Theorem: The "oracle definition" of $\Sigma_k$, $\Pi_k^p$ (and $\Sigma_k$, $\Pi_k$) coincides with the earlier "logic definition" for all $k$.

Proof. The utility of the logic definition compared to the text is that it already encapsulates some of the inductive steps, thus simplifying what remains to be done. The rest is to show for $k \geq 2$:

$$N^p \Sigma_{k-1}^p = \Pi_{k-1}^p$$

(using the logic definition)

First, containment $\subseteq$ is quick. Let $A \in \Sigma_k^p$ be given. Then by the inductive "logic definition" of $\Sigma_k^p$, there are a polynomial $q(n)$ and a predicate $R(x,y)$ decidable in $\Pi_{k-1}^p$ such that for all $x$, $y \in A$

$$(\exists z : |z| \leq q(|x|)) R(x,z)$$

Now design a polynomial-time $\text{NORM} N$ to accept $A$ with oracle $R$ as follows: On any input $x$, $N^R$ guesses a string of length up to $q(|x|)$ and accepts if the oracle answers "yes" to the query $(x,z)$. Then $A \in \text{LE}(N^R)$, so $A \in \Sigma_k^p$, so $A \in \Pi_k^p$. [Also this idea works to show $\Sigma_k \subseteq \text{RE} \Pi_{k-1}^p$ without the polynomial $q(n)$ bounding the length.]

The other direction, getting $N^p \Pi_{k-1}^p \subseteq \Sigma_k^p$, is harder. Let any $A$ in the left-hand class be given. Then there is an oracle $B \in \Pi_{k-1}^p$ such that $A \in \Sigma_k^p$. We recurse on $B$—that's the trick. By the inductive definition (via logic) for $\Pi_{k-1}^p$ there exists a predicate $R(u,z)$ with language in $\Sigma_{k-2}^p$ such that for all strings $u$, $u \in B \iff (\exists z : |z| \leq q(|u|)) R(u,z)$. We need $k \geq 2$ to do this. What about $k = 1$? Well then $\Pi_0^p = P$, so we have $N^p \subseteq \Sigma_1^p = N^p$. While we already know. So $k = 1$ works as another "base case."
To finish “unpacking” the meaning of $A \in \text{NP}^B$, this means we can take a polynomial time bound $N$-Turing machine $N$ such that $A = L(N^B)$. Our goal is to show $A \in \Sigma^p_k$ by writing a $\Sigma^p_k$ logical definition of “$x \in A$” using polynomial length-bounded quantifiers. We have, for all $x$,

$$x \in A \iff N^B \text{ accepts } x \iff \text{there exists an accepting computation}$$

(which must have at most $p(|x|)$ steps).

What is “an accepting oracle computation”? It is like a regular computation but includes also the queries that were made and their answers from the oracle. It doesn’t have to include any more information about the oracle, but apart from the string $\tilde{C}$ denoting the computation, we need to verify that the answers are correct. That’s where we use $R(u,v)$ for $B$. Given any $\tilde{C}$, let $u_1, \ldots, u_r$ stand for the queries that $\tilde{C}$ records as having been answered “yes,” and let $v_1, \ldots, v_s$ stand for the queries listed as “no.” Then we “roll”:

$$x \in A \iff (\exists \tilde{C} : |\tilde{C}| \leq p(1|X|))$$

$$\left[ \tilde{C} \text{ is an accepting computation that lists strings} u_1, \ldots, u_r \text{ as being answered “yes” and } v_1, \ldots, v_s \text{ as being answered “no”, and these answers are correct for oracle } B \right]$$

$$\iff (\exists \tilde{C} \exists \tilde{C} \exists \tilde{C}) \left[ \tilde{C} \text{ is an accepting computation} \right]$$

$$\left[ \bigwedge_{1 \leq i \leq r} \left[ V_{i, :} = 1 \iff u_i \in \{1, \ldots, 7\} \right] \right] \land \left[ \bigwedge_{1 \leq j \leq s} \left[ \overline{V}_{j, :} = 1 \iff v_j \in \{1, \ldots, 7\} \right] \right]$$

Now $R(u, \cdot, \cdot)$ has a $\Sigma^p_k$ definition — and $\overline{R}(v, \cdot, \cdot)$ has a $\Pi^p_k$ definition. The latter begins with “$\forall$,” but that $\forall$ gets “absorbed” into the “$(\exists \cdot)(\exists \cdot)(\exists \cdot)$” quantifier block and so doesn’t cause a new alternation. Thus we have overall a $(\exists \cdot)(\forall \cdot)\Sigma^p_k$ pattern, which equates to a $\Sigma^p_k$ pattern. This shows $A \in \Sigma^p_k$, so $NP^{\Pi^p_k} \subseteq \Sigma^p_k$ have equal.