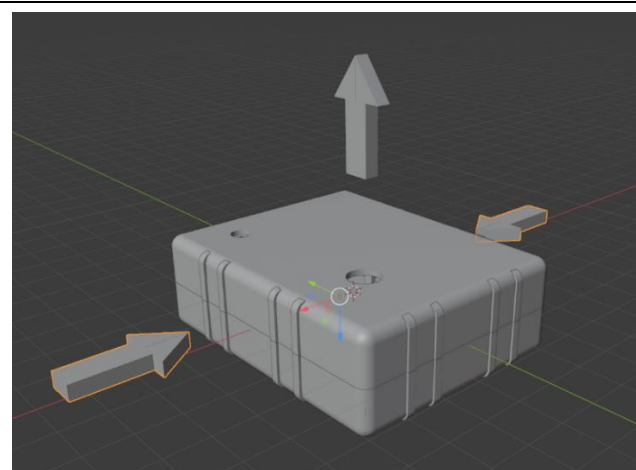


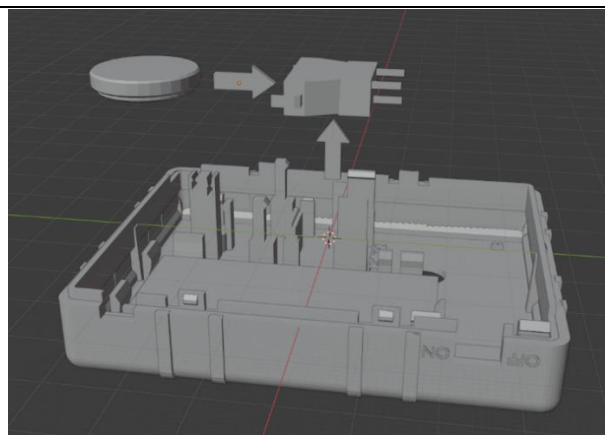
# e スポーツ関係者やゲーム開発に携わるプロフェッショナル向け測定機器

## ユニバーサル遅延測定器 v1.0

### 電池装着方法

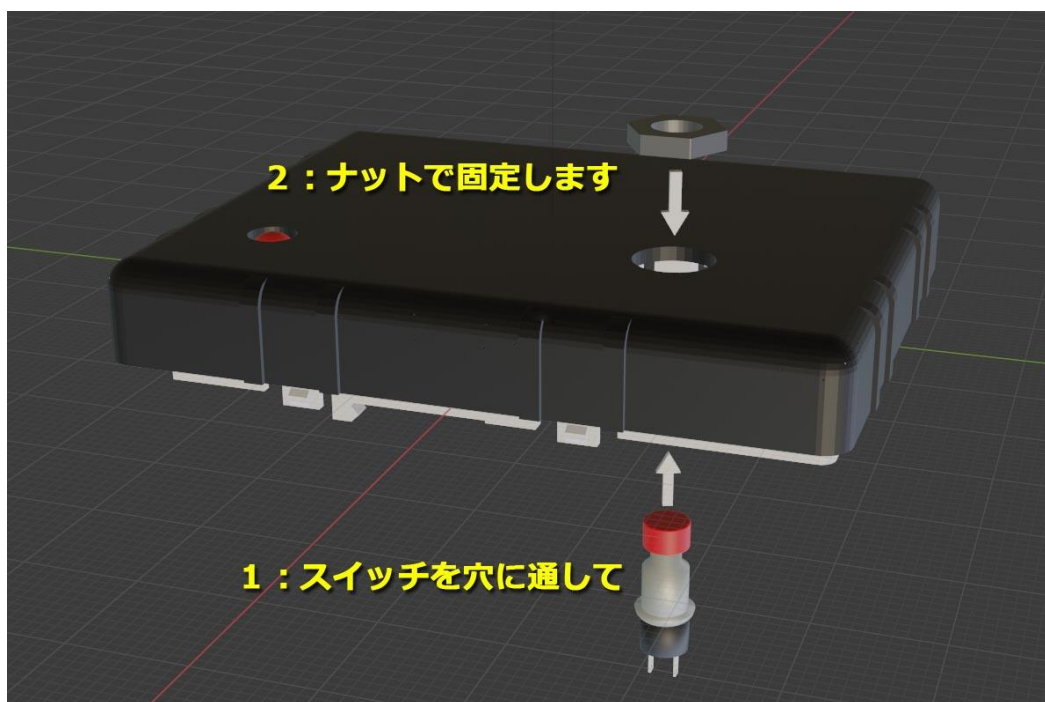


ケースの開け方：  
穴の開いている側を両側から押して開きます

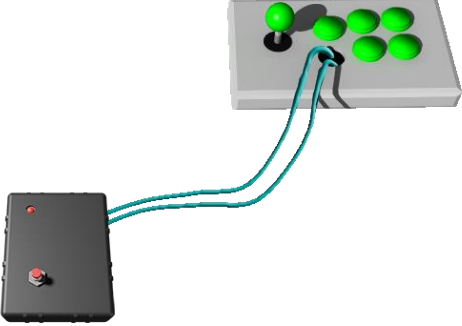



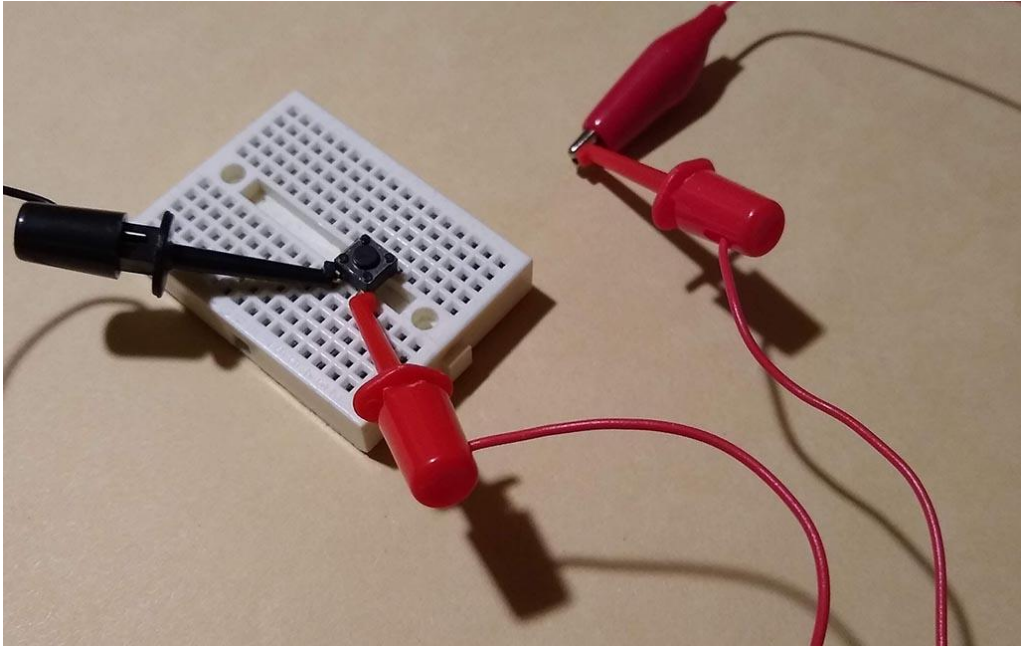
電池装着方法：  
電池ホルダーをケースから外して CR2032 を装着  
します

### 組み立て方法



## 接続例

	
<p>USB ゲームコントローラの内部にクリップを挿入します</p>	<p>スイッチの+/-それぞれの端子にクリップを接続します</p>


<p>マウスのスイッチのような小さな部品と繋ぐには IC クリップ等があると便利です</p>

## 使用方法



ケース横側にあるスライドスイッチを ON で使用可能になります

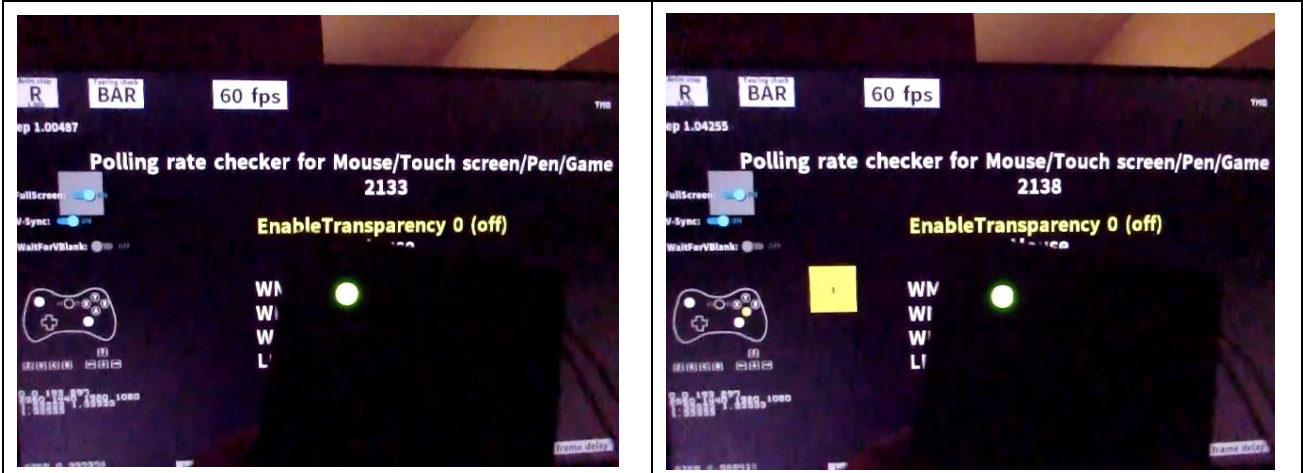
プッシュスイッチを ON にする事で内臓 LED 点灯回路とケーブルの先に接続された回路を同時に ON にします。

**電池の消耗は早いので、使用の際は予備の電池を準備しておくことを推奨します。**

## 撮影

スマホやデジタルカメラを使用して測定器とモニターを動画撮影します

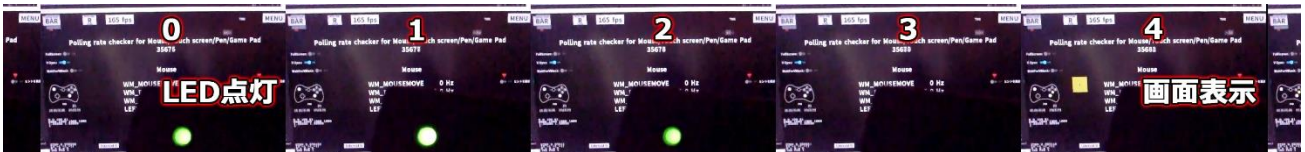
120fps 以上の高速撮影ができるものを推奨



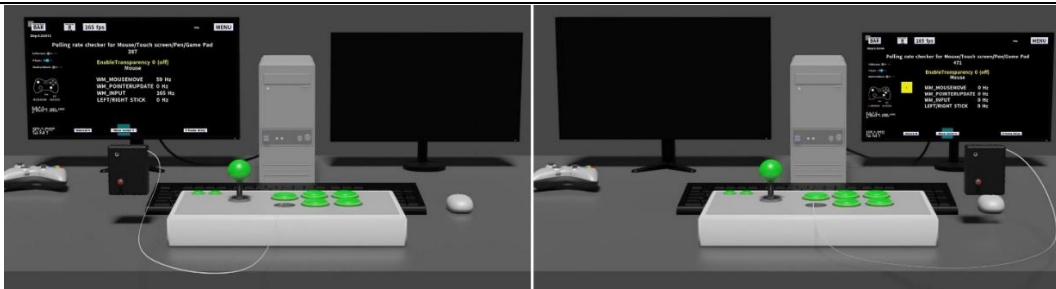
プッシュスイッチを ON にして LED が点灯したコマからカウントを開始します →

スイッチを ON にした情報が画面に表示されるまでのコマ数が、測定対象のレスポンスです

撮影した動画の確認は Windows 環境の場合、1フレーム毎コマ送りのできる PotPlayer や VLC media player 等の再生ソフトがあると便利です。



## 測定



	左	右
1	10	10
2	10	9
3	10	9
4	11	8
5	12	9
計	53	45
60fps換算	5.3	4.5

2 台のモニターを 120fps 動画で撮影した場合の例です。それぞれ 3 秒ぐらいで 5 回スイッチを押した動画を撮影し、LED が光ってからその情報が画面に反映されるまでをコマ送りして表にまとめ平均を求めます。レスポンスが 60fps 換算で 5.3 と 4.5 フレームであることが分かります。

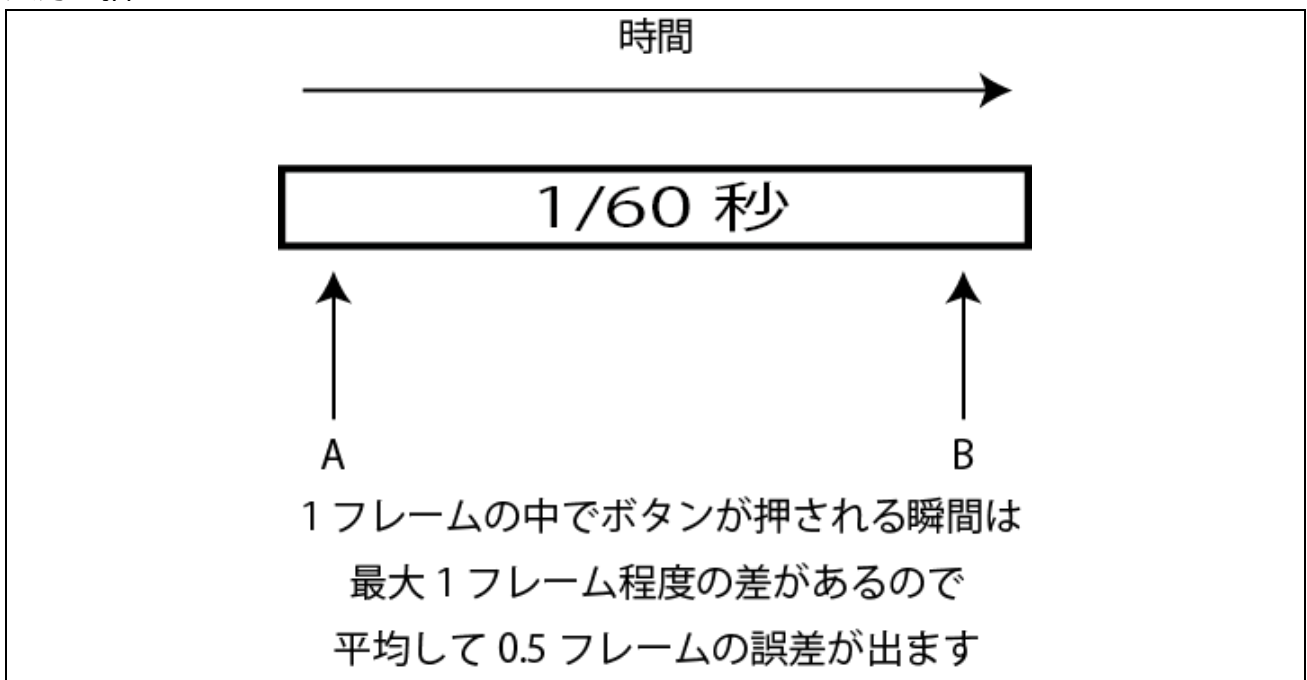


## どのぐらいのフレームレートの撮影が必要か

デジカメ	60fps撮影		デジカメ	120fps撮影		デジカメ	240fps撮影
1	2		1	4		1	6
2	1		2	4		2	8
3	1		3	4		3	8
4	1		4	4		4	6
5	1		5	3		5	7
計	6		計	19		計	35
60fps換算	1.2		60fps換算	1.9		60fps換算	1.75

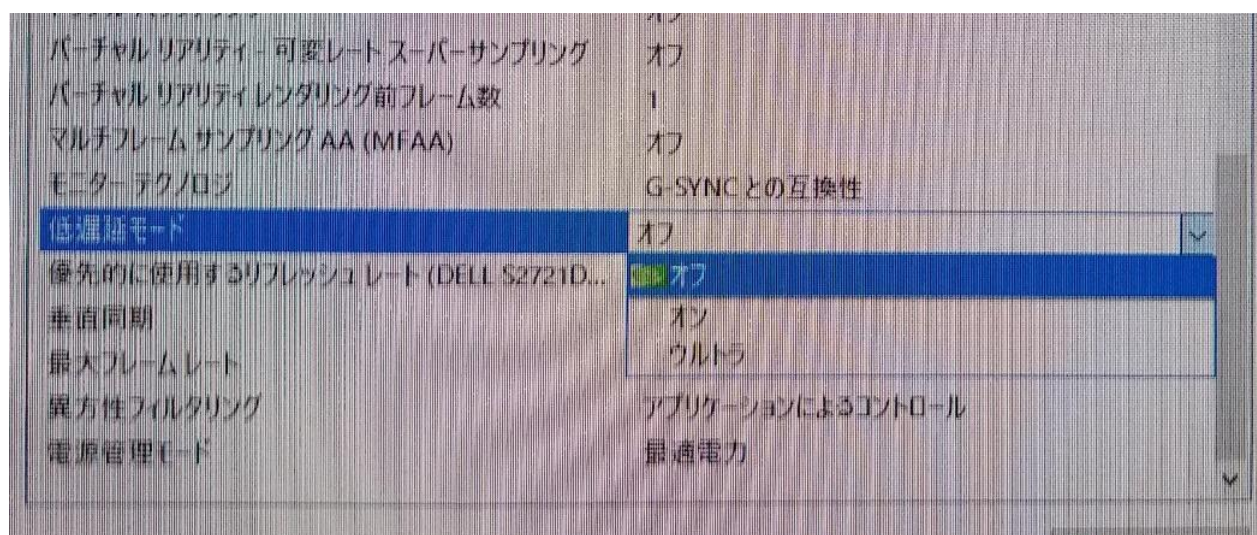
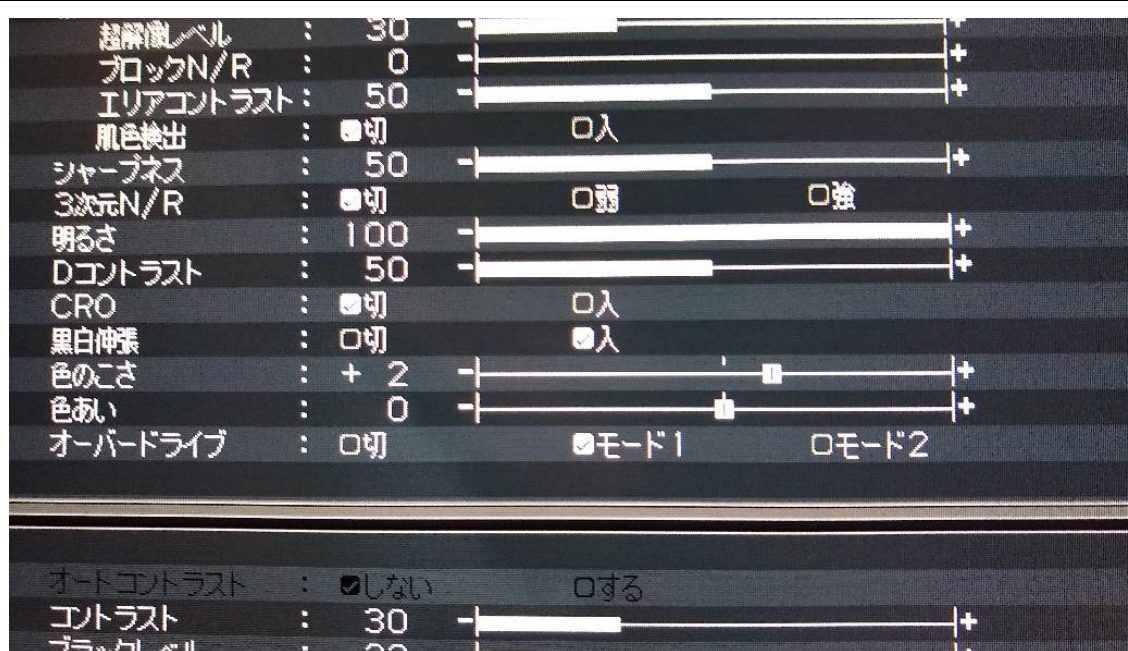
表は同じ条件のゲーム環境をデジタルカメラ動画撮影のフレームレート変えて撮影した結果です。秒間 60 フレーム動作のゲームの測定では最低 60fps での撮影が必要です。経験的に 120fps の撮影であれば十分な精度で測定できます。入力の揺らぎを観測したい場合や 120fps やもっと速いフレームレートのゲーム環境を測定する場合はそれに応じた速度で撮影可能なデジタルカメラ等の機材が必要になります。

## 入力の揺らぎ



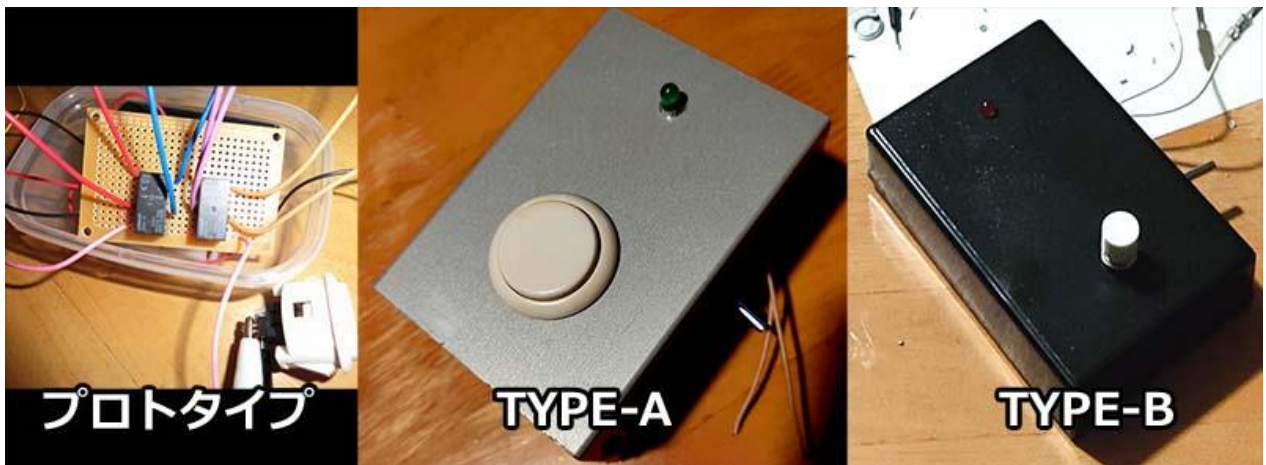
1 フレームの中でボタンが押されるタイミングは図の A と B のように最大 1 フレーム程度の差があります。その差を考慮する必要があるので遅延測定器のボタンを数回押して平均を求めます。ボタンを押す回数を 10 回や 100 回と増やして測定すれば入力の揺らぎの影響を少なくすることが出来ますが、経験的にボタンを押す回数は 5 回で十分です。

## 使用例



モニター設定の各項目を変更し、どの項目が遅延と関係あるかを調べたり、ビデオカード設定の各項目を総当たりで変更してゆき、どの項目が遅延に影響があるのか調べることができます。

アーケードゲームを計測する場合は、テストモードのインプットテストを使うと便利です。



10年ぐらい前、あるゲームを作っている時にターゲットプラットフォームの遅延が大きいことが発覚し、それは、その状態でローンチしてしまうとプレイヤーからの信頼を失ってしまうレベルのもので、関係者と協議してみたが、そういった状態である事を認識していた人はおらず、その時点でローンチまでになにかしらの対策を試せる時間は2ヶ月ぐらいしかなかった。

自社ゲームのマスター提出までの残り時間もあと少し、大切な追込み期間だったけれど、そもそも遅延問題が解決しなければ自社のゲームはリリースしない方が賢明と思っていたので、自社ゲームの開発を止めて解決策を提案する方法を考えた。

まず遅延を計測し定量化する事でプレイした時のフィーリングが悪いとただ言っているだけではない証明をする方法を検討する事になった。

ボタンを押してからその情報が画面に表示されるまでの時間を計測する方法が必要で、ボタン入力と同期してLEDが光る装置があれば実現可能だと考え、その装置を作るために必要そうな部品を秋葉原で買い漁って乏しい電子回路の知識で作ったのがプロトタイプ。あと必要なものは120FPSといった高速動画撮影ができる機材。当時はスマホで高速動画撮影できるものは存在しなかった頃。ビックカメラのデジタルカメラ売り場でそういった機能があるものを探してきた。ゴルフのスイング確認にも使えます！といったタイプのもの。それらを使って該当のターゲットプラットフォーム、ライバル会社の製品や代表的なゲーム機の遅延状況を調べ一覧表を作り、どのぐらい遅延を減らす必要があるのか目標が見えた。

ターゲットプラットフォームの入力と出力、使い方や設定を総当たりで試しボトルネックの特定と改善方法を調べ、この装置を使って原因と改善効果のエビデンス資料を起こしプラットフォームに検証・対応してもらって奇跡的にローンチまでに遅延問題は解決し、数多くのゲームが快適に遊べる環境が構築されました。マスター間近だった自社ゲームへの影響は大きかったけど。

しばらくしてプロトタイプをきれいに作り直して市販のケースに収めたのがTYPE-A。だいぶ長い間使用していました。2~3年前ぐらいに電子回路の知識をアップデートして諸々改良して省電力化とコンパクト化を実現したのがTYPE-B。これも市販のケースを使用していて、やっぱり専用のケースとか作れたらいいよなーなんて思っていたら、そういえば3Dプリンターってそういう事出来るんじゃないかと考えたら楽しくなってきた、試すついでに量産化してしまおう！というのが、今回のユニバーサル遅延測定器です。



## 後日談

遅延対策が施された SDK がリリースされ、それを適用した対戦格闘ゲームのアップデートリリースで「遅延を減らしました」との記載が有り、そういえばせっかく対応してもらったけど、まだ遅れていた自社ゲームの開発が手いっぱい新しい SDK の検証していなかったと思い遅延測定器を使って確認してみたところ、遅延は解消されていなかったの、「え?! 何事!」軽く混乱。良く調べたら資料の記述ミスでそのままコピペしても効果がない事が分かり、それを直ぐに担当者に伝えたところ「ギリギリ間に合いましたありがとうございます!」と。未然に間違った状態でのリリースを回避したことがありました。また、当社よりも前からあるゲームプラットフォームで対戦格闘ゲームを幾つか作っていた開発者と同席する事があって「何であんな状態の遅延を放っておいたんですか?」と訊いたところ、とてもキョトンとされてまるで認識されていなかった様子で。

問題が表出する前に解決。誰も知らないところで地球のピンチを救ったみたいな話。プレイヤーやロケーション、多くのゲーム開発者やプラットフォーム、当社のゲームも快適に遊べるようになって、誰にとってもメリットしかない出来事だったんですが、その結果、遅延を知っておかなければならない立場の人が知らないままでも問題の無い状況を作ってしまったような気がしないでもない。ちょっと複雑な気持ちを抱く今日この頃といった次第。こちらのツイートはそんな気持ちから。



TriangleService  
@GameCenterLove

...

ゲームにおける遅延の理由のほとんどは「測定していない」からです。(たぶん)  
開発中から数値化しておけば「認識」できるので、遅延をどうするか検討する事が可能になります。  
「ユニバーサル遅延測定器」が役に立ちます。

午後6:20 · 2022年5月14日 · Twitter Web App

## ユニバーサル遅延測定器 説明書

2022年4月初版

2022年7月後日談追加

©2022 TriangleService

<https://triangleservice.co.jp>

本書の全部または一部を無断で複写・複製・転載・改ざん・公衆送信をすることを禁じます(著作権法上の例外を除く)。