The Official ALTTPR Race-legal Auto-tracker Specification

Version 1.2 - May 19th, 2021

Preface

This document guides auto-tracker developers on what is and is not acceptable, in terms of race-legal auto-tracking for ALTTPR, as outlined by the ALTTPR Racing Council. This specification doc contains the rules for creating a race-legal auto-tracker, including permitted memory addresses, behavior, and features.

Anything not covered in this document is, by default, considered not permitted. If you have any questions regarding this specification document or auto-tracker development, please ask in the #general-dev channel on the main randomizer discord (https://discord.gg/alttprandomizer) for assistance. If you have questions regarding the overall rules on auto-tracker legality for racing, please ask in the #racing-council channel in the main randomizer discord.

Note from ALTTPR Racing Council Regarding Auto-tracking

Announcement posted 2020-08-14

The council has voted on auto-tracking. Before we get into specifics, we would like to stress an important caveat - **Auto-tracking must react to in-game information in the same way the player would. A tracker cannot provide information to a runner that the runner has not yet experienced in the game.**

The council has decided that it is legal to have a program automatically tracking the following items on behalf of a runner:

Items and Keys (including the collection of a dungeon prize)
Cleared map locations (Note: This does not include entrance tracking)

The council has determined that the following items are currently **forbidden** from auto-tracking:

Dungeon Item Counts (How many non-dungeon items you've received in a dungeon, or chest counts in keysanity)
Dungeon Prizes (Map-check crystal/pendants)
Item Spheres
Entrances

The council did not feel it necessary to review the existing functionalities of tracking, so it remains legal for a tracking program (auto or not) to indicate what you logically have access to given the items in your inventory. Any other functions of auto-tracking are banned unless explicitly stated by the racing council. Ensuring conformance with this policy is ultimately the responsibility of the player using the auto tracker. This announcement does not include ChatHUD. ChatHUD is still banned from competitive play.

If you have questions about the legality of features, feel free to bring it up in #racing-council. If you have questions about the development of auto-trackers, please bring your questions to #general-dev. As a reminder, this applies to races on RTGG, SRL, and SRTV. It's up to the Admins of the Ladder whether they wish to allow auto-tracking.

Update posted 2020-10-18

Here's a pingless update based off of the recent autotracking discussion.

First item: To clarify the autotracking dev document, a "Checked Location" means the item in that location has been acquired. An auto-tracker can mark the collection of an item, but shouldn't make any effort to determine if a scoutable item has been scouted or not.

Second item: Any porting of the HUD into an auto-tracker is forbidden. This includes bomb/arrow/rupee counts, equipped item, and heart piece counts. Additionally, bottle contents are currently forbidden from autotracking.

If there are any questions related to legality, feel free to ask away in #this-is-racing-council channel. If there are any questions related to autotracker development, feel free to ask in #general-dev.

Specification

Definitions

Checked location - A chest that has been opened, a NPC in which you've received the NPC's item, or a freestanding location where you've actually received the item at that location. Simply "viewing" the item at a location does not count as checked.

Dungeons location - A chest, NPC, or overworld viewable item (such as a torch) that is located within a dungeon. The dungeons are:

- Hyrule Castle (including the Dark Cross and Sewers chests)
- Eastern Palace
- Desert Palace
- Tower of Hera
- Aghanim's (Castle) Tower
- Palace of Darkness
- Swamp Palace
- Skull Woods
- Thieves Town
- Ice Palace
- Misery Mire
- Turtle Rock
- Ganon's Tower

Permitted SNES memory addresses

Auto-trackers are allowed to read the following memory addresses, and the internal ROM header (https://sneslab.net/wiki/SNES_ROM_Header).

0xFFB0 to 0xFFDF (SNES internal rom header)
0x7E0000 to 0x7E00B0
0x7E010B (strictly for determining current MSU-1 track that is playing)
0x7EF000 to 0x7EF500
0x308210 to 0x30823F

All memory addresses are in the SNES LoROM address space. Auto-trackers may read any allowed address, any memory mirrors of that address, or the equivalent PC offset directly from the ROM file.

Reading memory outside the above memory addresses is prohibited.

You can find a detailed map of ALTTP's SRAM at http://alttp.run/hacking/index.php?title=SRAM_Map. A map of rando-specific SRAM addresses can be found at http://alttp.mymm1.com/wiki/ALTTPR_SRAM_Map.

Permitted Behavior and Features

Auto-trackers may do the following, and only the following:

1. Record and track any item collected by a player, or record items already pre-equipped (starting boots, sword, flute, etc.).
2. Record and track any small or big keys collected by a player.
    a. A patch is currently in development to provide a small key count in SRAM. If we implement that patch in the live randomizer, auto-tracker developers will be allowed to use that table to identify a count of small keys from chests. This spec will be updated if that table lies outside of the currently permitted memory addresses.
3. Record and track defeated dungeon bosses.
4. Record and track checked locations that are **not** in a dungeon, including non-dungeon underworld caves and indoor areas. The tracker **cannot** track checks inside a dungeon.
5. Record and track when the player starts playing the game.
6. Record and track if the player has at least one bomb, or at least one arrow.
7. Record and track when the player has completed the goal (warped to triforce room).
    a. The memory address 0x7E0010 is set to 0x19 at game completion.
    b. The tracker may display game statistics stored in SRAM **after** the game has concluded.
8. Read the currently playing MSU-1 track for on-stream display or other similar use (using 0x7E010B). This information cannot be used to infer if a dungeon is a pendant or crystal.

Auto-trackers may **not** do anything else not listed above. Prohibited behaviors include, but is not limited to:

1. Record or track any information the player would not have reasonable access to view.
2. Record or track checked locations inside a dungeon, including presenting this information as the number of remaining "chests."
3. Record or track information on pendants and crystal locations, regardless of the source of information.
4. Record or track any entrance locations, or where they lead.
5. Record or track the "sphere," or logical progression, of an item.
6. Record or track any information on the HUD. This includes bomb/arrow/rupee counts, equipped item, and heart piece counts. Additionally, current bottle contents are forbidden from autotracking.
7. Anything else not explicitly defined as a permitted auto-tracking behavior.

This spec does not prohibit any tracker abilities that were allowed for manual tracking. Entrance tracking is still permitted, for example, as long as it's done manually. This rule includes showing logic on a map tracker.

There is no limit on the response-time of auto-tracking. Trackers are not required to ensure an equal response-time across all supported platforms. There may be technical differences between platforms that may result in an inconsistent experience when it comes to the responsiveness of an auto-tracker.

Creating Non-Compliant Auto-Tracking Features

We still encourage auto-tracker developers to develop whatever auto-tracking features they want to create, for non-race usage. We don't want to limit our community devs' creativity in any fashion, and we do not wish for this specification to do so. If you're going to expand auto-tracking beyond what this specification provides, please do so!

If you create features for your auto-tracker that will not conform to the specification, we ask the following:

1) Add a conspicuous button or toggle that sets the tracker into a "Race Mode." "Race Mode" should limit the functionality to what is permitted by this specification doc.
2) The auto-tracker should read the race flag at memory address 0x308213 and automatically enable a race-compliant mode (this value will be 0x00 for off, and 0x01 for on). This feature will help players ensure the tracker's user is not accidentally violating the auto-tracker rules while racing. You may allow the user to override this option, as long as a warning is displayed.

Responsibility for Compliance

In the end, it is the responsibility of the player to ensure the auto-tracker they're using complies with this specification document. Testing is encouraged to ensure the tracker does not reveal information the player should not have. If the tracker violates the auto-tracking rules, it could result in the user's disqualification in the race they're playing.

Change Log

2020-08-13 - Synack - Initial draft
2020-08-15 - Synack - First release (version 1.0)
2020-08-15 - Synack - Added a link to a rando-specific map of SRAM.
2020-08-16 - Synack - Clarified that tracking equipment the player starts with is also permitted. (version 1.0.1)
2020-10-21 - Synack - Added definitions of "checked location" and "dungeon location" to the document. Also clarified that tracking HUD data was not permitted. (version 1.1)
2020-10-22 - Synack - Clarified tracking of bombs/arrows
2021-05-19 - Synack - Allowed reading of an offset that has the currently-playing MSU- track (version 1.2)