



REVERSE LAMPE AJNA

Chercheur du Vrai

Table des matières

I. Contexte	2
II. Informations	3
II.1. LUCIA N°3	3
II.2. AJNA	3
II.3. PANDORASTAR	4
III. Brevets	4
IV. Informations techniques et diverses	5
V. AJNA, Reverse-engineering	6
V.1. Chaîne de pilotage de l’AJNA	7
V.2. Notes en vrac	7
V.3. Correspondance appels portail et fonctionnement binaires : test.cgi	8
V.4. Dérive temporelle dans les boucles du PWM :	9
V.5. Conclusions sur l’aspect Software	9
VI. DUMP des patterns AJNA et explication des colonnes	10

I. Contexte

Suite à l'ouverture du sujet « Lumière Hypnagogique » sur le forum ChecheurDuVrai, et à de nombreuse lecture sur le sujet, il semble que l'excitation des capteurs des yeux permettent d'induire des états de conscience modifié.

Ce type de stimuli externe, permettant de guider le cerveau et de le synchroniser sur des états de conscience spécifique, se retrouve également dans le procédé Hemi-Sync de Robert Monroe.

Ainsi, dans le Zome « Bien-être » de Moncaut, endroit dédié à la méditation et à ce type de pratique, les accompagnants ont opté pour l'utilisation des deux procédés (Hemi-Sync + Lucia). Dans une vidéo de Bob vous dit toute la vérité, des témoignages permettent de s'autoriser à penser que ces procédés (notamment lumineux) génèrent bel et bien des états de consciences altérés.

Un utilisateur du forum Chercheur du Vrai, suite à plusieurs sessions autour de la LUCIA et AJNA atteste que la lampe permet d'obtenir des effets, sans que toute fois ces effets ne s'apparente à des sorties du corps ou des voyages « astraux » comme peuvent l'indiquer certains utilisateurs.

Des contacts de Thaïlande ont pu me transférer une partie des données sur l'AJNA afin de comprendre et d'obtenir les données brutes utilisées par la lampe, afin de disposer dans un premier temps d'une source de données brutes, et dans un second temps si le cœur m'en disait, d'ouvrir l'information au plus grand nombre.

II. Informations

II.1. LUCIA N°3

LUCIA est la lampe originale, née de la contribution de Dirk W. Proeckl (neurologist – NDE à 7ans) et Engelbert J. Winkler (clinical psychologist). Le constat d'origine, guidé par des études autour des mystiques de l'ancien temps qui jouaient avec les bougies, est que la lumière vibrante permet de conditionner le cerveau et le faire fonctionner dans des états différents que l'état de veille habituel. On peut supposer qu'avec un Background de medecin et une ouverture d'esprit, ces personnes ont pu mener une vraie étude sur le sujet, tout en recoupant leurs tests avec des graphs Encéphalographie, traçant l'activité du cerveau.

La Lucia est donc, à première vue, le matériel qui sera le plus « propre » et le plus « fidèle » à ce que doit être la technologie de lumière vibrante induisant des états de consciences modifiées. <http://lucialight.eu/purchase-a-lucia-n03>



Lucia N°3, Coût : 20.000\$

II.2. AJNA

L'AJNA est née de l'idée d'un utilisateur de la LUCIA N°3, Guy Harriman, de concevoir un système reprenant le principe de la Lucia, tout en limitant les coûts de la lampe LUCIA (en effet cette dernière est de grande qualité aussi bien dans les composants, que sa conception).

Afin de faire tomber les coûts, G.Harriman s'oriente vers un système fait maison qu'il aura conçu (avec ou sans aide, peu importe). Ainsi l'ordinateur complet et la lampe en Alu brossé laisse place à un RaspBerryPI, une carte de puissance et une boîte embarquant 5 lampes à forte luminosité.



Théorie et science autour de l'AJNA : [Theory and Science of the Ajna Light.pdf](#)



AJNA, Coût : 3333\$

II.3. PANDORASTAR

PANDORASTAR semble être également une copie de la LUCIA, disposant d'une lampe se rapprochant de la LUCIA, et un logiciel pour piloter cette dernière.

Pas beaucoup d'information à son sujet, mais permettrait contrairement à l'AJNA et à la LUCIA de créer et partager des fichiers de sessions entre utilisateur.

D'après « Light Alchimie, Richard Yiap » c'est la lampe qui se rapprocherait le plus de la LUCIA en terme d'effets.



Coût estimé, 6000€

III. Brevets

Light treatment apparatus (LUCIA N°3), US 20120136198 A1 / Dispositif de photothérapie (LUCIA N°3), EP 2437839 B1

La présente invention concerne un dispositif de photothérapie destiné à simuler un état limite psychophysique, doté d'un dispositif émettant une lumière pouvant être reçue par un œil humain et d'un dispositif de commande conçu pour commander le dispositif émettant de la lumière. Selon l'invention, le dispositif émettant de la lumière possède au moins une source de lumière continue et au moins une source de lumière clignotante dont la lumière clignotante est superposable à la lumière continue émise par la source de lumière continue, dans une structure thérapeutique. Le dispositif de commande présente un composant de commande de fréquence permettant, lors d'un cycle d'accélération, d'augmenter ou de réduire la fréquence de la source de lumière clignotante pour passer d'une fréquence de sortie à une fréquence cible.



US20120136198.pdf



EP2437839B1.pdf

Illumination equipment, US5403261

An illumination system comprises a 1st signal generator to generate signals with a frequency range of the alpha rhythms of human brain waves, a 2nd signal generator to generate signals with a power spectrum distribution expressed by $1/f^n$, where n is between 0 and 2, and also with a frequency range below 50 Hz, a driver amplifier a and a driver amplifier b, both serving as light controllers to control the illumination of light sources according to the signals from the 1st and 2nd signal generators respectively and finally a light source 1 and a light source 2 to shed lights, and contributes to inducing a person under the light shedding from it to generate alpha rhythms without having an uncomfortable impression due to flickering illumination in a readily available illumination environment. Thus, the illumination from the light source 1 contributes to the entrainment of alpha rhythms and that from the light source 2 contributes to masking the illumination from the light source 1 eliminating the unpleasant flickering impression with a resultant ready creation of an illumination environment wherein alpha rhythms are induced. If necessary, it is possible to enhance the alpha rhythm entrainment effect by using sounds and also it is possible to apply a bio-feedback for stabilizing and intensifying the alpha rhythms through a brain wave feedback loop formed visually and/or acoustically, if necessary, with the analyzed signals of the brain waves from a person under the illumination environment used as the control signals for the 1st signal generator.



US5403261.pdf

A partir de cette page, les données seront concentrées autour de AJNA, et certaines références seront faite sur la LUCIA. L'AJNA sera en effet le point central de l'étude.

IV. Informations techniques et diverses

Suite à la vidéo https://youtu.be/_VtwYVz55ys informations recueillies :

LUCIA	AJNA
1 low voltage lampe hallogène centrale 8 lampes LED Nécessite un ordinateur pour piloter la lampe Dispose de plus de 100 programmes	5 LED Nécessite un navigateur pour se connecter à la lampe
	Ne produirait que 30% à 50% des effets obtenus par la LUCIA d'après « Light Alchimy, Richard Yiap » revendeur de la LUCIA.

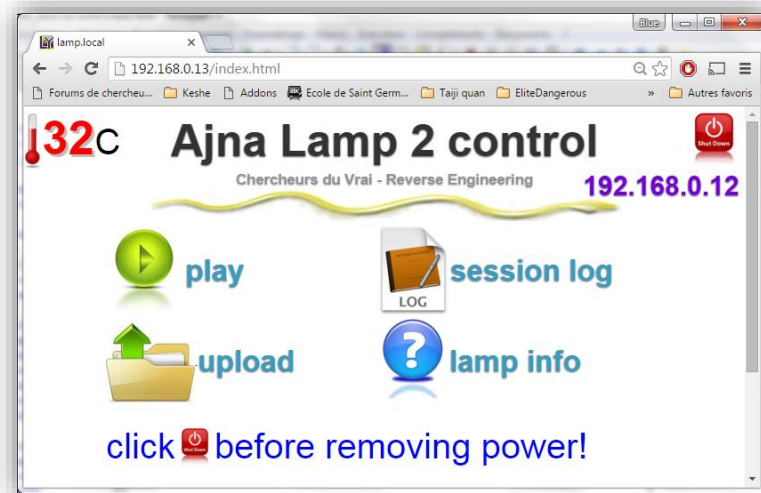
Information recueillies sur le forum « Chercheur du Vrai »

LUCIA	AJNA
utilise des LEDs blanches de forte puissance lumineuse	utilise des LEDs blanches de forte puissance lumineuse
Utilise un message pulsé en fréquence, et gradué en intensité lumineuse	utilise un message pulsé en fréquence, et gradué en intensité lumineuse

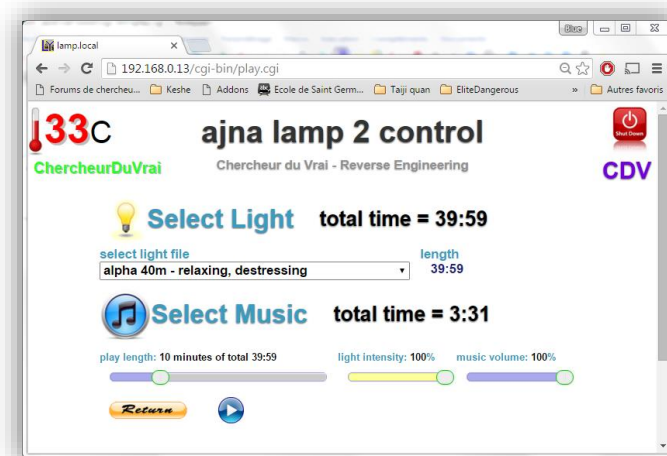
- Au moins 5 LEDs de 5W ou 10W, de couleur blanche (le spectre lumineux doit inclure la plage 470nm - 490nm)
- chaque LED est piloté indépendamment en fréquence et en intensité
- La luminosité générale doit être au moins de 5000 lumens (cf. specs de Ajna, 5 LEDs de 10W/500 lumens).
- La solution utilisée par Lucia est différente, avec un spot halogen central piloté en intensité qui se charge de générer des "lumens", et 8 LEDs de puissance moindre pour le clignotement.

V. AJNA, Reverse-engineering

L'AJNA est constitué d'un RaspBerryPI et d'une carte de puissance permettant de driver les LED. La partie carte de puissance ne fera pas l'objet d'un reverse engineering, et seule la partie software / design / conception sera étudiée, sans pour autant dévoiler l'ensemble des clefs de l'AJNA afin de protéger le travail de l'auteur (Guy Harriman). En effet, l'objectif est de pouvoir répliquer le fonctionnement de l'effet hypnagogique, et non copier la machine en elle-même. En synthèse, le RaspBerryPI est configuré afin de fournir un service http et ne nécessite donc pour l'utilisateur final que d'un navigateur. Le service http fournit l'interface suivante à la connexion :



- Le menu Play permet de sélectionner une session de travail (avec le couple Lumière / Musique)
- Le menu Upload permet de télécharger de nouveaux MP3 ou fichiers AJNA
- Le menu session permet d'avoir la liste des sessions d'utilisation de la lampe
- Lamp info permet d'afficher le menu de test pour tester les différentes LEDS et avoir des informations littéraires



Le menu play sera le menu privilégié pour l'étude car c'est via ce biais que l'utilisateur choisit un fichier AJNA pour la lecture.

Le menu play appelle un script cgi-bin (play.cgi) qui permet l'interface entre le navigateur de l'utilisateur. Ce script cgi a pour objectif de mettre à jour les infos dynamiquement dans le navigateur via la technique AJAX. AJAX permet d'appeler depuis le navigateur du client, d'autres scripts, puis de mettre à jour dynamiquement les informations obtenues dans le navigateur, ce qui évite les sempiternels rechargements de page.

V.1. Chaîne de pilotage de l'AJNA

Utilisateur ⇔ Navigateur ⇔ Script CGI ⇔ Driver de lampe ⇔ RaspBerry (BCM2835 / GPIO) => Carte de puissance => Lampes

Après analyse, le fichier AJNA dispose du paramétrage complet de la session (hors programme iLST) pour la durée définie. Dans les fichiers AJNA, il n'y a aucune randomisation des paramètres : tout est planifié à la seconde, en spécifiant quelles LEDs doivent travailler, à quelle fréquence (globale), etc. Un fichier AJNA est donc une suite d'instruction (codée) qui permet de piloter les LED suivant un schéma précis.

Le script CGI a pour rôle de lire le fichier AJNA, de retravailler les variables (un travail de fond a été fait sur la sécurisation de l'AJNA pour éviter le reverse-engineering et le piratage), puis de transférer chaque instruction au Driver de lampe.

Le Driver de lampe est un programme C compilé, embarquant la librairie BCM2835 des GPIO du RaspBerry. Ce binaire permet notamment de transformer les instructions qu'il reçoit en instructions compatibles avec les GPIO (On/OFF, gestion du temps écoulés, PWM, etc.).

La liaison entre les GPIO (TTL 3.3v à très faible puissance) et les LED à forte luminosité est faite via une carte de puissance compatible avec les fréquences utilisées.

Afin de protéger le travail de l'auteur, la façon de décoder les fichiers AJNA, le fonctionnement exact des scripts CGI, ou les différentes protections mises en place par l'auteur ne seront pas abordées. Pour rappel, seul le paramétrage des sessions (Pattern) sera étudié en détail.

V.2. Notes en vrac

- Le BackgroundIntensity est toujours bloqué à 50, même si la valeur initiale du pattern dépasse 50
- Si flickerIntensity est > 50, alors la variable invertFlickerPWM passe à 1
- Les patterns fonctionnent par seconde entière. Le programme fonctionne à base de BEAT, et 1 BEAT = 500ms
- Chaque BEAT est découpé en 1ms, au travers d'une boucle FOR à 500 increments (boucle PWM)
- La base de temps principale est la seconde, ainsi toutes les fréquences sont calculées à partir de cette base de temps. Le nombre de BEATS est multiplié par deux ce qui permet d'insérer la notion de « périodes » du signal (1Hz sinusoïde c'est une période, soit 500ms positif et 500ms négatif). Ainsi 1 BEATS = 500ms et 1 incrément PWM = 1ms. Un découpage d'un incrément est fait par tranche de 10µs pour pouvoir jouer avec le LightIntensity et le BackgroundIntensity ($100 * 10\mu s = 100\% = 1$ incrément PWM complet).
- Dans le cadre du PWM, des tests conditionnels sont fait :
 - Dans tous les cas, les lampes sont utilisées :
 - Pour le flicker si elles servent au flicker pour la periode en cours
 - Pour le Background si elles ne sont pas utilisées pour le flicker
 - Pour le Background, le calcul est fait vis-à-vis des lampes qui ne sont pas encore utilisées pour le flicker. Ces lampes servent alors au BackGround.
 - En synthèse :
 - Le LightIntensity est gérée non pas avec la puissance de la LED, mais avec le fait qu'elle sera allumée plus ou moins longtemps (impact sur la fréquence à confirmer).
 - Cela implique que pour une fréquence de 12hz, l'impulsion de LED doit se faire toute les 83,3ms. Ce qui signifie qu'un état haut du PWM sera de $83.3ms/2$ et un état base de $83.3ms/2$, soit un temps allumé de 42ms et un temps éteint de 42ms.

- Pour gérer l'intensité du Flickering, Guy Harriman utilise une fraction du temps allumé de la LED, ainsi si pour 12Hz, la lumière doit rester allumée 42ms pour une intensité de 100%, elle ne sera allumée que pendant 21ms si l'intensité est de 50%. Cela ne change rien à la période d'attente de 41ms global pour l'état haut ou bas. Ainsi l'intensité est gérée sur le temps d'allumage de la lampe, et non son intensité.
- Le BackgroundIntensity ne peut être supérieur à 50% puisque ce sont les mêmes LED qui servent à fournir l'effet strobo et la lumière de fond.
 - Le Background peut être placé sur des LED qui ont une intensité lumineuse inférieure au temps de Background. Ainsi, si une LED participe à une fréquence de 12hz, mais que son intensité est de 50%, et que le Background demandé est de 10%, alors elle peut servir aussi au background, car le background est < à l'intensité et la somme est inférieure à 100%
- En aucun cas le BackgroundIntensity ne rallumera une lampe éteinte, ce qui entraînerait inévitablement des dérapages sur la fréquence voulue

Les paramètres principaux à retenir sont donc :

- La fréquence de pulsation
- La puissance de la lumière ambiante
- La puissance de la lumière des LED
- Les LED à allumer ou non

Les améliorations à imaginer sont :

- La séparation du BackgroundLight des LED de flickering
- La gestion de la puissance de la LED via le matériel et non une astuce software

V.3. Correspondance appels portail et fonctionnement binaires : test.cgi

The image shows a web browser window displaying the 'ajna lamp 2 control' interface. The interface includes a header with '33C ChercheurDuVrai' and 'ajna lamp 2 control meditate into light'. It features several sliders: 'flicker frequency: 4 Hz' (set to 4), 'light test time (seconds): 10', 'flicker power: 100%', and 'background power: 20%'. There are also buttons for 'Return' and a play icon. A 'Shutdown' button is visible in the top right. The browser address bar shows '192.168.0.13/cgi-bin/test.cgi'. To the left, a Notepad++ window shows the C++ code for 'test.cgi', with lines 1-30. The code defines parameters and their corresponding event counts. Colored dashed lines connect the code variables to the UI sliders: green for flicker frequency, orange for light test time, blue for flicker power, and purple for background power.

Les valeurs de passage de paramètre correspondent. A noter toutefois que la variable 'lengthplay' vaut le double de ce qui est renseigné sur la page de paramétrage.

V.4. Dérive temporelle dans les boucles du PWM :

	A	B	C	D
34	PWM	Delta	Delta	Delta
35		Timer	Attendu	Obtenu
36	35			
37	36	36484	37000	
38	37	40777	37000	3777
39	38	41045	38000	3045
40	39	41164	39000	2164
41	40	41283	40000	1283
42	41	41400	41000	400
43	42	58780	42000	16780
44	43	59084	43000	16084
45	44	59204	44000	15204
46	45	59320	45000	14320
47	46	59435	46000	13435
48	47	59548	47000	12548
49	48	59664	48000	11664
50	49	59780	49000	10780
51	50	59897	50000	9897
52	51	60012	51000	9012
53	52	60128	52000	8128
54	53	60244	53000	7244
55	54	60360	54000	6360
56	55	60476	55000	5476
57	56	60742	56000	4742
58	57	60860	57000	3860
59	58	60977	58000	2977
60	59	61091	59000	2091
61	60	61207	60000	1207
62	61	61320	61000	320
63	62	63291	62000	1291

On voit donc que des dérives temporelles apparaissent (En rouge), dû probablement au fait que le RaspBerry travaille sur d'autres tâches (architecture applicative et non architecture industrielle). La conception permet d'absorber la dérive temporelle en démarrant plus rapidement le prochain incrément (les chiffres rouges diminuent jusqu'à devenir blanc), mais cela a un impact sur la fluidité et l'exactitude du PWM, et donc du Flicker. L'impact sera d'autant plus visible et ressenti à haute fréquence où il sera impossible d'absorber la dérive temporelle. Un axe d'amélioration serait de déléguer le driving du PWM à une puce dédiée à architecture industrielle et non applicative, style Arduino (AVR, PIC ou autre)

V.5. Conclusions sur l'aspect Software

La lampe AJNA est un joli modèle d'intégration avec son portail WEB, ses scripts CGI et son driveur de lampe. Un très gros effort de protection du software en fait, et il y a eu une réelle étude pour limiter les coûts matériels de la lampe, notamment en travaillant sur l'algorithme qui permet d'utiliser les LED pour le flickering et pour le BeckGround => cette façon de procéder dénote une vraie réflexion sur la réalisation.

Cette version souffre cependant de limitations liées aux choix du design :

- Pas de BackGroundLight : obligation d'utiliser les LED qui servent à pulser la lumière
- Dérive temporelle des fréquences pulsées due à l'architecture du RaspBerry
- Puissance des LED gérée au niveau software et non hardware : obligation d'éteindre plus rapidement les LED au détriment de la fréquence

VI. DUMP des patterns AJNA et explication des colonnes

	A	B	C	D	E	F	G	H	I	J	K	L
1	lengthPlayString	flickerFrequencyString	flickerIntensityString	backgroundIntensityString	flickerLampOneString	flickerLampTwoString	flickerLampThreeString	flickerLampFourString	flickerLampFiveString	turnOffLights	LightIntensity	Timeout
2	10,10,8,9,9,10,7,6	10,12,27,22,8,12,17,11	59,50,46,51,54,48,27,44	14,6,10,14,10,10,9,3	1,0,1,0,1,1,0,0	0,0,0,0,0,1,0,1	1,1,1,0,0,0,0,0	0,0,1,1,1,0,1,1	1,1,1,0,0,1,1,1	1	100	1200
3	10,10,8,9,9,10,7,6	10,12,27,22,8,12,17,11	59,50,46,51,54,48,27,44	14,6,10,14,10,10,9,3	1,0,1,0,1,1,0,0	0,0,0,0,0,1,0,1	1,1,1,0,0,0,0,0	0,0,1,1,1,0,1,1	1,1,1,0,0,1,1,1	1	100	1166
4	8,8,7,7,8,7,10,6	10,11,12,11,22,8,27,8	57,54,25,51,51,61,55,60	6,6,3,8,12,4,11,6	1,1,1,0,0,1,0,1	1,1,1,1,1,0,1,1	1,1,1,0,1,0,0,0	1,1,0,1,0,1,0,1	1,0,0,0,0,1,1,0	1	100	1131
5	8,8,7,7,8,7,10,6	10,11,12,11,22,8,27,8	57,54,25,51,51,61,55,60	6,6,3,8,12,4,11,6	1,1,1,0,0,1,0,1	1,1,1,1,1,0,1,1	1,1,1,0,1,0,0,0	1,1,0,1,0,1,0,1	1,0,0,0,0,1,1,0	1	100	1100
6	7,10,10,10,7,9,10,6	12,12,16,8,22,24,24,9	65,55,45,27,28,47,60,41	11,10,13,11,6,12,5,2	1,0,0,0,0,1,0,1	0,1,1,0,1,0,0,0	0,0,0,1,0,0,1,1	1,1,1,0,0,1,1,1	1,1,1,0,0,0,0,0	1	100	1070
7	8,8,10,6,6,8,8,7	8,25,12,19,11,13,29,10	57,45,50,56,28,57,61,31	12,7,11,10,8,2,5,6	0,1,0,1,0,1,0,1	1,1,0,1,0,1,1,0	1,1,1,1,1,1,1,0	0,1,1,0,0,1,0,0	1,1,0,1,1,1,1,0	1	100	1035

- lengthPlayString : Durée en BEAT de la fréquence à jouer. A diviser par deux pour obtenir les secondes.
- flickerFrequencyString : Fréquence de la lumière (en Hz)
- flickerIntensityString : % d'intensité de la lumière (Attention, valeur relative à l'AJNA*)
- backgroundIntensityString : % d'intensité de la lumière de fond (Attention, valeur relative à l'AJNA*)
- flickerLampOneString : Etat (On/Off) de la lampe 1
- flickerLampTwoString : Etat (On/Off) de la lampe 2
- flickerLampThreeString : Etat (On/Off) de la lampe 3
- flickerLampFourString : Etat (On/Off) de la lampe 4
- flickerLampFiveString : Etat (On/Off) de la lampe 5
- turnOffLights : Non utilisé
- LightIntensity : Intensité lumineuse globale (non utilisé dans le reverse effectué)
- Timeout : Temps restant de la session utilisateur

Attention, valeur relative à l'AJNA* : Cette remarque indique que le mode de fonctionnement de l'AJNA est très spécifique de par l'utilisation des LED pour le Flickering et le Background. Il faut se rappeler que cette valeur correspond à un pourcentage de l'état haut d'un incrément du PWM. Toutefois, ceci peut servir de référence pour concevoir ses propre Pattern en donnant une idée de l'intensité lumineuse utilisée pour une fréquence. Concernant le BackGround, pour considérer la valeur comme un % fonctionnel, il conviendra de multiplier la valeur par 2 puisque cette valeur est conçue comme ne pouvant jamais dépasser 50.

Information à connaître : Les patterns 60minutes, 40minutes et 20minutes ne se ressemblent pas.