

**CUADERNO ABIERTO PARA LA SIMULACIÓN DE CÉLULAS SOLARES DE TRES TERMINALES
DE TIPO TRANSISTOR BIPOLAR DE HETEROUNION**

Martí A.*, **Antolín E.***, **Zehender M.H.***, **Martínez M.***, **Svatek S.A.***,
García-Linares P.*, **Ramiro I.***, **Giliberti G.*****, **Cappelluti F.**** y **Cristóbal A.B.***

* Instituto de Energía Solar, ETSI Telecomunicación, Universidad Politécnica de Madrid, Ciudad Universitaria sn,
28040 Madrid, España. e-mail: antonio.marti@upm.es

** Department of Electronics and Telecommunications, Politecnico di Torino,
Corso Duca degli Abruzzi 24, 10129 Torino, Italy. E-mail: gemma.giliberti@polito.it

<https://doi.org/10.34637/cies2020.1.1001>

RESUMEN

Los cuadernos abiertos (Open Notebooks), como los que pueden realizarse en el entorno Jupyter, son una herramienta excelente, no solo para documentar los programas que se implementan para realizar tal o cual cálculo, sino también para: a) facilitar la docencia sobre el asunto de que se trate, b) facilitar que terceros verifiquen con facilidad los cálculos realizados, c) posibilitar el cálculo interactivo. En el contexto del proyecto Europeo GRECO, dedicado al desarrollo de la ciencia e innovación responsable (RRI) aplicado al campo de la energía solar fotovoltaica, estamos desarrollando un “Open Notebook” para modelar analíticamente la denominada “célula solar de tres terminales de tipo transistor bipolar de heterounión”. En este trabajo describimos cómo acceder a dicho cuaderno, describimos el modelo utilizado para modelar dicha célula y comentamos algunas de las lecciones aprendidas en relación con su uso y el desarrollo de la ciencia abierta.

PALABRAS CLAVE: Ciencia abierta, Cuadernos abiertos, Célula solar de tres terminales de tipo transistor bipolar de heterounión.

ABSTRACT

Open Notebooks, such as those that can be made in the Jupyter environment, are an excellent tool, not only to document the codes that are implemented to perform this or that calculation, but also to: a) facilitate teaching on the subject in question, b) facilitate third parties to easily verify the calculations performed, c) enable interactive calculation. In the context of the European GRECO project, dedicated to the development of responsible research and innovation (RRI) in the field of photovoltaics, we are developing an “Open Notebook” to analytically model the so-called “three terminal heterojunction bipolar transistor solar cell”. In this paper, we describe how to access this Open Notebook, describe the model used to model said cell, and comment on some of the lessons learned in relation to its use and the development of open science.

KEYWORDS: Open Science, Open notebooks, Three Terminal Hetero-junction Bipolar Transistor Solar Cell.

INTRODUCCIÓN

La célula solar de tres terminales de tipo transistor bipolar de heterounión (3T-HBTSC) fue propuesta en (Martí and Luque 2015). Su estructura se muestra en la Fig. 1. Como su nombre indica, se trata de una célula de tres terminales en la que dos de sus terminales se usan para obtener energía eléctrica de la célula superior o “top” (constituida por las capas que en la figura se han etiquetado como p-emisor y n-base) y otros dos terminales para sacar energía eléctrica de la célula inferior o “bottom” (constituida por las capas que en la figura se han etiquetado como n-base y p-colector) siendo uno de los terminales, el denominado de base, común a ambas células. Su nombre, “de tipo transistor” hace referencia a que su estructura semiconductor básica se asemeja a la de un transistor bipolar, pudiendo ser de tipo “pnp” (como la que se ha ilustrado) o “npn”.

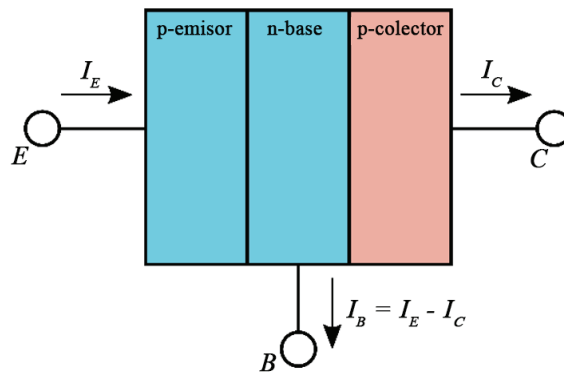


Fig. 1. Estructura de una célula solar de tres terminales de tipo transistor bipolar de heterounión. Las regiones azules indican semiconductores de alto gap y, el color rojo, semiconductores de bajo gap. Las flechas indican el sentido de las corrientes que tomaremos como positivo: cuando el resultado sea negativo, indicará que la corriente fluye en sentido contrario.

La breve descripción que acabamos de hacer servirá a la mayoría de los lectores para captar la idea general relativa al funcionamiento de esta célula. Sin embargo, el investigador interesado en profundizar en la comprensión de su funcionamiento o, incluso, en desarrollarla experimentalmente, pronto se encontrará con la necesidad de modelar su funcionamiento. En esa línea, con el fin de facilitar a terceros esta labor, hemos publicado en abierto un código que modela analíticamente el funcionamiento de esta célula solar. El código, y su documentación, se encuentra disponible en GitHub (Martí et al. 2020) y puede utilizarse bajo “Licencia GNU Public Release v3.0”(Free Software Foundation 2007)”. Se ha implementado en Python 3, en el entorno Jupyter. En las siguientes secciones describiremos brevemente el modelo analítico de la célula 3T-HBTSC implementado, recordaremos dónde pueden obtenerse las herramientas necesarias para obtener y ejecutar el código y, finalmente, describimos nuestras impresiones, desde la perspectiva de la ciencia abierta, relacionadas con la puesta en abierto de este código.

MODELO ANALÍTICO DE LA CÉLULA 3T-HBTSC

La Fig. 2 muestra, de nuevo, la estructura de una célula 3T-HBTSC, a través de su diagrama de bandas en equilibrio, poniendo, esta vez, énfasis en ciertas coordenadas: $-x_{pE}$ y x_{nE} para los extremos de la zona de carga espacial de la unión emisor–base; x_{nC} y x_{pC} para los extremos de la zona de carga espacial de la unión base–colector; 0 y W_B para la posición de la unión metalúrgica entre el emisor y la base y entre la base y el colector respectivamente, y $-W_E$ y W_C para los extremos del emisor y del colector respectivamente. Estas coordenadas serán necesarias para identificar con claridad dónde establecemos las condiciones de contorno necesarias para resolver las ecuaciones diferenciales de nuestro modelo.

Las ecuaciones, unidimensionales en aras de la simplicidad, que formulamos para cada una de las regiones de la célula (emisor, base y colector) se obtienen a partir de la ecuación ambipolar de difusión de portadores en los semiconductores (McKelvey 1966), operando en baja inyección, e incorporando las condiciones de iluminación según un modelo planteado por primera vez, según nuestro conocimiento, por Hovel (Hovel 1975) al modelar células solares de AlGaAs/GaAs. Recordemos que estos modelos no consideran la posibilidad de transporte termiónico a través de las uniones sino, únicamente, la posibilidad de que los portadores puedan transportarse por mecanismos de arrastre y difusión. Así, para la región de emisor planteamos que:

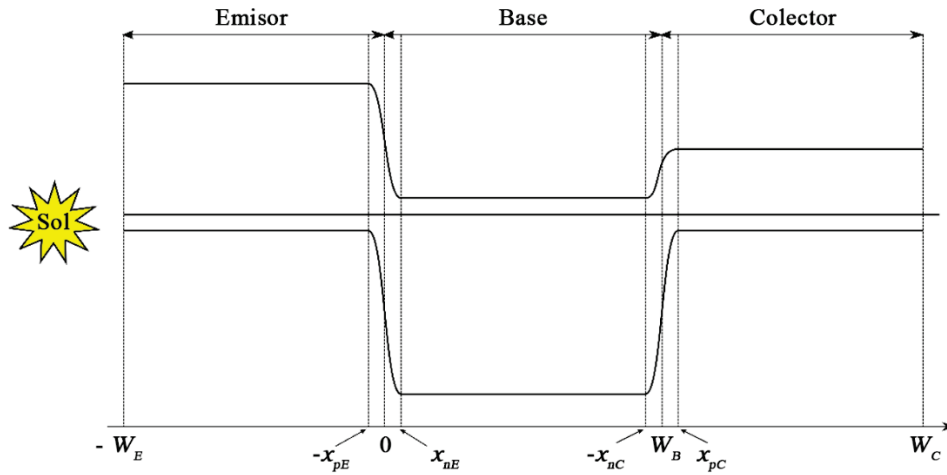


Fig. 2. Diagrama de bandas aproximado de una célula de tipo 3T-HBTSC en equilibrio donde se indican, además, las coordenadas de los extremos de ciertas regiones de interés y que se utilizan para identificar la localización de las condiciones de contorno.

$$D_E \frac{d^2 m_E}{dx^2} - \frac{m_E}{\tau_E} = -g_E(x, \lambda) \quad (1a)$$

$$D_E \frac{dm_E}{dx}(-W_E) = S_E m_E(-W_E) \quad (1b)$$

$$m_E(-x_{pE}) = \frac{n_{iE}^2}{N_E} \exp\left(\frac{eV_{EB}}{kT} - 1\right) \quad (1c)$$

donde: m_E es el exceso de concentración de portadores minoritarios en el emisor sobre el equilibrio; D_E es la constante de difusión de los portadores minoritarios del emisor; τ_E es el tiempo de vida de los portadores minoritarios del emisor; S_E es la velocidad de recombinación superficial de los portadores minoritarios en la superficie del emisor; e es la carga del electrón en valor absoluto; V_{EB} es la tensión directa de polarización de la unión emisor–base (que tomara un valor negativo si la unión se polariza en inversa); k es la constante de Boltzmann; T es la temperatura absoluta de operación de la célula (que se supone uniforme en toda la célula); n_{iE} es la concentración intrínseca del semiconductor con el que se haya fabricado el emisor; N_E es la concentración neta de impurezas ionizadas en el emisor; $g_E(x, \lambda)$ es la generación de pares electrón–hueco que se produce en el emisor, a consecuencia de la iluminación solar, en la posición x para fotones de longitud de onda λ . Asumiendo que no inciden en el emisor fotones debido a una reflexión posterior, esta generación viene dada por:

$$g_E(x, \lambda) = \alpha_E F_0(\lambda) \exp[-\alpha_E(\lambda) \cdot (x + W_E)] d\lambda \quad (2)$$

donde: $F_0(\lambda)$ es la densidad de flujo espectral de fotones incidente en la cara frontal de la célula (fotones por unidad de área, tiempo y longitud de onda) una vez descontadas las pérdidas por reflectividad y factor de sombra, y $\alpha_E(\lambda)$ es el coeficiente de absorción del material que constituye el emisor (descontando absorciones, como las intrabanda, que no generen pares electrón–hueco). Por otro lado, la formulación de las Eqs. (1) en términos de exceso de portadores minoritarios, sin distinguir si se trata de electrones o de huecos, permite formular estas ecuaciones con independencia de que la célula sea de tipo pnp ó npn. Aplicando el principio de superposición, m_E se puede expresar como:

$$m_E = m_{E,D} + m_{E,L} \quad (3)$$

donde $m_{E,D}$ satisface la ecuación diferencial:

$$D_E \frac{d^2 m_{E,D}}{dx^2} - \frac{m_{E,D}}{\tau_E} = 0 \quad (4a)$$

$$D_E \frac{dm_{E,D}}{dx}(-W_E) = S_E m_{E,D}(-W_E) \quad (4b)$$

$$m_{E,D}(-x_{pE}) = \frac{n_{iE}^2}{N_E} \exp\left(\frac{eV_{EB}}{kT} - 1\right) \quad (4c)$$

y es por ello denominado “exceso de portadores minoritarios en el emisor en oscuridad” y $m_{E,L}$ satisface:

$$D_E \frac{d^2 m_{E,L}}{d^2 x} - \frac{m_{E,L}}{\tau_E} = -g_E(x, \lambda) \quad (5a)$$

$$D_E \frac{dm_{E,L}}{dx} (-W_E) = S_E m_{E,L} (-W_E) \quad (5b)$$

$$m_{E,L}(-x_{pE}) = 0 \quad (5c)$$

y, es por ello conocido como “exceso de portadores minoritarios en el emisor en iluminación y en cortocircuito”.

Para la región de base planteamos, para el exceso de minoritarios en la base m_B , una ecuación similar a las Ecs. (1) si bien ahora deberemos tener en cuenta que esta región está comprendida entre dos uniones para formular así que:

$$D_B \frac{d^2 m_B}{d^2 x} - \frac{m_B}{\tau_B} = -g_B(x, \lambda) \quad (6a)$$

$$m_B(x_{nE}) = \frac{n_{iB}^2}{N_B} \exp\left(\frac{eV_{EB}}{kT} - 1\right) \quad (6b)$$

$$m_B(x_{nC}) = \frac{n_{iB}^2}{N_B} \exp\left(\frac{eV_{CB}}{kT} - 1\right) \quad (6c)$$

donde: D_B es la constante de difusión de los portadores minoritarios de la base; τ_B es el tiempo de vida de los portadores minoritarios de la base; V_{CB} es la tensión directa de polarización de la unión colector–base; n_{iB} es la concentración intrínseca del semiconductor que constituye la base; N_B es la concentración neta de impurezas ionizadas en la base y $g_B(x, \lambda)$ es la generación de pares electrón–hueco producidas en la base en la coordenada x a consecuencia de la iluminación solar y que, para fotones de longitud de onda λ viene dada por:

$$g_B(x, \lambda) = \alpha_B F_0(\lambda) \exp[-\alpha_E(\lambda) \cdot W_E] \exp[-\alpha_B(\lambda) \cdot x_{nE}] \exp[-\alpha_B(\lambda) \cdot (x - x_{nE})] d\lambda \quad (7)$$

donde $\alpha_B(\lambda)$ es el coeficiente de absorción del material que constituye la base. El exceso de minoritarios m_B puede nuevamente descomponerse en dos componentes, correspondiente a los excesos de portadores en oscuridad y de iluminación en cortocircuito, de modo que:

$$m_B = m_{B,D} + m_{B,L} \quad (8)$$

y donde $m_{B,D}$ y $m_{B,L}$ satisfacen, respectivamente, las ecuaciones diferenciales:

$$D_B \frac{d^2 m_{B,D}}{d^2 x} - \frac{m_{B,D}}{\tau_B} = 0 \quad (9a)$$

$$m_{B,D}(x_{nE}) = \frac{n_{iB}^2}{N_B} \exp\left(\frac{eV_{EB}}{kT} - 1\right) \quad (9b)$$

$$m_{B,D}(x_{nC}) = \frac{n_{iB}^2}{N_B} \exp\left(\frac{eV_{CB}}{kT} - 1\right) \quad (9c)$$

y

$$D_B \frac{d^2 m_{B,L}}{d^2 x} - \frac{m_{B,L}}{\tau_B} = -g_B(x, \lambda) \quad (10a)$$

$$m_{B,L}(x_{nE}) = 0 \quad (10b)$$

$$m_{B,L}(x_{nC}) = 0 \quad (10c)$$

Finalmente, el exceso de portadores minoritarios en el colector, m_C , se puede calcular a partir de la ecuación diferencial:

$$D_C \frac{d^2 m_C}{d^2 x} - \frac{m_C}{\tau_C} = -g_C(x, \lambda) \quad (11a)$$

$$-D_C \frac{dm_C}{dx} (W_C) = S_C m_C (W_C) \quad (11b)$$

$$m_c(x_{pc}) = \frac{n_{ic}^2}{N_c} \exp\left(\frac{eV_{CB}}{kT} - 1\right) \quad (11c)$$

donde: D_c es la constante de difusión de los portadores minoritarios en el colector; τ_c es el tiempo de vida de los portadores minoritarios en el colector; n_{ic} es la concentración intrínseca del semiconductor que constituye el colector; S_c es la velocidad de recombinación superficial en la superficie exterior del colector; N_c es la concentración neta de impurezas ionizadas en el colector y $g_c(x, \lambda)$ es la generación de pares electrón–hueco que se generan en el colector a consecuencia de la iluminación solar y que, para fotones de longitud de onda λ viene dada por:

$$g_c(x, \lambda) = \alpha_c F_0(\lambda) \exp[-\alpha_E(\lambda) \cdot W_E] \exp[-\alpha_B(\lambda) \cdot W_B] \times \exp[-\alpha_c(\lambda) \cdot (x_{pc} - W_B)] \exp[-\alpha_c(\lambda) \cdot (x - x_{pc})] d\lambda \quad (12)$$

Como en el caso de las otras regiones, m_c puede separarse en una componente de oscuridad, $m_{c,D}$, y otra de iluminación, $m_{c,L}$, tal que:

$$m_c = m_{c,D} + m_{c,L} \quad (13)$$

que satisfacen, respectivamente las siguientes ecuaciones diferenciales:

$$D_c \frac{d^2 m_{c,D}}{dx^2} - \frac{m_{c,D}}{\tau_c} = 0 \quad (14a)$$

$$-D_c \frac{dm_{c,D}}{dx}(W_c) = S_c m_{c,D}(W_c) \quad (14b)$$

$$m_{c,D}(x_{pc}) = \frac{n_{ic}^2}{N_c} \exp\left(\frac{eV_{CB}}{kT} - 1\right) \quad (14c)$$

$$D_c \frac{d^2 m_{c,L}}{dx^2} - \frac{m_{c,L}}{\tau_c} = -g_c(x, \lambda) \quad (15a)$$

$$-D_c \frac{dm_{c,L}}{dx}(W_c) = S_c m_{c,L}(W_c) \quad (15b)$$

$$m_c(x_{pc}) = 0 \quad (15c)$$

Todas las ecuaciones anteriores son ecuaciones diferenciales lineales con coeficientes constantes que, por lo tanto, tienen solución analítica. No obstante, escribir aquí su solución resultaría farragoso e innecesario ya que el código que hemos implementado en Python no las necesita porque utiliza la librería SymPy para resolverlas simbólicamente. Así pues, daremos las soluciones para los excesos de minoritarios $m_{E,D}$, $m_{E,L}$, $m_{B,D}$, $m_{B,L}$, $m_{c,D}$ y $m_{c,L}$ por conocidas y expresaremos a continuación, a partir de ellas, la característica corriente–tensión correspondiente a los terminales de emisor y colector. Así, para la corriente de emisor, I_E , expresada en amperios, si A corresponde al área de la unión base–emisor, que supondremos igual al área de la unión base–colector, podemos escribir:

$$\frac{I_E}{A} = J_E = J_{EE} + J_{EB} - J_{L,EE} - J_{L,EB} \quad (16a)$$

donde

$$J_{EE} = \pm e D_E \frac{dm_{E,D}}{dx}(-x_{pE}) \quad (17b)$$

$$J_{EB} = \pm e D_B \frac{dm_{B,D}}{dx}(-x_{nE}) \quad (17c)$$

son conocidas normalmente como las componentes de oscuridad. En estas ecuaciones, y en las similares que seguirán, el signo superior (positivo en este caso) debe tomarse para el caso en el que el exceso de portadores minoritarios corresponda a electrones y el signo inferior para al caso en el que corresponda a huecos. Por otro lado (nota: $d\lambda$ no falta en la siguiente ecuación y similares porque está incluida en la expresión analítica de la derivada del perfil de minoritarios, habiéndose heredado de su inclusión en el término de generación),

$$-J_{L,EE} = \int_{\lambda} \pm e D_E \frac{dm_{E,L}}{dx}(-x_{pE}) \quad (16d)$$

$$-J_{L,EB} = \int_{\lambda} \pm e D_B \frac{dm_{B,L}}{dx}(x_{nE}) \quad (16e)$$

se conocen normalmente como “corrientes fotogeneradas” siendo normalmente expresadas en términos de “eficiencias cuánticas” como:

$$J_{L,EE} = e \int_{\lambda} Q E_E(\lambda) F_0(\lambda) d\lambda \quad (16f)$$

$$J_{L,EB} = e \int_{\lambda} Q E_{BE}(\lambda) F_0(\lambda) d\lambda \quad (16g)$$

siendo

$$Q E_E(\lambda) = \mp D_E \frac{dm_{E,L}}{dx}(-x_{pE}) / (F_0(\lambda) d\lambda) \quad (16h)$$

$$Q E_{BE}(\lambda) = \mp D_B \frac{dm_{B,L}}{dx}(x_{nE}) / (F_0(\lambda) d\lambda) \quad (16i)$$

las eficiencias cuánticas que denominaremos “de emisor” y “de la base en la unión emisor” respectivamente. Por otro lado, de forma similar, para la corriente de colector tenemos que:

$$\frac{I_C}{A} = J_C = J_{CC} + J_{CB} - J_{L,CC} - J_{L,CB} \quad (17a)$$

donde

$$J_{CC} = \pm e D_C \frac{dm_{C,D}}{dx}(x_{pC}) \quad (17b)$$

$$J_{CB} = \pm e D_B \frac{dm_{B,D}}{dx}(x_{nC}) \quad (17c)$$

$$J_{L,CC} = e \int_{\lambda} Q E_C(\lambda) F_0(\lambda) d\lambda \quad (17d)$$

$$J_{L,CB} = e \int_{\lambda} Q E_{BC}(\lambda) F_0(\lambda) d\lambda \quad (17e)$$

siendo

$$Q E_C(\lambda) = \mp D_C \frac{dm_{C,L}}{dx}(x_{pC}) / (F_0(\lambda) d\lambda) \quad (17f)$$

$$Q E_{BE}(\lambda) = \mp D_B \frac{dm_{B,L}}{dx}(x_{nC}) / (F_0(\lambda) d\lambda) \quad (17g)$$

las que denominaremos eficiencia cuántica “de colector” y “de la base en la unión colector” respectivamente. Como el lector habrá notado, las ecuaciones anteriores consideran que en las zonas de carga de espacio no se produce ni generación ni recombinación de pares electrón–hueco.

Finalmente, la corriente de base viene dada simplemente por:

$$I_B = I_E - I_C \quad (18)$$

REQUISITOS PARA LA UTILIZACION DEL CÓDIGO

Nuestro código permite calcular las corrientes de oscuridad e iluminación anteriores a partir de los datos estructurales de la célula (como espesores de las distintas regiones y dopajes) y parámetros de minoritarios en las distintas regiones que constituye la células (tiempo de vida, constante de difusión, etc.). El código se ha implementado en “Python 3” (Python 2019) habiendo sido “Jupyter” (Project Jupyter 2017) el entorno en el que hemos programado en este lenguaje. Jupyter nos permite desarrollar el código también en forma de cuaderno abierto (Open Notebook) ofreciéndonos, además, una gran facilidad para documentar nuestro código al permitimos incluir figuras, ecuaciones,

etc., por lo que una descripción detallada aquí resultaría redundante e innecesaria para el lector no interesado. A su vez, nosotros hemos instalado Python y Jupyter utilizando el entorno “Anaconda” (Anaconda 2018), que favorece el control y coherencia de las distintas subrutinas que pueden utilizarse en Python.

Como avanzamos en la introducción, nuestro código puede descargarse de la plataforma GitHub (Martí et al. 2020). No obstante, conviene aclarar algunos detalles que, a nosotros, como novatos en la utilización de estas herramientas, nos resultó de utilidad conocer. Así, conviene saber que “Git” es un software de control de versiones. Explicaremos en qué consiste la idea de “control de versiones” con un ejemplo. La mayoría de nosotros, cuando creamos una nueva versión de un documento, lo salvamos con un nuevo nombre en nuestro ordenador, algo así como “versión_01”, “versión_02”, etc., ya que, por ejemplo, tememos que podamos cambiar de idea y necesitar en el futuro algún fragmento de la versión 01 que habríamos eliminado en la versión 02. La mayoría de las veces, incluso cuando concluimos la versión definitiva, solo los más disciplinados borran las versiones anteriores de sus ordenadores. Procediendo de esta manera, si el documento final ocupa, digamos 10 MB y conservamos 10 versiones anteriores, nuestro ordenador bien podría acabar consumiendo 100 MB de espacio en su unidad de almacenamiento cuando, en realidad, solo necesitaría 10 MB. Una de las cosas realmente inteligentes que hace “Git” es que, cuando salvamos una nueva versión de nuestro archivo, se las ingenia para almacenar solo los cambios reales, de forma transparente para el usuario. Por ejemplo, si cambiamos una sola coma en nuestro documento de 10 MB, salvarlo de nuevo como una nueva versión al modo tradicional, consumirá otros 10 MB, mientras que haciéndolo con el sistema de control de versiones Git, consumirá únicamente unos pocos bytes más ya que Git solo almacena la parte modificada y guarda el control sobre los mismos. La segunda cosa inteligente que hace “Git” es que nos permite una especie de viaje en el tiempo que consiste en permitirnos recuperar nuestro documento tal y como era, por ejemplo, hace dos meses. “Git” (Git 2017) se encuentra disponible tanto para usuarios de GNU-Linux como de Mac y Windows.

“GitHub” (Git 2020) es un sistema de almacenamiento en la nube, que incorpora el control de versiones “Git”, en el que, además, podemos almacenar nuestros códigos y hacerlo disponible para terceros así como facilitar el acceso a nuestros colaboradores. “Github” es gratuito mientras no se colabore con más de 3 personas. Pero colaborar con tantas personas como se desee solo cuesta unos 4 \$ al mes.

“GitHub Desktop” (Desktop GitHub 2017) es un entorno gráfico, de ventanas, para manejar “GitHub” con comodidad. Desgraciadamente solo está disponible para usuarios de Mac y Windows. A pesar de ello, los usuarios de GNU-Linux disponemos de herramientas como “Git Cola” (Git-Cola 2020) para gestionar gráficamente sus ficheros “Git”.

CONCLUSIONES RELACIONADAS CON LA UTILIZACION DE “OPEN-NOTEBOOKS”

Con este trabajo, además de proporcionar una herramienta que pudiese facilitar la comprensión del funcionamiento de la 3T-HBTSC a aquellos lectores interesados, también perseguíamos evaluar los factores a favor y en contra respecto a la implementación en código abierto y libre de nuestras herramientas de desarrollo en entornos de tipo “Open Notebook”. Así, hemos aprendido que:

- Abrir nuestro código en un entorno colaborativo como GitHub puede ser un proceso doloroso para principiantes: algunos tendremos miedo de mostrar al mundo nuestras pobres habilidades programadoras. Por otro lado, sentir que cualquiera puede revisar nuestro código nos hace esforzarnos en hacerlo y documentarlo mejor.
- El primer grupo de personas con las que Ud. querrá trabajar y probar son sus colaboradores más cercanos. Para que la colaboración sea efectiva, todos ellos tendrán que aprender las herramientas que hemos descrito en el apartado anterior y es posible que no todos ellos estén motivados para hacerlo o aprecien sus ventajas. La mayoría sentirá la presión de hacer otras cosas más urgentes (y seguramente así es). Esté preparado para dedicarle un tiempo extra: primero aprenda Ud. y luego, enséñeles.
- No es suficiente con adquirir un conocimiento superficial para poderlas utilizar de forma efectiva. Se necesita un nivel avanzado lo que incrementa el tiempo necesario para dominarlas.
- Por ello recomendamos que las herramientas de trabajo colaborativas y las herramientas informáticas necesarias para la implementación de la ciencia abierta se comiencen a enseñar ya a nivel de graduado. Más tarde, cuando los investigadores ya han alcanzado sus hábitos de trabajo, será muy difícil cambiarlos. De forma alternativa o complementaria, se deberían dar cursos intensivos de formación para formar adecuadamente al personal en las habilidades informáticas necesarias.
- Tomar la iniciativa y abrir el código a nivel individual es un paso, pero la iniciativa debería tomarse a nivel institucional con el fin de proporcionar un entorno uniforme en la adopción las plataformas de desarrollo necesarias.
- Para el uso eficiente de plataformas como GitHub se necesita el pago de una cuota mensual. Algunos investigadores se muestran reticentes al pago de esta cuota para iniciar el proceso de abrir colaborativamente su código porque temen que no recuperarán esta cuota de sus Instituciones bajo el argumento de que ha sido

usada para fines personales, o la dificultad de cargar estas cuotas a diferentes proyectos o, simplemente, el excesivo papeleo que es necesario para recuperar esta cuota. Quizá sería recomendable la creación de una Plataforma Europea Abierta, similar a GitHub, que cubra de por vida el desarrollo del código abierto que se desarrolle, al menos en el seno de Proyectos Europeos.

- La dispersión de sistemas operativos (GNU/Linux, Windows, Mac...) añade una complejidad adicional ya que puede que no todas las herramientas necesarias estén desarrolladas para todos los sistemas operativos. Por ejemplo, "GitHub Desktop" solo funciona en Mac y Windows lo que impide que los usuarios de GNU/Linux puedan usarlo, lo cual no deja de ser paradójico, ya que la mayoría de las herramientas para el desarrollo de código abierto, entre ellas Git, parecen haber sido promovidas por desarrolladores de GNU/Linux.
- Abrir su código no implica necesariamente que el "software" necesario para ejecutarlo sea también abierto. Por ejemplo, cuando Ud. crea un documento escrito en Microsoft Word para su libre distribución no significa que Microsoft Word esté disponible en código abierto. Para dar un paso más, el "software" necesario para ejecutarlo también debería ser abierto (por ejemplo, Ud. debería usar LibreOffice en vez de Microsoft Word). Por esta razón hemos elegido Python y Jupyter para nuestro desarrollo. Sin embargo, como hemos dicho, aprenderlas le supondrá un esfuerzo adicional que retrasará el desarrollo de su aplicación si esa es su prioridad.

DESCARGO DE RESPONSABILIDAD

En este documento, el género masculino, en frases como "el lector..." se refiere tanto al género femenino como al masculino y, su utilización, en esa frase u otras similares empleadas en este documento, se ha elegido para dar fluidez al texto y en ningún momento con la intención de discriminar por género. Se ha hecho todo lo posible por escribir aquí las ecuaciones libres de errores. No obstante, en caso de conflicto, tienen prevalencia las ecuaciones de nuestro código en GitHub. Este documento contiene extractos de nuestro borrador de "deliverable" *Open Software and Manual for designing novel solar cells*, que será enviado a la Comisión Europea dentro del Proyecto GRECO.

AGRADECIMIENTOS

Este trabajo ha sido posible gracias a los proyectos GRECO (787289 H2020 Grant Agreement), financiado por la Comisión Europea, que garantiza que este trabajo está alineado con la Política Europea de Ciencia Abierta, el Proyecto MADRID-PV2-CM (P2018/EMT-4308) financiado por la Comunidad de Madrid apoyado con fondos FEDER y el Proyecto MULTIPLIER (RTI2018-096937-B-C21) financiado por el Ministerio de Ciencia, Innovación y Universidades. E.A. está financiada por un contrato Ramón y Cajal del Ministerio de Ciencia Español (RYC-2015-18539). M.H.Z. agradece a la Universidad Politécnica de Madrid la financiación del 'Programa Propio para Ayudas Predoctorales de la UPM'. M.M. agradece la financiación del Ministerio de Ciencia, Innovación y Universidades por medio de la beca FPI PRE2019-087894.

REFERENCIAS

- Anaconda. 2018. "Anaconda Documentation." <https://docs.anaconda.com/>.
- Desktop GitHub. 2017. "GitHub Desktop | Simple Collaboration from Your Desktop." 2017. <https://desktop.github.com/>.
- Free Software Foundation. 2007. "The GNU General Public License v3.0." 2007. <https://www.gnu.org/licenses/gpl-3.0.en.html>.
- Git-Cola. 2020. "Git Cola: The Highly Caffeinated Git GUI." 2020. <https://git-cola.github.io/>.
- Git. 2017. "Git - Downloads." Git-Scm.Com. 2017. <https://git-scm.com/downloads>.
- Github. 2020. "The World's Leading Software Development Platform · GitHub." Github.Com. 2020. <https://github.com/>.
- Hovel, H.J. 1975. *Solar Cells*. Edited by R. K. Willardson and A. C. Beer. New York: Academic Press.
- Martí, A., and A. Luque. 2015. "Three-Terminal Heterojunction Bipolar Transistor Solar Cell for High-Efficiency Photovoltaic Conversion." *Nature Communications* 6 (April): 6902–6902. <https://doi.org/10.1038/ncomms7902>.

Martí, A, E Antolín, M Zehender, and I Ramiro. 2020. “Modelling the 3T- Heterojunction Bipolar Transistor Solar Cell in GitHub.” https://Github.Com/Amartiv/3_Terminal_Transistor_Solar_Cell.

Mckelvey, J P. 1966. *Solid State and Semiconductor Phycis*. Berkeley: Harper & Row Publishers.

Project Jupyter. 2017. “Project Jupyter | Home.” 2017. <https://jupyter.org/>.

Python. 2019. “Python 3.0 Release | Python.Org.” 2019. <https://www.python.org/download/releases/3.0/>.