случае Вы можете, тем не менее, задать свойства, определив код, как « \* »

Пример:

```
* = POINT (
    STYLE(0, 1)
    COLOUR(RED)
    LAYER("Error Code")
)
```

### Общие свойства

LAYER

Свойство LAYER задает, в каком слое Pythagoras будет размещать объекты, имеющие этот код (эту кодировку). Если слой на чертеже отсутствует, то Pythagoras создает его.

Введите между скобками название слоя, а если название состоит из нескольких слов, разделенных пробелом, или Вы хотите использовать другие специальные символы, то возьмите название слоя в одинарные или двойные кавычки. В названии слоя используются разрешенные Pythagoras символы, и их количество не должно превышать 12.

Рабочим по умолчанию является слой, который был активным на момент импорта файла накопителя данных.

Свойство LAYER может быть использовано для всех типов кодировки.

Пример:

```
LAYER("my layer")
```

## COLOUR

Цветом по умолчанию для всех объектов, создаваемых при импортировании файла накопителя данных, является BLACK. Чтобы получить другой цвет, добавьте ключевое слово COLOUR и задайте цвет по одному из следующих параметров:

BLACK, BLUE, CYAN, RED, YELLOW, GREEN или MAGENTA. Также Вы можете использовать номер цвета 1..255.

Пример:

```
COLOUR(RED)
```

```
COLOUR(135)
```

## WIDTH

Свойство WIDTH задает ширину линии в 1/10 миллиметра. Это свойство может быть использовано для кодировки типа LINE, RECTANGLE или POLYLINE.

По умолчанию : 1. (0.1 мм.)

## OPTION

В зависимости от типа используемой кодировки, Вы можете добавить одну или несколько опций, которые следуют за кодом. Например, при измерении прямоугольника Вы обязательно должны ввести ширину прямоугольника в виде опции. Однако, вполне возможно, что помимо ширины Вы захотите также добавить и текст, который будет размещаться на чертеже. Если Вы хотите иметь возможность задавать несколько опций кодировок, то перед каждой опцией Вы должны ставить аббревиатуру EXT\_GLOBAL\_OPTION, чтобы программа знала, как ей интерпретировать эти данные.

Однако, намного удобнее спроектировать коды таким образом, чтобы Вам пришлось добавлять только одну опцию. В этом случае аббревиатуру можно не вводить, а программа сама знает, как ей интерпретировать данные.

Есть одно исключение : если Вы используете опцию *eccentric*, то Вам всегда придется указывать направление : на продолжении, перпендикуляр влево или вправо.

Свойство OPTION указывает, какая же опция используется вместе с кодом. Параметр может иметь следующие значения : TEXT, POINTID, ECCENTRIC, WIDTH, PARALLEL\_LINES.

Рекомендуется добавить свойство OPTION с каждым кодом (кодировкой) (кроме кодировок LINE и TEXT) в файл описаний кодов.

	TEXT	POINT NUMBER	ECCENTRIC	WIDTH	PARALLEL
POINT	X	-	X	-	-
LINE	-	D	X	-	-
TEXT	D	-	-	-	-
POLYLINE	-	-	X	-	X
POINT IN POLYLINE	X	-	X	-	-
RECTANGLE	x	-	X	X	-

- x : опции допускаются
- : опции не допускаются
- D : опция по умолчанию

Пример :

Полилинии : допустимыми являются eccentric points и automatic generated parallel polylines. Вы можете задать одну из этих опций, поскольку только одну опцию Вы будете использовать для данной полилинии (экономя тем самым время на ввод аббревиатур в файлах накопителя данных), применяя ключевое слово OPTION. Чтобы ограничить число опций по отношению к automatic generated parallel polylines, добавьте OPTION(PARALLEL\_LINES) следом за кодировкой в файле описаний кодов. При использовании ECCENTRIC Вы ограничиваете опции автоматической обработкой eccentric measured points, но Вам по-прежнему придется добавлять аббревиатуру в файл накопителя данных, чтобы указать, лежит ли эта точка на продолжении последнего сегмента полилинии, или на перпендикуляре к нему.

### STYLE

Свойство STYLE задает:

- a) стиль точки, если имеет место тип кодировки POINT или POINT\_WITHIN\_POLYLINE,
- b) стиль линии, если имеет место тип кодировки LINE, POLYLINE или RECTANGLE .
- c) стиль текста, если имеет место тип кодировки TEXT.

При указании стиля точки или стиля линии ключевое слово STYLE имеет 2 параметра, разделенных запятой. Первый параметр выдает группу условного знака или стиль линии. Второй параметр – это название условного знака в группе. Если Вы хотите задать стандартную точку или стиль линии, то группа = 0, а название представляет собой последовательный номер стиля, как показано в меню.

Стандартные стили точек : STYLE (x, y)

<u>x, y</u>	<u>Стиль</u>
0, 0	Hidden (Invisible)
0, 1	Cross (+) <i>Default</i>
0, 2	x Cross (x)
0, 3	Slash (/)
0, 4	Round dot (•)
0, 5	Round dot hollow (o)
0, 6	Rectangle dot
0, 7	Rectangle dot hollow
0, 8	Triangle (-)

Стандартные стили линий: STYLE (x, y)

<u>x, y</u>	<u>Стиль</u>
0, 0	Solid <i>по умолчанию</i>
0, 1	Dash
0, 2	Dot
0, 3	Dash Dot

## 0, 4 Dash Dot Dot

Если вместо стиля по умолчанию используется другой стиль, то этот условный знак или этот стиль линии обязательно должны присутствовать в библиотеке Pythagoras. Если их там нет, то при загрузке файла описаний кодов Pythagoras выдает сообщение об ошибке.

Пример:

```
STYLE("Trees", "Oak")
```

**Примечание** : При переименовании или удалении объектов из библиотеки условных знаков будьте внимательны, чтобы не затронуть объекты, которые используются в данный момент файлом описаний кодов.

При завершении работы в Pythagoras программа сохраняет название рабочего файла описаний кодов в файле PYTHAGOR.DAT. (При работе в среде Macintosh название сохраняется в ресурсном файле Pythagoras). При запуске Pythagoras загружается рабочий файл описаний кодов. Если условные знаки или стили линий в библиотеке не найдены, то выдается сообщение об ошибке и файл описаний кодов не загружается.

POINT

Во избежание недоразумений обратите внимание на то, что ключевое слово POINT может использоваться в 2 случаях. Либо, как указатель типа кодировки, о чем было уже сказано выше, либо, в качестве свойства кодировки типа POLYLINE.

В последнем случае, задаются свойства точки для точек полилинии ***POLYLINE***. За ключевым словом свойства POINT могут идти одно или несколько свойств : STYLE, LAYER, COLOUR. Это означает, что точка полилинии может иметь различный цвет или слой, по сравнению с сегментами полилинии.

Если STYLE относится к вращаемому условному знаку, то этот условный знак на чертеже будет ***повернут и ориентирован, как и линия, предшествующая точке***. Первая точка полилинии будет иметь направление первой линии.

TEXT

Во избежание недоразумений обратите внимание на то, что ключевое слово TEXT может использоваться в 2 случаях. Либо, как указатель типа кодировки, о чем было уже сказано выше, либо, в качестве свойства кодировки типа POINT, POINT\_WITHIN\_POLYLINE и RECTANGLE.

В последнем случае, это ключевое слово задает текст и свойства, которые должны использоваться в сочетании с вышеупомянутыми типами.

NETWORKPOINT

Это свойство показывает, что измеренную точку можно уравнивать. Параметр задается, если точка является простой точкой хода или фиксированной точкой хода (т.е. точкой, координаты которой известны). Выглядит все это следующим образом:

```
; Simple points

; Traverse Point
V = POINT (
    LAYER("Traverse")
    STYLE(0, 1)
    COLOUR(0)
    NETWORKPOINT(TP)           ; regular traverse point
)

; Fixed point
```

```
N = POINT      (  
                LAYER("Traverse")  
                STYLE(0, 1)  
                COLOUR(0)  
                NETWORKPOINT(FP)           ; traverse point with known coordinates.  
                )
```

### Свойства текста (TEXT)

Ниже приводится список свойств, используемых для кодировок типа TEXT или для других видов текстовых кодировок.

#### LAYER

См. Общие свойства.

Примечание : Если для свойств текстовых кодировок не задано свойство LAYER, то текст будет располагаться в том же слое, что и основной объект. (Точка или прямоугольник).

#### COLOUR

См. Общие свойства.

Примечание : Если для свойств текстовых кодировок не задано свойство COLOUR, то текст будет в том же цвете, что и основной объект. (Точка или прямоугольник).

#### WEIGHT

Свойство WEIGHT задает **плотность** текста. Этот параметр может иметь следующие значения : LIGHT, MEDIUM, BOLD или EXTRA\_BOLD. По умолчанию плотность текста средняя.

#### STYLE

Этот параметр может быть либо ITALIC, либо NORMAL. Оба параметра соответствуют стилю, используемому для текста на чертеже Pythagoras.

Если STYLE отсутствует, то по умолчанию задается параметр NORMAL.

#### SIZE

Свойство SIZE задает размер шрифта текста. Данный параметр имеет цифровое отображение и может содержать значения от 4 до 72. По умолчанию шрифт имеет размер 10.

#### HORIZONTAL ALIGNMENT

Свойство ALIGNMENT задает горизонтальное выравнивание текста. Этот параметр может иметь значение : LEFT, RIGHT или CENTERED. По умолчанию задается параметр CENTERED.

#### VERTICAL ALIGNMENT

Свойство VALIGNMENT задает вертикальное выравнивание текста. Этот параметр может иметь значение : TOP, BOTTOM или CENTERED. По умолчанию задается параметр BOTTOM.

#### BORDER

Свойство BORDER задает тип рамки. Этот параметр может иметь значение : SINGLE или DOUBLE. Если данный параметр отсутствует, то текст не имеет рамки.

#### UNDERLINE

Свойство UNDERLINE задает тип подчеркивания. Этот параметр может иметь значение : SINGLE, DOUBLE или DASHED. Если данный параметр отсутствует, то текст остается не подчеркнутым.

#### POSITION

Свойство POSITION задает положение текста относительно точки измерения. Это свойство нельзя использовать для кодировок типа TEXT. Его можно использовать только для кодировок типа POINT, POINT\_WITHIN\_POLYLINE и RECTANGLE, а также в сочетании со свойством TEXT.

За свойством POSITION идут 2 значения: *x* и *y*. Оба значения выражаются в миллиметрах и задают положение текста относительно точки. При импортировании Pythagoras вынужден преобразовывать миллиметры в метры. (Pythagoras сохраняет все координаты внутри файла в метрах). Это означает, что при преобразовании используется фактический масштаб чертежа. Поэтому важно, чтобы окончательный масштаб чертежа был выбран до начала процесса импортирования данных. На практике Вы могли бы импортировать данные, определить, каким будет масштаб окончательного чертежа, отменить импорт, изменить масштаб на тот, который Вы хотели бы иметь в итоге, и снова начать импортирование данных. Добавленный текст, после того как он создан, конечно же можно переместить в другое место.

По умолчанию параметр имеет значение: POSITION(0, 0).

Если свойство POSITION используется для кодировки типа RECTANGLE, положение будет вычисляться относительно середины задней стороны прямоугольника. По умолчанию текст в прямоугольнике (с прямоугольником) имеет ориентировку OBLIQUE (см. ниже).

Если свойство POSITION используется для кодировок типа POINT\_WITHIN\_POLYLINE, то положение вычисляется в системе координат с началом в этой точке и с ориентировкой направления полилинии в точке измерения.

#### OBLIQUE /HORIZONTAL

Ключевые слова OBLIQUE и HORIZONTAL могут быть использованы со свойством TEXT для кодировок типа RECTANGLE и для кодировок типа POINT\_WITHIN\_POLYLINE. Во всех других случаях текст на чертеже ориентирован горизонтально.

#### OBLIQUE

- а) Для прямоугольников : текст будет параллельным измеренной линии (стороне) прямоугольника.
- б) Для полилиний : текст будет параллельным полилинии в точке измерения.

#### HORIZONTAL

Текст на чертеже ориентирован горизонтально. Даже если Вы впоследствии повернете чертеж, то текст все равно сохранит горизонтальную ориентировку.

По умолчанию параметр имеет значение OBLIQUE.

#### FONT

Здесь Вы можете задать шрифт текста. За свойством "Font" должны идти название нужного шрифта и соответствующая кодовая страница (только для WINDOWS). В большинстве европейских стран используется кодовая страница "Western".

Пример синтаксиса: FONT("Arial", Western)

Если это свойство не появляется в файле описаний кодов, то в качестве шрифта по умолчанию будет использоваться Векторный шрифт Pythagoras ( '*Pythagoras Vector Font*' ).

### OPAQUE

Это свойство определяет, будет или нет текст иметь заливку. Используется, когда текст появляется в шаблонах (*hachings*).

По умолчанию текст не имеет заливки.

Синтаксис : OPAQUE

### DISPLAY LEVEL

С помощью этого свойства Вы можете задать уровень, в котором будут отображаться объекты. По умолчанию параметр имеет значение "0".

Синтаксис : DISPLAY\_LEVEL(x) (где x – номер нужного уровня)

### CONTENT

Свойство CONTENT задает текст, который будет отображаться на чертеже. Свойство CONTENT содержит 1 параметр. Это строка длиной в 15 символов. Если Вы хотите использовать специальные символы или пробел, возьмите строку с текстом в кавычки.

Пример :

```
BT = POINT(
    TEXT(
        CONTENT("Big Tree")
    )
)
```

Точка с кодом BT отобразится на чертеже в виде небольшого крестика с надписью "Big Tree" (выровненной по центру и размером шрифта 10 пунктов) непосредственно сверху над точкой.

Если в свойстве CONTENT используется символ процента « % », то часть файла описаний кодов имеет особый смысл. Действительными являются следующие сочетания:

- %T
- %S
- %I
- %H

#### 1. %T

Значение %T заменяется строкой, которая вводится в файл накопителя данных. (См. Выше описание опции TEXT в файле накопителя данных).

Пример:

```
VL = POINT(
    TEXT(
        CONTENT("Value = %T")
    )
)
```

Если за кодом идет текстовый параметр в виде строки «ABC», то на чертеже появляется текст : "Value = ABC".

## 2. %S

Текстовый параметр в файле накопителя данных будет использоваться в качестве индекса в STRING таблице. Значение %S заменяется значением из string таблице.

Пример:

```
TEST = CDF (LENGTH (2)
            STRINGS (T1 = "Text 1"
                    PB = "This is a text"
                    )
            )
VL = POINT (
            TEXT (
                CONTENT ("Value = %S")
            )
        )
```

## 3. %H

Значение %H в строке заменяется высотной отметкой точки. Высотная отметка всегда дается в местной системе координат, даже если данные импортируются относительно системы координат пользователя с другой высотной отметкой.

Пример:

```
Z = POINT (
            TEXT (
                CONTENT ("z=(%H) ")
            )
        )
```

Использование кодировки типа Z ставит точку на чертеж с надписью « z=(5.66) » вверху, если высота точки в местной системе координат составляет 5,66 метра.

Примечание : Высотные отметки отображаются в единицах, заданных в меню «Preferences» в Pythagoras.

## 4. %I

Значение %I в строке заменяется номером точки.

По умолчанию задается следующее содержание :

- а) с кодировкой типа текст : "%T"
- а) со свойством text : «no text»

Примечание : %I и %H создают только для текста с Точками и Точками на полилинии.

### Свойства для кодировки простых точек (POINT)

#### LAYER

См. Общие свойства.

По умолчанию : активный (рабочий) слой.

COLOUR

См. Общие свойства.  
По умолчанию : черный.

STYLE

См. Общие свойства.  
По умолчанию : (0, 1) : Cross (***Крестик***).

OPTION

См. Общие свойства.  
Может быть : TEXT или ECCENTRIC.

TEXT

См. Общие свойства.

До трех текстовых фрагментов могут быть добавлены к точке.

Параметры по умолчанию для текста :

- HORIZONTAL

- CONTENT(«No text»)

- LAYER : тот же слой, что и условный знак точки.

- все другие свойства имеют стандартные параметры по умолчанию для текста. (черный, 10 пт, ***normal, medium, centered***).

NETWORK POINT (NETWORKPOINT)

См. Общие свойства

Параметры: TP (простая точка хода)

FP (фиксированная (***опорная***) точка)

**Свойства для кодировки простых линий (LINE)**LAYER

См. Общие свойства.  
По умолчанию : активный (***рабочий***) слой.

COLOUR

См. Общие свойства.  
По умолчанию : ЧЕРНЫЙ.

STYLE

См. Общие свойства.  
По умолчанию : сплошная линия.

STYLE

См. Общие свойства.

По умолчанию : (0, 0) : **Solid**

### WIDTH

См. Общие свойства.

По умолчанию : 1. (0,1 мм)

Пример :

```
ML = LINE (
    LAYER("Border")           ; слой «Border»
    COLOUR(RED)                ; красная линия
    STYLE("Road marks", "Border") ; Название стиля линии в библиотеке .
    WIDTH(2)                   ; толщина 0,2 мм
)
```

При использовании простой линии есть только одна действительная опция, и это – идентификатор точки, которую Вы хотите соединить с текущей точкой. Если Вы используете точку (.) или двоеточие (..), **то это означает, что Вы хотите соединить точку с предыдущей точкой, имеющей тот же самый код, или предыдущей точкой типа LINE.**

### Свойства для прямоугольников (RECTANGLE)

#### LAYER

См. Общие свойства.

По умолчанию : активный (**рабочий**) слой.

#### COLOUR

См. Общие свойства.

По умолчанию : ЧЕРНЫЙ.

#### STYLE

См. Общие свойства.

По умолчанию : сплошная линия.

#### TYPE

Свойство TYPE задает число диагоналей в прямоугольнике :

- a) R0 : нет диагоналей
- b) R1 : 1 диагональ
- c) R2 : 2 диагонали

По умолчанию : R0

#### OPTION

См. Общие свойства.

Могут быть : TEXT, WIDTH или ECCENTRIC.

Обычно прямоугольники всегда имеют свойство OPTION(WIDTH). При использовании кодировки типа прямоугольник **обязательно** необходимо указать WIDTH.

#### WIDTH

См. Общие свойства.  
По умолчанию : 1. (0,1 мм)

#### TEXT

См. Общие свойства и Свойства для типа Text.  
До трех текстовых фрагментов могут быть добавлены к прямоугольнику.  
Параметры по умолчанию для текста :  
- OBLIQUE  
- CONTENT («No text»)  
- LAYER : тот же слой, что и условный знак точки.  
- все другие свойства имеют стандартные параметры по умолчанию для текста. (черный, 10 пт, ***normal, medium, centered***).

Текст ориентирован параллельно длине большей стороны прямоугольника (***длина*** большей стороны прямоугольника образуется по линии между 2 измеренными точками, найденными в файле накопителя данных). Середина противоположной стороны прямоугольника – положение по умолчанию.

#### **Свойства для полиний (POLYLINE)**

##### LAYER

См. Общие свойства.  
По умолчанию : активный (***рабочий***) слой.

##### COLOUR

См. Общие свойства.  
По умолчанию : ЧЕРНЫЙ.

##### STYLE

См. Общие свойства.  
По умолчанию : сплошная линия.

##### OPTION

См. Общие свойства.  
Может быть : PARALLEL\_LINES или ECCENTRIC.

При съемке участков дорог часто требуется получить параллельные линии. Если у Вас есть кодировка, которая требуется для параллельных линий, рекомендуется задать свойство OPTION(PARALLEL\_LINES).

##### WIDTH

См. Общие свойства.  
По умолчанию : 1. (0,1 мм)

##### LENGTH

Свойство LENGTH задает длину автоматически добавляемого перпендикуляра **фиксированной длины**. По умолчанию длина этого перпендикуляра составляет 5.0 метров. Вы можете изменить его длину, добавив опцию LENGTH, где задать нужный параметр длины в метрах.

### GROUP

Свойство GROUP задает, к какой группе принадлежит полилиния. При измерении полилиний одной и той же группы Pythagoras соединяет точки даже, если у них разные коды (**кодировка**). В поле Вы можете одновременно выполнять измерения на точки, лежащие на различных линиях, пока коды для этих линий принадлежат разным группам. Группа задает название, например, «Электричество»

Если свойство GROUP не задано, то полилиния по умолчанию относится к безымянной группе.

### OPEN\_PARALLELS

Если начертить полилинию с автоматически создаваемыми параллельными полилиниями, то Pythagoras обычно соединяет измеренную точку с начальной и конечной точкой каждой параллельной полилинии.

Чтобы отключить все эти соединения, добавьте свойство OPEN\_PARALLELS.

### **Свойство для POINT\_WITHIN\_POLYLINE**

#### LAYER

См. Общие свойства.

По умолчанию : активный (**рабочий**) слой.

#### COLOUR

См. Общие свойства.

По умолчанию : ЧЕРНЫЙ.

#### STYLE

См. Общие свойства.

По умолчанию : (0, 1) : Cross (**Крестик**).

### GROUP

Свойство GROUP задает, к какой группе принадлежит точка. Когда одновременно открыты несколько различных линий, и был измерен код типа POINT\_WITHIN\_POLYLINE, то он будет относиться к открытой линии той же самой группы.

Коды, которые не имеют свойство GROUP, принадлежат по умолчанию к той же самой группе. Группа задает название, например, «Электричество»

Если свойство GROUP не задано, то полилиния по умолчанию относится к безымянной группе.

### NO\_PROJECTION - PROJECTION

Если задано свойство PROJECTION, то измеренная точка проецируется на полилинию, к которой эта точка принадлежит. По умолчанию задан параметр NO\_PROJECTION.

Примечание : Если использовать свойство PROJECTION, то координаты точки изменятся.

### OPTION

См. Общие свойства.

Может быть : TEXT или ECCENTRIC.

### TEXT

См. Общие свойства.

До трех текстовых фрагментов могут быть добавлены к точке.

Параметры по умолчанию для текста :

- OBLIQUE

- CONTENT («No text»)

- LAYER : тот же слой, что и условный знак точки.

- все другие свойства имеют стандартные параметры по умолчанию для текста. (черный, 10 пт, ***normal, medium, centered***).

Тексты по умолчанию имеют стиль ***oblique*** и ориентированы параллельно полилинии. Если Вы хотите изменить этот параметр по умолчанию, то смещение в направлении *x* означает, что Вы смещаете текст параллельно полилинии. (Смещение со знаком «+» - совпадает с направлением линии). Смещение в направлении *y* означает, что Вы смещаете текст в перпендикулярном направлении.

## Примеры

### Example 1

```
EP = POINT (
  LAYER ("Electricity")           ; --> Слой «Электричество»
  OPTION (ECCENTRIC)              ; единственная опция - ECCENTRIC
  STYLE ("Utilities", "Electr. Pole") ; (группа в библиотеке, название)
  COLOUR (BLUE)                  ;
  TEXT (
    LAYER ("Texts")              ; --> слой "Texts"
    STYLE (NORMAL)               ; Стиль
    WEIGHT (MEDIUM)             ; Weight
    SIZE (6)                    ; Шрифт
    ALIGNMENT (CENTERED)        ; Выравнивание
    COLOUR (BLACK)              ; Цвет
    CONTENT ("EP")
    POSITION (0, 2)               ; DELTA X = 0 мм
                                ; DELTA Y = 2 мм.
  )
)
```

Если используется код EP, то на чертеже создается условный знак с обозначением ЛЭП, коммуникационных линий «Utilities», «Electr. Pole». Этот условный знак **обязательно** должен быть в библиотеке. Если условного знака с таким названием в библиотеке нет, то файл описаний кодов загружаться не будет.

Условный знак будет синего цвета и располагаться в слое «Электричество». Если слой на чертеже все еще не сформирован, то слой он формируется. Вместе с условным знаком создается текст «EP», который располагается в 2 мм вверх от главной точки условного знака. Размер текста - 6 пт, черный и располагается в слое «Texts». Если свойство LAYER не указано, то текст все равно будет располагаться в слое «Электричество».

Пример 2

```

DH = POINT_WITHIN_POLYLINE(
    LAYER("Buildings")          ; --> Layer "Buildings"
    OPTION(TEXT)                 ; valid option = text only
    STYLE ("Buildings","Door House"); Name of the symbol
    PROJECTION                    ; Project point on polyline
    GROUP("Buildings")          ; Same group as code BD.
    TEXT(
        CONTENT("%T")           ; Content : %T replaced
                                ; by text in record
        HORIZONTAL               ; OBLIQUE /HORIZONTAL
        POSITION(0,6)            ; DELTA X = 0 mm DELTA Y = 6 mm
    )
    TEXT(
        STYLE(ITALIC)           ; Style
        WEIGHT(LIGHT)           ; Weight
        SIZE(6)                  ; point size
        CONTENT("(%H)")          ; Content : %H replace by
                                ; elevation of point
        POSITION(0,2.4)
    )
)

```

Так как код DH принадлежит к типу POINT\_WITHIN\_POLYLINE, точка должна быть измерена, когда полилиния группы «Buildings» является открытой (незамкнутой).

При использовании кода DH создается условный знак, обозначающий на чертеже «Здания», «Дверь дома», и параллельный полилинии. Координаты точки будут перевычислены, так чтобы главная точка лежала бы на полилинии. Этот условный знак обязательно должен быть в библиотеке. Если условного знака с таким названием в библиотеке нет, то файл описаний кодов загружаться не будет.

Условный знак и оба текста будут на чертеже черного цвета и располагаться в слое «Здания». Если слой на чертеже все еще не сформирован, то слой он формируется. Вместе с условным знаком создаются 2 текста. Первый текст ориентирован в горизонтальном направлении, а его содержание определяется параметром, который следует за кодом DH. Текст по умолчанию имеет следующие свойства: выравнивание по центру, стиль normal, размер шрифта 10 пт, цвет – черный). Положение текста (на момент импорта) – 6 мм сверху от полилинии.

Второй текст будет ориентирован параллельно полилинии и располагаться в 2,4 мм вверх от главной точки условного знака. Размер шрифта – 6 пт, цвет – черный, *light* и *italic*. В тексте содержатся данные о высотной отметке точки на чертеже (относительно МЕСТНОЙ системы координат). Эти данные о высотной отметке заключены в скобки.

## Приложение

В данном приложении приводится файл описаний кодов в том виде, в каком он принимается Pythagoras.

### КОДЫ ПРОСТЫХ ТОЧЕК

Код 1	Код 2	Описание	Условный знак на чертеже
KT	(1)	(small)TV box	Rectangle 30x30
BL	(1)	Broad-leaved tree	Trunk (d=0.2) en crown (d=4.0)
BN	(1)	Pine tree	Round point
BF	(1)	Fruit tree	Round point
IP	(1)	Inspection cover	Circle (d=0.7)
VL	(1)	Traffic light	Round point + "VL"
VT	(1)	Traffic Sign	Round point + "VT"
EP	(1)	Overhead line mast	Round point + "EP"
VP	(1)	Street light	Round point
CP	(1)	Combination pole	Round point + "CP"
BK	(1)	Fire cock	Round point + "BK"
SM	(1)	Sleutelmond	Round point + "SM"
AP	(1)	Close picket	Round point
GP	(1)	Boundary mark	Rectangle point
Z		Elevation	Point
C	text	Comment	The text in Code 2.
V		Traverse point	Point

(1) Расширенная кодировка для точек, на которые невозможно выполнить прямые измерения.

### КОДЫ ЛИНИЙ

Код 1	Описание	Условный знак линии на чертеже
AW	Enclosure	Similar line type.
AD	Wire fence	Similar line type.
AB	Fence concrete	Similar line type.
AH	Hedge	Similar line type.
AM	Wall	Similar line type.
BS	Border	Normal line 0.1.
KS	Marginal strip	Normal line 0.1
WG	Road gully	Normal line 0.1.
KV	Kantverharding	Normal line 0.1.
AS	Horizontal alignment	Dash Dot line 0.1.
ML	Reference line	Normal line 0.1.
GR	Ditch	Similar line type.
TB	Talud up	Similar line type.
TO	Talud down	Similar line type.
PG	Plot border	Normal line 0.1.
GB	Building	Normal line 0.5

### КОДЫ ДЛЯ ТОЧЕК ПОЛИЛИНИИ

Код 1	Код 2	Описание	Условный знак на чертеже
DH	housesnr.	Doorstep	Triangle on frontline + height + housesnr
DG	housesnr.	Garage	Triangle on frontline + height

За кодами DH и DG может идти буква L или R, чтобы указать, лежит ли точка слева или справа от опорной линии. Если никакая буква не указана, то предполагается, что точка расположена слева.

Коды DH и DG могут иметь место только при съемке фасада здания **GB (building)**.

SK	(1)	Street inlet	Symbol Street inlet (30x70). The coordinates determine the middle of the street inlet
----	-----	--------------	--

Код SK имеет место только при съемке дороги (BS, KS, ...).

(1) Расширенная кодировка для точек, на которые невозможно выполнить прямые измерения.

**КОДЫ ДЛЯ ПРЯМОУГОЛЬНИКОВ**

Код 1	Код 2	Описание	Условный знак на чертеже
STx	(2)	Stable	Rectangle with 2 diagonals
KMx	(2)	Wall	Rectangle
KEx	(2)	Electricity cabin	Rectangle with diagonal + "E"
KGx	(2)	Gas cabin	Rectangle with diagonal + "G"
KTx	(2)	TV Cabin	Rectangle with diagonal + "T"

x = S or E. S & E обязательно должны появляться попарно.

(2) Ширина прямоугольника (в см). Направление перпендикулярно измеренной линии и измеряется по направлению S-E. (См. Направление линий)

**ФАЙЛ ОПИСАНИЙ КОДОВ**

```

TEST = CDF (LENGTH (2)                ; TEST : name of CDF
                                ; Length code = 2
                                LINEMODE (START_END)
                                SIGNCONVENTION (RIGHTPOS))

; abbreviations

S = GLOBAL_OPTION (START)             ; Abbreviation for start of line
E = GLOBAL_OPTION (END)               ; Abbreviation for end of line
B = GLOBAL_OPTION (ARC)               ; Abbreviation for arc
L = GLOBAL_OPTION (PERPENDICULAR_LEFT) ; Abbreviation for Perpendicular Line to the left of a polyline
R = GLOBAL_OPTION (PERPENDICULAR_RIGHT) ; Abbreviation for Perpendicular Line to the left of a polyline

* = EXT_GLOBAL_OPTION (PARALLEL_LINES) ; no Abbreviation for ParallelLines
T = EXT_GLOBAL_OPTION (TEXT)          ; Abbreviation for Text
W = EXT_GLOBAL_OPTION (WIDTH)         ; Abbreviation for width
V = EXT_GLOBAL_OPTION (EXCENTRIC (INEXTENSION))
                                ; Abbreviation for InExtension
L = EXT_GLOBAL_OPTION (EXCENTRIC (PERPENDICULAR_LEFT))
R = EXT_GLOBAL_OPTION (EXCENTRIC (PERPENDICULAR_RIGHT))
                                ; Abbreviations for excentric and invisible points Left and right.

; Simple points

; BROAD-LEAVED TREE
BL = POINT
(
    LAYER ("Environment")             ; layer
    OPTION (ECCENTRIC)                ; optional coding
    STYLE (0, 5)                      ; (name of the pointlibrary ,Id)
    COLOR (GREEN)                    ; color
)

; FRUIT TREE
BF = POINT
(
    LAYER ("Environment")             ; layer
    OPTION (ECCENTRIC)                ; optional coding
    STYLE (0, 5)                      ; (name of the pointlibrary ,Id)
    COLOUR (GREEN)                   ; color
)

; PINE TREE
BN = POINT
(
    LAYER ("Environment")             ; layer
    OPTION (ECCENTRIC)                ; optional coding
    STYLE (0, 5)                      ; (name of the pointlibrary ,Id)
    COLOUR (GREEN)                   ; color
)

; INSPECTION COVER
    
```

```

IP = POINT
    (
        LAYER("Roads")           ; layer
        OPTION(ECCENTRIC)        ; optional coding
        STYLE(1, 3)              ; (name of the pointlibrary ,Id)
        COLOUR(BLACK)           ; color
    )

; STREET LIGHT
VP = POINT
    (
        LAYER("Environment")     ; layer
        OPTION(ECCENTRIC)        ; optional coding
        STYLE(0, 5)              ; (name of the pointlibrary ,Id)
        COLOR(BLACK)            ; color
    )

; CLOSE PICKET
AP = POINT
    (
        LAYER("Grenzen")         ; layer
        OPTION(eccentric)        ; optional coding
        STYLE(0, 5)              ; (name of the pointlibrary ,Id)
        COLOR(BLACK)            ; color
    )

; BOUNDARY MARK
GP = POINT
    (
        LAYER("Grenzen")         ; layer
        OPTION(ECCENTRIC)        ; optional coding
        STYLE(0, 7)              ; (name of the pointlibrary ,Id)
        COLOR(BLACK)            ; color
    )

; OVERHEAD LINE MAST
EP = POINT
    (
        LAYER("Environment")     ; layer
        OPTION(ECCENTRIC)        ; optional coding
        STYLE(0 , 5)             ; (name of the pointlibrary ,Id)
        COLOR(BLACK)            ; color
        TEXT
        (
            LAYER("Environment") ; Layer
            STYLE(NORMAL)        ; Style
            WEIGHT(MEDIUM)       ; Weight
            SIZE(6)              ; point size
            ALIGNMENT(CENTERED)  ; Alignment
            COLOR(BLACK)         ; Color
            CONTENT("EP")
            HORIZONTAL            ; OBLIQUE /HORIZONTAL
            POSITION(0 , 0)       ; DELTA X, DELTA Y in mm (page units)
        )
    )

; TRAFFIC LIGHT
VL = POINT
    (
        LAYER("Signals")         ; layer
        OPTION(ECCENTRIC)        ; optional coding
        STYLE(0 , 5)             ; (name of the pointlibrary ,Id)
        COLOR(BLACK)            ; color
        TEXT
        (
            LAYER("Signals")     ; Layer
            STYLE(NORMAL)        ; Style
            WEIGHT(MEDIUM)       ; Weight
            SIZE(6)              ; point size
            ALIGNMENT(CENTERED)  ; Alignment
            COLOR(BLACK)         ; Color
            CONTENT("VL")
            HORIZONTAL            ; OBLIQUE /HORIZONTAL
            POSITION(0 , 0)       ; DELTA X, DELTA Y in mm (page units)
        )
    )

; TRAFFIC SIGN
VT = POINT
    (
        LAYER("Signals")         ; layer
        OPTION(ECCENTRIC)        ; optional coding
        STYLE(0 , 5)             ; (name of the pointlibrary ,Id)
        COLOR(BLACK)            ; color
        TEXT
        (
            LAYER("Signals")     ; Layer
            STYLE(NORMAL)        ; Style
            WEIGHT(MEDIUM)       ; Weight
            SIZE(6)              ; point size
            ALIGNMENT(CENTERED)  ; Alignment
            COLOR(BLACK)         ; Color
            CONTENT("VT")
            HORIZONTAL            ; OBLIQUE /HORIZONTAL
            POSITION(0,0)         ; DELTA X, DELTA Y in mm (page units)
        )
    )

; COMBINATION LINE MAST AND STREET LIGHT
CP = POINT
    (
        LAYER("Environment")     ; layer
        OPTION(ECCENTRIC)        ; optional coding
    )

```

```

STYLE(0 , 5)          ; (name of the pointlibrary ,Id)
COLOR(BLACK)         ; color
TEXT
(
  LAYER("Environment") ; Layer
  STYLE(NORMAL)        ; Style
  WEIGHT(MEDIUM)       ; Weight
  SIZE(6)              ; point size
  ALIGNMENT(CENTERED) ; Alignment
  COLOR(BLACK)         ; Color
  CONTENT("CP")
  HORIZONTAL           ; OBLIQUE /HORIZONTAL
  POSITION(0 , 0)       ; DELTA X, DELTA Y in mm (page units)
)
)

; FIRE COCK
BK = POINT
(
  LAYER("Environment") ; layer
  OPTION(ECCENTRIC)    ; optional coding
  STYLE(0 , 5)         ; (name of the pointlibrary ,Id)
  COLOR(BLACK)        ; color
  TEXT
  (
    LAYER("Environment") ; Layer
    STYLE(NORMAL)        ; Style
    WEIGHT(MEDIUM)       ; Weight
    SIZE(6)              ; point size
    ALIGNMENT(CENTERED) ; Alignment
    COLOR(BLACK)         ; Color
    CONTENT("BK")
    HORIZONTAL           ; OBLIQUE /HORIZONTAL
    POSITION(0 , 0)       ; DELTA X, DELTA Y in mm (page units)
  )
)

; SLEUTELMOND
SM = POINT
(
  LAYER("Environment") ; layer
  OPTION(ECCENTRIC)    ; optional coding
  STYLE(0 , 5)         ; (name of the pointlibrary ,Id)
  COLOR(BLACK)        ; color
  TEXT
  (
    LAYER("Environment") ; Layer
    STYLE(NORMAL)        ; Style
    WEIGHT(MEDIUM)       ; Weight
    SIZE(6)              ; point size
    ALIGNMENT(CENTERED) ; Alignment
    COLOR(BLACK)         ; Color
    CONTENT("SM")
    HORIZONTAL           ; OBLIQUE /HORIZONTAL
    POSITION(0 , 0)       ; DELTA X, DELTA Y in mm (page units)
  )
)

; TRAVERSE POINT
V = POINT
(
  LAYER("Basis")       ; Layer
  STYLE(0,1)           ; name of the pointlibrary, id
  COLOR(BLACK)
)

; KABELTELEVISION
TV = POINT
(
  LAYER("Environment") ; Layer
  STYLE(0,5)           ; name of the pointlibrary, id
  COLOR(BLACK)
)

; ELEVATION
Z = POINT
(
  LAYER("Basis")       ; Layer
  STYLE(0,1)           ; name of the pointlibrary, id
  COLOR(BLACK)
  OPTION(TEXT);
)

; POINTS WITHIN POLYLINE

; STREET INLET

SK = POINT_WITHIN_POLYLINE
(
  LAYER("Roads")       ; Layer
  STYLE(1, 2)          ; Name of the pointlibrary,id
  COLOR(RED)           ; color
  NO_PROJECTION        ; Project point on polyline ?
  GROUP("Roads")
)

; DOORSTEP
DH = POINT_WITHIN_POLYLINE
(

```

```

LAYER("Buildings")      ; Layer
OPTION(Text)            ; valid option = text only
STYLE(1, 1)             ; Name of the pointlibrary
COLOR(BLUE)             ; color
PROJECTION              ; Project point on polyline
GROUP("Buildings")
TEXT (
  LAYER("Buildings")    ; Layer
  STYLE(NORMAL)         ; Style
  WEIGHT(MEDIUM)        ; Weight
  SIZE(10)              ; point size
  ALIGNMENT(CENTERED)  ; Alignment
  VALIGNMENT(CENTERED) ; Vertical Alignment
  COLOR(BLUE)           ; Color
  CONTENT("%T")         ; Content : %T replace by text in record
  HORIZONTAL            ; OBLIQUE /HORIZONTAL
  POSITION(0, 8)         ; DELTA X, DELTA Y in mm (page units)
)
TEXT (
  LAYER("Buildings")    ; Layer
  STYLE(ITALIC)         ; Style
  WEIGHT(LIGHT)         ; Weight
  SIZE(6)               ; point size
  ALIGNMENT(CENTERED)  ; Alignment
  VALIGNMENT(CENTERED) ; Vertical Alignment
  COLOR(BLUE)           ; Color
  CONTENT("%H")         ; Content : %H replace by elevation of point
  OBLIQUE               ; Oblique text
  POSITION(0, 2.9)       ; Delta x, Delta y in mm (page units)
)
)

; GARAGE
DG = POINT_WITHIN_POLYLINE
(
  LAYER("Buildings")    ; Layer
  OPTION(TEXT)          ; valid option = text only
  STYLE(1, "1")         ; Name of the pointlibrary
  COLOR(BLACK)          ; color
  PROJECTION            ; Project point on polyline
  GROUP("Buildings")
  TEXT (
    LAYER("Buildings")  ; Layer
    STYLE(ITALIC)       ; Style
    WEIGHT(MEDIUM)     ; Weight
    SIZE(6)             ; point size
    ALIGNMENT(CENTERED) ; Alignment
    VALIGNMENT(CENTERED) ; Vertical Alignment
    COLOR(BLUE)         ; Color
    CONTENT("%H")       ; Content : %H replace by elevation of point
    OBLIQUE             ; Oblique text
    POSITION(0, 2.9)     ; Delta x, Delta y in mm (page units)
  )
)

; TEXTS
C = TEXT (
  LAYER("Comment")      ; Layer
  STYLE(NORMAL)         ; Style
  WEIGHT(MEDIUM)        ; Weight
  SIZE(6)               ; point size
  ALIGNMENT(CENTERED)  ; Alignment
  COLOR(BLACK)          ; Color
  CONTENT (" %T ")      ; %T replace by text in record
)

; POLYLINES

; ENCLOSURE
AW = POLYLINE (
  LAYER("Buildings")    ; Layer
  OPTION(ECCENTRIC)     ; optional coding only eccentric points allowed
  STYLE(1, 0)           ; Name of the line library
  WIDTH(1)              ; Line Width
  COLOR(BLACK)          ; Color
  LENGTH(5.00)          ; Length of the perpendicular line in m
)

; WIRE FENCE
AD = POLYLINE (
  LAYER("Buildings")    ; Layer
  OPTION(ECCENTRIC)     ; optional coding only eccentric points allowed
  STYLE(1, 1)           ; Name of the line library
  WIDTH(1)              ; Line Width
  COLOR(BLACK)          ; Color
  LENGTH(5.00)          ; Length of the perpendicular line in m
)

; HEDGE

```

```

AH = POLYLINE (
    LAYER("Buildings") ; Layer
    OPTION(ECCENTRIC) ; optional coding only eccentric points allowed
    STYLE(1, 3) ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(GREEN) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
)

; FENCE CONCRETE
AB = POLYLINE (
    LAYER("Buildings") ; Layer
    OPTION(ECCENTRIC) ; optional coding only eccentric points allowed
    STYLE(1, 2) ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(BLACK) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
)

; WALL
AM = POLYLINE (
    LAYER("Buildings") ; Layer
    OPTION(ECCENTRIC) ; optional coding only eccentric points allowed
    STYLE(1, 4) ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(BLACK) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
)

;BUILDING
GB = POLYLINE (
    LAYER("Buildings") ; Layer
    OPTION(ECCENTRIC) ; optional coding only in extension allowed
    STYLE("DEFAULT", 0) ; Name of the line library
    WIDTH(5) ; Line width
    COLOR(BLUE) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
    GROUP("Buildings")
)

; HORIZONTAL ALIGNMENT
AS = POLYLINE (
    LAYER("Roads") ; Layer
    OPTION(ECCENTRIC) ; optional coding only eccentric points allowed
    STYLE("DEFAULT", "3") ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(RED) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
)

; BORDER
BS = POLYLINE (
    LAYER("Roads") ; Layer
    STYLE("DEFAULT", "0") ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(RED) ; Color
    LENGTH(0.00) ; Length of the perpendicular line in m
    GROUP("Roads")
)

; ROAD GULLY
WG = POLYLINE (
    LAYER("Roads") ; Layer
    STYLE("DEFAULT", "0") ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(RED) ; Color
    LENGTH(0.00) ; Length of the perpendicular line in m
    GROUP("Roads")
)

; MARGINAL STRIP
KS = POLYLINE (
    LAYER("Roads") ; Layer
    STYLE("DEFAULT", "0") ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(RED) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
    GROUP("Roads")
)

;KANTVERHARDING
KV = POLYLINE (
    LAYER("Roads") ; Layer
    STYLE("DEFAULT", "0") ; Name of the line library
    WIDTH(1) ; Line Width
    COLOR(RED) ; Color
    LENGTH(5.00) ; Length of the perpendicular line in m
    GROUP("Roads")
)

; LINE

```

```

ML = POLYLINE (
    LAYER("Basis")           ; Layer
    STYLE("DEFAULT", "0")    ; Name of the line library
    WIDTH(1)                 ; Line Width
    COLOR(BLACK)             ; Color
    LENGTH(5.00)            ; Length of the perpendicular line in m
)

; DITCH
GR = POLYLINE (
    LAYER("Environment")    ; Layer
    STYLE(1, 15)            ; Name of the line library
    WIDTH(1)                 ; Line Width
    COLOR(BLUE)             ; Color
    LENGTH(5.00)            ; Length of the perpendicular line in m
)

; TALUD UP
TB = POLYLINE (
    LAYER("Environment")    ; Layer
    STYLE("DEFAULT", "0")    ; Name of the line library
    WIDTH(1)                 ; Line Width
    COLOR(BLACK)            ; Color
    LENGTH(5.00)            ; Length of the perpendicular line in m
)

; TALUD DOWN
TO = POLYLINE (
    LAYER("Environment")    ; Layer
    STYLE("DEFAULT", "0")    ; Name of the line library
    WIDTH(1)                 ; Line Width
    COLOR(BLACK)            ; Color
    LENGTH(5.00)            ; Length of the perpendicular line in m
)

; PLOT BORDER
PG = POLYLINE (
    LAYER("Grenzen")        ; Layer
    STYLE("DEFAULT", "0")    ; Name of the line library
    WIDTH(1)                 ; Line Width
    COLOR(BLACK)            ; Color
    LENGTH(5.00)            ; Length of the perpendicular line in m
)

; RECTANGLES

; STABLE
ST = RECTANGLE (
    LAYER("Buildings")      ; Layer
    OPTION(WIDTH)           ; Option = Width
    STYLE(0,0)              ; Solid line
    WIDTH(2)                ; Width
    COLOR(BLUE)             ; Color
    TYPE(R2)                ; Rectangle with 2 diagonals
)

; ELECTRICITY CABIN
KE = RECTANGLE (
    LAYER("NutsVoorz")      ; Layer
    OPTION(WIDTH)           ; Option = Width
    STYLE("DEFAULT", 0)     ; Solid Line
    WIDTH(2)                ; Width
    COLOR(BLACK)            ; Color
    TYPE(R1)                ; Rectangle with 1 diagonal
    TEXT (
        LAYER("NutsVoorz")  ; Layer
        STYLE(NORMAL)       ; Style
        WEIGHT(MEDIUM)      ; Weight
        SIZE(6)             ; point size
        ALIGNMENT(CENTERED) ; Alignment
        COLOR(BLACK)        ; Color
        CONTENT("E")        ; Fixed content
        POSITION( 0, 1.5)     ; Delta x, Delta y in mm (page units)
    )
)

; GAS CABIN
KG = RECTANGLE (
    LAYER("NutsVoorz")      ; Layer
    OPTION(WIDTH)           ; Option = Width
    STYLE("DEFAULT", 0)     ; Solid Line
    WIDTH(1)                ; Width
    COLOR(BLACK)            ; Color
    TYPE(R1)                ; Rectangle with 1 diagonal
    TEXT (
        LAYER("NutsVoorz")  ; Layer
        STYLE(NORMAL)       ; Style
        WEIGHT(MEDIUM)      ; Weight
        SIZE(6)             ; point size
        ALIGNMENT(CENTERED) ; Alignment
        COLOR(BLACK)        ; Color
    )
)

```

```

        CONTENT("G")          ; Fixed content
        POSITION(0, 1.5)       ; Delta x, Delta y in mm (page units)
    )
)

; TELEVISION CABIN
KT = RECTANGLE (
    LAYER("NutsVoorz")       ; Layer
    OPTION(WIDTH)            ; Option = Width
    STYLE("DEFAULT", 0)      ; Solid Line
    WIDTH(1)                 ; Width
    COLOR(BLACK)             ; Color
    TYPE(R1)                 ; Rectangle with 1 diagonal
    TEXT (
        LAYER("NutsVoorz")   ; Layer
        STYLE(NORMAL)        ; Style
        WEIGHT(MEDIUM)       ; Weight
        SIZE(6)              ; point size
        ALIGNMENT(CENTERED)  ; Alignment
        COLOR(BLACK)         ; Color
        CONTENT("T")         ; Fixed content
        POSITION(0, 1.5)      ; Delta x, Delta y in mm (page units)
    )
)

; WALL
KM = RECTANGLE (
    LAYER("Environment")    ; Layer
    OPTION(WIDTH)           ; Option = Width
    STYLE("DEFAULT", 0)     ; Solid Line
    WIDTH(1)               ; Width
    COLOR(BLACK)           ; Color
    TYPE(R0)               ; Rectangle with 1 diagonal
)

```

- (1) В файле описаний кодов целесообразно заменить стиль условных знаков библиотеки и стили линий (1, n) на полное название условного знака или стиля линии, как они указаны в библиотеке.

Например, (1, 2) -> ("Example, "Doorstep")