Trigger Reference Guide

by XGASOFT

1. Welcome to Trigger - Better Trigonometry for GameMaker Studio

Trigonometry is an important part of many facets of game development. Enter Trigger: a suite of efficient, easy-to-learn trigonometry functions from XGASOFT which you can use **for free**.

Trigger functions store the results of expensive calculations in memory so that they can be re-used until a new calculation is necessary. Since many uses of trigonometry will use the same angle to calculate X and Y coordinates, Trigger can cut performance costs of trigonometric functions while also being simpler to use. Trigger makes learning trigonometry programming easy by providing multiple functions to suit different styles. Each function is named and explained clearly.

Trigger is **fast for experts** and **easy for beginners**!

In this reference guide, you'll learn…

- How to use each available function
- Individual script arguments and what they mean

To get started, choose a topic from the menu to the leftabove to learn more.

2. Buy Now(https://xgasoft.itch.io/trigger)

3. Download PDF(https://docs.xgasoft.com/wp-content/uploads/sites/2/trigger-reference-guide-17.pdf)

4. Version History

1.0.0

- Initial release

5. VNgen Reference Guide

In simple terms, trigonometry is the study of triangles. In programming, it is often used to determine the 2D coordinates of points which have been rotated a certain distance away from another point. You may have a mental image of a line being drawn from point A to point B, creating an angle. While calculating this angle is the objective we're trying to achieve, how we get there is by imagining not just a line, but a *triangle* instead—two flat lines following the X and Y axis like normal, while the angle is the triangle's hypotenuse.

Trigonometry demonstrates that it is possible to determine the position, orientation, and length of a triangle's hypotenuse based on its other two sides. While the formulae required are logically quite simple, actually calculating them is not. For programs that heavily rely on trigonometry, having an efficient way to perform these calculations is important. And for newcomers who may not yet be used to working with trigonometry in programming, making them easy to understand is equally so.

Trigger fundamentally only has three functions: `point_rot_prefetch`, `point_rot_x`, and `point_rot_y`. However, by applying the same basic principles in different ways, users may find Trigger's other functions to be easier to use for their particular use-cases. In this reference guide, we'll examine each one in detail.

5.1. The "point_rot_prefetch" Function

Syntax:

```
point_rot_prefetch(deg);
```

Argument	Type	Description
`deg`	real	Angle to calculate sine and cosine, in degrees

Description:

Pre-calculates the sine and cosine of an angle in degrees, which can then be used by `point_rot_x` and `point_rot_y` (or other variants) without re-calculating. This is highly useful for improving performance when calculating multiple points based on the same rotation.

Note that setting an angle in `point_rot_x` and `point_rot_y` will override this script's calculations with a new sine and cosine. For the same reason, running this script is not necessary so long as the first instance of `point_rot_x` or `point_rot_y` is supplied with an angle instead. However, this script can still be quite useful for calculating an angle in a different event than the event in which `point_rot_x` or `point_rot_y` is run.

`Example:`

```
point_rot_prefetch(90);
x = point_rot_x(5, 10);
y = point_rot_y(5, 10);
```

5.2. The "point_rot_x" Function

Syntax:

```
point_rot_x(x, y, [deg]);
```

Argument	Type	Description
x	real	Horizontal distance from the rotation center point
y	real	Vertical distance from the rotation center point
[deg]	real	*Optional:* Angle of rotation in degrees

Description:

Returns the X component of a point the given distance away, rotated by the given angle in degrees. (Center point is assumed as 0.)

Supplying an angle is optional. As calculating the sine and cosine of angles is costly to performance, these values are stored in memory for use with further instances of trigonometry functions based on the same angle. If no angle is supplied, the previous angle's sine and cosine will be used instead. This is highly useful for improving performance when calculating multiple points based on the same rotation.

Example:

```
x = 128 + point_rot_x(64, 64, image_angle);
y = 128 + point_rot_y(64, 64);
```

5.3. The "point_rot_y" Function

Syntax:

```
point_rot_y(x, y, [deg]);
```

Argument	Type	Description
x	real	Horizontal distance from the rotation center point
y	real	Vertical distance from the rotation center point
[deg]	real	*Optional:* Angle of rotation in degrees

Description:

Returns the Y component of a point the given distance away, rotated by the given angle in degrees. (Center point is assumed as 0.)

Supplying an angle is optional. As calculating the sine and cosine of angles is costly to performance, these values are stored in memory for use with further instances of trigonometry functions based on the same angle. If no angle is supplied, the previous angle's sine and cosine will be used instead. This is highly useful for improving performance when calculating multiple points based on the same rotation.

Example:

```
x = 128 + point_rot_x(64, 64, image_angle);
y = 128 + point_rot_y(64, 64);
```


5.4. The "dist_rot_x" Function

Syntax:

```
dist_rot_x(dist, [deg]);
```

Argument	Type	Description
`dist`	real	Horizontal distance from the rotation center point
`[deg]`	real	*Optional:* Angle of rotation in degrees

Description:

Returns the X component of a point the given distance away, rotated by the given angle in degrees. (Center point is assumed as 0.)

Supplying an angle is optional. As calculating the sine and cosine of angles is costly to performance, these values are stored in memory for use with further instances of trigonometry functions based on the same angle. If no angle is supplied, the previous angle's sine and cosine will be used instead. This is highly useful for improving performance when calculating multiple points based on the same rotation.

`Example:`

```
x = 128 + dist_rot_x(64, image_angle);
y = 128 + dist_rot_y(64);
```

5.5. The "dist_rot_y" Function

Syntax:

```
dist_rot_y(dist, [deg]);
```

Argument	Type	Description
`dist`	real	Vertical distance from the rotation center point
`[deg]`	real	*Optional:* Angle of rotation in degrees

Description:

Returns the Y component of a point the given distance away, rotated by the given angle in degrees. (Center point is assumed as 0.)

Supplying an angle is optional. As calculating the sine and cosine of angles is costly to performance, these values are stored in memory for use with further instances of trigonometry functions based on the same angle. If no angle is supplied, the previous angle's sine and cosine will be used instead. This is highly useful for improving performance when calculating multiple points based on the same rotation.

`Example:`

```
x = 128 + dist_rot_x(64, image_angle);
y = 128 + dist_rot_y(64);
```


5.6. The "vec_rot_x" Function

Syntax:

```
vec_rot_x(x1, y1, x2, y2, [deg]);
```

Argument	Type	Description
x1	real	Horizontal center point
y1	real	Vertical center point
x2	real	Horizontal distance from the rotation center point
y2	real	Vertical distance from the rotation center point
[deg]	real	*Optional:* Angle of rotation in degrees

Description:

Returns the X component of a point the given distance away from the given center point, rotated by the given angle in degrees (or in other words, the X component of the tip of a rotated line).

Supplying an angle is optional. As calculating the sine and cosine of angles is costly to performance, these values are stored in memory for use with further instances of trigonometry functions based on the same angle. If no angle is supplied, the previous angle's sine and cosine will be used instead. This is highly useful for improving performance when calculating multiple points based on the same rotation.

Example:

```
x = vec_rot_x(128, 128, 64, 64, image_angle);
y = vec_rot_y(128, 128, 64, 64);
```

5.7. The "vec_rot_y" Function

Syntax:

```
vec_rot_y(x1, y1, x2, y2, [deg]);
```

Argument	Type	Description
x1	real	Horizontal center point
y1	real	Vertical center point
x2	real	Horizontal distance from the rotation center point
y2	real	Vertical distance from the rotation center point
[deg]	real	*Optional:* Angle of rotation in degrees

Description:

Returns the Y component of a point the given distance away from the given center point, rotated by the given angle in degrees (or in other words, the Y component of the tip of a rotated line).

Supplying an angle is optional. As calculating the sine and cosine of angles is costly to performance, these values are stored in memory for use with further instances of trigonometry functions based on the same angle. If no angle is supplied, the previous angle's sine and cosine will be used instead. This is highly useful for improving

performance when calculating multiple points based on the same rotation.

Example:

```
x = vec_rot_x(128, 128, 64, 64, image_angle);
y = vec_rot_y(128, 128, 64, 64);
```

6. Special Thanks

Patreon Credits

This product is made possible by the generous support of XGASOFT patrons on Patreon. Every contribution counts, no matter how big or small. To all fans and patrons around the globe, thanks for being a part of XGASOFT's story!

Very special thanks goes out to:

Patreon "Enthusiasts"

Marvin Mrzyglod

Patreon "Developers"

AshleeVocals

AutumnInAprilArt

Daniel Sato

Meyaoi Games

Patreon "Gamers"

Kampmichi (Forgers of Novelty)

Patreon "Supporters"

Adam Miller (Actawesome)

Cosmopath

D Luecke

Alex Lepinay

Tarquinn J Goodwin

Creative Credits

XGASOFT is also privileged to work with other creators from around the world, in some cases on the very developer tools used to make XGASOFT products possible.

Special credit goes out to the following talents for their contributions:

VNgen Demo Voiceover

Kanen *(as Miki and Mei)*

7. End-User License Agreement ("EULA")

LAST UPDATED: 02/14/2019

We know that reading EULAs isn't very exciting, but this is important. Please take your time to review and ensure you understand the terms of this document before proceeding to use XGASOFT products in your own work.

If you have any questions or concerns about the terms outlined in this document, please feel free to contact us at contact@xgasoft.com or by visiting our Contact & Support page.

LICENSE AGREEMENT

This License Agreement (the "Agreement") is entered into by and between XGASOFT (the "Licensor"), and you (the "Licensee"). This agreement is legally binding, and becomes effective when you purchase and/or download a free product from XGASOFT or authorized third-party distributors. If you do not agree to the terms of this Agreement, do not purchase, download, or otherwise use XGASOFT products.

In order to accept this Agreement, you must be at least eighteen (18) years of age or whatever age is of legal majority in your country. Otherwise, you must obtain your parent's or legal guardian's approval and acceptance of this Agreement in your stead. XGASOFT accepts no liability for your failure to meet this requirement.

XGASOFT delivers content through authorized third-party distributors, each of which may require its own separate End-User License Agreement ("EULA"). XGASOFT accepts no liability for the terms of any third-party agreements, nor for your failure to meet them.

STANDARD LIFETIME LICENSE

This is a license, not a sale. XGASOFT retains ownership of all content (including but not limited to any copyright, trademarks, brand names, logos, software, images, animations, graphics, video, audio, music, text, and tutorials) comprising digital products and services offered by XGASOFT (the "Property"). All rights not expressly granted are reserved by XGASOFT.

Subject to your acceptance of the terms of this Agreement, XGASOFT grants you a worldwide, revocable, non-exclusive, non-transferable, and **perpetual** license to download, embed, and modify for your own purposes

XGASOFT Property solely for incorporation with electronic applications and other interactive media, including both commercial and non-commercial works, wherever substantial value has been added by you.

Any source code included as part of XGASOFT Property must be compiled prior to redistribution as an incorporated work, whether for commercial or non-commercial purposes.

PATREON LIMITED LICENSE

When you register as a recurring financial supporter of XGASOFT through Patreon (Patreon, Inc.), XGASOFT may provide free access to XGASOFT Property as a reward, subject to the terms of each contribution tier. This is a privilege, not a right.

XGASOFT retains ownership of all content (including but not limited to any copyright, trademarks, brand names, logos, software, images, animations, graphics, video, audio, music, text, and tutorials) comprising digital products and services offered by XGASOFT (the “Property”). All rights not expressly granted are reserved by XGASOFT.

Subject to your acceptance of the terms of this Agreement, XGASOFT grants you a worldwide, revocable, non-exclusive, non-transferable, and **temporary** license to download, embed, and modify for your own purposes XGASOFT Property solely for incorporation with electronic applications and other interactive media, including both commercial and non-commercial works, wherever substantial value has been added by you.

Any source code included as part of XGASOFT Property must be compiled prior to redistribution as an incorporated work, whether for commercial or non-commercial purposes.

This license shall remain effective for the duration of your subscription to XGASOFT through Patreon. In the event that you cancel or reduce your contribution to a lower tier not qualifying for free access to XGASOFT Property, this license will be considered revoked and void for any and all public commercial and non-commercial activities. In order to continue using XGASOFT Property publicly, you must purchase a standard lifetime license.

This limitation shall not be applied retroactively, so that any existing, complete, and publicly available commercial and non-commercial properties using XGASOFT Property will not be considered in violation of this agreement.

SINGLE-USER

This Agreement grants one (1) user an applicable license to use XGASOFT Property on unlimited devices. This license may not be transferred, shared with, or sold to other users.

However, you, the Licensee, may use XGASOFT Property along with a team or company of collaborators wherever substantial value has been added by you.

This limitation does not extend a license to other users. For any works unrelated to you, collaborators must purchase separate licenses.

MODIFICATIONS

In accordance with the terms of this Agreement, you may freely modify, or alter the functionality of XGASOFT Property exclusively for your own use.

Modifying the Property will not terminate your license, however XGASOFT cannot guarantee the quality and functionality of modified versions of the Property, nor its compatibility with other products.

XGASOFT accepts no liability for any loss or damage incurred by the modified Property, and reserves the right to refuse technical support for the modified Property.

Modifications made to XGASOFT Property in no way represent a change of ownership of the Property.

You may not reverse-engineer XGASOFT Property for the purpose of commercial exploitation which may be in competition with XGASOFT.

MUTABILITY

License fees are determined for each product and service on a case-by-case basis, and XGASOFT reserves the right to change fees on the Property with or without prior notice.

XGASOFT reserves the right to modify, suspend, or terminate this Agreement, the Property, or any service to which it connects with or without prior notice and without liability to you, the Licensee.

LIABILITY

By using XGASOFT Property, you agree to indemnify and hold harmless XGASOFT, its employees, and agents from and against any and all claims (including third party claims), demands, actions, lawsuits, expenses (including attorney’s fees) and damages (including indirect or consequential loss) resulting in any way from your use or reliance on XGASOFT Property, any breach of terms of this Agreement, or any other act of your own.

This limitation will survive and apply even in the event of termination of this Agreement.

GOVERNING LAW AND JURISDICTION

This Agreement shall be governed by and interpreted according to the laws of the United States of America and the State of Kansas.

If any provision of this Agreement is held to be unenforceable or invalid, such provision will be changed and interpreted to accomplish the objectives of such provision to the greatest extent possible under applicable law, and the remaining provisions will continue in full force and effect.

CONCLUSION

This document contains the whole agreement between XGASOFT and you, the Licensee, relating to the Property and licenses thereof and supersedes all prior Agreements, arrangements and understandings between both parties regarding XGASOFT Property and licenses.

