WLAProbe

Автоматический пробинг детали (M199)

Руководство пользователя - 13.11.2026

Содержание

Содержание 2
1 Назначение 4
2 Установка 4
3 Как устроена команда M199 4
3.1 Система координат входных координат (cs) 4
3.2 Обязательная последовательность 5
4 Режимы 5
4.1 Замеры 5
4.1.1 probeZ — плоскость 5
4.1.2 probeXY — точка касания 5
4.1.3 probeCirc3P — окружность по трём точкам 5
4.1.4 probeCirc4P — окружность по четырём точкам 5
4.1.5 probeSize2P — размер по двум точкам 5
4.1.6 probeRot — поворот по двум точкам 5
4.1.7 probeQuad — угол заготовки 6
4.2 Установка (без перемещения) 6
4.2.1 setPosSC — координаты в точке 6
4.2.2 setCurPosSC — текущие координаты 6
4.2.3 setRotSC — поворот СК 6
4.2.4 setZprobe — Z по замеру 6
4.3 Сохранение точки (savePoint) 6
4.4 Расчёты по сохранённым точкам 6
4.4.1 calcCirc3P — окружность по трём точкам 6
4.4.2 calcCirc4P — окружность по четырём точкам 6
4.4.3 calcSizeXY — расстояние между точками 7
4.4.4 calcRot — поворот по двум точкам 7
4.4.5 calcAngle3P — угол и проекция точки на прямую 7
4.5 Проверка точки (checkPoint) 7
4.6 Проверка значения (checkValue) 7
5 Куда сохраняется результат 7
5.1 Поля результата 8
6 Примеры 9
6.1 Пример 1. Угол детали и начало координат (X=0, Y=0) 9
6.2 Пример 2. Диаметр круга с допуском 9
6.3 Пример 3. Размер с допуском 9
6.4 Пример 4. Расстояние между двумя кругами и его проверка 9

6.5 Пример 5. Поворот детали 10
7 Коды ошибок 10
8 Частые ошибки 10

1 Назначение

Модуль `WLAProbe` добавляет в WLMill команду **`M199(...)`** — автоматический пробинг прямо из управляющей программы. Одна строка G-кода измеряет деталь щупом, считает геометрию и отдаёт результат в переменную. Руками диалоги вызывать не нужно.

Что умеет:

- **замеры** — плоскость Z, точка XY, окружность по 3 и 4 точкам, размер по 2 точкам, поворот, угол заготовки;
- **установка** — координаты, поворот и Z по результату замера;
- **сохранение точки** — запись точки с явными координатами в список точек (`savePoint`);
- **расчёты** — окружность, расстояние и поворот по сохранённым точкам;
- **проверка** — контроль параметра точки или числового значения с допуском и остановка программы при выходе за допуск.

2 Установка

1. Скопируйте каталог `WLAProbe` в:

```
<каталог программы>/wlmillconfig/script/
```

2. Убедитесь, что рядом лежит каталог `WLProbe` — без него замеры не работают.
3. Подключите оба каталога в `mscript.js`:

```
SCRIPT.include("/WLAProbe/*")
SCRIPT.include("/WLProbe/*")
```

4. Всё: команда `M199` доступна в программах, а на панели инструментов появляется кнопка `WLAProbe`, которая собирает строку за вас.

3 Как устроена команда M199

```
M199(mode="probeCirc3P", x=40, y=25, diameter=20, depth=-1, type="inside",
save="H1")
```

Правила:

- параметры пишутся в скобках того же кадра, через запятую, как `имя=значение`;
- строки — в кавычках (`mode="probeZ"`), числа — без (`x=10`);
- регистр не важен: `probeZ` = `PROBEZ`, `X=10` = `x=10`;
- параметры не переносятся на следующий кадр — всё указывайте в одном `M199(...)`;
- координаты результатов замеров возвращаются в **машинной** СК (G53).

Если какая-то координата не задана, берётся текущее положение станка. Например, `M199(mode="probeXY", x=40, depth=-1)` поедет в точку `X=40` с **текущим** `Y`.

3.1 Система координат входных координат (cs)

cs	Смысл
`g53` (по умолчанию)	`x`, `y`, `z` заданы в машинной СК
`cur`	`x`, `y`, `z` заданы в текущей СК (пересчитываются в машинную)

3.2 Обязательная последовательность

Сначала один раз выполните замер плоскости, потом — остальные замеры:

```
M199(mode="probeZ", x=100, y=50, z_st=5, z_en=-5, result="PLANE")
```

Без этого режимы замера (`probe*`) вернут ошибку `-2`. Установки, расчёты и проверка базовой плоскости не требуют.

4 Режимы

4.1 Замеры

4.1.1 probeZ — плоскость

```
M199(mode="probeZ", x=100, y=50, z_st=5, z_en=-5, result="PLANE")
```

Щуп ищет плоскость **от `z_st` вниз до `z_en`**. `z_en` обязателен (ожидаемая поверхность), `z_st` — начальная высота поиска, по умолчанию текущая Z; значения должны различаться. Результат: `PLANE_Z`, `PLANE_X`, `PLANE_Y`.

4.1.2 probeXY — точка касания

```
M199(mode="probeXY", x=10, y=20, depth=-1, angle=45, result="P")
```

Точка касания в плоскости XY. `angle` — направление подвода в градусах. Результат: `P_X`, `P_Y`, `P_Z`, `P_ANGLE`.

4.1.3 probeCirc3P — окружность по трём точкам

```
M199(mode="probeCirc3P", x=40, y=25, diameter=20, depth=-1, type="inside",
save="H1", result="H")
```

`diameter` (или `radius`) — примерный диаметр/радиус; укажите ровно одно из них, больше нуля. `type` — `inside` (отверстие) или `outside` (выступ). Углы `a1`, `a2`, `a3` задают три точки; по умолчанию `0`, `120`, `-120`. **Все три угла должны быть разными** (0 и 360 — один угол). Результат: `H_CX`, `H_CY`, `H_DIAMETER`, `H_RADIUS`.

4.1.4 probeCirc4P — окружность по четырём точкам

```
M199(mode="probeCirc4P", x=40, y=25, diameter=20, depth=-1, angle=0, type="inside",
result="H")
```

То же, но четырьмя касаниями. `diameter` (или `radius`) — ровно одно, больше нуля; дополнительно `angle` — угол первой пары точек. Результат: `H_CX`, `H_CY`, `H_DIAMETER`, `H_RADIUS`, `H_ERROR` (расхождение диаметров проходов).

4.1.5 probeSize2P — размер по двум точкам

```
M199(mode="probeSize2P", x=60, y=30, size=40, depth=-1, angle=0, type="inside", re-
sult="S")
```

`size` — примерная ширина (паз) или диаметр, `angle` — направление измерения, `type` — `inside`/`outside`. Результат: `S_CX`, `S_CY`, `S_SIZE`, `S_ANGLE`.

4.1.6 probeRot — поворот по двум точкам

```
M199(mode="probeRot", x=60, y=30, size=40, depth=-1, angle=0, result="R")
```

`size` — расстояние от середины до точек касания (половина длины базы). Результат: `R_ANGLE` — угол наклона прямой.

4.1.7 probeQuad — угол заготовки

```
M199(mode="probeQuad", x=30, y=30, size=10, depth=-1, quadrant=1, type="inside",
result="Q")
```

`quadrant` — номер квадранта 1…4, `size` — расстояние отступа от угла. Результат: `Q_X`, `Q_Y`, `Q_QUADRANT`.

4.2 Установка (без перемещения)

4.2.1 setPosSC — координаты в точке

```
M199(mode="setPosSC", p="A", x=0, y=0, z=0)
```

Точка `p` из списка точек получает указанные координаты (хотя бы одну ось). Станок не перемещается.

4.2.2 setCurPosSC — текущие координаты

```
M199(mode="setCurPosSC", x=50, y=60)
```

Аналог `G92`: текущее положение получает указанные координаты.

4.2.3 setRotSC — поворот СК

```
M199(mode="setRotSC", p="ROT", angle=45)
```

Поворачивает систему координат на `angle` вокруг точки `p`.

4.2.4 setZprobe — Z по замеру

```
M199(mode="setZprobe", z=5)
```

Выполняет `G92 Z...`. При `cs="g53"` (по умолчанию, в том числе когда `cs` не указан) значение `z` считается машинной координатой и пересчитывается в локальную; при `cs="cur"` — `z` уже в текущей СК и применяется без пересчёта.

4.3 Сохранение точки (savePoint)

```
M199(mode="savePoint", p="POINT", x=5, y=10, z=2)
M199(mode="savePoint", p="POINT", x=5, y=10, z=2, cs="cur")
```

Сохраняет точку с именем `p` и заданными координатами в общий список точек (без замера и без перемещения). Если точка с таким именем уже есть, она перезаписывается целиком. Не указанные оси берутся из текущего положения. При `cs="cur"` координаты пересчитываются в машинную СК (`g53`), как и для всех остальных точек. Дополнительно можно указать `result` (запишет `ИМЯ_X`, `ИМЯ_Y`, `ИМЯ_Z`) и `comment`.

4.4 Расчёты по сохранённым точкам

Точки предварительно сохраняются параметром `save="ИМЯ"` в любом режиме замера.

4.4.1 calcCirc3P — окружность по трём точкам

```
M199(mode="calcCirc3P", p1="A", p2="B", p3="C", result="C")
```

Результат: `C_CX`, `C_CY`, `C_Z`, `C_DIAMETER`, `C_RADIUS` — как у `probeCirc3P`.

4.4.2 calcCirc4P — окружность по четырём точкам

```
M199(mode="calcCirc4P", p1="A", p2="B", p3="C", p4="D", result="C")
```

Результат: `C_CX`, `C_CY`, `C_Z`, `C_RADIUS`, `C_DIAMETER`, `C_ERROR`.

4.4.3 calcSizeXY — расстояние между точками

```
M199(mode="calcSizeXY", p1="A", p2="B", result="L")
```

Результат: `L_DX`, `L_DY`, `L_DZ`, `L_DISTANCE`, `L_DISTANCEXY`.

4.4.4 calcRot — поворот по двум точкам

```
M199(mode="calcRot", p1="A", p2="B", result="R")
```

Угол прямой от `p1` к `p2`. Результат: `R_ANGLE`, `R_DX`, `R_DY`, `R_DISTANCE`.

4.4.5 calcAngle3P — угол и проекция точки на прямую

```
M199(mode="calcAngle3P", p1="A", p2="B", p3="C", result="G")
```

`p1` и `p2` задают прямую (базу) в направлении `p1 → p2`, `p3` — точка, которая проецируется на неё. Результат: `G_ANGLE` (наклон прямой), `G_X`, `G_Y`, `G_Z` (проекция `p3` на прямую), `G_DIST` (расстояние от `p3` до прямой).

> Порядок `p1`/`p2` меняет знак угла на 180° — указывайте их по направлению измерения. `p3` на угол не влияет.

4.5 Проверка точки (checkPoint)

```
M199(mode="checkPoint", name="H1", param="diameter", value=20, tolerance=0.05, result="CHK")
```

Сравнивает параметр точки с номиналом. `param` — `x`, `y`, `z`, `cx`, `cy`, `diameter`, `radius`, `size`, `distance`, `distanceXY`, `angle`, `error`. При выходе за допуск показывает вопрос «Продолжить / Остановить»; при «Остановить» программа останавливается (код `-21`). Результат: `CHK_NOMINAL`, `CHK_ACTUAL`, `CHK_DEVIATION`, `CHK_TOLPLUS`, `CHK_TOLMINUS`.

4.6 Проверка значения (checkValue)

```
M199(mode="checkValue", name=#H_DIAMETER, value=20, tolerance=0.05, result="CHK")
```

Сравнивает произвольное число с номиналом — обычно переменную, оставленную предыдущим замером (`#H_DIAMETER` подставляется парсером G-кода). Допуски те же: `tolerance` (±) либо раздельные `tolPlus`/`tolMinus`. Результат: `CHK_NOMINAL`, `CHK_ACTUAL`, `CHK_DEVIATION`, `CHK_TOLPLUS`, `CHK_TOLMINUS`.

5 Куда сохраняется результат

Три независимых параметра:

Параметр	Что делает
`result="ИМЯ"`	Числа из результата попадают в переменные `ИМЯ_ПОЛЕ`. Используются в G-коде как `#ИМЯ_ПОЛЕ`
`save="ИМЯ"`	Точка сохраняется в общий список точек (доступна для `p1`…`p4`, `p`, `name`)
`saveFile="файл.csv"`	Результат дописывается строкой в CSV-файл

Пример использования переменной:

```
M199(mode="probeCirc4P", x=40, y=25, diameter=20, depth=-1, type="inside", result="HOLE")
G0 X#HOLE_CX Y#HOLE_CY
```

5.1 Поля результата

Каждый режим записывает в переменную свои числа. Полное имя переменной — `ИМЯ_ПОЛЕ` (например, `HOLE_DIAMETER`), где `ИМЯ` — это `result`, а `ПОЛЕ` берётся из таблицы ниже.

Поле	В каких режимах	Описание
`X`, `Y`	`probeZ`, `probeXY`, `probeQuad`, `calcAngle3P`, `savePoint`	Координата точки касания, вершины угла, проекции или сохранённой точки (машинная СК)
`Z`	`probeZ`, `probeXY`, `calcCirc3P`, `calcCirc4P`, `calcAngle3P`, `savePoint`	Координата Z: плоскость (`probeZ`), уровень касания (`probeXY`), Z точки расчёта или сохранённой точки
`CX`, `CY`	`probeCirc3P`, `probeCirc4P`, `probeSize2P`, `calcCirc3P`, `calcCirc4P`	Центр окружности или середина размера по X и Y
`DIAMETER`	`probeCirc3P`, `probeCirc4P`, `calcCirc3P`, `calcCirc4P`	Измеренный/рассчитанный диаметр
`RADIUS`	`probeCirc3P`, `probeCirc4P`, `calcCirc3P`, `calcCirc4P`	Радиус (половина диаметра)
`SIZE`	`probeSize2P`	Измеренный размер (ширина паза или выступа)
`ANGLE`	`probeXY`, `probeSize2P`, `probeRot`, `calcRot`, `calcAngle3P`	Угол в градусах: направление касания, поворота или наклона прямой
`DX`, `DY`, `DZ`	`calcSizeXY`, `calcRot`	Приращения между двумя точками по осям (`DY` и `DZ` в `calcRot` нет)
`DISTANCE`	`calcSizeXY`, `calcRot`	Полное расстояние между точками
`DISTANCEXY`	`calcSizeXY`	Расстояние между точками в плоскости XY
`DIST`	`calcAngle3P`	Расстояние от третьей точки до прямой
`QUADRANT`	`probeQuad`	Номер квадранта угла (1…4)
`ERROR`	`probeCirc4P`, `calcCirc4P`	Погрешность: расхождение диаметров проходов или отклонение радиуса
`NOMINAL`	`checkPoint`, `checkValue`	Номинальное значение проверяемого параметра
`ACTUAL`	`checkPoint`, `checkValue`	Фактическое значение проверяемого параметра
`DEVIATION`	`checkPoint`, `checkValue`	Отклонение: фактическое минус номинальное

Поле	В каких режимах	Описание
`TOLPLUS`, `TOLMINUS`	`checkPoint`, `checkValue`	Применённые допуски в плюс и в минус

Строковые поля (`mode`, `type`, `status`, `message`) в переменную не попадают — сохраняются только числа.

6 Примеры

Каждый пример начинается с `probeZ` — он обязателен перед всеми XY-замерами.

6.1 Пример 1. Угол детали и начало координат (X=0, Y=0)

```
M199(mode="probeZ", x=50, y=50, z_st=5, z_en=-5, result="BASE")

(вершина угла: внутренний угол, квадрант 1)
M199(mode="probeQuad", x=30, y=30, size=10, depth=-1, quadrant=1, type="inside",
     save="CORNER", result="Q")

(вершина угла — начало координат X=0, Y=0)
M199(mode="setPosSC", p="CORNER", x=0, y=0)

(обнуляем Z по поверхности)
M199(mode="setZprobe", z=#BASE_Z)
```

6.2 Пример 2. Диаметр круга с допуском

```
M199(mode="probeZ", x=0, y=0, z_st=5, z_en=-5, result="BASE")

M199(mode="probeCirc3P", x=40, y=25, diameter=20, depth=-1, type="inside",
     save="HOLE", result="H")

M199(mode="checkPoint", name="HOLE", param="diameter", value=20,
     tolerance=0.05, result="CHK")
```

При выходе за допуск программа остановится (код `-21`).

6.3 Пример 3. Размер с допуском

```
M199(mode="probeZ", x=0, y=0, z_st=5, z_en=-5, result="BASE")

(ширина паза по двум точкам)
M199(mode="probeSize2P", x=60, y=30, size=40, depth=-1, angle=0, type="inside",
     save="SLOT", result="S")

M199(mode="checkPoint", name="SLOT", param="size", value=40,
     tolPlus=0.1, tolMinus=0.05, result="CHK")
```

Для выступа — `type="outside"`.

6.4 Пример 4. Расстояние между двумя кругами и его проверка

```
M199(mode="probeZ", x=0, y=0, z_st=5, z_en=-5, result="BASE")

M199(mode="probeCirc3P", x=40,  y=25, diameter=20, depth=-1, type="inside",
save="H1", result="R1")
M199(mode="probeCirc3P", x=140, y=25, diameter=20, depth=-1, type="inside",
save="H2", result="R2")

M199(mode="calcSizeXY", p1="H1", p2="H2", save="DIST", result="D")

M199(mode="checkPoint", name="DIST", param="distanceXY", value=100,
```

```
      tolerance=0.1, result="CHK")
```

6.5 Пример 5. Поворот детали

```
M199(mode="probeZ", x=0, y=0, z_st=5, z_en=-5, result="BASE")

M199(mode="probeRot", x=60, y=30, size=40, angle=0, save="ROT", result="R")
M199(mode="setRotSC", p="ROT", angle=#R_ANGLE)
```

7 Коды ошибок

При ошибке станок останавливается (`MACHINE.reset()`).

Код	Причина
`-1`	Не указан режим (`mode`)
`-2`	Сначала не выполнен замер Z (`probeZ`)
`-3`	Датчик щупа не включён
`-4`	Замер или расчёт не удался (в том числе `probeZ` без `z_en` или при `z_st == z_en`)
`-5`	Неизвестный режим
`-7`	Не задана ни одна координата или отсутствует `z`
`-8`	Не указан обязательный параметр точки или угла
`-9`	Точка с указанным именем не найдена
`-10`	Не указан `type`/`size`/`quadrant`/`diameter`/`radius` или совпадают углы `a1`…`a3`
`-21`	Проверка `checkPoint`/`checkValue` не пройдена, выбрано «Остановить»
`-999`	Внутренняя ошибка скрипта

8 Частые ошибки

- **`M199` без параметров** — комментарии не переносятся на следующий кадр, всё указывайте в одном вызове.
- **Забыли `probeZ`** — все режимы `probe*` требуют предварительного замера плоскости.
- **Обязательные параметры**: `probeCirc3P`/`probeCirc4P` — `diameter` или `radius` (ровно одно, > 0); `probeSize2P`/`probeRot`/`probeQuad` — `size`; `type` — для окружностей/размера/угла; `quadrant` — для `probeQuad`.
- **Углы `probeCirc3P` должны различаться**, иначе ошибка `-10`.
- **Результаты — в машинной СК**, независимо от `cs`.

Полное описание (внутреннее устройство, справочник функций, контракт с `WLProbe.mscr`) — в документе **WLAProbe**.